UNIVERSITE DES SCIENCES ET DE LA TECHNOLOGIE HOUARI BOUMEDIENNE ALGER
INSTITUT D'INFORMATIQUE

N° d'ordre : 09/95-M

THESE

Présentée à

L'USTHB

pour obtenir

LE DIPLOME DE MAGISTER

Spécialité : **INFORMATIQUE**

Par

Melle AIT KACI AZZOU SAMIRA

ETABLISSEMENT ET APPARIEMENT DE TRAJECTOIRES DANS UN SYSTEME DE VISION STEREOSCOPIQUE NON ANTHROPOMORPHE EN MOUVEMENT

Devant le jury

M^{me} DRIAS	PRESIDENTE
M^{R} S.LARABI	RAPPORTEUR
M^{R} K.ACHOUR	EXAMINATEUR
M^{R} Z.BELMESK	EXAMINATEUR
M^{R} N.DJEDI	EXAMINATEUR

Introduction Générale

Un problème commun pour l'étude des variables aléatoires spatialement distribuées est le plan d'échantillonnage pour ces variables . Différentes approches ont été proposées dans la littérature existante , la plus utilisée est basée sur des techniques géostatistiques où la dépendance spatiale est exprimée à travers le variogramme. Dans ce cas , le plan optimal est obtenu en général en minimisant l'estimation de la variance de l'erreur ou le maximum des variances estimées. (voir Cressie [2]).

D'autres fonctions peuvent être utilisées en se basant par exemple sur l'estimation de la matrice variance- covariance , comme la trace ou le déterminant (Voir Christakos [1], Cressie [2]). Une approche naturelle proposée par (Angulo, Bueso et Alonso [30]) est basée sur la théorie de l'information en utilisant l'entropie dans le cas où la variable d'intérêt ne peut pas être observer directement mais une information peut être récolter en faisant un échantillonnage.

Le problème d'échantillonnage optimal pour l'estimation de quantités aléatoires est un sujet d'actualité qui présente un intérêt théorique et pratique considérable, notamment pour des situations réelles où les quantités observées sont accompagnées par des erreurs de mesure.

Dans le cas où les erreurs sont corrélées, le problème d'échantillonnage a été traité, et des méthodologies de recherche globale ont été élaborées.

K- Boukhetala et S- Benamara [34] , 1999 ont traité le problème dans le cas scalaire par des méthodes heuristiques comme l'algorithme génétique et le recuit simulé.

K- Boukhetala et Z- Habib [33], ont confirmé l'effet de la quantification sur la vitesse de convergence des algorithmes utilisés. Ce qui a été démontré asymptotiquement auparavant, par K- Benhenni et S- Cambanis [6] . (Ils ont prouvé que le taux de convergence de l'erreur quadratique moyenne a été réduit de $n^{-2}\, a \quad n^{-1..5}$)

Après K- Boukhetala et N- Mehassouel [35] ont généralisé les travaux précédents au cas spatial en développant des algorithmes adaptés.

Dans notre travail, on propose une généralisation des travaux de K- Boukhetala et Z- Habib qui concernent l'échantillonnage optimal spatial avec quantification, par le développement d'algorithmes de recherche globale adéquats.

En général, l'échantillonnage optimal spatial se réfère à un processus qui vise un arrangement d'observations futures sur un domaine *D* donné tout en satisfaisant des conditions mathématiques**,** monétaires ou de régularité (voir Christakos [1]).

Le processus aléatoire spatial qui modélise notre processus naturel prend en compte la variabilité, l'incertitude et d'hétérogénéité du processus qui a un rôle dans la manière avec laquelle le processus est spatialement distribué à l'intérieur d'une région d'observation *D*.

Mathématiquement, l'échantillonnage spatial est une procédure sous optimale. Son application en pratique engendre un nombre de paramètres comme l'échantillonnage de points ' Sampling Pattern ', l'échantillonnage de densité et de voisinage, les mécanismes physiques et les objectifs de l'exploration. (Voir Cressie [2])

De plus les méthodes d'échantillonnage spatial peuvent être obtenues comme un résultat d'une procédure expérimentale en respectant des règles d'un bon échantillonnage. Cela se fait en général comme suit :

1. On définit les hypothèses de travail fondamentales, qui sont essentielles pour l'implémentation d'une méthode de recherche stochastique.

