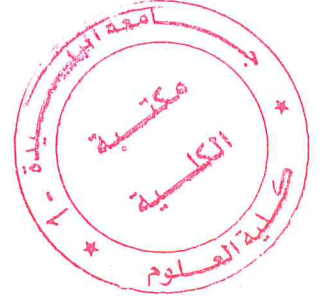


الجمهورية الجزائرية الديمقراطية الشعبية
 République Algérienne démocratique et populaire
 وزارة التعليم العالي والبحث العلمي
 Ministère de l'enseignement supérieur et de la recherche scientifique
 جامعة سعد دحلب البليدة
 Université SAAD DAHLAB de BLIDA



كلية العلوم
 Faculté des Sciences
 قسم الرياضيات
 Département de Mathématique



Mémoire de Master

Filière : Mathématiques

Spécialité : Modélisation Stochastique et Statistique

Thème

PLANIFICATION D'EXPÉRIENCES NUMÉRIQUES A
 PARTIR DU PROCESSUS PONCTUEL DE BADELLEY

Proposé par :

ELMOSSAOUI HICHAM

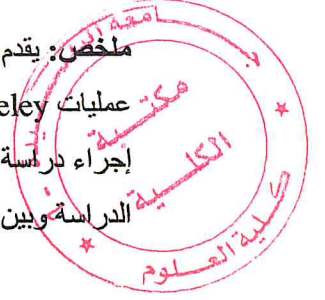
Présenté par :

EL ALEM KABER MOHAMED

BOUTEBAL MANAL

Année universitaire : 2018-2019

ملخص: يقدم هذا العمل طريقة لبناء نماذج تجريبية رقمية. هذه الطريقة تعتمد على نظرية عمليات النقاط، وخاصة عمليات Baddeley. يتم الحصول على هذه النماذج من خلال استعمال طريقة (سلسلة Markov Monte Carlo). تم إجراء دراسة مفصلة حول تقارب سلسلة Markov. كما قمنا كذلك بإجراء مقارنة بين النموذج الذي اعتمدهنا خلال هذه الدراسة وبين بعض النماذج الرقمية الأخرى الموجودة.



Résumé: Ce travail propose une méthode de construction de plans d'expériences numériques. Cette méthode est basée sur la théorie des processus ponctuels, particulièrement les processus de Baddeley. Ces plans sont obtenus à partir de la méthode de Monte Carlo par chaîne de Markov (MCMC) (Markov Chain Monte Carlo). Une étude détaillée sur la convergence de la chaîne de Markov a été réalisée. Une comparaison entre notre approche et celle donnée par d'autres plans numériques existants a été effectuée.

Abstract: This work proposes a method for building numerical experiment designs. This method is basing on the theory of point processes, particularly the Baddeley processes. These designs are obtaining from Monte Carlo Markov Chain Method (MCMC). A detailed study about the convergence of the Markov chain has been carrying out. Moreover, a comparison between our approach and the one, which is giving by other existing numerical designs, has been drawing.

REMERCIEMENTS

Ce mémoire marque une étape très importante dans notre vie. Nous voudrions remercier ici les personnes qui nous ont accompagnés au cours de notre parcours et qui ont participé de près ou de loin à la réalisation de ce projet.

Nous remercions avant tout Allah, tout puissant pour la volonté, la santé et la patience qu'il nous a données durant toutes ces longues années d'études afin que nous puissions arriver à ce stade.

En second lieu, nous tenons à remercier notre encadreur Dr :MOUSSAOUI Hicham, d'avoir proposé et encadré ce sujet, on lui exprime notre profonde gratitude pour l'aide précieuse qu'il nous a apporté et qui nous a permis de profiter de ses connaissances, mais aussi de ses méthodes de travail, et surtout de sa rigueur scientifique.

Egalement, nous remercions l'ensemble des professeurs et enseignants de département de Mathématique de l'université de saad dahlab qui ont participé à cette formation .

Enfin, nous oublions pas de remercier nos familles ,nos amis et nos collègues et toutes les personnes qui auront contribué de près ou de loin à l'élaboration de ce mémoire ainsi qu'à la réussite de cette formidable année universitaire.

DÉDICACES

Je dédie ce mémoire

A ma très chère Imma Fatma Zohra Zahra Affable, honorable, aimable :

Tu représentes pour moi le symbole de la bonté par excellence, la source de tendresse et l'exemple du dévouement qui n'a pas cessé de m'encourager et de prier pour moi. Ta prière et ta bénédiction m'ont été d'un grand secours pour mener à bien mes études. Aucune dédicace ne saurait être assez éloquente pour exprimer ce que tu mérites pour tous les sacrifices que tu n'as cessé de me donner depuis ma naissance, durant mon enfance et même à l'âge adulte. Tu as fait plus qu'une mère puisse faire pour que ses enfants suivent le bon chemin dans leur vie et leurs études. Je te dédie ce travail en témoignage de mon profond amour. Puisse Dieu, le tout puissant, te préserver et t'accorder santé, longue vie et bonheur incha-allah.

A mon très cher Vava Boutebal Mohamed :

Aucune dédicace ne saurait exprimer l'amour, l'estime, le dévouement et le respect que j'ai toujours eu pour vous. Rien au monde ne vaut les efforts fournis jour et nuit pour mon éducation et mon bien être. Ce travail est le fruit de tes sacrifices que tu as consentis pour mon éducation et ma formation.

A tous ceux qui connaissent Manal de près ou de loin.

Manal BOUTEBAL

DÉDICACES

Je dédie ce modeste mémoire de fin d'étude :

À DIEU, pour m'avoir donné la force dans les moments difficiles d'éditer ce mémoire.

À mes chers parents :

Ma mère, qui a œuvré pour ma réussite, de par son amour, son soutien, tous les sacrifices consentis et ses précieux conseils, pour toute son assistance et sa présence dans ma vie, reçois à travers ce travail aussi modeste soit-il, l'expression de mes sentiments et de mon éternelle gratitude.

Mon père, qui peut être fier et trouver ici le résultat de longues années de sacrifices et de privations pour m'aider à avancer dans la vie. Puisse Dieu faire en sorte que ce travail porte son fruit, merci pour les valeurs nobles, l'éducation et le soutien permanent venu de toi.

À mes chères sœurs et chers frères pour leurs encouragements permanents, et leur soutien moral, à toute ma famille pour leur soutien tout au long de mon parcours universitaire.

À mes amis et frères à Blida sans exception et au pays en particulier mes amis à Néma .

À tous mes chers collègues de de département de Mathématique l'Université de blida1. Et à tous ce qui m'ont enseigné au long de ma vie scolaire.

À tous ceux qui ont m'aider de près ou de loin.

Mohamed Kaber El Alem

TABLE DES FIGURES

1.1	Plan aléatoire à 30 points en dimension 2.	4
1.2	Grille régulière à 25 points en dimension 2.	5
1.3	Grille régulière randomisée à 6^2 points.	6
1.4	Un échantillonnage par hypercube latin à 5 points en dimension 2	7
1.5	Un hypercube latin à 20 points en dimension 2	8
1.6	Un plan généré par un tableau orthogonal $OA_1(25, 5, 5, 2)$ dont les points sont centrés et projetés sur le sous-espace (X_1, X_2)	9
1.7	Un plan généré par un tableau orthogonal $OA_1(25, 5, 5, 2)$ dont les points sont randomisés et projetés sur le sous-espace (X_1, X_2)	10
1.8	premiers points de la suite de Halton pour $b_1 = 2$ et $b_2 = 3$	13
1.9	Représentation d'une suite de $n = 30$ points en dimension $d = 2$ et du pavé $P = [0, t_1[\times]0, t_2[$	15
1.10	Plans maximin et minimax de $N=6$ points dans $[0, 1]^2$	18
2.1	Répartition de la population humaine sur la Terre en 2005.	19
2.2	La Terre vue de l'espace la nuit.	20
2.3	Ensemble de points, ou configuration, de $\chi = [0, 1] \times [0, 1]$	21
2.4	configuration à 4 cliques.	22
2.5	Processus poisson : a. d'intensité $\lambda = 0,01$ b. d'intensité $\lambda = 0,02$ c. processus non-homogène.	24
3.1	Les familles d'algorithmes pour la simulation des chaînes MCMC	30
4.1	Une réalisation de l'ensemble $B(x)$ permettant de définir l'aire $a(x)$ et le nombre de composantes connexes $c(x)$ (ici $c(x) = 19$).	38
4.2	(a) une configuration initiale de 25 points et (b) une configuration finale pour $\gamma = 0.01$ et $r = 0.1$	42

4.3	(a) une configuration de 25 points $\gamma = 0.01$ et $r = 0.1$ et (b) une configuration pour $\gamma = 0.01$ et $r = 0.2$	44
4.4	(a) une configuration de 25 points $\gamma = 0.001$ et $r = 0.1$ et (b) une configuration pour $\gamma = 0.1$ et $r = 0.1$	44
5.1	Boxplots des critères de qualité pour deux dimensions	52
5.2	Boxplots des critères de qualité pour trois dimensions	53
5.3	Boxplots des critères de qualité pour cinq dimensions	54
5.4	Boxplots des critères de qualité pour sept dimensions	55
5.5	Boxplots des critères de qualité pour dix dimensions	56

LISTE DES TABLEAUX

5.1	Listes des différents plans selon les quartes critères avec $n=10$, $R=0.1$, $\gamma = 0.05$, $NMC=500$, $d=2$	51
5.2	Listes des différents plans selon les quartes critères avec $n=20$, $R=0.1$, $\gamma = 0.05$, $NMC=500$, $d=3$	52
5.3	Listes des différents plans selon les quartes critères avec $n=40$, $R=0.1$, $\gamma = 0.05$, $NMC=500$, $d=5$	53
5.4	Listes des différents plans selon les quartes critères avec $n=50$, $R=0.1$, $\gamma = 0.05$, $NMC=500$, $d=7$	54
5.5	Listes des différents plans selon les quatre critères avec $n=80$, $R=0.1$, $\gamma = 0.05$, $NMC=500$, $d=10$	56

LISTE DES ALGORITHMES

1	Algorithme de RWMH pour générer des plans dits de Baddeley	41
2	L'Algorithme proposé pour calculer le nombre de composantes connexes	43

TABLE DES MATIÈRES

Table des Figures	i
Liste des Tables	iii
Liste des Algorithmes	iv
Introduction générale	1
1 les Plans « space filling »	3
1.1 Introduction	3
1.2 Plans stratifiés	3
1.2.1 Grilles régulières	5
1.2.2 Hypercubes latins	6
1.2.3 Tableaux orthogonaux	8
1.3 Suites à faible discrédance	10
1.3.1 Construction d'une suite à discrédance faible	12
1.4 Plans optimisés	13
1.5 Critères de comparaison de plans d'expériences	13
1.5.1 Mesure de recouvrement (Rec)	14
1.5.2 Distance entre points (minimax et maximin)	14
1.5.3 Discrédance en norme L2 (DisL2)	15
1.6 Plans maximin et minimax	16
1.6.1 Plans maximin	16
1.6.2 Plans minimax	18
2 Processus ponctuels spatiaux	19
2.1 Introduction et motivations	19

2.2	Définitions et notations	20
2.2.1	Espace des configurations	21
2.2.2	Notions de voisinage et clique	22
2.2.3	Processus ponctuels	23
2.3	processus ponctuel de Poisson	23
2.4	Processus ponctuels marqués	25
2.5	Processus ponctuels de Markov	26
3	La simulation par Méthodes de Monte Carlo par chaine de Markov	28
3.1	Chaînes de Markov	28
3.1.1	Premières définitions	28
3.1.2	Quelques critères de classification	29
3.1.3	Théorème ergodique et distribution stationnaire	29
3.1.4	La convergence des chaines de Markov	29
3.1.5	L'échantillonnage des chaînes de Markov	30
3.2	Les méthodes MCMC	31
3.2.1	Chaîne de Markov Monte-Carlo (MCMC)	32
3.2.2	L'algorithme de type naissance-mort	32
3.2.3	L'algorithme de type Metropolis-Hastings	33
3.2.4	L'algorithme Metropolis-Hastings-Green	35
3.2.5	L'échantillonneur Gibbs	36
4	Processus ponctuel de Baddeley et planification d'expériences	37
4.1	Le processus ponctuel de Baddeley	37
4.2	Le processus de Baddeley est Markovien	38
4.2.1	Propriété de Markov au sens de Ripley-Kelly	39
4.2.2	Propriété de Markov aux plus proches voisins	40
4.3	Générer des plans d'expériences de Baddeley	40
4.3.1	Algorithme Metropolis-Hastings avec marche aléatoire	41
4.3.2	Le calcul de $c(x)$: nombre de composantes connexes	42
4.3.3	Influence des paramètres	43
4.4	Étude de convergence	45
5	IMPLÉMENTATION, DISCUSSIONS ET COMPARAISON	49
5.1	Matlab comme outil de programmation	49
5.2	Etude de comparaison avec les plans usuels	50
5.3	Résultats et discussions	51
5.3.1	Résultat pour 10 points et 2 dimensions	51
5.3.2	Résultat pour 20 points et 3 dimensions	52
5.3.3	Résultat pour 40 points et 5 dimensions	53
5.3.4	Résultat pour 50 points et 7 dimensions	54
5.3.5	Résultat pour 80 points et 10 dimensions	55
	Conclusion générale	57
	Bibliographie	61

INTRODUCTION GÉNÉRALE

Dans le cadre de la simulation numérique, les expériences simulées sont coûteuses en temps de calcul, il est donc important de disposer des méthodes permettant d'optimiser leurs planification. De nombreux phénomènes sont modélisés par des codes de simulation numériques comme outil d'optimisation et de prévision. Ces codes sont constitués de modèles numériques reliant des variables descriptives de l'état du système à un certain nombre de paramètres caractéristiques. Le simulateur prend ses paramètres en entrée et délivre en sortie les réponses sur les variables d'état du système. Malgré les progrès toujours croissants de l'informatique, ces codes de calcul sont complexes et souvent très long à exécuter. Pour remédier à ce problème, l'idée c'est de remplacer le simulateur par une ou plusieurs fonctions approchées qui sont construites à l'aide de méthodes d'approximation ou d'interpolation et à partir d'expériences numériques.

Dans le cas général, les techniques des plans classiques fondées sur les travaux de Fisher [1] telles que la randomisation et la répétabilité ne sont plus forcément de mise. En effet, les plans classiques ont tendance à positionner les points aux bords du domaine pour prendre en compte la variation aléatoire et offrir une tendance plus fiable en présence d'erreurs de mesures, on pense par exemple aux plans Box Behnken, plans composites, ..etc. Et La plupart des critères de ces plans utilisent l'erreur expérimentale qui existe dans le cadre d'expériences réelles. tandis qu'en numérique, c'est une erreur liée au modèle et non à l'expérimentation. Aussi la répétition des expériences classiques est d'un grand intérêt pour évaluer l'erreur expérimentale. Tandis que dans le cas des expériences numériques la répétition d'une expérience dans les mêmes conditions est sans intérêt puisqu'elle fournit la même réponse.

Notre objectif étant de modéliser le code en phase exploratoire (quand aucune simulation n'a encore été réalisée), on recherchera souvent à satisfaire certains critères comme par exemples :

- a) Le remplissage de l'espace expérimental en maximisant la distance entre de points les plus proche du plan.
- b) L'uniformité (discrétance) : on cherche à minimiser la distance entre une distribution de points donnée et une distribution uniforme.
- c) L'indépendance : on cherche à maximiser le déterminant de la matrice des corrélation des paramètres ou des points.

Portant du costal que les méthodes usuelles de construction de plans d'expériences ne sont pas adaptées, nous proposons d'utiliser les processus ponctuelles pour simulés les n expériences (points) qui constituent le plans d'expériences. Dans ce contexte il a été naturel de considérer les processus de Baddeley qui incluent la notion d'interaction d'aire. Pour générer de tels processus nous utilisons les techniques de simulation par chaine de markov(MCMC), et plus précisément l'algorithme de Métropolis- Hasting. L'optimisation globale d'un processus objet est maintenant un domaine pour lequel la littérature est riche de résultats directement applicables à notre approche, en particulier grâce à l'apport des méthodes MCMC. Nous avons réparti notre mémoire en cinq chapitres, une introduction et une conclusion.

Le premier chapitre de ce manuscrit présente la théorie de la méthodologie des plans d'expériences et un état de l'art des plans d'expériences couramment utilisés en expérimentation numérique ainsi que de certains critères pour étudier la qualité intrinsèque de ces plans au niveau du remplissage de l'espace.

Le deuxième chapitre introduit la théorie des processus ponctuels, processus ponctuels marqués, les processus Gibbs et Markov qui incluent la notion d'interactions entre objets. Quelques résultats théoriques assurant la convergence des principaux algorithmes de simulation des chaines de Markov Monte Carlo sont présentés.

Dans le chapitre 3 nous décrivons quelques généralités et notions de base des chaines de Markov, ainsi nous parlons des différentes méthodes de simulation de type Monte-Carlo par Chaîne de Markov (MCMC).

Dans le chapitre 4 nous présentons une nouvelle méthode de planification d'expériences basée sur des méthodes de Monte-Carlo par chaîne de Markov pour simuler une suite des plans d'expériences convergente tout en utilisant les processus ponctuels de Baddeley.

Enfin le dernier chapitre présente les résultats obtenus par l'implémentation des différentes méthodes étudiées dans le premier et le quatrième chapitre.

CHAPITRE

1

LES PLANS « SPACE FILLING »

1.1 Introduction

Dans le cadre de l'exploration initiale d'un code de calcul dépendant de k paramètres à l'aide de N calculs, l'utilisation n'a souvent pas de connaissance sur les paramètres influents et sur les paramètres non influents de son code, il est donc judicieux dans ce cas-là de disposer les N points de calculs de manière à bien recouvrir l'espace K -dimensionnel, il ne faut donc ne sous-échantillonner aucune zone et n'en sur-échantillonner aucune autre (pour ne pas gaspiller des points de calcul). Les plans permettant de remplir cette mission sont nommés plans à bon remplissage de l'espace (space filling design en anglais). On trouve essentiellement deux types de ces plans :

1. Les plans à marges uniformes tels que les hypercubes latins ou les tableaux orthogonaux (Mc Kay et al. 1979 [2], Stein 1987 [3], Owen 1992 [4]). En projection sur un axe ou une face du domaine la répartition des points est uniforme.
2. Les suites de faible discrédance telles que les lattices (réseaux) à un générateur (Niederreiter 1992 [5], Fang et Wang 1994 [6]). La distribution empirique des points doit être proche de la distribution uniforme.

1.2 Plans stratifiés

La construction d'un plan d'expériences numériques fait appel à une méthode d'échantillonnage (de tirage) pour sélectionner les points en lesquels le modèle sera évalué. La méthode d'échantillonnage la plus simple consiste à générer indépendamment pour chaque variable X_j un ensemble de n réalisations indépendantes d'une variable aléatoire

de loi uniforme sur $[0, 1]$. Plus formellement, l'ensemble des points résultant d'un tel échantillonnage est noté $\mathcal{P} = \{X_i\}_{i=1}^n$ avec

$$X_i = (U_{i,1}, \dots, U_{i,d}), \quad i \in \{1, \dots, n\} \quad (1.1)$$

où les $U_{i,j}$, $j \in \mathcal{D}$, sont des variables aléatoires indépendantes uniformément distribuées sur $[0, 1]$. Un tel échantillonnage est qualifié d'échantillonnage aléatoire, le plan d'expériences \mathcal{P} en résultant est appelé plan aléatoire.

