

ETUDE DU COUVERT FORESTIER À PARTIR D'UN PROCESSUS OBJET

GUILLAUME PERRIN, PROMO 2003

Encadrement : XAVIER DESCOMBES Chargé de Recherche
JOSIANE ZERUBIA Directrice de Recherche

Entreprise : INRIA Sophia Antipolis
Projet Ariana (CNRS/INRIA/UNSA)



Résumé

Ce stage s'est déroulé à l'**INRIA**¹ Sophia Antipolis, dans le projet **Ariana**, du mois de mai au mois de novembre 2003.

L'objectif de ce stage est d'extraire et de caractériser les différents arbres à l'aide d'images de télédétection. La méthodologie préconisée repose sur les **processus ponctuels marqués**. La notion d'objet est prépondérante puisqu'un des buts de ce travail est de compter les arbres. De plus, les paramètres des objets permettront de caractériser, voire de reconnaître, les différents arbres.

D'une part, nous définirons un modèle géométrique adapté à la reconnaissance des houppiers, puis nous nous attacherons à définir un **algorithme MCMC**² à sauts réversibles pour l'optimiser. Cet algorithme est itératif, puisqu'à chaque itération une perturbation aléatoire de la configuration courante est proposée. La nouvelle configuration obtenue est acceptée avec une probabilité qui dépend du modèle, évalué pour la configuration courante et pour la nouvelle configuration, ainsi que de la loi de proposition. Les différentes perturbations proposées consistent typiquement en l'ajout ou la suppression (naissance ou mort) d'un objet dans la configuration courante, ou encore en la modification d'un de ces objets.

D'autre part, nous définirons un estimateur et un algorithme convergeant vers cet estimateur afin d'extraire les houppiers à partir d'une scène. Les statistiques de la configuration finale obtenue permettront de récupérer des informations précieuses sur le couvert forestier comme la densité de boisement, la répartition spatiale des arbres ou encore la distribution de leur taille, paramètres qui sont aujourd'hui évalués par les organismes forestiers suite à de longues investigations sur le terrain.

Au cours de ce stage, nous avons étudié des **images de peupleraies** fournies par l'**IFN**³. Par la suite, au cours d'un travail de thèse, des images comportant plusieurs espèces d'arbres, et ayant été obtenues avec différentes conditions d'illumination, de prise de vue, et sur divers types de terrains, seront abordées.

Mots clés :

Processus ponctuel, configuration d'objets, chaîne de Markov, algorithme MCMC à sauts réversibles, recuit simulé, analyse bayésienne, maximum a posteriori, télédétection, houppiers.



1. Institut National de Recherche en Informatique et en Automatique
2. Markov Chain Monte Carlo
3. Inventaire Forestier National

Remerciements

Bien entendu, mes premiers remerciements s'adressent à l'ensemble des membres du projet Ariana, pour m'avoir accueilli au cours de ce stage et m'avoir fait découvrir l'univers de la recherche publique. En particulier, je tiens à remercier Xavier Descombes et Josiane Zerubia, qui bien plus qu'avoir encadré ce travail scientifique, ont toujours été à mon écoute et m'ont placé dans un environnement technique et humain idéal. Je remercie aussi Caroline Lacoste et Mathias Ortner, tous deux doctorants dans l'équipe, pour leurs conseils avisés et le temps qu'ils m'ont consacré.

D'autre part, je souhaiterais exprimer toute ma gratitude à Jean-Guy Boureau, de l'IFN, et à Michel Deshayes, du Cemagref, pour leur disponibilité et leur collaboration.

Enfin, je remercie Arezou Modaressi, ma correspondante à l'Ecole Centrale, Christian Saguez et Paul-Henry Cournède, du laboratoire Mathématiques Appliquées aux Systèmes, et Philippe de Reffye, du Cirad, pour leur soutien et le vif intérêt qu'ils ont porté à ce travail. Souhaitons qu'une collaboration fructueuse s'établisse entre le projet Ariana et le laboratoire MAS, au cours des années de doctorat qui vont succéder à ce stage.

★★★

Table des matières

Résumé	3
Remerciements	5
Introduction	9
1 Etat de l'art en détection de houppiers	11
1.1 Les différentes approches existantes	12
1.2 L'approche contour	13
1.2.1 Le principe	13
1.2.2 Les images de données	13
1.2.3 Etapes de prétraitement	14
1.2.4 Principe du suivi de vallées	14
1.2.5 Règles d'obtention des contours: le "rule-based system"	18
1.2.6 Résultats et interprétations	20
1.3 L'approche objets	22
1.3.1 Le principe	22
1.3.2 Récapitulatif de l'algorithme	23
1.4 Conclusion	24
2 Processus objets: définitions et simulation	25
2.1 Les processus objets: définitions et exemples	26
2.1.1 Première approche des processus ponctuels	26
2.1.2 Définition d'un processus ponctuel par une densité non normalisée	30
2.1.3 Processus ponctuels de Markov	33
2.2 Algorithme de simulation: Metropolis-Hastings-Green	38
2.2.1 Présentation de l'algorithme	38
2.2.2 Convergence de la chaîne de Markov considérée	41
2.2.3 Quelques pistes à explorer	47
3 Notre modèle pour la détection des houppiers	49
3.1 Analyse des données: définition des objets d'intérêt	50
3.1.1 Des sources d'images THR	50
3.1.2 Objectifs fixés	51
3.1.3 Objets d'intérêt	54
3.2 Modèle de la loi objectif	57
3.2.1 Cadre Bayésien	57
3.2.2 Terme a priori $h_p(\mathbf{x})$	58
3.2.3 Terme d'attaches aux données $\mathcal{L}(\mathcal{I} \mathbf{x})$	64
3.2.4 Stabilité du processus	67
3.2.5 Estimateurs	73
3.3 Noyau de propositions adopté	76
3.3.1 Mouvements simples	76
3.3.2 Mouvements complexes	81
3.3.3 Noyau de propositions de fusion/division	81

3.4	Bilan : notre algorithme	92
4	Résultats, Interprétations des résultats et Perspectives	93
4.1	Images étudiées	94
4.2	Résultats et Interprétations	97
4.2.1	Choix des paramètres	97
4.2.2	Résultats obtenus sans le noyau de fusion/division	98
4.2.3	Résultats obtenus avec le noyau de fusion/division	103
4.3	Comparaison avec le suivi de vallées	107
4.3.1	Résultats obtenus avec le suivi de vallées	107
4.3.2	Tableau comparatif	107
4.4	Perspectives	109
	Conclusion	113
A	Le projet Ariana, INRIA	117
A.1	Les axes de recherche du projet Ariana	118
A.2	L'organisation interne du projet	119
B	Chaînes de Markov et Théorie ergodique	121
B.1	Notions sur les chaînes de Markov	122
B.1.1	Premières définitions	122
B.1.2	Stationnarité et Invariance	123
B.1.3	Réversibilité	123
B.1.4	Irréductibilité	124
B.1.5	Ensembles petits et Apériodicité	125
B.2	Etude de la stabilité des chaînes de Markov	126
B.2.1	Ergodicité	126
B.2.2	Récurrence au sens de Harris	126
B.2.3	Théorème limite	127
C	Recueil de démonstrations	129
C.1	Réversibilité de la chaîne dans l'algorithme de Green	130
C.2	Irréductibilité et Ensembles petits	132
C.3	Ergodicité géométrique de l'algorithme de Green	133
C.4	Propriété sur $n_{\sim_I}(\mathbf{x})$	134
C.5	Ergodicité géométrique de l'algorithme A	136

Introduction

Cadre de travail

Ce stage de fin d'études de l'Ecole Centrale Paris s'est déroulé au sein du **projet Ariana**, à l'**INRIA** Sophia-Antipolis. Créé en tant que projet commun CNRS⁴/INRIA/UNSA⁵ en 1998, Ariana regroupe son activité autour de l'étude des problèmes inverses en observation de la Terre, par des approches variationnelles ou probabilistes.

Notre travail s'inscrit dans le cadre d'un **projet COLORS**⁶ en collaboration avec l'**IFN** et le **Cemagref**, tous deux basés à Montpellier.

Besoins en géoréférencement

De nombreuses données sur les milieux forestiers sont nécessaires aux organismes de gestion des forêts (ONF⁷,...) ou chargés d'en effectuer l'inventaire (IFN). Ces besoins comprennent aussi bien des besoins généraux de gestion des forêts, que des besoins plus particuliers comme celui de cartographier les types de peuplements en forêts de montagne par exemple.

En raison des progrès technologiques réalisés en matière de résolution des images (aériennes ou satellitaires), et de la baisse du coût d'obtention de telles images, **de nouvelles perspectives s'ouvrent dans la connaissance des arbres et des forêts.**

Ces données pourraient en effet servir à évaluer certains paramètres de la biodiversité en fournissant des informations plus précises sur les arbres et leur situation (isolés, en rangées, en bosquets ou en peuplement) dans les paysages européens, et à mieux évaluer le reboisement se produisant dans les secteurs sujets à l'abandon de terre, anciens pâturages ou terres agricoles moins productives en zones défavorisées, ou péri-urbaines.

Dans ce contexte, un outil pour aider les forestiers, les gestionnaires du territoire et les écologues du paysage dans l'inventaire des arbres isolés, en alignement, en bosquet ou en forêt, serait le bienvenu. Si cet outil pouvait servir à estimer d'autres paramètres, tels que la densité, le taux de couvert, la taille et le diamètre des arbres, jusqu'à présent renseignés uniquement par des travaux sur le terrain, cela s'avérerait fortement utile du point de vue économique.

Objectifs de notre étude

L'objectif de ce stage est de parvenir à une identification semi-automatique des **houppiers**⁸ sur des images aériennes de peuplements forestiers.

4. Centre National de Recherche Scientifique

5. Université de Nice Sophia Antipolis

6. COopération LOcale de Recherche Scientifique

7. Office National des Forêts

8. Partie supérieure d'un arbre, comprenant l'ensemble des branches et les feuilles

Cet objectif doit être atteint sur différentes espèces d'arbres (bouleaux, chênes, pins, sapins, épicéas, ...), avec différentes formes de couronne, des conditions variables d'illumination, de pente et d'exposition.

Une nouvelle approche

L'approche que nous avons choisi d'adopter repose sur les *processus ponctuels marqués*. Leur utilisation en traitement d'images a été proposée par A. J. Baddeley et M.N.M. van Lieshout dans [35]. Notre objectif est de pouvoir l'adapter à des images réelles.

Les processus ponctuels marqués semblent être particulièrement bien adaptés à notre problème pour plusieurs raisons. Tout d'abord, les objets mis en jeu par ces processus peuvent facilement être nos objets d'intérêt, c'est à dire des arbres. D'autre part, l'approche est sur-pixélique: on se détache totalement de l'image pour se rapprocher du modèle de réalité physique. Enfin, les méthodes mises en jeu sont stochastiques, ce qui permet d'utiliser une panoplie d'outils classiques dans les problèmes inverses.

Contenu du rapport

Aucune information sur la répartition spatiale et les interactions entre houppiers n'est prise en compte dans le cadre des approches usuelles d'analyse. La première partie de ce rapport sera consacrée à un état de l'art des méthodes de détection de houppiers existantes.

Le deuxième chapitre présentera le cadre mathématique des processus ponctuels, ainsi que les éléments essentiels de l'algorithme de simulation de ces processus employé.

Puis, nous décrirons le modèle que nous avons utilisé dans le cadre de notre application au traitement des images: la détection des houppiers.

Enfin, nos résultats, obtenus sur des images de peupleraies fournies par l'IFN, seront exposés, discutés et comparés avec les méthodes classiques dans une dernière partie.

Etat de l'art en détection de houppiers



Différentes méthodes de détection de houppiers ont fait leur apparition ces dernières années. Deux grandes écoles en sont principalement à l'origine, toutes deux appartenant à des pays où il existe un réel besoin dans le domaine de la gestion des forêts: le Canada, et les pays scandinaves.

Il convient de distinguer entre toutes ces méthodes celles qui s'appliquent à des images sans *MNE*¹, ce qui sera notre cas, de celles qui utilisent des MNE. Les MNE permettent en effet d'obtenir l'altitude de chaque pixel de l'image, et donnent de bons résultats avec certains algorithmes peu performants sur des images classiques. Ces MNE sont issus d'un couple d'images stéréo, mais représentent un surcoût dissuasif pour la plupart des industriels.

Dans cette partie, nous limiterons notre état de l'art aux techniques de détection des houppiers n'utilisant pas de MNE, puisque c'est dans ce cadre que nous avons travaillé.



1. Modèle Numérique d'Élévation

1.1. Les différentes approches existantes

Même si aucune méthode n'est totalement satisfaisante à ce jour en ce qui concerne la détection de houppiers, un certain nombre de travaux fournissent un cadre préliminaire.

Un premier type de méthodes consiste à détecter les maxima locaux de l'image. Un algorithme de suivi de vallées permet alors de déterminer les zones d'influence de ces maxima, chaque zone étant assimilée à un houppier. Ce genre de technique délimite donc les contours des arbres.

Pour des vignes, ou d'autres types de plantations présentant un alignement régulier, le comptage est aussi réalisable en travaillant sur la transformée de Fourier de l'image. Ce type d'approche est développé au Cemagref.

D'autres méthodes fournissent de bons résultats pour les arbres situés proches du point Nadir², en estimant les positions de ces arbres à l'aide d'un lissage de l'image par convolution avec un masque Gaussien, suivi d'une recherche de maxima locaux. Seules les positions du point le plus lumineux des tiges d'arbres ne sont alors disponibles.

Pour détecter les arbres éloignés du point Nadir, de bons résultats sont obtenus à l'aide d'un modèle optique de l'arbre. Il consiste à faire coïncider au mieux sur l'image ces modèles optiques ("template matching"), puis d'en déduire les positions des sommets des arbres, en trouvant les maxima de la fonction de corrélation ainsi calculée.

On peut distinguer enfin les approches fondées sur une analyse de la texture. Dans les milieux denses, ces approches ont tendance à fusionner les houppiers proches les uns des autres. En outre, la précision géométrique de la délimitation des houppiers reste faible.

Nous allons présenter dans cette partie deux de ces techniques de détection, qui nous ont paru représentatives des deux grands familles d'approches existantes : l'approche contour avec le suivi de vallées, ainsi que l'approche objet, avec le "template matching".

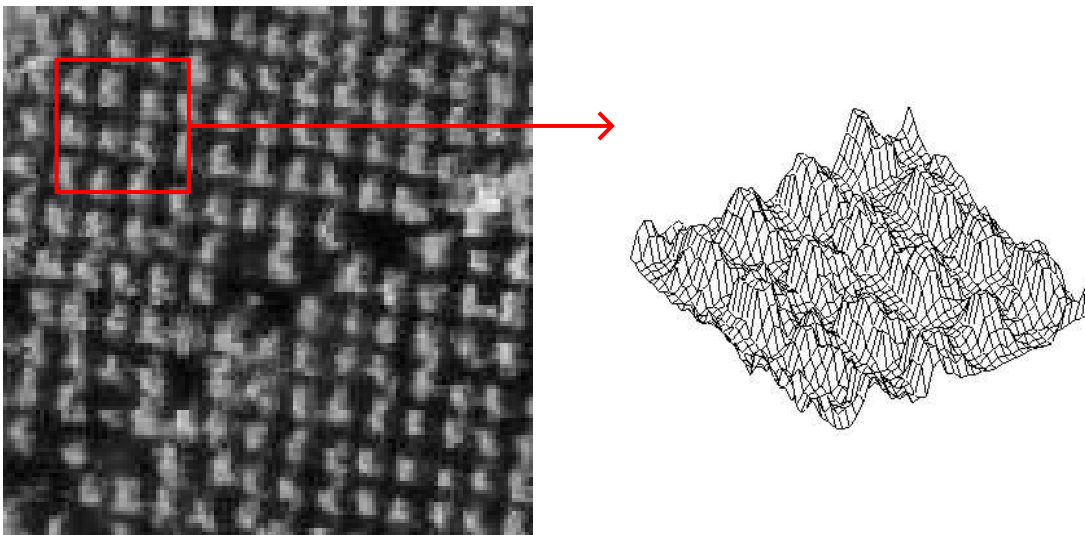


FIG. 1.1 – *A gauche : image de données ©IFN. A droite : extrait de sa carte topographique 3D, la troisième dimension étant les niveaux de gris.*

2. Projeté orthogonal de la caméra sur le sol

1.2. L'approche contour

Dans cette partie, nous présentons une méthode de délimitation automatique des couronnes d'arbres, proposée à l'origine dans [5]. Largement développée au Pacific Forestry Centre (Victoria, British Columbia, Canada), celle-ci fait partie d'un programme plus large de délimitation et de classification par espèces d'arbres. Aussi, nous retrouvons dans d'autres publications du même laboratoire des éléments sur ce sujet pouvant intéresser le lecteur. Celui-ci se reportera aux travaux [6], [7], [8], [9], [10], [18], [19], [20], mais aussi à [26], mémoire de thèse réalisé à Copenhague, au sein de la Royal Veterinary and Agricultural University.

1.2.1 Le principe

La base de cette approche réside dans l'analogie qui existe avec la topographie : l'idée générale est de se représenter l'image en niveaux de gris en trois dimensions, la troisième étant la valeur du niveau de gris en chaque pixel. L'image apparaît alors comme un paysage, où les arbres, plus lumineux et donc avec des niveaux de gris élevés, sont les *montagnes*, et le reste de l'image, les *vallées*. Une des étapes de l'algorithme consiste à propager ces vallées afin de délimiter les contours des montagnes. Il est à noter que, lors d'une détection manuelle des arbres, l'oeil humain réagit d'une façon similaire à celle utilisée par l'algorithme de suivi des vallées : ce sont en effet les fines bandes d'ombre entre les arbres qui lui permettent de les détecter.

L'algorithme proposé consiste, dans un premier temps, à isoler les couronnes entre elles et à les séparer de la végétation environnante, grâce à un programme de suivi de vallées³. Puis, dans un second temps, il s'intéresse aux contours de ces couronnes, obtenus en appliquant un certain nombre de règles prédéfinies⁴.

Nous verrons par la suite qu'un prétraitement de l'image est indispensable pour obtenir de bons résultats.

1.2.2 Les images de données

Les images sur lesquelles les auteurs ont effectué leurs tests étaient des images MEIS⁵ [4]. Celles-ci font une meilleure utilisation du spectre des couleurs que les photographies aériennes classiques, acquérant jusqu'à 8 bandes spectrales, et sont directement reliées à des coordonnées géographiques, facilitant par là-même l'interprétation des experts. Leur résolution peut être moyenne (1 à 10 mètres par pixel) ou haute (0.1 à 1 mètre par pixel) selon les besoins.

Cependant, une seule image en niveaux de gris est nécessaire pour l'algorithme de suivi de vallées. Les résolutions des images de tests furent 31 cm/pixel et 70 cm/pixel principalement, mais d'autres résolutions moins fines furent aussi étudiées afin de comparer les résultats obtenus.

Enfin, le choix du type de peuplement d'arbres est problématique, certains étant plus facilement détectables que d'autres. Les premiers résultats furent obtenus sur des peuplements de conifères.

Afin de mettre en évidence les étapes de l'algorithme de suivi des vallées, nous l'avons implanté et testé sur un extrait d'une image de peupliers de l'IFN.

3. *valley-following algorithm*

4. *rule-based system*

5. Multispectral Electro-optical Imaging Sensor

1.2.3 Etapes de prétraitement

L'image de données, que nous noterons \mathcal{I} par la suite, doit subir un prétraitement avant d'être injectée dans l'algorithme de suivi de vallées. Il convient tout d'abord de choisir une bande spectrale de l'image aérienne, avant de la convertir en image de niveaux de gris. Nous rappelons qu'une hypothèse de départ stipule que les arbres doivent être plus lumineux, en terme de niveaux de gris, que le reste de l'image. C'est pourquoi, et cela se généralise dans la plupart des images forestières, la bande spectrale choisie est le **proche infra-rouge**. Cette bande est en effet sensible aux variations d'illumination et obtient de bonnes réponses de la part des matières végétales chlorophylliennes. Les arbres sans feuilles répondant moins bien, il faut adapter la bande de fréquence lorsque l'on souhaite les étudier, et choisir une longueur d'onde dans le visible (de préférence). Remarquons qu'il est aussi envisageable de combiner les informations comprises dans plusieurs bandes de fréquences.

Après l'obtention de notre image sur la bande de fréquence désirée, celle-ci est lissée afin d'y ôter le bruit à la fois dû à la discrétisation et au capteur. Ce lissage nous permet de faire disparaître de nombreuses erreurs de mesure, qui peuvent se révéler très gênantes notamment si elles sont situées dans les vallées, car elles empêcheraient celles-ci de se rejoindre (voir plus bas le principe du suivi de vallées). Un filtre moyenneur de largeur $s > 0$ convient pour lisser \mathcal{I} . Les coefficients $(f_{i,j})$ de ce filtre, dans une matrice de taille $(2s + 1) \times (2s + 1)$ sont

$$(f_{i,j})_{ij} = \frac{1}{(2s + 1)^2}. \quad (1.1)$$

Enfin, un dernier traitement consiste à isoler la partie qui nous intéresse dans \mathcal{I} , à savoir la forêt, ceci pouvant s'effectuer manuellement par l'utilisateur, ou automatiquement par un processus de classification adéquat. L'objectif de cette dernière étape du prétraitement est de ne garder que les parties de l'image dans lesquelles les arbres sont séparés par une fine bande d'ombre, car ce sont ceux-la que l'on pourra délimiter. On peut par exemple appliquer un masque sur le reste de l'image.

1.2.4 Principe du suivi de vallées

Tout d'abord, nous commençons par chercher dans l'image \mathcal{I} des pixels qui n'appartiennent pas aux couronnes des arbres, faisant partie de zones très ombragées (les vallées). Ces pixels sont supposés avoir un niveau de gris inférieur à un seuil θ , que l'on peut trouver en regardant l'histogramme de l'image. Tous les pixels concernés par ce seuillage sont marqués dans une image binaire \mathcal{J} comme appartenant à la classe des vallées, l'autre classe étant bien entendu celle des arbres. Là encore, le choix du seuillage détermine les résultats que l'on obtiendra. Pour trouver la valeur de θ optimale, nous pourrions envisager une méthode automatique, elle est pour l'instant obtenue par l'expérience, en testant sur les images différentes valeurs.

L'idéal serait en fait d'employer à la place de ce seuillage uniforme un seuillage adaptatif où θ serait dépendant de la position dans l'image. En effet, selon l'inclinaison du terrain et la position du soleil par rapport à la caméra, certaines zones risquent d'être mieux éclairées que d'autres et le seuillage uniforme en sera d'autant moins efficace. Les auteurs mentionnent ce fait, mais se limitent dans une première approche à un seuillage uniforme.

On a donc simplement :

$$\mathcal{I}_{i,j} < \theta \Rightarrow \mathcal{J}_{i,j} = 1. \quad (1.2)$$

Puis, un second passage sur les pixels de l'image qui n'ont pas été marqués recherche les minima locaux, en comparant la valeur en niveau de gris de chaque pixel avec celles de ses 8 voisins (ie

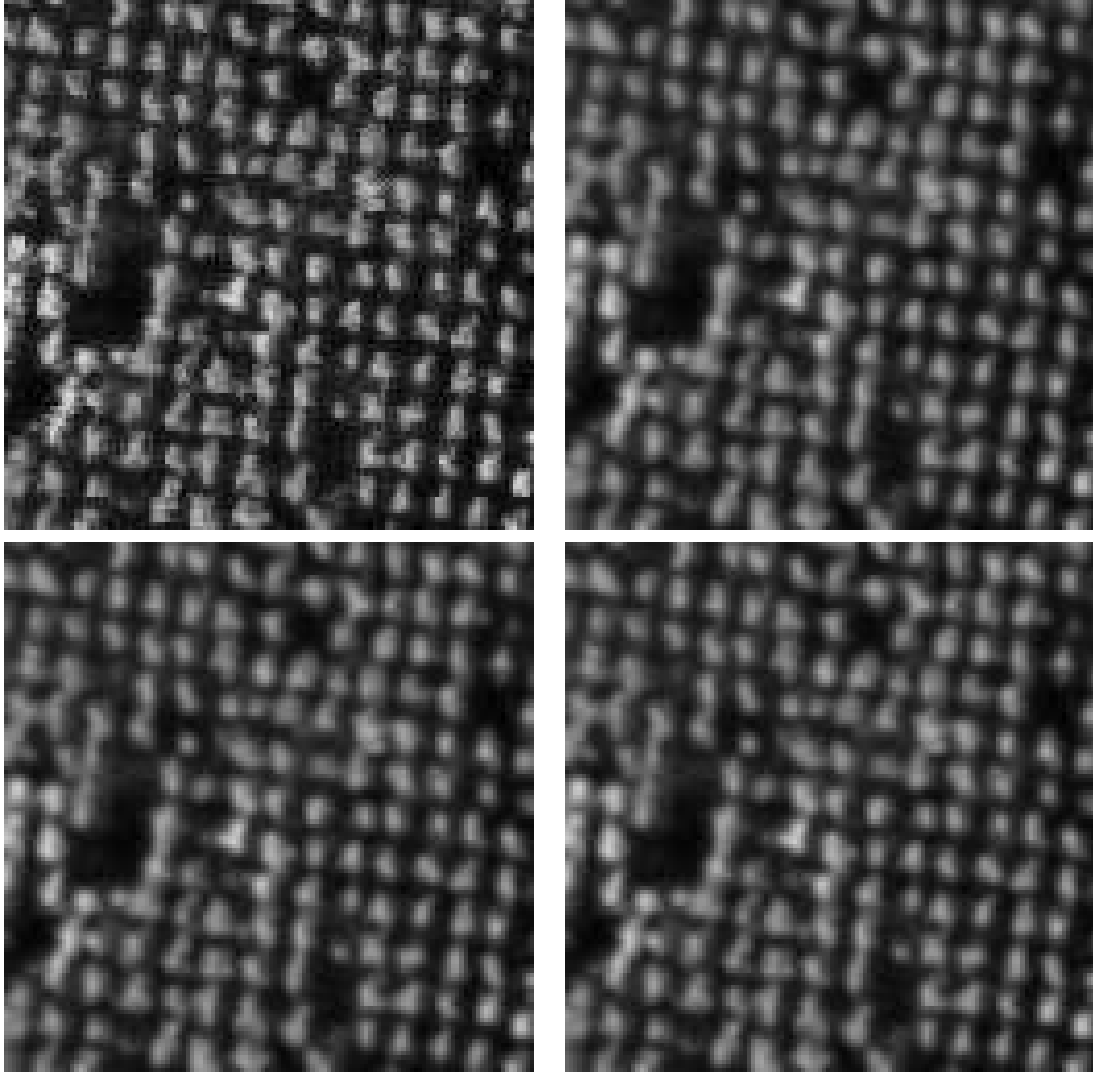


FIG. 1.2 – *Lissage de l'image. En haut à gauche : $I \odot IFN$. En haut à droite : filtrage avec une fenêtre moyenne de largeur 1. En bas à gauche : filtrage avec une fenêtre moyenne de largeur 3. En bas à droite : filtrage avec une fenêtre moyenne de largeur 5.*

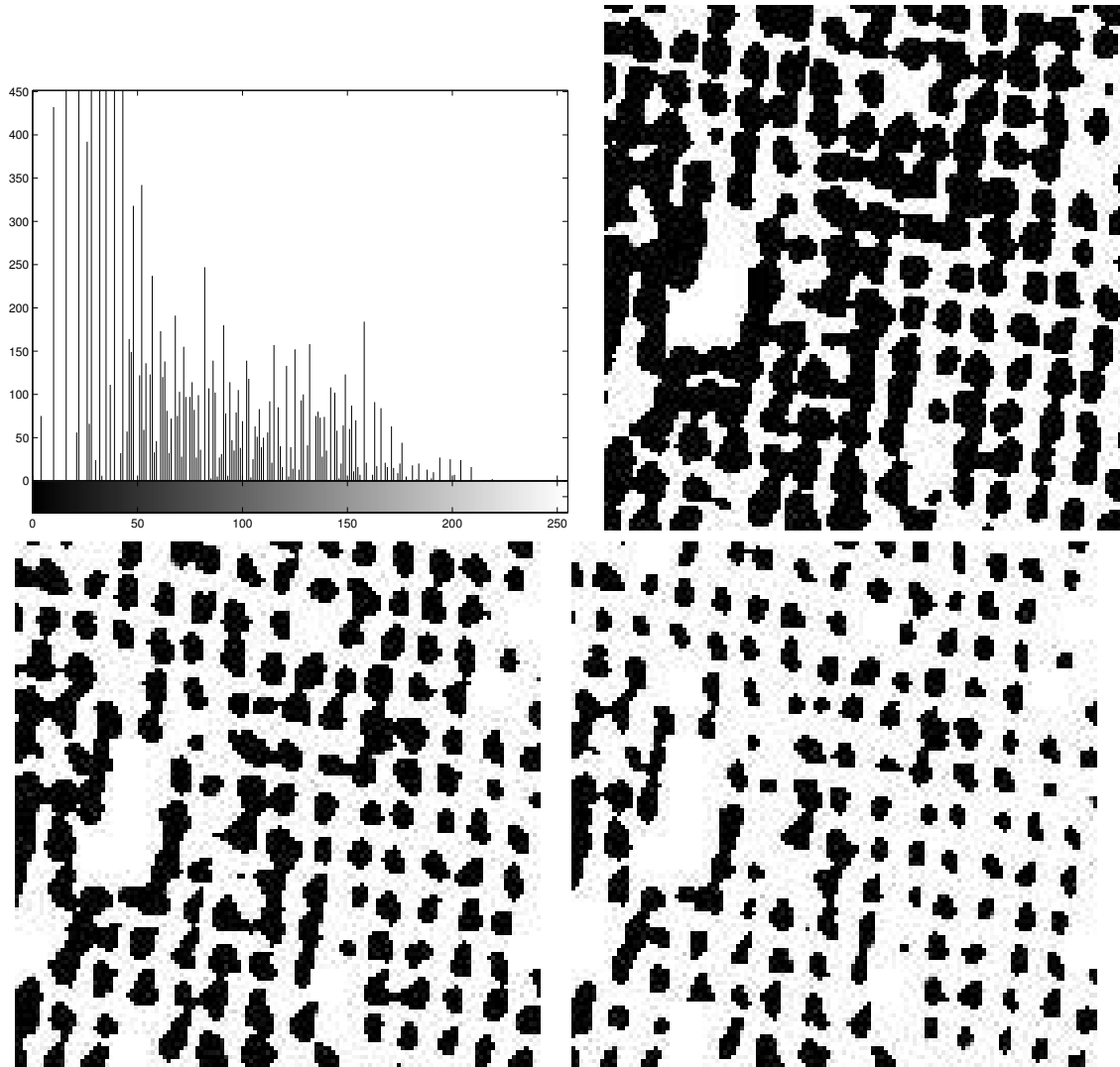


FIG. 1.3 – Seuillage de l'image filtrée (en blanc les vallées, en noir les montagnes) avec une fenêtre moyenne de largeur 5. En haut à gauche : histogramme de \mathcal{I} . En haut à droite : \mathcal{J} obtenue avec $\theta = 50$. En bas à gauche : \mathcal{J} obtenue avec $\theta = 70$. En bas à droite : \mathcal{J} obtenue avec $\theta = 90$.

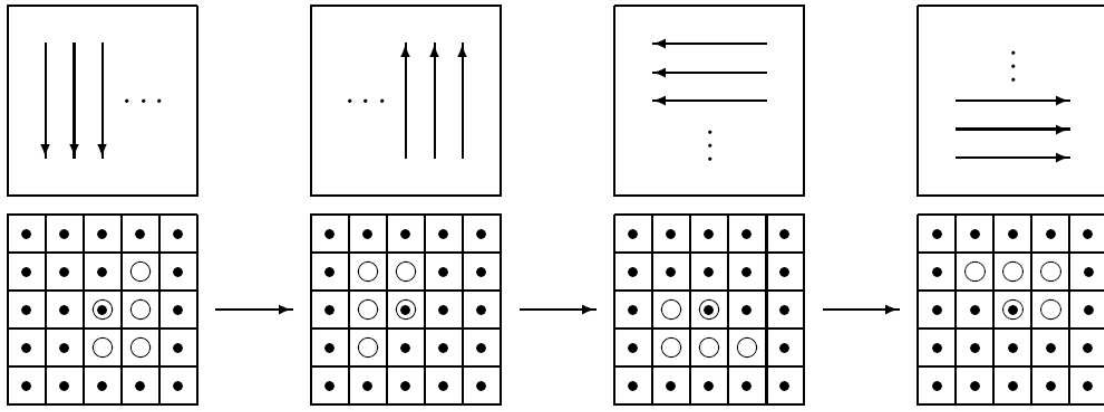


FIG. 1.4 – Les 4 sens de parcours de l'image. Le pixel sur lequel on s'arrête est le point encerclé, ses voisins sont les 4 cercles, et les pixels que l'on sera amené à étudier pour détecter de nouvelles vallées sont les points.

8-connectité). Là encore, les pixels vérifiant ce test sont marqués dans l'image \mathcal{J} comme étant des vallées :

$$\text{si } \mathcal{J}_{i,j} \neq 1 : \min(\mathcal{I}_{i+\alpha,j+\beta}, \alpha \in \{-1,0,1\}, \beta \in \{-1,0,1\}) = \mathcal{I}_{i,j} \Rightarrow \mathcal{J}_{i,j} = 1. \quad (1.3)$$

La prochaine étape consiste à élargir ces vallées. Nous parcourons l'image \mathcal{J} dans ses 4 directions successivement (suivant ses lignes et ses colonnes) et, dès que l'on rencontre un pixel de \mathcal{J} qui se situe dans une vallée, on explore certains de ses voisins (suivant le sens du parcours) dans l'image \mathcal{I} afin d'y détecter des vallées de largeur 3, 2, ou 1. Nous allons maintenant décrire plus précisément cette étape, en prenant par exemple le cas d'un parcours des pixels de l'image de haut en bas et de gauche à droite (schéma de gauche dans Fig. (1.4)).

Nous nous arrêtons donc sur un pixel (i,j) tel que $\mathcal{J}_{i,j} = 1$, supposé suffisamment éloigné du bord pour que les pixels représentés dans Fig. (1.4) existent dans les images \mathcal{I} et \mathcal{J} . Ses 4 voisins qui nous intéressent sont dans ce cas les pixels $(i-1,j+1), (i,j+1), (i+1,j+1)$ et $(i+1,j)$. En premier lieu, il faut regarder si les 3 pixels de la colonne $j+1$ constituent une vallée de largeur 3, ce qui est vérifié si

$$\mathcal{I}_{i+\alpha,j+1} < \mathcal{I}_{i+\beta,j+1}, \text{ avec } \alpha \in \{-1,0,1\} \text{ et } \beta \in \{-2,2\}. \quad (1.4)$$

Dans ce cas ces 3 pixels sont marqués dans \mathcal{J} comme appartenant à la classe des vallées. Sinon, nous cherchons des vallées de largeur 2 parmi les couples de pixels suivants : $\{(i-1,j+1), (i,j+1)\}$, $\{(i,j+1), (i+1,j+1)\}$ et $\{(i+1,j), (i+1,j+1)\}$. Le principe est le même que pour les vallées de largeur 3, il suffit que ces pixels aient un niveau de gris moins élevé que les 2 pixels situés à leurs extrémités pour qu'ils soient acceptés. Voici un exemple des inégalités à vérifier pour le couple $\{(i-1,j+1), (i,j+1)\}$:

$$\mathcal{I}_{i+\alpha,j+1} < \mathcal{I}_{i+\beta,j+1}, \text{ avec } \alpha \in \{-1,0\} \text{ et } \beta \in \{-2,1\}. \quad (1.5)$$

Enfin, nous finissons par une recherche des vallées de largeur 1. Chacun des 4 voisins est étudié de façon similaire : si par exemple

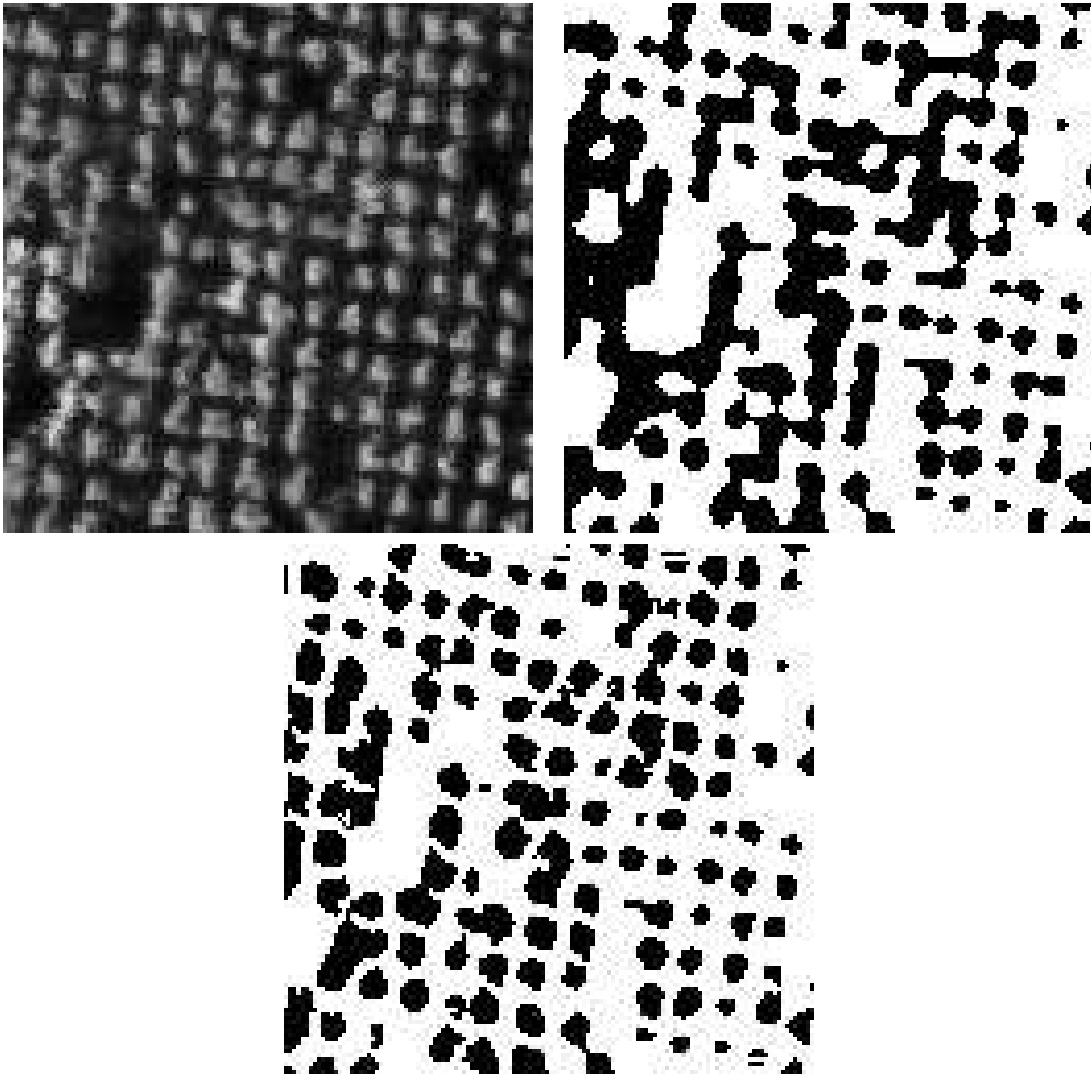


FIG. 1.5 – Résultat du suivi de vallées. En haut à gauche : $\mathcal{I} \odot IFN$. En haut à droite : \mathcal{J} après filtrage et seuillage, $s = 3$ et $\theta = 70$. En bas, \mathcal{J} après les cycles du suivi de vallées.

$$\mathcal{I}_{i-1,j+1} < \mathcal{I}_{i+\beta,j+1}, \text{ avec } \beta = \{-2,0\}, \quad (1.6)$$

alors le pixel $(i - 1, j + 1)$ est marqué dans \mathcal{J} , les mêmes tests s’appliquant aux trois autres.

Nous répétons ces manipulations pour chacun des 4 sens de parcours de l’image (formant un cycle) jusqu’à ce qu’aucun nouveau pixel de vallées ne soit découvert au cours d’un cycle entier. La première partie de l’algorithme est alors achevée, mais il est nécessaire de délimiter plus précisément les couronnes qui pour la plupart sont mal formées, les vallées n’étant pas assez connectées entre elles : ce sera le rôle du “rule-based system”.

1.2.5 Règles d’obtention des contours : le “rule-based system”

Les couronnes obtenues précédemment étant incomplètes, et regroupant parfois même plusieurs arbres, un traitement après le suivi de vallées est indispensable. Au cours de cette seconde partie de l’algorithme, les contours des couronnes d’arbres sont extraits l’un après l’autre. Auparavant, on applique une fermeture à l’image (dilatation puis érosion) à l’aide d’un élément structurant de

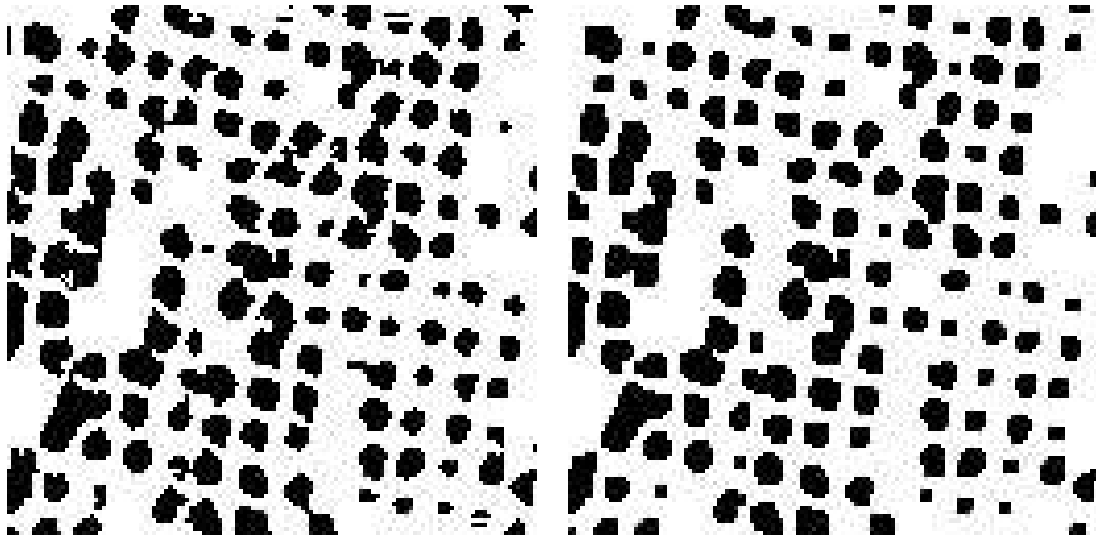


FIG. 1.6 – Résultat du suivi de vallées avec les paramètres $s = 3$ et $\theta = 70$. A gauche, avant la fermeture. A droite, après la fermeture.

taille 3×3 . Cette fermeture a pour effet de faire disparaître tous les éléments des couronnes de dimension inférieure à 3 pixels, comme le montrent les images de la Fig. (1.6).

Tout d’abord, nous plaçons l’image \mathcal{J} dans un cadre plus grand, en lui rajoutant 2 lignes et 2 colonnes à ses extrémités. Nous aurons besoin de ce cadre pour ne pas sortir de l’image dans les prochaines étapes. Nous parcourons alors \mathcal{J} dans une des 4 directions possibles, par exemple de haut en bas et de gauche à droite, à la recherche de pixels (i, j) non marqués tels qu’aucun de leurs voisins ne soit marqué :

$$\sum_{r=-1}^1 \sum_{s=-1}^1 \mathcal{J}_{i+r, j+s} = 0. \quad (1.7)$$

Puis, à partir d’un pixel (i, j) vérifiant (1.7), nous nous déplaçons vers la gauche, jusqu’à trouver un pixel de \mathcal{J} appartenant à une vallée. Ce pixel appartiendra au contour de l’arbre sur lequel nous nous sommes arrêtés. Il sera le premier maillon d’une chaîne qui va contourner l’arbre.

A partir de là, on applique des règles, au nombre de 5, détaillées dans [5] et [26]. Chacune de ces 5 règles contient des critères sur les pixels de l’image binaire, situés dans une direction privilégiée selon le numéro de la règle. Dans le cas où l’un de ces critères est vérifié, le ou les prochains pixels du contour sont déterminés : ils sont par construction reliés avec celui de départ. Puis on recommence cette recherche à partir du dernier maillon de la chaîne.

Ces règles sont appliquées dans l’ordre, de la première à la cinquième, et proposent donc des directions de recherche particulières, s’éloignant progressivement de celle donnée par les deux derniers pixels du contour (de 0 à 360 degrés dans le sens des aiguilles d’une montre par rapport à cette direction). Par exemple, la règle 1 propose des pixels dans deux directions, 0 et 45 degrés, par rapport à l’ancienne direction. On obtiendrait un contour convexe si cette règle était toujours vérifiée, les contours concaves étant quant à eux autorisés par les autres règles.

Une fois que le contour se referme sur lui-même, au niveau du pixel de départ ou de tout autre pixel, nous contrôlons sa longueur pour éviter que ne se produisent des erreurs : il ne doit être ni trop petit ni trop grand. De proche en proche, nous parvenons ainsi à délimiter chaque couronne.