2. Un ensemble d'hypothèses auxiliaires, ainsi que quelque relations de dualité, qui lient les processus naturels aux mathématiques des modèles de processus aléatoires.

3. Une heuristique – un ensemble de règles méthodologiques- pour des modèles de corrélations spatiales en utilisant des procédures expérimentales dans le cas de fonctions de covariance et de semi-variogramme ordinaires , et pour des procédures automatiques lorsqu'on a affaire à des modèles généralisés .

L'heuristique développée dans notre cas est l'algorithme génétique spatial avec un codage réel des individus. Ce dernier représente une méthode efficace pour la résolution des problèmes combinatoires compliqués dans un domaine compact.
L'originalité du travail réside dans le type de codage utilisé, ainsi qu'aux opérateurs de croisement tout à fait adéquats au problème de localisation spatiale [19]. De plus la quantification des erreurs d'observation permettra d'avoir des perturbations dans le graphe de la fonction objectif, bénéfique à l'utilisation de *l'A.G*.

Cette thèse est subdivisé en quatre chapitres :

- Le premier chapitre est une introduction permettant au lecteur de se familiariser avec la notion de processus aléatoire spatial, à travers des définitions et des outils indispensables pour la modélisation et la résolution d'un problème d'échantillonnage spatial.

- Le deuxième chapitre à été consacré à décrire un algorithme de recherche global qui n'est autre que, l'Algorithme Génétique, où les études précédentes ont prouvé son efficacité dans la résolution des problèmes combinatoire compliqués, En commençant par décrire l'*AG* le plus simple i.e. avec un codage binaire jusqu'à aboutir à un *Algorithme Génétique Spatial* avec des opérateurs adaptés.

- Le troisième chapitre introduit le concept de « *Quantification* » que nous avons associé à *l'A.G.S*, afin d'améliorer sa performance de recherche globale.

- Le quatrième chapitre contient l'implémentation de l'application de localisation spatiale en se basant sur *l'A.G.S* proposé, où les résultats obtenus sont nettement améliorés par rapport aux études précédentes. L'effet de la quantification sur l'efficacité de *l'A.G.S* est approuvée.

I.0- Introduction

Les modèles spatiaux sont nouveaux dans la littérature des statistiques. Ces derniers sont utilisés dans n'importe quel domaine qui utilise des données collectées de différentes localisations spatiales, comme la génétique, l'écologie ou la biologie des populations, ce qui a motivé le développement de modèles statistiques qui indiquent quand- est ce qu'on a une dépendance entre les mesures dans les différentes localités.

En inférence statistique spatial, on a deux approches, la première est le modèle de base et la deuxième est le plan de base [3].

- La première approche commence par un modèle de processus stochastique qui est utilisé dans la prédiction après l'estimation de ces paramètres inconnus.

- La seconde se produit sur la base d'un simple échantillon aléatoire ou à partir d'autre méthodes basées sur la génération de sites d'observations. Dans ce cas l'inférence est basée sur la variabilité du plan d'échantillonnage .

l'inférence sous la deuxième approche est généralement élémentaire d'un point de vu statistique, et c'est prouvé qu'elle a plus de robustesse envers les variables d'intérêt, seulement une bonne analyse du modèle de base donne aussi de bons résultats, particulièrement, ils sont plus intéressants s'ils y a une corrélation entre les sites.

I. 1- Notions Générales

Dans ce chapitre, on introduit les notions de base de l'analyse stochastique , essentiellement celles liées aux probabilités.

▪ Définitions

Définition1 :

Soit (Ω, F, P) un espace de probabilité, Ω est l'espace fondamental, F est une σ- algèbre et P est la mesure de probabilité définie sur l'espace mesurable (Ω, F) et satisfaisant les axiomes de *Kolmogorov*.

a) $P(\Omega)=1$.

b) $0 \le P(A_i) \le 1, \quad \forall \quad A_i \in F$

c) Si $A_i \cap A_j = \varnothing, \ (i \ne j), alors \quad P(\bigcup_{i=1}^{\infty} A_i) = \sum_{i=1}^{\infty} P(A_i)$.