Lorsqu'on travaille à n fixé, un tel plan ne donne aucune garantie quant à une bonne répartition des points dans l'espace. Un plan aléatoire fait très souvent apparaître à la fois des zones lacunaires et des zones où se concentrent les points échantillonnés. Ce phénomène est illustré sur la figure 1.1 pour une dimension $d = 2$ et un nombre de points $n = 30$.

Ce type de problèmes peut être amélioré en considérant des plans d'expériences issus de méthodes d'échantillonnage consistant à partitionner l'espace des entrées en sous-domaines distincts (d'intersection vide deux à deux). Une fois ce découpage effectué, un point est échantillonné aléatoirement dans chaque sous-domaine.

Les plans que nous décrivons dans cette section reposent sur cette idée de découpage et sont appelés plans stratifiés. Le terme strate réfère à un des sous-domaines issu du découpage. Nous nous focalisons en particulier sur les cas de grilles régulières, hypercubes latins et tableaux orthogonaux.

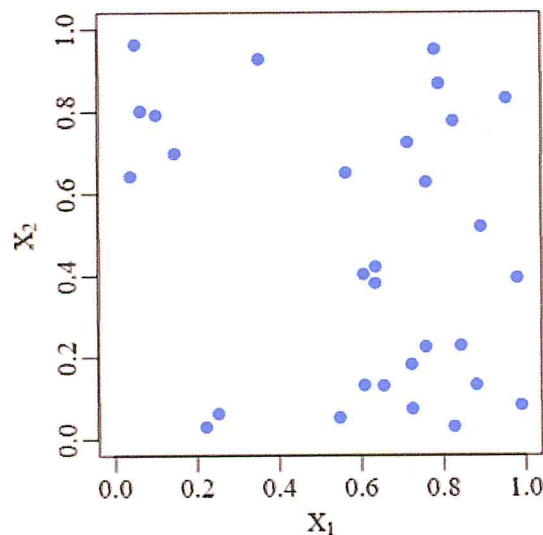


FIGURE 1.1 – Plan aléatoire à 30 points en dimension 2.

1.2.1 Grilles régulières

Une grille régulière est obtenue en discrétisant indépendamment chaque variable X_j , $j \in D$, en un ensemble de n valeurs distinctes puis en formant les n^d combinaisons possibles. La figure 1.2 illustre un exemple d'une grille régulière à 25 points en dimension 2. Une grille régulière correspond à un plan factoriel complet à d facteurs où chaque facteur possède n niveaux distincts.

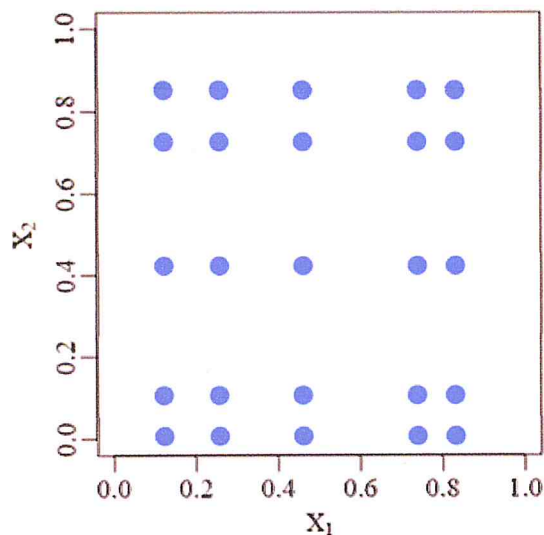


FIGURE 1.2 – Grille régulière à 25 points en dimension 2.

Les points d'une grille régulière sont alignés le long d'axes identifiés par les n valeurs choisies pour chaque variable. Il est possible de casser cette structure en considérant le plan suivant :

Definition 1. Soit $\mathcal{P} = \{X_i\}_{i=1}^n$ l'ensemble de points issus de la grille régulière construite en considérant pour chaque variable X_j , $j \in D$, le même jeu de n valeurs $\{1/n, 2/n, \dots, 1\}$. Soit l'ensemble de points $\mathcal{P}_0 = \{Z_i\}_{i=1}^n$ construit par :

$$Z_i = (X_{i,1} - U_{i,1}/n, \dots, X_{i,d} - U_{i,d}/n). \quad i \in \{1, \dots, n\}, \quad (1.2)$$

où les $U_{i,j}$, $j \in D$, sont des variables aléatoires indépendantes uniformément distribuées sur $[0, 1]$. Alors \mathcal{P}_0 est une grille régulière randomisée.

La figure 1.3 illustre une grille régulière randomisée \mathcal{P}_0 à 6^2 points construite suivant (1.2). Une grille régulière randomisée est obtenue en faisant varier chaque point d'une grille régulière indépendamment les uns des autres à l'intérieur de leur strate respective. Ces types de plans stratifiés sont adaptés uniquement dans le cas de petites dimensions, 2 ou 3. Pour des plus grandes valeurs de d , le nombre de points créés (n^d) croît exponentiellement et devient rapidement inadapté aux modèles requérant un temps d'évaluation important.

Une alternative consiste à sélectionner un sous-ensemble des points Z_i créés suivant l'équation (1.2), de façon à assurer une bonne répartition des valeurs prises par chaque variable X_j . Une catégorie de plans d'expériences résultant d'une telle approche correspond aux hypercubes latins.

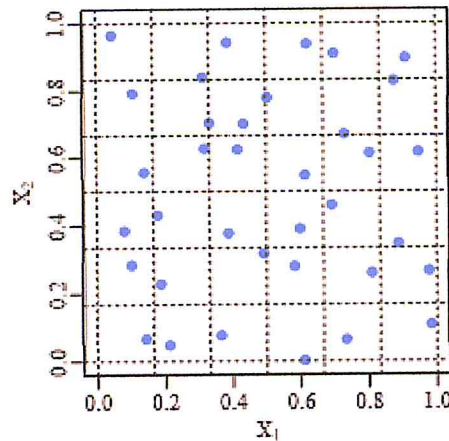


FIGURE 1.3 – Grille régulière randomisée à 6^2 points.

1.2.2 Hypercubes latins

En expérimentation numérique, il est courant de traiter des problèmes avec une faible taille d'échantillon (de l'ordre de la centaine) et un grand nombre de paramètres d'entrée (plusieurs dizaines). Des alternatives à la méthode de Monte Carlo ont donc été développées, non seulement pour accélérer la convergence des estimations, mais aussi pour s'assurer que les domaines de variation des paramètres d'entrée soient bien échantillonnés. En effet, placer quelques points au hasard dans un espace de grande dimension conduit le plus souvent à une mauvaise couverture de certaines dimensions. L'une des premières méthodes développées pour contrecarrer ce problème a été celle des hypercubes latins, connue sous l'acronyme LHS pour Latin Hypercube Sampling .

Description

Chaque axe $[0, 1]$ du cube unité est découpé en n segments de même longueur de la façon suivante $\{[0, \frac{1}{n}], [\frac{1}{n}, \frac{2}{n}], \dots, [\frac{n-1}{n}, 1]\}$. En faisant le produit cartésien de ces intervalles, on obtient un maillage de n^d cellules de même taille. Ensuite, on choisit n cellules parmi les n^d possibles de telle manière que les cellules des marges de dimension 1 soient représentées une fois chacune. Enfin, on tire un point au hasard dans chaque cellule présélectionnée.

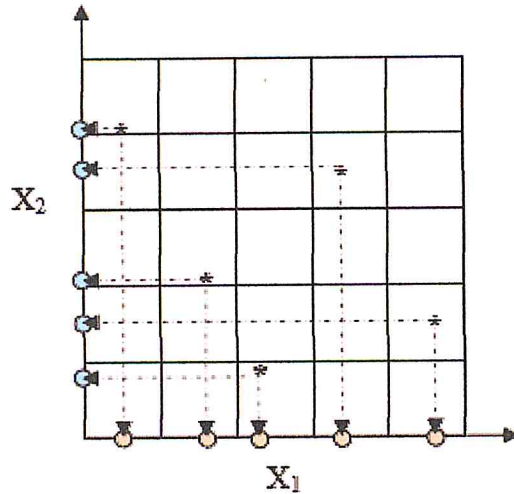


FIGURE 1.4 – Un échantillonnage par hypercube latin à 5 points en dimension 2

Definition 2. Un hypercube latin à n points sur $[0, 1]^d$, est défini par l'ensemble de points X^i tel que

$$X_j^i = \frac{\pi_j(i) + U_j^{(i)}}{n} \quad (1.3)$$

où π_j est une permutation de $1, \dots, n$ et $U_j^{(i)} = U[0, 1]$ est une valeur aléatoire de distribution uniforme sur $[0, 1]$. Ainsi, $(\pi_1(i), \dots, \pi_d(i))$ représente la cellule dans laquelle se trouve le point X^i , et $(U_1^{(i)}, \dots, U_d^{(i)})$ désigne à quel endroit le point se trouve dans la cellule. L'hypercube latin obtenu correspond donc à la matrice de n lignes et d colonnes à coefficients X_j^i .

Remarque 1. Nous pouvons choisir de placer le point au centre des cellules pour ainsi supprimer l'aléa de la planification.

Remarque 2. Un hypercube latin, défini par la matrice π , est un plan très facile à construire puisque chaque colonne est une permutation de 1 à n .

Discussion

Les points d'un hypercube latin ont la propriété intéressante d'être uniformément distribués sur les axes factoriels (voir Figure 1.4). Cependant, cette propriété n'assure pas l'uniformité des points sur tout le domaine expérimental. Pour n fixé, il existe $n!$ permutations possibles pour chacune des d colonnes, et donc $(n!)^{d-1}$ hypercubes latins éventuels qui n'assurent pas tous l'uniformité des points dans l'espace.

De façon générale, le contexte dans lequel nous sommes ne nous autorisant pas à prendre un grand nombre de points, nous aurons un nombre de mailles réduit. Il sera donc fréquent d'obtenir des plans où l'on peut observer des zones lacunaires et des agglomérats de points comme on peut l'observer sur la Figure 1.5.

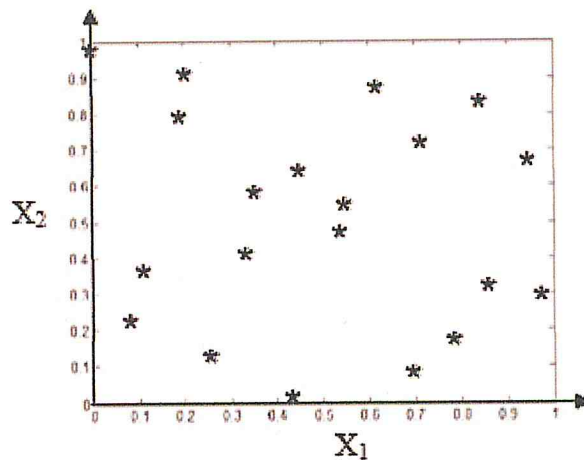


FIGURE 1.5 – Un hypercube latin à 20 points en dimension 2

Amélioration

Pour pallier ces problèmes, il existe deux possibilités. La première consiste à construire les $(n!)^{d-1}$ hypercubes latins possibles et de choisir le meilleur à l'aide d'un critère de sélection. Cependant, cette méthode n'est pas envisageable du fait du grand nombre d'hypercubes latins à générer, même si construire un hypercube latin demande peu de temps d'exécution. La deuxième possibilité, plus réaliste, consiste à générer un hypercube latin et à lui appliquer un critère afin de le rendre meilleur au sens des exigences souhaitées.

1.2.3 Tableaux orthogonaux

les tableaux orthogonaux (Owen, 1992 [7]) sont très proches des hypercubes latins. En effet, ils ont l'avantage d'avoir les bonnes propriétés de projection en grande dimension qu'ont les hypercubes latins en dimension 1.

Definition 3. *Un tableau orthogonal de force t à q symboles est une matrice de n lignes et $(d > t)$ colonnes avec ses éléments prenant q valeurs possibles, telle que toutes les sous matrices de taille $q \times t$ aient q^t lignes apparaissant le même nombre λ fois. Il est évident que $n = \lambda q^t$. On note un tel tableau orthogonal $OA_\lambda(n, d, q, t)$*

Description

Géométriquement, cela revient à découper les axes du cube unité en q segments égaux afin d'obtenir q^d cellules de même taille. On choisit n cellules de telle façon qu'elles forment un tableau orthogonal défini comme ci-dessus. Cette définition assure que chaque t colonnes de la matrice du plan, soit chaque t -uplet, apparaît exactement λ -fois.

Remarque 3. *Un tableau orthogonal de force 1 est un hypercube latin. Comme pour les hypercubes latins, on peut choisir le point aléatoirement dans la cellule mais aussi au centre de la cellule. Dans ce dernier cas, toutes les projections sur les sous espaces de dimensions t conduisent alors à une grille régulière comme on peut le constater sur la Figure 1.6.*

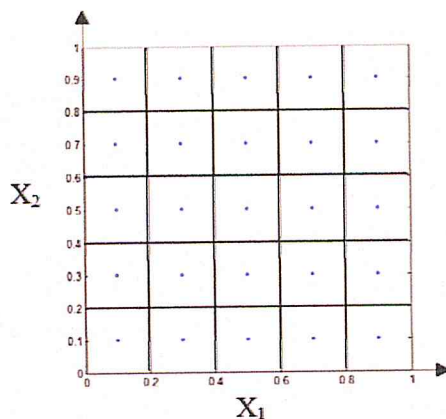


FIGURE 1.6 – Un plan généré par un tableau orthogonal $OA_1(25, 5, 5, 2)$ dont les points sont centrés et projetés sur le sous-espace (X_1, X_2) .

Definition 4. *Un échantillonnage par tableau orthogonal (abusivement appelé par la suite tableau orthogonal) à n points sur $[0, 1]^d$ est un ensemble de points X^i tel que*

$$X_j^i = \frac{\pi_j(A_j^{(i)}) + U_j^{(i)}}{q}, \quad 1 \leq i \leq n, \quad 1 \leq j \leq d \quad (1.4)$$

où π_j est une permutation de $0, \dots, q - 1$, $A_j^{(i)}$ sont les éléments du tableau orthogonal et $U_j^{(i)} \sim U[0, 1]$ est une valeur aléatoire de distribution uniforme sur $[0, 1]$. Ainsi, $(\pi_1(A_1^{(i)}), \dots, \pi_d(A_d^{(i)}))$ représente la cellule dans laquelle se trouve le point X^i , et $(U_1^{(i)}, \dots, U_d^{(i)})$ désigne à quel endroit le point se trouve dans la cellule. Le tableau orthogonal correspond à la matrice de n lignes et d colonnes à coefficients X_j^i .

Concernant la génération des plans, on a une propriété analogue à celle des hypercubes latins :

Propriété 1. *Si on effectue une permutation des symboles sur chaque colonne d'un tableau orthogonal de force t , alors on obtient toujours un tableau orthogonal de force t .*

Discussion

La Figure 1.7 montre les points générés par un tableau orthogonal de type $OA_1(25, 5, 5, 2)$ projetés sur le sous-espace (X_1, X_2) . Nous pouvons observer qu'il y a bien un unique point par cellule et donc que les points projetés sur le sous-espace (X_1, X_2) sont répartis uniformément.

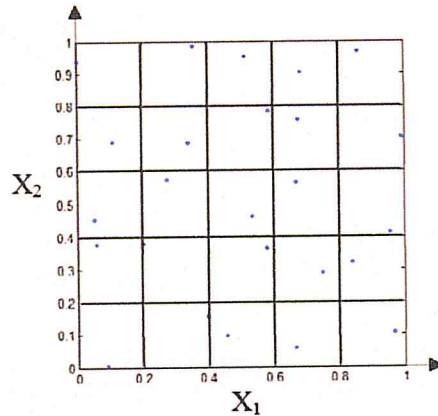


FIGURE 1.7 – Un plan généré par un tableau orthogonal $OA_1(25, 5, 5, 2)$ dont les points sont randomisés et projetés sur le sous-espace (X_1, X_2) .

Propriété 2. *Un tableau orthogonal de force t et d'index λ est aussi un tableau orthogonal de force $s \leq t$ et d'index $\lambda' = \frac{n}{q^s} \geq \lambda$*

Si l'on reprend les exemples des Figure 1.6 et Figure 1.7, ces tableaux orthogonaux de force 2 et d'index $\lambda = 1$ (1 point par cellule) sont également de force 1 et d'index $\lambda' = 5$ (5 points par strate).

Les tableaux orthogonaux de force t permettent de généraliser aux dimensions t les bonnes propriétés de projections en dimension 1 des hypercubes latins. Ainsi, les points sont globalement répartis plus uniformément que pour les hypercubes latins.

1.3 Suites à faible discrédance

Dans la partie précédente, nous avons discuté de plans dont les points sont bien répartis en projection mais pas forcément dans l'espace. Nous présentons ici des plans dont l'objectif est de bien remplir l'espace. Aussi, nous nous attachons à regarder leurs propriétés en projection.

La plupart de ces suites à faible discrédance sont générées par des algorithmes déterministes afin d'obtenir des points répartis le plus uniformément possible dans le

domaine expérimental.

La discrédance mesure l'écart entre une distribution de points uniforme et une distribution de points donnée, en d'autres termes elle permet de mesurer l'irrégularité de la distribution. Soit \hat{F}_n la fonction de répartition empirique des points x^0, \dots, x^{n-1} , on peut définir la discrédance par ([8]) :

$$D_n(X) = \sup_{x \in [0,1]} |\hat{F}_n - F_U(x)| \quad (1.5)$$

avec F_U la fonction de répartition de la loi uniforme sur $[0, 1]$.

Remarque 4. $D_n(X)$ est la statistique de Kolmogorov-Smirnov permettant de tester l'adéquation à la loi uniforme.

Definition 5. Soit X un espace compact et μ une probabilité régulière sur la tribu borélienne de X . Une suite $(X_n)_{n \in \mathbb{N}}$ de points de X est uniformément distribuée si pour tout $f \in \mathcal{C}(X)$,

$$\frac{1}{n} \sum_{k=1}^n f(x_k) \xrightarrow[n \rightarrow \infty]{} \int_X f d\mu \quad (1.6)$$

Remarque 5. C'est la loi forte des grands nombres qui assure presque sûrement cette convergence.

Une propriété importante, ici, est la suivante.

Propriété 3. Une suite (X_n) est uniformément distribuée si $\lim_{n \rightarrow \infty} D_n(x) = 0$.

Il existe de nombreux résultats de majoration de la discrédance. Commençons par énoncer le résultat le plus fameux, connu sous le nom d'inégalité de Koksma-Hlawka.

Théorème 1 (Hlawka, 1961 [9]). Si f est une fonction à variation $V(f)$ bornée au sens de Hardy et Krause, alors pour toute suite de points x_1, \dots, x_n sur $[0, 1]^d$, on a

$$\left| \frac{1}{n} \sum_{i=1}^n f(x^i) - \int_{[0,1]^d} f(t) dt \right| \leq V(f) D_n(x) \quad (1.7)$$

Ainsi, l'erreur d'approximation précédente est, dans le pire des cas, égale au produit de la variation $V(f)$ (une grandeur qui ne reflète que l'irrégularité de la fonction f) et de la discrédance $D_n(x)$ (qui mesure uniquement la qualité de la répartition de la suite).