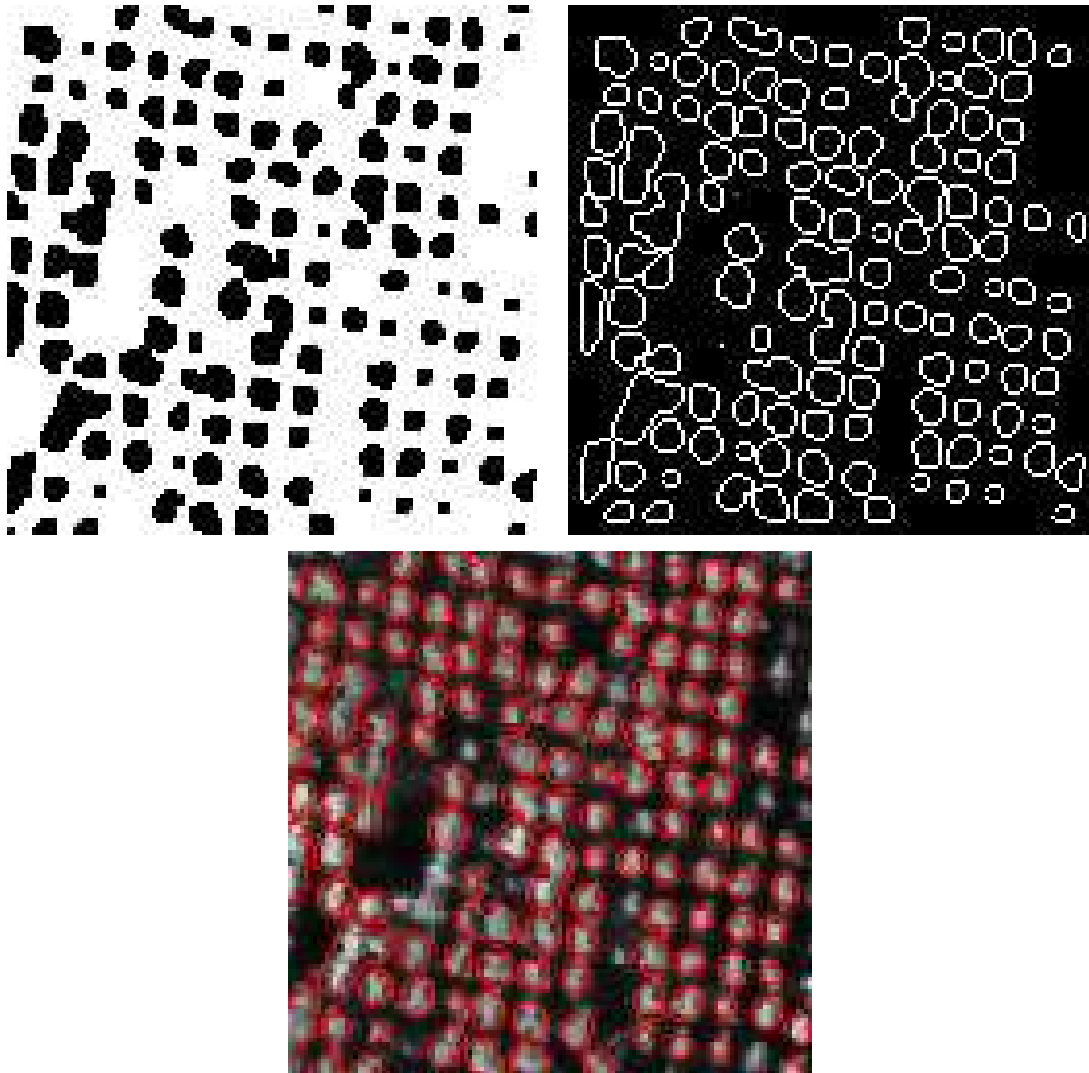


FIG. 1.7 – Résultat du suivi de vallées avec les paramètres $s = 3$ et $\theta = 70$. En haut à gauche, l'image binaire avant l'application des règles. En haut à droite, le contour final des houppiers. En bas, le résultat final visible sur l'image de départ.

Par contre, s'il ne se referme pas et que sa longueur excède une longueur donnée, on stoppe la propagation du contour, et on cherche à nouveau un pixel de départ.

Ce processus est répété un nombre de fois suffisant afin de délimiter les couronnes ne pouvant être isolées qu'une fois les contours adjacents établis. En effet, on remplit le contenu de ces contours dans l'image binaire \mathcal{J} , ce qui apporte de nouvelles informations. Il faut attendre qu'un parcours complet de l'image sans aucun nouveau contour détecté ne soit effectué, avant de finir l'algorithme.

1.2.6 Résultats et interprétations

Un premier résultat, obtenu sur des images de plantations conifériennes (épicéas rouge, blanc et de Norvège, pin rouge) d'une résolution de 31 cm/pixel, avance un dénombrement des couronnes se situant à 7.7% près de celui réalisé par des experts au sol. Mieux encore, le dénombrement fait par photo-interprétation, par des experts, s'est révélé être à seulement 18.1% près. Une grande partie des couronnes d'arbres a donc été correctement détectée, très exactement 81% de ces contours sont

semblables aux contours manuels dessinés par des experts sur le terrain.

Les différentes espèces d'arbres parmi les conifères ne donnent pourtant pas le même résultat, comme en atteste le besoin des auteurs de ne pas tenir compte d'une des parcelles conifériennes pour laquelle l'algorithme ne fonctionne pas : les épicéas de Norvège. Ayant un diamètre beaucoup plus gros que les autres, ceux-ci sont coupés en de nombreuses couronnes au cours de l'algorithme (138 détectés au lieu de 39 sur le terrain). Mais des tests supplémentaires effectués sur des images à résolution plus faible (70 cm/pixel) font disparaître ces inconvénients, de même qu'un lissage plus conséquent. En outre, certaines parcelles de pins, plus éclairées que les autres, ont posé de grandes difficultés tant à l'algorithme qu'aux experts humains. Dans ce cas, le programme ne sépare pas assez les arbres entre eux, car les vallées tranchent moins par rapport aux montagnes à cause de cette illumination. Une analyse plus poussée des erreurs révèle d'ailleurs que la plus commune fut l'omission d'une des couronnes, dû à l'amalgame avec une autre. Ceci arriva pour 15.3% des couronnes.

Dans [18], les tests furent effectués sur des images de résolution 36 cm/pixel représentant des parcelles de "Tsuga heterophylla" (western hemlock), conifères typiques de l'Amérique du Nord, et de cèdres, sur un site qui regroupait des peuplements d'arbres d'âges différents. Les auteurs commencent par décrire des degrés de précision de recouvrement des arbres détectés par l'algorithme sur les arbres existants, afin de mieux quantifier l'exactitude des résultats obtenus.

Se dégage une interprétation intéressante de ces critères en fonction de la densité des troncs d'arbres par hectare et de leur âge. Ainsi, l'algorithme de détection des contours fonctionne mieux sur des vieux arbres, plus gros, pour lesquels le recouvrement est pour 60% d'entre eux parfait (selon les critères d'évaluation définis dans [18]), alors qu'il est rare de trouver des arbres jeunes bien délimités. En ce qui concerne la densité enfin, les auteurs remarquent que pour des densités supérieures à 900 arbres/ha, le nombre d'omissions augmente de manière conséquente, et la sous-estimation du nombre total de couronnes par là-même.

En conclusion de ces travaux, les conditions pour lesquelles cet algorithme fonctionne très bien sont :

- un soleil bas, car une trop grande luminosité sur les parcelles diminue le contraste entre les vallées et les montagnes.
- des peuplements d'arbres appropriés, à savoir des conifères, dont la forme conique est favorable.
- des densités inférieures à 900 arbres/ha, au-delà desquelles le nombre d'omissions augmente considérablement.
- une résolution de l'image adaptée à la taille des arbres que l'on doit détecter, pour éviter le morcellement des grosses couronnes.

1.3. L'approche objets

Dans [15], [16] et [17], une autre approche au problème de détection des houppiers est proposée. Des patrons ("template"), obtenus grâce à un modèle optique des arbres, nous permettent de trouver les positions des arbres sur des images aériennes.

Cette méthode a pour nom le "**template matching**", et est principalement développée au sein du Danish Agricultural and Veterinary Research Council.

1.3.1 Le principe

► Définition d'un patron

Toute la difficulté de cette méthode consiste à définir le patron que l'on essaiera d'assortir dans l'image. Ce patron est en fait une image représentant un unique houppier obtenue grâce à un modèle optique de la scène. Il est calculé à partir d'un modèle géométrique de l'arbre (ellipsoïde), et d'un lancer de rayon initialisé grâce à des informations que donne l'utilisateur au programme, comme l'inclinaison du soleil à l'instant de la photographie, et la position de la caméra.

Ce patron est alors coupé par une fenêtre elliptique, puisque les arbres ne sont pas visibles entièrement sur une image aérienne, à cause de leurs voisins. Les paramètres de forme de cette fenêtre peuvent être renseignés par des connaissances a priori sur la hauteur des arbres, ou leur espacement (Fig. (1.8)). Un exemple de template ainsi que la fenêtre elliptique associée sont visibles sur la Fig. (1.9).

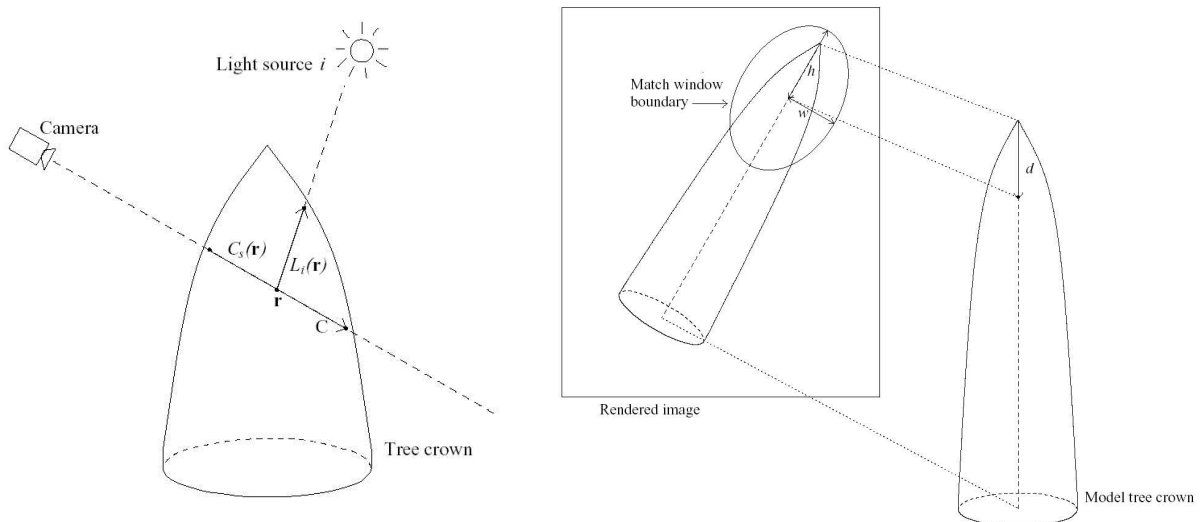


FIG. 1.8 – *Modèle optique, obtention du patron (fenêtre elliptique) par un lancer de rayons. Tiré de [15].*

► Positionnement des sommets des arbres

On transpose alors cette fenêtre dans l'image aérienne, calculant en chaque pixel sa corrélation avec l'image. A partir des maxima locaux dans l'image de corrélation résultante, on peut déduire les positions des arbres.

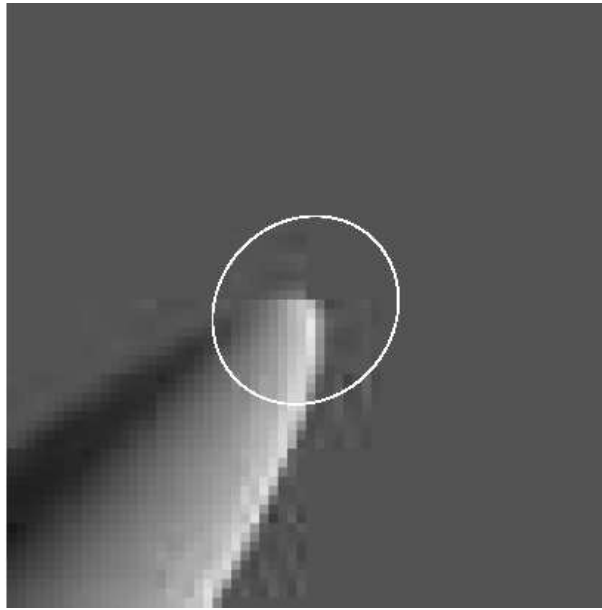


FIG. 1.9 – *Exemple de patron et sa fenêtre elliptique associée. Tiré de [15].*

► Résultats

Avec des paramètres optimaux pour la fenêtre elliptique, une reconnaissance de près de 90% des sommets des arbres pour les parcelles les plus favorables est obtenue. Ces résultats peuvent être altérés dans le cas où aucune connaissance a priori sur les arbres du peuplement forestier n'est disponible.

Il convient parfois de faire varier la taille de la fenêtre elliptique, et de calculer plusieurs images de corrélation. Les positions des arbres seront les pixels de l'image qui ont le plus souvent été cités par les différentes fenêtres. Pour cela, on attribue autant de votes aux pixels d'une image de votes, de même taille que l'image aérienne, que de fenêtres qui y détectent un arbre. Les maxima locaux de l'image de votes ainsi obtenue, une fois lissée, représentent les positions des houppiers. Cette technique est appelée le "template voting".

1.3.2 Récapitulatif de l'algorithme

L'algorithme de template matching comporte donc 3 étapes :

1. modélisation du patron grâce au lancer de rayons. Idéalement, ce patron doit être calculé en chaque pixel de l'image, situé plus ou moins loin du point Nadir. Dans la pratique, un même patron est utilisé pour des zones de l'image regroupant plusieurs pixels.
2. pour chaque pixel de l'image aérienne, on calcule la corrélation de l'image avec le patron.
3. recherche des maxima locaux de l'image de corrélation résultante. Si l'on connaît a priori le nombre d'arbres à détecter, il suffit alors de les prendre dans l'ordre décroissant, en ajoutant une règle qui interdit que deux positions d'arbres ne soient trop proches. Sinon, il est conseillé d'utiliser le principe des votes, en reproduisant la démarche précédente avec plusieurs fenêtres.

1.4. Conclusion

En conclusion de cet état de l'art, nous pouvons remarquer les limites des méthodes actuelles qui ne s'appliquent qu'à des cas particuliers de peuplements et de prises de vue. Il est donc nécessaire de proposer une nouvelle approche, qui puisse englober le problème entier de détection des houppiers.

Processus objets : définitions et simulation



La résolution de problèmes complexes nécessite dans bien des cas l'utilisation de modèles géométriques. Souvent, il est intéressant de se tourner vers les outils statistiques pour les simuler. Ces problèmes relèvent de la *géométrie stochastique*.

Les processus objets font partie intégrante de la géométrie stochastique : ce sont des modèles probabilistes permettant d'étudier des structures géométriques complexes, à partir d'éléments de base simples.

Leurs applications dans le domaine du traitement d'images sont nombreuses. Comme premiers exemples, le lecteur pourra se reporter à [14] et [33], où l'on extrait à partir d'images aériennes les segments de droite constituant le réseau linéique routier, à [2] et [25], où l'on extrait les caricatures des bâtiments, mais aussi à [13], qui applique ces méthodes à la segmentation d'images.

L'objectif de ce chapitre est de présenter les points qui nous ont paru essentiels autour de la théorie des processus objets. Le cadre mathématique donnant naissance à ces processus sera dans un premier temps exposé, avec des exemples à la clé. Puis, dans un deuxième temps, nous aborderons la question de la simulation de ces processus.



2.1. Les processus objets : définitions et exemples

Les éléments de théorie que nous exposons ici ne comprennent que le nécessaire à l'élaboration de notre modèle, le lecteur pouvant trouver des compléments en consultant [34] et [35].

2.1.1 Première approche des processus ponctuels

► L'espace objet U

Chaque objet du processus que l'on souhaite simuler appartient à un même espace que nous appelons l'espace objet, noté U . Dans la pratique, et surtout en traitement d'images, ce sera souvent un sous-ensemble compact de \mathbb{R}^n .

Cet espace U doit contenir toute l'information sur les objets, à savoir leur position, leurs paramètres de forme, puisque ce sont des objets géométriques, et tout autre paramètre servant à les déterminer de façon unique.

Nous munissons U d'une tribu $\mathcal{B}(U)$ ainsi que d'une mesure $\lambda(\cdot)$ telle que $0 < \lambda(U) < \infty$.

► L'espace des configurations Ψ

Définition 2.1

Une **configuration** \mathbf{x} est un ensemble dénombrable, non ordonné, de points de U .

$$\mathbf{x} = \{x_1, \dots, x_n, \dots\}, x_i \in U \forall i.$$

Par exemple, $\{-3, 5.2, \pi, 177\}$ est une configuration de points de \mathbb{R} .

L'espace des configurations Ψ sur U est alors défini comme la réunion de toutes les configurations \mathbf{x} .

Cet espace se décompose sous la forme d'une partition $\Psi = \bigcup_n \Psi_n$, où Ψ_n contient l'ensemble des configurations \mathbf{x} composées exactement de n objets non nécessairement distincts deux à deux.

On note $\Psi_0 = \{\emptyset\}$ la configuration vide.

Le lien entre l'espace objet U et l'espace des configurations Ψ s'obtient par l'intermédiaire de projections de l'espace produit U^n , constitué de tous les n -uplets ordonnés d'éléments non nécessairement distincts deux à deux, sur l'espace des configurations Ψ_n :

$$\omega_n : \begin{array}{l} U^n \rightarrow \Psi_n \\ (x_1, \dots, x_n) \rightarrow \{x_1, \dots, x_n\}. \end{array} \quad (2.1)$$

A un élément de Ψ_n correspondent donc $n!$ éléments de U^n . Les fonctions ω_n sont surjectives.

Par exemple, au sens des configurations, $(4,3,5)$, $(4,5,3)$, $(3,4,5)$, $(3,5,4)$, $(5,3,4)$ et $(5,4,3)$, bien qu'étant différents dans U^3 , $U \subset \mathbb{R}$, sont égaux.

Nous allons désormais restreindre notre étude à un sous-espace de l'espace des configurations.

Définition 2.2

Une configuration \mathbf{x} est dite **localement finie** si pour tout borélien $A \subseteq U$, le nombre de points de \mathbf{x} dans A est fini. On note Ψ^{lf} le sous ensemble de Ψ formé des configurations localement finies.
 Pour tout n , les configurations $\mathbf{x} \in \Psi_n$ sont par définition finies.

Définition 2.3

Une configuration \mathbf{x} est dite **simple** si elle ne contient que des points deux à deux distincts. On note Ψ_s le sous ensemble de Ψ formé des configurations simples.

Par la suite, nous travaillerons sur des configurations localement finies et simples. Afin d'alléger les notations, nous remplacerons abusivement l'écriture de $\Psi^{lf} \cap \Psi_s$ par celle de Ψ .

► **Définition d'un processus ponctuel**

Une fois les configurations d'objets définies, nous pouvons alors aborder la notion de processus ponctuel.

Définition 2.4

Un **processus ponctuel** sur U est une application d'un espace probabilisé $(\Omega, \mathcal{A}, \mathbb{P})$ dans Ψ^{lf} , tel que pour tout borélien borné $A \subseteq U$, le nombre $N(A)$ de points dans A soit une variable aléatoire.

Définition 2.5

Un **processus ponctuel marqué**, ou **processus objet**, sur $U = \mathcal{P} \times \mathcal{K}$ est un processus ponctuel sur $U = \mathcal{P} \times \mathcal{K}$ dont les positions sont dans \mathcal{P} et les marques dans \mathcal{K} , tel que le processus des points non marqués soit un processus ponctuel bien défini sur \mathcal{P} .

D'après [35], pour définir un processus ponctuel il suffit de connaître ses probabilités vides :

Proposition 2.6 : Probabilités vides

Un processus ponctuel est entièrement déterminé par les probabilités vides des boréliens bornés : $\{\mathbb{P}(N(A) = 0) : A \in \mathcal{B}(U)\}$, où $N(A)$ est la variable aléatoire du nombre de points du processus dans A .

Cette proposition peut se révéler utile en pratique comme nous le verrons dans le cadre du processus ponctuel de Poisson défini ci-dessous.

► **Premiers exemples de processus ponctuels**

L'exemple le plus simple qui nous vient à l'esprit est celui d'un processus ponctuel ne contenant qu'un point du plan, ξ , uniformément distribué sur un compact $U \in \mathbb{R}^2$. En munissant l'espace objet U de la mesure de Lebesgue $\lambda(\cdot)$ de \mathbb{R}^2 , ξ est tel que

$$P(\xi \in A) = \frac{\lambda(A)}{\lambda(U)},$$

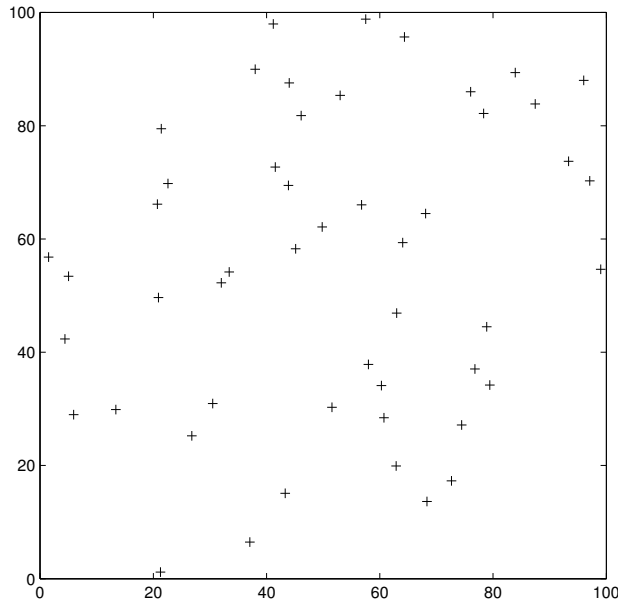


FIG. 2.1 – Simulation d'un processus binomial de 50 points sur $[0,100]^2$

pour tout borélien $A \in U$.

D'autres exemples méritent d'être abordés.

▷ Processus ponctuel binomial

Nous pouvons généraliser l'exemple précédent à des processus ponctuels composés de n points uniformément distribués sur U . On parle dans ce cas de processus ponctuel binomial. La Fig. (2.1) montre un exemple de simulation d'un tel processus, avec $n = 50$ et $U = [0,100]^2$.

Ce type de processus est appelé binomial puisque pour tout borélien $A \in U$, la variable aléatoire $N(A)$ du nombre de points dans A suit une loi binomiale de paramètres n et $\frac{\lambda(A)}{\lambda(U)}$.

▷ Processus ponctuel de Poisson

Un autre exemple de processus ponctuel, sans doute le plus utilisé en simulation, est celui du processus ponctuel de Poisson de loi d'intensité $\nu(\cdot)$. Dans ce cas, la variable aléatoire $N(A)$ définie plus haut suit une loi de Poisson d'espérance $\nu(A)$ pour tout borélien $A \in U$.

Une façon de les simuler consiste à tout d'abord générer le nombre de points n sur U du processus, selon la distribution

$$\mathbb{P}(N(U) = n) = e^{-\nu(U)} \frac{\nu(U)^n}{n!}$$

d'espérance $\nu(U)$, puis de choisir ces n points sur U indépendamment suivant la loi $\frac{\nu(\cdot)}{\nu(U)}$. On parle de processus de Poisson homogène lorsque $\nu(\cdot)$ est proportionnelle à la mesure de Lebesgue $\lambda(\cdot)$ sur U .

Proposition 2.7 : Processus de Poisson conditionné = Processus binomial

|| Soit \mathbf{X} un processus de Poisson de loi d'intensité $\nu(\cdot)$. Pour tout borélien borné $A \in U$, conditionnellement à $\{N(A) = n\}$, la restriction du processus \mathbf{X} à A est un processus ponctuel binomial de points ν -uniformément distribués.

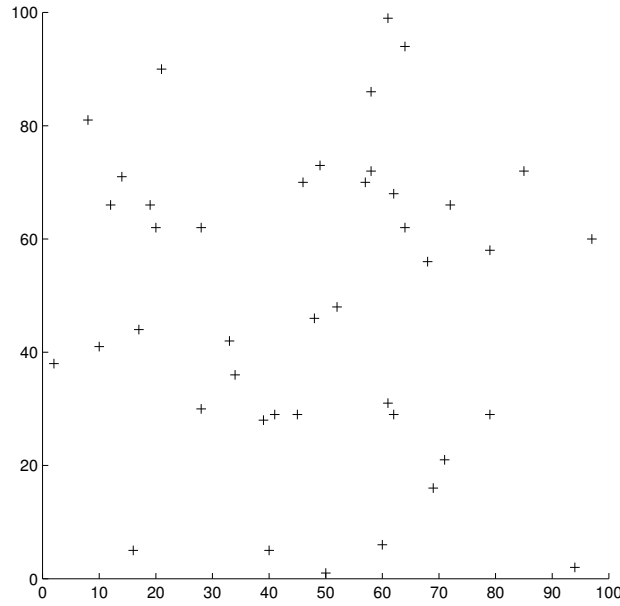


FIG. 2.2 – Réalisation d'un processus de Poisson homogène, $a=0.005$.

◇ **Démonstration :**

Il suffit de montrer que les probabilités vides de ces deux processus sont égales, d'après la proposition (2.1.1), ce qui est fait dans [34] ou [35].

◇

La Fig. (2.2) nous donne un exemple de processus de Poisson homogène, toujours sur $U = [0,100]^2$, de paramètre de proportionnalité $a = 0.005$. Dans ce cas, on vérifie aisément que l'espérance du nombre de points dans U , $\mathbb{E}[N(U)]$, est

$$\mathbb{E}[N(U)] = a \cdot \lambda(U) = 50. \quad (2.2)$$

Enfin, la Fig. (2.3) nous donne une réalisation d'un processus de Poisson accordant 3 fois plus de poids au sous ensemble $U' = [0,50]^2$. Un tel processus peut par exemple être simulé par la loi

$$\nu(A) = a \cdot [\lambda(A \cap (U/U')) + 3\lambda(A \cap (U'))].$$

Avec le même paramètre $a = 0.005$, l'espérance du nombre de points d'une réalisation d'un tel processus est $\nu(U) = 75$.

► **Tribu et mesure de l'espace des configurations**

Pour tout $n > 0$, munissons l'espace produit U^n de la tribu produit \mathcal{B}^n déduite de $\mathcal{B}(U)$, ainsi que Ψ_0 de la tribu triviale $\{\{\emptyset\}, \Psi\}$. Chaque projection ω_n permet d'obtenir une famille $\omega_n(\mathcal{B}^n)$ de parties de Ψ_n . Nous choisissons alors comme tribu sur Ψ la tribu notée $\mathcal{B}(\Psi)$ et engendrée par :

$$\omega_0(\mathcal{B}^0), \omega_1(\mathcal{B}^1), \dots, \omega_n(\mathcal{B}^n), \dots$$

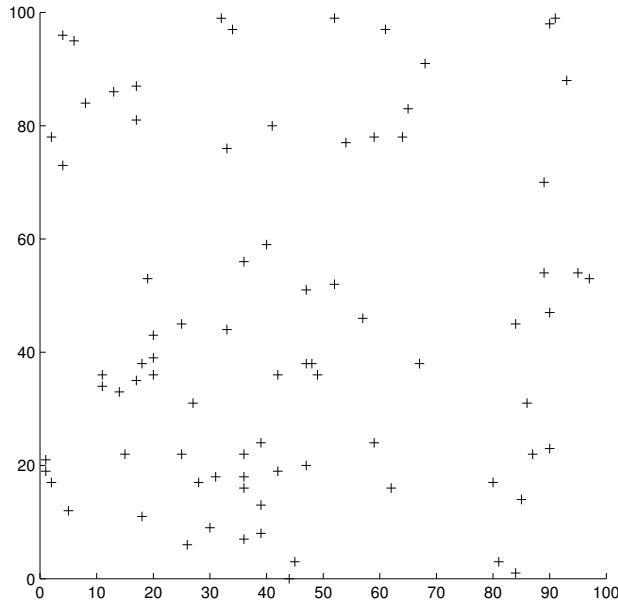


FIG. 2.3 – Réalisation d'un processus de Poisson non homogène, donnant trois fois plus de poids au coin en bas à droite.

On considère alors comme mesure de référence la mesure $\mu(\cdot)$ définie pour tout $B \in \mathcal{B}(\Psi)$ par :

$$\mu(B) = e^{-\lambda(U)} \left(\mathbf{1}_{[\emptyset \in B]} + \sum_{n=1}^{\infty} \frac{\mu_n(B)}{n!} \right) \quad (2.3)$$

avec :

$$\mu_n(B) = \int \dots \int \mathbf{1}_{[\{u_1, \dots, u_n\} \in B]} \lambda(du_1) \dots \lambda(du_n). \quad (2.4)$$

Cette mesure est une mesure de probabilité, qui correspond à la distribution d'un processus de Poisson homogène \mathbf{X} sur U de mesure d'intensité $\lambda(\cdot)$.

Ainsi, le nombre d'objets $n(\mathbf{X})$ de la configuration suit une loi de Poisson de paramètre $\lambda(U)$.

De plus, comme tout processus de Poisson, conditionnellement à l'événement $\{n(\mathbf{X}) = n\}$, la loi $\mu(\cdot | n(\mathbf{X}) = n)$ suit une loi binomiale à n points sur U de densité $\lambda(\cdot)/\lambda(U)$. Elle est donc la superposition de n points indépendants et uniformément répartis dans U suivant la loi $\lambda(\cdot)/\lambda(U)$.

L'existence de l'espace probabilisé de l'ensemble des configurations, $(\Psi, \mathcal{B}(\Psi), \mu(\cdot))$, nous permet alors de définir un processus ponctuel par sa densité par rapport à la loi de Poisson de référence sur cet espace.

2.1.2 Définition d'un processus ponctuel par une densité non normalisée

► Processus ponctuel défini par une loi par rapport à un processus de Poisson de référence

Sur un espace probabilisé de référence $(\Psi, \mathcal{B}(\Psi), \mu(\cdot))$, un processus ponctuel peut être défini par sa loi par rapport à la mesure de Poisson de référence.

Définition 2.8

Un **processus ponctuel** est une application mesurable d'un espace probabilisé et à valeurs dans $(\Psi, \mathcal{B}(\Psi), \mu(\cdot))$. On note $P(\cdot)$ sa loi sur $(\Psi, \mathcal{B}(\Psi), \mu(\cdot))$.

Par la suite, nous travaillerons sur des processus \mathbf{X} dont les lois $P(\cdot)$ sont uniformément continues par rapport à $\mu(\cdot)$, que l'on note $P \ll \mu$. Le théorème de Radon Nikodym nous donne une condition nécessaire et suffisante pour qu'une telle hypothèse sur nos mesures soit vérifiée, qui est que

$$\forall B \in \mathcal{B}(\Psi) : \mu(B) = 0 \implies P(B) = 0. \quad (2.5)$$

La dérivée de Radon Nikodym est alors la densité $f(\cdot)$ de $P(\cdot)$ par rapport à $\mu(\cdot)$, et s'écrit

$$f = \frac{dP}{d\mu}$$

La définition d'un processus objet se réduit donc à la définition d'une application $h(\cdot)$ mesurable positive sur $(\Psi, \mathcal{B}(\Psi), \mu(\cdot))$, et à la vérification de sa μ -intégrabilité, c'est-à-dire à l'existence d'une constante de normalisation. On a alors :

$$f(\mathbf{x}) = \frac{h(\mathbf{x})}{\int_{\Psi} h(\mathbf{x}) \mu(d\mathbf{x})} \quad (2.6)$$

La mesure $P(\cdot)$ du processus s'écrit alors $\forall B \in \mathcal{B}(\Psi)$:

$$\begin{aligned} P(B) &= \int_B f(\mathbf{x}) \mu(d\mathbf{x}) \\ &= \sum_{n=0}^{\infty} \frac{1}{n!} \int_{B \cap U^n} f(\mathbf{x}) \mu_n(d\mathbf{x}). \end{aligned}$$

► Expression générale de la densité non normalisée

Généralement, les densités considérées sont de la forme :

$$h(\mathbf{x}) = \alpha(n(\mathbf{x}))g(\mathbf{x}) \quad (2.7)$$

où $n(\mathbf{x})$ représente le nombre d'objets de la configuration \mathbf{x} , et où $\alpha(\cdot)$ et $g(\cdot)$ sont deux applications mesurables positives définies respectivement sur \mathbb{N} et sur Ψ . La constante de normalisation n'est a priori pas connue lors de la définition du modèle.

Le premier terme correspond à une densité de processus considéré par rapport à la distribution de référence $\mu(\cdot)$, et donc contrôle le nombre moyen d'objets, alors que le second terme permet de modéliser les interactions entre les objets de la configuration.

Nous donnerons par la suite des exemples de processus utilisant ce type de densités.

A l'exception du processus de Poisson, les processus objets peuvent être regroupés en deux catégories : les processus de type attractif et les processus de type répulsif. La plupart du temps, ce comportement peut être caractérisé de manière locale à l'aide de l'intensité conditionnelle de Papangelou.

► Intensité conditionnelle de Papangelou

Définition 2.9

Soit \mathbf{x} un processus ponctuel défini par une densité $f(\mathbf{x})$ par rapport à un processus ponctuel de Poisson de mesure d'intensité finie et non-atomique $\mu(\cdot)$. Alors \mathbf{x} a une **intensité conditionnelle de Papangelou** définie par :

$$\lambda^*(\mathbf{x}, \xi) = \frac{f(\mathbf{x} \cup \{\xi\})}{f(\mathbf{x})}, \forall \xi \in \{U - \bigcup_{i=1}^{n(\mathbf{x})} x_i\}.$$

Soient alors deux configurations \mathbf{x}_1 et \mathbf{x}_2 . Nous interprétons

$$\mathbf{x}_2 \subset \mathbf{x}_1 \Rightarrow \lambda^*(\mathbf{x}_2, \cdot) \geq \lambda^*(\mathbf{x}_1, \cdot) \quad (2.8)$$

comme un cas répulsif, et

$$\mathbf{x}_2 \subset \mathbf{x}_1 \Rightarrow \lambda^*(\mathbf{x}_2, \cdot) \leq \lambda^*(\mathbf{x}_1, \cdot) \quad (2.9)$$

comme un cas attractif.

► Critères de stabilité

Pour que la densité non normalisée soit intégrable par rapport à $\mu(\cdot)$, et comme le calcul de la constante de normalisation est très laborieux, des critères de stabilité ont été proposés.

Ces critères servent aussi à assurer la convergence des algorithmes de simulation des processus ponctuels.

Proposition 2.10 : Stabilité au sens de Ruelle

Un processus défini par une densité non normalisée $h(\cdot)$ par rapport à $\mu(\cdot)$ est **stable au sens de Ruelle** s'il existe un nombre réel M_h tel que :

$$h(\mathbf{x}) \leq M_h^{n(\mathbf{x})}, \forall \mathbf{x} \in \Psi \quad (2.10)$$

◇ Démonstration :

Cette condition est suffisante pour que $h(\cdot)$ soit intégrable, avec comme constante de normalisation :

$$\begin{aligned} \int_{\Psi} h(\mathbf{x}) \mu(d\mathbf{x}) &\leq \int_{\Psi} M_h^{n(\mathbf{x})} \mu(d\mathbf{x}) \\ &= \sum_{k=0}^{\infty} \frac{M_h^k}{k!} \lambda(U)^k \\ &= e^{M_h \lambda(U)}. \end{aligned}$$

◇

Ce critère est peu contraignant pour la fonction $h(\cdot)$ mais peut poser des problèmes de vérification. Aussi est-il utile d'introduire le second critère, plus contraignant mais plus facilement vérifiable.

Cette seconde condition de stabilité est d'ailleurs utilisée pour établir certaines propriétés des schémas de simulation des processus ponctuels par des chaînes de Monte Carlo [3].

Proposition 2.11 : Stabilité locale

Un processus défini par une densité non normalisée $h(\cdot)$ par rapport à $\mu(\cdot)$ est **localement stable** s'il existe un nombre réel M_h tel que :

$$h(\mathbf{x} \cup \xi) \leq M_h h(\mathbf{x}), \forall \mathbf{x} \in \Psi, \forall \xi \in U \quad (2.11)$$

c'est à dire si l'intensité conditionnelle de Papangelou est bornée.

◇ **Démonstration :**

Cette condition implique la condition (2.10), puisque

$$h(\mathbf{x}) \leq h(\emptyset) M_h^{n(\mathbf{x})}, \forall \mathbf{x} \in \Psi.$$

◇

2.1.3 Processus ponctuels de Markov

► **Définition des processus ponctuels de Markov**

Les processus ponctuels de Markov sont des processus ponctuels dont l'intensité conditionnelle de Papangelou ne dépend que d'un certain voisinage. Autrement dit, du point de vue énergétique, leurs densités s'écrivent au moyen de potentiels d'interactions entre les objets du processus. On parle alors de processus de Gibbs.

Il convient dans un premier temps de se fixer une relation de voisinage \sim , symétrique et réflexive sur U . On peut par exemple considérer la relation de proximité entre deux objets x_1 et x_2 de $U = \mathbb{R}^2$, où $d(\cdot, \cdot)$ est la fonction distance :

$$x_1 \sim x_2 \Leftrightarrow d(x_1, x_2) \leq d_\sim, d_\sim \in \mathbb{R}^+ \quad (2.12)$$

Il nous reste à définir la notion de densité héréditaire avant d'introduire la notion de processus ponctuel de Markov.

Définition 2.12

Une densité $f(\cdot)$ est dite **héréditaire** si elle vérifie la relation :

$$f(\mathbf{x}) > 0 \Rightarrow f(\mathbf{y}) > 0, \forall \mathbf{y} \subseteq \mathbf{x}, \forall \mathbf{x} \in \Psi.$$

Définition 2.13

On appelle **processus ponctuel de Markov** (au sens de Ripley-Kelly) sous la relation symétrique et réflexive \sim , tout processus dont la densité $f(\cdot)$ est telle que :

1. $f(\cdot)$ est héréditaire
2. Pour toute configuration \mathbf{x} telle que $f(\mathbf{x}) > 0$, l'intensité conditionnelle de Papangelou $\lambda^*(\mathbf{x}, \xi)$ dépend uniquement de ξ et des objets de \mathbf{x} voisins de ξ au sens de la relation de voisinage \sim .

L'introduction de la notion de clique permet ensuite la caractérisation des processus ponctuels de Markov.

Définition 2.14

Soit \sim une relation de voisinage symétrique et réflexive sur Ψ . Une configuration \mathbf{x} de Ψ est appelée une **clique** si tous les éléments de \mathbf{x} sont voisins les uns des autres, c'est à dire si :

$$\forall x_1, x_2 \in \mathbf{x}, x_1 \sim x_2.$$

Le théorème suivant nous donne une condition nécessaire et suffisante sur la densité d'un processus pour que ce soit un processus ponctuel de Markov.

Théorème 2.15 : Processus ponctuel de Markov

Un processus ponctuel de densité $f(\cdot)$ est un **processus ponctuel de Markov** (au sens de Ripley-Kelly) si et seulement si $f(\cdot)$ est de la forme :

$$\begin{aligned} f(\mathbf{x}) &= \prod_{\text{cliques } \mathbf{y} \subseteq \mathbf{x}} \phi(\mathbf{y}) \\ &= \prod_{\mathbf{y} \subseteq \mathbf{x}} \psi(\mathbf{y}), \forall \mathbf{x} \in \Psi \end{aligned}$$

où $\psi(\cdot)$ est une fonction d'interaction, c'est à dire une fonction telle que $\psi(\mathbf{y}) = 1$ lorsque \mathbf{y} n'est pas une clique.

Nous allons maintenant présenter deux exemples de processus ponctuels de Markov.

► **Exemples de processus ponctuels de Markov**

On peut exprimer les densités des processus des exemples suivants sous forme exponentielle, c'est à dire qu'il existe un vecteur $\theta \in \mathbb{R}^d$ et une application $\rho(\cdot) : \Psi \rightarrow \mathbb{R}^d$ telles que :

$$h(\mathbf{x}) = e^{\langle \theta, \rho(\mathbf{x}) \rangle}$$

où $\langle \cdot, \cdot \rangle$ est le produit scalaire sur \mathbb{R}^d .

▷ **Processus de Strauss**

Soit \mathbf{x} une configuration de l'espace $(\Psi, \mathcal{B}(\Psi), \mu(\cdot))$. Nous notons d'une part $n(\mathbf{x})$ le nombre d'objets contenus dans \mathbf{x} . Autrement dit, $\mathbf{x} \in \Psi_{n(\mathbf{x})}$. D'autre part, $s(\mathbf{x})$ représente le nombre de paires d'objets voisins, au sens de la relation de voisinage \sim , dans la configuration \mathbf{x} . La densité d'un processus de Strauss s'écrit alors

$$h(\mathbf{x}) = e^{\beta \cdot n(\mathbf{x}) + \gamma \cdot s(\mathbf{x})}, (\beta, \gamma) \in \mathbb{R}^2 \tag{2.13}$$

avec

$$s(\mathbf{x}) = \#(\{\xi, \eta\} \subseteq \mathbf{x} : \xi \sim \eta, \xi \neq \eta). \tag{2.14}$$

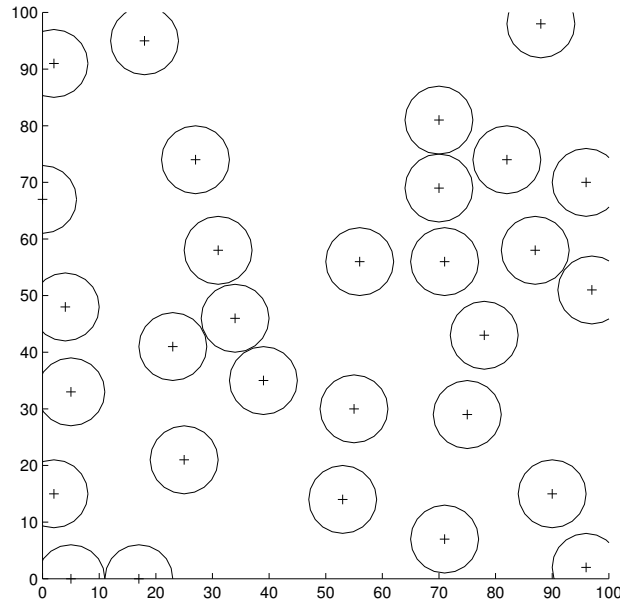


FIG. 2.4 – Réalisation d'un processus de Strauss répulsif : $\beta = \ln(0.003)$, $\gamma = \ln(0.001)$ et $d_\sim = 6$

En choisissant par exemple pour relation de voisinage \sim la relation (2.12) l'intensité conditionnelle de Papangelou s'exprime sous la forme :

$$\lambda^*(\mathbf{x}, \xi) = e^{\beta + \gamma \cdot s(\mathbf{x}, \xi)}$$

où

$$s(\mathbf{x}, \xi) = \sum_{x_i \in \mathbf{x}} \mathbf{1}_{d(x_i, \xi) \leq d_\sim}$$

Selon les valeurs de γ , un processus de Strauss peut avoir différents aspects :

1. Si $\gamma = 0$ on retrouve un processus ponctuel de Poisson homogène d'intensité $e^\beta \cdot \lambda(\cdot)$.
2. Si $\gamma < 0$ le processus montre une répulsion entre les points voisins, de plus en plus forte quand γ tend vers $-\infty$ (cf Fig. (2.4)). La limite $\gamma \rightarrow -\infty$ est un processus dit *hard-core*, la densité interdisant d'avoir deux objets voisins dans la configuration \mathbf{x} .
3. Si $\gamma > 0$, le processus de Strauss est attractif. Il y a explosion du processus (cf Fig. (2.5)). La densité n'est intégrable que si l'on change le terme $\beta^{n(\mathbf{x})}$ en un terme adéquat, comme $\mathbf{1}_{\{n(\mathbf{x}) \leq n_0\}}$ par exemple. Ce dernier n'autorise pas de configurations \mathbf{x} de taille supérieure à n_0 .

▷ Processus d'interaction d'aires

Le processus d'interaction d'aires possède des propriétés attractives intéressantes en traitement d'images, où il y a souvent des zones d'intérêt que l'on souhaite explorer sans risquer de faire exploser notre processus ponctuel (ce qui arrive avec le processus de Strauss).

Prenons par exemple le cas d'un processus ponctuel défini sur un espace $U \subset \mathbb{R}^{d>2}$. Sa densité comprend un terme dépendant de la mesure $\nu(\cdot)$ de la réunion des supports $Z(x_i)$ des objets de la configuration \mathbf{x} , $Z(\mathbf{x})$:

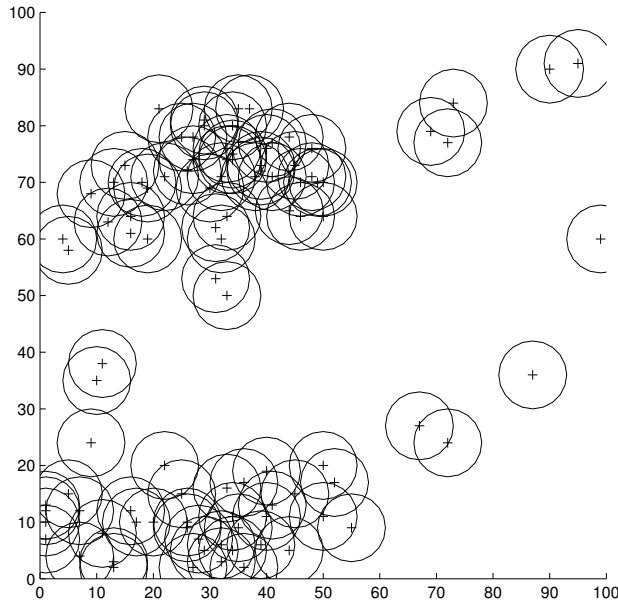


FIG. 2.5 – Réalisation d'un processus de Strauss attractif: $\beta = \ln(0.003)$, $\gamma = \ln(2)$ et $d_\sim = 6$

$$Z(\mathbf{x}) = \bigcup_{x_i \in \mathbf{x}} Z(x_i).$$

Elle s'écrit

$$h(\mathbf{x}) = e^{\beta \cdot n(\mathbf{x}) - \gamma \cdot \nu(Z(\mathbf{x}))}. \quad (2.15)$$

Plusieurs cas sont à différencier :

1. Lorsque $\gamma = 0$, ce processus est un processus de Poisson homogène d'intensité $e^\beta \cdot \lambda(\cdot)$.
2. Pour $\gamma > 0$ ce processus est attractif car il a tendance à favoriser les configurations \mathbf{x} avec une faible valeur de $\nu(Z(\mathbf{x}))$.
3. Pour $\gamma < 0$, ce processus est régulier. Lorsque $\gamma \rightarrow -\infty$, ce processus converge vers un processus uniforme sur l'ensemble

$$\mathbb{H} = \{\mathbf{x} \in \Psi : \nu(Z(\mathbf{x})) = \nu^*\}, \nu^* = \max(\nu(Z(\mathbf{x})), \mathbf{x} \in \Psi).$$

\mathbb{H} est donc constitué des configurations dont le support a une mesure maximale.