Les ensembles $A_i \in F$ sont appelés événement, et (Ω, F, P) est le modèle sur lequel les calculs stochastiques sont basés.

Définition2 :

Soit (Ω, F, P) un espace de probabilité. L'espace $L_p(\Omega, F, P)$, $p \ge 1$ est dit espace linéaire normé de la variable aléatoire X sur (Ω, F, P) qui satisfait la condition : $E(|X|)^p = \int |X(\varpi)|^p P(d\omega) < \infty$

La norme correspondante est définie par la formule : $\|X\| = \left(E|X|^p\right)^{1/p}$.

Remarque

Tout au long de la thèse, on suppose que les variables aléatoires satisfassent la définition 2 avec $p=2$. Ces variables aléatoires sont appelées variables de second ordre.

On note que l'espace L_2 est muni du produit scalaire : $\langle x_1, x_2 \rangle = E(x_1.x_2) = \int x_1(\omega).x_2(\omega) P(d\omega)$

Définition3 :

Soit (Ω, F, P) un espace de probabilité, et soit *(R,F)* un espace mesurable.

Un processus aléatoire spatial *X(s),* $s \in IR^n$ est une famille de variables aléatoires $(X_1, X_2, ...,)$ aux points $(s_1, s_2, ...)$

Où chaque variable X_i, $i = 1, ..., n$ est définie de (Ω, F, P) dans *(R,F).*

Définition4 :

Soit $L_2(\Omega,F,P)$ un espace de *Hilbert* de variables aléatoires *X(s),* $s \in IR^n$.
Un processus aléatoire spatial *X(s)* est défini par l'application :

$$X : IR^n \rightarrow L_2(\Omega, F, P)$$

Les données spatiales représentent les observations du processus aléatoire spatial $\{X(s)\ s \in D\}$ où $D \subset R^n$.

Définition5 :

Un processus aléatoire spatial vectoriel (ou multivarié) *X(s),* $s \in IR^n$ est un ensemble de processus aléatoires spatiaux $X_1(s), X_2(s), \ldots X_k(s)$ qu'on note :

$$X(s)=[X_1(s), X_2(s), \ldots X_k(s)]^T$$

Oú: $X_1(s), X_2(s), \ldots X_k(s)$ sont les composantes du processus aléatoire spatial multivarié *X(s).*

La spécification du processus aléatoire spatial multivarié, est faite par analogie avec le cas univarié. i.e. toutes les densités de probabilités multivariés des composantes $X_1(s), X_2(s), \ldots X_k(s)$ doivent être connues.

I.2-Caractérisation des processus aléatoires spatiaux par leurs moments et leur fonction de covariance :

I.2.1- les moments d'un processus aléatoire spatial :

Soit *X(s),* $s \in IR^n$ un processus aléatoire spatial.

- La moyenne de la variable aléatoire *X(s)* : $m_X(s)$ est définie par :

$$m_X(s) = E(X(s)) = \int x f_s(x) dx$$

- La fonction de covariance du processus aléatoire *X(s)* est donnée par :

$$C_X(s,s') = E([X(s) - m_X(s)][X(s') - m_X(s')]) = \iint (x - m_X(s))(x' - m_X(s'))dx \quad dx'$$

$C_X(s,s')$ est appelée aussi moment centré d'ordre 2 du processus aléatoire *X(s)*.

Connaissant la fonction de covariance spatiale, on peut définir :

i) *la variance* : $\sigma_X^2 = C_X(s,s)$ qui décrit les variations aléatoires locales.

ii) *La fonction de corrélation spatiale* : $\rho_X(s,s') = \dfrac{C_X(s,s')}{\sqrt{\sigma_X^2(s)}\sqrt{\sigma_X^2(s')}}$.

Un processus aléatoire spatial est dit non corrélé si :

$$C_{.X}(s,s') = \begin{cases} \sigma_X^2 \quad si \quad s = s' \\ 0 \quad \text{sinon} \end{cases}$$

Ce processus est appelé aussi : *Bruit blanc spatial*.