Remarque 6. Les suites discutées ici seront finies et indicées par convention (pour contenir l'origine du domaine) de 0 à $n - 1$

Il est possible de construire des suites pour lesquelles la discrédance est plus faible que celle d'une suite aléatoire égale à $\frac{1}{\sqrt{n}}$. On les appelle des suites à faible discrédance. Ces suites ont la particularité de remplir le cube unité uniformément et de manière extrêmement régulière.

Des exemples de suites à faible discrédance sont données par Halton (1960) [10], Hammersley (1960) [11], Sobol' (1967) [12], Faure (1982) [13] et Niederreiter (1987) [14]. Dans les sections suivantes, nous étudierons la construction et les propriétés de ces différentes suites utilisées en planification d'expériences.

Une notion importante sur laquelle repose la construction de la plupart de ces suites est la définition de la fonction radicale inverse en base b donnée ci-dessous.

Toutes les suites données ci-après sont définies pour tout n . Nous verrons que la plupart de ces suites sont intéressantes vis-à-vis de l'aspect itératif. Par exemple, lorsque l'on souhaite rajouter q points à un plan de taille n . C'est notamment cet aspect qui va influencer le choix de la suite à utiliser.

Definition 6. Soit b un entier ≥ 2 . La fonction radicale inverse en base b est donnée par

$$\phi_b(i) = \frac{p_0}{b} + \frac{p_1}{b^2} + \dots + \frac{p_m}{b^m + 1} \quad (1.8)$$

$$\text{avec } i = p_0 + p_1b + \dots + p_mb^m \text{ et } m = \begin{cases} 1 + \lfloor \log_b(i) \rfloor & \text{si } i \neq 0 \\ 1 & \text{sinon} \end{cases}$$

La suite $C_b = \{x^0; x^1; \dots, x^{n-1}\}$ avec $x^i = \phi_b(i)$ est appelée suite de Van Der Corput en base b .

1.3.1 Construction d'une suite à discrédance faible

Suites de Van der Corput

Soit un entier $b \geq 2$. Pour $i \geq 1$, on définit le i ème terme de la suite de Van der Corput en base b comme le réel défini comme suit :

$$C_i^{(b)} = \sum_{k=1}^K \frac{d_k^{(b)}(i)}{b^k}$$

où $d_K^{(b)}(i) \dots d_2^{(b)}(i)d_1^{(b)}(i)$ est l'écriture de i en base b .

Autrement dit, $C_i^{(b)}$ est obtenu en renversant l'écriture en base b de i , en incluant le séparateur "décimal". Par exemple, les premiers termes de la suite de Van der Corput en base 10 sont :

$$0.1, 0.2, 0.3, \dots, 0.9, 0.01, 0.11, 0.21, 0.31 \dots$$

Suites de Halton

La construction de Halton se base sur les suites de Van der Corput. On se donne b_1, \dots, b_d un ensemble de d entiers premiers entre eux, et le i ème terme de la suite de Halton est le point de $[0; 1]^d$ défini par :

$$x_i = (C_i^{(b_1)}, C_i^{(b_2)}, \dots, C_i^{(b_d)})$$

Les suites de Halton sont la version en dimension $d \geq 1$ des suites de Van Der Corput qui quant à elles constituent la version unidimensionnelle de ces suites. L'idée pour générer les suites de Halton consiste à utiliser une base différente pour chaque dimension.

Par ailleurs il existe bien d'autres constructions de suites à discrédance faible : Hammersley, Faure, Sobol, ...

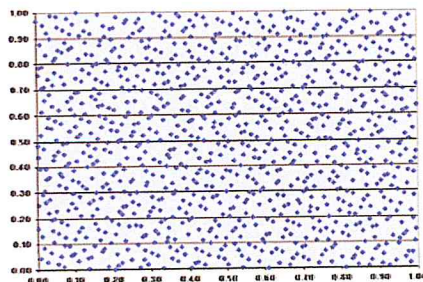


FIGURE 1.8 – premiers points de la suite de Halton pour $b_1 = 2$ et $b_2 = 3$.

1.4 Plans optimisés

Pour assigner à un plan d'expérience les propriétés qui nous intéressent, il est judicieux de développer des stratégies d'optimisations d'un plan initial selon les critères d'intérêt. La méthode usuelle consiste à se placer dans le cadre des plans à projections factorielles uniformes et à optimiser l'un des critères décrits ci-dessus (discrédance ou Maximin par exemple). De nombreuses études se sont penchées sur la manière d'optimiser les plans, par exemple par des algorithmes de recuit simulé. On citera ici uniquement les deux solutions les plus classiques qui consternent toutes deux la classe des hypercubes latins :

- On maximise le critère mindist. Ces plans sont appelés les LHS maximin,
- On minimise la discrédance L_2 . Ces plans sont appelés les LHS à discrédance faibles.

1.5 Critères de comparaison de plans d'expériences

Afin de comparer les plans générés et faire notre choix sur les expériences à réaliser pour la campagne, il est nécessaire de faire appel à des critères usuels permettant de

juger du bon remplissage de l'espace ainsi que de la distribution uniforme.

La qualité de la répartition spatiale est mesurée soit à l'aide de critères déterministes comme les distances minimax ou maximin [15], soit à l'aide de critères statistiques comme la discrédance (Niederreiter [16], Hickernell [17] et Fang et al. [18]). Les trois critères retenus sont la mesure de recouvrement, distance entre points (le critère maximin et minmax) et la discrédance en norme L2.

1.5.1 Mesure de recouvrement (Rec)

La mesure de recouvrement permet de mesurer l'écart entre les points du plan et ceux d'une grille régulière (Gunzburger et al., [19]), et est définie par

$$Rec = \frac{1}{\gamma} \left(\frac{1}{n} \sum_{i=1}^n (\gamma_i - \bar{\gamma}_i) \right)^{\frac{1}{2}}, \text{ avec } \gamma_i = \min_{k \neq i} \rho_2(x_i, x_k) \text{ et } \bar{\gamma} = \frac{1}{n} \sum_{i=1}^n \gamma_i \quad (1.9)$$

où ρ_2 est la distance euclidienne, définie par l'équation 1.14 avec $\rho = 2$. γ_i désigne donc la distance (euclidienne) minimale entre le point x_i et les autres points du plan et $\bar{\gamma}$, la moyenne des γ_i . Cette expression fait apparaître explicitement le coefficient de variation de l'échantillon des γ_i , c'est-à-dire le rapport de l'écart-type avec la moyenne.

Ce critère est nul pour une grille régulière ($\gamma_1 = \dots = \gamma_n = \bar{\gamma}$). L'objectif est donc de le minimiser pour se rapprocher d'une grille régulière, et ainsi assurer le remplissage de l'espace, sans toutefois l'atteindre pour respecter une distribution uniforme notamment en projection sur les axes.

1.5.2 Distance entre points (minimax et maximin)

Johnson et al., [20], ont introduit les distances maximin et minimax afin de construire des plans répondant à la question de remplissage de l'espace. Dans [21], Johnson introduit deux critères de distance entre points. Ces critères permettent d'évaluer si la distribution des points d'un plan d'expériences $\mathcal{P} = \{X_i\}_{i=1}^n$ est proche de celle d'une grille régulière ou non. Le premier critère appelé critère minimax, est défini par :

$$\phi_{mM}^p(\mathcal{P}) = \max_{v \in [0,1]^d} \min_{X_i \in \mathcal{P}} \|v - X_i\|_p \quad (1.10)$$

où $\|\cdot\|$ désigne la norme L^p dans $[0, 1]^d$:

$$\|v\|_p = \left(\sum_{j=1}^d v_j^p \right)^{\frac{1}{p}}$$

Le critère minimax est appelé critère de dispersion par Niederreiter [14]. Ce critère est le plus couramment défini pour $p = 2$ et l'on abrège le critère par ϕ_{mM} . Une petite valeur de $\phi_{mM}^p(\mathcal{P})$ signifie qu'il n'existe pas de point v dans $[0, 1]^d$ qui ne soit pas proche d'un point X_i de l'ensemble \mathcal{P} . L'utilisation de ce critère requiert le calcul de toutes les distances entre les points v du domaine $[0, 1]^d$ et les points X_i de \mathcal{P} . En pratique, ce

critère est approximé à l'aide d'une discrétisation fine du domaine $[0, 1]^d$, les points issus de la discrétisation substituant v dans (1.10). Dans [22], Pronzato et Müller propose un calcul exact de $\phi_{mM}^p(\mathcal{P})$ à partir de la triangulation de Delauney de l'ensemble de points \mathcal{P} . Ces deux dernières approches (approximation et calcul exact) ne sont cependant envisageables que pour des petites valeurs de d .

Le second critère, appelé critère maximin, est défini par :

$$\phi_{Mm}^p(\mathcal{P}) = \min_{(X_i, X_j) \in \mathcal{P}^2} \|X_i - X_j\| \tag{1.11}$$

Le critère maximin est également le plus souvent défini pour $p = 2$ et l'on abrège ce critère par ϕ_{Mm} . Ce critère ne fait appel qu'aux points du plan \mathcal{P} et évalue le minimum des distances entre toutes les paires de points de \mathcal{P} . Si ce minimum vaut c , tous les points de \mathcal{P} sont espacés les uns des autres d'une distance d'au moins c . Ainsi, plus la valeur de ϕ_{Mm} est grande, meilleur est le remplissage de l'espace par \mathcal{P} .

1.5.3 Discrédance en norme L2 (DisL2)

Avant de présenter le critère utilisé, nous introduisons la notion de discrédance. La discrédance est une mesure de l'écart existant entre une situation de référence (l'uniformité théorique) et une configuration donnée. Soient n points $x = (x_1, \dots, x_n)$ dans $[0, 1]^d$, $x_i = (x_{i1}, \dots, x_{id})$ pour $i = 1, \dots, n$ et $t = (t_1, \dots, t_s) \in [0, 1]^d$. La discrédance locale, notée $disc$, est définie par

$$disc(t) = \frac{A(t, x)}{n} - \prod_{j=1}^d t_j. \tag{1.12}$$

où $A(t, x)$ est le nombre de points de x qui appartiennent au pavé $P = [0, t_1[\times \dots \times [0, t_d[$. La figure 1.9 nous permet d'illustrer la notion de discrédance en dimension 2. Pour que

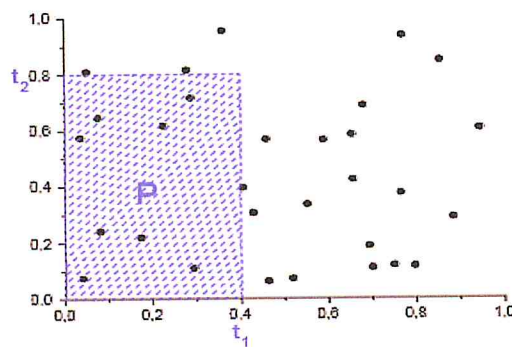


FIGURE 1.9 – Représentation d'une suite de $n = 30$ points en dimension $d = 2$ et du pavé $P = [0, t_1[\times [0, t_2[$.

la répartition de notre ensemble x se rapproche de l'uniformité, on s'attend à ce que la

proportion de points situés dans le pavé P soit proche de son aire $\lambda(P) = \prod_{j=1}^2 t_j$. Par conséquent, il semble naturel de juger peu uniforme une suite finie pour laquelle il est possible d'exhiber un pavé avec une grande différence entre les deux, c'est-à-dire une importante discrédance locale. Cette idée exprime la notion d'équirépartition.

Une suite est équirépartie lorsque sa fonction de répartition empirique converge vers la fonction de répartition uniforme. On voit ainsi qu'une suite de variables aléatoires uniformes indépendantes est équirépartie. La définition de discrédance découle directement de cette notion.

Soient F_n , la fonction de répartition empirique des points du plan et U celle de la loi uniforme, on définit la discrédance par,

$$\sup_{t \in [0,1]^d} |F_n(t) - U(t)| \quad \text{avec} \quad F_n(t) = \frac{1}{n} \sum_{i=1}^n \mathbb{1}_{x_i \leq t} \quad (1.13)$$

Contrairement aux deux critères précédents, la discrédance n'est pas basée sur la distance entre les points. Il existe différentes mesures de discrédance (Fang et al., [23] et Thiernard [24]).

1.6 Plans maximin et minimax

Les plans basés sur un critère de distance sont relativement courants. Ce critère permet de quantifier la répartition d'un ensemble de points. Parmi eux, les plus célèbres sont les plans maximin et minimax introduits par Johnson et al., [27].

Introduisons, tout d'abord, quelques notations : soit D le domaine expérimental, et $S \subset D \subset \mathbb{R}^d$ un plan consistant en n points distincts (x_1, \dots, x_n) . On se donne une distance ρ sur D associée à la norme L_p :

$$\rho_p(w, x) = \left(\sum_{i=1}^d \|w_i - x_i\|^p \right)^{\frac{1}{p}} \quad (1.14)$$

D'autres critères de distance existent, notamment le critère de distance moyenne, pour la présentation duquel nous renvoyons le lecteur à l'ouvrage de Santner et al., [28].

1.6.1 Plans maximin

Pour les plans maximin, l'approche est la suivante : on considère que les points d'un plan sont "bien répartis", si les points sont aussi éloignés les uns des autres que possible.

Une façon de quantifier la proximité des points de l'ensemble S , est de prendre la plus petite distance entre deux points quelconques de S , soit

$$\min_{w, x \in S} \rho_p(w, x) \quad (1.15)$$

Un plan qui maximise cette quantité est appelé plan de distance maximin, et est noté S^* , et

$$\min_{w,x \in S^*} \rho_p(w,x) = \max_{S \subset D} \min_{w,x \in S} \rho_p(w,x) \quad (1.16)$$

Ce type de plan garantit que deux points quelconques ne sont pas trop proches, et donc que les points du plan sont bien répartis dans D .

Typiquement, ces plans ont tendance à contenir des points sur le bord du domaine expérimental. Pour des illustrations de ce phénomène, nous renvoyons le lecteur à l'article de base de Johnson et al., [28].

Une façon pratique de construire un plan maximin approché est de se focaliser sur les hypercubes latins. Chaque arête du domaine expérimental est divisé en n segments de même longueur de façon à obtenir un maillage du domaine. Un hypercube latin sélectionne alors n points parmi les n^d points de la grille de façon à ce que les n niveaux des variables d'entrée soient testés une fois par les simulations. Les hypercubes latins présentent beaucoup d'avantages :

1. Ils sont simples à construire. En effet, chaque colonne d'un hypercube est une permutation de $\{1, \dots, n\}$ ou de n symboles quelconques.
2. Les points sont uniformément distribués sur chaque axe du domaine.

La distribution uniforme sur chaque axe n'assure pas l'uniformité sur le domaine expérimental. Cependant, pour n fixé, ils existent $(n!)^d$ hypercubes latins possibles. La démarche est donc la suivante : on part de l'ensemble des hypercubes latins à n points et on choisit celui (ou un de ceux) qui satisfait le critère maximin, appelé hypercube latin maximin. Park [29] propose également un algorithme d'échange pour déterminer un hypercube optimal (localement) pour un critère donné.

Enfin, il est intéressant de préciser que le problème de construction d'un plan maximin à n points dans $[0, 1]^d$ est équivalent au problème de placer dans $[0, 1]^d$ les centres de n sphères, disjointes ou d'intersection égale à un singleton, de même rayon r , de façon à ce que r soit le plus grand possible (Bettinger, [30]).

Cette vision permet de faire le lien avec les plans de Strauss (Strauss [31], Ripley et Kelly [32], Franco [33]). Ces derniers sont similaires aux plans maximin, puisqu'ils reposent sur le processus ponctuel de Strauss, et dont le principe est de remplir l'espace de particules (points dans le domaine représentant les expériences) en minimisant les interactions entre elles, chacune ayant une sphère d'influence de rayon $r/2$.

Cependant, ces plans nécessitent la mise en place de méthodes de Monte-Carlo par chaîne de Markov, qui peuvent être coûteuses en temps de calcul. De plus, l'intérêt principal de ces plans est de pouvoir incorporer une connaissance a priori, par exemple sur les liens entre les entrées ou sur des zones de densité plus forte, connaissance dont nous ne disposons pas pour notre cas.

1.6.2 Plans minimax

Pour les plans minimax, l'approche est différente : on considère qu'un plan S est bien réparti dans D , si tout point de l'espace D est suffisamment proche d'un point de S .

Formellement, définissons la distance entre un point quelconque $x \in D$, et le plan $S \subset D$, par :

$$\rho_p(x, S) = \min_{x_i \in S} \rho_p(x, x_i) \quad (1.17)$$

Un plan est appelé plan de distance minimax, et noté S^* , si la distance maximale, entre un point $x \in D$ et le plan S^* , est minimale parmi tous les plans (à n points) $S \subset D$:

$$\max_{x \in D} \rho_p(x, S^*) = \min_{S \subset D} \max_{x \in D} \rho_p(x, S) \quad (1.18)$$

L'inconvénient de ce type de plan est qu'ils sont difficiles à générer. En effet, la fonction utilisée pour la construction de ces plans, utilise non seulement l'ensemble des points candidats pour le plan S , mais aussi l'ensemble des points de tout le domaine expérimental D .

Dans ce second type de plan, le problème de construction d'un plan minimax à n points dans $[0, 1]^d$ est équivalent au problème de placer dans $[0, 1]^d$ les centres de n sphères de même rayon r qui recouvrent D , de façon à ce que r soit le plus petit possible (Bettinger, [35]).

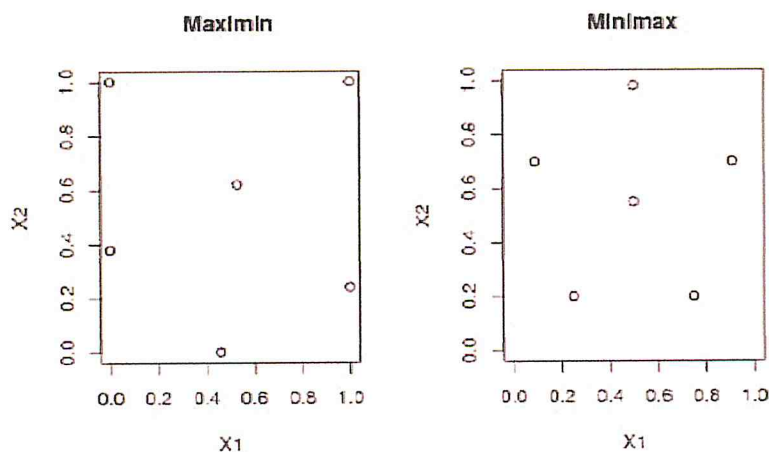


FIGURE 1.10 – Plans maximin et minimax de $N=6$ points dans $[0, 1]^2$.

CHAPITRE

2

PROCESSUS PONCTUELS SPATIAUX

Ce chapitre a pour but de présenter quelques éléments de la théorie des processus ponctuels, ainsi que quelques exemples illustratifs.

2.1 Introduction et motivations

De nombreuses expériences, issues de différents domaines, fournissent des données ponctuelles. Par exemple, en démographie, lorsqu'on s'intéresse à la répartition de la population humaine sur le globe, on peut à l'aide d'outils de recensement et d'outils statistiques fournir des résultats sous forme ponctuelle. Sur l'image suivante, chaque point représente 500 000 habitants.



FIGURE 2.1 – Répartition de la population humaine sur la Terre en 2005.

L'image suivante, devenue célèbre pour sa beauté et son originalité - photographier par satellite la Terre pendant la nuit - présente aussi la caractéristique de donner une répartition de points (blancs) associés aux sources de lumière créées par l'homme.



FIGURE 2.2 – La Terre vue de l'espace la nuit.