Couplé à une répulsion entre les objets se chevauchant, on peut alors obtenir des configurations qui recouvrent l'espace U entièrement (si on choisit γ_1 et γ_2 négatifs: Fig. (2.6) ou Fig. (2.7)), dont la densité s'exprime sous la forme :

$$h(\mathbf{x}) = e^{\beta \cdot n(\mathbf{x}) + \gamma_1 \cdot s(\mathbf{x}) - \gamma_2 \cdot \nu(Z(\mathbf{x}))}.$$

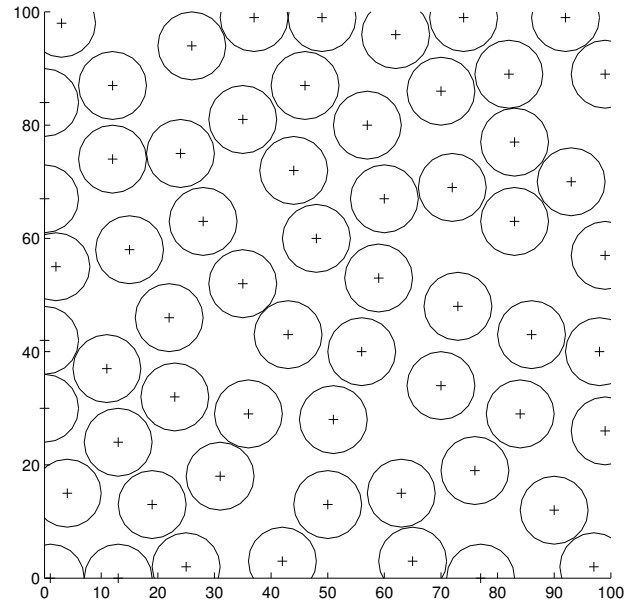


FIG. 2.6 – Réalisation d'un processus d'interaction d'aires, avec une répulsion : $\beta = \ln(0.003)$, $\gamma_1 = \ln(0.001)$, $\gamma_2 = -5$ et $d_{\sim} = 6$

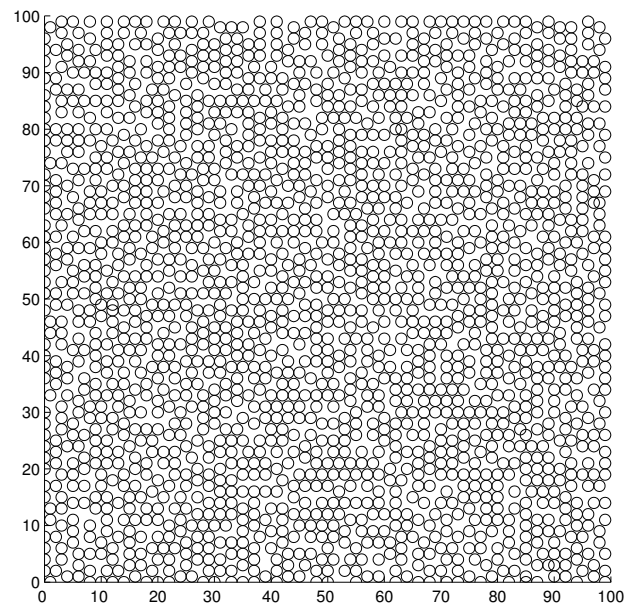


FIG. 2.7 – Réalisation d'un processus d'interaction d'aires, avec une répulsion : $\beta = \ln(0.003)$, $\gamma_1 = \ln(0.001)$, $\gamma_2 = -5$ et $d_{\sim} = 1$

2.2. Algorithme de simulation : Metropolis-Hastings-Green

Quelques rappels sur les chaînes de Markov et la théorie ergodique sont disponibles dans l'Annexe B de ce rapport. Nous considérerons ces notions comme acquises dans cette partie.

Soit $\pi(\cdot)$ la loi objectif que l'on souhaite simuler. Green, dans [11], présente un algorithme de type Metropolis-Hastings à sauts réversibles, permettant de simuler une loi connue à un facteur multiplicatif près grâce à une chaîne de Markov. Cela nous permettra de ne pas calculer la constante de normalisation.

L'autre avantage de cet algorithme est qu'il autorise cette simulation sur des espaces dont la dimension varie, ce qui se prête tout à fait à notre étude.

Nous exposerons tout d'abord dans cette partie le principe de l'algorithme en l'appliquant au cas des processus ponctuels. Puis, nous remarquerons que sous certaines conditions, cet algorithme converge vers la loi objectif $\pi(\cdot)$.

2.2.1 Présentation de l'algorithme

L'algorithme de Metropolis-Hastings-Green (MHG) permet la simulation au sein d'un même espace de dimension fixée, ainsi que les transitions (réversibles) d'un espace vers un autre espace de dimension différente. Nous pourrions donc explorer l'espace des configurations Ψ dans son ensemble, c'est à dire tous les Ψ_n .

Il est présenté par Green comme une généralisation de l'algorithme de Metropolis-Hastings. La principale différence entre les deux est que Green remplace les densités non normalisées de l'algorithme de Metropolis-Hastings par des mesures non normalisée sur l'espace des configurations Ψ . Quant à la densité de transition auxiliaire, elle est remplacée par le noyau $Q(\mathbf{x}, B)$, avec $\mathbf{x} \in \Psi$ et $B \in \mathcal{B}(\Psi)$.

► Le principe

Soient alors :

- $\pi(\cdot)$ la loi objectif définie sur l'espace des configurations Ψ .
- $Q(\mathbf{x}, B)$ le noyau de propositions défini sur $\Psi \times \mathcal{B}(\Psi)$.
- (\mathbf{x}_i) une chaîne de Markov sur Ψ .

Le principe de l'algorithme de Metropolis-Hastings-Green est le suivant : à chaque itération, on propose une modification de l'état courant \mathbf{x}_t suivant le noyau de propositions $Q(\mathbf{x}_t, \cdot)$, que l'on accepte ou que l'on rejette avec une probabilité α . Cette modification de l'état courant \mathbf{x}_t vers un état \mathbf{y} est communément appelée **mouvement**.

Comme démontré dans [11] par Green, la probabilité d'acceptation d'un mouvement est calculée pour assurer la π -réversibilité, et donc l'invariance de la mesure $\pi(\cdot)$ pour notre chaîne de Markov.

Pour obtenir un tel résultat, Green suppose l'existence d'une mesure symétrique $\xi(\cdot, \cdot)$, de telle sorte que $\pi(\cdot)Q(\cdot, \cdot)$ soit absolument continu par rapport à $\xi(\cdot, \cdot)$: $\pi(\cdot)Q(\cdot, \cdot) \ll \xi(\cdot, \cdot)$.

Une condition suffisante pour qu'une mesure soit absolument continue par rapport à une autre est donnée par la proposition suivante.

Proposition 2.16 : Radon-Nikodym

$$\left\| \begin{array}{l} \text{Soient } \mu_1(\cdot) \text{ et } \mu_2(\cdot) \text{ deux mesures définies sur un même ensemble } \Psi. \text{ Si pour} \\ \text{tout borélien } B \in \Psi, \\ \mu_2(B) = 0 \Rightarrow \mu_1(B) = 0, \\ \text{alors } \mu_1(\cdot) \text{ est absolument continue par rapport à } \mu_2(\cdot). \end{array} \right. \quad (2.16)$$

Dans ce cas, $\pi(d\mathbf{x})Q(\mathbf{x},d\mathbf{y})$ admet une densité finie $f(\mathbf{x},\mathbf{y})$ par rapport à $\xi(d\mathbf{x},d\mathbf{y})$ sur $\Psi \times \Psi$, et la dérivée de Radon-Nikodym s'écrit :

$$f(\mathbf{x},\mathbf{y}) = \frac{\pi(d\mathbf{x})Q(\mathbf{x},d\mathbf{y})}{\xi(d\mathbf{x},d\mathbf{y})}. \quad (2.17)$$

Ce point sera détaillé après la présentation du noyau de propositions.

► Rôle et composition du noyau de propositions

Nous allons présenter cet algorithme dans le cadre d'un noyau de propositions multiple : $Q(\cdot, \cdot)$ sera alors formé de plusieurs noyaux de propositions $Q_i(\cdot, \cdot)$ décrivant chacun un mouvement différent que l'on autorise :

$$Q(\mathbf{x}, \cdot) = \sum_{i \in I} Q_i(\mathbf{x}, \cdot) < 1.$$

Ainsi, à chaque itération t , nous sélectionnons dans un premier temps parmi les noyaux $Q_i(\cdot, \cdot)$ celui que l'on va utiliser pour proposer un mouvement à partir de la configuration courante \mathbf{x}_t . Chacun des noyaux Q_i est choisi avec la probabilité $Q_i(\mathbf{x}_t, \Psi)$, tandis qu'avec la probabilité restante, à savoir $1 - \sum_{i \in I} Q_i(\mathbf{x}_t, \Psi)$, nous imposons de rester dans l'état courant.

Comme nous le verrons plus tard, pour tout $i \in I$, il est nécessaire que $\pi(\cdot)Q_i(\cdot, \cdot)$ soit absolument continue par rapport à une mesure symétrique $\xi_i(\cdot, \cdot)$ sur $\Psi \times \Psi$. De cette façon, on peut définir pour chacun des noyaux une dérivée de Radon Nikodym particulière $f_i(\cdot, \cdot)$:

$$f_i(\mathbf{x},\mathbf{y}) = \frac{\pi(d\mathbf{x})Q_i(\mathbf{x},d\mathbf{y})}{\xi_i(d\mathbf{x},d\mathbf{y})}. \quad (2.18)$$

Le rôle de ces noyaux de propositions est fondamental. Ils jouent essentiellement sur la vitesse de convergence de notre algorithme, et sur l'exactitude du résultat obtenu. En effet, ils doivent nous permettre à chaque itération d'explorer une grande partie de l'espace des configurations afin de ne pas rester bloqué dans un minimum local de l'énergie à minimiser (\Leftrightarrow densité à maximiser).

► Détails de l'algorithme

Metropolis-Hastings-Green

1. A partir de la configuration courante \mathbf{x}_t :
 - on choisit avec la probabilité $p_i(\mathbf{x}_t) = Q_i(\mathbf{x}_t, \Psi)$, un des noyaux de propositions $Q_i(\mathbf{x}_t, \cdot)$ et on passe à l'étape 2.
 - avec la probabilité $1 - \sum_{i \in I} p_i(\mathbf{x})$, on reste dans l'état courant et on passe à la prochaine itération.
2. On génère la configuration \mathbf{y} , connaissant la configuration \mathbf{x}_t , selon le noyau sélectionné $\frac{Q_i(\mathbf{x}_t, \cdot)}{p_i(\mathbf{x})}$.
3. On calcule le **rapport de Green** relatif au noyau i , $R_i(\mathbf{x}_t, \mathbf{y})$:

$$R_i(\mathbf{x}_t, \mathbf{y}) = \frac{f_i(\mathbf{y}, \mathbf{x}_t)}{f_i(\mathbf{x}_t, \mathbf{y})} = \frac{\pi(d\mathbf{y})Q_i(\mathbf{y}, d\mathbf{x}_t)}{\pi(d\mathbf{x}_t)Q_i(\mathbf{x}_t, d\mathbf{y})} \quad (2.19)$$

4. Avec la probabilité $\alpha_i(\mathbf{x}_t, \mathbf{y}) = \min(1, R_i(\mathbf{x}_t, \mathbf{y}))$ on accepte le nouvel état \mathbf{y} , sinon on conserve \mathbf{x}_t .

► Preuve de π -réversibilité

On rappelle ici la définition de la réversibilité d'une chaîne de Markov :

Définition 2.17

Une chaîne de Markov est dite **réversible** pour la loi objectif $\pi(\cdot)$ si son noyau de transition vérifie :

$$\int_B K(\mathbf{x}, A) \pi(d\mathbf{x}) = \int_A K(\mathbf{x}, B) \pi(d\mathbf{x}), \forall A, B \in \mathcal{B}(\Psi). \quad (2.20)$$

La richesse de l'algorithme de Green repose sur un choix judicieux de la probabilité d'acceptation du mouvement, qui implique la π -réversibilité de la chaîne, puisque l'équation (2.20) est alors satisfaite.

La proposition suivante ainsi que sa démonstration nous indiquent comment Green parvient à l'expression de cette probabilité.

Proposition 2.18 : Rapport de Green

Soient $Q_i(\mathbf{x}, A)$, $i \in I$ et $A \in \mathcal{B}(\Psi)$, les différents noyaux de propositions vérifiant $\sum_{i \in I} Q_i(\mathbf{x}, \Psi) \leq 1$ pour tout $\mathbf{x} \in \Psi$. Alors l'algorithme proposé par Green simule une chaîne de Markov réversible.

◇ Démonstration :

Cette démonstration est disponible dans l'annexe (C.1).

◇

2.2.2 Convergence de la chaîne de Markov considérée

OBJECTIF : Nous allons démontrer dans cette sous-partie que l'*algorithme de Green converge bien vers la mesure objectif* $\pi(\cdot)$.

► Convergence en variations totales : rappels

Comme exposé dans l'Annexe B, nous rappelons que les conditions suffisantes pour que la chaîne de Markov simulée par l'algorithme Metropolis-Hastings-Green converge en variations totales vers une mesure objectif $\pi(\cdot)$ sont :

1. la **stationnarité** de la mesure $\pi(\cdot)$ (déjà démontrée)
2. l'**irréductibilité** de la chaîne
3. l'**apériodicité**
4. la **réurrence au sens de Harris**

Le cas de l'**apériodicité** peut se régler rapidement. Elle est en effet assurée dès que le noyau $Q(\cdot, \cdot)$ permet de rester dans l'état courant \mathbf{x} . Autrement dit, il suffit d'imposer aux différents noyaux $Q_i(\cdot, \cdot)$, quel que soit \mathbf{x} , l'inégalité $\sum_{i \in I} p_i(\mathbf{x}) = \sum_{i \in I} Q_i(\mathbf{x}, \Psi) < 1$.

Nous allons démontrer que les autres propriétés découlent de l'hypothèse de stabilité (2.11) faite sur la densité du processus à modéliser.

Nous arrivons alors à un point où il nous faut définir des mouvements pour parcourir l'espace des configurations. En effet, l'irréductibilité de la chaîne par exemple impose que tout ensemble de mesure non nulle soit accessible à partir de n'importe quel point de l'espace des configurations Ψ .

Nous allons donc commencer par **définir un noyau de propositions** qui nous permettra d'explorer tout cet espace et de changer de dimension : le **noyau de naissance et de mort**.

► Description des mouvements de naissance et de mort

On souhaite simuler un processus ponctuel de densité $h(\cdot)$ non normalisée par rapport à une mesure d'un processus de Poisson de référence $\mu(\cdot)$, de mesure d'intensité la mesure de Lebesgue $\lambda(\cdot)$ sur U . Par la suite, on notera $\mathbf{x} = \{x_1, \dots, x_{n(\mathbf{x})}\}$ un élément de l'espace des configurations Ψ .

A partir de la configuration courante \mathbf{x}_t , une **naissance** consiste à proposer d'ajouter un objet u de l'espace objet U à cette configuration, c'est à dire à remplacer \mathbf{x}_t par $\mathbf{y} = \mathbf{x}_t \cup \{u\}$.

A l'inverse, la **mort** consiste en la suppression d'un des objets de la configuration courante \mathbf{x}_t . C'est à dire que l'on choisit un entier $i \in [1, n(\mathbf{x})]$, et que l'on propose de remplacer \mathbf{x}_t par $\mathbf{y} = \mathbf{x}_t \setminus \{x_i\} = \{x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_{n(\mathbf{x})}\}$.

Il est essentiel de définir simultanément ces deux mouvements, puisqu'ils sont symétriques : si on peut passer grâce à une naissance d'une configuration \mathbf{x} à une configuration \mathbf{y} , on doit être capable, pour la réversibilité de la chaîne, de passer de \mathbf{y} à \mathbf{x} .

Nous pouvons désormais définir le noyau de propositions lié à ces deux mouvements.

► Noyau de propositions de naissance et de mort

On considère un noyau de propositions global $Q_{nm}(\cdot, \cdot)$, composé lui-même de deux noyaux, l'un proposant une naissance et l'autre une mort :

$$Q_{nm}(\mathbf{x}, \cdot) = Q_n(\mathbf{x}, \cdot) + Q_m(\mathbf{x}, \cdot)$$

Nous pourrions faire varier les probabilités de proposer une naissance ou une mort en fonction de la configuration courante \mathbf{x}_t , comme par exemple

- en proposant plus souvent des naissances quand le nombre d'objets de la configuration est faible.
- et inversement en proposant plus de morts lorsque la configuration comporte un grand nombre d'objets

Mais afin d'alléger les calculs, nous supposons fixées ces grandeurs :

$$Q_n(\mathbf{x}, \Psi) = p_n, Q_m(\mathbf{x}, \Psi) = p_m. \quad (2.21)$$

Pour s'assurer de l'apériodicité de la chaîne de Markov simulée par l'algorithme, il suffirait de choisir p_n et p_m tels que $p_n + p_m = p_{nm} < 1$. Une mise en facteur dans (2.21), ainsi qu'une redéfinition (à ce facteur près) des noyaux de propositions nous mène à :

$$Q_{nm}(\mathbf{x}, \cdot) = p_n Q_n(\mathbf{x}, \cdot) + p_m Q_m(\mathbf{x}, \cdot).$$

Dans ce cas, on se retrouve avec $Q_n(\mathbf{x}, \Psi) = Q_m(\mathbf{x}, \Psi) = 1$.

▷ Naissance

Pour la naissance, on choisit λ -uniformément un objet de U et on propose de l'ajouter à la configuration. On rappelle que $\lambda(\cdot)$ est la mesure de Lebesgue sur l'espace objet U , et qu'elle est la mesure d'intensité du processus de Poisson de référence par rapport auquel notre processus ponctuel est défini.

Soient alors B un borélien de $\mathcal{B}(\Psi)$, et \mathbf{x}_t la configuration courante :

$$Q_n(\mathbf{x}_t, B) = \int_{u \in U} \mathbf{1}_B(\mathbf{x}_t \cup \{u\}) \frac{\lambda(du)}{\lambda(U)} \quad (2.22)$$

On remarque bien que si $\mathbf{x}_t \in \Psi_n$, alors ce noyau concentre tout son poids dans Ψ_{n+1} . Autrement dit, le couple de configurations $(\mathbf{x}_t, \mathbf{y})$ est dans l'ensemble

$$\mathcal{N} = \{(\mathbf{x}_t, \mathbf{y}) \in \Psi_n \times \Psi_{n+1} : x_i = y_i, \forall i \leq n\}.$$

▷ Mort

Pour la mort, on choisit uniformément un objet de la configuration courante et on propose de le supprimer. Soient alors B un borélien de $\mathcal{B}(\Psi)$, et \mathbf{x}_t la configuration courante :

$$Q_m(\mathbf{x}_t, B) = \sum_{u \in \mathbf{x}_t} \mathbf{1}_B(\mathbf{x}_t \setminus \{u\}) \frac{1}{n(\mathbf{x}_t)} \quad (2.23)$$

On remarque bien que si $\mathbf{x}_t \in \Psi_{n+1}$, alors ce noyau concentre tout son poids dans Ψ_n . Les configurations étant non ordonnées, on peut s'arranger pour ce que les n premiers objets des deux configurations soient les mêmes. Ainsi le couple de configurations $(\mathbf{x}_t, \mathbf{y})$ est dans l'ensemble

$$\mathcal{M} = \{(\mathbf{x}_t, \mathbf{y}) \in \Psi_{n+1} \times \Psi_n : \exists j \in [1, n+1], x_i = y_i \text{ quand } i < j, x_{i+1} = y_i \text{ quand } i \geq j\}.$$

► Mesure symétrique ξ_{nm}

Il nous reste à trouver la mesure symétrique $\xi_{nm}(\cdot, \cdot)$ par rapport à laquelle $\pi(\cdot)Q_{nm}(\cdot, \cdot)$ est absolument continu. Cette mesure $\xi_{nm}(\cdot, \cdot)$ est à définir sur $\Psi \times \Psi$. Soient alors $A, B \in \mathcal{B}(\Psi)$. Si ni une naissance ni une mort n'est possible pour passer de A à B , alors $\xi(A, B)$ peut être nul puisque dans ce cas $\pi(\cdot)Q_{nm}(\cdot, \cdot) = 0$ (et donc (2.16) reste vérifiée).

Choisissons alors pour $\xi_{nm}(\cdot, \cdot)$ la mesure suivante :

$$\xi_{nm}(A, B) = \int_{\Psi} \int_{u \in U} \mathbf{1}_A(\mathbf{x}) \mathbf{1}_B(\mathbf{x} \cup \{u\}) \lambda(du) \mu(d\mathbf{x}) + \int_{\Psi} \mathbf{1}_A(\mathbf{x}) \sum_{u \in \mathbf{x}} \mathbf{1}_B(\mathbf{x} \setminus \{u\}) \mu(d\mathbf{x}).$$

En remarquant que par définition, $\Psi = \bigcup_n \Psi_n$, on a alors :

$$\xi_{nm}(A, B) = \sum_n \int_{\Psi_n} \int_{u \in U} \mathbf{1}_{A \cap \Psi_n}(\mathbf{x}) \mathbf{1}_{B \cap \Psi_{n+1}}(\mathbf{x} \cup \{u\}) \lambda(du) \mu(d\mathbf{x}) + \sum_n \int_{\Psi_n} \mathbf{1}_{A \cap \Psi_{n+1}}(\mathbf{x}) \sum_{u \in \mathbf{x}} \mathbf{1}_{B \cap \Psi_n}(\mathbf{x} \setminus \{u\}) \mu(d\mathbf{x}).$$

On pose alors $A_n = A \cap \Psi_n$, $B_n = B \cap \Psi_n$. On obtient donc pour $\xi_{nm}(A, B)$ et $\xi_{nm}(B, A)$:

$$\begin{aligned} \xi_{nm}(A, B) &= \sum_n \int_{\Psi_n} \int_{u \in U} \mathbf{1}_{A_n}(\mathbf{x}) \mathbf{1}_{B_{n+1}}(\mathbf{x} \cup \{u\}) \lambda(du) \mu(d\mathbf{x}) + \sum_n \int_{\Psi_n} \mathbf{1}_{A_n}(\mathbf{x}) \sum_{u \in \mathbf{x}} \mathbf{1}_{B_{n-1}}(\mathbf{x} \setminus \{u\}) \mu(d\mathbf{x}) \\ &= \sum_n \xi_{nm}(A_n, B_{n+1}) + \xi_{nm}(A_n, B_{n-1}) \end{aligned}$$

et

$$\begin{aligned} \xi_{nm}(B, A) &= \sum_n \int_{\Psi_n} \int_{u \in U} \mathbf{1}_{B_n}(\mathbf{x}) \mathbf{1}_{A_{n+1}}(\mathbf{x} \cup \{u\}) \lambda(du) \mu(d\mathbf{x}) + \sum_n \int_{\Psi_n} \mathbf{1}_{B_n}(\mathbf{x}) \sum_{u \in \mathbf{x}} \mathbf{1}_{A_{n-1}}(\mathbf{x} \setminus \{u\}) \mu(d\mathbf{x}) \\ &= \sum_n \xi_{nm}(B_n, A_{n+1}) + \xi_{nm}(B_n, A_{n-1}). \end{aligned}$$

Il suffit alors, pour obtenir la symétrie de $\xi_{nm}(\cdot, \cdot)$, c'est à dire l'égalité de $\xi_{nm}(A, B)$ et $\xi_{nm}(B, A)$, de prouver que

$$\xi_{nm}(A_{n+1}, B_n) = \xi_{nm}(B_n, A_{n+1}).$$

Or, d'après la définition de la mesure $\mu(\cdot)$ de référence (2.3)

$$\begin{aligned}
\xi_{nm}(A_{n+1}, B_n) &= \frac{e^{-\lambda(U)}}{(n+1)!} \int_{U^{n+1}} \mathbf{1}_{A_{n+1}}(\mathbf{x}) \sum_{u \in \mathbf{x}} \mathbf{1}_{B_n}(\mathbf{x} \setminus \{u\}) \lambda^{n+1}(d\mathbf{x}) \\
&= \frac{e^{-\lambda(U)}}{(n+1)!} \int_{U^{n+1}} \mathbf{1}_{A_{n+1}}(x_1, \dots, x_n, x_{n+1}) \sum_{i=1}^{n+1} \mathbf{1}_{B_n}(\mathbf{x} \setminus \{x_i\}) \lambda(dx_1) \dots \lambda(dx_{n+1}) \\
&= \frac{e^{-\lambda(U)}}{(n+1)!} \int_{U^{n+1}} (n+1) \mathbf{1}_{A_{n+1}}((x_1, \dots, x_n) \cup \{x_{n+1}\}) \mathbf{1}_{B_n}(x_1, \dots, x_n) \lambda(dx_1) \dots \lambda(dx_{n+1}) \\
&= \frac{e^{-\lambda(U)}}{n!} \int_{U^n} \int_U \mathbf{1}_{B_n}(x_1, \dots, x_n) \mathbf{1}_{A_{n+1}}((x_1, \dots, x_n) \cup \{x_{n+1}\}) \lambda(dx_{n+1}) \lambda(dx_1) \dots \lambda(dx_n) \\
&= \xi_{nm}(B_n, A_{n+1}).
\end{aligned}$$

Il nous faut désormais calculer la dérivée de Radon Nikodym pour nos deux noyaux.

► Dérivées de Radon Nikodym

Une autre démonstration des dérivées de Radon Nikodym pour les noyaux naissance/mort est disponible dans [3]. Elle décompose le noyau $Q_{nm}(\cdot, \cdot)$ en une somme de noyaux $Q_k(\cdot, \cdot)$ n'agissant que sur les configurations $\mathbf{x} \in \Psi_k$.

▷ Cas d'une naissance

A partir de la configuration courante \mathbf{x}_t , on propose la naissance λ -uniforme d'un objet $u \in U$, à travers la configuration $\mathbf{y} = \mathbf{x}_t \cup \{u\}$.

La densité objectif $\pi(\cdot)$ est supposée avoir une densité non normalisée $h(\cdot)$ par rapport au processus de Poisson de référence, comme décrit dans les parties antérieures. Le calcul de la dérivée de Radon Nikodym devient alors :

$$\begin{aligned}
f(\mathbf{x}_t, \mathbf{y}) &= \frac{\pi(d\mathbf{x}_t) Q_{nm}(\mathbf{x}_t, d\mathbf{y})}{\xi_{nm}(d\mathbf{x}_t, d\mathbf{y})} \\
&= \frac{h(\mathbf{x}_t) \mu(d\mathbf{x}_t) p_n Q_n(\mathbf{x}_t, d\mathbf{y})}{\mu(d\mathbf{x}_t) \lambda(du)} \\
&= \frac{h(\mathbf{x}_t) p_n \lambda(du)}{\lambda(U) \lambda(du)} \\
&= \frac{h(\mathbf{x}_t) p_n}{\lambda(U)}
\end{aligned}$$

▷ Cas d'une mort

A partir de la configuration courante \mathbf{x}_t , on propose la mort d'un objet $u \in \mathbf{x}_t$, à travers la configuration $\mathbf{y} = \mathbf{x}_t \setminus \{u\}$.

La dérivée de Radon Nikodym s'écrit :

$$\begin{aligned}
 f(\mathbf{x}_t, \mathbf{y}) &= \frac{\pi(d\mathbf{x}_t)Q_{nm}(\mathbf{x}_t, d\mathbf{y})}{\xi_{nm}(d\mathbf{x}_t, d\mathbf{y})} \\
 &= \frac{h(\mathbf{x}_t)\mu(d\mathbf{x}_t)p_m Q_m(\mathbf{x}_t, d\mathbf{y})}{\mu(d\mathbf{x}_t)} \\
 &= \frac{h(\mathbf{x}_t)p_m}{n(\mathbf{x}_t)}
 \end{aligned}$$

► Calcul du rapport de Green

En gardant les mêmes notations que précédemment, le calcul du rapport de Green est maintenant immédiat :

▷ Cas d'une naissance

$$\boxed{R(\mathbf{x}_t, \mathbf{x}_t \cup \{u\}) = \frac{f(\mathbf{x}_t \cup \{u\}, \mathbf{x}_t)}{f(\mathbf{x}_t, \mathbf{x}_t \cup \{u\})} = \frac{p_m}{p_n} \frac{h(\mathbf{x}_t \cup \{u\})}{h(\mathbf{x})} \frac{\lambda(U)}{n(\mathbf{x}) + 1}} \quad (2.24)$$

▷ Cas d'une mort

$$\boxed{R(\mathbf{x}_t, \mathbf{x}_t \setminus \{u\}) = \frac{f(\mathbf{x}_t \setminus \{u\}, \mathbf{x}_t)}{f(\mathbf{x}_t, \mathbf{x}_t \setminus \{u\})} = \frac{p_n}{p_m} \frac{h(\mathbf{x}_t \setminus \{u\})}{h(\mathbf{x})} \frac{n(\mathbf{x})}{\lambda(U)}} \quad (2.25)$$

Ainsi donc, l'algorithme de Green, une fois les deux mouvements de naissance et de mort définis, simule une chaîne de Markov ayant pour **distribution stationnaire** $\pi(\cdot)$.

► Irréductibilité de la chaîne

On suppose toujours que la densité objectif $\pi(\cdot)$ possède une densité que l'on note $h(\cdot)$ par rapport au processus de Poisson de référence.

Définition 2.19

On appelle **ensemble borné** de l'espace des configurations Ψ , tout ensemble A tel que

$$\exists m : \forall \mathbf{x} \in A, n(\mathbf{x}) \leq m.$$

Proposition 2.20 : Irréductibilité, Ensembles petits

Si la densité non normalisée $h(\cdot)$ vérifie la condition de stabilité (2.11), alors l'algorithme de Green simule une chaîne de Markov irréductible et tout ensemble borné de Ψ est petit.

◇ Démonstration :

Cette démonstration est disponible dans l'annexe (C.2).

◇

Ce résultat démontre que l'algorithme de Green assure l'*irréductibilité* de la chaîne de Markov.

► Récurrence de Harris et Ergodicité géométrique

Définition 2.21

Une chaîne de Markov est dite **ergodique géométriquement**, s'il existe une constante $r > 1$ telle que $\forall \mathbf{x} \in \Psi$:

$$\sum_{n=1}^{\infty} r^n \|K^n(\mathbf{x}, \cdot) - \pi(\cdot)\| < \infty.$$

Un résultat de [23] nous donne une condition suffisante pour qu'une chaîne de Markov soit ergodique géométriquement.

Proposition 2.22 : Condition de drift géométrique

Soit une chaîne de Markov de noyau de transition $K(\cdot, \cdot)$. S'il existe une fonction $V : \Psi \rightarrow [1, \infty)$, des constantes $a < 1$ et $b < \infty$, ainsi qu'un ensemble petit $\mathcal{C} \in \mathcal{B}(\Psi)$, tels que $\forall \mathbf{x} \in \Psi$

$$KV(\mathbf{x}) = \int_{\Psi} K(\mathbf{x}, d\mathbf{y}) V(\mathbf{y}) \leq aV(\mathbf{x}) + b\mathbf{1}_{\mathcal{C}}(\mathbf{x}), \quad (2.26)$$

alors la chaîne de Markov est ergodique géométriquement.

D'après les rappels sur la théorie ergodique dans l'Annexe B, l'apériodicité, l'*irréductibilité* et la **récurrence au sens de Harris** impliquent la convergence en variation totale de la chaîne.

Le résultat suivant nous permettra donc de conclure sur la convergence en variations totales de la chaîne.

Proposition 2.23 : Récurrence de Harris, Ergodicité géométrique

Si la densité non normalisée $h(\cdot)$ vérifie la condition de stabilité (2.11), alors l'algorithme de Green simule une chaîne de Markov récurrente au sens de Harris et ergodique géométriquement.

◇ Démonstration :

Cette démonstration est disponible dans l'annexe (C.3).

◇

► Conclusion

L'algorithme Metropolis-Hastings-Green réunit bien toutes les propriétés suffisantes à démontrer sa convergence en variations totales vers la mesure objectif $\pi(\cdot)$, si la condition de stabilité (2.11) tient.

Une première simulation par cet algorithme MHG que nous avons menée, fut celle du processus de Poisson homogène décrit plus haut, et dont la Fig. (2.2) est l'une des réalisations. Nous pouvons remarquer sur la Fig. (2.8), que le nombre moyen d'objets du processus tend bien vers l'espérance (2.2) après un grand nombre d'itérations (algorithme arrêté après 5 millions d'itérations), ce qui est conforme à nos attentes.

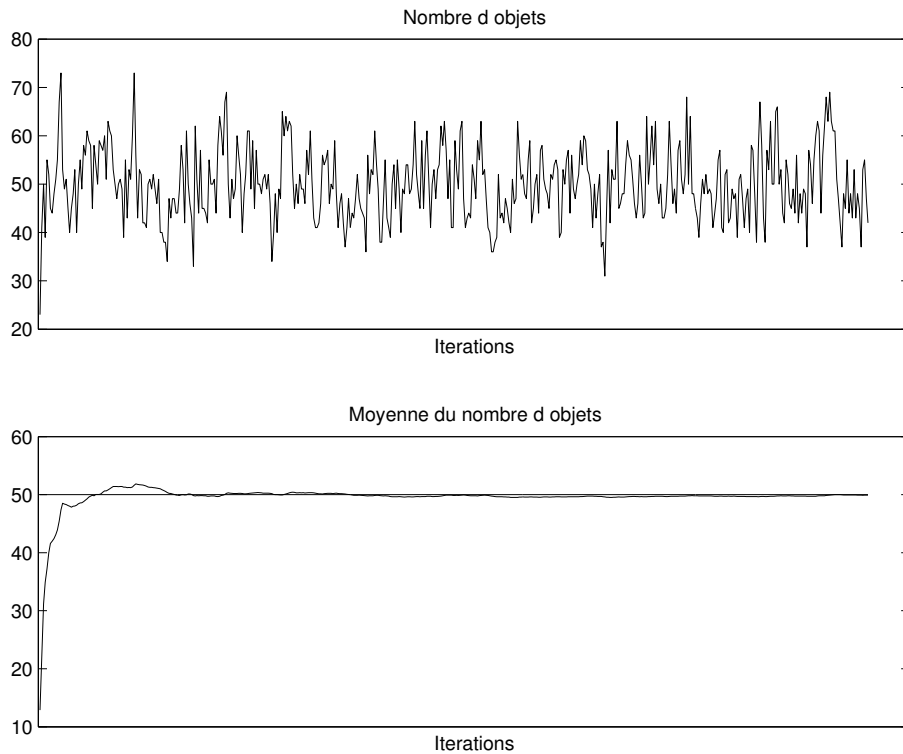


FIG. 2.8 – *Statistiques d'un processus de Poisson homogène, $a=0.005$.*

2.2.3 Quelques pistes à explorer

L'algorithme de Green n'est bien entendu pas le seul permettant de simuler une chaîne de Markov, ne connaissant que la densité non normalisée de la loi objectif. De même, certaines améliorations de l'algorithme MHG sont envisageables. Cette partie expose différentes pistes, en cours d'exploration, qui pourront se révéler utiles à notre étude.

► Simulation parfaite

Ces méthodes de simulation nous donnent un critère d'évaluation de la convergence de la chaîne de Markov vers la loi objectif. Dans ses travaux, [21] et [22] en particulier, Lund applique ce modèle à la détection de sommets d'arbres.

De même, la question du contrôle de la convergence vers la loi objectif de notre algorithme de simulation n'a pas encore été abordée. Elle est développée dans plusieurs ouvrages, notamment [27], et mérite d'être étudiée profondément dans le futur.

► Rejet reporté

Dans [12], Green et Mira introduisent une variante de l'algorithme MHG. Supposons que nous sommes dans l'état \mathbf{x}_t , et qu'une proposition de mouvement vers un état \mathbf{y} vient d'être rejetée. L'algorithme MHG, dans sa forme classique, consisterait à oublier ce rejet, et à proposer un nouveau mouvement à partir de \mathbf{x}_t . Ceci a pour conséquence d'augmenter l'autocorrélation de la chaîne de Markov, ainsi que le temps de convergence.

Dans leur nouvelle approche, Green et Mira, après le rejet du premier mouvement, nous suggèrent

d'effectuer un nouvel essai, selon une distribution qui peut dépendre du premier état rejeté. Cette seconde proposition augmente la probabilité de quitter l'état courant. L'article de Green et Mira nous explique comment s'assurer de la réversibilité de la chaîne malgré tout, propriété indispensable pour la convergence.

Typiquement, cela peut servir dans le cas où l'on se retrouve bloqué dans un état, et que le noyau de propositions ne nous permet d'en sortir qu'avec une trop faible probabilité. L'information contenue dans les premiers rejets peut se révéler utile au calcul de nouvelles distributions de noyaux de propositions, et augmenter la dynamique de la chaîne.

Notre modèle pour la détection des houppiers

★★★

Comme annoncé dans la partie précédente, nous allons nous placer dans le cadre d'un processus objet simulé par l'algorithme de Green. Plusieurs étapes sont nécessaires à la définition de notre modèle géométrique.

Tout d'abord, il convient de spécifier les objets d'intérêt sur lesquels nous allons travailler, et qui composeront les configurations que nous rechercherons dans notre image. Puis, nous étudierons quelle loi objectif nous devons donner à la chaîne de Markov, afin d'obtenir les résultats escomptés. Enfin, les mouvements du noyau de propositions seront décrits, et nous aborderons le problème de la convergence de l'algorithme.

Nous veillerons à conserver les notations introduites au cours des parties précédentes, sauf avis contraire.

★★★

3.1. Analyse des données : définition des objets d'intérêt

Pour observer des objets tels que les arbres, une résolution submétrique est indispensable. En effet, l'ordre de grandeur de la dimension transversale (horizontale) variant de quelques mètres pour certains, à une dizaine de mètres pour d'autres, il est essentiel de travailler sur de tels supports afin d'obtenir une image de qualité visuelle suffisante.

Les données sur lesquelles tout notre travail repose seront donc des images aériennes **THR**¹.

3.1.1 Des sources d'images THR

Les nouvelles données numériques de télédétection THR ouvrent des perspectives nouvelles dans la connaissance des arbres et des forêts, que n'ont pas été capables d'explorer les satellites d'observation à résolution décimétrique qui opéraient jusqu'alors.

L'évolution du contexte technologique nous permet aujourd'hui d'envisager l'acquisition de données THR, par le biais du scannage de photographies aériennes analogiques, de l'utilisation de caméras numériques 4.000 * 4.000 pixels (opérationnelles à l'IGN), ou de satellites THR. Ces derniers ont récemment fait leur apparition sur le marché par le biais d'Ikonos (résolution de 1m), et sont en plein développement (futurs satellites Pléiades, de résolution 0,7m) .

► Images satellitaires

Auparavant, les satellites nous offraient des images de résolution 30m (Landsat TEM), 20m (SPOT XS), 15m (Landsat Pan), 10m (SPOT Pan), ou 5,8m (IRS-ID, codage sur 6 bits seulement, au lieu de 8 bits pour les autres). Aussi les objets identifiés (surfaces d'eau, champs, forêts) avaient-ils des dimensions trop élevées pour pouvoir étudier des arbres isolés, ainsi que la structure interne (densité des arbres, espacement) des forêts.

A l'heure actuelle, très rares sont les satellites commerciaux pouvant nous offrir une résolution submétrique. Sur le marché, on trouve aujourd'hui le satellite américain Ikonos, à un mètre de résolution en mode panchromatique. Il s'agit du plus utilisé en haute résolution. On peut également citer le satellite israélien Eros A, qui atteint 1.8m de résolution, toujours en mode panchromatique. Mais dans un futur proche, de grands espoirs sont placés sur les satellites Pléiades dont le CNES² sera maître d'oeuvre. Pour avoir un ordre d'idées de ce qui se fait, voici la gamme proposée par Spot Image en 2002 :

Produits Spot Image	Satellite	Nombre de bandes	Résolution
2.5m couleurs	Spot 5	3 bandes	2.5m
2.5m noir et blanc	Spot 5	1 bande	2.5m
5m couleurs	Spot 5	3 bandes	5m
5m noir et blanc	Spot 5	1 bande	5m
10m couleurs	Spot 4 et 5	3 bandes	10m
10m noir et blanc	Spot 1 à 4	1 bande	10m
20m couleurs	Spot 1 à 4	3 bandes (4 pour Spot 4)	20m

TAB. 3.1 – *Produits SPOT en juillet 2002.*

D'après cette table, la plus fine image que puisse aujourd'hui fournir Spot 5 est une image de résolution 2.5m, couleurs ou noir et blanc. Ces produits à 2.5m sont obtenus après fusion "en quin-

1. Très Haute Résolution

2. Centre National d'Etudes Spatiales

conce” de deux images panchromatiques de résolution 5m acquises simultanément. Pour la couleur, il faut en outre une image en mode multispectral (3 bandes : typiquement vert, rouge, et proche infra-rouge) de résolution 10m, ce qui fait 3 images à acquérir.

Le mode panchromatique se caractérise par une observation réalisée dans une seule bande spectrale située dans la partie visible du spectre électromagnétique. La bande panchromatique de Spot 5 est comprise entre $0.48\mu\text{m}$ et $0.71\mu\text{m}$, ce qui représente une largeur de bande deux fois plus élevée que celle des 3 bandes des images multispectrales.

Nous nous retrouvons face à un problème physique de tout premier plan, les capteurs ne pouvant pas indéfiniment gagner en précision sur le domaine spatial sans perdre en précision sur le domaine fréquentiel.

► Images aériennes

Les photographies aériennes qui nous serviront de base dans notre étude, et sur lesquelles travaillent les experts de l'*Inventaire Forestier National*, sont des clichés analogiques de $23\text{cm} \times 23\text{cm}$, que l'on peut scanner jusqu'à une précision de 50cm au sol.

C'est d'ailleurs la résolution qui est atteinte pour nos images, scannées avec une définition de 600 dpi³, et desquelles on extrait des zones d'environ 1800×2400 pixels, qui représentent sur le terrain un espace rectangulaire de 126ha (0.54m par pixel).

Les 3 bandes de fréquences sont situées entre 520 et 900 νm , le bleu est donc filtré. Une des bandes spectrales des capteurs aériens est la bande proche infra-rouge (700 à 900 νm), qui fait ressortir sur l'image les chlorophylliens, c'est à dire les feuillus. Ceux-ci auront en effet une teinte plus rouge, alors que les résineux auront une teinte plus foncée. Les deux autres sont de 520 à 620 νm (vert-jaune) et de 620 à 700 νm (rouge).

La Fig. (3.1) est la première donnée réelle sur laquelle nous avons testé nos algorithmes. C'est une zone de forêts se trouvant en Saône et Loire. La Fig. (3.2) est extraite de cette image. A droite de celle-ci se trouve une zone de peupliers sur laquelle nous allons travailler. A gauche, on a une zone dite de taillis sous futaie, c'est à dire un étage supérieur, la futaie, composée ici de chênes principalement, et un étage inférieur, sans doute composé de charmes et de chênes. Cette structure étagée est classique dans nos forêts.

3.1.2 Objectifs fixés

► Aide apportée aux experts

En collaboration avec l'IFN et le Cemagref, cette étude sur la détection des houppiers devra mettre en évidence des processus automatiques d'interprétation d'images aériennes de forêts.

Bien que l'objectif soit d'automatiser ces traitements, la présence d'un expert restera de toute évidence indispensable, tant pour donner des renseignements utiles à l'algorithme, comme par exemple lui apporter des connaissances géologiques du terrain (types d'arbres que l'on peut y trouver, pente, ...), que pour valider ses résultats.

Aussi, notre objectif à long terme sera avant tout d'aider les experts forestiers, et de leur faire gagner un temps précieux dans leur travail de reconnaissance sur ordinateur.

3. dots per inch = points/inch (1 inch \simeq 2.54 cm)



FIG. 3.1 – Image aérienne d'un peuplement de Saône et Loire (1878 × 2370 pixels) ©IFN.



FIG. 3.2 – *Image de Saône et Loire : à gauche des taillis sous futaies principalement composés de chênes, à droite des peupliers (438 × 365 pixels) © IFN.*

► Différents paramètres

Plusieurs paramètres, naturels pour la plupart, sont à prendre en considération au cours de notre étude. Ceux-ci jouent en effet un rôle capital dans l'interprétation des images aériennes, et nous devront nous adapter spécifiquement à chacun d'eux dans nos algorithmes.

Il s'agit principalement :

1. des espèces d'arbres distinctes présentes sur le cliché. Nous serons en effet amenés à étudier des images pouvant contenir différentes espèces d'arbres. Mais la variabilité intraspécifique (variation de tailles, de forme, liées à l'âge, au sein d'une espèce) n'est pas pour autant à négliger, elle est même toute aussi importante que celle interspécifique (entre espèces) selon les experts.
2. du nombre d'arbres, ainsi que de la densité de la zone de forêt étudiée.
3. de l'illumination.
4. de la pente du terrain, qui avec l'illumination peut créer des jeux d'ombres difficiles à faire interpréter par un algorithme automatique.

A long terme, nous espérons être en mesure de traiter des images faisant intervenir des variations de ces paramètres, c'est à dire de créer un algorithme sachant s'adapter à différentes illuminations, différentes prises de vue, et différentes espèces d'arbres.

► Premier champ d'investigation : les peupleraies

Dans un premier temps, il a été convenu avec l'IFN et le Cemagref de se concentrer sur une première espèce d'arbres : les peupliers.

Les peupliers recouvrent quelques 200.000 hectares du territoire français, regroupés en zones que l'on nomme des **peupleraies**. Ils poussent typiquement dans les fonds des vallées, les pieds dans l'eau, sur des terrains par conséquent plats.

De plus, la connaissance a priori des peupleraies est riche. La plupart du temps, ces arbres plantés par l'homme suivent en effet un certain alignement, et sont espacés régulièrement. Leur espacement est tel que la plupart du temps, on observe un espace vide entre deux arbres sur les images. Cette information se révélera très utile dans la mise en place de notre modèle.

Il s'agit donc d'un cas assez simple, du moins en comparaison avec d'autres, sur lequel nous engageons nos séries de tests. Une fois l'étude des peupliers bien avancée, au cours de la thèse qui suivra ce stage, nous pourrons aborder d'autres situations plus complexes.