I.2.2- Propriétés de la Fonction de Covariance Spatial :

1) Une fonction $C_X(s,s')$ est une fonction de covariance spatiale (ou $C_X(s,s') \in C_n$) si et seulement si : elle est définie non négative : $\sum_{i=1}^{m}\sum_{j=1}^{m} q_i q_j C_X(s_i,s_j) \geq 0 \quad \forall m \in IR_+^*, \quad \forall s_i, s_j \in IR^n$ et $\forall q_1, q_2, ..., q_n$ des réels (ou des complexes). Ce qui est une conséquence directe de :

$$E\left[\sum_i q_i (X(s_i) - m_X(s_i)\right]^2 = \sum_{i=1}^{m}\sum_{j=1}^{m} q_i q_j \{E[X(s_i) - m_X(s_i)][X(s_j) - m_X(s_j)]\} \geq 0$$

2) Toute covariance $C_X(s,s')$ est une fonction symétrique i.e. $C_X(s,s') = C_X(s',s) \in C_n$

3) On a $\lim_{|s-s'| \to \infty} C_X(s,s') = 0$

4) la classe C_n des covariances est fermée, pour l'addition, la multiplication, et au passage à la limite : en d'autres termes , si $\{C_{X,k}(s,s')\} \in C_n, k = 1.2....$ alors :

$$\sum_{\rho \leq k} C_{X,\rho}(s,s') \in C_n \quad \text{et} \quad \lim_{k \to \infty} C_{X,k}(s,s') = C_X(s,s') \in C_n$$

En supposant que la limite existe pour chaque couple (s,s').

I.3- Processus aléatoires Spatiaux Multivariés : [1]

I.3.1- Les moments d'un processus aléatoire spatial multivarié

Soit $X_1(s) \quad et \quad X_2(s'), \quad s,s' \in IR^n$ deux processus aléatoires spatiaux.

La *fonction de covariance croisée spatiale* est définie comme suit :

$$C_{X_1X_2}(s,s') = E[[X_1(s) - m_{X_1}(s)][X_2(s') - m_{X_2}(s')]$$
$$\iint [x_1 - m_{X_1}(s)][x_2 - m_{X_2}(s')] f_{s,s'}(x_1,x_2) dx_1 dx_2$$

où $f_{s,s'}(x_1,x_2)$ est la densité de probabilité jointe de $X_1(s) \quad et \quad X_2(s'), \quad s,s' \in IR^n$

Le coefficient de corrélation croisé est donné par :

$$\rho_{X_1,X_2}(s,s') = \frac{C_{X_1,X_2}(s,s')}{\sqrt{\sigma^2_{X_1}(s)}\sqrt{\sigma^2_{X_2}(s')}}$$

Où $\sigma^2_{X_1}$ *et* $\sigma^2_{X_2}$ sont respectivement les variances des processus aléatoires spatiaux

$X_1(s)$ *et* $X_2(s')$,

La définition du coefficient de corrélation spatiale $\rho_{X_1,X_2}(s,s')$, peut être généralisée à tout ordre k>2 .

Soit $X(s)=[X_1(s),....,X_k(s)]^t$ un vecteur de processus aléatoire spatiaux, la matrice de covariance correspondante, entre les composantes du vecteur *X*, est donnée par :

$$C_X = \left(C_{X_p,X_l}(s_i,s_j) \right) \quad \forall i,j = 1,2,...m. \qquad \forall p,l \in 1,2,...,k.$$

I.3.2 – Propriétés de la Fonction de Covariance Croisée :

La fonction de covariance croisée possède des propriétés importantes dont on cite :

♦ La matrice C_X est définie non négative, tel que : $q^t C_X q \geq 0$, $\forall \quad q^t = [q_1, q_2, ..., q_k]$
Ceci est dû au fait que $Var[q^t X(s)] \geq 0$.

♦ La matrice C_X est symétrique : $C_{X_p,X_l}(s_i,s_j) = C_{X_p,X_l}(s_j,s_i)$.