Ceci donne une idée assez vague de la répartition de la population humaine en termes de villes - les points lumineux correspondent aux rayonnements des villes dans lesquelles les hommes sont concentrés.

On peut par exemple noter que les États Unis sont très lumineux pour 300 millions d'habitants comparé à la Chine et ses 1 307 millions d'habitants, et que l'Afrique ou des pays comme la Corée du Nord sont "éteints".

Sur le plan ponctuel, on remarque que les points sont rarement isolés et semblent s'attirer. Ils forment des taches, poreuses par endroits (Europe, Ouest des États Unis, ...), à cavité (Sud de l'Afrique, Australie, Asie Centrale, ...), Éclatées (Afrique Centrale, Nord de l'Amérique du Sud, Alaska, ...).

La démographie n'est pas le seul domaine dans lequel les points fourmillent. La géographie ainsi que les matières qui s'y rapportent (géologie, agriculture, sismologie ...) fournissent une multitude de données ponctuelles.

2.2 Définitions et notations

Comme le nom processus ponctuels l'indique, l'origine de la théorie était l'étude de séquences de points aléatoires sur l'échelle du temps. De tels processus jouent toujours un rôle important, par exemple pour modéliser les queues ou files d'attente, ou dans le

domaine des télécommunications. Aujourd'hui cependant, les processus ponctuels spatiaux semblent être le domaine dominant, et parfois même des méthodes de statistiques de processus ponctuels spatiaux sont converties dans le domaine temporel.

A partir de maintenant, nous emploierons sans distinction les expressions de processus spatiaux, de processus ponctuels, de processus ponctuels marqués ou de processus objets pour parler des mêmes objets mathématiques.

2.2.1 Espace des configurations

Soit χ un espace donné, muni d'une métrique d tel que (χ, d) soit complet et séparable (bien souvent \mathbb{R}^d muni de la distance euclidienne). On appelle point tout élément $x \in \chi$, et on s'intéresse aux ensembles de points (figure (2.3)).

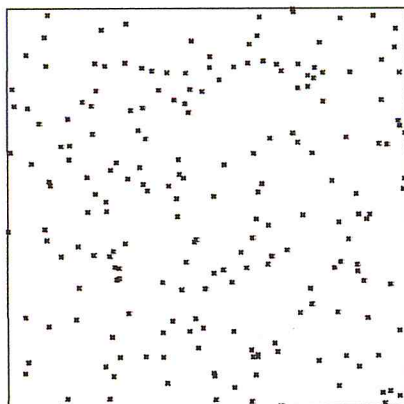


FIGURE 2.3 – Ensemble de points, ou configuration, de $\chi = [0, 1] \times [0, 1]$.

Definition 7. On appelle configuration et on note x un ensemble dénombrable, non ordonné de points de χ :

$$x = \{x_1, \dots, x_n\}, n \in \mathbb{N}^*$$

Definition 8. On dit d'une configuration $x \subseteq \chi$ qu'elle est localement finie si dans tout borélien borné $A \subseteq \chi$ elle place un nombre $N_x(A)$ fini de points.

La famille de toutes les configurations localement finies sera notée N^{lf} .

Par la suite, on s'intéressera aux configurations x dites localement finies (espace N^{lf}), ie qui placent dans tout borélien borné $A \subseteq \chi$ un nombre $N_x(A)$ fini de points, et simples, ie tous les points $x_i \in x$ sont distincts. De plus, on se limitera à l'étude des processus définis sur des régions bornées, du fait de notre application à la recherche d'objets dans les images (ensemble borné, et objets définis sur un compact). L'espace des configurations finies et simples sera noté N^f .

On équipe alors χ d'une mesure borélienne localement finie notée $\nu(\cdot)$, en général la mesure de Lebesgue (notée $\Lambda(\cdot)$ par la suite), et on note par extension $\nu^n(\cdot)$ la mesure produit sur χ^n . On définit, ensuite, les sous ensembles de N^f de la façon suivante :

$$N_n^f = \{x \in N^f : N_x(\chi) = n\}.$$

Une des particularités de chaque élément de N^f est de contenir un ensemble de points non ordonnés. Ainsi, la mesure d'un tel espace limité à n points, N_n^f , est $\nu(\chi)^n/n!$, le facteur $n!$ venant du fait que χ^n est ordonné tandis que N^f ne l'est pas. On a donc :

$$\nu(N^f) = \sum_{n=0}^{\infty} \frac{\nu(\chi)^n}{n!} = e^{\nu(\chi)} \quad (2.1)$$

2.2.2 Notions de voisinage et clique

Definition 9 (voisin). Soit \sim une relation binaire symétrique et réflexive sur (χ, d) - dite relation de voisinage. Soit u et v deux points de χ . On dit que u et v sont voisins si $u \sim v$.

Exemple 1 (La relation de portée fixe). Soit $R > 0$. On définit la relation de portée R par : $u \sim v \Leftrightarrow d(u, v) < R$. En d'autres termes, les boules $B(u, \frac{R}{2})$ et $B(v, \frac{R}{2})$ sont d'intersection vide.

Definition 10 (voisinage). Soit A un ensemble. On définit le voisinage de A par

$$\partial(A) = \{x \in \chi : x \sim a, a \in A\}.$$

En particulier, le voisinage de $\{a\}$ contient tous les voisins de a .

Definition 11 (Clique). Soit \sim une relation de voisinage sur χ . Une configuration x est appelée clique si tous les points de x sont voisins les uns des autres : $\forall u, v \in x, u \sim v$. Par convention, \emptyset sera aussi une clique.

Exemple 2. Sur le schéma ci-dessous, la configuration possède 4 cliques non vides pour la relation de portée fixe $R > 0$.

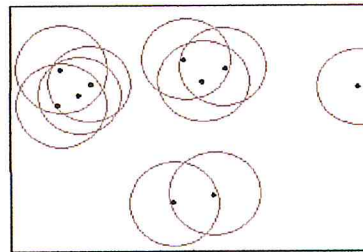


FIGURE 2.4 – configuration à 4 cliques.

2.2.3 Processus ponctuels

Les processus ponctuels (distributions spatiales aléatoires de points) sont le fondement de la géométrie stochastique. Des exemples de l'utilisation de ces processus stochastiques peuvent être observés dans des domaines aussi variés que la biologie, la mécanique, la physique des particules, l'ingénierie au sens large, etc.

On définit alors la notion de processus ponctuel :

Definition 12. *Un processus ponctuel sur χ est une application X d'un espace probabilisé (Ω, A, P) dans N^{lf} , telle que pour tout borélien $A \subseteq \chi$, $N(A) = N_X(A)$ est une variable aléatoire (finie).*

Ce qui induit une nouvelle définition :

Definition 13. *Si l'espace χ est borné ou si $N_X(\chi)$ est fini presque sûrement, le processus ponctuel est dit **processus ponctuel fini**.*

Les réalisations d'un processus ponctuel X sont donc des configurations aléatoires de points tels que pour tout borélien $A \subseteq \chi$ le nombre de points dans A soit une variable aléatoire. Cela signifie qu'un processus ponctuel est une variable aléatoire à valeur dans l'espace mesurable $(N^{lf}, \mathcal{N}^{lf})$, où \mathcal{N}^{lf} est la plus petite σ -algèbre telle que pour tout borélien borné $A \subseteq \chi$ l'application $x \rightarrow N_x(A)$ soit mesurable.

Pour obtenir une réalisation d'un processus ponctuel, il suffit alors, par exemple, d'utiliser une distribution de probabilités discrètes $(p_n)_{n \in \mathbb{N}}$ pour fixer le nombre de points de la configuration, ainsi qu'une famille de densités de probabilité symétriques $\{j_n(x_1, \dots, x_n)\}_{n \in \mathbb{N}}$ sur χ^n pour les positionner dans χ .

2.3 processus ponctuel de Poisson

Un exemple de ce type de processus, sans doute le plus naturel, est le processus de Poisson. Il traduit la notion d'uniformité dans l'espace N^f .

Definition 14. *Un processus ponctuel X sur χ est appelé processus ponctuel de Poisson de mesure d'intensité $\nu(\cdot)$ si :*

- (P1) $N_X(A)$ suit une loi de Poisson d'espérance $\nu(A)$ pour tout borélien borné $A \subseteq \chi$.
- (P2) Pour k boréliens disjoints A_1, \dots, A_k , les variables aléatoires $N_X(A_1) \dots N_X(A_k)$ sont indépendantes.

Le processus de Poisson est dit homogène si la mesure d'intensité $\nu(\cdot)$ est proportionnelle à la mesure de Lebesgue $\Lambda(\cdot)$. Dans ce cas, $\nu(\cdot)$ est l'intensité du processus. Il faut noter qu'il existe des processus non homogènes, où il est nécessaire de définir une fonction d'intensité $\lambda(\cdot) > 0$. Selon le théorème de Radon-Nikodym [34], l'intensité du processus devient :

$$\nu(A) = \int_A \lambda(x) \Lambda(x) dx < \infty \quad (2.2)$$

où $\lambda(\cdot)$ est la dérivée de Radon-Nikodym de $\nu(\cdot)$ par rapport à la mesure de Lebesgue du processus.

La mesure de probabilité $\pi_{\nu(\cdot)}$ d'un processus de Poisson d'intensité $\lambda(\cdot)$ peut s'écrire pour tout borélien A :

$$\pi_{\nu}(A) = e^{-\nu(X)} \left(1_{[\emptyset \in A]} + \sum_{n=0}^{\infty} \frac{\pi_{\nu_n}(A)}{n!} \right) \text{ avec } \pi_{\nu_n}(A) = \int_{\mathcal{X}} \dots \int_{\mathcal{X}} 1_{[\{x_1, \dots, x_n\} \in A]} \nu(dx_1) \dots \nu(dx_n)$$

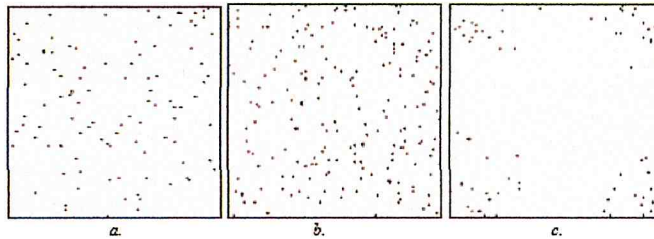


FIGURE 2.5 – Processus poisson : a. d'intensité $\lambda = 0,01$ b. d'intensité $\lambda = 0,02$ c. processus non-homogène.

Les processus de Poisson ne constituent pas un modèle capable de modéliser la répartition des objets dans une image par exemple. Mais leur simplicité leur permet d'être utilisés comme référence pour des modèles plus complexes. Les connaissances sur les objets et la configuration des objets peuvent être incorporées dans une densité de probabilité définie par rapport à un processus ponctuel de référence.

Il est possible de définir la densité d'un processus ponctuel par rapport à un processus de Poisson de référence de mesure $\pi_{\nu}(\cdot)$ à travers une fonction de densité positive et mesurable sur N^f notée f telle que :

$$\int_{N^f} f(X) d\pi_{\nu}(X) = 1 \quad (2.3)$$

La fonction f est nommée la dérivée de Radon-Nykodym, et elle est définie comme :

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

si μ et π sont les mesures des deux processus, π étant la densité du processus de référence.

La distribution du nombre des points dans un processus point défini par une fonction de densité f est :

$$p_n = \frac{e^{-\Lambda(x)}}{n!} \int_{\mathcal{X}} \dots \int_{\mathcal{X}} f(\{x_1, \dots, x_n\}) \nu d(x_1) \dots \nu d(x_n) \quad (2.4)$$

2.4 Processus ponctuels marqués

Ce type de processus ponctuels a été décrit par Daley dans [Daley et Jones, 1988][34].

Les processus ponctuels sont les exemples les plus simples de la géométrie stochastique, mais on a souvent besoin de simuler des processus impliquant des objets plus complexes. Pour décrire des objets au lieu de points, il est simplement nécessaire d'ajouter à chaque point une « marque » qui décrit les propriétés de l'objet (sa dimension, son orientation, sa forme, etc.). Ce processus est alors nommé « processus de points marqués » ou « processus objet » (Daley et Vere-Jones 2002) [35].

Généralement, on travaille avec des objets plutôt que des points. Les objets sont définis par leur position dans un espace de positions \mathcal{P} et par leurs marques, attributs géométriques par exemple, dans un espace de marques \mathcal{M} . On parle alors de processus ponctuels marqués .

Definition 15. *Un processus ponctuel marqué ou processus objet sur $\chi = P \times M$ est un processus ponctuel sur χ dont les positions des objets sont dans \mathcal{P} et les marques dans M , tel que le processus des points non marqués soit un processus ponctuel bien défini sur P .*

Un processus objet est une séquence aléatoire $X = \{x_n, m_n\}$ formé d'un processus de points x_n défini sur un sous-espace P de \mathbb{R}^d et des marques m_n correspondantes à chaque x_n dans l'espace des marques M .

Si nous cherchons à simplifier la définition, nous pouvons considérer le processus objet X avec les points dans P et les marques dans M comme un simple processus ponctuel sur $P \times M$. Pourtant, en pratique, il est souvent plus intéressant de séparer les points et les marques, pour pouvoir définir plus simplement des translations dans l'espace des points tout en laissant inchangées les marques attachées à chaque objet ou au contraire, pour avoir la possibilité de changer la forme de l'objet sans nécessairement le déplacer.

Un processus objet de Poisson homogène sur $P \times M$ est un processus objet pour lequel les points sont distribués selon une loi de Poisson dans l'espace P et pour lequel les marques sont distribuées de façon uniforme et indépendante sur l'espace M . Nous noterons Λ_p la loi d'intensité des points et λ_m la loi de distribution des marques. La mesure de Lebesgue du processus point défini sur $P \times M$ est :

$$\nu(A \times B) = \Lambda_p(A) \times \Lambda_m(B) = \int_A \lambda_p(x) dx \int_B \lambda_m(m) dm \quad (2.5)$$

pour A borélien de P et B borélien de M . La mesure de probabilité sur P de ce processus ponctuel est :

$$\pi_\nu(A) = \sum_{n=0}^{\infty} \frac{e^{-\lambda(p)}}{n!} \int_{P \times M} 1_A(\{z_1, \dots, z_n\}) d\nu^n(z_1, \dots, z_n) \quad (2.6)$$

2.5 Processus ponctuels de Markov

Une classe intéressante de processus ponctuels est celle des processus ponctuels de Markov. Ils regroupent les processus ponctuels finis définis par une densité pouvant s'écrire sous forme énergétique comme une somme de potentiels d'interactions. Ce sont les plus utilisés en traitement des images puisqu'ils permettent de modéliser les interactions entre les objets du processus, et qu'ils sont facilement programmables [36].

La définition d'un processus de Markov [Ripley et Kelly,1977] [37] est la suivante :

Definition 16. Soit (χ, d) un espace métrique complet et séparable, $\nu(\cdot)$ une mesure borélienne finie non atomique, et $\pi_\nu(\cdot)$ la loi d'un processus ponctuel de Poisson de mesure d'intensité $\nu(\cdot)$. Soit X un processus ponctuel sur χ défini par sa densité $f(\cdot)$ par rapport à $\pi_\nu(\cdot)$.

Alors X est un processus ponctuel de Markov sous la relation symétrique et réflexive \sim sur χ si, pour toute configuration $x \in N^f$ telle que $f(x) > 0$.

- (a). $f(y) > 0$ pour tout $y \subseteq x$, ie tous les objets de y sont dans x (condition d'hérédité).
- (b). pour tout $u \in \chi$, $\frac{f(x \cup u)}{f(x)}$ ne dépend que de u et de son voisinage $\partial(\{u\}) \cap x = \{x \in x : u \sim x\}$.

Remarque 7. Les hypothèses qui portent sur $\nu(\cdot)$ font de X un processus fini et simple. La condition (a) signifie que si une configuration peut se produire, alors toutes les sous-configurations quelle contient peuvent se produire aussi. La condition (b) exprime une propriété de Markov locale : le comportement d'un point u par rapport à la configuration entière ne dépend que des ses proches voisins dans cette configuration.

On appelle alors clique une configuration dont tous les points sont voisins les uns des autres par rapport à une relation symétrique et réflexive notée \sim . L'ordre d'une clique est le nombre d'objets qu'elle contient. Le théorème suivant, équivalent de celui d'Hammersley-Clifford pour les processus ponctuels, permet d'exprimer la densité d'un processus de Markov sous une forme énergétique décomposée sur les cliques de la configuration x .

Théorème 2 (Hammersley - Clifford). Une densité de processus ponctuel $f : N^f \rightarrow [0, \infty[$ est markovienne sous une relation de voisinage \sim si et seulement si il existe une fonction mesurable $\phi : N^f \rightarrow [0, +\infty[$ telle que :

$$f(x) = \alpha \prod_{\text{cliques } y \subseteq x} \phi(y) \quad (2.7)$$

pour tout $x \in N^f$.

Exemple (Processus de Strauss)

Un des processus de Markov les plus connus est le processus de Strauss [Strauss,1975]. Sa densité par rapport à un processus ponctuel de Poisson de loi $\pi_\nu(\cdot)$ s'écrit :

$$f(x) = \alpha \gamma^{s(x)} \quad (2.8)$$

où $\beta > 0$, $\gamma \geq 0$, $n(x) = N_X(\chi)$ ($n(x)$ représente le nombre de points d'une configuration x) et $s(x)$ représente le nombre de cliques d'ordre 2 en relation par rapport à \sim : $u \sim v \Leftrightarrow d(u, v) < R$ ($s(x)$ représente le nombre de paires de points en relation dans la configuration x). Tout d'abord, notons que le paramètre β est un paramètre d'échelle, qui corrige l'intensité du processus de référence, puisque celle-ci vaut désormais $\beta\lambda(\cdot)$. D'autre part, le processus répond différemment selon les valeurs de γ :

1. si $\gamma = 1$, le processus est simplement un processus ponctuel de Poisson d'intensité $\beta\lambda(\cdot)$.
2. si $\gamma \in]0, 1[$, le processus induit une répulsion entre les points proches au sens de la relation \sim .
3. si $\gamma = 0$, la densité interdit d'avoir deux points de la configuration qui soient voisins au sens de la relation \sim ("hard core process").