3.1.3 Objets d'intérêt

► Approche objet

L'approche objet prend tout son sens dans notre domaine d'application, la détection de houppiers. En effet, les objets du processus que nous simulerons seront justement l'objet arbre.

Plusieurs niveaux d'étude sont alors envisageables : de l'arbre à la forêt, en passant par les bosquets. Toute l'information que nous pourrions donner sur les bosquets d'arbres contenus dans la

forêt, ainsi que sur les arbres les composant, sera intéressante.

► Modèle géométrique

Sur la partie droite de l'image de la Fig. (3.2), on s'aperçoit que la forme géométrique des peupliers est très difficile à déterminer avec exactitude, d'autant plus qu'elle varie d'un arbre à l'autre. Le choix du modèle géométrique de nos objets est donc une première décision d'importance à prendre.

Plusieurs possibilités de modélisation de nos objets s'offrent à nous :

1. forme polygonale: on pourrait tout à fait modéliser chaque arbre par un polygone, d'un nombre indéterminé de côtés. Cependant, étant donné qu'un peuplier ne représente sur l'image qu'une dizaine de pixels, ce serait inutilement se compliquer la tâche, pour une amélioration peu sensible. En outre, l'espace objet serait alors difficilement identifiable.
2. forme elliptique: dans ce cas, l'objet est caractérisé par sa position, la dimension du grand axe, la dimension du petit axe, ainsi que son orientation.
3. forme circulaire: cas particulier du précédent, celui-ci a pour avantage de ne faire entrer en jeu que trois variables pour caractériser chacun des objets, puisque l'orientation n'existe plus, et que les deux dimensions sont les mêmes.

Nous avons opté dans un premier temps, pour une modélisation circulaire des peupliers. Ainsi, chaque objet $u \in U$ est parfaitement caractérisé par la position de son centre dans l'image (X_u, Y_u) et par son rayon R_u . L'espace objet sur lequel nous travaillerons sera par conséquent un sous-ensemble de \mathbb{R}^3 qu'il faudra déterminer.

► L'espace objet U

▷ Espace des positions \mathcal{P}

Les images de données étant finies, les coordonnées des objets seront inscrites dans un pavé de \mathbb{R}^2 . On note \mathcal{P} cet espace des positions: $\mathcal{P} = [0, X_M] \times [0, Y_M]$, avec $X_M > 0$ et $Y_M > 0$.

▷ Espace des marques \mathcal{K}

Il en sera de même pour les marques des objets, c'est à dire les rayons des cercles. On peut raisonnablement supposer qu'aucun arbre n'aura une dimension supérieure à R_M , ni inférieure à R_m (pouvant être pris égal à 0 si nous n'avons aucune information supplémentaire), avec $R_m < R_M$.

L'espace des marques, noté \mathcal{K} , est alors simplement $\mathcal{K} = [R_m, R_M]$.

L'espace objet a donc la forme suivante :

$$U = \mathcal{P} \times \mathcal{K} = [0, X_M] \times [0, Y_M] \times [R_m, R_M].$$

Pour définir un élément u de U , on écrira $u = (X_u, Y_u, R_u)$. On emploiera aussi la notation $C_u = (X_u, Y_u) \in \mathcal{P}$ pour désigner son centre dans l'espace des positions.

L'espace des configurations Ψ est défini de la même façon que dans la partie précédente, à partir de notre espace objet U .

3.2. Modèle de la loi objectif

3.2.1 Cadre Bayésien

► Processus ponctuel de référence

Le processus ponctuel de référence, conformément à notre exposé de la partie précédente, sera de Poisson de mesure d'intensité $\lambda(\cdot)$. Cette mesure $\lambda(\cdot)$ sur l'espace objet sera la mesure de Lebesgue définie sur \mathbb{R}^3 .

La loi de ce processus sera notée $\mu(\cdot)$. Nous rappelons que, par définition d'un processus objet, il doit être processus ponctuel sur l'espace U ainsi que sur l'espace des positions. De plus, la loi $\mu(\cdot)$ s'écrit

$$\mu(B) = e^{-\lambda(U)} \left(\mathbf{1}_{[\emptyset \in B]} + \sum_{n=1}^{\infty} \frac{\mu_n(B)}{n!} \right)$$

avec

$$\mu_n(B) = \int \cdots \int \mathbf{1}_{\{u_1, \dots, u_n\} \in B} \lambda(du_1) \cdots \lambda(du_n).$$

► Densité à simuler

Soit $h(\cdot)$ la densité du processus ponctuel que l'on souhaite simuler. La mesure objectif $\pi(\cdot)$, mesure invariante de la chaîne de Markov, s'écrira alors :

$$\pi(d\mathbf{x}) = h(\mathbf{x})\mu(d\mathbf{x}).$$

L'expression de la densité $h(\cdot)$ est assurément un des points-clés de notre démarche. Non seulement elle doit contenir les informations suffisantes nous permettant de détecter les houppiers sur les images de forêts, mais aussi elle doit vérifier la condition de stabilité (2.11) que l'on rappelle ici :

$$h(\mathbf{x} \cup \xi) \leq M_h h(\mathbf{x}), \forall \mathbf{x} \in \Psi, \forall \xi \in U.$$

Sous cette hypothèse, ainsi que celle de réversibilité, nous sommes assurés de la convergence de la chaîne de Markov vers la mesure objectif $\pi(\cdot)$.

► Densité a posteriori

La densité $h(\mathbf{x})$ modélise en réalité la densité de probabilité de la configuration \mathbf{x} , étant donnée l'image \mathcal{I} de départ. Nous l'écrivons $h(\mathbf{x})$:

$$h(\mathbf{x}) = h(\mathbf{x}|\mathcal{I}).$$

Cette écriture est celle de la densité a posteriori des configurations. Dans un cadre Bayésien, on peut alors appliquer la règle de Bayes :

$$h(\mathbf{x}) = \frac{h_p(\mathbf{x})\mathcal{L}(\mathcal{I}|\mathbf{x})}{f(\mathcal{I})} \propto h_p(\mathbf{x})\mathcal{L}(\mathcal{I}|\mathbf{x}). \quad (3.1)$$

L'écriture bayésienne est très robuste mathématiquement. Cependant elle nécessite d'être capable d'exprimer la vraisemblance de l'image étant donnée la configuration \mathbf{x} , $\mathcal{L}(\mathcal{I}|\mathbf{x})$, c'est à dire qu'à partir d'une configuration d'objets nous devons être en mesure de reconstituer une image de données. Le terme $h_p(\mathbf{x})$ est quant à lui une connaissance a priori sur les configurations.

Nous rechercherons en fait la "meilleure" configuration d'objets possible sur notre image \mathcal{I} . Par la suite, nous expliquerons comment nous avons choisi d'interpréter ce terme "meilleur". Nous verrons que dans notre approche, cela consiste à chercher le maximum a posteriori

$$\mathbf{x}^*(\mathcal{I}) = \arg \max_{\mathbf{x} \in \Psi} (h(\mathbf{x}))$$

c'est à dire la configuration maximisant la densité a posteriori $h(\cdot)$.

Enfin, étant donné que nous travaillerons sur une densité connue à un facteur multiplicatif près, nous écrirons simplement

$$h(\mathbf{x}) = h_p(\mathbf{x})\mathcal{L}(\mathcal{I}|\mathbf{x}). \quad (3.2)$$

Nous consacrerons les deux sous-parties suivantes aux choix de la densité a priori et du terme de vraisemblance.

3.2.2 Terme a priori $h_p(\mathbf{x})$

Ce terme renseigne sur toutes les connaissances a priori des configurations que l'on recherche. Par exemple, on peut y inclure un terme répulsif de Strauss, afin de s'affranchir des configurations comportant des objets qui se chevauchent.

Une bonne définition des contraintes que l'on souhaite imposer aux configurations nous permettrait d'accélérer la convergence de l'algorithme : au cours des itérations, nous n'accepterions en effet que les configurations intéressantes au sens de cet a priori et pourrions nous retrouver plus rapidement dans un mode de la densité a posteriori.

► Contrôle sur le nombre d'objets

Un terme de contrôle sur l'intensité du processus, et donc influant sur l'espérance du nombre d'objets détectés, doit être inclus dans l'expression de la densité a priori.

Celui-ci s'écrit simplement $\beta^{n(\mathbf{x})}$, avec $\beta > 1$ si l'on désire favoriser la naissance d'objets dans la configuration, ou $\beta < 1$ dans le cas contraire.

$$h_p(\mathbf{x}) \propto \beta^{n(\mathbf{x})}$$

En fait, ce terme revient à biaiser le terme de vraisemblance, c'est pourquoi nous verrons par la suite qu'il est préférable de le lier à la mesure de référence $\mu(\cdot)$.

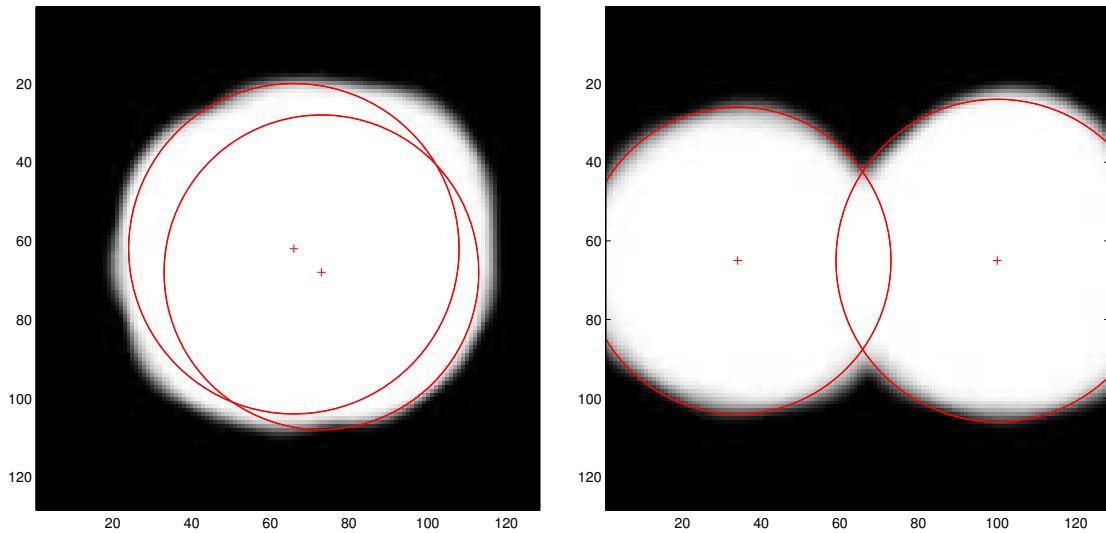


FIG. 3.3 – *A gauche : Cas de superposition à éviter, les deux cercles détectant le même objet. A droite : Superposition acceptable, deux objets distincts étant délimités.*

► Pénalisations envisageables

Partant du principe que l'un de nos objectifs est de dénombrer les arbres, il faut éviter de compter deux fois le même. Ainsi, une configuration comportant deux cercles se superposant, et détectant le même houppier, est à bannir. Par contre, une configuration avec deux cercles se superposant en partie, et détectant deux houppiers distincts, doit être possible si nous étudions une image où les arbres sont très proches les uns des autres. Ces deux cas de figure sont représentés sur la Fig. (3.3).

Afin de caractériser dans l'espace des objets U le phénomène de superposition décrit ci-dessus, nous allons introduire quelques notions telles qu'une relation d'équivalence d'intersection sur U , et la silhouette d'un objet ou d'une configuration.

► Caractérisation des intersections d'objets dans U

Tout d'abord, il s'agit de définir un lien entre l'espace U et \mathbb{R}^2 , espace dans lequel ont lieu ces intersections.

Définition 3.1

On appelle **silhouette d'un objet** $u \in U$ l'ensemble $\mathcal{S}_{\mathcal{P}}(u) \in \mathcal{P} \subset \mathbb{R}^2$, intersection de l'espace des positions et de la boule fermée dans \mathbb{R}^2 (ou disque), de centre $C_u = (X_u, Y_u)$ et de rayon R_u :

$$\mathcal{S}_{\mathcal{P}}(u) = \mathbb{B}_f(C_u, R_u) \cap \mathcal{P}.$$

La silhouette d'un objet u n'est donc pas toujours un disque, celui-ci peut en effet se retrouver tronqué si le centre $C_u \in \mathcal{P}$ est à une distance inférieure à R_u des limites de l'espace des positions. Par extension, on définit la notion de silhouette d'un ensemble d'objets, ou d'une configuration

d'objets.

Définition 3.2

On appelle **silhouette d'un ensemble d'objets** $\mathcal{U} = (u_1, \dots, u_n)$ (respectivement d'une configuration d'objets $\mathbf{x} = (u_1, \dots, u_n)$), et on note $\mathcal{S}_{\mathcal{P}}(\mathcal{U})$ (resp. $\mathcal{S}_{\mathcal{P}}(\mathbf{x})$), l'ensemble défini par la réunion des silhouettes de chacun des objets :

$$\mathcal{S}_{\mathcal{P}}(\mathcal{U}) = \bigcup_{u \in \mathcal{U}} \mathcal{S}_{\mathcal{P}}(u).$$

$$\mathcal{S}_{\mathcal{P}}(\mathbf{x}) = \bigcup_{u \in \mathbf{x}} \mathcal{S}_{\mathcal{P}}(u).$$

On définit ensuite une **relation de voisinage** \sim_I , symétrique et réflexive, qui met en relation deux objets de U dont les silhouettes s'intersectent sur un ensemble non réduit à un point :

$$x_i \sim_I x_j \Leftrightarrow d(C_{x_i}, C_{x_j}) < R_{x_i} + R_{x_j},$$

où $d(.,.)$ est la distance euclidienne sur \mathbb{R}^2 .

Définition 3.3

Soient \mathbf{x} une configuration d'objets sur U , et \sim une relation d'équivalence définie sur U . On appelle **voisinage d'un objet** $u \in \mathbf{x}$ au sens de la relation \sim l'ensemble

$$\mathcal{V}_{\sim, \mathbf{x}}(u) = \{x_i \in \mathbf{x} \mid x_i \neq u, x_i \sim u\}.$$

Ainsi, le voisinage $\mathcal{V}_{\sim_I, \mathbf{x}}(u)$ sera constitué de l'ensemble des voisins de u au sens de \sim_I .

On note alors $n_{\sim_I}(\mathbf{x})$ le nombre de paires d'objets (ou cliques) d'une configuration \mathbf{x} voisins au sens de \sim_I . D'après cette définition, ainsi que celle du voisinage, on a :

$$\begin{aligned} n_{\sim_I}(\mathbf{x}) &= \sum_{x_i, x_j \in \mathbf{x}, x_i \neq x_j} \mathbf{1}_{x_i \sim_I x_j} \\ &= \frac{1}{2} \sum_{u \in \mathbf{x}} \#(\mathcal{V}_{\sim_I, \mathbf{x}}(u)). \end{aligned}$$

Définition 3.4

On appelle **ensemble d'intersection d'un objet u dans une configuration** \mathbf{x} , $u \in \mathbf{x}$, et on note $\cap_{\mathbf{x}}(u)$ l'ensemble

$$\cap_{\mathbf{x}}(u) = \mathcal{S}_{\mathcal{P}}(u) \cap \mathcal{S}_{\mathcal{P}}(\mathcal{V}_{\sim_I, \mathbf{x}}(u)).$$

Par définition du voisinage $\mathcal{V}_{\sim_I, \mathbf{x}}(u)$, on a aussi

$$\cap_{\mathbf{x}}(u) = \mathcal{S}_{\mathcal{P}}(u) \cap \mathcal{S}_{\mathcal{P}}(\mathbf{x})$$

puisque seuls les voisins de u au sens de \sim_I l'intersectent.

Cette dernière définition nous permet ainsi de caractériser, pour chaque objet u d'une configuration, le sous-ensemble de \mathcal{P} où sa silhouette intersecte celles de ses voisins.

► Différentes modélisations des interactions

D'après ce qui a été dit auparavant, nous devons veiller à ce que deux cercles de notre configuration ne se superposent pas trop. Deux pistes sont alors envisageables : nous pouvons inclure dans le terme a priori une répulsion similaire à celle introduite dans la densité de Strauss (2.13), c'est à dire de même poids sur toutes les cliques de deux objets voisins au sens de \sim_I , ou nous pouvons faire dépendre l'intensité de l'aire de contact entre les différents objets.

▷ Processus de Strauss

Celui-ci a déjà été décrit dans le chapitre d'introduction aux processus ponctuels. La densité a priori s'écrit :

$$h_p(\mathbf{x}) = \beta^{n(\mathbf{x})} e^{-\gamma_s n_{\sim_I}(\mathbf{x})}$$

avec γ_s positif suivant cette nouvelle notation, puisque nous souhaitons instaurer une répulsion entre les objets voisins.

Le principal inconvénient d'une telle modélisation est qu'elle attribue le même poids à toutes les interactions entre les objets. Sur nos images d'arbres, nous avons parfois deux houppiers qui s'intersectent légèrement, et nous voudrions les détecter sans être pénalisé de la même façon que si ces deux objets étaient confondus.

▷ Processus d'interaction d'aires

Ce processus s'intéresse aux aires de l'intersection des différents objets de la configuration. La densité a priori s'écrit alors :

$$h_p(\mathbf{x}) = \beta^{n(\mathbf{x})} e^{-\gamma_a \mathcal{A}_i(\mathbf{x})} \quad (3.3)$$

avec γ_a positif pour les mêmes raisons qu'auparavant.

Différents modèles d'interaction d'aires sont envisageables. Le choix de notre modèle parmi ceux présentés ci-dessous sera effectué par la suite.

Tout d'abord, un premier modèle consiste à s'intéresser aux paires d'objets en interaction, et à pénaliser ces aires d'intersection :

$$\mathcal{A}_1(\mathbf{x}) = \sum_{x_i \sim_I x_j} \lambda_2(\mathcal{S}_{\mathcal{P}}(x_i) \cap \mathcal{S}_{\mathcal{P}}(x_j)) \quad (3.4)$$

avec $\lambda_2(\cdot)$ la mesure de Lebesgue sur \mathbb{R}^2 .

Un second modèle consiste à ne pénaliser pour chaque objet que la plus forte interaction qu'il a avec les autres objets de la configuration \mathbf{x} :

$$\mathcal{A}_2(\mathbf{x}) = \sum_{u \in \mathbf{x}} \max(\lambda_2(\mathcal{S}_{\mathcal{P}}(u) \cap \mathcal{S}_{\mathcal{P}}(v)), v \in \mathcal{V}_{\sim_I, \mathbf{x}}(u)) \quad (3.5)$$

Enfin, un troisième modèle consiste à pénaliser l'aire de l'ensemble d'interaction de chaque objet avec la configuration \mathbf{x} :

$$\mathcal{A}_3(\mathbf{x}) = \sum_{u \in \mathbf{x}} \lambda_2(\cap_{\mathbf{x}}(u)) \quad (3.6)$$

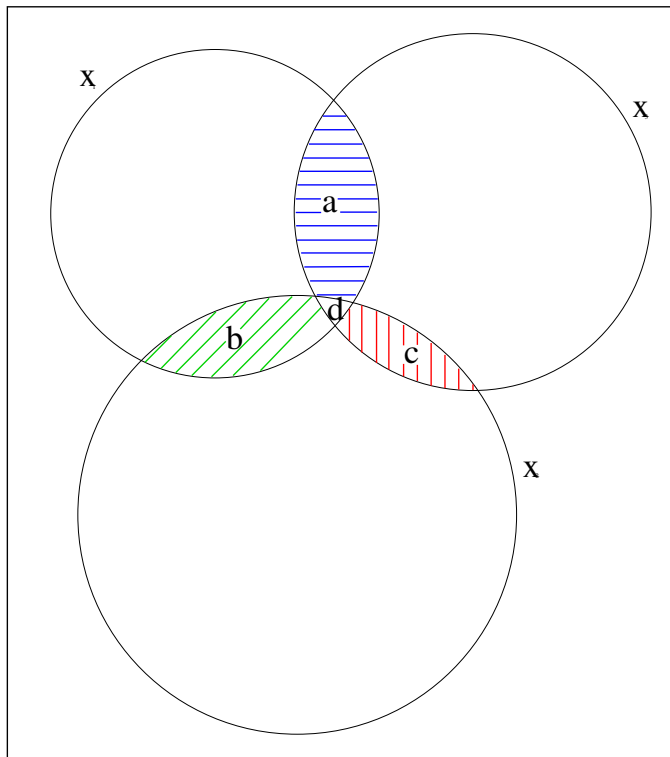


FIG. 3.4 – Exemple de configuration comportant 3 objets.

Remarquons dès maintenant que par définition de $\cap_{\mathbf{x}}(u)$, on a toujours $\mathcal{A}_3(\mathbf{x}) \geq \mathcal{A}_2(\mathbf{x})$.

Afin de bien comprendre quelles différences il y a entre ces expressions de la densité a priori, prenons deux exemples simples de configuration \mathbf{x} , et observons comment se comportent les aires d'interaction $\mathcal{A}_i(\mathbf{x})$ pour chacun d'eux.

► Exemples de calcul des $\mathcal{A}_i(\mathbf{x})$

▷ Exemple 1 : Fig. (3.4)

Avec les notations de la Fig. (3.4), on suppose vérifiée la relation suivante entre les aires : $a > b > c > d$.

D'après les définitions des aires $\mathcal{A}_i(\mathbf{x})$:

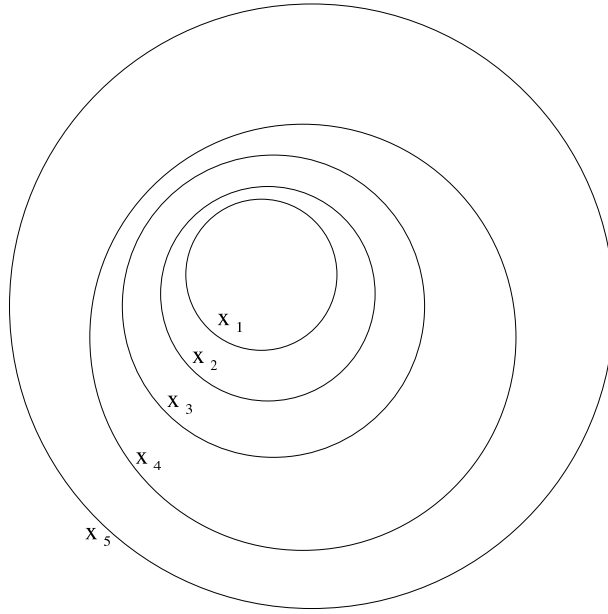


FIG. 3.5 – Exemple de configuration comportant 5 objets.

$$\begin{aligned}\mathcal{A}_1(\mathbf{x}) &= \lambda_2(\mathcal{S}_{\mathcal{P}}(x_1) \cap \mathcal{S}_{\mathcal{P}}(x_2)) + \lambda_2(\mathcal{S}_{\mathcal{P}}(x_1) \cap \mathcal{S}_{\mathcal{P}}(x_3)) + \lambda_2(\mathcal{S}_{\mathcal{P}}(x_2) \cap \mathcal{S}_{\mathcal{P}}(x_3)) \\ &= a + b + c + 3d\end{aligned}$$

$$\begin{aligned}\mathcal{A}_2(\mathbf{x}) &= \lambda_2(\mathcal{S}_{\mathcal{P}}(x_1) \cap \mathcal{S}_{\mathcal{P}}(x_2)) + \lambda_2(\mathcal{S}_{\mathcal{P}}(x_2) \cap \mathcal{S}_{\mathcal{P}}(x_1)) + \lambda_2(\mathcal{S}_{\mathcal{P}}(x_3) \cap \mathcal{S}_{\mathcal{P}}(x_1)) \\ &= 2a + b + 3d\end{aligned}$$

$$\begin{aligned}\mathcal{A}_3(\mathbf{x}) &= \lambda_2(\cap_{\mathbf{x}}(x_1)) + \lambda_2(\cap_{\mathbf{x}}(x_2)) + \lambda_2(\cap_{\mathbf{x}}(x_3)) \\ &= 2a + 2b + 2c + 3d.\end{aligned}$$

On a donc pour ce premier exemple $\mathcal{A}_1(\mathbf{x}) < \mathcal{A}_2(\mathbf{x}) < \mathcal{A}_3(\mathbf{x})$.

▷ **Exemple 2 : Fig. (3.5)**

Avec les notations de la Fig. (3.5), on suppose vérifiée la relation suivante entre les aires : $a_5 = 2a_4 = 4a_3 = 8a_2 = 16a_1$, où $a_i = \lambda_2(\mathcal{S}_{\mathcal{P}}(x_i))$. D'après les définitions des aires $\mathcal{A}_i(\mathbf{x})$:

$$\begin{aligned}\mathcal{A}_1(\mathbf{x}) &= 4a_1 + 3a_2 + 2a_3 + a_4 = 26a_1 \\ \mathcal{A}_2(\mathbf{x}) &= a_1 + a_2 + a_3 + 2a_4 = 23a_1 \\ \mathcal{A}_3(\mathbf{x}) &= a_1 + a_2 + a_3 + 2a_4 = 23a_1\end{aligned}$$

On a donc pour ce deuxième exemple $\mathcal{A}_2(\mathbf{x}) = \mathcal{A}_3(\mathbf{x}) < \mathcal{A}_1(\mathbf{x})$.

Nous voyons bien qu'en cas d'une accumulation d'objets dans une même zone de l'espace, l'aire $\mathcal{A}_1(\mathbf{x})$ risque d'exploser. En effet, une configuration similaire de p objets, avec des rapports d'aires identiques, mène à :

$$\begin{aligned}
\mathcal{A}_1(\mathbf{x}) &= (2^p - p + 1) a_1 \\
\mathcal{A}_2(\mathbf{x}) &= (3 \cdot 2^{p-2} - 1) a_1 \\
\mathcal{A}_3(\mathbf{x}) &= (3 \cdot 2^{p-2} - 1) a_1
\end{aligned}$$

Par contre, les comportements des différents modèles sont comparables dans le cas d'un petit nombre de voisins pour chaque objet, ce qui sera notre cas pour les arbres. La seule différence est que des mêmes aires ont tendance à être comptées plusieurs fois pour les deux derniers modèles. Il nous faudra utiliser un paramètre γ_a adapté au modèle choisi.

► Bilan

Au final, $h_p(\cdot)$ s'écrit :

$$h_p(\mathbf{x}) = e^{-\mathbf{E}_p(\mathbf{x})} \quad (3.7)$$

avec

$$\mathbf{E}_p(\mathbf{x}) = -\log(\beta)n(\mathbf{x}) + \gamma_a \mathcal{A}_i(\mathbf{x}). \quad (3.8)$$

Le modèle d'interaction d'aires $\mathcal{A}_i(\mathbf{x})$ sera choisi par la suite.

3.2.3 Terme d'attaches aux données $\mathcal{L}(\mathcal{I}|\mathbf{x})$

Le terme de vraisemblance $\mathcal{L}(\mathcal{I}|\mathbf{x})$ correspond à la probabilité, étant donnée une configuration d'objets \mathbf{x} , d'observer une image \mathcal{I} .

Travaillant sur des images discrètes dans [31] et [32], H. Rue fait de l'inférence bayésienne à l'aide de processus ponctuels simulés par des chaînes de Markov. Son terme de vraisemblance est intéressant, puisqu'il distingue deux classes de pixels dans l'image, selon qu'ils sont ou non contenus dans un des objets de son processus, chacune des classes ayant des propriétés distinctes.

Nous utiliserons la même démarche pour notre attache aux données, qui sera une attache au niveau pixelique : nous regarderons si tous les pixels ont correctement été classifiés. Se posent alors le problème de la discrétisation, ainsi que celui de la formulation de cette vraisemblance.

► Problème de la discrétisation

L'image de données étant discrète, et les objets de la configuration définis sur un espace U continu (plus exactement, c'est la continuité de l'espace des positions qui nous intéresse ici), il nous faut définir une relation d'attache entre un objet défini sur U , et son représentant sur l'image.

Dans la pratique, l'espace des positions est $\mathcal{P} = [0, X_M] \times [0, Y_M]$, avec X_M et Y_M entiers, correspondant à la largeur et la hauteur de l'image.

On peut visualiser l'ensemble de ces positions sur l'image en se munissant d'un repère orthonormé avec pour origine le coin en haut à gauche. Ainsi donc, le pixel $p = \{i, j\}$, $0 \leq i < X_M$, $0 \leq j < Y_M$, de la $i^{\text{ème}}$ colonne et de la $j^{\text{ème}}$ ligne, recouvre le sous-ensemble $[i, i+1] \times [j, j+1]$ de \mathcal{P} .

Définition 3.5

Soit p un pixel $\{i, j\}$. Le **centre du pixel** p est le point central de p dans l'espace des positions \mathcal{P} , c'est à dire $(i + \frac{1}{2}, j + \frac{1}{2})$.

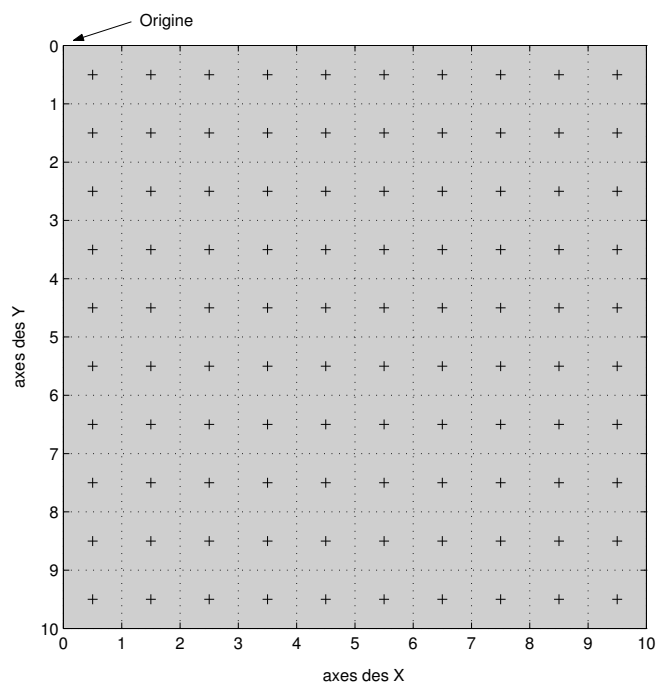


FIG. 3.6 – Repère de l'espace des positions sur la matrice d'une image \mathcal{I} de taille 10×10 . Le quadrillage sépare les pixels de l'image \mathcal{I} , et les croix représentent les centres de chacun des pixels.

Toutes ces notations sont reprises dans le schéma récapitulatif de la Fig. (3.6).

Soit alors $\mathbf{x} = (x_1, \dots, x_n)$ une configuration d'objets. Pour chacun de ses composants x_i , on peut alors définir la notion de représentant sur l'image.

Définition 3.6

On appelle **représentant sur l'image** \mathcal{I} de $x_i = (X_{x_i}, Y_{x_i}, R_{x_i})$, l'ensemble $\mathcal{T}_{\mathcal{I}}(x_i)$ des pixels de \mathcal{I} ayant leur centre dans le support de x_i .
Si les x_i sont des cercles, ces pixels vérifient l'inégalité suivante :

$$\sqrt{\left(i + \frac{1}{2} - X_{x_i}\right)^2 + \left(j + \frac{1}{2} - Y_{x_i}\right)^2} \leq R_{x_i} \implies (i, j) \in \mathcal{T}_{\mathcal{I}}(x_i).$$

La Fig. (3.7) donne un exemple d'objet circulaire et de son représentant sur l'image. La discrétisation des données, notamment sur des objets de petite taille comme des arbres (sur une image de résolution 0.5m), pose donc de nombreuses difficultés quant à leur représentation dans le domaine continu. Nous nous sommes limités à des cercles dans un premier temps, mais sans doute qu'une forme elliptique sera plus adaptée dans nos applications futures.

► Définition des classes de pixels

On peut alors attribuer à chacun des pixels de l'image \mathcal{I} une appartenance à une des deux classes suivantes :

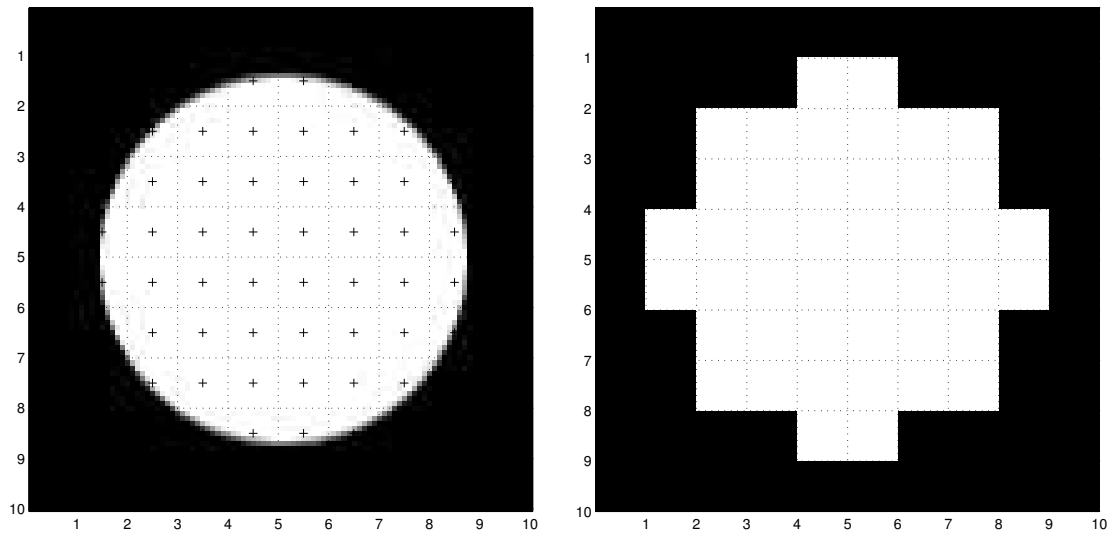


FIG. 3.7 – A gauche : objet dans l'espace continu des positions, placé sur la grille de l'image. A droite : représentant discret de l'objet.

- **La classe des arbres** \mathcal{C}_A : un pixel appartient à \mathcal{C}_A si il appartient à un des représentants $\mathcal{T}_I(x_i)$ des objets de la configuration \mathbf{x} :

$$\mathcal{C}_A = \bigcup_{x_i \in \mathbf{x}} \mathcal{T}_I(x_i).$$

- **La classe du fond** \mathcal{C}_F : un pixel appartient à cette classe si il n'appartient pas à la classe des arbres.

Autrement dit, nous avons réalisé une partition de l'image discrète en deux ensembles de pixels.

► Modélisation des classes

Dans le terme de vraisemblance de l'expression de la densité a posteriori, nous devons quantifier la vraisemblance d'une image étant donnée une configuration d'objets. Se reportant aux travaux de Rue, nous avons émis l'idée de calculer une vraisemblance sur chacun des pixels de l'image.

A partir d'une configuration \mathbf{x} , nous avons introduit précédemment les notions suffisantes pour obtenir une classification de tous les pixels, dans l'une des classes \mathcal{C}_A ou \mathcal{C}_F . Se pose désormais la question essentielle : comment juger si cette classification est bonne ?

En observant nos images, on remarque que la bande infra-rouge sépare nettement le niveau de gris des arbres et du fond. Dans une première approche, on décide alors de modéliser chacune de ces classes par une distribution normale de ses niveaux de gris :

$$\mathcal{C}_A : \mathcal{N}(\mu_A, \sigma_A), \mathcal{C}_F : \mathcal{N}(\mu_F, \sigma_F),$$

Ainsi, pour un pixel $p \in \mathcal{C}_A$, de niveau de gris y_p sur \mathcal{I} , la vraisemblance de cette classification est

$$\mathbb{P}(p \in \mathcal{C}_A) = \frac{1}{\sqrt{2\pi}\sigma_A} \exp\left(-\frac{(y_p - \mu_A)^2}{2\sigma_A^2}\right).$$

Si ce même pixel était classifié dans \mathcal{C}_F , sa vraisemblance serait

$$\mathbb{P}(p \in \mathcal{C}_F) = \frac{1}{\sqrt{2\pi}\sigma_F} \exp\left(-\frac{(y_p - \mu_F)^2}{2\sigma_F^2}\right).$$

Ce modèle pourrait tout à fait être développé si on avait plusieurs espèces d'arbres différentes. Cependant, les distributions gaussiennes de ces niveaux de gris risqueraient de se confondre, et l'évaluation de la vraisemblance serait faussée puisqu'un pixel risquerait d'être vraisemblable pour plusieurs classes.

L'information donnée par les niveaux de gris est donc très faible. Il nous faudra par la suite, lorsque nous étudierons des images comportant différentes espèces d'arbres, établir une signature robuste de chacune de ces espèces, en calculant par exemple certains **attributs de texture**.

► Formulation de la vraisemblance

La vraisemblance de l'image $\mathcal{L}(\mathcal{I}|\mathbf{x})$, puisque l'on suppose les pixels indépendants conditionnellement à leur appartenance à une classe donnée, est donc :

$$\mathcal{L}(\mathcal{I}|\mathbf{x}) = \prod_{p \in \mathcal{C}_A} \frac{1}{\sqrt{2\pi}\sigma_A} \exp\left(-\frac{(y_p - \mu_A)^2}{2\sigma_A^2}\right) \prod_{p \in \mathcal{C}_F} \frac{1}{\sqrt{2\pi}\sigma_F} \exp\left(-\frac{(y_p - \mu_F)^2}{2\sigma_F^2}\right). \quad (3.9)$$

3.2.4 Stabilité du processus

La loi objectif $\pi(\cdot)$ s'exprime alors :

$$\pi(d\mathbf{x}) = h(\mathbf{x})\mu(d\mathbf{x}) = h_p(\mathbf{x})\mathcal{L}(\mathcal{I}|\mathbf{x})\mu(d\mathbf{x}). \quad (3.10)$$

L'intensité conditionnelle de Papangelou se déduit de (3.7) et de (3.10) :

$$\begin{aligned} \frac{h(\mathbf{x} \cup \{\xi\})}{h(\mathbf{x})} &= \frac{\mathcal{L}(\mathcal{I}|\mathbf{x} \cup \{\xi\})}{\mathcal{L}(\mathcal{I}|\mathbf{x})} \frac{h_p(\mathbf{x} \cup \{\xi\})}{h_p(\mathbf{x})} \\ &= \frac{\mathcal{L}(\mathcal{I}|\mathbf{x} \cup \{\xi\})}{\mathcal{L}(\mathcal{I}|\mathbf{x})} \beta e^{-\gamma_a[\mathcal{A}_i(\mathbf{x} \cup \{\xi\}) - \mathcal{A}_i(\mathbf{x})]} \\ &= \frac{\mathcal{L}(\mathcal{I}|\mathbf{x} \cup \{\xi\})}{\mathcal{L}(\mathcal{I}|\mathbf{x})} \beta e^{-\gamma_a \Delta \mathcal{A}_i(\mathbf{x}, \xi)} \end{aligned}$$

Nous rappelons que la **majoration de cette intensité conditionnelle de Papangelou** est une condition suffisante de convergence de l'algorithme de Green.

Nous souhaitons en outre imposer une condition supplémentaire à notre modèle, la **minoration de l'intensité conditionnelle de Papangelou**. Cette condition nous sera utile pour prouver la validité d'un mouvement du noyau de propositions, c'est à dire la convergence de l'algorithme de Green avec ce noyau.

C'est pourquoi, pour chacun des modèles d'interaction d'aires $\mathcal{A}_i(\mathbf{x})$, nous allons vérifier ces deux propriétés.

▷ **Majoration de l'intensité conditionnelle de Papangelou**

D'une part, remarquons que la vraisemblance $\mathcal{L}(\mathcal{I}|\mathbf{x})$ est majorée et minorée par \mathcal{L}_{min} et \mathcal{L}_{max} définis de la façon suivante :

$$\mathcal{L}_{min} = \prod_{p \in \mathcal{C}_A \cup \mathcal{C}_F} \frac{1}{\sqrt{2\pi} \max(\sigma_A, \sigma_F)} \exp\left(\frac{-\max(|y_p - \mu_A|, |y_p - \mu_F|)^2}{2 \min(\sigma_A, \sigma_F)^2}\right) > 0 \quad (3.11)$$

et

$$\mathcal{L}_{max} = \prod_{p \in \mathcal{C}_A \cup \mathcal{C}_F} \frac{1}{\sqrt{2\pi} \min(\sigma_A, \sigma_F)} \exp\left(\frac{-\min(|y_p - \mu_A|, |y_p - \mu_F|)^2}{2 \max(\sigma_A, \sigma_F)^2}\right) > 0. \quad (3.12)$$

On a ainsi

$$\frac{\mathcal{L}(\mathcal{I}|\mathbf{x} \cup \{\xi\})}{\mathcal{L}(\mathcal{I}|\mathbf{x})} \leq \frac{\mathcal{L}_{max}}{\mathcal{L}_{min}}, \forall \xi \in \{U - \bigcup_{i=1}^{n(\mathbf{x})} x_i\}. \quad (3.13)$$

D'autre part, l'intensité conditionnelle de Papangelou du terme a priori $h_p(\cdot)$ est majorée puisque pour nos trois modèles, $\Delta \mathcal{A}_i(\mathbf{x}, \xi) \geq 0$, et donc

$$\frac{h_p(\mathbf{x} \cup \{\xi\})}{h_p(\mathbf{x})} \leq \beta, \forall \xi \in \{U - \bigcup_{i=1}^{n(\mathbf{x})} x_i\}. \quad (3.14)$$

En conclusion de (3.13) et (3.14) on obtient

$$\frac{h(\mathbf{x} \cup \{\xi\})}{h(\mathbf{x})} \leq \beta \frac{\mathcal{L}_{max}}{\mathcal{L}_{min}} = M_h, \forall \xi \in \{U - \bigcup_{i=1}^{n(\mathbf{x})} x_i\}$$

ce qui nous assure la stabilité locale du processus, et donc les bonnes propriétés de la chaîne de Monte Carlo qui le simule.

▷ **Minoration de l'intensité conditionnelle de Papangelou**

D'après les majoration et minoration de la vraisemblance, on a déjà

$$\frac{\mathcal{L}(\mathcal{I}|\mathbf{x})}{\mathcal{L}(\mathcal{I}|\mathbf{x} \cup \{\xi\})} \geq \frac{\mathcal{L}_{min}}{\mathcal{L}_{max}}, \forall \xi \in \{U - \bigcup_{i=1}^{n(\mathbf{x})} x_i\}. \quad (3.15)$$

Il nous reste à essayer de minorer l'intensité conditionnelle de Papangelou de la densité a priori $h_p(\mathbf{x})$ pour chacun des 3 modèles d'interaction d'aires présentés. Cela équivaut à majorer le terme

$\Delta\mathcal{A}_i(\mathbf{x},\xi)$.

1. **Modèle 1** : $\Delta\mathcal{A}_1(\mathbf{x},\xi)$

Notre objectif est de prouver, contre-exemple à l'appui, que nous ne pouvons pas minorer l'intensité conditionnelle de Papangelou de $h_p(\mathbf{x})$, ou de façon équivalente majorer $\Delta\mathcal{A}_1(\mathbf{x},\xi)$. Autrement dit,

$$\forall M > 0, \exists (\mathbf{x}, \xi) \in \Psi \times U \text{ tels que } \Delta\mathcal{A}_1(\mathbf{x}, \xi) > M.$$

Supposons que les dimensions de l'espace des positions \mathcal{P} et de l'espace des marques \mathcal{K} sont telles qu'un objet de marque maximale R_M puisse être positionné dans \mathcal{P} sans que sa silhouette n'intersecte les bords. Cette hypothèse peut s'écrire : $X_M \geq 2R_M$ et $Y_M \geq 2R_M$. Nous n'avons pas vraiment besoin de cette hypothèse, mais celle-ci simplifie les notations.

On appelle $\xi = (X, Y, R_M)$ un tel objet, et suite à une translation du repère, que l'on centre en (X, Y) , on a $\xi = (0, 0, R_M)$.

Puis, on crée 4 objets supplémentaires, de marque minimale R_m , dont la silhouette est contenue dans celle de ξ . Ces objets sont :

$$\begin{aligned} x_{nord} &= (0, R_M - R_m, R_m) \\ x_{sud} &= (0, R_m - R_M, R_m) \\ x_{est} &= (R_M - R_m, 0, R_m) \\ x_{ouest} &= (R_m - R_M, 0, R_m) \end{aligned}$$

Ces 5 objets sont visibles sur la Fig. (3.8).

Les 4 centres de ces derniers objets sont les sommets d'un carré dans \mathcal{P} , sur lequel on définit un maillage uniforme de dimension $n \times n$. Sur chacun des $(n+1)^2$ sommets de ce maillage, on place un objet $x_{i,j}$, $(i,j) \in [1, n+1]^2$, de marque R_m . On a par exemple $x_{nord} = x_{1,1}$, $x_{sud} = x_{n+1, n+1}$, $x_{est} = x_{1, n+1}$ et $x_{ouest} = x_{n+1, 1}$.

Notons que la distance minimale entre deux objets, qui est aussi la distance caractéristique de ce maillage, est $d_n = \frac{\sqrt{2}(R_M - R_m)}{n}$. Notons d'autre part que pour les n pairs, l'un des points du maillage est le centre du repère, c'est à dire $x_{\frac{n}{2}, \frac{n}{2}} = (0, 0, R_m)$. Nous travaillerons donc par la suite avec des n impairs afin d'éviter que l'un des objets du maillage ait la même position que ξ , puisque notre processus doit être simple.

On appelle \mathbf{x}_n la configuration composée de ces $(n+1)^2$ objets. On se propose alors de calculer $\Delta\mathcal{A}_1(\mathbf{x}_n, \xi)$.