♦ Une application directe de l'inégalité de Schwartz sur C_X nous conduit à l'inégalité suivante : $\left| C_{X_p,X_l}(s_i,s_j) \right| \leq \sqrt{\sigma^2_{X_p}(s_i)}\sqrt{\sigma^2_{X_l}(s_j)}$ $\forall p,l=1,2,...,k$ *et* $i,j=1,2,...,m$

♦ Soit $X=[x_1,x_2,....,x_m]_t$ *et* $Y=[y_1,y_2,....,y_m]_t$ des vecteurs observations respectivement des deux processus spatiaux *X(s)* et *Y(s)*. Alors :

a) La moyenne conditionnelle de *X* sachant $Y=\Psi=(\psi_1,\psi_2,...,\psi_m)$ *:* le vecteur déterministe est donnée par : $m_{X/\Psi} = E[X/\Psi] = \int_{IR^m} x \quad f_{X/Y}(x/\Psi)dx$.

b) La covariance conditionnelle de *X* sachant Ψ est donnée par :

$$C_{X/\Psi} = E\left\{ [X - m_{X/\Psi}][X - m_{X/\Psi}]^t / \Psi \right\} \quad = \quad \iint_{IR^m} [x - m_{X/\Psi}][x - m_{X/\Psi}]^t f_{X/Y}(x/\Psi)dx$$

Remarque : si *Y* est considéré comme vecteur aléatoire alors :

$$m_{X/Y} = \int_{IR^m} x \quad f_{X/Y}(x/Y)dx \quad \text{et} \quad C_{X/Y} = \int_{IR^m} [x - m_{X/Y}][x - m_{X/Y}]^t f_{X/Y}(x/Y)dx$$

I.4 Certaines Propriétés Géométriques des Processus Aléatoires Spatiaux

I.4.1- La Convergence Stochastique

Définition : soit $\{X_n\}$ une suite de variables aléatoires de $L_2(\Omega, F, P)$.
$\{X_n\}$ converge vers la variable aléatoire X :

i) *En moyenne quadratique* : $X_n \xrightarrow{mq} X \quad si \quad \lim_{n\to\infty} E(X_n - X)^2 = 0.$

ii) *Presque sûrement* : $X_n \xrightarrow{ps} X \quad si \quad P(\lim_{n\to\infty} X_n = X) = 1.$

iii) *En probabilité P* : $X_n \xrightarrow{P} X \quad si \quad \forall \varepsilon > 0 \quad \lim_{n\to\infty} P(|X_n - X| > \varepsilon) = 0.$

iv) *En distribution* : $X_n \xrightarrow{F} X \quad si \quad \lim_{n\to\infty} F_{X_n}(x) = F_X(x).$

Les types de convergence sont liés entre eux par le schéma suivant :

$$\left.\begin{array}{l} X_n \xrightarrow{m.q} X \\ X_n \xrightarrow{p.s} \end{array}\right\} \Rightarrow X_n \xrightarrow{P} X \Rightarrow X_n \xrightarrow{F} X.$$

Proposition 1 : Si $X_n \xrightarrow{m.q} X \quad et \quad Y_n \xrightarrow{m.q} Y$ alors : $E(X_n) \to E(X) \quad et \quad E(X_n Y_n) \to E[XY]$.

Un processus aléatoire spatial *X(s)* est dit convergeant vers $X(s_0)$ lorsque $s \to s_0$ en un sens parmi ceux cités auparavant, si la suite de variables aléatoire $\{X_n = X(s)\}$ au point $s = s_1, s_2, ...s_n, ...$

Tend vers $X_0 = X(s_0) \quad quand \quad n \to \infty$.

Pour la théorie des processus aléatoires spatiaux, la convergence au sens ˝ moyenne quadratique ˝ et au sens ˝ presque sûr ˝ sont les plus utilisées.

Proposition 2 : (*Le critère de convergence m.q*) :

Soit *X(s)* un processus aléatoire spatial et soit s_0 un point fixe dans IR^n. Alors
$X(s) \xrightarrow{m.q} X(s_0)$, si on a toujours $E[X(s)X(s')] \longrightarrow E[X(s_0)]^2$
Lorsque $s, s' \to s_0$.

I.4.2- La Continuité Stochastique

Définition 1 : Un processus aléatoire spatial *X(s)* est dit continu en moyenne quadratique (*m.q*) au point $s \in IR^n$. Si $X(s') \xrightarrow{mq} X(s) \quad quand \quad s' \to s$.

Ou bien : $X(s+h) \xrightarrow{m.q} X(s) \qquad lorsque \quad h \to 0$

Proposition 1 :

Un processus aléatoire spatial *X(s)* est dit continu en moyenne quadratique, si et seulement si la fonction de covariance $C_X(s,s')$ est continue au point $(s,s'=s)\in IR^n\times IR^n$. *X(s)* est partout continu si $C_X(s,s')$ est continue aux points diagonales $(s,s'=s)\in IR^n\times IR^n$.