CHAPITRE

3

LA SIMULATION PAR MÉTHODES DE MONTE CARLO PAR CHAÎNE DE MARKOV

3.1 Chaînes de Markov

3.1.1 Premières définitions

On appelle chaîne de Markov à temps discret toute suite de variables aléatoires $\{X_n\}_{n \geq 0}$, provenant d'un certain support S et respectant la propriété markovienne :

$$\mathbb{P}(X_n | X_{n-1}, \dots, X_0) = \mathbb{P}(X_n | X_{n-1}).$$

Autrement dit, la valeur d'une variable aléatoire de cette suite ne dépend que de celle qui la précède. Nous nous intéresserons ici aux chaînes dites homogènes, où les probabilités $\mathbb{P}(X_n | X_{n-1})$ sont indépendantes de la valeur de n . Ainsi, une chaîne de Markov homogène est complètement définie par la valeur ou la distribution de X_0 et les probabilités de transition en un pas $p_{ij} = \mathbb{P}(X_n = j | X_{n-1} = i)$, $n \geq 1$, habituellement regroupées dans une matrice $P = \{p_{ij}\}$, définie sur l'ensemble des valeurs $i, j \in S$ (l'espace d'états). Finalement, on définit les probabilités de transition en k pas ainsi :

$$p_{ij}^{(k)} = \mathbb{P}(X_k = j | X_0 = i)$$

3.1.2 Quelques critères de classification

Pour qu'une chaîne ait des propriétés asymptotiques intéressantes, on souhaitera que tous ses états soient récurrents positifs et apériodiques. On dira alors que la chaîne est ergodique. Un état i est dit récurrent positif si

$$p_{ii}^n \not\rightarrow 0, \text{ quand } n \rightarrow \infty.$$

Un état est apériodique si sa période vaut 1, la période étant définie comme étant le plus grand commun diviseur de $\{n \in \mathbb{N} : p_{ii}^{(n)} > 0\}$. Pour vérifier ces propriétés, il sera plus simple de passer par la notion de classe d'états, que nous définissons maintenant. On dit qu'un état j est accessible à partir d'un état i , s'il est possible, en partant de i , d'atteindre l'état j en un nombre fini de transitions. Formellement,

$$i \rightarrow j \Leftrightarrow \exists n \geq 0, \text{ tel que } p_{ii}^{(n)} > 0.$$

Si l'on a à la fois $i \rightarrow j$ et $j \rightarrow i$, on dira que i et j communiquent. On montre alors facilement que les états communiquant entre eux forment une classe d'équivalence sur l'ensemble des états de la chaîne. Les états d'une même classe possèdent plusieurs propriétés communes. Entres autres, ils auront la même période et seront tous récurrents positifs si l'un d'entre eux l'est.

Remarque 8. Dans le cadre de ce projet, nous verrons uniquement des chaînes irréductibles, c'est-à-dire des chaînes ne possédant qu'une seule classe d'états.

3.1.3 Théorème ergodique et distribution stationnaire

La distribution stationnaire, lorsqu'elle existe est une distribution de probabilités prenant valeur sur S et respectant la condition :

$$\pi = \pi P \Leftrightarrow \forall j : \pi_j = \sum_i \pi_i p_{ij}.$$

Dans le cas d'une chaîne ergodique, la distribution stationnaire existera toujours, et on aura alors, pour tous les états $i, j \in S$:

$$p_{ij}^{(n)} \xrightarrow{n \rightarrow \infty} \pi_j > 0.$$

Cette distribution et ses propriétés seront primordiales pour les développements qui vont suivre. Mentionnons seulement pour l'instant qu'une chaîne de Markov ergodique convergera en loi vers sa distribution stationnaire peu importe la valeur initiale X_0 .

3.1.4 La convergence des chaînes de Markov

Avant de présenter les méthodes de simulation des chaînes MCMC, nous présentons les conditions nécessaires pour qu'une chaîne puisse converger et atteindre la distribution recherchée, la stationnarité ne pouvant pas garantir à elle seule la convergence de la chaîne.

La chaîne est convergente vers la distribution invariante π si :

$$X \xrightarrow[n \rightarrow \infty]{} \pi \quad (3.1)$$

donc il est nécessaire de pouvoir s'assurer de la convergence de la chaîne vers sa distribution finale. L'équation peut être écrite comme :

$$P^t(x, A) \xrightarrow[t \rightarrow \infty]{} \pi(A) \quad (3.2)$$

où $P^t(x, A) = p(X_t \in A / X_0 = x)$ est un noyau de transition de pas t .

Pour assurer la convergence d'un processus il faut vérifier que la chaîne puisse atteindre la distribution invariante à partir de n'importe quel état initial. Pour cela, elle doit satisfaire trois conditions supplémentaires, elle doit être :

- a. Irréductible.
- b. Apériodique.
- c. Ergodique.

3.1.5 L'échantillonnage des chaînes de Markov

Dans le but de simuler une configuration d'objets dans une image par un processus objet nous allons construire une chaîne de type Monte Carlo. Il existe plusieurs algorithmes pour la simulation d'une telle chaîne ; nous allons en présenter quelques uns dans cette section.

Nous pouvons distinguer deux types d'algorithmes pour échantillonner les chaînes MCMC, selon le type d'application, il est en effet possible d'utiliser des chaînes en temps continu ou des processus discrets utilisés préférentiellement pour simuler des processus spatiaux [?].

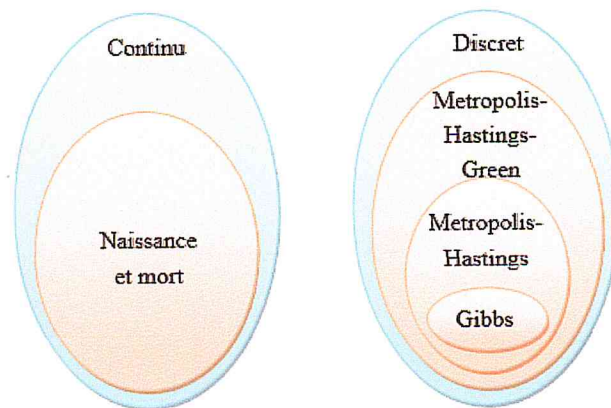


FIGURE 3.1 – Les familles d'algorithmes pour la simulation des chaînes MCMC

1. Les chaînes en temps continu : la méthode la plus importante est la méthode de naissance-mort [41].
2. Les chaînes discrètes : ces méthodes se basent en général sur la méthode Metropolis-Hastings [42]. Cet algorithme a été généralisé à travers l'algorithme Metropolis-Hastings-Green [43]. Un autre algorithme classique permettant de simuler les processus discrets également basé sur l'algorithme Metropolis-Hastings est l'échantillonneur de Gibbs, décrit dans [44].

Le choix d'un algorithme peut être effectué en fonction de ses performances [45] :

- Convergence : la vitesse de la chaîne (temps pour les chaînes continues ou par pas pour les chaînes discrètes) pour atteindre la distribution d'équilibre.
- Efficacité : la précision de l'estimation de la distribution recherchée.

D'autres critères peuvent être considérés :

- Choix entre l'utilisation d'une chaîne longue ou de plusieurs chaînes courtes : on peut estimer la moyenne s_n pour une longue chaîne ou considérer la moyenne de plusieurs chaînes courtes indépendantes.
- La simplicité d'implantation ainsi que le coût de calcul pour la simulation de la chaîne.

3.2 Les méthodes MCMC

Les méthodes de Monte-Carlo par chaînes de Markov permettent d'élargir grandement l'éventail des distributions pouvant être simulées numériquement. Elles sont relativement simples à implémenter et ne requièrent souvent que la connaissance de la fonction de densité cible à une constante près, ce qui les rend intéressantes dans de nombreuses situations. Cependant, une implémentation naïve peut mener à des temps de calcul très longs, puisque la convergence de ces méthodes est relativement lente lorsqu'elles ne sont pas bien calibrées à une situation donnée.

L'idée de base est de simuler une distribution de densité f en utilisant une chaîne de Markov ergodique $\{X_t\}$ dont la distribution stationnaire est f . Le théorème ergodique garantit alors la convergence en loi de $\{X_t\}$ vers une variable de densité f .

Nous appellerons MCMC toute méthode permettant de simuler une distribution en utilisant une chaîne de Markov ergodique ayant celle-ci comme distribution stationnaire. Pour construire un tel algorithme, il faut donc déterminer un ensemble de probabilités de transition P approprié, c'est-à-dire irréductible, ergodique et ayant la bonne distribution stationnaire.

3.2.1 Chaîne de Markov Monte-Carlo (MCMC)

Le terme "méthode de Monte-Carlo" (suggéré par John von Neumann et S. M. Ulam, dans les années 1940) fait référence à la simulation de procédés, en utilisant des nombres aléatoires. Le terme Monte-Carlo (ville réputée pour ses casinos) provient du fait que des "nombres aléatoires" (c'est-à-dire, les méthodes de simulation de Monte Carlo) étaient utilisés pour résoudre certaines des intégrales des équations complexes impliquées dans la conception des premières bombes nucléaires (intégrales des quantum dynamiques). Les intégrales de ces distributions (complexes) peuvent être estimées à partir des données (générées) en générant des grands ensembles d'échantillons de nombres aléatoires à partir, par exemple, de mélanges de distributions .

Definition 17. *On appelle algorithme MCMC (pour Monte Carlo Markov Chain) toute méthode produisant une chaîne de Markov $(X_{(n)})$ ergodique de loi stationnaire la distribution cible.*

3.2.2 L'algorithme de type naissance-mort

Les processus de naissance-mort sont des processus de Markov en temps continu. Ils sont utilisés pour la modélisation des processus ponctuels qui évoluent en espace et en temps. Les transitions entre états sont réalisées à travers la naissance d'un nouvel objet ou par le mort d'un objet existant. Ce sont des processus de Markov dans la mesure où l'état actuel du processus dépend seulement de l'état antérieur.

L'algorithme naissance-mort prévoit les changements d'état (translations) suivant :

- naissance d'un objet $u : X \rightarrow X \cup u$ avec une probabilité $B(x) \times h$ dans l'intervalle $(t, t + h)$ selon la loi de densité $b(X, u)$.
- mort d'un objet : $X \rightarrow X \setminus u$ avec une probabilité $D(x) \times h$ dans l'intervalle $(t, t + h)$ et selon une densité $d(X, u)$.

La possibilité d'observer plusieurs transitions pendant une période de temps h est notée O/h . Si $h \searrow 0$, $O/h \rightarrow 0$.

Le temps d'attente pour une transition (naissance ou mort) T a une distribution exponentielle inversement proportionnelle à la quantité $B(x) + D(x)$ où :

$$B(x) = \int_X b(x, u) d\lambda(u) \quad (3.3)$$

$$D(x) = \sum_{x_k \in X} d(x, x_k) \quad (3.4)$$

A un instant donné t , la probabilité d'une naissance est $B(x)/(B(x) + D(x))$ et la probabilité d'une mort est $D(x)/(B(x) + D(x))$.

Pour un processus naissance-mort le système reste dans le même état pendant un temps T_S , après lequel il change son état soit par une naissance, soit par un mort. L'algorithme naissance-mort génère cette succession d'intervalles T_S , en indiquant les moments de

transition pour la chaîne.

S'il existe un processus de naissance-mort tel que :

$$b(x, \xi)\pi(x) = D(x \cup \xi, \xi)\pi(x \cup \xi) \quad (3.5)$$

où $\pi(x \cup \xi) > 0$, la distribution d'équilibre unique est un processus de densité π .

L'inconvénient majeur de cet algorithme est que la convergence est atteinte quand le temps $\sum_s T(s)$ tend vers l'infini. Le processus associé doit être poursuivi pendant un temps $\sum_s T(s) > C$ où C est un nombre très grand. En conséquence l'algorithme naissance-mort n'est pas utilisé pour la détection des objets dans un espace donné (sans évolution dans le temps), comme dans le cas du traitement des images.

Le processus naissance-mort est un processus en temps continu, mais la succession des états forme une chaîne de Markov discrète, cette chaîne a en général, une distribution d'équilibre différente du processus en temps continu. Notons que certains problèmes peuvent être modélisés seulement en temps continu.

3.2.3 L'algorithme de type Metropolis-Hastings

Commençons par décrire l'algorithme dans sa forme générale telle que décrite par [?] pour ensuite montrer sa validité. L'algorithme ne nécessite qu'une valeur de départ X_0 et le choix d'une distribution conditionnelle de densité $q(x, y) = q(y | x)$. À une étape donnée t , les manipulations suivantes sont effectuées.

La procédure MH

1. À partir de la valeur $X_t = x$, on génère $Y_{t+1} = y$ selon la distribution de densité $q(y | x)$.
2. On pose $X_{t+1} = \begin{cases} Y_{t+1} & \text{avec probabilité } \alpha(x, y) \\ X_t & \text{avec probabilité } 1 - \alpha(x, y) \end{cases}$

où les seuils α doivent avoir la forme générale

$$\alpha(x, y) = \frac{s(x, y)}{1 + r(x, y)}.$$

$r(x, y)$ est le ratio $\frac{\pi(x)q(x, y)}{\pi(y)q(y, x)}$, et la fonction s est choisie de façon à ce que $s(x, y) = s(y, x)$ et $0 \leq \alpha \leq 1$. Habituellement, on utilise exclusivement

$$\alpha(x, y) = \min \left[1, \frac{\pi(y)q(x | y)}{\pi(x)q(y | x)} \right].$$

qui correspond au choix $s(x, y) = \min\{1 + r(x, y), 1 + r(y, x)\}$. Si en plus la densité q est symétrique ($q(y | x) = q(x | y)$), le rapport devient tout simplement

$$\alpha(x, y) = \min \left\{ 1, \frac{\pi(y)}{\pi(x)} \right\}.$$

Où q : la densité de proposition des candidats, ou densité instrumentale.
 α : les probabilités d'acceptation de ces derniers.
On donne maintenant les propriétés théoriques de cette procédure.

Propriétés

L'algorithme tel que défini génère une chaîne de Markov dont les probabilités de transition sont données par :

$$P(x, y) = q(x, y)\alpha(x, y), \text{ si } x \neq y$$

$$P(x, x) = 1 - \int P(x, y)dy, \text{ autrement}$$

Pour prouver que la distribution stationnaire de cette chaîne est π , il suffit de montrer qu'elle est réversible par rapport à π . Or,

$$\begin{aligned} \pi(x)P(x, y) &= \pi(x)q(x, y)\alpha(x, y) \\ &= \frac{\pi(x)q(x, y)s(x, y)}{1 + \frac{\pi(x)q(x, y)}{\pi(y)q(y, x)}} \\ &= \frac{\pi(x)\pi(y)q(x, y)q(y, x)s(x, y)}{\pi(y)q(y, x) + \pi(x)q(x, y)} \\ &= \frac{\pi(y)q(y, x)s(y, x)}{\frac{\pi(y)q(y, x)}{\pi(x)q(x, y)} + 1} \\ &= \pi(y)q(y, x)\alpha(y, x) = \pi(y)P(y, x). \end{aligned}$$

Maintenant, il faut s'assurer que la chaîne converge bien vers sa distribution stationnaire, c'est-à-dire qu'elle est ergodique. Or, ceci est facilement vérifié la plupart du temps. Par exemple, si $q(x, y)$ est positive pour toute paire (x, y) appartenant au support de π , $P(x, y)$ aussi sera toujours positive. Ainsi, à partir d'une valeur X_t donnée, toute valeur X_{t+1} sera atteignable en une seule étape avec une probabilité positive. La chaîne est donc irréductible. La chaîne sera aussi apériodique du moment qu'il existe au moins une paire (x, y) telle que $\alpha(x, y) < 1$, car on aura alors $P(x, x) > 0$. Cela sera pratiquement toujours vrai et on conclut donc que la chaîne est effectivement ergodique.

Remarques

1. C'est le choix de la distribution des candidats $q(x, y)$ qui influencera la qualité de l'algorithme. En effet, un choix judicieux favorisera une exploration rapide de l'espace d'états et accélérera en conséquence la convergence de la chaîne vers la distribution stationnaire. On veut donc une variabilité modérée dans les candidats possibles, de façon à favoriser les fluctuations tout en restreignant la probabilité de rejet. On verra dans les sections suivantes plusieurs façons d'ajuster la distribution instrumentale q pour une distribution cible donnée.

Pour des raisons de performance, on choisira évidemment une distribution q facile à simuler. En pratique, sur le support \mathbb{R}^d , ce sera presque toujours une loi normale

centrée à la valeur précédente de la chaîne : $X_{t+1} \rightsquigarrow N(X_t, \Sigma)$. C'est donc par le choix de la matrice Σ que l'on pourra optimiser les performances de l'algorithme. Notons que ce choix rend q symétrique et positive sur S .

2. En pratique, on éliminera souvent de l'échantillon obtenu les premières valeurs, qui sont fortement influencées par le choix de la valeur initiale. De même, comme l'indépendance est asymptotique, pour éviter qu'il y ait corrélation entre des valeurs consécutives de l'échantillon, on pourrait choisir de conserver uniquement une valeur sur 10 par exemple.
3. Puisque les probabilités $\alpha(x, y)$ ne dépendent de la distribution cible que sous la forme du rapport $\pi(y)/\pi(x)$, il est souvent possible d'appliquer l'algorithme sans connaître la constante de normalisation de la densité de π .

3.2.4 L'algorithme Metropolis-Hastings-Green

Cette extension de l'algorithme Metropolis-Hastings proposée par (MHG) construit aussi une chaîne Monte Carlo à sauts réversibles, mais propose des perturbations plus complexes que la naissance et mort présentées dans la section précédente, en utilisant un mélange de noyaux simples de transition.

Comme les transitions utilisées sont issues d'une combinaison des noyaux simples, nous pouvons noter :

$$Q(x, \cdot) = \sum_m P_m Q_m(x, \cdot)$$

Une transition dans la chaîne utilise le noyau combiné Q ; les noyaux Q_m sont les noyaux de base. La probabilité de choisir un noyau simple Q_m est P_m .

Pour construire une chaîne de Markov, l'algorithme MHG introduit une mesure symétrique $\xi(\cdot, \cdot)$ définie sur $\chi \times \chi$. La mesure $\xi_m(dx, dy)$ doit être continue par rapport à $\pi(dx)Q_m(x, dy)$. La mesure Radon-Nykodym est définie par :

$$f_m(x, y) = \frac{\pi(dx)Q_m(x, dy)}{\xi(dx, dy)}$$

Par suite, l'algorithme s'écrit :

A partir d'un état $X_t = x$:

1. Choisir un noyau de proposition Q_m avec une probabilité P_m .
2. Générer $y \sim Q(x, \cdot)$
3. Calculer le rapport d'acceptation :

$$R_m = \frac{f_m(y, x)}{f_m(x, y)}$$

4. Accepter l'état y avec la probabilité $\alpha = \min(1, R_m)$
5. Recommencer avec l'état actuel.

La π -réversibilité de la chaîne est assurée par le choix des taux d'acceptation et la symétrie de la mesure ξ . Comme dans le cas précédent, la stabilité du processus ponctuel est nécessaire pour la stabilité de la chaîne.

La chaîne restera convergente en utilisant les noyaux de naissance et mort simples (comme dans l'algorithme Metropolis-Hastings), les mouvements qui ne changent pas le nombre d'objets dans la configuration, et les noyaux de fusion de deux objets ou de division d'un objet en deux [47].

3.2.5 L'échantillonneur Gibbs

L'échantillonneur Gibbs est un cas spécial de l'algorithme Metropolis-Hastings. La différence entre les deux algorithmes est que l'échantillonneur Gibbs accepte tous les nouveaux états, mais ces états sont générés selon la densité vers laquelle on souhaite converger, contrairement à l'algorithme Metropolis qui génère les nouveaux états selon une densité auxiliaire uniforme sur l'ensemble des états.

L'algorithme Gibbs à partir d'une configuration $X = \{x_1, \dots, x_n\}$:

1. Choisir un objet à modifier $x_j \in X$.
2. Générer une nouvelle valeur x'_j selon $\pi(x_1, \dots, x_{j-1}, \cdot, x_{j+1}, \dots, x_n)$.
3. Recommencer avec la nouvelle configuration $X = \{x_1, \dots, x'_j, \dots, x_n\}$.

CHAPITRE

4

PROCESSUS PONCTUEL DE BADDELEY ET PLANIFICATION D'EXPÉRIENCES

Un processus ponctuel de poisson PPP modélise une répartition spatiale “au hasard” d'autre distribution sont plus régulières ; d'autres sont moins régulières, présentant par exemple des agrégats. Le processus ponctuel de Baddeley modélise des phénomènes attractifs.