Le fait de rajouter l'objet ξ à la configuration \mathbf{x}_n crée $(n+1)^2$ nouvelles cliques puisque $\xi \sim_I x_{i,j} \forall (i,j) \in [1, n+1]^2$. D'après (3.4) :

$$\begin{aligned} \Delta\mathcal{A}_1(\mathbf{x}_n, \xi) &= \sum_{u \in \mathbf{x}_n} \lambda_2(\mathcal{S}_{\mathcal{P}}(\xi) \cap \mathcal{S}_{\mathcal{P}}(u)) \\ &= (n+1)^2 \lambda_2(\mathcal{S}_{\mathcal{P}}(x_{i,j})) \\ &= (n+1)^2 \pi R_m^2. \end{aligned}$$

Ce terme n'est donc pas majoré :

$$\lim_{n \rightarrow +\infty} \Delta\mathcal{A}_1(\mathbf{x}_n, \xi) = +\infty \tag{3.16}$$

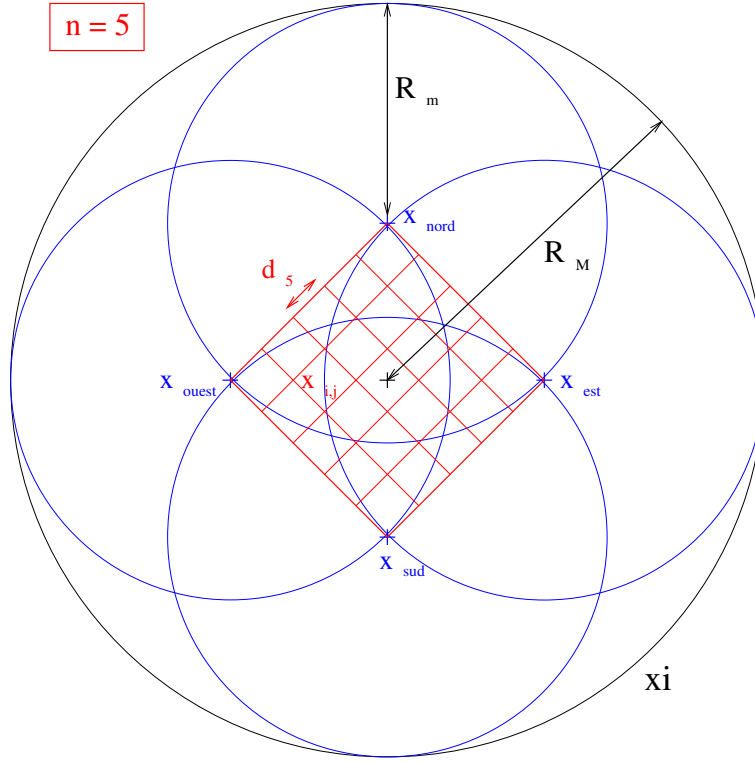


FIG. 3.8 – Description du contre-exemple, avec $R_m = 7$, $R_M = 12$, et $n = 5$.

Il est donc impossible de minorer l'intensité conditionnelle de Papangelou avec le premier modèle d'interaction d'aires, puisque

$$\forall M > 0, \exists (\mathbf{x}_n, \xi) \in \Psi \times U, n = \sqrt{\frac{M}{\pi R_m^2}}, \text{ tels que } \Delta \mathcal{A}_1(\mathbf{x}_n, \xi) > M.$$

2. Modèle 2 : $\Delta \mathcal{A}_2(\mathbf{x}, \xi)$

De la même façon, à partir d'un contre-exemple nous allons montrer qu'il est impossible de minorer l'intensité conditionnelle de Papangelou de $h_p(\mathbf{x})$ avec ce modèle d'interaction, en montrant que l'on ne peut pas majorer $\Delta \mathcal{A}_2(\mathbf{x}, \xi)$. Le contre-exemple est en fait le même, et nous garderons les notations du paragraphe précédent.

D'après (3.5) :

$$\begin{aligned} \Delta \mathcal{A}_2(\mathbf{x}_n, \xi) &= \max(\lambda_2(\mathcal{S}_P(\xi) \cap \mathcal{S}_P(v)), v \in \mathcal{V}_{\sim_I, \mathbf{x}_n \cup \{\xi\}}(\xi)) \\ &\quad + \sum_{u \in \mathbf{x}_n} \max(\lambda_2(\mathcal{S}_P(u) \cap \mathcal{S}_P(v)), v \in \mathcal{V}_{\sim_I, \mathbf{x}_n \cup \{\xi\}}(u)) \\ &\quad - \sum_{u \in \mathbf{x}_n} \max(\lambda_2(\mathcal{S}_P(u) \cap \mathcal{S}_P(v)), v \in \mathcal{V}_{\sim_I, \mathbf{x}_n}(u)) \end{aligned}$$

$= \lambda_2(\mathcal{S}_P(u) \cap \mathcal{S}_P(\xi)) = \lambda_2(\mathcal{S}_P(u))$

Dans la configuration \mathbf{x}_n , tout objet possède au moins 1 plus proche voisin sur le maillage, avec lequel l'aire d'intersection des silhouettes est maximale. En notant A l'aire d'intersection de ces deux objets de la configuration, distants de d_n , on obtient

$$\Delta \mathcal{A}_2(\mathbf{x}_n, \xi) = \pi R_m^2 + (n+1)^2 \pi R_m^2 - (n+1)^2 A$$

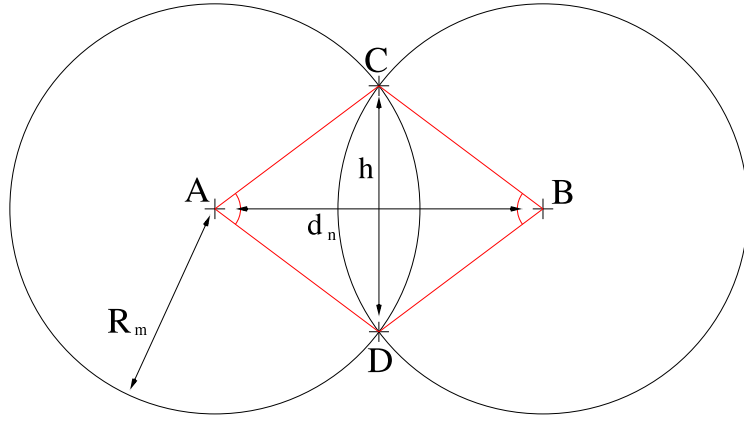


FIG. 3.9 – Calcul de l'intersection de deux cercles de même rayon.

Avec les notations de la Fig. (3.9), on peut exprimer A en fonction des variables du problème :

$$\begin{aligned}
 A &= R_m^2 \left(\frac{\widehat{CAD} + \widehat{CBD}}{2} \right) - \frac{d_n h}{2} \\
 &= R_m^2 \widehat{CBD} - 2 \frac{d_n}{2} \frac{h}{2} \\
 &= R_m^2 \widehat{CBD} - 2 \cos \left(\frac{\widehat{CBD}}{2} \right) R_m \sin \left(\frac{\widehat{CBD}}{2} \right) R_m \\
 &= R_m^2 \left(\widehat{CBD} - \sin(\widehat{CBD}) \right) \\
 &= R_m^2 \left(2 \arccos \left(\frac{d_n}{2R_m} \right) - \sin(\widehat{CBD}) \right) \\
 &= R_m^2 \left(2 \arccos \left(\frac{K}{n} \right) - \sin(\widehat{CBD}) \right)
 \end{aligned}$$

avec $K = \frac{R_M - R_m}{\sqrt{2}R_m}$. Puis, comme $\sin(\widehat{CBD}) > 0$ et $\pi R_m^2 > 0$, on peut écrire

$$\Delta \mathcal{A}_2(\mathbf{x}_n, \xi) > (n+1)^2 R_m^2 \left(\pi - 2 \arccos \left(\frac{K}{n} \right) \right). \quad (3.17)$$

Montrons que le terme de droite tend vers l'infini quand $n \rightarrow +\infty$. Pour cela, on écrit le développement limité de la fonction $\arccos \left(\frac{K}{n} \right)$ à l'ordre 2 en $\frac{1}{n}$:

$$\arccos\left(\frac{K}{n}\right) = \frac{\pi}{2} - \frac{K}{n} + o\left(\frac{1}{n^2}\right). \quad (3.18)$$

D'après (3.18), on obtient

$$\begin{aligned} (n+1)^2 R_m^2 \left(\pi - 2 \arccos\left(\frac{K}{n}\right) \right) &= (n+1)^2 R_m^2 \left(\frac{2K}{n} + o\left(\frac{1}{n^2}\right) \right) \\ &= (2K R_m^2)(n+2) + o(1) \end{aligned}$$

Ce terme n'est donc pas majoré quand $n \rightarrow +\infty$. (3.17) nous permet de conclure :

$$\lim_{n \rightarrow +\infty} \Delta \mathcal{A}_2(\mathbf{x}_n, \xi) = +\infty \quad (3.19)$$

Il est donc impossible de minorer l'intensité conditionnelle de Papangelou avec le second modèle d'interaction d'aire.

3. Modèle 3 : $\Delta \mathcal{A}_3(\mathbf{x}, \xi)$

Cette fois, nous allons démontrer que l'intensité conditionnelle de Papangelou est minorée. Comme pour les deux modèles précédents, nous allons nous intéresser au terme $\Delta \mathcal{A}_3(\mathbf{x}, \xi)$ et montrer qu'il est majoré.

D'après (3.6) :

$$\Delta \mathcal{A}_3(\mathbf{x}_n, \xi) = \underbrace{\lambda_2(\cap_{\mathbf{x}_n \cup \{\xi\}}(\xi))}_{g_1(\mathbf{x}_n, \xi)} + \underbrace{\sum_{u \in \mathbf{x}} [\lambda_2(\cap_{\mathbf{x}_n \cup \{\xi\}}(u)) - \lambda_2(\cap_{\mathbf{x}_n}(u))]}_{g_2(\mathbf{x}_n, \xi)}$$

avec

$$g_1(\mathbf{x}_n, \xi) \leq \lambda_2(\mathcal{S}_{\mathcal{P}}(\xi)) \leq \pi R_M^2 \quad (3.20)$$

puisque $\cap_{\mathbf{x}_n \cup \{\xi\}}(\xi) \subset \mathcal{S}_{\mathcal{P}}(\xi)$, et

$$g_2(\mathbf{x}_n, \xi) = \sum_{u \in \mathbf{x}_n} \lambda_2 \left(\underbrace{\cap_{\mathbf{x}_n \cup \{\xi\}}(u) \setminus \cap_{\mathbf{x}_n}(u)}_{A_u} \right)$$

puisque $\cap_{\mathbf{x}_n}(u) \subset \cap_{\mathbf{x}_n \cup \{\xi\}}(u)$.

Lemme 3.7 :

$$\left\| \begin{array}{l} \text{Les ensembles } A_u \text{ vérifient les deux propriétés suivantes :} \\ - \forall u \in \mathbf{x}_n, A_u \subset \mathcal{S}_{\mathcal{P}}(\xi) \\ - \text{les ensembles } A_u \text{ sont disjoints : } A_u \cap A_v = \emptyset, u \neq v \end{array} \right.$$

◇ **Démonstration :**

$\forall u \in \mathbf{x}_n$ tel que $A_u \neq \{\emptyset\}$, $\forall x \in A_u$, d'après la définition de A_u :

$$x \in A_u \subset \cap_{\mathbf{x}_n \cup \{\xi\}}(u) = \mathcal{S}_{\mathcal{P}}(u) \cap \mathcal{S}_{\mathcal{P}}(\mathcal{V}_{\sim I, \mathbf{x}_n \cup \{\xi\}}(u))$$

Donc $x \in \mathcal{S}_{\mathcal{P}}(u)$, et $\exists v \in \mathbf{x}_n \cup \{\xi\}$, $v \neq u$, tel que $x \in \mathcal{S}_{\mathcal{P}}(v)$. Montrons que cet objet v ne peut être que ξ , en raisonnant par l'absurde.

Si $v \neq \xi$, alors $x \in \mathcal{S}_{\mathcal{P}}(u) \cap \mathcal{S}_{\mathcal{P}}(v) \subset \cap_{\mathbf{x}_n}(u) \implies x \in \cap_{\mathbf{x}_n}(u)$. Par définition des ensembles A_u , on a aussi $x \notin \cap_{\mathbf{x}_n}(u)$, ce qui est contradictoire. Ainsi, le seul objet v possible est ξ lui-même, et on a bien $x \in \mathcal{S}_{\mathcal{P}}(\xi)$, ce qui démontre la première propriété.

Nous venons de montrer que $\forall v \in \mathbf{x}_n, v \neq u, x \notin \mathcal{S}_{\mathcal{P}}(v)$. Or, $\mathcal{S}_{\mathcal{P}}(v) \supset A_v$, et donc $x \notin A_v$. Ce qui démontre la seconde propriété.

◇

D'après le lemme précédent, les ensembles A_u étant disjoints deux à deux et contenus dans $\mathcal{S}_{\mathcal{P}}(\xi)$,

$$\begin{aligned} g_2(\mathbf{x}_n, \xi) &= \sum_{u \in \mathbf{x}_n} \lambda_2(A_u) \\ &= \lambda_2\left(\bigcup_{u \in \mathbf{x}_n} A_u\right) \\ &\leq \lambda_2(\mathcal{S}_{\mathcal{P}}(\xi)). \end{aligned}$$

Un majorant de $g_2(\mathbf{x}_n, \xi)$ est dans ce cas :

$$g_2(\mathbf{x}_n, \xi) \leq \pi R_M^2. \quad (3.21)$$

(3.20) et (3.21) nous permettent de conclure sur la majoration de $\Delta \mathcal{A}_3(\mathbf{x}_n, \xi)$:

$$\Delta \mathcal{A}_3(\mathbf{x}_n, \xi) \leq 2\pi R_M^2. \quad (3.22)$$

En conclusion, le modèle 3 d'interaction d'aires a bien son intensité conditionnelle de Papan-gelou bornée des deux côtés.

3.2.5 Estimateurs

On cherche une estimée de la configuration d'objets qui correspond le mieux à notre image, c'est à dire la configuration qui minimise l'espérance d'une fonction de coût suivant la loi a posteriori. Il faut donc déjà se donner une fonction de coût sur l'espace des configurations.

► Estimateur de Bayes

Soit \mathcal{I} une image, toute la connaissance que l'on détient sur la configuration d'objets optimale sur cette image est contenue dans la densité a posteriori $h(\mathbf{x}) = h(\mathbf{x}|\mathcal{I})$.

Soit $\mathcal{L}(\mathbf{x}, \hat{\mathbf{x}})$ une fonction de coût, qui mesure la perte due à l'estimation de la configuration optimale par $\hat{\mathbf{x}}$. On pose classiquement $\mathcal{L}(\mathbf{y}, \mathbf{y}) = 0$, et $\mathcal{L}(\mathbf{y}, \mathbf{z}) > 0$ si \mathbf{y} est différent de \mathbf{z} .

Une fois cette fonction définie, on peut obtenir l'**estimateur de Bayes** \mathbf{x}^* en minimisant l'espérance a posteriori de ce coût appelé **risque de Bayes** :

$$\mathbf{x}^* = \arg \min_{\hat{\mathbf{x}}} (\mathbb{E}_{\mathbf{x}|\mathcal{I}}[\mathcal{L}(\mathbf{x}, \hat{\mathbf{x}})]) \quad (3.23)$$

Un premier estimateur est le **maximum a posteriori**. Cependant, nous verrons que ce n'est pas le seul, et que d'autres sont plus adaptés pour les problèmes de traitement d'images.

► Maximum a posteriori

▷ Calcul de l'estimateur

La fonction de coût qui lui est associée est :

$$\mathcal{L}_{MAP}(\mathbf{x}, \hat{\mathbf{x}}) = \begin{cases} 0 & \text{si } \mathbf{x} = \hat{\mathbf{x}}; \\ 1 & \text{sinon.} \end{cases} \quad (3.24)$$

L'estimateur de Bayes associé, \mathbf{x}_{MAP} , solution du problème de minimisation (3.23), est donc la configuration qui maximise la densité a posteriori :

$$\begin{aligned} \mathbf{x}_{MAP} &= \arg \max_{\hat{\mathbf{x}}} (h(\mathbf{x}|\mathcal{I})) \\ &= \arg \max_{\hat{\mathbf{x}}} (h(\mathbf{x})). \end{aligned}$$

▷ Recuit simulé

Dans la pratique, pour obtenir la configuration qui maximise la densité a posteriori $h(\cdot)$, on utilise un **recuit simulé**. Son principe est de simuler une chaîne de Markov de densité objectif non plus $h(\cdot)$, mais $h(\cdot)^{\frac{1}{T_k}}$, où T_k décroît vers 0 quand le nombre d'itérations augmente.

Afin de contrôler la dynamique de l'algorithme, il est courant de sortir le terme $\beta^{n(\mathbf{x})}$ de la densité $h(\mathbf{x})$, et de le rapprocher de la mesure du processus de Poisson de référence $\mu(\cdot)$ qu'il pondère. L'expression de la loi objectif $\pi(\cdot)$ à une température T_k est alors :

$$\pi_{T_k}(d\mathbf{x}) = e^{\frac{\gamma_a \mathcal{A}_3(\mathbf{x})}{T_k}} \mathcal{L}(\mathcal{I}|\mathbf{x})^{T_k} \beta^{n(\mathbf{x})} \mu(d\mathbf{x}). \quad (3.25)$$

Ainsi, cela revient à faire évoluer $h(\cdot)$ en dirac au niveau de ses modes principaux. La décroissance vers 0 de la température nous assure donc de trouver le maximum de la densité a posteriori.

Cependant, de fortes conditions sur la vitesse de décroissance des températures T_k sont nécessaires pour s'assurer de la convergence de l'algorithme de recuit simulé. Les travaux de D. et S. Geman, ainsi que de B. Hajek, montrent qu'une décroissance logarithmique

$$T_k = \frac{a}{\ln(k+1)}$$

convient pour assurer la convergence d'un algorithme de type Metropolis (ce qui est notre cas) par recuit simulé.

Une telle décroissance est trop lente pour être implantée. Lors des simulations sur ordinateur, nous lui préférons une décroissance géométrique, ce qui nous fait sortir du cadre théorique de la convergence, mais donne en pratique des résultats corrects.

$$T_{k+1} = \alpha_{temp} T_k, 0 < a < 1, 1 - a \ll 1.$$

► Autres estimateurs

La difficulté de fournir des estimateurs relève du fait que l'on doit être capable de trouver la solution du problème de minimisation (3.23).

H. Rue, dans [28], [29], [30], et [32], nous fournit des méthodes de modélisation de fonctions de coûts, pour lesquelles on peut résoudre (3.23). En particulier, la **distance de Baddeley**, introduite par A.J. Baddeley, paraît tout à fait appropriée à notre étude, puisqu'elle mesure une distance entre deux images.

H. Rue s'est donc chargé de modéliser une fonction de coût s'inspirant de la distance de Baddeley. Ce nouvel estimateur pourrait se révéler plus adapté que le premier, lorsque nous effectuerons une série de tests en le substituant au maximum a posteriori utilisé jusqu'alors.

3.3. Noyau de propositions adopté

En théorie, l'algorithme de Metropolis-Hastings-Green, pourvu des mouvements de naissance et de mort seulement, converge vers la loi objectif. Les propriétés mathématiques sont en effet toutes vérifiées. Dans la pratique, le noyau de propositions est plus complexe pour nous permettre d'accélérer cette convergence, ralentie par l'utilisation d'un recuit simulé et de la lente décroissance des températures.

Notre objectif est en effet de modéliser des noyaux de propositions adaptés à la loi objectif, afin d'éviter au maximum des mouvements inutiles et de minimiser le taux de rejet. Ces mouvements seront groupés en deux catégories dans cette partie : les mouvements dits **simples**, c'est à dire ceux modifiant un des objets de la configuration courante, et les mouvement **complexes**, qui changent le nombre d'objets de la configuration (comme la naissance et la mort).

Soient \mathbf{x} une configuration, et $A \in \mathcal{B}(\Psi)$. Le noyau de propositions $Q(\mathbf{x}, A)$, défini sur $\Psi \times \mathcal{B}(\Psi)$, peut se décomposer sous la forme d'une somme de noyaux de propositions, choisis avec les probabilités p_i :

$$Q(\mathbf{x}, A) = \sum_{i \in I} p_i Q_i(\mathbf{x}, A)$$

où chacun des noyaux $Q_i(\mathbf{x}, \cdot)$ décrit un mouvement.

3.3.1 Mouvements simples

Principalement deux mouvements simples nécessitent d'être inclus dans notre noyau de propositions : la translation des objets (qui affecte leurs composants dans l'espace des positions \mathcal{P}), ainsi que la dilatation (qui affecte leur composant dans l'espace des marques \mathcal{K}).

Ces mouvements simples sont la combinaison d'une mort et d'une naissance, il font donc gagner du temps à l'algorithme, qui n'a plus besoin d'accepter ces deux mouvements, mais juste un seul.

► Translation

▷ Principe

Soit $x_i = (X_{x_i}, Y_{x_i}, R_{x_i})$ un des objets d'une configuration \mathbf{x} . Le mouvement de translation propose de modifier ses coordonnées (X_{x_i}, Y_{x_i}) dans l'espace des positions \mathcal{P} .

Pour cela, nous introduisons une variable aléatoire θ sur un compact symétrique de \mathbb{R}^2 , Θ : ainsi, si $\theta \in \Theta$, alors $-\theta \in \Theta$. Cette variable aléatoire suit une loi λ_2 -uniforme sur Θ , $\lambda_2(\cdot)$ étant la mesure de Lebesgue sur \mathbb{R}^2 . Dans notre algorithme, nous avons choisi pour Θ le disque de rayon 1.

En posant $\Theta = [-a_X, a_X] \times [-b_Y, b_Y]$, les étapes de la translation sont donc les suivantes :

1. Choix uniforme d'un objet x_i de la configuration, avec la probabilité $\frac{1}{n(\mathbf{x})}$, que l'on va translater.
2. Génération de la variable aléatoire $\theta = (\theta_X, \theta_Y) \in \Theta$.
3. Calcul de l'objet x_i translaté, noté $\theta(x_i) = (X_{x_i} + \theta_X, Y_{x_i} + \theta_Y, R_{x_i})$. Si celui-ci n'appartient pas à U (car sa position est hors de l'espace des positions \mathcal{P}), le mouvement est aussitôt rejeté.
4. Calcul du rapport de Green R_{tr} .

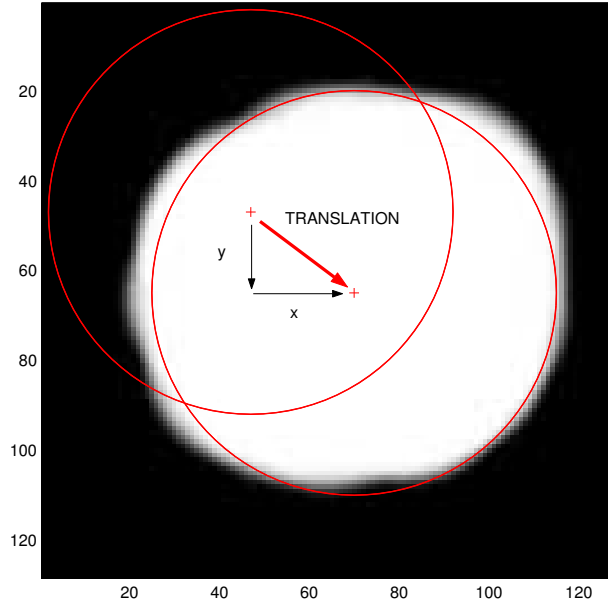


FIG. 3.10 – *Mouvement de translation*

5. Avec la probabilité $\alpha_{tr} = \min(1, R_{tr})$, on accepte la nouvelle configuration $\mathbf{y} = \mathbf{x} \cup \{\theta(x_i)\} \setminus \{x_i\}$. Avec la probabilité $1 - \alpha_{tr}$, on reste dans la configuration courante : $\mathbf{y} = \mathbf{x}$.

Ce mouvement est illustré dans la Fig. (3.10).

▷ Noyau de translation et mesure symétrique

Avec les mêmes notations que précédemment, le noyau de translation d'une configuration \mathbf{x} vers un borélien A sera noté $Q_{tr}(\mathbf{x}, A)$, et choisi avec une probabilité p_{tr} . On a :

$$Q_{tr}(\mathbf{x}, A) = \sum_{x_i \in \mathbf{x}} \frac{1}{n(\mathbf{x})} \int_{\theta \in \Theta} \mathbf{1}_A(\mathbf{x} \cup \{\theta(x_i)\} \setminus \{x_i\}) \frac{\lambda_2(d\theta)}{\lambda_2(\Theta)}. \quad (3.26)$$

La mesure symétrique $\xi_{tr}(\cdot, \cdot)$ définie sur $\Psi \times \Psi$, par rapport à laquelle $\pi(\cdot)Q_t(\cdot)$ sera absolument continue, est :

$$\xi_{tr}(A, B) = \int_{\Psi} \mathbf{1}_A(\mathbf{x}) \sum_{x_i \in \mathbf{x}} \int_{\theta \in \Theta} \mathbf{1}_B(\mathbf{x} \cup \{\theta(x_i)\} \setminus \{x_i\}) \lambda_2(d\theta) \mu(d\mathbf{x}). \quad (3.27)$$

En posant $\mathbf{y} = \mathbf{x} \cup \{\theta(x_i)\} \setminus \{x_i\}$, la dérivée de Radon Nikodym $f_{tr}(\mathbf{x}, \mathbf{y})$, associée à ce mouvement de translation, s'exprime facilement :

$$\begin{aligned}
f_{tr}(\mathbf{x}, \mathbf{y}) &= \frac{\pi(d\mathbf{x})Q_{tr}(\mathbf{x}, d\mathbf{y})}{\xi_{tr}(d\mathbf{x}, d\mathbf{y})} \\
&= \frac{h(\mathbf{x})\mu(d\mathbf{x})Q_{tr}(\mathbf{x}, d\mathbf{y})}{\mu(d\mathbf{x})\lambda_2(d\theta)} \\
&= \frac{h(\mathbf{x})}{n(\mathbf{x})\lambda_2(\Theta)}.
\end{aligned}$$

▷ Rapport de Green

Le rapport de Green associé à cette transformation s'obtient donc :

$$\boxed{R_{tr}(\mathbf{x}, \mathbf{x} \cup \{\theta(x_i)\} \setminus \{x_i\}) = \frac{h(\mathbf{x} \cup \{\theta(x_i)\} \setminus \{x_i\})}{h(\mathbf{x})}}. \quad (3.28)$$

► Dilatation

▷ Principe

Soit $x_i = (X_{x_i}, Y_{x_i}, R_{x_i})$ un des objets d'une configuration \mathbf{x} . Le mouvement de dilatation propose quant à lui de modifier sa coordonnée R_{x_i} dans l'espace des marques \mathcal{K} .

Pour cela, nous introduisons une variable aléatoire δ sur un compact symétrique de \mathbb{R} , Δ : ainsi, si $\delta \in \Delta$, alors $-\delta \in \Delta$. Cette variable aléatoire suit une loi λ -uniforme sur Δ , $\lambda(\cdot)$ étant la mesure de Lebesgue sur \mathbb{R} . Dans notre algorithme, Δ est le segment $[-1, 1]$.

En posant $\Delta = [-c_R, c_R]$, les étapes de la translation sont donc les suivantes :

1. Choix uniforme d'un objet x_i de la configuration, avec la probabilité $\frac{1}{n(\mathbf{x})}$, que l'on va dilater.
2. Génération de la variable aléatoire $\delta \in \Delta$.
3. Calcul de l'objet x_i dilaté, noté $\delta(x_i) = (X_{x_i}, Y_{x_i}, R_{x_i} + \delta)$. Si celui-ci n'appartient pas à U (car son rayon est hors de l'espace des marques $\mathcal{K} = [R_{min}, R_{max}]$), le mouvement est aussitôt rejeté.
4. Calcul du rapport de Green R_{di} .
5. Avec la probabilité $\alpha_{di} = \min(1, R_{di})$, on accepte la nouvelle configuration $\mathbf{y} = \mathbf{x} \cup \{\delta(x_i)\} \setminus \{x_i\}$. Avec la probabilité $1 - \alpha_{di}$, on reste dans la configuration courante : $\mathbf{y} = \mathbf{x}$.

Ce mouvement est illustré dans la Fig. (3.11).

▷ Noyau de dilatation et mesure symétrique

Le noyau de dilatation d'une configuration \mathbf{x} vers un borélien A sera noté $Q_{di}(\mathbf{x}, A)$, et choisi avec une probabilité p_{di} . On a :

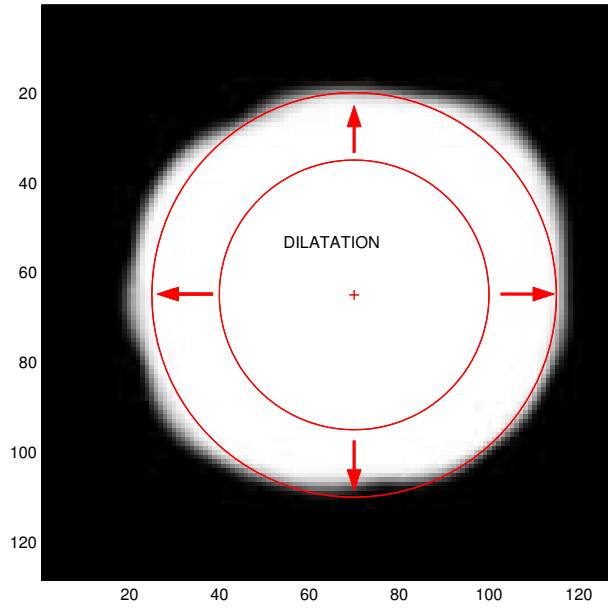


FIG. 3.11 – *Mouvement de dilatation*

$$Q_{d_i}(\mathbf{x}, A) = \sum_{x_i \in \mathbf{x}} \frac{1}{n(\mathbf{x})} \int_{\delta \in \Delta} \mathbf{1}_A(\mathbf{x} \cup \{\delta(x_i)\} \setminus \{x_i\}) \frac{\lambda(d\delta)}{\lambda(\Delta)}. \quad (3.29)$$

La mesure symétrique $\xi_{d_i}(\cdot, \cdot)$ définie sur $\Psi \times \Psi$, par rapport à laquelle $\pi(\cdot)Q_{d_i}(\cdot, \cdot)$ sera absolument continue, est :

$$\xi_{d_i}(A, B) = \int_{\Psi} \mathbf{1}_A(\mathbf{x}) \sum_{x_i \in \mathbf{x}} \int_{\delta \in \Delta} \mathbf{1}_B(\mathbf{x} \cup \{\delta(x_i)\} \setminus \{x_i\}) \lambda(d\delta) \mu(d\mathbf{x}). \quad (3.30)$$

En posant $\mathbf{y} = \mathbf{x} \cup \{\delta(x_i)\} \setminus \{x_i\}$, la dérivée de Radon Nikodym $f_{d_i}(\mathbf{x}, \mathbf{y})$, associée à ce mouvement de dilatation, s'exprime facilement :

$$\begin{aligned} f_{d_i}(\mathbf{x}, \mathbf{y}) &= \frac{\pi(d\mathbf{x})Q_{d_i}(\mathbf{x}, d\mathbf{y})}{\xi_{d_i}(d\mathbf{x}, d\mathbf{y})} \\ &= \frac{h(\mathbf{x})\mu(d\mathbf{x})Q_{d_i}(\mathbf{x}, d\mathbf{y})}{\mu(d\mathbf{x})\lambda(d\delta)} \\ &= \frac{h(\mathbf{x})}{n(\mathbf{x})\lambda(\Delta)}. \end{aligned}$$

▷ Rapport de Green

Le rapport de Green associé à cette transformation s'obtient donc :

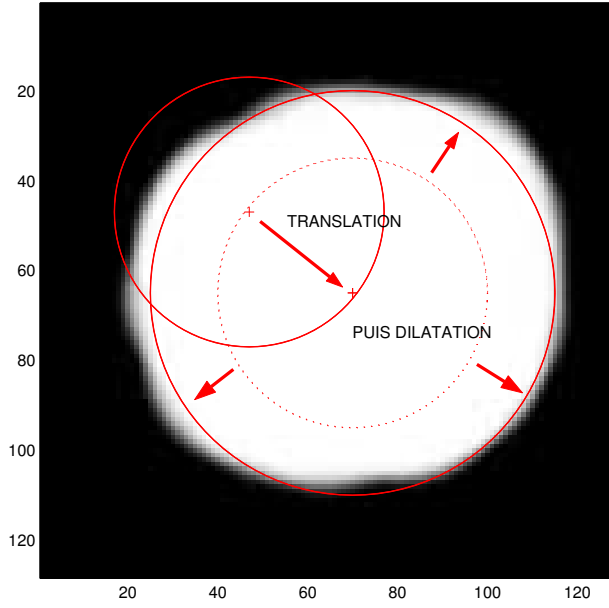


FIG. 3.12 – *Mouvement de translation et de dilatation*

$$\boxed{R_{di}(\mathbf{x}, \mathbf{x} \cup \{\delta(x_i)\} \setminus \{x_i\}) = \frac{h(\mathbf{x} \cup \{\delta(x_i)\} \setminus \{x_i\})}{h(\mathbf{x})}.} \quad (3.31)$$

► Combinaison de mouvements simples

Enfin, le dernier mouvement simple que nous proposons est la combinaison d'une translation et d'une dilatation. Parfois en effet, une simple translation ou une simple dilatation ne sont pas acceptés, alors que la combinaison des deux peut se révéler efficace.

Avec les mêmes notations que précédemment, le mouvement de translation-dilatation consiste en la succession d'une dilatation puis translation. La Fig. (3.12) illustre cette transformation.

Son noyau de propositions, d'une configuration \mathbf{x} vers un borélien A , sera noté $Q_{td}(\mathbf{x}, A)$ et choisi avec une probabilité p_{trdi} . On a :

$$Q_{trdi}(\mathbf{x}, A) = \sum_{x_i \in \mathbf{x}} \frac{1}{n(\mathbf{x})} \int_{\theta \in \Theta} \int_{\delta \in \Delta} \mathbf{1}_A(\mathbf{x} \cup \{\theta(\delta(x_i))\} \setminus \{x_i\}) \frac{\lambda(d\delta)}{\lambda(\Delta)} \frac{\lambda_2(d\theta)}{\lambda_2(\Theta)}. \quad (3.32)$$

Le rapport de Green associé à cette transformation s'obtient de la même façon que les précédents :

$$\boxed{R_{trdi}(\mathbf{x}, \mathbf{x} \cup \{\theta(\delta(x_i))\} \setminus \{x_i\}) = \frac{h(\mathbf{x} \cup \{\theta(\delta(x_i))\} \setminus \{x_i\})}{h(\mathbf{x})}.} \quad (3.33)$$

3.3.2 Mouvements complexes

On appelle mouvement complexe tout mouvement aboutissant à un changement de dimension de la configuration \mathbf{x} .

Les mouvements de naissance et de mort uniformes ont déjà été présentés. Les démonstrations de convergence de l'algorithme de Green vers la densité objectif avec un tel noyau ont aussi été apportées précédemment.

Cependant, ce ne sont pas les seuls mouvements complexes qui méritent d'être conçus. Ainsi nous verrons dans notre prochain chapitre qu'un mouvement proposant la fusion de deux objets, ou bien au contraire la scission d'un cercle en deux, améliore nos résultats.

Nous allons par conséquent présenter dès maintenant le noyau de propositions de fusion/division.

3.3.3 Noyau de propositions de fusion/division

On rappelle que la relation d'intersection \sim_I a été définie comme suit :

$$x_i \sim_I x_j \Leftrightarrow d(C_{x_i}, C_{x_j}) < R_{x_i} + R_{x_j},$$

où $d(.,.)$ est la distance euclidienne sur \mathbb{R}^2 . Le nombre de paires d'objets (ou cliques) d'une configuration \mathbf{x} , voisins au sens de \sim_I , avait été noté $n_{\sim_I}(\mathbf{x})$:

$$n_{\sim_I}(\mathbf{x}) = \sum_{x_i, x_j \in \mathbf{x}, x_i \neq x_j} \mathbf{1}_{x_i \sim_I x_j}.$$

A partir de la configuration courante \mathbf{x}_t , une fusion consiste à supprimer deux objets x_i et x_j de \mathbf{x}_t , voisins au sens de la relation \sim_I , et de rajouter à la configuration un nouvel objet $u \in U$ obtenu à partir des deux objets supprimés.

Une division consiste à supprimer un objet x_i de la configuration, et de rajouter deux objets u et v , voisins au sens de \sim_I , obtenus à partir de x_i .

Les deux mouvements de fusion et de division doivent être construits de façon à être symétriques afin de préserver la réversibilité de la chaîne de Markov.

► Le principe

▷ Fusion

Les étapes de la fusion de deux objets x_i et x_j d'une configuration sont les suivantes (cf Fig. (3.13)) :

1. Choix d'une clique d'objets (x_i, x_j) dans la configuration, avec une probabilité uniforme sur l'ensemble des cliques $\frac{1}{n_{\sim_I}(\mathbf{x})}$. Ces deux objets sont décrits par leurs coordonnées dans l'espace objet U :

$$x_i = (X_{x_i}, Y_{x_i}, R_{x_i}) \text{ et } x_j = (X_{x_j}, Y_{x_j}, R_{x_j}).$$

2. On calcule l'objet $u = (X_u, Y_u, R_u)$, résultat de la fusion de x_i et x_j .

Le centre de cet objet $(X_u, Y_u) \in \mathcal{P}$ se situe dans un disque de centre le milieu des deux objets à fusionner, $\left(\frac{X_{x_i} + X_{x_j}}{2}, \frac{Y_{x_i} + Y_{x_j}}{2}\right)$, et de rayon 1. Ce centre est caractérisé par ses coordonnées

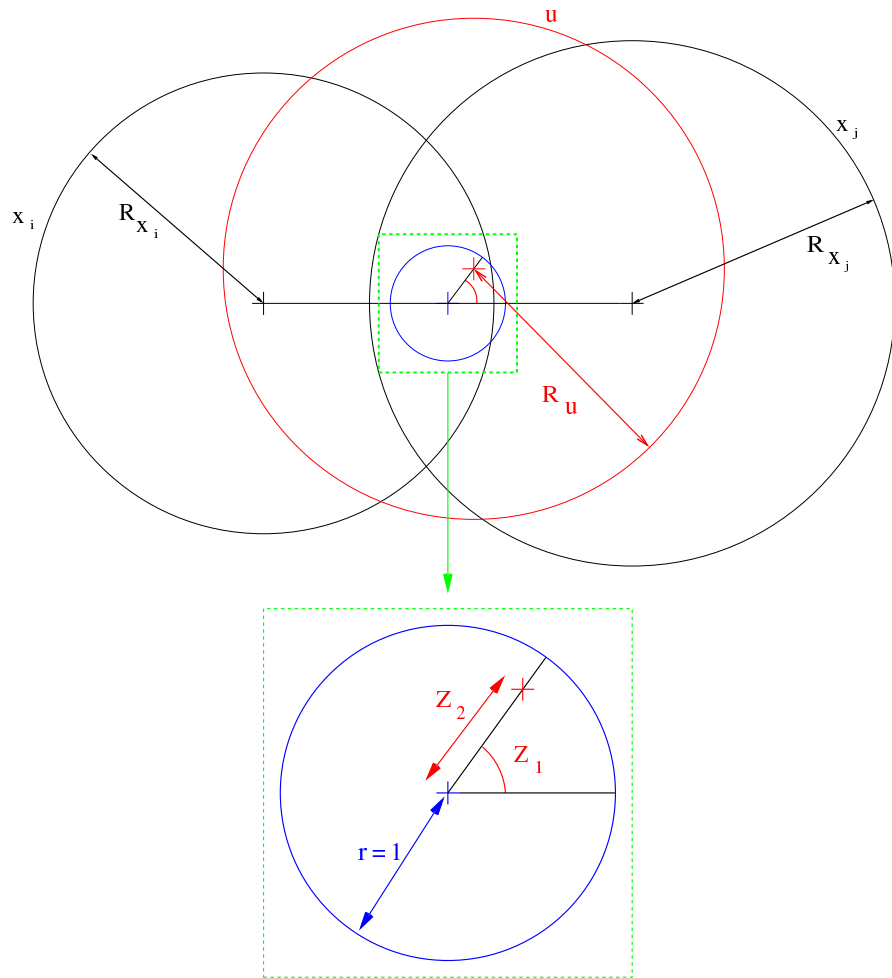


FIG. 3.13 – Schéma explicatif du mouvement de fusion.

polaires dans le disque. Pour les obtenir, on génère une variable aléatoire z_1 uniformément sur $[0, 2\pi[$, ainsi qu'une variable aléatoire $z_2 \in]0, 1]$ (0 exclu pour assurer la différentiabilité de la transformation inverse) selon une loi normale tronquée de moyenne 0 et de variance $\frac{1}{9}$:

$$f_{z_2}(x) = \frac{\frac{3}{\sqrt{2\pi}} \exp\left(\frac{-9x^2}{2}\right)}{\int_0^1 \frac{3}{\sqrt{2\pi}} \exp\left(\frac{-9t^2}{2}\right) dt}$$

Nous choisissons une loi normale pour z_2 , et non pas une simple loi uniforme sur tout le disque, car nous souhaitons donner plus de poids aux centres situés près du milieu des deux centres des objets x_i et x_j . Les expériences menées avec ces deux types d'approches donnent raison à celle choisie.

Si, suite au choix des variables z_1 et z_2 , $(X_u, Y_u) \notin \mathcal{P}$, le mouvement de fusion est refusé. Ce cas de figure peut se présenter si les objets à fusionner sont près de la frontière de l'espace des positions.

Puis, on génère une troisième variable aléatoire z_3 qui nous permet de calculer le rayon du nouvel objet R_u , en l'ajoutant à la moyenne des deux précédents $R_{moy} = \frac{R_{x_i} + R_{x_j}}{2}$. Comme nous souhaitons pouvoir augmenter ou diminuer cette valeur, nous permettons au nouveau cercle d'avoir un rayon situé dans la fourchette suivante : $[R_{moy} - 1, R_{moy} + 1]$. Ceci, compte tenu des restrictions liées aux bornes de l'espace des marques, fait que l'on choisit z_3 uniformément dans $[\max(-1, R_m - R_{moy}), \min(1, R_M - R_{moy})] = [m_{z_3}, M_{z_3}]$.

Remarquons dès maintenant que $M_{z_3} - m_{z_3} \geq \min(1, R_M - R_m)$, ceci nous sera utile par la suite.

Les opérations décrites plus haut peuvent se résumer à l'aide des équations suivantes, avec la variable aléatoire $z = (z_1, z_2, z_3) \in Z_f = [0, 2\pi[\times]0, 1] \times [m_{z_3}, M_{z_3}]$:

$$\begin{pmatrix} X_u \\ Y_u \\ R_u \end{pmatrix} = \begin{pmatrix} \frac{X_{x_i} + X_{x_j}}{2} + z_2 \cos(z_1) \\ \frac{Y_{x_i} + Y_{x_j}}{2} + z_2 \sin(z_1) \\ \frac{R_{x_i} + R_{x_j}}{2} + z_3 \end{pmatrix}$$

La loi de probabilité de la variable aléatoire z est donc :

$$d\mathbf{P}_f(z) = f_f(z) \lambda(dz) = \frac{1}{2\pi} f_{z_2}(z_2) \frac{1}{M_{z_3} - m_{z_3}} \lambda(dz)$$

La fonction η_{x_i, x_j} , qui à un élément de Z_f , associe l'objet résultat de la fusion, est un difféomorphisme :

$$\eta_{x_i, x_j} : \begin{pmatrix} Z_f & \rightarrow & U \\ z & \rightarrow & \eta_{x_i, x_j}(z) \end{pmatrix} \quad (3.34)$$

Le calcul de son Jacobien s'effectue ainsi :

$$|J_{\eta_{x_i, x_j}}(z)| = \left| \det \begin{pmatrix} -z_2 \sin(z_1) & z_2 \cos(z_1) & 0 \\ \cos(z_1) & \sin(z_1) & 0 \\ 0 & 0 & 1 \end{pmatrix} \right| = z_2 > 0$$

Ce calcul nous sera utile par la suite, lorsque nous souhaiterons exprimer le rapport de Green associé à cette transformation.

3. Calcul du rapport de Green R_f .
4. Avec la probabilité $\alpha_f = \min(1, R_f)$, on accepte la nouvelle configuration $\mathbf{y} = \mathbf{x} \setminus \{x_i \cup x_j\} \cup \{u\}$. Avec la probabilité $1 - \alpha_f$, on reste dans la configuration courante: $\mathbf{y} = \mathbf{x}$.

▷ **Division**

Les étapes de la division d'un objet x_i d'une configuration sont les suivantes (cf Fig. (3.14)) :

1. Choix d'un objet x_i dans la configuration, avec une probabilité uniforme sur la configuration $\frac{1}{n(\mathbf{x})}$. Cet objet est décrit par ses coordonnées sur U :

$$x_i = (X_{x_i}, Y_{x_i}, R_{x_i})$$

2. On calcule les objets $u_1 = (X_{u_1}, Y_{u_1}, R_{u_1})$ et $u_2 = (X_{u_2}, Y_{u_2}, R_{u_2})$, résultats de la division de x_i .

Pour que ce mouvement soit symétrique avec le mouvement de fusion, le milieu des centres de ces objets $\left(\frac{X_{u_1} + X_{u_2}}{2}, \frac{Y_{u_1} + Y_{u_2}}{2}\right) \in \mathcal{P}$ doit se situer dans un disque de centre (X_{x_i}, Y_{x_i}) , et de rayon 1. Pour l'obtenir, on raisonne toujours en coordonnées polaires, et on génère une variable aléatoire z_1 uniformément sur $[0, 2\pi[$, ainsi qu'une variable aléatoire $z_2 \in]0, 1]$ selon la loi f_{z_2} explicitée plus haut.

Si, suite au choix des variables z_1 et z_2 , le milieu des centres des objets u_1 et u_2 n'appartient pas à l'espace des positions \mathcal{P} , le mouvement de division est refusé.

Puis, on génère uniformément sur $[\max(-1, R_m - R_{x_i}), \min(1, R_M - R_{x_i})] = [m_{z_3}, M_{z_3}]$ une variable aléatoire z_3 , qui nous permet de calculer la moyenne des rayons des objets créés $\frac{R_{u_1} + R_{u_2}}{2}$, en fonction du rayon R_{x_i} .