Pour générer un tel processus, nous utiliserons les techniques usuelles de simulation par chaînes de Markov et plus précisément l'algorithme de Metropolis Hastings. C'est essentiellement la propriété de Markov spatiale du processus ponctuel de Baddeley qui en pratique permet de simuler de manière efficace ces nouveaux plans d'expériences, notamment en grande dimension. Ainsi, on peut parler de planification d'expériences par méthodes de Monte-Carlo par chaînes de Markov (MCMC).

4.1 Le processus ponctuel de Baddeley

Les modèles à interaction d'aire ou les modèles à interaction de connexité sont des alternatives au modèles de Strauss permettant de modéliser, sans contraintes sur les paramètres, des modèles ayant des répartitions spatiales plus ou moins régulières [48].

Ces modèles gardent la même forme de densité que celle de Strauss, à savoir :

$$f(x) = \alpha \beta^{n(x)} \gamma^{h(x)}.$$

Définissons alors la fonction h : Soit $r > 0$ et $B(x) = \{\cup_{x_i} B(x_i, \frac{r}{2})\} \cap S$ la réunion des boules de rayon $\frac{r}{2}$ centrées aux points de la configuration spatiale x , réunion limitée à S . Soit $a(x)$ l'aire de cette réunion et $c(x)$ le nombre de ses composantes connexes. Alors le processus à interaction d'aire est associé au choix $h(x) = a(x)$, alors que le modèle à interaction de connexité est associé à $h(x) = c(x)$.

Ainsi, les deux densités sont définies sans contraintes sur les paramètres (β, γ) . Pour les deux modèles la répartition spatiale sera plus (respectivement moins) régulière si $\gamma < 1$ (respectivement $\gamma > 1$), $\gamma = 1$ correspondant à un $PPP(\beta)$.

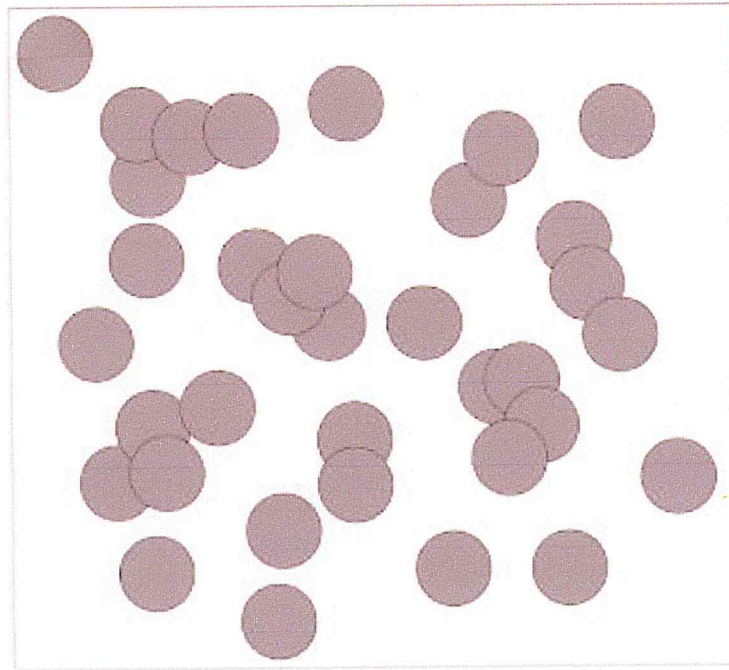


FIGURE 4.1 – Une réalisation de l'ensemble $B(x)$ permettant de définir l'aire $a(x)$ et le nombre de composantes connexes $c(x)$ (ici $c(x) = 19$).

Nous nous focalisons sur l'étude de processus ponctuel de Baddeley à interaction de connexité de densité :

$$f(x) = \alpha \beta^{n(x)} \gamma^{c(x)} \quad (4.1)$$

$\alpha > 0$ est une constante de normalisation qui fait de f une densité, $c(x)$ désigne le nombre de composantes connexes, $n(x)$ dénote le nombre de points d'une configuration x , β est un paramètre d'échelle et γ est un paramètre de répulsion.

4.2 Le processus de Baddeley est Markovien

La notion de processus ponctuel de Markov a été introduite par Ripley et Kelly [49]. Sa généralisation à la propriété de Markov aux plus proches voisins est due à Baddeley et Moller [50].

4.2.1 Propriété de Markov au sens de Ripley-Kelly

Soit X un processus ponctuel de densité f par rapport à un processus ponctuel de poisson d'intensité λ . Soit $\xi \sim \eta$ une relation de voisinage symétrique sur S : par exemple, pour un $r > 0$ fixé, $\xi \sim_r \eta$ si $\|\xi - \eta\| \leq r$. Le voisinage de $A \subset S$ est

$$\partial A = \{\eta \in S \text{ et } \eta \notin A : \exists \xi \in A \text{ t.q. } \xi \sim \eta\}.$$

On note $\partial\{\xi\} = \partial\xi$ si $\xi \in S$.

Definition 18. *Le processus X de densité $f(\cdot)$ est markovien pour la relation \sim si, pour toute configuration x , on a :*

1. $f(x) > 0$ implique $f(y) > 0$ pour tout $y \subset x$: on dit que $f(\cdot)$ est héréditaire
2. Si $f(x) > 0$, alors $\lambda(\xi, x) = f(x \cup \{\xi\})/f(x)$ ne dépend que de ξ et de $\partial\xi \cap x$.

La propriété de Markov dit que cette intensité conditionnelle ne dépend que des points de x voisins de ξ .

Quelques processus markoviens

1. Le modèle à noyau dur, de densité :

$$f(\{x_1, x_2, \dots, x_n\}) = \alpha \beta^n \mathbf{1}_{\{\forall i \neq j : \|x_i - x_j\| > 2r\}}$$

est markovien pour la relation : $u \sim v$ si $\|u - v\| \leq 2r$ sur R^d . En effet,

- a. $f(x) > 0$ si les points de x sont deux à deux à une distance $> 2r$; si $y \subset x$, alors $f(y) > 0$;
 - b. $\lambda(u; x) = \beta \mathbf{1}_{\{\partial u \cap x = \emptyset\}}$ si $f(x) > 0$: $\lambda(u; x)$ ne dépend que des x_i tels que $x_i \sim u$
2. Le processus de Strauss, de densité :

$$f(\{x_1, x_2, \dots, x_n\}) = \alpha \beta^{n(x)} \gamma^{\sum_{i \neq j} \mathbf{1}_{\{\|x_i - x_j\| \leq 2r\}}}, \quad \text{avec } \gamma \leq 1.$$

est markovien, d'intensité conditionnelle :

$$\lambda(u; x) = \beta \gamma^{\sum_i \mathbf{1}_{\{\|x_i - u\| \leq 2r\}}}$$

Par contre, comme on va le voir dans la suite, pour tout $r > 0$, le processus ponctuel de Baddeley à interaction de connexité n'est pas \sim_r -markovien :

En effet, deux points de S pouvant être connectés dans $B(x) = \{\cup_{x_i} B(x_i, \frac{r}{2})\} \cap S$ tout en étant arbitrairement éloignés, l'intensité conditionnelle $\lambda(\xi, x)$ peut dépendre de points de x arbitrairement éloignés de ξ .

4.2.2 Propriété de Markov aux plus proches voisins

Une propriété de Markov plus générale, dite propriété de Markov aux plus proches voisins (ppv) a été développée par Baddeley et Moller. Nous allons brièvement la présenter dans le contexte particulier d'un processus ponctuel de Baddeley à interaction de connexité. Processus ponctuels qui n'est pas markovien au sens de Ripley-Kelly mais l'est pour une nouvelle relation de voisinage qui dépend de la configuration x , la relation \sim_x aux x -plus proches voisins [51].

Soit x une configuration sur S , $r > 0$ fixé et $B(x) = \{\cup_{x_i} B(x_i, \frac{r}{2})\} \cap S$: on dira que deux points ξ et η de x sont connectés pour x si ξ et η sont dans une même composante connexe de $B(x)$. On notera $\xi \sim_x \eta$ cette relation de voisinage : \sim_x est une relation sur les points de x qui dépend de x . Notons que deux points de S peuvent être voisins pour \sim_x bien qu'arbitrairement éloignés pour la distance euclidienne : en effet, la connexion par composantes connexes reliera deux points s'il existe une chaîne de boules de rayon r centrées en des points de x et joignant l'un à l'autre.

Soit $c(x)$ le nombre de composantes connexes de $B(x)$. Si S est bornée, la densité conditionnelle de Papangelou du processus de Baddeley à interaction de connexité vaut :

$$\lambda(\xi, x) = \beta \gamma^{c(x \cup \{\xi\}) - c(x)}$$

Avec $\beta > 0$ et $\gamma > 0$. Pour voir que $\lambda(\xi, x)$ peut dépendre d'un point $\eta \in x$ arbitrairement éloigné de u dans S , on peut choisir une configuration z telle que pour tout $x = z \cup \{\eta\}$, $c(z) = 2$, $c(x) = 1$, $c(x) = c(x \cup \{\xi\}) = 1$. Ainsi pour tout $R > 0$, X n'est pas markovien au sens de Ripley-Kelly pour la relation de r -voisinage habituelle.

Par contre, si $\eta \in x$ n'est pas connecté à ξ dans $B(x \cup \{\xi\})$, η ne contribue pas à la différence $c(x \cup \{\xi\}) - c(x)$ et $\lambda(\xi, x)$ ne dépend pas de η : X est markovien pour la relation aux ppv par composantes connexes.

4.3 Générer des plans d'expériences de Baddeley

De toutes les familles de méthodes de Monte Carlo par chaînes de Markov (MCMC), la plus générale est sans doute l'algorithme Metropolis-Hastings, dans le sens qu'il impose le moins de conditions sur la densité cible. Il existe plusieurs sous-catégories d'algorithmes Metropolis-Hastings, suivant le type de distribution q choisie. Il est souvent préférable (entendre plus facile) de se tourner vers une approche plus locale de l'exploration du support. Pour ce faire, on crée l'algorithme de sorte à générer une chaîne de Markov du type marche aléatoire, c'est-à-dire que la densité instrumentale q sera centrée sur la valeur courante de la chaîne et sera symétrique. La valeur proposée y_{t+1} suivra donc la forme $x_t + \epsilon$, où ϵ est une perturbation aléatoire distribuée selon q et indépendante de x_t .

4.3.1 Algorithme Metropolis-Hastings avec marche aléatoire

L'Algorithme de type Metropolis-Hastings avec marche aléatoire (Random Walk Metropolis-Hastings (RWMH)) comporte une chaîne de Markov instrumentale dont la densité de transition dépend de l'état présent, plus précisément elle satisfait $q(x, y) = q(y - x)$. On peut choisir une densité instrumentale $N(x, \sigma^2)$. Donc, étant donné un état présent x , l'algorithme génère un état potentiel y provenant d'une distribution normale centrée à x et avec variance σ^2 . Si le nouvel état y est accepté, le prochain état potentiel sera généré selon une distribution $N(y, \sigma^2)$. Sinon, x sera posé comme état actuel et un autre état y éventuel sera proposé selon une loi $N(x, \sigma^2)$. L'algorithme RWMH avec loi instrumentale $N(x, \sigma^2)$ a une propriété additionnelle conférée par la symétrie de la densité instrumentale, notamment le fait que $q(x, y) = q(y, x)$. Cette caractéristique permet de simplifier le calcul de α :

$$\alpha(x, y) = \min \left[1, \frac{\pi(y)}{\pi(x)} \right]$$

Nous considérons ici la simulation à partir de $\pi(x)$ pour un n fixe. Ainsi,

$$\pi(x) = \alpha \beta^{n(x)} \gamma^{c(x)} \quad (4.2)$$

Où $x = (x_1, \dots, x_n)$.

En calculant le rapport $\frac{\pi(y)}{\pi(x)}$ on aura une simplification pour obtenir à la fin le rapport simplifié $\frac{\gamma^{c(y)}}{\gamma^{c(x)}}$ utilisé dans l'algorithme suivant :

Algorithm 1 Algorithme de RWMH pour générer des plans dits de Baddeley

Initialisation : choisir un plan d'expériences $x_1 \cdots x_n$ selon une loi de probabilité donnée (loi uniforme par exemple)

POUR N = 1 à NMC

Choisir une expérience x_i au hasard

Simuler y_i suivant la loi de proposition $q \sim N(x_i, \sigma^2)$

Accepter le changement $x_i = y_i$ avec la probabilité

$$\alpha(x, y) = \min \left[1, \frac{\gamma^{c(y)}}{\gamma^{c(x)}} \right]$$

où $y = (x_1, \dots, x_{i-1}, y_i, x_{i+1}, \dots, x_n)$.

Fin pour N

Voici un exemple de l'exécution de notre programme MATLAB qui implémente un Algorithme RWMH pour simuler un processus de Baddeley avec $n = 25$ points sur l'unité carrée $[0, 1] \times [0, 1]$ et de $NMC = 1000$ itérations en utilisant des valeurs de paramètre $\gamma = 0,01$ et $r = 0,1$.

Étant donné l'état actuel $x = (x_1, \dots, x_n)$, l'état de la proposition $Y = (Y_1, \dots, Y_n)$ est identique à x sauf pour le $j^{\text{ème}}$ composant où j est un indice uniformément tiré de l'ensemble $\{1, \dots, n\}$. Plus précisément, $Y_j = x_j + Z$ où $Z \sim N(0, (0,1)^2)$.

La proposition Y pour cet échantillonneur de marche aléatoire est acceptée avec la probabilité suivante : $\alpha(x, Y) = \min\{\gamma^{c(Y)}/\gamma^{c(x)}, 1\}$.

La fonction $c(x)$ est implémenté en tant que le nombre de composantes connexes de la configuration x , qu'on va expliquer dans la suite les procédures adaptées pour la calculer.

La figure ci-dessous montre l'évolution ou la transformation effectuée par notre algorithme de simulation MCMC (RWMH), celle ci est bien traduite par un départ d'une configuration initiale aléatoire pour faire aboutir à une configuration finale qui représente un plan de Baddeley de paramètres tels que définies dans la paragraphe précédente.

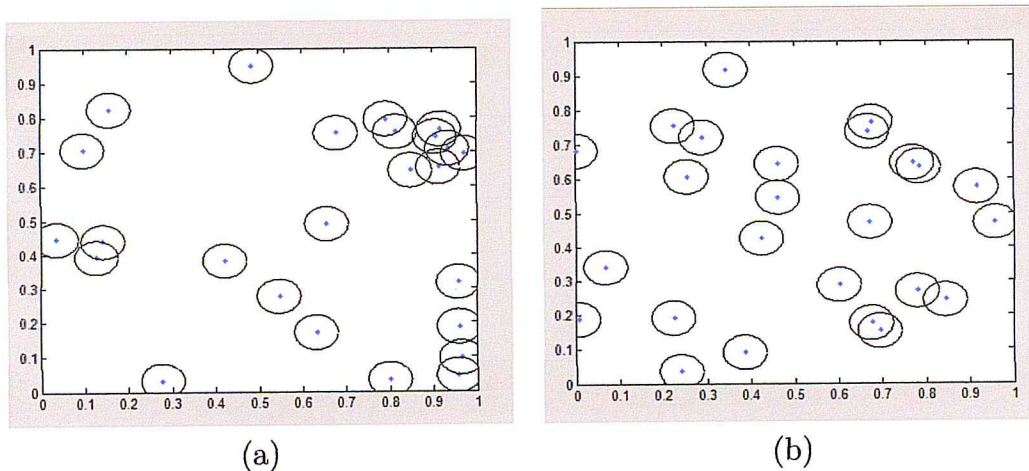


FIGURE 4.2 – (a) une configuration initiale de 25 points et (b) une configuration finale pour $\gamma = 0.01$ et $r = 0.1$

4.3.2 Le calcul de $c(x)$: nombre de composantes connexes

Pour le calcul de $c(x)$, nous avons utilisé une approche basée sur la théorie des graphe. En effet, nous avons associé à la configuration x un graphe simple non orienté $G(X, E)$ constitué par :

- L'ensemble des sommets : $X = \{x_1, \dots, x_n\}$
- L'ensemble des arêtes : $E = \{(x_i, x_j) \in X \times X \text{ tel que } \|x_i - x_j\| \leq 2r\}$

Tout graphe $G = (X, E)$ peut être représenté par une matrice. Les relations d'adjacences (si deux sommets sont reliés par une arête ils sont adjacents) sont représentés par sa matrice d'adjacence. Elle est définie par :

$$a_{ij} = \begin{cases} 1 & \text{si } (x_i, x_j) \in E \\ 0 & \text{sinon} \end{cases}$$

Ensuite nous avons proposé un algorithme qui, tout d'abord construit le graphe $G(X, E)$ par la donnée des ensembles X et E , puis il calcule et affiche sa matrice d'adjacence et finalement il réalise une suite de traitement sur cette dernière pour obtenir à la fin le

nombre de composantes connexes $c(x)$.

Une fois la matrice d'adjacence est construite on adapte une procédure spécifique pour le calcul de $c(x)$ résumée par les démarches suivantes :

En premier lieu on fixe la première ligne de la matrice d'adjacence ($i = 1$) et on le sauvegarde dans un tableau de sommets parcourus TNP , puis on varie j qui parcourt les colonnes de M pour obtenir tous les j qui vérifient $M[1][j] = 1$ puis on refait le même travail pour les $i = j$ et on les ajoute tous à TNP , à ce point là l'ensemble des sommets contenus dans TNP forme une seule composante connexe.

Deuxièmement on passe aux sommets qui n'appartiennent pas encore au tableau des nœuds parcourus TNP puis on reprend la même astuce que précédemment.

Troisièmement à chaque fois qu'on trouve une composante connexe on incrémente le nombre de ces composantes, jusqu'à ce que tous les sommets soient visités.

Algorithm 2 L'Algorithme proposé pour calculer le nombre de composantes connexes

Debut

Remplissage de la Matrice d'adjacence M de taille $n \times n$

Initialisation de TNP le tableau des nœuds parcourus à 0

Recherche des composantes connexes ncx

Tant que ($i \leq n$)

si (i appartient à TNP)

i est déjà parcouru

 passer au nœud suivant

Sinon // i n'existe pas dans TNP donc :

 ajouter i à la 1ère position nulle de TNP

 chercher les voisins de i puis les voisins de ses

 voisins s'ils existent

fin si ;

$ncx = ncx + 1$;

$i = i + 1$;

fin Tant que ;

Ecrire("le nombre de composantes connexes $c(x) = \%d$ ", ncx)

fin.

4.3.3 Influence des paramètres

L'influence du paramètre r :

Les figures ci-dessous montrent l'influence du paramètre r sur la distribution finale. Le choix du rayon s'avère être important :

1. un rayon trop petit engendre une distribution sans interaction mais avec de nombreuses lacunes (distribution uniforme),
2. un rayon trop grand conduit à une distribution avec des agglomérats (clusters).

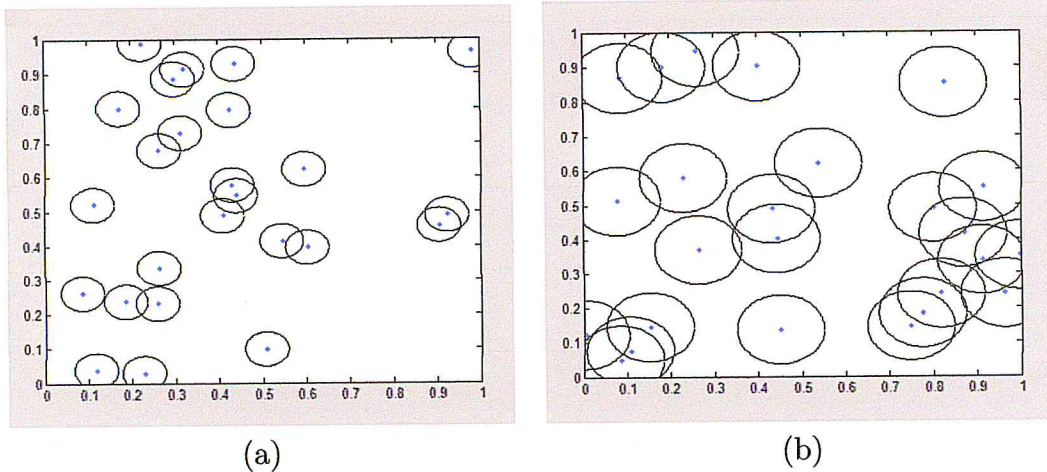


FIGURE 4.3 – (a) une configuration de 25 points $\gamma = 0.01$ et $r = 0.1$ et (b) une configuration pour $\gamma = 0.01$ et $r = 0.2$

L'influence du paramètre de répulsion γ :

De même que pour le rayon d'interaction, il est important de bien fixer le paramètre de répulsion γ . La figure ci-dessous montre qu'il est plus aisé de générer une distribution répondant au critère de remplissage de l'espace avec un paramètre de répulsion faible.

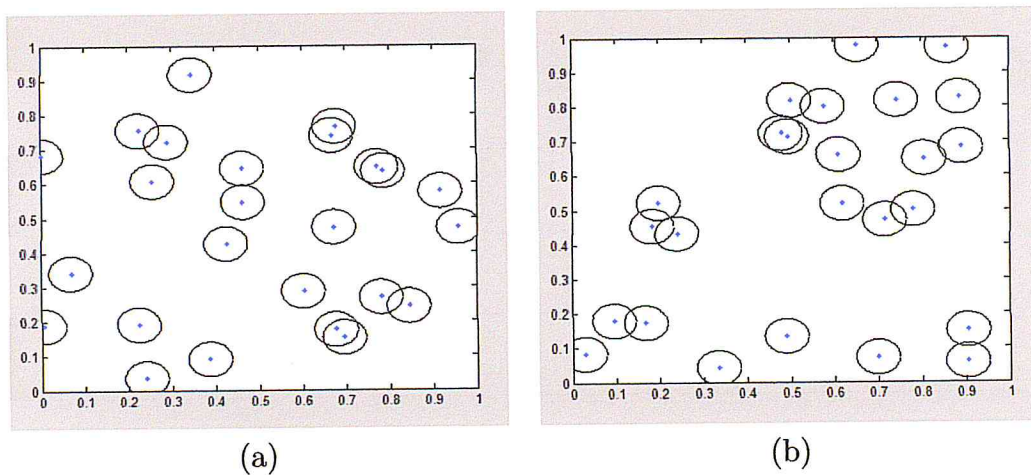


FIGURE 4.4 – (a) une configuration de 25 points $\gamma = 0.001$ et $r = 0.1$ et (b) une configuration pour $\gamma = 0.1$ et $r = 0.1$

4.4 Étude de convergence

On cherche maintenant à prouver que la chaîne de Markov $\{X_n\}_{n \geq 0}$ construite par notre algorithme de simulation MCMC converge vers une distribution dite stationnaire. On commence par définir la propriété de stationnarité d'une loi. Soit $|E|$ le cardinal de l'ensemble E .

Definition 19 (Loi stationnaire). *Le vecteur ligne $\pi = (\pi(1); \dots; \pi(|E|))$ est une distribution stationnaire d'une chaîne de Markov de matrice de transition P si $\pi P = \pi$.*

La propriété suivante d'irréductibilité vise à garantir l'unicité de la loi stationnaire de la chaîne de Markov, indépendamment de la condition initiale.

Definition 20 (Chaîne irréductible). *Une chaîne de Markov est dite irréductible si pour tout $x; y \in E$ il est possible d'atteindre l'état y à partir de x en un nombre fini de transition i.e., il existe pour tout $n \geq 0$ un entier m tel que*

$$P(X_{n+m} = y | X_n = x) = P_{xy}^m > 0$$

avec P^m est la matrice pour m transition.

Proposition 1 (Unicité de la loi stationnaire). *Si la chaîne est irréductible, alors la loi stationnaire π est unique. De plus $\pi(x) > 0$ pour tout $x \in E$.*

La propriété suivante sert à garantir que la chaîne possède la propriété de rester dans un état pendant un temps arbitraire.

Definition 21 (apériodicité). *La période d'un état $x \in E$ est définie par*

$$p.g.c.d.\{m \geq 1; P_{xy}^m > 0\},$$

où *p.g.c.d.* désigne le plus grand commun diviseur. *Une chaîne est dite apériodique si ses états possèdent une période égale à 1.*

Une condition suffisante pour garantir l'apériodicité est que pour tout $x \in E$ on ait $P_{xx} > 0$.

La dernière propriété qui nous intéressera ici sera la réversibilité de la chaîne.

Definition 22. *Une chaîne de Markov sera réversible par rapport à une distribution π si :*

$$\pi(x)P_{xy} = \pi(y)P_{yx}$$

pour tout x, y dans E .

Il s'agit en quelque sorte d'une symétrie entre la chaîne et son processus inverse (avec le temps à reculons). Ceci est une condition suffisante (mais non nécessaire), parfois facile à vérifier, pour que π soit une distribution stationnaire. Par construction, la réversibilité des chaînes MCMC garantit l'existence d'une distribution stationnaire π de la chaîne.

Nous avons à présent réuni des propriétés suffisantes pour garantir la convergence de la chaîne vers la loi stationnaire indépendamment de la distribution initiale [52].

Théorème 3 (Convergence en loi). [52] *Si la chaîne est irréductible et apériodique et de loi stationnaire π alors pour toute loi initiale μ , la loi de X_n converge vers π :*

$$P(X_n) = \mu P^n \rightarrow \pi \quad \text{quand } n \rightarrow \infty$$

Bien que ce théorème garantisse la convergence asymptotique des chaînes MCMC, vérifions d'abord que la chaîne générée par l'algorithme possède les propriétés nous assurant de l'existence de sa distribution stationnaire et de la convergence de celle-ci vers la densité cible $\pi(x)$

Démonstration 1. *Pour prouver que la distribution stationnaire de cette chaîne est π , il suffit de montrer qu'il est réversible par rapport à π . Ensuite il faut s'assurer que la chaîne converge bien vers sa distribution stationnaire, c'est à dire qu'elle est irréductible et apériodique. Nous montrons les propriétés importantes pour le noyau de transition P : la réversibilité, la stationarité, l'irréductibilité et l'apériodicité.*

1. La réversibilité

Par définition, le noyau de transition P est π -réversible si π satisfait la condition : $\pi(x)P(x, y) = \pi(y)P(y, x)$, dite (detailed balance condition). Ceci se vérifie facilement :

$$\begin{aligned} \pi(x)P(x, y) &= \pi(x) \min \left[1, \frac{\pi(y)q(y, x)}{\pi(x)q(x, y)} \right] q(x, y) \\ &= \min [\pi(x)q(x, y), \pi(y)q(y, x)] \\ &= \pi(y) \min \left[\frac{\pi(y)q(y, x)}{\pi(x)q(x, y)}, 1 \right] q(y, x) \\ &= \pi(y)P(y, x) \end{aligned} \tag{4.3}$$

En utilisant le fait que $q(x, y) = q(y, x)$ (par hypothèse) et $\pi(x)$ la densité tel que définie en (4.1) alors on aura :

$$\begin{aligned} \pi(x)a(x, y)q(x, y) &= \alpha \beta^{n(x)} \gamma^{c(x)} \min \{ 1, \beta^{n(y)-n(x)} \gamma^{c(y)-c(x)} \} q(x, y) \\ &= \alpha \min \{ \beta^{n(x)} \gamma^{c(x)}, \beta^{n(y)} \gamma^{c(y)} \} q(x, y) \\ &= \alpha \beta^{n(y)} \gamma^{c(y)} \min \{ \beta^{n(x)-n(y)} \gamma^{c(x)-c(y)}, 1 \} q(x, y) \\ &= \pi(y)a(y, x)q(x, y) \end{aligned} \tag{4.4}$$

En particulier, pour le cas particulier où nous considérons la simulation à partir de $\pi(x|n)$ pour un n fixe. Ainsi, $\pi(x|n) = \gamma^{c(x)}$, où $x = (x_1, \dots, x_n)$. On aura :

$$\begin{aligned} \pi(x)P(x, y) &= \gamma^{c(x)} \min \{ 1, \gamma^{c(y)-c(x)} \} q(x, y) \\ &= \min \{ \gamma^{c(x)}, \gamma^{c(y)} \} q(x, y) \\ &= \gamma^{c(y)} \min \{ \gamma^{c(x)-c(y)}, 1 \} q(y, x) \\ &= \pi(y)P(y, x) \end{aligned} \tag{4.5}$$

Donc, la chaîne est réversible et π est sa distribution stationnaire.

2. La stationnarité

Démontrons donc maintenant que π est invariante (stationnaire). Il suffit pour cela de démontrer que, quel que soit A ensemble mesurable :

$$\int P(x, A)\pi(x)dx = \int_A \pi(x)dx$$

Or :

$$\int P(x, A)\pi(x)dx = \int \int_A P(x, y)\pi(x)dydx = \int \int 1_A(y)P(x, y)\pi(x)dydx$$

On utilise alors l'équation d'équilibre $\pi(x)P(x, y) = \pi(y)P(y, x)$ et on obtient :

$$\int P(x, A)\pi(x)dx = \int \int 1_A(y)P(y, x)\pi(y)dydx$$

Toutes les quantités dans l'intégrale double étant positives on peut appliquer le théorème de Fubini :

$$\int P(x, A)\pi(x)dx = \int 1_A(y)\pi(y) \left(\int P(y, x)dx \right) dy$$

Or $P(\cdot, \cdot)$ étant une probabilité de transition on a $\forall y, \int P(y, x)dx = 1$ donc :

$$\int P(x, A)\pi(x)dx = \int_A \pi(y)dy$$

Et comme on a :

$$\begin{aligned} \int P(x, A)\pi(x)dx &= \alpha\beta^{n(x)}\gamma^{c(x)} \min\{1, \beta^{n(A)-n(x)}\gamma^{c(A)-c(x)}\}q(x, y) \\ &= \int \int_A \alpha\beta^{n(x)}\gamma^{c(x)} \min\{1, \beta^{n(y)-n(x)}\gamma^{c(y)-c(x)}\}q(x, y)dydx \\ &= \int \int 1_A(y)\alpha\beta^{n(x)}\gamma^{c(x)} \min\{1, \beta^{n(y)-n(x)}\gamma^{c(y)-c(x)}\}q(x, y)dydx \\ &= \int \int 1_A(y)\alpha\beta^{n(y)}\gamma^{c(y)} \min\{1, \beta^{n(x)-n(y)}\gamma^{c(x)-c(y)}\}q(y, x)dydx \\ &= \int 1_A(y)\alpha\beta^{n(y)}\gamma^{c(y)} \left(\int \min\{1, \beta^{n(x)-n(y)}\gamma^{c(x)-c(y)}\}q(y, x)dx \right) dy \\ &= \int_A \alpha\beta^{n(y)}\gamma^{c(y)} dy \\ &= \int_A \pi(x)dx \end{aligned} \tag{4.6}$$

ce qui est exactement ce qu'il fallait démontrer.

3. L'irréductibilité

$q(x, y) > 0, \forall (x, y) \in E \times E, P(x, y) = \alpha(x, y)q(x, y)$ aussi sera toujours positive. Ainsi à partir d'une valeur X_n donnée, toute valeur X_{n+1} sera atteignable en une seule étape avec une probabilité positive. Les états communiquent donc tous entre eux et la chaîne est ainsi composée d'une seule classe. La chaîne est donc irréductible.

4. L'apériodicité

La chaîne sera aussi apériodique du moment qu'il existe au moins une paire (x, y) telle que $\alpha(x, y) < 1$, car on aura alors $P(x, x) > 0$. On voit rapidement que la chaîne est apériodique puisque l'évènement $X_{n+1} = X_n$ est possible pratiquement à tout moment. En effet, chaque état peut donc être visité à deux itérations consécutives, d'où $P_{xx}^1 > 0$, leur période étant ainsi de 1.

Conclusion : Enfin la chaîne de Markov $\{X_n\}_{n \geq 0}$ de noyau de transition P est réversible, stationnaire, irréductible et apériodique. On est donc dans les conditions d'applications du théorème de convergence en loi, on conclut donc que la chaîne possède les bonnes propriétés pour converger vers la distribution cible $\pi(x)$.

CHAPITRE

5

IMPLÉMENTATION, DISCUSSIONS ET COMPARAISON

Les expérimentations ont été réalisées sur un PC équipé d'un processeur Intel(R) Core (TM) i3-3217U CPU @ 1.80 GHz avec une mémoire de 4 GB sous Windows 10 Professionnel. Les algorithmes des différentes méthodes utilisées dans la comparaison sont implémentés en MATLAB. Les programmes prennent comme donnée d'entrée des matrices d'expériences contenant des points générés d'une manière aléatoire, selon une densité donnée (en général on utilise la loi uniforme).

5.1 Matlab comme outil de programmation

Matlab est un langage hautes performances pour le calcul scientifique et technique. Il intègre la possibilité de calculs, de visualisation et de programmation dans un environnement très simple d'emploi. Les résultats sont exprimés sous une forme mathématique standard. L'utilisation typique est :

1. Calcul scientifique.
2. Développement d'algorithmes.
3. Acquisition de données.
4. Modélisation et simulation.
5. Analyse de données, exploration et visualisation.
6. Graphisme scientifique.
7. Développement d'applications, interface graphique.

Matlab est un système interactif dont la brique de base est un tableau dont la taille n'est pas nécessairement connue. Ceci permet de résoudre des problèmes, en particulier ceux qui ont une formulation matricielle, en un minimum de temps (contrairement aux langages de bas niveau comme le C ou le fortran). Matlab a été développé à l'origine pour avoir un accès simple et rapide aux projets EISPACK et LINPACK. Aujourd'hui, Matlab intègre les bibliothèques LAPACK et BLAS incorporant ainsi les dernières techniques pour le calcul matriciel.

Dans l'enseignement universitaire Matlab s'est imposé comme un standard pour l'apprentissage de l'algorithmique scientifique. Dans l'industrie, il est l'outil de choix pour une productivité accrue en recherche et développement. Matlab peut aussi être enrichi à l'aide de Toolbox (boîtes à outils) pour des problèmes spécifiques. Matlab fournit une documentation détaillée à la fois en ligne et sous forme de fichiers PDF.

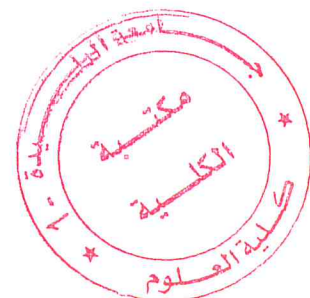
5.2 Etude de comparaison avec les plans usuels

Avant de comparer les plans en dimension supérieure où égale à 2, il est important de faire appel à des critères usuels permettant de juger le bon remplissage de l'espace ainsi la distribution uniforme. (Pour plus de détail voir le chapitre 1)

1. Mesure de recouvrement : permet de mesurer l'écart entre les points du plan et ceux d'une grille régulière. Ce critère est nul pour une grille régulière. L'objectif est donc de le minimiser pour se rapprocher d'une grille régulière, et ainsi assurer le remplissage de l'espace, sans toutefois l'atteindre pour respecter une distribution uniforme notamment en projection sur les axes factoriels.
2. Critère de distance : Le critère maximin (Mindist) consiste à maximiser la distance minimale entre deux points du plan.
3. Critère de discrédance : La discrédance permet de mesurer l'écart entre la fonction de répartition empirique des points du plan et celle de la loi uniforme. Contrairement aux deux critères précédents, la discrédance n'est pas basée sur la distance entre les points. Il existe différentes mesures de discrédance. Nous retenons la discrédance en norme L2.
4. Le critère R est le rapport entre la distance maximale et minimale entre deux points du plan d'expérience.

Les plans proposés dans ce mémoire sont comparés avec les plans usuels utilisés en expériences numériques :

- Plans aléatoires (RD).
- Hyper cubes latins (LHS).
- Plans maximinLHS (mLHS)
- Suite de Sobol .
- Suite de Halton .
- Suite de Faure .
- Plans de Strauss (SD).
- Plans à entropie maximal (Dmax)



5.3 Résultats et discussions

Dans cette partie, nous présentons les tables regroupant les résultats d'implémentation des différents algorithmes pour les différents plans considérés. Ces tables sont divisées selon le nombre de points et la dimension de l'espace expérimental. Les plans proposés donnent en globalité les meilleurs résultats quel que soit le critère. Ils semblent concurrencer les plans de Strauss construit par l'approche utilisant des processus ponctuels simple. Même donnent un meilleur résultat que les suites de faible discrédance traditionnellement utilisées en phase exploratoire et ceci même pour le critère de discrédance.