A ce stade des opérations, nous connaissons le milieu des centres et la moyenne des rayons des objets u_1 et u_2 :

$$\begin{pmatrix} \frac{X_{u_1} + X_{u_2}}{2} \\ \frac{Y_{u_1} + Y_{u_2}}{2} \\ \frac{R_{u_1} + R_{u_2}}{2} \end{pmatrix} = \begin{pmatrix} X_{x_i} + z_2 \cos(z_1) \\ Y_{x_i} + z_2 \sin(z_1) \\ R_{x_i} + z_3 \end{pmatrix}$$

Pour un même milieu des centres et une même moyenne des rayons, il existe une infinité de divisions possibles. Il nous faut introduire trois nouvelles variables aléatoires pour que cette transformation soit bijective.

Remarquons qu'en terme de positionnement dans l'espace \mathcal{P} , il nous suffit de placer l'un des deux objets puisque l'autre sera son symétrique par rapport au milieu des centres explicité plus haut. Il en est de même pour les rayons, puisque la connaissance d'un des deux rayons nous suffit. C'est pourquoi nous nous attacherons par la suite à calculer les coordonnées de u_1 , à partir desquelles nous déduirons celles de u_2 .

Plaçons nous alors dans un repère d'origine le milieu des centres, et raisonnons une nouvelle fois en terme de coordonnées polaires. Une variable aléatoire z_4 tirée uniformément sur $[0, \pi[$ nous donne l'orientation de u_1 . Restent à fixer la distance entre les 2 objets, ainsi que l'écart entre leurs rayons.

Pour cela, remarquons que dans le but que $u_1 \sim_I u_2$, il faut que

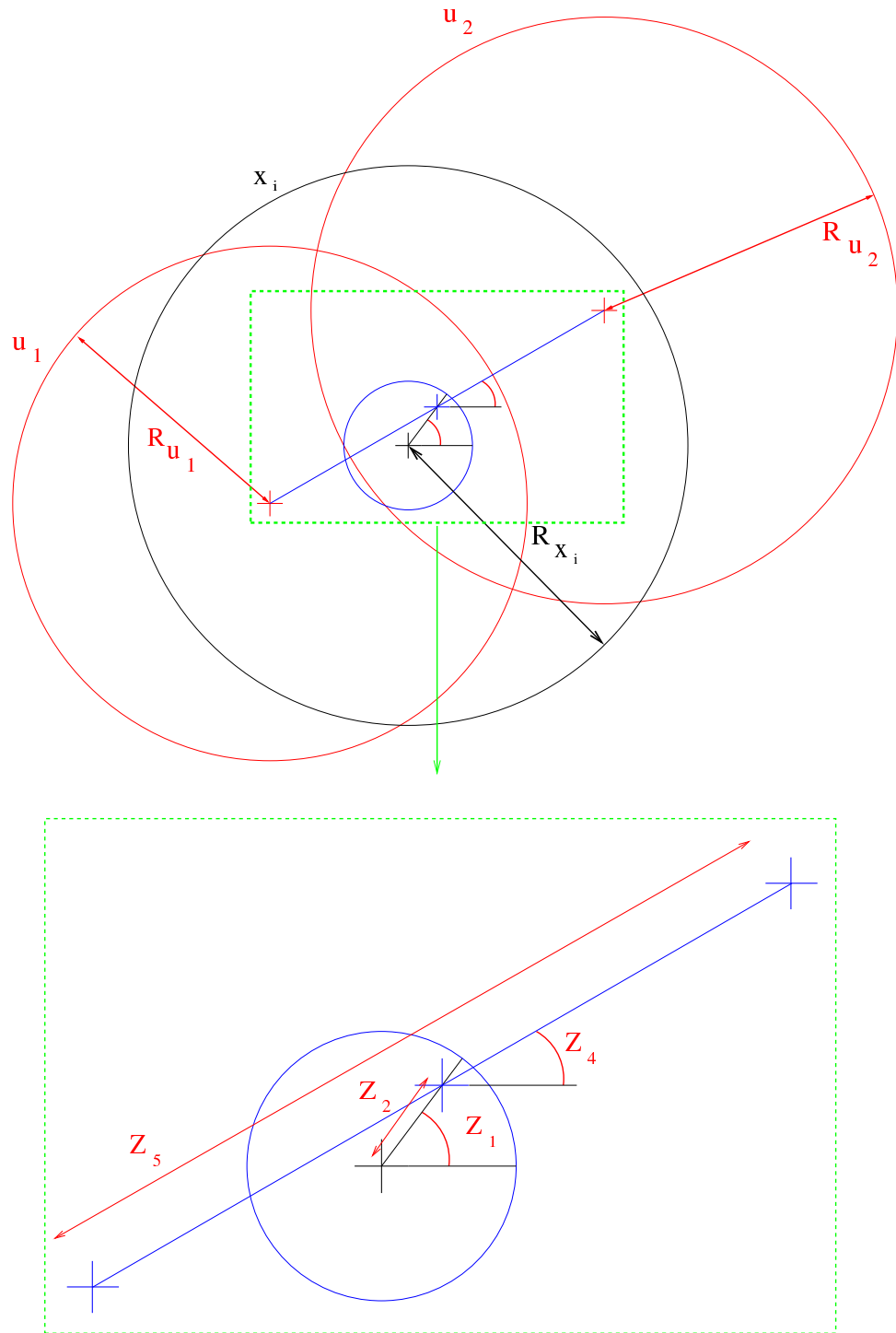


FIG. 3.14 – Schéma explicatif du mouvement de division.

$$d(u_1, u_2) < R_{u_1} + R_{u_2} = 2(R_{x_i} + z_3).$$

Ainsi, on tire une variable aléatoire z_5 uniformément sur $]0, 2(R_{x_i} + z_3)[$, afin de fixer la distance en question : $d(u_1, u_2) = z_5$. Celle-ci doit être non nulle de manière à respecter notre choix de simuler un processus ponctuel marqué simple, processus ponctuel simple sur l'espace des positions, ne pouvant comporter deux objets avec la même position.

Une dernière variable z_6 nous donne la différence entre les rayons $R_{u_1} - R_{u_2}$. Celle-ci est choisie uniformément sur $[-M_{z_6}, M_{z_6}]$, où $M_{z_6} = \min(R_M - (R_{x_i} + z_3), (R_{x_i} + z_3) - R_m)$, afin que l'on ait $R_m \leq \frac{R_{u_1}}{R_{u_2}} \leq R_M$.

Les opérations décrites plus haut peuvent se résumer à l'aide des équations suivantes, avec la variable aléatoire $z = (z_1, z_2, z_3, z_4, z_5, z_6) \in Z_d = [0, 2\pi[\times]0, 1] \times [-1, 1] \times [0, \pi[\times [0, 2(R_{x_i} + z_3)] \times [-M_{z_6}, M_{z_6}]$:

$$\begin{pmatrix} X_{u_1} & = & X_{x_i} + z_2 \cos(z_1) + \frac{z_5 \cos(z_4)}{2} \\ X_{u_2} & = & X_{x_i} + z_2 \cos(z_1) - \frac{z_5 \cos(z_4)}{2} \\ Y_{u_1} & = & Y_{x_i} + z_2 \sin(z_1) + \frac{z_5 \sin(z_4)}{2} \\ Y_{u_2} & = & Y_{x_i} + z_2 \sin(z_1) - \frac{z_5 \sin(z_4)}{2} \\ R_{u_1} & = & R_{x_i} + z_3 + \frac{z_6}{2} \\ R_{u_2} & = & R_{x_i} + z_3 - \frac{z_6}{2} \end{pmatrix}$$

La loi de probabilité de la variable aléatoire z est donc :

$$d\mathbf{P}_d(z) = f_d(z)\lambda(dz) = \left(\frac{1}{2\pi}\right)^2 f_{z_2}(z_2) \frac{1}{M_{z_3} - m_{z_3}} \frac{1}{2(R_{x_i} + z_3)} \frac{1}{M_{z_6}} \lambda(dz)$$

La fonction ζ_{x_i} , qui à un élément de Z_d , associe les objets résultats de la division, est un difféomorphisme :

$$\zeta_{x_i} : \begin{pmatrix} Z_d & \rightarrow & U \times U \\ z & \rightarrow & \zeta_{x_i}(z) \end{pmatrix} \quad (3.35)$$

Le calcul de son Jacobien s'effectue ainsi :

$$|J_{\zeta_{x_i}}(z)| = \left| \det \begin{pmatrix} -z_2 \sin(z_1) & -z_2 \sin(z_1) & z_2 \cos(z_1) & z_2 \cos(z_1) & 0 & 0 \\ \cos(z_1) & \cos(z_1) & \sin(z_1) & \sin(z_1) & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ -\frac{z_5 \sin(z_4)}{\cos(\frac{z_5}{2})} & \frac{z_5 \sin(z_4)}{\cos(\frac{z_5}{2})} & \frac{z_5 \cos(z_4)}{\sin(\frac{z_5}{2})} & -\frac{z_5 \cos(z_4)}{\sin(\frac{z_5}{2})} & 0 & 0 \\ \frac{z_5 \cos(z_4)}{\sin(\frac{z_5}{2})} & -\frac{z_5 \cos(z_4)}{\sin(\frac{z_5}{2})} & \frac{\sin(z_5)}{2} & -\frac{\sin(z_5)}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & -\frac{1}{2} \end{pmatrix} \right| = z_2 z_5 > 0$$

Ce calcul nous sera utile par la suite, lorsque nous souhaiterons exprimer le rapport de Green associé à cette transformation.

3. Calcul du rapport de Green R_d .

4. Avec la probabilité $\alpha_d = \min(1, R_d)$, on accepte la nouvelle configuration $\mathbf{y} = \mathbf{x} \setminus \{x_i\} \cup \{u_1 \cup u_2\}$. Avec la probabilité $1 - \alpha_d$, on reste dans la configuration courante: $\mathbf{y} = \mathbf{x}$.

▷ Bijection

Il est à noter qu'il existe une bijection associant les deux mouvements de fusion et de division à travers les objets et les variables aléatoires mises en jeu. En effet, à un vecteur de $U \times U \times Z_f$, on associe un et un seul vecteur sur $U \times Z_d$, et réciproquement :

$$\left(\begin{array}{l} U \times U \times Z_f \\ (X_{x_i}, Y_{x_i}, R_{x_i}, X_{x_j}, Y_{x_j}, R_{x_j}, z_1, z_2, z_3) \end{array} \leftrightarrow \begin{array}{l} U \times Z_d \\ (X_u, Y_u, R_u, z'_1, z'_2, z'_3, z'_4, z'_5, z'_6) \end{array} \right) \quad (3.36)$$

Ceci montre bien la symétrie des transformations proposées. On remarque alors que $z_2 = z'_2$, ce qui fait que par la suite nous noterons simplement z_2 dans les deux cas. On rappelle aussi que par construction, $z'_5 = d(x_i, x_j)$.

► Noyaux de propositions du fusion et de division

Soit $Q_{fd}(\cdot, \cdot)$ le noyau de propositions du mouvement de fusion/division. Il s'écrit :

$$Q_{fd}(\mathbf{x}, \cdot) = p_f Q_f(\mathbf{x}, \cdot) + p_d Q_d(\mathbf{x}, \cdot).$$

Explicitons pour chacun des mouvements le noyau propre qui lui est associé :

▷ Fusion

$$\begin{aligned} Q_f(\mathbf{x}, A) &= \frac{1}{n_{\sim_I}(\mathbf{x})} \sum_{x_i \sim_I x_j} \int_{Z_f} \mathbf{1}_A(\mathbf{x} \setminus \{x_i \cup x_j\} \cup \{\eta_{x_i, x_j}(z)\}) \overbrace{f_f(z) \lambda(dz)}^{d\mathbf{P}_f(z)} \\ &= \frac{1}{n_{\sim_I}(\mathbf{x})} \sum_{x_i \sim_I x_j} \int_{\eta_{x_i, x_j}(Z_f) \subset U} \mathbf{1}_A(\mathbf{x} \setminus \{x_i \cup x_j\} \cup \{u\}) f_f(\eta_{x_i, x_j}^{-1}(u)) |J_{\eta_{x_i, x_j}^{-1}}(u)| \lambda(du) \\ &= \frac{1}{n_{\sim_I}(\mathbf{x})} \int_U \sum_{x_i \sim_I x_j} \mathbf{1}_A(\mathbf{x} \setminus \{x_i \cup x_j\} \cup \{u\}) \mathbf{1}_{u \in \eta_{x_i, x_j}(Z_f)} f_f(\eta_{x_i, x_j}^{-1}(u)) |J_{\eta_{x_i, x_j}^{-1}}(u)| \lambda(du) \end{aligned}$$

▷ Division

$$\begin{aligned} Q_d(\mathbf{x}, A) &= \frac{1}{n(\mathbf{x})} \sum_{x_i \in \mathbf{x}} \int_{Z_d} \mathbf{1}_A(\mathbf{x} \setminus \{x_i\} \cup \{\zeta_{x_i}(z)\}) \overbrace{f_d(z) \lambda(dz)}^{d\mathbf{P}_d(z)} \\ &= \frac{1}{n(\mathbf{x})} \sum_{x_i \in \mathbf{x}} \int_{\zeta_{x_i}(Z_d) \subset U \times U} \mathbf{1}_A(\mathbf{x} \setminus \{x_i\} \cup \{u_1 \cup u_2\}) f_d(\zeta_{x_i}^{-1}(u_1, u_2)) |J_{\zeta_{x_i}^{-1}}(u_1, u_2)| \lambda(du_1) \lambda(du_2) \\ &= \frac{1}{n(\mathbf{x})} \int_{U^2} \sum_{x_i \in \mathbf{x}} \mathbf{1}_A(\mathbf{x} \setminus \{x_i\} \cup \{u_1 \cup u_2\}) \mathbf{1}_{(u_1, u_2) \in \zeta_{x_i}(Z_d)} f_d(\zeta_{x_i}^{-1}(u_1, u_2)) |J_{\zeta_{x_i}^{-1}}(u_1, u_2)| \lambda(du_1) \lambda(du_2) \end{aligned}$$

► Mesure symétrique ξ_{fd}

Pour le mouvement de fusion/division, on définit comme suit la mesure symétrique $\xi_{fd}(\cdot, \cdot)$ sur $\Psi \times \Psi$:

$$\begin{aligned} \xi_{fd}(A, B) &= \int_{\Psi} \int_{u \in U} \mathbf{1}_A(\mathbf{x}) \sum_{(x_i, x_j) \in \mathbf{x}} \mathbf{1}_B(\mathbf{x} \setminus \{x_i \cup x_j\} \cup \{u\}) \lambda(du) \mu(d\mathbf{x}) \\ &\quad + \int_{\Psi} \int_{(u_1, u_2) \in U} \mathbf{1}_A(\mathbf{x}) \sum_{u \in \mathbf{x}} \mathbf{1}_B(\mathbf{x} \setminus \{u\} \cup \{u_1 \cup u_2\}) \frac{\lambda(du_1) \lambda(du_2)}{2} \mu(d\mathbf{x}). \end{aligned}$$

Le $\frac{1}{2}$ apparaît dans la deuxième partie de l'expression puisque du point de vue de l'espace des configurations Ψ , étant donné qu'il est non ordonné, apporter à une configuration \mathbf{x} deux objets $\{u_1 \cup u_2\}$ est équivalent à lui apporter les deux objets $\{u_2 \cup u_1\}$. Il convient de ne compter qu'une fois ce couple pour obtenir la symétrie de la mesure $\xi_{fd}(\cdot, \cdot)$.

Pour montrer que cette mesure est symétrique, il nous suffit alors de vérifier l'égalité suivante :

$$\xi_{fd}(A_{n+1}, B_n) = \xi_{fd}(B_n, A_{n+1}),$$

où l'on rappelle que $A_n = A \cap \Psi_n$.

Or, d'après la définition de la mesure $\mu(\cdot)$ de référence (2.3) :

$$\begin{aligned} \xi_{fd}(A_{n+1}, B_n) &= \int_{\Psi_{n+1}} \int_{u \in U} \mathbf{1}_{A_{n+1}}(\mathbf{x}) \sum_{(x_i, x_j) \in \mathbf{x}} \mathbf{1}_{B_n}(\mathbf{x} \setminus \{x_i \cup x_j\} \cup \{u\}) \lambda(du) \mu(d\mathbf{x}) \\ &= \frac{e^{-\lambda(U)}}{(n+1)!} \int_{U^{n+1}} \int_{u \in U} \mathbf{1}_{A_{n+1}}(\mathbf{x}) \sum_{(x_i, x_j) \in \mathbf{x}} \mathbf{1}_{B_n}(\mathbf{x} \setminus \{x_i \cup x_j\} \cup \{u\}) \underbrace{\lambda(du) \lambda^{n+1}(d\mathbf{x})}_{\lambda^{n+2}(\cdot)} \\ &= \frac{e^{-\lambda(U)}}{(n+1)!} \int_{U^{n+1}} \int_{u \in U} C_{n+1}^2 \mathbf{1}_{A_{n+1}}(x_1, \dots, x_{n+1}) \mathbf{1}_{B_n}(\mathbf{x} \setminus \{x_n \cup x_{n+1}\} \cup \{u\}) \lambda^{n+2}(\cdot) \\ &= \frac{e^{-\lambda(U)}}{n!} \int_{U^n} \int_{(x_n, x_{n+1}) \in U} \frac{n}{2} \mathbf{1}_{B_n}(\mathbf{y} = \{x_1, \dots, x_{n-1}, u\}) \mathbf{1}_{A_{n+1}}(\mathbf{y} \setminus \{u\} \cup \{x_n \cup x_{n+1}\}) \lambda^{n+2}(\cdot) \\ &= \frac{e^{-\lambda(U)}}{n!} \int_{U^n} \int_{(x_n, x_{n+1}) \in U} \mathbf{1}_{B_n}(\mathbf{y}) \frac{1}{2} \sum_{u \in \mathbf{y}} \mathbf{1}_{A_{n+1}}(\mathbf{y} \setminus \{u\} \cup \{x_n \cup x_{n+1}\}) \lambda^{n+2}(\cdot) \\ &= \int_{\Psi_n} \int_{(x_n, x_{n+1}) \in U} \mathbf{1}_{B_n}(\mathbf{y}) \sum_{u \in \mathbf{y}} \mathbf{1}_{A_{n+1}}(\mathbf{y} \setminus \{u\} \cup \{x_n \cup x_{n+1}\}) \frac{\lambda(dx_n) \lambda(dx_{n+1})}{2} \mu(d\mathbf{y}) \\ &= \xi_{fd}(B_n, A_{n+1}). \end{aligned}$$

Cela montre bien que la mesure $\xi_{fd}(\cdot, \cdot)$ ainsi définie sur $\Psi \times \Psi$ est symétrique.

► Dérivées de Radon Nikodym

Avant d'accéder à l'expression des rapports de Green, nous devons calculer les dérivées de Radon Nikodym associées à chacun des mouvements.

▷ Fusion

Afin d'alléger les notations, nous noterons dans la suite de cette sous-partie

$$\mathbf{y} = \mathbf{x} \setminus \{x_i \cup x_j\} \cup \{u\}.$$

Comme requis dans l'algorithme de Green, $\pi(\cdot)Q_{fd}$ est absolument continu par rapport à la mesure symétrique $\xi_{fd}(\cdot, \cdot)$ puisque

$$\pi(d\mathbf{x})Q_{fd}(\mathbf{x}, d\mathbf{y}) = h(\mathbf{x})\mu(d\mathbf{x}) \frac{p_f}{n_{\sim I}(\mathbf{x})} f_f(\eta_{x_i, x_j}^{-1}(u)) |J_{\eta_{x_i, x_j}^{-1}}(u)| \lambda(du)$$

et

$$\xi_{fd}(d\mathbf{x}, d\mathbf{y}) = \lambda(du)\mu(d\mathbf{x})$$

La dérivée de Radon Nikodym associée à ce mouvement est donc :

$$f(\mathbf{x}, \mathbf{y}) = p_f \frac{h(\mathbf{x})}{n_{\sim_I}(\mathbf{x})} \frac{f_f(\eta_{x_i, x_j}^{-1}(u))}{z_2}$$

▷ Division

Afin d'alléger les notations, nous noterons dans la suite de cette sous-partie

$$\mathbf{y} = \mathbf{x} \setminus \{x_i\} \cup \{u_1 \cup u_2\}$$

Comme requis dans l'algorithme de Green, $\pi(\cdot)Q_{f_d}$ est absolument continu par rapport à la mesure symétrique $\xi_{f_d}(\cdot, \cdot)$ puisque

$$\pi(d\mathbf{x})Q_{f_d}(\mathbf{x}, d\mathbf{y}) = h(\mathbf{x})\mu(d\mathbf{x}) \frac{p_d}{n(\mathbf{x})} f_d(\zeta_{x_i}^{-1}(u_1, u_2)) |J_{\zeta_{x_i}^{-1}}(u_1, u_2)| \lambda(du_1) \lambda(du_2)$$

et

$$\xi_{f_d}(d\mathbf{x}, d\mathbf{y}) = \lambda(du_1) \lambda(du_2) \mu(d\mathbf{x})$$

La dérivée de Radon Nikodym associée à ce mouvement est donc :

$$f(\mathbf{x}, \mathbf{y}) = p_d \frac{h(\mathbf{x})}{n(\mathbf{x})} \frac{f_d(\zeta_{x_i}^{-1}(u_1, u_2))}{z_2 \cdot d(u_1, u_2)}$$

► Rapports de Green

Nous avons désormais accès aux rapports de Green et donc aux probabilités d'acceptation des mouvements de fusion et de division.

Pour la fusion :

$$\begin{aligned} R_f(\mathbf{x}, \mathbf{x} \setminus \{x_i \cup x_j\} \cup \{u\}) &= \frac{f(\mathbf{y}, \mathbf{x})}{f(\mathbf{x}, \mathbf{y})} \\ &= \frac{p_d h(\mathbf{y})}{p_f h(\mathbf{x})} \frac{n_{\sim_I}(\mathbf{x})}{n(\mathbf{x}) - 1} \frac{f_d(\zeta_u^{-1}(x_i, x_j))}{f_f(\eta_{x_i, x_j}^{-1}(u))} \frac{1}{d(x_i, x_j)} \end{aligned}$$

Pour la division :

$$\begin{aligned} R_d(\mathbf{x}, \mathbf{x} \setminus \{x_i\} \cup \{u_1 \cup u_2\}) &= \frac{f(\mathbf{y}, \mathbf{x})}{f(\mathbf{x}, \mathbf{y})} \\ &= \frac{p_f h(\mathbf{y})}{p_d h(\mathbf{x})} \frac{n(\mathbf{x})}{n_{\sim_I}(\mathbf{y})} \frac{f_f(\eta_{u_1, u_2}^{-1}(x_i))}{f_d(\zeta_{x_i}^{-1}(u_1, u_2))} d(x_i, x_j) \end{aligned}$$

► Algorithme

Voici reformulé l'algorithme de Metropolis-Hastings-Green à sauts réversibles que nous avons implanté. Il tient compte des différents noyaux de propositions des mouvements simples et des mouvements complexes.

Algorithme A

1. A partir de la configuration courante \mathbf{x}_t :

→ on choisit avec la probabilité $p_i(\mathbf{x}_t)$, un des noyaux de propositions $Q_i(\mathbf{x}_t, \cdot)$ et on passe à l'étape 2.

Dans notre approche, ces probabilités seront indépendantes de \mathbf{x}_t . Il s'agit de p_{nm} (noyau de naissance/mort), p_{fd} (noyau de fusion/division), p_{tr} (noyau de translation), p_{di} (noyau de dilatation), p_{trdi} (noyau de translation combinée à une dilatation).

→ avec la probabilité $p_\emptyset = 1 - \sum_{i \in I} p_i$, on reste dans l'état courant et on passe à la prochaine itération.

2. On génère la configuration \mathbf{y} , connaissant la configuration \mathbf{x}_t , selon le noyau sélectionné $Q_i(\mathbf{x}_t, \cdot)$.

3. On calcule le **rapport de Green** relatif au noyau i , $R_i(\mathbf{x}_t, \mathbf{y})$:

$$R_i(\mathbf{x}_t, \mathbf{y}) = \frac{f_i(\mathbf{y}, \mathbf{x}_t)}{f_i(\mathbf{x}_t, \mathbf{y})} = \frac{\pi(d\mathbf{y})Q_i(\mathbf{y}, d\mathbf{x}_t)}{\pi(d\mathbf{x}_t)Q_i(\mathbf{x}_t, d\mathbf{y})} \quad (3.37)$$

4. Avec la probabilité $\alpha_i(\mathbf{x}_t, \mathbf{y}) = \min(1, R_i(\mathbf{x}_t, \mathbf{y}))$ on accepte le nouvel état \mathbf{y} , sinon on conserve \mathbf{x}_t .

► Convergence de l'algorithme

L'irréductibilité de la chaîne se démontre de la même façon que précédemment. En revanche, la récurrence de Harris, suite à l'introduction d'un nouveau mouvement qui permet à la chaîne de changer de dimension dans l'espace des configurations, doit être redémontrée. Pour cela, nous nous appuyerons sur un lemme ainsi que sur 3 conditions nécessaires pour assurer cette propriété.

Lemme 3.8 : Propriété sur $n_{\sim_I}(\mathbf{x})$

Il existe un entier N_{dec} tel que pour toute configuration de points de $U = \mathcal{P} \times \mathcal{K}$, notée \mathbf{x} , on ait

$$n(\mathbf{x}) \geq N_{dec} \implies n_{\sim_I}(\mathbf{x}) \geq \frac{n(\mathbf{x})(n(\mathbf{x}) - N_{dec})}{2 \cdot N_{dec}}. \quad (3.38)$$

Autrement dit, le rapport $\frac{n(\mathbf{x})}{n_{\sim_I}(\mathbf{x})}$ tend vers 0 quand $n(\mathbf{x}) \rightarrow \infty$.

◇ Démonstration :

Cette démonstration est disponible dans l'annexe (C.4).

◇

► Quelques conditions

– CONDITION c_1 : majoration de la densité $f_f(z)$.

Dans notre modèle, celle-ci est obtenue par construction puisque

$$f_f(z) = \frac{1}{2\pi} f_{z_2}(z_2) \frac{1}{M_{z_3} - m_{z_3}} \leq \frac{\min(1, R_M - R_m)}{2\pi} f_{z_2}(0) = M_{f_f}.$$

– CONDITION c_2 : minoration de la densité $f_d(z)$.

Celle-ci est aussi obtenue par construction puisque

$$f_d(z) = \left(\frac{1}{2\pi}\right)^2 f_{z_2}(z_2) \frac{1}{M_{z_3} - m_{z_3}} \frac{1}{2(R_{x_i} + z_3)} \frac{1}{M_{z_6}} \geq \frac{2 \cdot \min(1, R_M - R_m)}{(2\pi)^2 2R_M(R_M - R_m)} f_{z_2}(1) = m_{f_d}.$$

– CONDITION c_3 : minoration de l'intensité conditionnelle de Papangelou.

Une nouvelle fois, le choix de la densité que nous simulons implique automatiquement la minoration de l'intensité conditionnelle de Papangelou, comme démontré dans ce chapitre :

$$\frac{h(\mathbf{x} \cup \{\xi\})}{h(\mathbf{x})} \geq m_h, \forall \xi \in \{U - \bigcup_{i=1}^{n(\mathbf{x})} x_i\}$$

Proposition 3.9 : Récurrence de Harris, Ergodicité géométrique

|| Si la densité non normalisée $h(\cdot)$ vérifie la condition de stabilité (2.11), ainsi que les conditions c_1 , c_2 et c_3 explicitées précédemment, alors **l'algorithme A** simule une chaîne de Markov récurrente au sens de Harris et ergodique géométriquement.

◇ **Démonstration :**

Cette démonstration est disponible dans l'annexe (C.5).

◇

Nous avons donc démontré la convergence vers la loi objectif $\pi(\cdot)$ de notre algorithme de simulation de processus objet.

3.4. Bilan : notre algorithme

Voici en résumé, l'algorithme de processus ponctuels simulés par une chaîne de Markov que nous avons implanté. Sont décrites ici les principales étapes de celui-ci, programmé en C++.

⊗ *Initialisation*

- Lecture de l'image de données.
- Entrée des paramètres.
- Initialisation de la configuration $\mathbf{x}_{t=0}$ à la configuration nulle, c'est à dire ne comportant aucun objet.

⊗ *Boucle d'itération*

○ Itération numéro k .

- Calcul de la température T_k .
- Choix d'un des 5 noyaux de propositions, avec les probabilités $\{p_{nm}, p_{fd}, p_{tr}, p_{di}, p_{trdi}\}$. Avec la probabilité complémentaire p_\emptyset , on impose à la chaîne de rester dans l'état courant.
- Proposition d'une nouvelle configuration \mathbf{y} à partir de la configuration courante \mathbf{x}_k , faite par le noyau sélectionné.
- Acceptation ou Rejet, suite au calcul du rapport de Green.
- Fin de l'itération k . $k = k + 1$.

⊗ *Fin de l'algorithme*

Quand la température T_k est inférieure à une limite donnée, on stoppe les itérations. Affichage des objets de la configuration sur l'image de données. Analyse du résultat obtenu.

Résultats, Interprétations des résultats et Perspectives

★★★

Nous avons effectué une série de tests de notre modèle sur des images. Tout d'abord, ils furent menés sur des images synthétiques, dans le but d'améliorer la syntaxe C++ et la vitesse de l'algorithme (au sens informatique, et non mathématique), puis sur des images fournies par l'IFN.

Dans ce chapitre, nous présenterons et exploiterons ces résultats, puis nous les comparerons avec ceux obtenus, à partir d'une même image, avec la méthode de suivi de vallées que nous avons implantée d'après les directives disponibles dans [5].

Les critères d'évaluation furent difficiles à déterminer, d'autant plus que nous ne disposions pas d'une vérité terrain censée être la "solution" à obtenir. Aussi, nous axerons principalement nos interprétations des images sur les erreurs importantes de détection que nous pouvons repérer.

Nous remarquerons ainsi l'importance du noyau de propositions de fusion/division, dont la présence améliore considérablement les résultats obtenus, et donnerons des perspectives de recherche à mener par la suite.

La machine sur laquelle nous avons travaillé est une *SUN - 440 MHz*.

★★★

4.1. Images étudiées

Dans un premier temps, nous avons extrait dans l'image de la Fig. (3.1) fournie par l'IFN des zones de peupliers d'environ 60m sur 60m (128×128 pixels), afin d'accélérer le traitement informatique, la vitesse de décroissance des températures du recuit simulé se devant d'être d'autant plus faible que l'espace objet à explorer est grand.

Les Fig. (4.1) et Fig. (4.2) montrent les 2 imageries qui ont été extraites de l'image des forêts de Saône et Loire, sur lesquelles nous avons testé nos algorithmes.

Puis, d'autres images plus grosses, équivalentes à 1 hectare sur le terrain, ont été extraites de la Fig. (3.1). La Fig. (4.3) représente une image de 200×140 pixels sur laquelle les résultats décrits par la suite ont été obtenus.

Toutes les images de notre base de données comportent des zones de peupliers à extraire parmi de la végétation, des vignes ou d'autres espèces d'arbres, ou des fleuves. Nous avons dans un premier temps contourné ce problème, en choisissant d'extraire manuellement les parties qui nous intéressaient.

L'idéal aurait été d'établir un processus automatique de segmentation de l'image par espèce d'arbres, en utilisant par exemple des informations de textures afin d'obtenir une signature particulière pour chacune des espèces. Suivant les objectifs que nous nous fixerons par la suite, nous aurons ou non à réaliser une telle classification.

Les images originales sont acquises par un capteur comportant trois bandes spectrales, puis transformées en images RGB (format JPG). Ne sachant pas comment cette transformation en RGB est effectuée, nous les avons transformés en images de niveaux de gris en prenant simplement la moyenne sur les trois composantes. Par la suite, lorsque nous serons renseignés sur les transformations subies par l'image avant notre traitement, il sera intéressant d'utiliser à meilleur escient l'information contenue dans chacune des bandes.

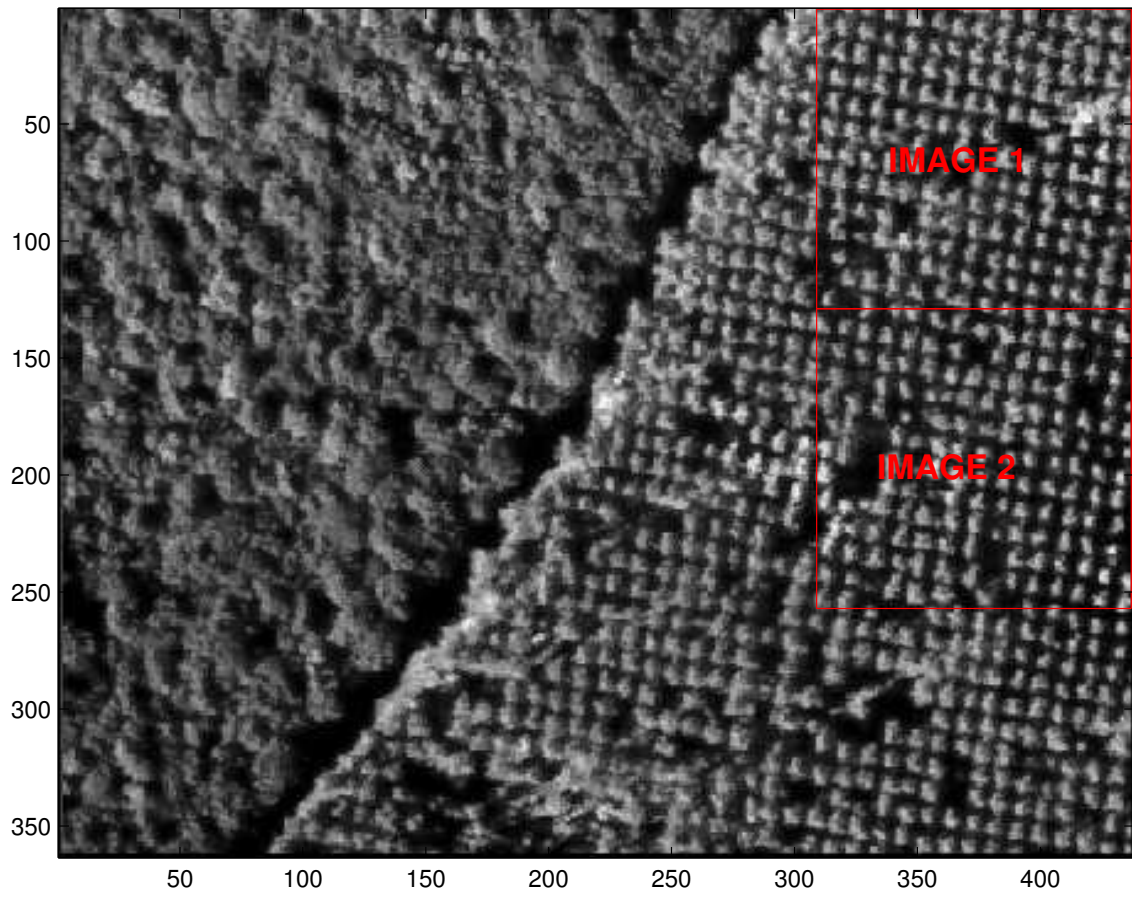


FIG. 4.1 – Partition de l'image initiale. Extrait de l'image originale, les carrés rouges représentent les deux imageries. ©IFN.

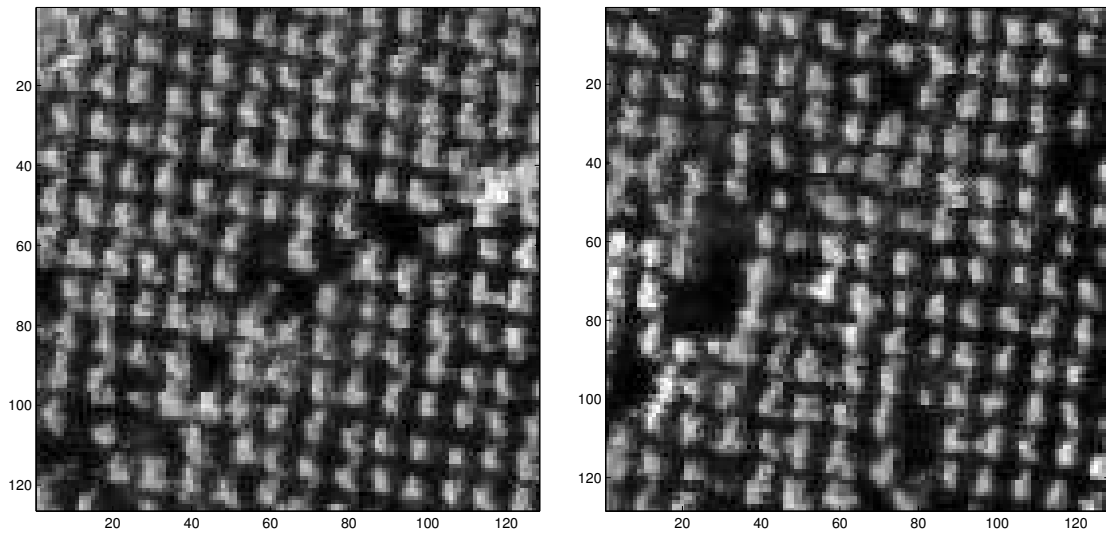


FIG. 4.2 – Partition de l'image initiale. A gauche : Image 1, 128×128 pixels. A droite : Image 2, 128×128 pixels. ©IFN.

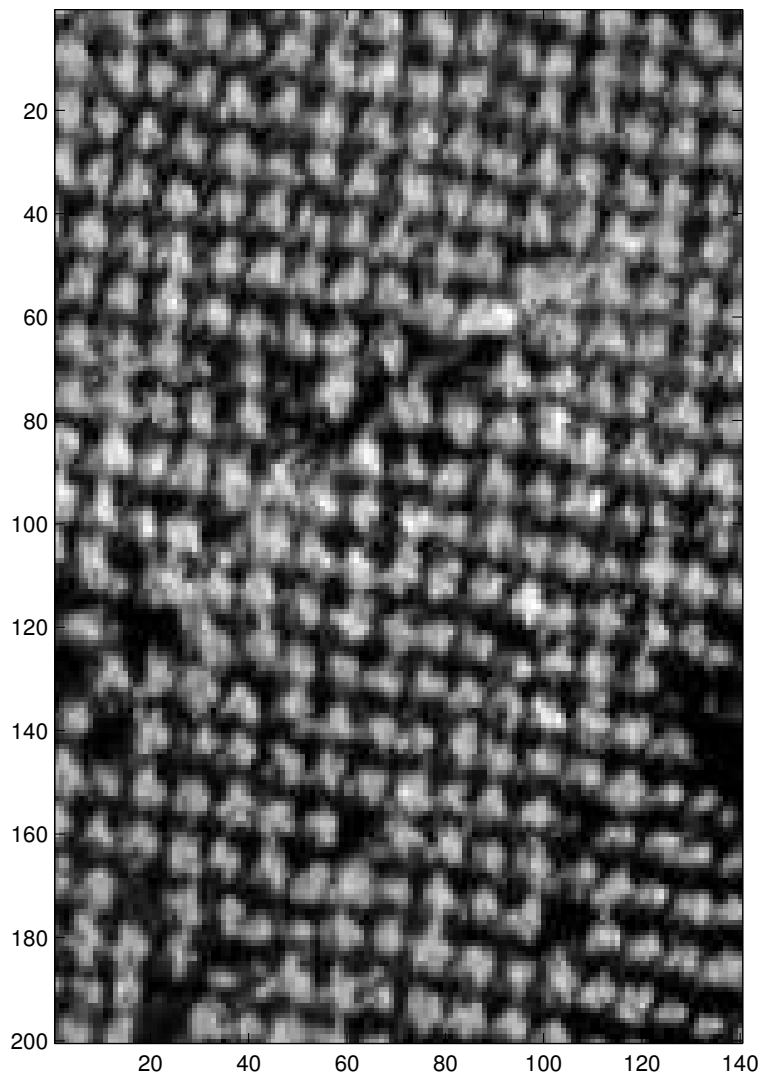


FIG. 4.3 – *Extrait de l'image de Saône et Loire, 200 × 140 pixels.* ©IFN.

4.2. Résultats et Interprétations

Dans cette partie, nous allons retracer les étapes de notre raisonnement qui nous ont par exemple amenés à élaborer un noyau de propositions de fusion/division d'objets. Tout d'abord, les résultats obtenus sans ce noyau seront exposés. Ceux-ci étant très semblables sur les deux imagettes 1 et 2, nous ne présenterons que ceux relatifs à l'imagette 1.

Puis, nous tirerons des interprétations issues de ces premiers résultats des conséquences qui ont fait évoluer notre modèle.

Nous présenterons alors notre deuxième série de tests, réalisés à partir de l'image de la Fig. (4.3).

4.2.1 Choix des paramètres

Si l'on se réfère à la partie précédente présentant notre modèle, les différents paramètres de ce dernier sont :

1. pour la densité a priori $h_p(\cdot)$: β et γ_a .

Nous avons utilisé une répulsion par interaction d'aires, avec un coefficient d'interaction assez fort : $\gamma_{aire} = \ln(0.0001)$. Celle-ci répond correctement à nos attentes, car elle pénalise les objets d'autant plus qu'ils se superposent à d'autres.

Nous avons de plus remarqué que le coefficient pondérateur β agissait principalement sur la dynamique de l'algorithme lorsque la température est élevée, en faisant naître plus ou moins d'objets, objets qui seront transformés voire tués par la suite s'ils sont mal positionnés. A faible température, son influence est négligeable comparée à celle de $h(\mathbf{x})^{\frac{1}{T}}$.

Nous devons donc à la fois faire naître assez d'objets dans la phase transitoire de l'algorithme (forte température), mais pas trop, afin d'éviter de perdre du temps par la suite en tuant ces objets qui s'avèrent mal placés. Le choix de $\beta = 10$ s'est avéré judicieux dans la pratique.

2. pour le terme de vraisemblance $\mathcal{L}(\mathcal{I}|\mathbf{x})$: les moyennes et variances des deux classes, à savoir $\mu_A, \sigma_A, \mu_F, \sigma_F$.

Au moment d'initialiser l'algorithme, nous choisissons les paramètres initiaux des deux classes. Des procédés automatiques (k-means) peuvent nous aider dans cette décision. Puis, nous lançons une première série d'itérations. A partir de celle-ci, nous estimons à nouveau ces paramètres du modèle en sélectionnant des échantillons parmi les configurations d'objets, et en utilisant les théorèmes limites (B.9).

Dans l'imagette 1, l'estimation donne alors : $\mu_A = 115$, $\sigma_A = 41$, $\mu_F = 48$, et $\sigma_F = 27$. La Fig. (4.4) montre que les distributions de ces classes sont éloignées, ce qui nous laisse présager une classification de qualité.

3. pour le recuit simulé : $T_{t=0}, \alpha_{temp}, T_{min}$.

Nous avons opté pour une décroissance géométrique par palliers, de largeur s . Ainsi, les températures $(T_{ks}, \dots, T_{(k+1)s-1})$ restent égales, et $T_{(k+1)s} = \alpha_{temp} \cdot T_{ks}$. Aux paramètres du recuit simulé s'ajoute donc s la largeur du pallier.

La décroissance géométrique fut assurée avec le paramètre $\alpha_{temp} = 0.99995$. Même si ceci peut paraître très lent, il n'en est rien en réalité : nous rappelons en effet que théoriquement, il

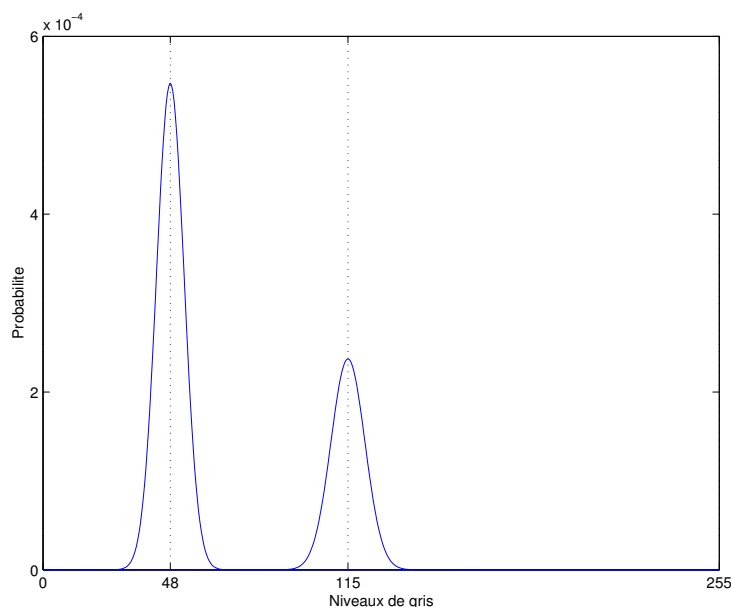


FIG. 4.4 – Répartition des niveaux de gris des deux classes

aurait fallu une décroissance logarithmique de la température pour nous assurer de la convergence de l'algorithme.

4. pour les noyaux de propositions : $p_{nm}, p_{fd}, p_{tr}, p_{di}, p_{trdi}$.

A chaque itération, les différents noyaux de propositions furent choisis de façon équiprobable. Nous avons veillé à ce que la somme des probabilités soit inférieures à 1 pour assurer l'apériodicité de la chaîne.

4.2.2 Résultats obtenus sans le noyau de fusion/division

Nous présentons deux de nos résultats, sur les Fig. (4.5) et Fig. (4.6), représentatifs de la série de tests que nous avons menée. Dans le deuxième exemple, un refroidissement du recuit simulé plus rapide a été opéré.

Comme expliqué plus haut, l'absence de vérité terrain pour cette parcelle rend impossible l'interprétation de nos résultats au niveau du nombre d'arbres détectés. Par contre, nous pouvons observer plusieurs erreurs de détection de houppiers.

► Manque de mouvements et problèmes avec le recuit simulé

La Fig. (4.7) met en évidence les deux cas principaux d'erreurs de détection. Le cas 1 est rencontré lorsqu'un gros cercle s'étale sur deux arbres, et le cas 2 lorsque deux petits cercles sont nécessaires pour détecter un arbre. Avec les mouvements que nous lui autorisons, la configuration aurait dû effectuer deux étapes pour résoudre d'elle-même ce problème. Dans le premier cas, il s'agissait de translater et de réduire la taille du cercle, puis d'en faire naître un deuxième sur l'autre arbre (translation-dilatation + naissance). Dans le deuxième cas, il aurait fallu proposer de tuer un des deux cercles, puis de translater et d'agrandir l'autre (mort + translation-dilatation).