5.3.1 Résultat pour 10 points et 2 dimensions

Plans \ Critères	Le rapport	Discrédance	Recouvrement	Distance
aléatoire	5.553536	0.03200207	0.5215326	0.07773079
LHS	3.323186	0.03009458	0.3359125	0.1340521
mLHS	3.11082	0.01432051	0.3161762	0.152288
Sobol	1.118034	0.003047299	0.1362939	0.1767767
Halton	2.080255	0.007570195	0.2810232	0.1607922
Faure	6	0.005308041	0.5775323	0,08838835
Dmax	3.238632	0.008652122	0.1547445	0.2925121
Strauss	4.458029	0.01307772	0.1642766	0.290778
Baddeley	2.93353	0.007214372	0.07933404	0.5248179

TABLE 5.1 – Listes des différents plans selon les quartes critères avec $n=10$, $R=0.1$, $\gamma = 0.05$, $NMC=500$, $d=2$

80 plans de chaque type sont générés afin de donner un sens aux résultats. Les figures ci-dessous représentent les box-plots obtenus à la suite de cette étude pour les plans :

- Plans de Baddeley (BD)
- Plans aléatoires (RD)
- Hyper cubes latins (LHS)
- Plans maximinLHS (mLHS)
- Plans de Strauss (SD)
- Plans à entropie maximal (Dmax)

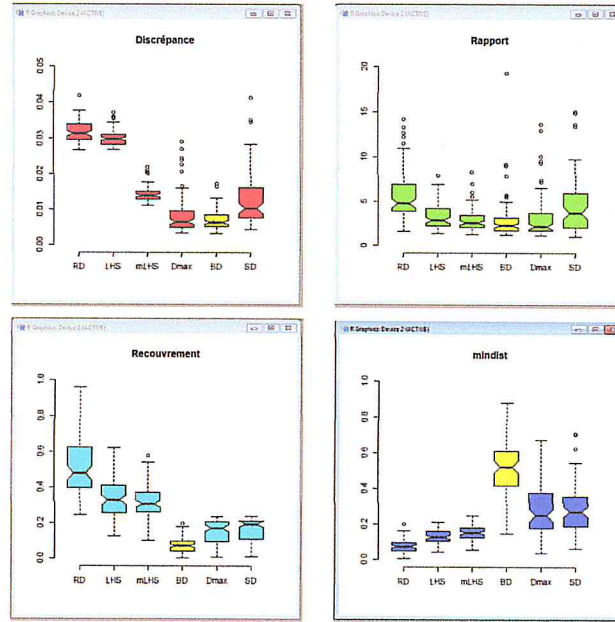


FIGURE 5.1 – Boxplots des critères de qualité pour deux dimensions

5.3.2 Résultat pour 20 points et 3 dimensions

Plans \ Critères	Le rapport	Discrepance	Recouvrement	Distance
aléatoire	6.229212	0.01432051	0.4084524	0.09660525
LHS	3.721318	0.01133634	0.3229826	0.1353985
mLHS	2.925884	0.01072663	0.2592255	0.1744015
Sobol	2.553139	0.001382617	0.2864609	0.1623798
Halton	2.562003	0.001732852	0.2601684	0.1719613
Faure	2	0.00113882	0.2082265	0.169725
Dmax	5.336851	0.01254935	0.08613133	0.5148527
Strauss	6.010683	0.01235391	0.08336419	0.4957753
Baddeley	5.349865	0.01224167	0.0764854	0.518426

TABLE 5.2 – Listes des différents plans selon les quartes critères avec $n=20$, $R=0.1$, $\gamma = 0.05$, $NMC=500$, $d=3$

Les figures ci-dessous représentent les résultats de 80 plans générés pour chaque types de plans suivants :

- Plans de Baddeley (BD)
- Plans aléatoires (RD)
- Hyper cubes latins (LHS)
- Plans maximinLHS (mLHS)

- Plans de Strauss (SD)
- Plans à entropie maximal (Dmax)

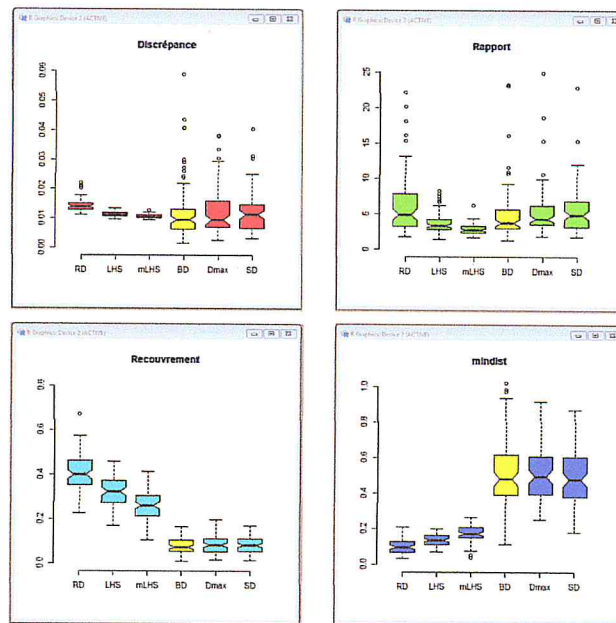


FIGURE 5.2 – Boxplots des critères de qualité pour trois dimensions

5.3.3 Résultat pour 40 points et 5 dimensions

Les résultats regroupés dans le tableau ci dessous, montrent une amélioration sur les valeurs des différents critères retenus.

Plans	Critères	Le rapport	Discrepance	Recouvrement	Distance
aléatoire		6.893661	0.01130093	0.2779045	0.1872624
LHS		2.527428	0.01067706	0.2537125	0.2654198
mLHS		2.690024	0.001754879	0.2030364	0.2606106
Sobol		2.066133	0.000260329	0.2110594	0.2952257
Halton		2.483516	0.000400047	0.2378431	0.2406925
Faure		2.008684	0.00028458	0,1952901	0.3179937
Dmax		6.025024	0.001628364	0.08142545	0.5382086
Strauss		6.219888	0.009685785	0.07330626	0.5521754
Baddeley		3.60482	0.001759214	0.07394665	0.5547977

TABLE 5.3 – Listes des différents plans selon les quartes critères avec $n=40$, $R=0.1$, $\gamma = 0.05$, $NMC=500$, $d=5$

Les figures ci-dessous représentent les résultats de 80 plans générés pour chaque types de plans suivants :

- Plans de Baddeley (BD)
- Plans aléatoires (RD)
- Hyper cubes latins (LHS)
- Plans maximinLHS (mLHS)
- Plans de Strauss (SD)
- Plans à entropie maximal (Dmax)

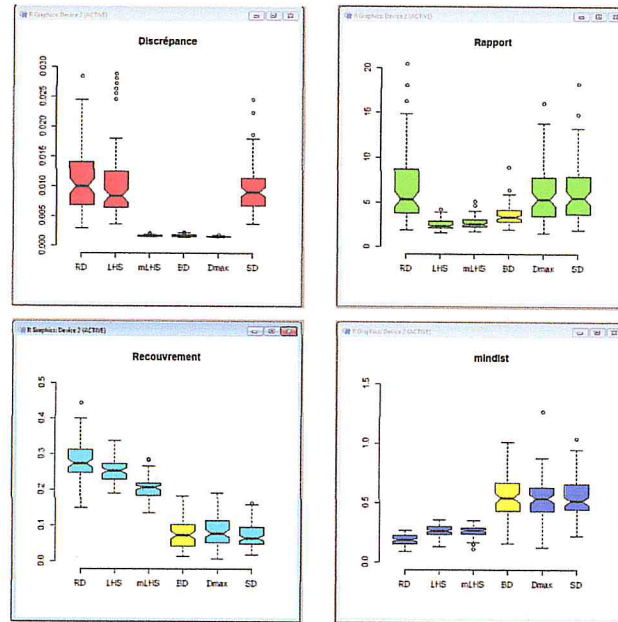


FIGURE 5.3 – Boxplots des critères de qualité pour cinq dimensions

5.3.4 Résultat pour 50 points et 7 dimensions

Plans \ Critères	Le rapport	Discrédence	Recouvrement	Distance
aléatoire	9.629222	0.01130093	0.2132717	0.2936924
LHS	2.571722	0.01824426	0.1984752	0.3156578
mLHS	2.251442	0.0003156118	0.1774884	0.3628827
Sobol	2.250821	0.00000743	0.110397	0,5595626
Halton	1.828507	0.0000560	0.1349754	0.507455
Faure	3.200336	0.00013279	0.3664268	0.2874798
Dmax	6.803585	0.01462145	0.06925627	0.5151662
Strauss	6.394307	0.01183629	0.07542196	0.5071142
Baddeley	2.804801	0.000257731	0.06991096	0.557594

TABLE 5.4 – Listes des différents plans selon les quartes critères avec $n=50$, $R=0.1$, $\gamma = 0.05$, $NMC=500$, $d=7$

Les figures ci-dessous représentent les résultats de 80 plans générés pour chaque types de plans suivants :

- Plans de Baddeley (BD)
- Plans aléatoires (RD)
- Hyper cubes latins (LHS)
- Plans maximinLHS (mLHS)
- Plans de Strauss (SD)
- Plans à entropie maximal (Dmax)

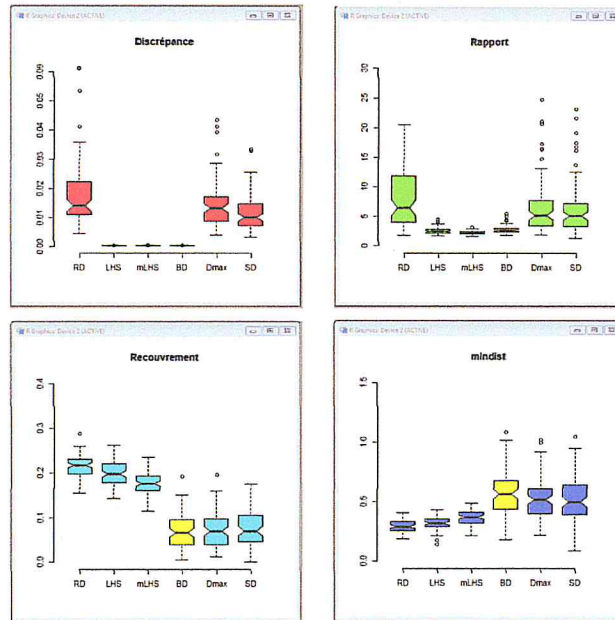


FIGURE 5.4 – Boxplots des critères de qualité pour sept dimensions

5.3.5 Résultat pour 80 points et 10 dimensions

Les figures ci-dessous représentent les résultats de 80 plans générés pour chaque types de plans suivants :

- Plans de Baddeley (BD)
- Plans aléatoires (RD)
- Hyper cubes latins (LHS)
- Plans maximinLHS (mLHS)
- Plans de Strauss (SD)
- Plans à entropie maximal (Dmax)

Plans \ Critères	Le rapport	Discrèpance	Recouvrement	Distance
aléatoire	6.000734	0.01679324	0.1677978	0.4303464
LHS	2.196265	0.000014	0.156489	0.4518074
mLHS	2.216084	0.000023	0.1570912	0.4673578
Sobol	1.766043	0.000007	0.110397	0.5595626
Halton	1.828507	0.000056	0.1349754	0.507455
Faure	3.200336	0.00013279	0.3664268	0.2874798
Dmax	4.958062	0.01331964	0.07310013	0.5354068
Strauss	5.846509	0.01239033	0.07493931	0.521444
Baddeley	2.217864	0.000140	0.07213461	0.5532225

TABLE 5.5 – Listes des différents plans selon les quatre critères avec $n=80$, $R=0.1$, $\gamma = 0.05$, $NMC=500$, $d=10$

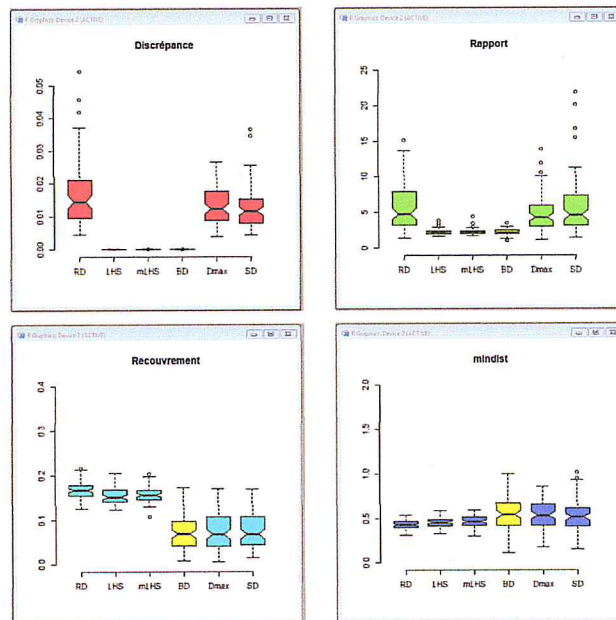


FIGURE 5.5 – Boxplots des critères de qualité pour dix dimensions

CONCLUSION GÉNÉRALE

La méthode des plans d'expériences est un ensemble de méthode et de mode de raisonnement destiné à tout expérimentateur désirant faire de la planification expérimentale, dans ce contexte nous nous sommes intéressés à la construction des plans d'expériences à partir de simulateurs déterministes et en phase exploratoire i.e. lorsque la dépendance entre les entrées et les sorties est à priori inconnue.

L'utilisation des méthodes MCMC et l'algorithme de Métropolis Hasting dans le contexte des plans d'expériences numériques permet de construire de nouveaux plans spécifiés à partir de la loi de Baddeley. Cette approche offre une grande souplesse dans la mesure où l'on peut facilement jouer sur cette loi via sa représentation pour imposer des propriétés comme par exemple le remplissage du domaine d'étude.

Enfin, une comparaison avec d'autres plans numériques a été effectuée et les résultats obtenus sont satisfaisants.

BIBLIOGRAPHIE

- [1] FISHER Ronald A. (1935). The design of Experiments. Oliver and Boyd.
- [2] McKay M.D., Beckman R.J., Conover W.J. (1979). Comparison of three methods for selecting values input variables in the analysis of output from a computer code. Technometrics, 21, 239-245.
- [3] Stein M. (1987). Large sample properties of simulations using latin hypercube sampling. Technometrics, 29, p. 143-151
- [4] Owen A.B. (1992). Orthogonal arrays for computer experiments, integration and visualization. Statist. Sinica 2, 439-452.
- [5] Niederreiter H. (1992). Random number generation and quasi-Monte Carlo methods. CBMS-NSF, SIAM, Philadelphia. Technical Report n° 4, Illinois, Dept. of Statistics.
- [6] Fang K.T., Wang Y. (1993). Number-theoretic methods in statistics. Chapman and Hall, London.
- [7] Owen A.B. (1992). Orthogonal arrays for computer experiments, integration and visualiazation. Statistica Sinica 2, 439-452.
- [8] Thiémard E. (2000). Sur le calcul et la majoration de la discrédance à l'origine. Thèse No 2259, Département de mathématiques, école polytechnique fédérale de Lausanne.
- [9] Hlawka E. (1961). Funktionen von Beschränkter Variation in der Theorie der Gleichverteilung. Annali di Mathematica Pura Ed Applicata, 54, 325-333.
- [10] Halton J. H. (1960). On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals, Numer. Math. 2, 84-90.
- [11] Hammersley J.M. (1960). Monte-Carlo methods for solving multivariate problems, Annals of the New York Academy of Sciences 86, 844-874.
- [12] Sobol' I. M. (1967). On the distribution of points in a cube and the approximate evaluation of integrals, USSR Comput.Math. Math. Phys. 7, No. 4, 86-112.

- [13] Faure H. (1982). Discrépance de suites associées à un système de numération (en dimension s), *Acta Arith.* 41, 337–351.
- [14] Niederreiter H. (1987). Low-Discrepancy and Low-Dispersion Sequences, *Journal of number theory*, 30, 51-70.
- [15] M.E. Johnson, L.M. Moore et D. Ylvisaker. (1990). Minimax and maximin distance design. *J. Statist. Plann. Inf.*, 26 :131–148.
- [16] H. Niederreiter. (1987). Low-discrepancy and low-dispersion sequences. *Journal of number theory*, 30 :51–70.
- [17] F. Hickernell. (1998). A generalized discrepancy and quadrature error bound. *Mathematics of computation*, 67 :299–322.
- [18] K.T. Fang, R. Li et A. Sudjianto. (2005). *Design and Modeling for computer experiments*. Chapman and Hall, Computer Science And Data Analysis Series.
- [19] M. Gunzburger et J. Burkardt. (2004). Uniformity measures for point sample un hypercubes, <http://people.sc.fsu.edu/~jburkardt/pdf/ptmeas.pdf>.
- [20] M. E. Johnson, L. M. Moore, and D. Ylvisaker. (1990). Minimax and maximin distance designs. *J. Statist. Plann. Inference*, 26(2) :131–148.
- [21] H. Niederreiter.(1992) *Random Number Generation and Quasi-Monte Carlo Methods*. SIAM, Philadelphia, pacbms-nsf regional conference series in applied math. edition.
- [22] L. Pronzato and W. G. Müller. (2012). Design of computer experiments : space filling and beyond. *Stat. Comput.*, 22(3) :681–701.
- [23] K.T. Fang, R. Li et A. Sudjianto. (2005). *Design and Modeling for computer experiments*. Chapman and Hall. Computer Science And Data Analysis Series.
- [24] E. Thiémard. (2000). *Sur le calcul et la majoration de la discr pance   l’origine*. Th se de doctorat.  cole polytechnique f d rale de Lausanne.
- [25] T.T. Warnock. (1972). Computational investigations of low discrepancy point sets. In S. K. Zaremba,  diteur *Applications of Number Theory to Numerical Analysis*, pages 319–343. Academic Press, New-York.
- [26] W.J. Morokoff et E. Caffish. (1994). Quasi-random sequences and their discrepancies. *SIAM Journal on Scientific Computing*, 15(6) :1251–1279.
- [27] M.E. Johnson, L.M. Moore et D. Ylvisaker. (1990). Minimax and maximin distance design. *J. Statist. Plann. Inf.*, 26 :131–148.
- [28] T.J. Santner, B.J. Williams et W.I. Notz. (2003). *The design and analysis of computer experiments*. SpringerVerlag, New York.
- [29] J.S. Park. (1994). Optimal latin hypercube designs for computer experiments. *J. of Statist. Planning and Inference*, 39 :95–111.
- [30] R. Bettinger.(2009). *Inversion d’un syst me par krigeage - Application   la synth se de catalyseurs   haut d bit*. Th se de doctorat, Universit  de Nice-Sophia Antipolis.
- [31] D.J. Strauss. (1975). A model for clustering. *Biometrika*, 62 :467–475.

- [32] B.D. Ripley et F.P. Kelly. (1977). Markov point processes. *Journal of The London Mathematical Society*, 15 :188–192.
- [33] J. Franco. (2008). Planification d’expériences numériques en phase exploratoire pour des codes de calculs simulant des phénomènes complexes. Thèse de doctorat, École Nationale Supérieure des Mines de Saint-Etienne.
- [34] D. Daley et V. Jones. (1988). An introduction to the theory of point processes. Springer, New York.
- [35] Daley, D.J., et D. Vere-Jones. (2002). *An Introduction to the Theory of Point Processes*. Springer Verlag.
- [36] Baddeley, A., et M.N.M van Lieshout. (1993). Stochastic geometry models in high-level vision. *Statistics and Images*, 1 : 231-256.
- [37] Ripley, B.D., et F.P. Kelly. (1977). Markov Point Processes. *Journal of the London Mathematical Society*, 188-192.
- [38] Strauss, D.J. (1975). A method for clustering. *Biometrika* 63, 467–475.
- [39] Ruegg, A. (1989). *Processus stochastiques*, Volume 6 of *Méthodes Mathématiques pour l’Ingénieur [Mathematical Methods for the Engineer]*. Lausanne : Presses Polytechniques Romandes. Avec applications aux phénomènes d’attente et de fiabilité. [With applications to queueing and reliability].
- [40] Winkler, G. (2003). *Image Analysis, Random Fields and Markov Chain Monte Carlo Methods*. Springer Verlag.
- [41] Preston, C. (1976). Spatial Birth and Death Processes. *Bulletin of the International Statistical Institute*, 371-391.
- [42] Hastings, W.K. (1970). Monte Carlo Sampling Methods using Markov Chains and Their Applications. *Biometrika*.
- [43] Green, Peter J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 57 : 97-109.
- [44] German, S., et D. German. (1984). Stochastic relaxation, Gibbs Distributions and the Bayesian Restoration of Images. *IEEE Trans. on PAMI*, 6 : 721-741.
- [45] Moller, Jesper. (1999). Notes on Markov Chain Monte Carlo Methods. Dina Research Summer School.
- [46] Metropolis, N., A. W. Rosenbluth, A. H. Teller, et E. Teller. (1953). Equation of state calculations by fast computing machines. *Journal of Chemical Physics*, 1087-1092.
- [47] Perrin, Guillaume. (2006). Etude du couvert forestier par processus ponctuels marqués. Thèse.
- [48] Baddeley, A.J. et van Lieshout, M.N.M. (1995). Area interaction point processes. *Ann. Inst. Stat. Math.*, 47, 601-619.
- [49] Ripley, B.D. et Kelly, F.P. (1977). Markov point processes. *J. London Math. Soc.*, 15, 188-192.
- [50] Baddeley, A.J. et J. Moller, J. (1989). Nearest-neighbour Markov point processes and random sets, *Int. Stat. Rev.*, 57, 90-121

- [51] van Lieshout, M.N.M. (2000). Markov point processes and their Applications, Imperial College Press, London.
- [52] Ross, S. M. (2003). Introduction to Probability Models, 8^e édition, Academic Press, San Diego.