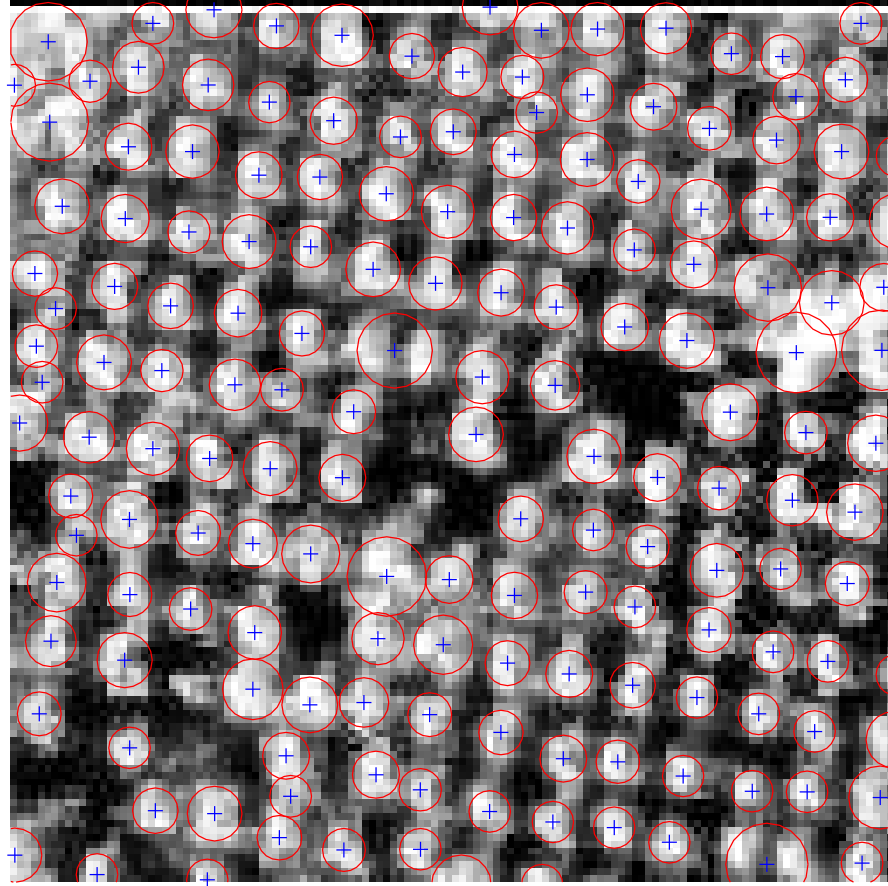


FIG. 4.5 – $T_{i=0} = 0.1$, $s = 100$, 10 millions d'itérations, $T_{min} = 6.73 \cdot 10^{-4}$, temps CPU: 812 secondes

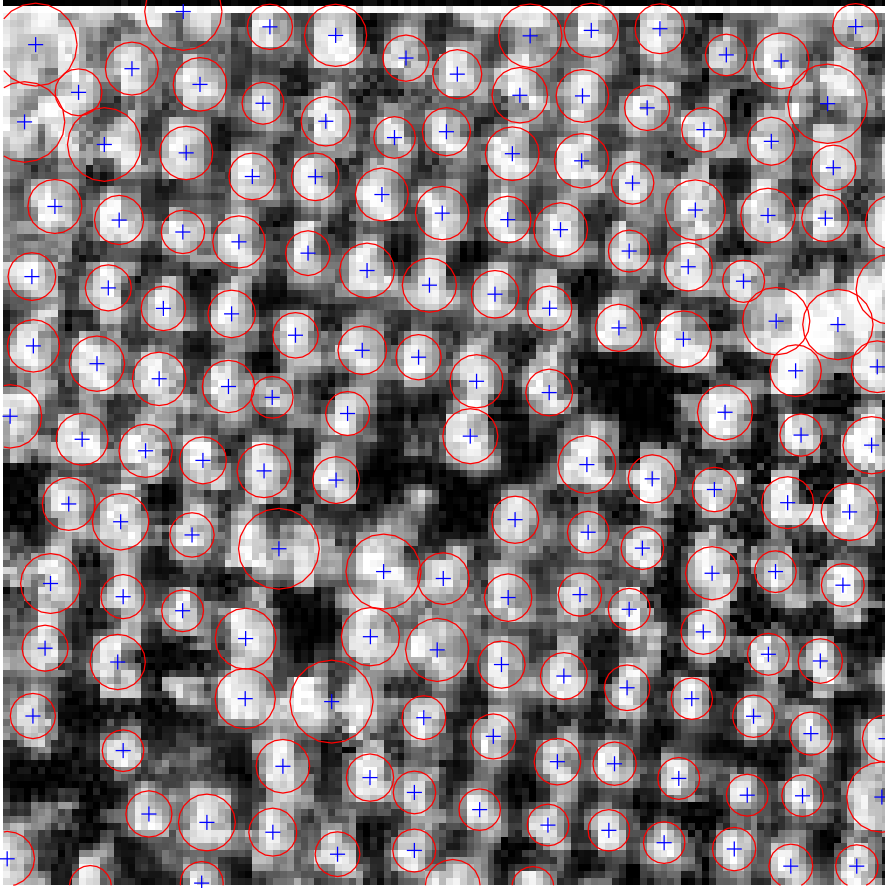


FIG. 4.6 - $T_{t=0} = 0.1$, $s = 10$, 1 million d'itérations, $T_{min} = 6.73 \cdot 10^{-4}$, temps CPU : 89 secondes

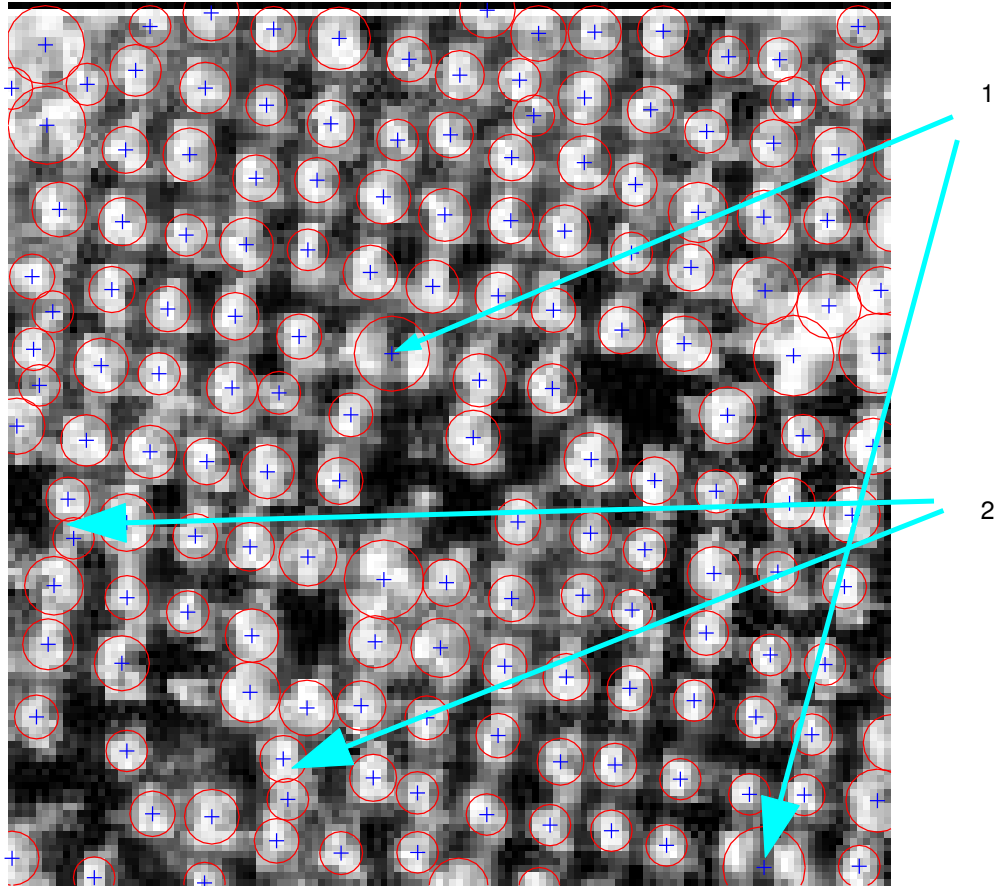


FIG. 4.7 – Quelques erreurs de détection. Cas 1 : sous-détection. Cas 2 : sur-détection.

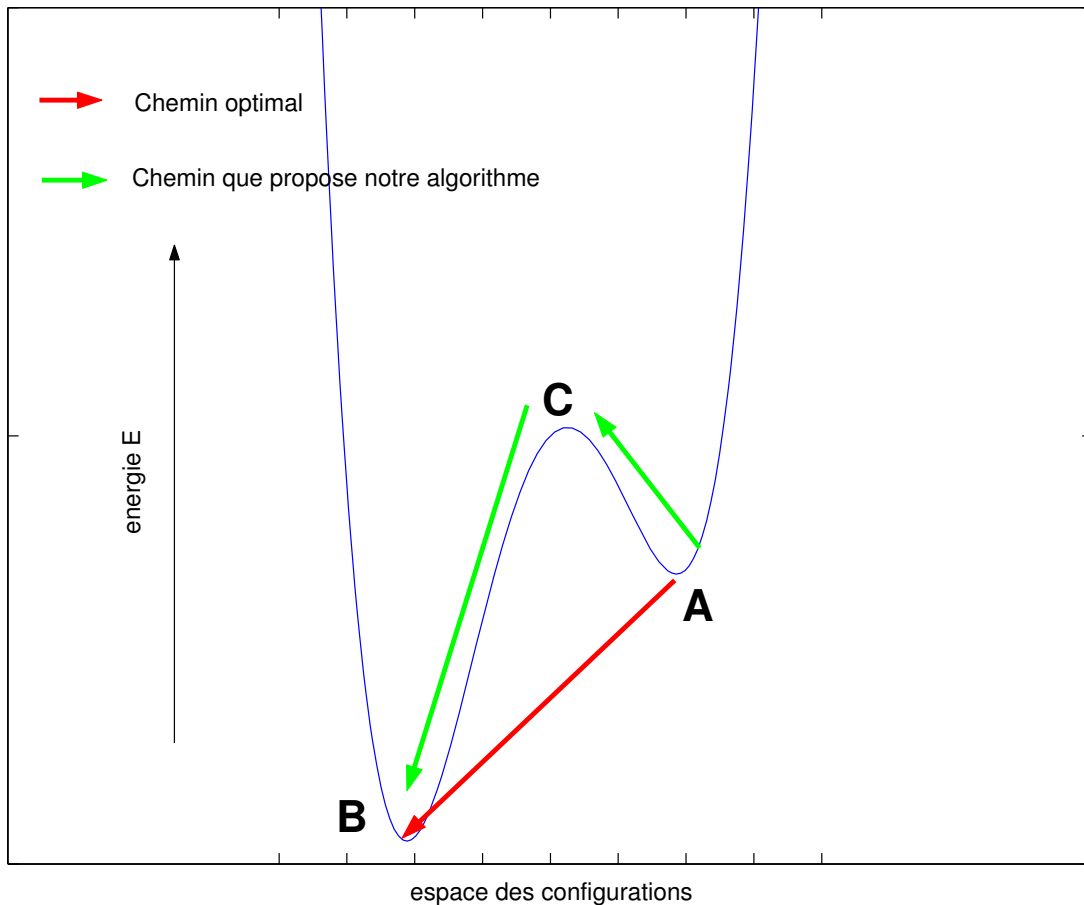


FIG. 4.8 – L’analogie énergétique

Raisonnons au niveau énergétique afin de mieux comprendre ce phénomène, et considérons qu’il nous faut trouver le minimum global d’une énergie $E(\mathbf{x})$, reliée à la densité a posteriori par : $h(\mathbf{x}) = e^{-\frac{E(\mathbf{x})}{T_k}}$. Reportons nous pour cela au schéma de la Fig. (4.8).

Nous nous retrouvons dans les deux cas “coincés” dans un minimum local, duquel nous ne pourrions sortir que s’il existait un seul mouvement capable de passer de la configuration actuelle, c’est à dire le point A sur le schéma, à une configuration d’énergie plus faible, le point B.

Avec les mouvements disponibles dans notre noyau de propositions, nous avons vu que pour sauter de A à B, il nous fallait passer par un point intermédiaire C, une configuration d’énergie plus grande que A. Or, le passage de A à C est rejeté d’autant plus facilement que la température est faible, puisque le rapport de Green est proportionnel à $e^{\frac{E(A)-E(C)}{T_k}}$. En effet, pour les températures proches de 0, le comportement de la chaîne est quasi déterministe, et n’accepte de nouvelles configurations que si elles ont une énergie plus faible que la configuration courante.

Autrement dit, c’est en baissant extrêmement lentement la température que nous pourrions éviter de nous retrouver bloqués dans un minimum local, afin d’avoir le temps d’effectuer ces deux mouvements. Le temps CPU en serait d’autant plus grand.

Une autre solution à ce problème est d’inclure dans le noyau de propositions des mouvements qui permettent de passer de A à B en une étape, à savoir la fusion de deux cercles, ou inversement la scission en deux d’un cercle. C’est pourquoi nous avons créé ce noyau, dont le détail des mouve-

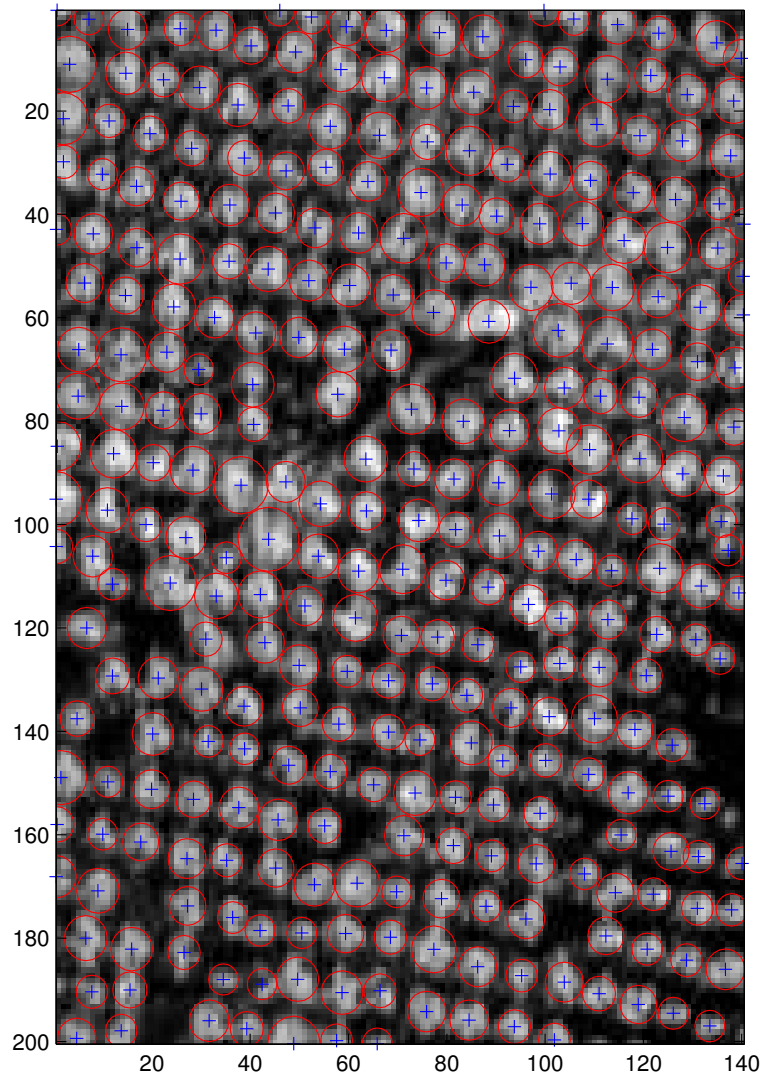


FIG. 4.9 – Extraction obtenue sur l'image 200×140 pixels avec le noyau de fusion/division. 1 million d'itérations, temps CPU : 212 secondes

ments est décrit dans la partie précédente du rapport.

4.2.3 Résultats obtenus avec le noyau de fusion/division

Nous présentons dans la Fig. (4.9) un nouveau résultat d'extraction des houppiers sur une image plus grande. Nous avons cette fois autorisé les mouvements de fusion et de division.

Nous remarquons que les erreurs de détection signalées précédemment ont disparu. Seule l'extraction des houppiers dans les zones denses reste parfois approximative.

A partir d'un tel résultat, il nous est possible d'exploiter la configuration finale obtenue dans le but d'obtenir des statistiques sur le peuplement telles que le nombre de peupliers, la densité ou la distribution des diamètres des arbres. Ceci est automatique puisque les coordonnées de chaque objet dans l'espace U représentent la position et le rayon des disques.

– nombre d'arbres : 306.

- densité: 374,8 arbres par hectare.
- diamètre moyen des arbres: 7,65 pixels soit environ 4m13.
- variance de la distribution des diamètres: 1,4187 pixels soit 0,7661m.
- taux de recouvrement (surface occupée par les arbres par rapport à la surface totale): approximativement 48%.

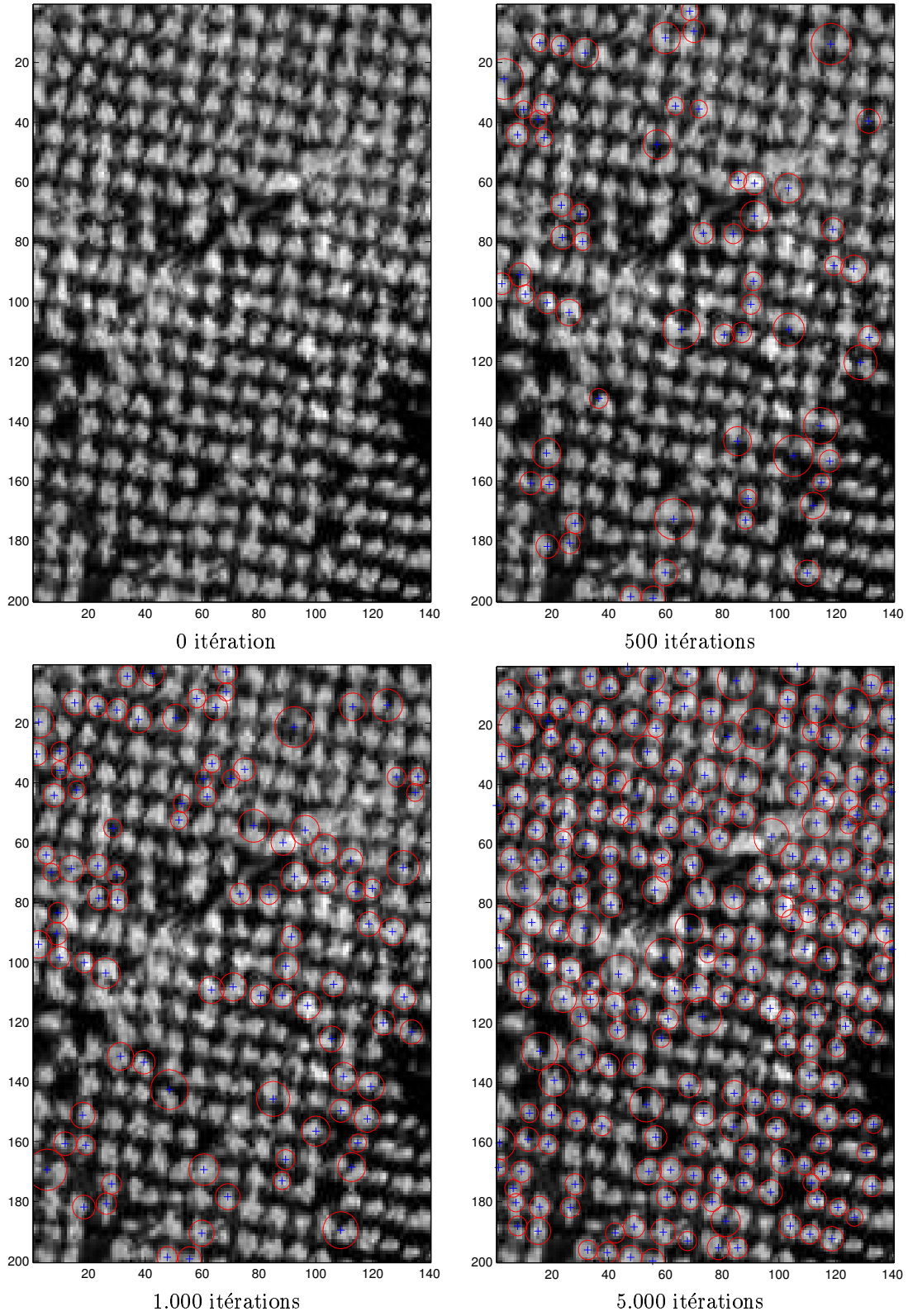
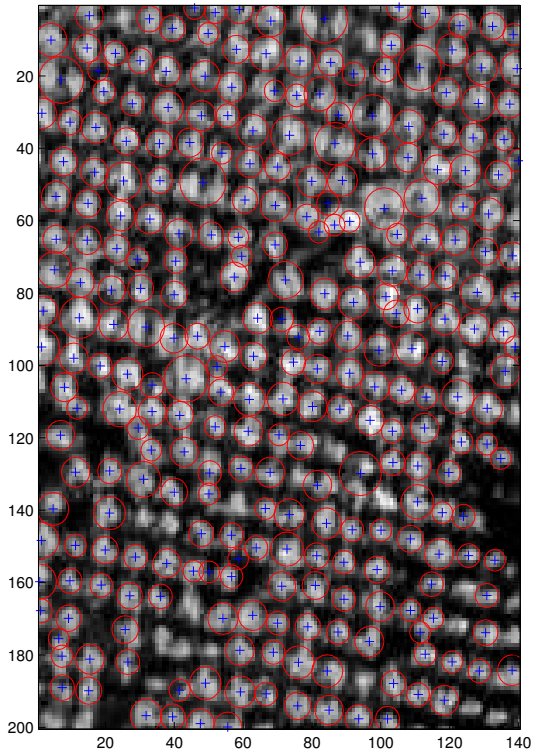
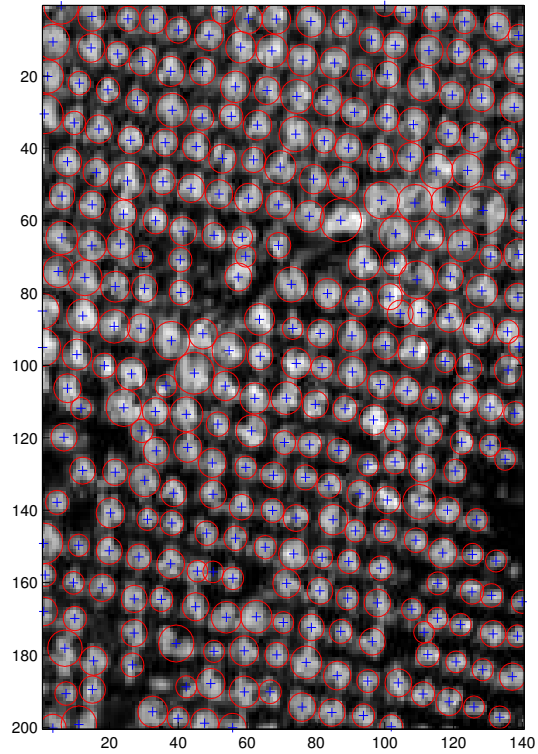


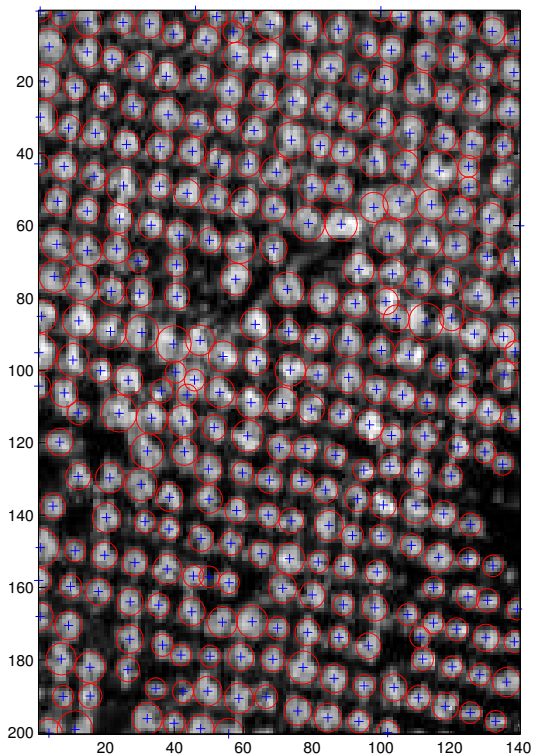
FIG. 4.10 – Evolution de la configuration d'objets en fonction du nombre d'itérations.



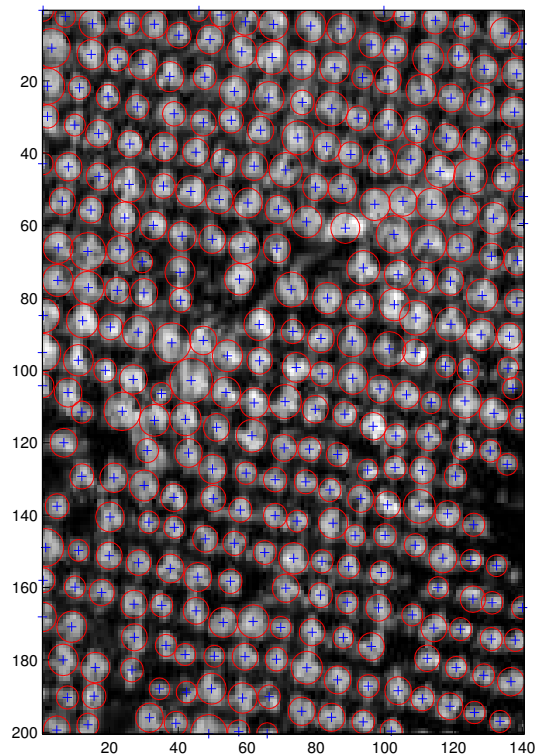
10.000 itérations



100.000 itérations



500.000 itérations



1.000.000 itérations

FIG. 4.11 – Evolution de la configuration d'objets en fonction du nombre d'itérations.

4.3. Comparaison avec le suivi de vallées

4.3.1 Résultats obtenus avec le suivi de vallées

Afin d'évaluer les performances de notre détection de houppiers à partir de processus objets, nous avons testé sur les mêmes images l'approche pixélique de l'algorithme de suivi de vallées. Les 4 images de la Fig. (4.12) montrent les résultats obtenus pour différentes valeurs des paramètres (s, θ) , qui sont respectivement les paramètres de filtrage et de seuillage.

Le meilleur résultat semble être obtenu pour les paramètres $(s=2, \theta = 70)$. La détection du contour de l'arbre n'est cependant pas toujours exacte, notamment dans les zones de forte densité. On peut imaginer qu'une telle méthode ne marche pas correctement avec d'autres espèces plus denses, comme le faisaient remarquer les auteurs.

4.3.2 Tableau comparatif

Nous souhaitons préciser une nouvelle fois que la série de tests réalisée sur des peupliers n'est que le début de notre travail de détection des houppiers à partir de processus objet, destiné à traiter une classe d'images plus vaste.

Il est clair que notre approche objet pourrait s'adapter plus facilement que l'approche pixélique à d'autres espèces d'arbres. Elle est plus évolutive que son homologue, cette dernière n'utilisant aucune autre information que les niveaux de gris. En effet une connaissance a priori des espèces peut se révéler précieuse, de même qu'une modélisation des interactions entre les arbres pour des milieux forestiers denses.

De plus, il est automatique avec notre modèle géométrique de disques de recueillir les informations souhaitées sur les objets, comme la distribution des diamètres des arbres, les types d'interactions entre les arbres, le taux de couvert, alors qu'un post-traitement de l'image est nécessaire dans le cadre du suivi de vallées pour obtenir ces statistiques.

Enfin, il existe des méthodes stochastiques d'estimation de nos paramètres, ce qui nous donne un avantage par rapport au suivi de vallées, pour lequel les paramètres doivent être évalués au moyen des expériences.

Critères d'évaluation	Suivi de vallées	Processus objet
Qualité de la détection	++	++
Rapidité de l'algorithme	++	-
Calcul des paramètres forestiers	+	++
Evolutions envisageables	-	++

TAB. 4.1 – *Tableau comparatif des deux méthodes testées, relatif aux peupleraies. Suivant les critères d'évaluation, des notes de -- à ++ sont attribuées aux deux approches.*

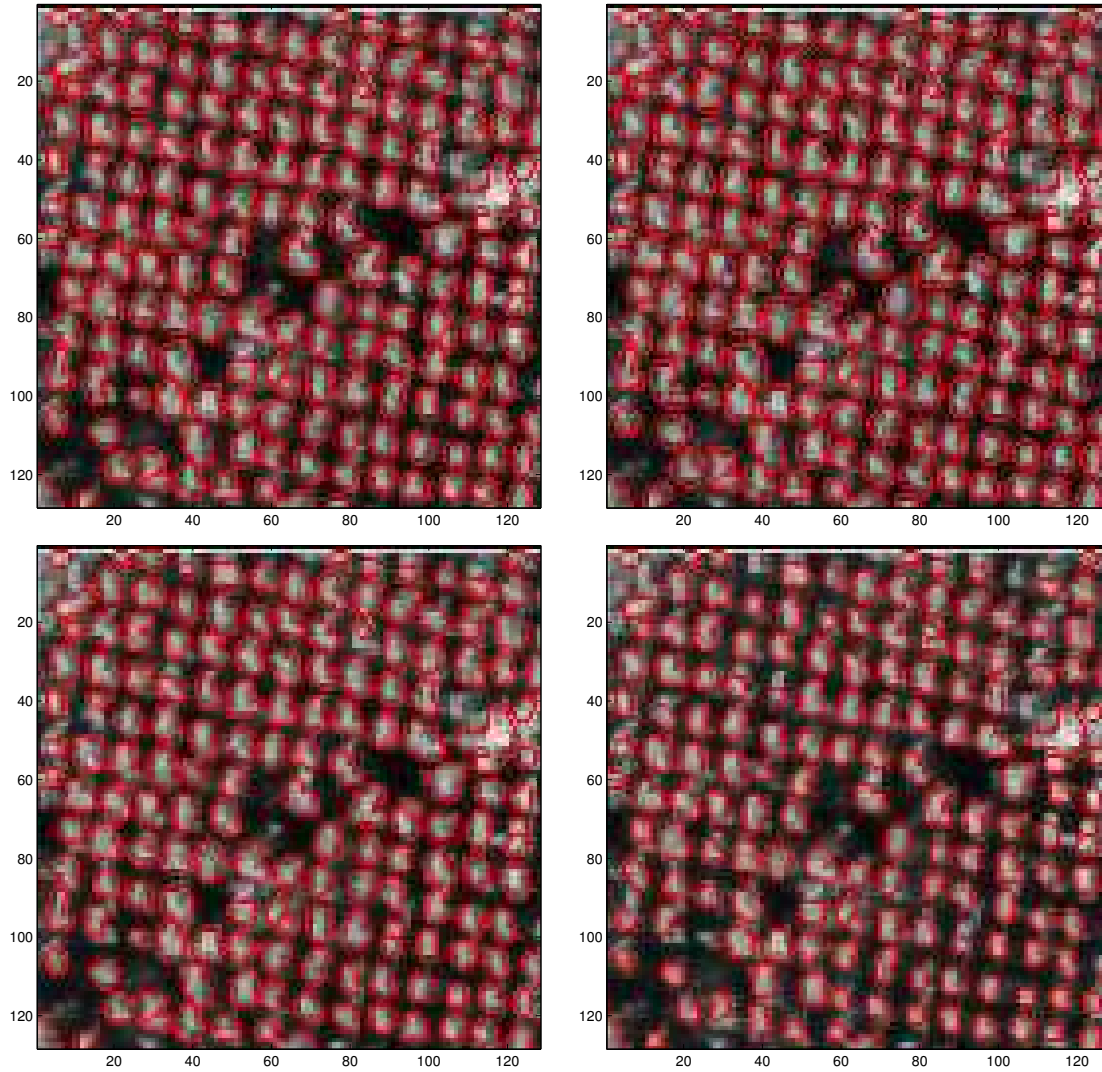


FIG. 4.12 – Résultats à partir du suivi de vallées. De haut en bas et de gauche à droite : $(s=4, \theta = 60)$, $(s=2, \theta = 50)$, $(s=2, \theta = 70)$ et $(s=2, \theta = 90)$. 4 secondes de traitement pour chacun.

4.4. Perspectives

Les résultats obtenus avec notre approche sur les peupleraies sont très encourageants. Bien sûr, notre objectif initial étant d'extraire plusieurs espèces d'arbres sur des images présentant des conditions d'illumination, de terrain et de prise de vue différentes n'est pas encore atteint. De nombreuses perspectives s'ouvrent à nous, et ce sont autant de voies qu'il faudra explorer au cours du travail de thèse. Parmi elles, citons :

– ***Fusion/Division :***

Le besoin de modéliser différemment le noyau de propositions lié à la fusion et à la division : nous pourrions en effet le simplifier, en ne laissant aucune liberté lors de la fusion, mais le coupler avec des mouvements simples tels qu'une translation et une dilatation. Ainsi, l'objet résultat de la fusion pourrait se positionner au mieux sur l'image.

– ***Transformée de Fourier :***

Plantés par l'homme, les peupleraies observent la plupart du temps un alignement et un espacement régulier dans deux directions. Nous pouvons le remarquer par exemple dans la Fig. (4.3), même si certains trous dans la zone sont présents. Aussi, une transformée de Fourier appliquée à l'image pourrait nous renseigner sur les fréquences fondamentales des positions des arbres.

La transformée de Fourier de l'image de taille 200×140 pixels, sur la Fig. (4.13), met bien en évidence les maxima spectraux. Ayant été recentrée, la fréquence nulle se situe au centre de l'image, celle-ci étant alors symétrique par rapport à son centre puisque la matrice de départ est réelle. On peut obtenir à partir de cette transformée les deux directions et distances principales d'alignement des peupliers. Cette approche est utilisée pour le comptage des oliviers dans [24].

Nous ne pouvons pas nous limiter au calcul de ces maxima pour détecter les arbres, puisque d'une part par endroit il y a des zones vides, et d'autre part les arbres ne respectent pas tous cet alignement régulier. Cependant, nous pouvons d'une part grâce à ces paramètres estimer le nombre d'arbres de la zone étudiée, ce qui nous permettrait de régler le paramètre d'intensité β , qui contrôle l'espérance du nombre d'objets du processus. D'autre part, nous pourrions nous servir de cette information pour proposer une naissance (et donc une mort associée) non uniforme des objets, proposant, à partir d'un objet de la configuration existant, de faire naître un nouvel objet à une distance et suivant une direction renseignés par ces maxima spectraux. Le schéma de la Fig. (4.14) donne un exemple de cette naissance non uniforme. Celle-ci accélérerait le processus des naissances à haute température, et le fait de limiter au maximum les naissances uniformes pour faire naître des objets de façon intelligente diminuerait le temps de traitement de l'algorithme.

– ***Estimation des paramètres***

Il serait aussi intéressant d'estimer les paramètres de notre modèle de densité, à savoir β et γ_a , au moyen de techniques d'estimation. Dans un cadre bayésien, cela consisterait à trouver les meilleurs paramètres, c'est à dire ceux permettant d'obtenir la meilleure densité a posteriori de la configuration solution étant donnée l'image. Ces méthodes, développées dans le projet Ariana, seront étudiées par la suite afin d'automatiser complètement notre processus d'extraction.

– ***Tests sur des images de synthèse***

Enfin, et nous rejoignons ici notre objectif de départ, il serait intéressant de tester nos algorithmes sur des images d'arbres fabriquées par nous-même. Cela nous permettrait en effet de

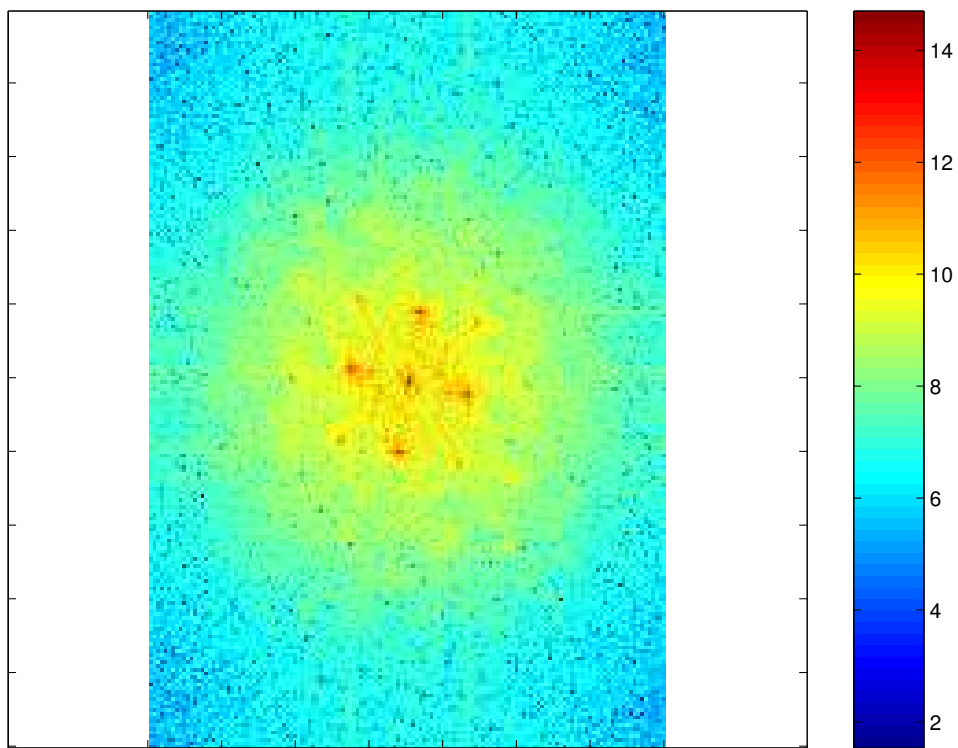


FIG. 4.13 – *Transformée de Fourier discrète de l'image de données, laissant apparaître les maxima spectraux*

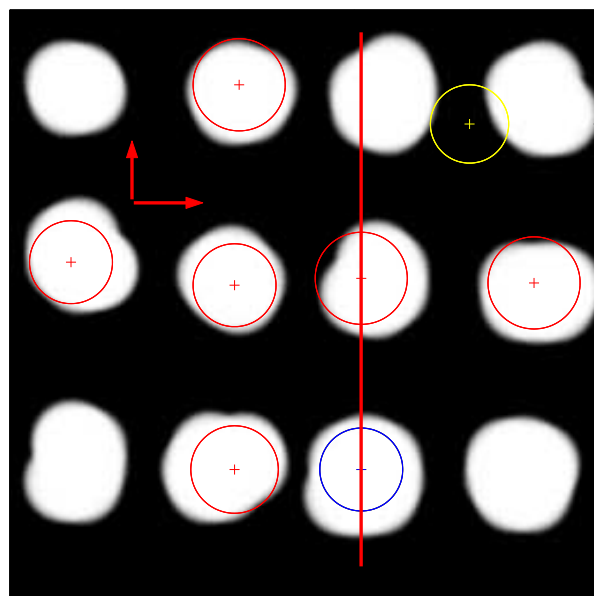


FIG. 4.14 – *Deux types de naissance. En rouge : les objets de la configuration à l'instant t . En jaune : proposition de naissance uniforme d'un objet. En bleu : proposition de naissance non uniforme d'un objet, utilisant les objets déjà existants ainsi que les directions privilégiées d'alignement.*

contrôler en amont le nombre d'arbres, les espèces présentes, la répartition, l'illumination et tous les autres paramètres qui font que l'image de sortie est différente, et en aval nous pourrions travailler sur notre modèle de densité et de vraisemblance, en ajoutant des paramètres de texture par exemple pour chacune des espèces afin de les distinguer.

Grâce à un partenariat entre le projet Ariana et le laboratoire MAS de l'Ecole Centrale Paris, nous aurons accès au cours du doctorat, au logiciel AMAP¹ développé au CIRAD et commercialisé par Bionatics. Il permet de concevoir totalement des paysages 3D, à l'aide d'une grande base de données de végétaux, et de choisir la prise de vue. Il nous suffirait de simuler la prise d'une image par un capteur afin d'obtenir des images semblables à celles que nous fournit l'IFN.

Ce travail sur des images synthétiques ne serait pas le but ultime, mais nous aiderait à affiner notre modèle afin d'obtenir de meilleurs résultats sur les images réelles.

1. botAnique et bioinforMatique de l'Architecture des Plantes

Conclusion

Des mathématiques à l'informatique

Ce stage fut l'occasion de travailler sur un sujet à la frontière entre les mathématiques et l'informatique. La théorie sur les processus objet est très riche, mais les domaines d'applications sont pour l'instant rares.

Face à la théorie, les dialogues avec les experts des milieux forestiers ont su nous confronter à la réalité. Ils ont fait état de l'importance pratique et économique que revêt le développement d'un processus semi-automatique de détection des houppiers.

L'apport de l'approche objet

Les processus objet se révèlent être très prometteurs pour la détection de houppiers, et l'obtention de statistiques sur les peuplements forestiers destinées aux organismes d'inventaire et de gestion nationaux. L'approche objet permet en effet d'exploiter les connaissances a priori des experts sur les différentes espèces, qui pourraient s'en servir pour résoudre de nombreux cas complexes de détection, auxquels les approches classiques ne peuvent s'attaquer.

Au cours de ce stage, nous avons été en mesure de développer un modèle géométrique évolutif, nous donnant déjà des résultats intéressants sur les peupleraies.

Développements à venir

Les voies d'exploration pour améliorer notre modèle sont très nombreuses, et seront étudiées au cours de la thèse qui succèdera au stage. Un des avantages principaux de notre méthode est sans aucun doute sa grande capacité de développement.

Le modèle géométrique en lui-même sera à retravailler, de même que l'attache aux données. Nous envisageons en effet une étude de la texture des arbres, afin d'affiner la détection et de permettre une classification suivant les différentes espèces. De plus, nous testerons de nouveaux estimateurs de la configuration d'objets solution, puisque certains semblent plus adaptés au traitement des images.

ANNEXES AU RAPPORT

- Annexe A :** Le projet Ariana, INRIA
- Annexe B :** Chaînes de Markov et Théorie ergodique
- Annexe C :** Recueil de démonstrations

Le projet Ariana, INRIA

★★★

Basé dans l'unité de recherche de Sophia Antipolis, Ariana est un projet commun au CNRS, à l'UNSA et à l'INRIA. Sa vocation est de mettre en commun des outils mathématiques théoriques, stochastiques et déterministes, et de les tester sur des *problèmes inverses* issus d'applications concrètes en *observation de la Terre et en cartographie*.

L'objectif de cette annexe est de présenter les dimensions techniques et humaines de mon stage au sein de ce projet. Seront développés les directions de recherche du projet, ainsi que la place que j'ai occupée dans l'équipe.

★★★

A.1. Les axes de recherche du projet Ariana

Les problèmes inverses en traitement d'images pour l'observation de la Terre sont nombreux et de nature variée. Ce sont des problèmes mal posés au sens de Hadamard. Il s'agit alors de régulariser la solution par l'introduction de contraintes a priori sur l'objet. Ces contraintes sont introduites sous deux formes : par **approche probabiliste** (par exemple en définissant un modèle de Markov sur les sites de l'image, l'estimation s'effectuant dans le cadre bayésien) ou par **approche déterministe** (en utilisant un modèle variationnel, l'estimation s'effectuant par minimisation de fonctionnelles).

Les 4 principaux axes de recherches du projet sont les suivants :

- La modélisation markovienne.
- L'approche variationnelle.
- La géométrie probabiliste.
- L'estimation des paramètres.

Le domaine de l'observation de la Terre et de la cartographie est très vaste. Il s'agit de pouvoir améliorer la qualité d'une image perturbée par du bruit ou floue afin de pouvoir **extraire une information pertinente** telle que l'ensemble d'un réseau routier (cf Fig. (A.1)), fluvial, ferroviaire, ou bien l'ensemble des classes d'une image pour permettre à un opérateur d'en faire une interprétation. Le domaine de l'observation de la Terre ne se réduit pas aux images 2D. Ainsi la création de modèles numériques de terrain ou d'élévation 3D est possible grâce à des techniques de stéréovision haute-résolution ou de super-résolution par imagerie optique ou radar.

Les relations internationales et industrielles du projet s'articulent autour de collaborations avec les universités de Cambridge (Royaume Uni), de Ben Gourion (Israël), de l'Illinois (USA), l'Université Nationale Autonome de Mexico (Mexique), le CWI¹ (Pays-Bas), la DLR² (Allemagne), ainsi que Astrium, Alcatel Space, Sagem, le CNES, l'IGN³, le BRGM⁴, le Cemagref et l'IFN.

-
1. Centrum voor Wiskunde en Informatica
 2. Deutschen Zentrum für Luft und Raumfahrt
 3. Institut Géographique National
 4. Bureau de Recherches Géologiques et Minières



FIG. A.1 – Extraction de réseau linéique routier, C. Lacoste [14]. A gauche, l'image de départ ©IGN. A droite, le résultat de l'extraction.

A.2. L'organisation interne du projet

L'équipe de recherche s'articule autour de **Josiane Zerubia**, responsable scientifique et directrice du projet, de trois chercheurs permanents que sont **Laure Blanc-Fraud** (CNRS), **Xavier Descombes** (INRIA) et **Ian Jermyn** (INRIA), et d'une assistante, **Corinne Zuzia**. Chacun des chercheurs permanents possède un axe de recherche qui lui est propre, et sont placés sous leur tutelle un certain nombre d'étudiants, stagiaires de DEA ou d'école d'ingénieur et doctorants. Viennent enfin participer à l'activité scientifique du projet des post-doctorants, des chercheurs invités, ainsi que des collaborateurs extérieurs.

La place du stagiaire est sensiblement la même que celle d'un étudiant en doctorat : on lui confère une grande autonomie dans l'organisation de son travail. Mes premiers mois de stage ont été consacrés à une vaste étude bibliographique, afin d'acquérir les bases mathématiques nécessaires, et à l'élaboration d'un premier algorithme de simulation sur des images synthétiques. Il me fallait, en effet, obtenir quelques résultats afin de les présenter à des jurys d'attribution de bourses de thèse. Puis, j'ai développé par la suite de nouveaux modèles d'extraction de houppiers, et consacré du temps à la rédaction de rapports pour le DEA et l'Ecole Centrale. Ceux-ci serviront de base pour un rapport de recherche INRIA, un article de revue, ainsi que la soumission d'un article dans une conférence internationale qui je l'espère seront prêts pour la fin de l'année 2003.

Des réunions sont régulièrement organisées au sein de chaque sous-équipe, composée de la directrice du projet, d'un chercheur permanent et des étudiants travaillant autour d'un même axe de recherche. La vie d'un chercheur est enfin rythmée par les missions, colloques, et conférences auxquels il participe. Au cours de ce stage, je me suis rendu à Montpellier afin de rencontrer nos collaborateurs de l'IFN et du Cemagref, et j'ai pu assister à de nombreux séminaires organisés à l'INRIA, qui propose une formation scientifique très vaste aux chercheurs.

Chaînes de Markov et Théorie ergodique

★★★

Comme nous l'avons remarqué dans le corps du rapport, les processus objets sont généralement définis par une densité a priori non normalisée sur l'espace probabilisé $(\Psi, \mathcal{B}(\Psi), \mu(\cdot))$. Autrement dit, ils sont définis par une densité par rapport à un processus de Poisson de référence, dont le calcul de la constante de normalisation $\int_{\Psi} h(\mathbf{x}) \mu(d\mathbf{x})$ n'est pas envisageable en pratique.

C'est pourquoi des méthodes pour obtenir des réalisations de tels processus sont utiles. Parmi elles, on distingue les algorithmes de type Monte Carlo, où l'on simule une chaîne de Markov convergeant vers la distribution recherchée.

Cette annexe a donc pour but de nous rappeler quelques notions autour des chaînes de Markov et sur la théorie ergodique. Le lecteur intéressé par la théorie des chaînes de Markov et les algorithmes Markov Chain Monte Carlo (ou MCMC) pourra se reporter à [1], [23] et [27].

Soit $\pi(\cdot)$ une loi sur l'espace probabilisé $(\Psi, \mathcal{B}(\Psi), \mu(\cdot))$, que nous cherchons à simuler. Dans l'étude ultérieure des algorithmes, on supposera que cette loi possède une densité $h(\cdot)$ par rapport à la loi de référence $\mu(\cdot)$. Sous certaines conditions une chaîne de Markov, dans l'espace probabilisé $(\Psi, \mathcal{B}(\Psi), \mu(\cdot))$, converge vers une suite d'échantillons suivant $\pi(\cdot)$.

★★★

B.1. Notions sur les chaînes de Markov

Nous recherchons des conditions suffisantes pour que la chaîne de Markov converge vers une distribution $\pi(\cdot)$ que l'on souhaite simuler, quel que soit l'état initial.

B.1.1 Premières définitions

Définition B.1

Une suite aléatoire de configurations (\mathbf{x}_i) forme une **chaîne de Markov** si la loi conditionnelle de \mathbf{x}_j sachant $\mathbf{x}_{j-1}, \mathbf{x}_{j-2}, \dots$ est la même que celle de \mathbf{x}_j sachant \mathbf{x}_{j-1} .

On dit de plus que la loi est **homogène** si la loi de $(\mathbf{x}_{j_1}, \dots, \mathbf{x}_{j_k})$ sachant \mathbf{x}_{j_0} est la même que celle de $(\mathbf{x}_{j_1-j_0}, \dots, \mathbf{x}_{j_k-j_0})$ sachant \mathbf{x}_0 pour tout k et tout $(k+1)$ -uplets $j_0 \leq j_1 \leq \dots \leq j_k$.

Une chaîne de Markov est caractérisée par :

1. Sa loi initiale, c'est à dire la loi selon laquelle est tiré le premier échantillon \mathbf{x}_0 .
2. Son noyau de transition, qui caractérise comment la chaîne passe d'un échantillon au suivant.

Définition B.2

On appelle **noyau de transition** toute fonction $K(\cdot, \cdot)$ définie sur $\Psi \times \mathcal{B}(\Psi)$ telle que :

1. $\forall \mathbf{x} \in \Psi, K(\mathbf{x}, \cdot)$ est une mesure de probabilité.
2. $\forall B \in \mathcal{B}(\Psi), K(\cdot, B)$ est mesurable.

Ce noyau nous permet de connaître le comportement de la chaîne puisque $\forall (k, B) \in (\mathbb{N} \times \mathcal{B}(\Psi))$,

$$\begin{aligned} \mathbb{P}(\mathbf{x}_{k+1} \in B | \mathbf{x}_0, \dots, \mathbf{x}_k) &= \mathbb{P}(\mathbf{x}_{k+1} \in B | \mathbf{x}_k) \\ &= \int_B K(\mathbf{x}_k, d\mathbf{x}). \end{aligned}$$

Notations :

Nous emploierons la notation $P_{\mathbf{x}}(\cdot)$ pour signifier $P(\cdot | \mathbf{x}_0 = \mathbf{x})$.

De même, en posant $K^1(\mathbf{x}, B) = K(\mathbf{x}, B)$, nous noterons $K^n(\mathbf{x}, B)$ le noyau de n transitions défini par récurrence par la formule suivante :

$$K^n(\mathbf{x}, B) = \int_{\Psi} K^{n-1}(\mathbf{y}, B) K(\mathbf{x}, d\mathbf{y}). \quad (\text{B.1})$$

B.1.2 Stationnarité et Invariance

La stationnarité ou invariance est une propriété de base que l'on désire pour une chaîne de Markov.

Définition B.3

Une mesure $\pi(\cdot)$ est dite **invariante** pour le noyau de transition $K(\cdot, \cdot)$ d'une chaîne de Markov si $\pi K = \pi$, à savoir :

$$\begin{aligned} (\pi K)(B) &= \int_{\Psi} K(\mathbf{x}, B) \pi(d\mathbf{x}) \\ &= \pi(B), \forall B \in \mathcal{B}(\Psi). \end{aligned}$$

Si en outre, la mesure $\pi(\cdot)$ est une mesure de probabilité, alors elle est appelée **mesure stationnaire** de la chaîne.

On dit que la chaîne est **π -invariante** ou **π -stationnaire**.

Remarque :

Si une mesure $\pi(\cdot)$ est stationnaire pour plusieurs noyaux de transitions $K_1(\cdot, \cdot), \dots, K_n(\cdot, \cdot)$, alors $\pi(\cdot)$ est aussi une mesure stationnaire pour le noyau de transition $K(\cdot, \cdot) = \sum_{i=1}^n p_i K_i(\cdot, \cdot)$, avec les $p_i > 0$ et $\sum_{i=1}^n p_i = 1$.

En effet,

$$\begin{aligned} (\pi K)(B) &= \int_{\Psi} K(\mathbf{x}, B) \pi(d\mathbf{x}) \\ &= \sum_{i=1}^n p_i \int_{\Psi} K_i(\mathbf{x}, B) \pi(d\mathbf{x}) \\ &= \sum_{i=1}^n p_i \pi(B) \\ &= \pi(B), \forall B \in \mathcal{B}(\Psi). \end{aligned}$$

Ceci nous sera très utile par la suite, lorsque notre noyau de transition sera défini par plusieurs noyaux de transitions choisis avec une probabilité p_i à chaque saut de la chaîne. Cette combinaison convexe de noyaux de proposition est en effet un des piliers de l'algorithme de simulation de processus ponctuels que nous implanterons.

B.1.3 Réversibilité

Définition B.4

La chaîne est dite **réversible** pour la loi objectif $\pi(\cdot)$ si son noyau de transition vérifie :

$$\int_B K(\mathbf{x}, A) \pi(d\mathbf{x}) = \int_A K(\mathbf{x}, B) \pi(d\mathbf{x}), \forall A, B \in \mathcal{B}(\Psi). \quad (\text{B.2})$$

Proposition B.5 :

|| La réversibilité implique la stationnarité de la loi $\pi(\cdot)$

◇ **Démonstration :**

En remplaçant dans l'équation (B.2) le borélien B par Ψ , on prouve que la π -réversibilité implique la stationnarité de la loi $\pi(\cdot)$ puisque ce cas particulier mène à :

$$\begin{aligned} \int_{\Psi} K(\mathbf{x}, A) \pi(d\mathbf{x}) &= \int_A K(\mathbf{x}, \Psi) \pi(d\mathbf{x}) \\ &= \pi(A), \forall A \in \mathcal{B}(\Psi). \end{aligned}$$

◇

Réversibilité \implies Stationnarité

B.1.4 Irréductibilité

Une première étude de la robustesse d'une chaîne de Markov aux conditions initiales, est le caractère irréductible de la chaîne. Dans le cadre des algorithmes de Monte Carlo, elle permet d'assurer la convergence vers la distribution objectif sans pour autant nécessiter une étude préalable poussée du noyau de transition nécessaire à la détermination de conditions initiales correctes.

La ϕ -irréductibilité traduit la capacité de la chaîne à passer par tous les ensembles de ϕ -mesure non nulle.

Définition B.6

Etant donnée une mesure $\phi(\cdot)$ sur $\mathcal{B}(\Psi)$, la chaîne (\mathbf{x}_i) est dite **ϕ -irréductible** si $\forall B \in \mathcal{B}(\Psi)$ tel que $\phi(B) > 0$:

$$\exists n : K^n(\mathbf{x}, B) > 0, \forall \mathbf{x} \in \Psi.$$

S'il existe une mesure $\phi(\cdot)$ telle que la chaîne soit ϕ -irréductible, alors elle est aussi π -irréductible pour une mesure stationnaire $\pi(\cdot)$, qui dans ce cas est la seule mesure stationnaire pour la chaîne.

Définition B.7

Si la chaîne de Markov est ϕ -irréductible et qu'il existe une mesure stationnaire, on dit qu'elle est **positive**.

Irréductibilité + Stationnarité \implies Positivité

B.1.5 Ensembles petits et Apériodicité

Définition B.8

On appelle **ensemble petit** tout ensemble $C \in \mathcal{B}(\Psi)$ tel qu'il existe un entier n et une mesure non nulle $\nu(\cdot)$ vérifiant :

$$K^n(\mathbf{x}, B) \geq \nu(B), \forall \mathbf{x} \in C, \forall B \in \mathcal{B}(\Psi).$$

Nous verrons par la suite que l'on attend d'une chaîne de Markov que tous les ensembles bornés mesurables de l'espace des configurations soient petits.

Nous pouvons alors définir la notion de période d'une chaîne de Markov sur un espace d'états continu. Celle-ci diffère de la définition donnée dans le cadre d'une chaîne de Markov simulée sur un espace d'états discret, pour laquelle la période est le plus grand commun diviseur (pgcd) des t tels que pour tout \mathbf{x} , $K^t(\mathbf{x}, \mathbf{x}) > 0$.

Définition B.9

Une chaîne (\mathbf{x}_i) ϕ -irréductible admet un **cycle de longueur** d s'il existe un ensemble petit C , d'entier et de loi de probabilité associés M et $\nu_M(\cdot)$, tel que d soit le p.g.c.d. de

$$\{m \geq 1; \exists \delta_m > 0 \mid C \text{ petit pour } m \text{ et } \nu_m \geq \delta_m \nu_M\}.$$

Une chaîne est dite **apériodique** si elle est irréductible de période 1.

Une condition suffisante pour assurer l'apériodicité est que l'on autorise la chaîne de rester dans le même état :

$$K(\mathbf{x}, \mathbf{x}) > 0, \forall \mathbf{x} \in \Psi \tag{B.3}$$

B.2. Etude de la stabilité des chaînes de Markov

Il s'agit désormais de faire converger cette chaîne vers la distribution objectif $\pi(\cdot)$. Il existe plusieurs critères de convergence.

Le critère de convergence le plus faible est la convergence en loi, aussi appelé ergodicité.

B.2.1 Ergodicité

Définition B.10

Une chaîne de Markov de noyau de transition $K(\cdot, \cdot)$ **converge en loi** vers la distribution objectif $\pi(\cdot)$ si pour presque tout $\mathbf{x} \in \Psi$:

$$\lim_{n \rightarrow \infty} |K^n(\mathbf{x}, \cdot) - \pi(\cdot)| = 0. \quad (\text{B.4})$$

On dit dans ce cas qu'elle est **ergodique**.

Proposition B.11 : Ergodicité

|| Une chaîne de Markov irréductible, apériodique, et ayant une distribution invariante $\pi(\cdot)$, est ergodique.

Positivité + Apériodicité \implies Ergodicité

B.2.2 Récurrence au sens de Harris

Un autre type de convergence, plus fort, est la convergence au sens de la norme en variations totales défini pour deux mesures $\mu_1(\cdot)$ et $\mu_2(\cdot)$ par :

$$\|\mu_1(\cdot) - \mu_2(\cdot)\|_{VT} = \max(|\mu_1(B) - \mu_2(B)|, B \in \mathcal{B}(\Psi)). \quad (\text{B.5})$$

Cette convergence permet de supprimer les problèmes de non-convergence de la chaîne pour certaines configurations initiales, mais requiert, en plus de la positivité et de l'apériodicité de la chaîne, la récurrence au sens de Harris.

Définition B.12

Une chaîne (\mathbf{x}_i) est **récurrence au sens de Harris** s'il existe une mesure $\phi(\cdot)$ telle qu'elle soit ϕ -irréductible, et si tout ensemble A , tel que $\phi(A) > 0$, est **récurent au sens de Harris**, c'est-à-dire vérifie

$$P_{\mathbf{x}} \left(\sum_{i=1}^{\infty} \mathbf{1}_A(\mathbf{x}_i) = \infty \right) = 1, \forall \mathbf{x} \in A. \quad (\text{B.6})$$

Ceci équivaut à $P_{\mathbf{x}}(\exists t : \mathbf{x}_t \in A) = 1$.

Une chaîne est **positive au sens de Harris** lorsqu'elle est récurrente au sens de Harris et qu'il existe une mesure de probabilité invariante pour cette chaîne.

Il s'agit d'une notion forte, puisqu'elle impose un retour certain en un temps fini dans tout ensemble de mesure positive de l'espace.

Suite à la définition de cette récurrence, nous pourrions énoncer un théorème qui assure la convergence en variations totales de la chaîne.

Théorème B.13 : Convergence en variations totales

$$\left\| \begin{array}{l} \text{Si } (\mathbf{x}_i) \text{ est positive au sens de Harris et apériodique,} \\ \lim_{n \rightarrow \infty} \left\| \int K^n(\mathbf{x}, \cdot) \nu(d\mathbf{x}) - \pi(\cdot) \right\|_{VT} = 0 \\ \text{pour toute distribution initiale } \nu(\cdot). \end{array} \right. \quad (\text{B.7})$$

Positivité au sens de Harris + Apériodicité \implies Convergence en variation totale

B.2.3 Théorème limite

Théorème B.14 : Loi des grands nombres

$$\left\| \begin{array}{l} \text{Soit } (\mathbf{x}_i) \text{ une chaîne positive au sens de Harris, de loi invariante intégrable } \pi(\cdot). \\ \text{Alors pour tout couple de fonctions } f \text{ et } g \text{ strictement positives dans } \mathcal{L}^1(\Psi), \\ \lim_{n \rightarrow +\infty} \frac{\sum_{i=1}^n f(\mathbf{x}_i)}{\sum_{i=1}^n g(\mathbf{x}_i)} = \frac{\int f(x) d\pi(x)}{\int g(x) d\pi(x)}. \end{array} \right. \quad (\text{B.8})$$

En particulier pour $g \equiv 1$:

$$\left\| \begin{array}{l} \lim_{n \rightarrow +\infty} \frac{\sum_{i=1}^n f(\mathbf{x}_i)}{n} = \int f(x) d\pi(x). \end{array} \right. \quad (\text{B.9})$$

Ce théorème pourra nous servir lorsque nous souhaiterons calculer des statistiques sur nos échantillons.

Recueil de démonstrations

★★★

Au cours des différents chapitres du mémoire, certaines démonstrations ont été volontairement omises afin de donner plus de poids au résultat, et de permettre une meilleure compréhension de la démarche de notre raisonnement.

Celles-ci sont toutes détaillées dans cette annexe, classées par ordre d'apparition dans le rapport.

★★★

C.1. Réversibilité de la chaîne dans l'algorithme de Green

Proposition C.1 : Rapport de Green

Soient $Q_i(\mathbf{x}, A)$, $i \in I$ et $A \in \mathcal{B}(\Psi)$, les différents noyaux de propositions vérifiant $\sum_{i \in I} Q_i(\mathbf{x}, \Psi) \leq 1$ pour tout $\mathbf{x} \in \Psi$. Alors l'algorithme proposé par Green simule une chaîne de Markov réversible.

◇ Démonstration :

Montrons en effet que (2.20) est vérifiée. Tout d'abord, notons que puisque $\sum_{i \in I} Q_i(\mathbf{x}, \Psi) \leq 1$, aucun mouvement n'est proposé avec la probabilité $r(\mathbf{x}) = 1 - \sum_{i \in I} Q_i(\mathbf{x}, \Psi)$.

Nous définissons alors $\alpha_i(\mathbf{x}, \mathbf{x}')$ comme la probabilité d'accepter un mouvement de \mathbf{x} à \mathbf{x}' avec le noyau de propositions Q_i , et

$$\rho_i(\mathbf{x}) = \int_{\Psi} Q_i(\mathbf{x}, d\mathbf{x}') [1 - \alpha_i(\mathbf{x}, \mathbf{x}')]$$

comme la probabilité de rester en \mathbf{x} malgré les propositions du noyau Q_i .

Le noyau de transition de la chaîne $K(\cdot, \cdot)$ peut alors s'écrire sous la forme suivante, $\forall \mathbf{x} \in \Psi$ et $\forall B \in \mathcal{B}(\Psi)$:

$$K(\mathbf{x}, B) = \rho(\mathbf{x}) \mathbf{1}_{\mathbf{x} \in B} + \sum_{i \in I} \int_B Q_i(\mathbf{x}, d\mathbf{x}') \alpha_i(\mathbf{x}, \mathbf{x}') \quad (\text{C.1})$$

avec $\rho(\mathbf{x})$ la probabilité de rester en \mathbf{x} , soit suite à un mouvement proposé par un des noyaux Q_i qui n'a pas été accepté, soit puisque aucun mouvement n'a été proposé :

$$\rho(\mathbf{x}) = r(\mathbf{x}) + \sum_{i \in I} \rho_i(\mathbf{x}).$$

En remplaçant (C.1) dans (2.20), on obtient l'équation suivante qui doit être vérifiée pour que notre noyau de propositions convienne :

$$\sum_{i \in I} \int_B \pi(d\mathbf{x}) \int_A Q_i(\mathbf{x}, d\mathbf{x}') \alpha_i(\mathbf{x}, \mathbf{x}') = \sum_{i \in I} \int_A \pi(d\mathbf{x}') \int_B Q_i(\mathbf{x}', d\mathbf{x}) \alpha_i(\mathbf{x}', \mathbf{x}). \quad (\text{C.2})$$

Pour que (C.2) tienne, il suffit que pour tout $i \in I$, $A, B \in \mathcal{B}(\Psi)$

$$\int_B \pi(d\mathbf{x}) \int_A Q_i(\mathbf{x}, d\mathbf{x}') \alpha_i(\mathbf{x}, \mathbf{x}') = \int_A \pi(d\mathbf{x}') \int_B Q_i(\mathbf{x}', d\mathbf{x}) \alpha_i(\mathbf{x}', \mathbf{x}).$$

En supposant que $\pi(d\mathbf{x}) Q_i(\mathbf{x}, d\mathbf{x}')$ soit dominé par une mesure symétrique ξ_i sur $\Psi \times \Psi$, et qu'il existe une dérivée au sens de Radon-Nikodym

$$f_i(\mathbf{x}, \mathbf{x}') = \frac{\pi(d\mathbf{x}) Q_i(\mathbf{x}, d\mathbf{x}')}{\xi_i(d\mathbf{x}, d\mathbf{x}')}$$

on a :

$$\int_B \pi(d\mathbf{x}) \int_A Q_i(\mathbf{x}, d\mathbf{x}') \alpha_i(\mathbf{x}, \mathbf{x}') = \int_B \int_A \xi_i(d\mathbf{x}, d\mathbf{x}') f_i(\mathbf{x}, \mathbf{x}') \alpha_i(\mathbf{x}, \mathbf{x}') \quad (\text{C.3})$$

et

$$\int_A \pi(d\mathbf{x}') \int_B Q_i(\mathbf{x}', d\mathbf{x}) \alpha_i(\mathbf{x}', \mathbf{x}) = \int_A \int_B \xi_i(d\mathbf{x}', d\mathbf{x}) f_i(\mathbf{x}', \mathbf{x}) \alpha_i(\mathbf{x}', \mathbf{x}). \quad (\text{C.4})$$

Finalement, (C.3) et (C.4) sont égales d'après la symétrie de la mesure ξ_i , et si en outre

$$f_i(\mathbf{x}, \mathbf{x}') \alpha_i(\mathbf{x}, \mathbf{x}') = f_i(\mathbf{x}', \mathbf{x}) \alpha_i(\mathbf{x}', \mathbf{x}).$$

Ceci est vérifié si on pose

$$\alpha_i(\mathbf{x}, \mathbf{x}') = \min \left(1, \frac{f_i(\mathbf{x}', \mathbf{x})}{f_i(\mathbf{x}, \mathbf{x}')} \right).$$

Avec ce choix de coefficient d'acceptation du mouvement, la chaîne est réversible et la loi $\pi(\cdot)$ est donc stationnaire.

◇

C.2. Irréductibilité et Ensembles petits

Proposition C.2 : Irréductibilité, Ensembles petits

|| Si la densité non normalisée $h(\cdot)$ vérifie la condition de stabilité (2.11), alors l'algorithme de Green simule une chaîne de Markov irréductible et tout ensemble borné de Ψ est petit.

◇ Démonstration :

Montrons tout d'abord que la chaîne de Markov est irréductible. Notons $\mu_0(\cdot)$ la mesure sur $\Psi_0 = \{\emptyset\}$ qui donne une masse un au point $\{\emptyset\}$.

Nous pouvons minorer, grâce à (2.11), la probabilité d'accepter une mort :

$$\min \left(1, \frac{p_n}{p_m} \frac{h(\mathbf{x} \setminus \{u\})}{h(\mathbf{x})} \frac{n(\mathbf{x})}{\lambda(U)} \right) \geq \frac{p_n}{p_m} \frac{1}{M_h \lambda(U)}$$

où M_h est la constante introduite dans (2.11).

La probabilité de proposer une mort est p_m , et donc en partant de la configuration nulle, la probabilité de rester dans Ψ_0 est $K(\{\emptyset\}, \Psi_0) \geq p_m$.

Pour toute configuration $\mathbf{x} \in \Psi$, $n(\mathbf{x})$ représente toujours son nombre d'objets. En choisissant un entier $m \geq n(\mathbf{x})$ on a :

$$\begin{aligned} K^m(\mathbf{x}, \Psi_0) &\geq K^{n(\mathbf{x})}(\mathbf{x}, \Psi_0) K^{m-n(\mathbf{x})}(\{\emptyset\}, \Psi_0) \\ &\geq \left(\frac{p_n}{p_m} \frac{1}{M_h \lambda(U)} \right)^{n(\mathbf{x})} (p_m)^{m-n(\mathbf{x})} \\ &\geq \left(\frac{p_n}{p_m} \frac{1}{M_h \lambda(U)} \right)^m (p_m)^m \end{aligned}$$

si l'on choisit M_h assez grand pour que $\frac{p_n}{p_m} \frac{1}{M_h \lambda(U)} \leq 1$. Donc on obtient

$$K^m(\mathbf{x}, \Psi_0) \geq \left(\frac{p_n}{M_h \lambda(U)} \right)^m. \quad (\text{C.5})$$

(C.5) démontre bien que la chaîne de Markov est μ_0 -irréductible. En effet, le seul ensemble μ_0 -positif est $\Psi_0 = \{\emptyset\}$, et dès que $m \geq n(\mathbf{x})$, $K^m(\mathbf{x}, \Psi_0) > 0$.

Ensuite, notons $\phi(\cdot)$ une mesure que nous choisissons proportionnelle à $\mu_0(\cdot)$: $\phi = a\mu_0$, avec $a = \left(\frac{p_n}{M_h \lambda(U)} \right)^m$. Soit $\mathcal{C}_m = \{\mathbf{x} \in \Psi : n(\mathbf{x}) \leq m\}$. \mathcal{C}_m est petit puisque

$$K^m(\mathbf{x}, \Psi_0) \geq a\mu_0(\Psi_0) = a, \forall \mathbf{x} \in \mathcal{C}_m.$$

Ceci démontre la deuxième partie de la proposition.

◇

C.3. Ergodicité géométrique de l'algorithme de Green

Proposition C.3 : Récurrence de Harris, Ergodicité géométrique

|| Si la densité non normalisée $h(\cdot)$ vérifie la condition de stabilité (2.11), alors l'algorithme de Green simule une chaîne de Markov récurrente au sens de Harris et ergodique géométriquement.

◇ **Démonstration :**

Nous pouvons tout d'abord, grâce à (2.11), minorer la probabilité d'accepter une mort et majorer celle d'accepter une naissance :

$$\begin{aligned} \alpha(\mathbf{x}, \mathbf{x} \setminus \{u\}) &= \min \left(1, \frac{p_n}{p_m} \frac{h(\mathbf{x} \setminus \{u\})}{h(\mathbf{x})} \frac{n(\mathbf{x})}{\lambda(U)} \right) \\ &\geq \min \left(1, \frac{p_n}{p_m} \frac{n(\mathbf{x})}{M_h \lambda(U)} \right) \end{aligned}$$

et

$$\begin{aligned} \alpha(\mathbf{x}, \mathbf{x} \cup \{u\}) &= \min \left(1, \frac{p_m}{p_n} \frac{h(\mathbf{x} \cup \{u\})}{h(\mathbf{x})} \frac{\lambda(U)}{n(\mathbf{x}) + 1} \right) \\ &\leq \frac{p_m}{p_n} \frac{M_h \lambda(U)}{n(\mathbf{x}) + 1}. \end{aligned}$$

Pour tout $\epsilon \in (0, 1)$, dès que $n(\mathbf{x}) \geq N_\epsilon = \frac{p_m}{p_n} \frac{M_h \lambda(U)}{\epsilon}$, on a :

$$\alpha(\mathbf{x}, \mathbf{x} \setminus \{u\}) = 1$$

et

$$\alpha(\mathbf{x}, \mathbf{x} \cup \{u\}) \leq \epsilon.$$

Dans ce cas, en posant $V(\mathbf{x}) = A^{n(\mathbf{x})}$, avec $A > 1$:

$$\begin{aligned} KV(\mathbf{x}) &= \left(\frac{p_m \alpha(\mathbf{x}, \mathbf{x} \setminus \{u\})}{A} + p_n \alpha(\mathbf{x}, \mathbf{x} \cup \{u\}) A + p_m (1 - \alpha(\mathbf{x}, \mathbf{x} \setminus \{u\})) + p_n (1 - \alpha(\mathbf{x}, \mathbf{x} \cup \{u\})) \right) V(\mathbf{x}) \\ &\leq \left(\frac{p_m}{A} + p_n \epsilon A + p_n (1 - \epsilon) \right) V(\mathbf{x}) \end{aligned}$$

On choisit alors ϵ assez petit pour que le terme entre parenthèses soit inférieur à 1, ce qui est possible puisque quand $\epsilon \rightarrow 0$, ce terme tend vers $\frac{p_m}{A} + p_n < 1$. Avec un tel ϵ , on trouve facilement une constante $a < 1$ telle que

$$KV(\mathbf{x}) \leq aV(\mathbf{x}),$$

dès que $n(\mathbf{x}) \geq N_\epsilon$.

Il suffit alors de choisir comme ensemble petit l'ensemble borné $\mathcal{C} = \{\mathbf{x} \in \Psi : n(\mathbf{x}) < N_\epsilon\}$ (qui est petit d'après la proposition précédente), et comme constante b un majorant de $KV(\mathbf{x})$ sur cet ensemble \mathcal{C} , à savoir $b = A^{N_\epsilon+1}$.

◇

C.4. Propriété sur $n_{\sim_I}(\mathbf{x})$

Lemme C.4 : Propriété sur $n_{\sim_I}(\mathbf{x})$

$$\left\| \begin{array}{l} \text{Il existe un entier } N_{dec} \text{ tel que pour toute configuration de points de } U = \mathcal{P} \times \mathcal{K}, \\ \text{notée } \mathbf{x}, \text{ on ait} \\ \\ n(\mathbf{x}) \geq N_{dec} \implies n_{\sim_I}(\mathbf{x}) \geq \frac{n(\mathbf{x})(n(\mathbf{x}) - N_{dec})}{2 \cdot N_{dec}}. \quad (\text{C.6}) \\ \\ \text{Autrement dit, le rapport } \frac{n(\mathbf{x})}{n_{\sim_I}(\mathbf{x})} \text{ tend vers } 0 \text{ quand } n(\mathbf{x}) \rightarrow \infty. \end{array} \right.$$

◇ **Démonstration :**

Toute la démonstration repose sur notre capacité à découper l'espace des objets $U = \mathcal{P} \times \mathcal{K}$ en un nombre N_{dec} de parties U_i , telles que deux points quelconques appartenant à un même U_i soient voisins au sens de \sim_I .

Etape 1 :

Supposons une telle partition $U = \bigcup_{i=1}^{N_{dec}} U_i$ réalisée. Soit \mathbf{x} une configuration, plaçant n_i (pouvant être nul) de ses points dans chacun des sous-espaces U_i . On a alors :

$$n(\mathbf{x}) = \sum_{i=1}^{N_{dec}} n_i \text{ et } n_{\sim_I}(\mathbf{x}) \geq \sum_{i=1}^{N_{dec}} C_{n_i}^2 = n_{\sim_I}(n_1, \dots, n_{N_{dec}})$$

où par convention C_0^2 et C_1^2 sont nuls. Le minimum de la fonctionnelle $n_{\sim_I}(n_1, \dots, n_{N_{dec}})$ définie sur $\mathbb{R}^{N_{dec}}$, sous la contrainte $\sum_{i=1}^{N_{dec}} n_i = n(\mathbf{x})$, est atteint pour $n_i = \frac{n(\mathbf{x})}{N_{dec}}$ pour tout i . Ceci mène à :

$$n_{\sim_I}(n_1, \dots, n_{N_{dec}}) \geq \frac{n(\mathbf{x})(n(\mathbf{x}) - N_{dec})}{2 \cdot N_{dec}}.$$

Ainsi donc, pour $n(\mathbf{x}) \geq N_{dec}$, le nombre de cliques d'objets de la configuration \mathbf{x} voisins au sens de \sim_I est tel que

$$n_{\sim_I}(\mathbf{x}) \geq \frac{n(\mathbf{x})(n(\mathbf{x}) - N_{dec})}{2 \cdot N_{dec}}.$$

Ceci achève la première partie de notre démonstration. Il nous faut désormais expliciter une telle partition de notre espace des objets U .

Etape 2 :

On rappelle que $U = \mathcal{P} \times \mathcal{K} = [0, X_M] \times [0, Y_M] \times [R_m, R_M]$. D'après les valeurs extrémales choisies pour l'espace des marques \mathcal{K} , deux objets x_1 et x_2 vérifiant $d(C_{x_1}, C_{x_2}) < 2 \cdot R_m = d_{min}$ sont voisins au sens de \sim_I puisqu'alors $d(C_{x_1}, C_{x_2}) < R_{x_1} + R_{x_2}$.

Il suffit alors de recouvrir U par des pavés de taille $d_{pave} < \frac{d_{min}}{\sqrt{2}}$ sur chacune des deux dimensions de l'espace des positions. Une telle partition peut par exemple s'écrire :

$$\bigcup_{p,q} U_{p,q} = \bigcup_{p,q} ([p \cdot d_{pave}, (p+1) \cdot d_{pave}] \times [q \cdot d_{pave}, (q+1) \cdot d_{pave}]) \cap \mathcal{P} \times \mathcal{K}$$

Dans ce cas, l'entier N_{dec} associé à ce découpage serait

$$N_{dec} = \mathcal{E}_{sup} \left(\frac{X_M}{d_{pave}} \right) \mathcal{E}_{sup} \left(\frac{Y_M}{d_{pave}} \right),$$

où $\mathcal{E}_{sup}(x)$ est le plus petit entier supérieur ou égal à x .

◇

C.5. Ergodicité géométrique de l'algorithme A

Proposition C.5 : Récurrence de Harris, Ergodicité géométrique

Si la densité non normalisée $h(\cdot)$ vérifie la condition de stabilité (2.11), ainsi que les conditions c_1 , c_2 et c_3 explicitées précédemment, alors **l'algorithme A** simule une chaîne de Markov récurrente au sens de Harris et ergodique géométriquement.

◇ **Démonstration :**

Etape 1 :

Nous pouvons tout d'abord, grâce à (2.11), minorer la probabilité d'accepter une mort et majorer celle d'accepter une naissance dans le noyau de propositions de naissance et de mort.

Pour tout $\epsilon \in (0,1)$, dès que $n(\mathbf{x}) \geq N_{nm,\epsilon} = \frac{p_m M_h \lambda(U)}{p_n \epsilon}$, on a :

$$\alpha_m(\mathbf{x}, \mathbf{x} \setminus \{u\}) = 1 \text{ et } \alpha_n(\mathbf{x}, \mathbf{x} \cup \{u\}) \leq \epsilon.$$

Etape 2 :

Nous devons faire de même avec le noyau de propositions de fusion et de division. On a :

$$\begin{aligned} \alpha_d(\mathbf{x}, \mathbf{x} \setminus \{x_i\} \cup \{u_1 \cup u_2\}) &= \min \left(1, \frac{p_f h(\mathbf{y})}{p_d h(\mathbf{x})} \frac{n(\mathbf{x})}{n_{\sim_I}(\mathbf{y})} \frac{f_f(\eta_{u_1, u_2}^{-1}(x_i))}{f_d(\zeta_{x_i}^{-1}(u_1, u_2))} d(x_i, x_j) \right) \\ &\leq \frac{p_f M_h^2}{p_d m_h} \frac{n(\mathbf{x})}{n_{\sim_I}(\mathbf{y})} \frac{M_{f_f}}{m_{f_d}} d_{max} \end{aligned}$$

d'après les conditions c_1 , c_2 et c_3 .

De plus, le lemme énoncé précédemment nous donne pour tout $n(\mathbf{x}) \geq N_{dec} - 1$, et donc pour tout

$$\boxed{n(\mathbf{x}) \geq N_{dec}}$$

$$\begin{aligned} \alpha_d(\mathbf{x}, \mathbf{x} \setminus \{x_i\} \cup \{u_1 \cup u_2\}) &\leq \frac{p_f M_h^2}{p_d m_h} \frac{n(\mathbf{x})}{\frac{(n(\mathbf{x})+1)(n(\mathbf{x})+1-N_{dec})}{2 \cdot N_{dec}}} \frac{M_{f_f}}{m_{f_d}} d_{max} \\ &\leq \frac{p_f M_h^2}{p_d m_h} \frac{2 \cdot N_{dec}}{n(\mathbf{x}) + 1 - N_{dec}} \frac{M_{f_f}}{m_{f_d}} d_{max}. \end{aligned}$$

Pour tout $\epsilon \in (0,1)$, on peut trouver un $N_{fd,\epsilon}$ assez grand tel que le membre de droite dans l'inégalité précédente soit inférieur à ϵ . Ainsi, dès que $\boxed{n(\mathbf{x}) \geq \max\{N_{fd,\epsilon}, N_{dec}\}}$, on a :

$$\alpha_d(\mathbf{x}, \mathbf{x} \setminus \{x_i\} \cup \{u_1 \cup u_2\}) \leq \epsilon$$

et avec les mêmes conditions

$$\alpha_f(\mathbf{x}, \mathbf{x} \setminus \{x_i \cup x_j\} \cup \{u\}) = 1.$$

Etape 3 :

On pose ensuite $V(\mathbf{x}) = A^{n(\mathbf{x})}$ avec $A > 1$, et on écrit $K(\cdot, \cdot)$ le noyau de transition de la chaîne de Markov. En reprenant les notations de (2.26), et en prenant $n(\mathbf{x}) \geq N_\epsilon = \max\{N_{nm, \epsilon}, N_{fd, \epsilon}, N_{dec}\}$:

$$KV(\mathbf{x}) = \left(\frac{p_{nm}p_m\alpha_m + p_{fd}p_f\alpha_f}{A} + \underbrace{p_{rms}}_{\text{rejets et mvmts simples}} + A.(p_{nm}p_n\alpha_n + p_{fd}p_d\alpha_d) \right) V(\mathbf{x})$$

avec

$$p_{rms} = p_{nm}p_m(1 - \alpha_m) + p_{fd}p_f(1 - \alpha_f) + p_{nm}p_n(1 - \alpha_n) + p_{fd}p_d(1 - \alpha_d) + p_{tr} + p_{di} + p_{trdi} + p_\emptyset.$$

D'où

$$KV(\mathbf{x}) \leq \underbrace{\left(\frac{p_{nm}p_m + p_{fd}p_f}{A} + [p_{nm}p_n + p_{fd}p_d + p_{tr} + p_{di} + p_{trdi} + p_\emptyset] + \epsilon.A.(p_{nm}p_n + p_{fd}p_d) \right)}_{G(\epsilon)} V(\mathbf{x}).$$

Or,

$$\begin{aligned} \lim_{\epsilon \rightarrow 0} G(\epsilon) &= p_{nm}\left(p_n + \frac{p_m}{A}\right) + p_{fd}\left(p_d + \frac{p_f}{A}\right) + p_{tr} + p_{di} + p_{trdi} + p_\emptyset \\ &< p_{nm} + p_{fd} + p_{tr} + p_{di} + p_{trdi} + p_\emptyset = 1 \end{aligned}$$

puisque $A > 1$.

En choisissant ϵ assez petit, on peut donc trouver une constante $a < 1$ telle que

$$KV(\mathbf{x}) \leq aV(\mathbf{x}),$$

dès que $n(\mathbf{x}) \geq N_\epsilon$.

Il suffit alors de choisir comme ensemble petit l'ensemble borné $\mathcal{C} = \{\mathbf{x} \in \Psi : n(\mathbf{x}) < N_\epsilon\}$, et comme constante b un majorant de $KV(\mathbf{x})$ sur cet ensemble \mathcal{C} , à savoir $b = A^{N_\epsilon + 1}$. La condition de drift géométrique est donc bien vérifiée puisque

$$KV(\mathbf{x}) \leq aV(\mathbf{x}) + b\mathbf{1}_{\mathcal{C}}(\mathbf{x}), \forall \mathbf{x} \in \Psi.$$

Ceci achève la démonstration. ◇

Bibliographie

- [1] P. Brémaud. *Markov chain : Gibbs fields, Monte Carlo simulation and queues*. Springer-Verlag, 2001.
- [2] L. Garcin, X. Descombes, J. Zerubia, and H. Le Men. Building detection by Markov object point processes and a MCMC algorithm. Rapport de Recherche n° 4206, INRIA, juin 2001.
- [3] C.J. Geyer and J. Moller. *Stochastic geometry, likelihood and computation*, chapter Likelihood inference for spatial point processes. Chapman and Hall, Londres, 1998.
- [4] F.A. Gougeon. Individual tree identification from high resolution MEIS images. In M.D. Gillis and D.G. Leckie, editors, *Proc. of the International Forum on Airborne Multispectral Scanning for Forestry and Mapping (with Emphasis on MEIS)*, pages 117–128, Forêts Canada, Val-Morin, Québec, 13-16 avril 1992.
- [5] F.A. Gougeon. A crown-following approach to the automatic delineation of individual tree crowns in high spatial resolution aerial images. *Canadian Journal of Remote Sensing*, 21(3), pages 274–284, 1995.
- [6] F.A. Gougeon. A system for individual tree crown classification of conifer stands at high spatial resolution. In *Proc. of the 17th Canadian Symposium on Remote Sensing*, pages 635–642, Saskatchewan, Canada, 13-15 juin 1995.
- [7] F.A. Gougeon. Automatic individual tree crown delineation using a valley-following algorithm and rule-based system. In D.A. Hill and D.G. Leckie, editors, *Proc. of the International Forum on Automated Interpretation of High Spatial Resolution Digital Imagery for Forestry*, pages 11–23, Victoria, British Columbia, Canada, 10-12 février 1998.
- [8] F.A. Gougeon and D.G. Leckie. Forest regeneration: individual tree crown detection techniques for density and stocking assessment. In D.A. Hill and D.G. Leckie, editors, *Proc. of the International Forum on Automated Interpretation of High Spatial Resolution Digital Imagery for Forestry*, pages 169–177, Victoria, British Columbia, Canada, 10-12 février 1998.
- [9] F.A. Gougeon and D.G. Leckie. Individual tree crown image analysis: a step towards precision forestry. In *Proc. of the 1st International Precision Forestry Symposium*, 17-20 juin 2001.
- [10] F.A. Gougeon, D.G. Leckie, I. Scott, and D. Paradine. Individual tree crown species recognition: the Nahmint study. In D.A. Hill and D.G. Leckie, editors, *Proc. of the International Forum on Automated Interpretation of High Spatial Resolution Digital Imagery for Forestry*, pages 209–223, Victoria, British Columbia, Canada, 10-12 février 1998.
- [11] P.J. Green. Reversible jump Markov chain Monte Carlo computation and bayesian model determination. *Biometrika* 82, pages 711–7320, 1995.
- [12] P.J. Green and A. Mira. Delayed rejection in Reversible Jump Metropolis Hastings. Technical Report S-01-99, University of Bristol, 1999.
- [13] M. Imberty and X. Descombes. Simulation de processus objets : étude de faisabilité pour une application à la segmentation d'image. Rapport de Recherche n° 3881, INRIA, février 2000.
- [14] C. Lacoste, X. Descombes, and J. Zerubia. A comparative study of point processes for line network extraction in remote sensing. Rapport de Recherche n° 4516, INRIA, juillet 2002.
- [15] M. Larsen. Finding an optimal match window for spruce top detection based on an optical tree model. In D.A. Hill and D.G. Leckie, editors, *Proc. of the International Forum on Automated Interpretation of High Spatial Resolution Digital Imagery for Forestry*, pages 55–66, Victoria, British Columbia, Canada, 10-12 février 1998.

- [16] M. Larsen. Individual tree top position estimation by template voting. In *Proc. of the Fourth International Airborne Remote Sensing Conference and Exhibition / 21st Canadian Symposium on Remote Sensing*, volume 2, pages 83–90, Ottawa, Ontario, 21-24 juin 1999.
- [17] M. Larsen and M. Rudemo. Using ray-traced templates to find individual trees in aerial photographs. In Michael Frydrych, Jussi Parkkinen, and Ari Visa, editors, *Proc. of the 10th Scandinavian Conference on Image Analysis*, volume 2, pages 1007–1014, Lappeenranta, Finland, 1997.
- [18] D.G. Leckie, C. Burnett, T. Nelson, C. Jay, N. Walsworth, F.A. Gougeon, and E. Cloney. Forest parameter extraction through computer-based analysis of high resolution imagery. In *Proc. of the Fourth International Airborne Remote Sensing Conference and Exhibition / 21st Canadian Symposium on Remote Sensing*, Ottawa, Ontario, 21-24 juin 1999.
- [19] D.G. Leckie, M. Gillis, F.A. Gougeon, M. Lodin, J. Wakelin, and Y. Xiaoping. Computer-assisted photointerpretation aid to forest inventory mapping: some possible approaches. In D.A. Hill and D.G. Leckie, editors, *Proc. of the International Forum of Automated Interpretation of High Spatial Resolution Digital Imagery for Forestry*, pages 335–343, Victoria, British Columbia, Canada, 10-12 février 1998.
- [20] D.G. Leckie and F.A. Gougeon. An assessment of both visual and automated tree counting and species identification with high spatial resolution multispectral imagery. In D.A. Hill and D.G. Leckie, editors, *Proc. of the International Forum of Automated Interpretation of High Spatial Resolution Digital Imagery for Forestry*, pages 141–152, Victoria, British Columbia, Canada, 10-12 février 1998.
- [21] J. Lund. *Statistical inference and perfect simulation for point processes observed with noise*. PhD thesis, The Royal Veterinary and Agricultural University, Department of Mathematics and Physics, Copenhagen, décembre 1999.
- [22] J. Lund and M. Rudemo. Models for point processes observed with noise. *Biometrika*, 87(2):235–249, 2000.
- [23] S.P. Meyn and R.L. Tweedie. *Markov Chains and Stochastic Stability*. Springer-Verlag, 1993.
- [24] V. Muron and C. Jacquet. Mise au point de méthodes pour le comptage des oliviers. *Bulletin SFPT*, (164/165), 2001.
- [25] M. Ortner, X. Descombes, and J. Zerubia. Building extraction from digital elevation model. Rapport de Recherche n° 4517, INRIA, juillet 2002.
- [26] P.S. Petersen. *Finding single trees in aerial imagery*. PhD thesis, The Royal Veterinary and Agricultural University, Department of Economics and Natural Resources, Copenhagen, décembre 2001.
- [27] C. Robert. *Méthodes de Monte Carlo par chaînes de Markov*. Economica, Paris, 1996.
- [28] H. Rue. New loss function in bayesian imaging. *American Statistical Association*, 90(431):900–908, septembre 1995.
- [29] H. Rue. A loss function model for the restoration of grey level images. *Scandinavian Journal of Statistics*, 24:103–114, 1997.
- [30] H. Rue. Baddeley’s delta metric. *Encyclopedia of Statistical Sciences*, 3:158–162, 1999.
- [31] H. Rue and M. Hurn. Bayesian object identification. *Biometrika*, 86:649–660, mai 1999.
- [32] H. Rue and A.R. Syversveen. Bayesian object recognition with Baddeley’s delta loss. *Advances Applied Probability*, 30:64–84, 1998.
- [33] R. Stoica, X. Descombes, and J. Zerubia. A Markov point process for road extraction in remote sensed images. Rapport de Recherche n° 3923, INRIA, avril 2000.
- [34] D. Stoyan, W.S. Kendall, and J. Mecke. *Stochastic Geometry and its applications*. Wiley, 2^{nde} édition, 1995.
- [35] M.N.M. van Lieshout. *Markov point processes and their applications*. Imperial College Press, Londres, 2000.