Définition 2 :

Un processus aléatoire spatial *X(s)* est dit continu presque sûrement au point $s\in IR^n$. Si $X(s+h)\xrightarrow{ps}X(s)$ *lorsque* $h\to 0$.

- On peut montrer que si on a : $E\left|X(s+h)-X(s)\right|^{\lambda}\le\dfrac{\alpha|h|^{2n}}{\left|\log|h|\right|^{1+\beta}}$

Où α est une constante positive, $\lambda>0$, et $\beta>\lambda$, alors le processus aléatoire spatial *X(s)*, $s\in IR^n$ est presque sûrement continu (*p.s*) sur tout ensemble compact $C\subset IR$.

Proposition 2 :

Soit $C_X(s,s')$ la fonction de covariance du processus *X(s)*, $s\in IR^n$.
Si pour tout $s,h\in C$, on a :

$$C_X(s+h,s+h)-C_X(s+h,s)-C_X(s,s+h)+C_X(s,s)\le\frac{\alpha|h|^{2n}}{\left|\log|h|\right|^{1+\beta}}$$

Où : $\beta>2$, alors *X(s)* est presque sûrement continu.

I. 4.3- La Diffèrentiabilité Stochastique :

Définition 1 :

Un processus aléatoire spatial du second ordre est différentiable en moyenne quadratique pour tout s_i de points $(s_1,s_2,....,s_n)\in IR^n$, s'il existe un processus spatial $X_{(i)}(s)$ tel que :

$$\frac{X(s+h\varepsilon_i)-X(s)}{h}\xrightarrow{mq}X_{(i)}(s)\quad lorsque\quad h\to 0.$$

Où : ε_i est le vecteur unité dans la direction i : c'est le vecteur dont le *ième* élément est égal à 1 et tous les autres sont égaux à 0 ($1\le i\le n$).

Le processus aléatoire spatial $X_{(i)}(s)=\partial X(s)/\partial s_i$ est appelée la *ième* dérivée partielle de *X(s)* au point *s*. On le note aussi :

$$\frac{\partial X(s)}{\partial s_i}=\lim_{h\to 0}\frac{X(s+h\varepsilon_i)-X(s)}{h}$$

On peut définir un ordre supérieur de diffèrentiabilité $\partial^p X(s)/\left(\partial s_{i_1},\partial s_{i_2},....,\partial s_{i_p}\right)$, de la même manière. Par exemple, le second ordre de i , la *jème* dérivée partielle est donnée par :

$$\frac{\partial^2 X(s)}{\partial s_i\partial s_j}=\lim_{h,r\to 0}\frac{1}{hr}\left[X(s+h\varepsilon_i+r\varepsilon_i)-X(s+h\varepsilon_i)-X(s+r\varepsilon_i)+X(s)\right]\quad \text{Où : } i\ge 1,\ j\le n.$$

Proposition :

Un processus aléatoire spatial *X(s)* est différentiable en *m.q* si et seulement si :

a) *E(X(s))* est différentiable.

b) La covariance :

$$Cov(\frac{\partial X(s)}{\partial s_i}, \frac{\partial X(s)}{\partial s_i'}) = \frac{\partial^2 C_X(s,s')}{\partial s_i \partial s_i'}$$

$$= \lim_{h,r\to\infty} \frac{1}{hr}\left[C_X(s+h\varepsilon_i, s'+r\varepsilon_i) - C_X(s, s'+r\varepsilon_i) - C_X(s+h\varepsilon_i, s') + C_X(s,s')\right]$$

est finie pour tout point *(s,s)*.

Remarque : La différentiabilité en moyenne quadratique d'un processus aléatoire spatial *X(s)* au point $s \in IR$ implique la continuité en *m.q* du processus *X(s)* au point $s \in IR$

Définition 2 :

Un processus aléatoire spatial est différentiable presque sûrement, s'il existe un processus spatial tel que :

$$\frac{X(s+h\varepsilon_i) - X(s)}{h} \xrightarrow{p.s} X_{(i)}(s) \quad \text{lorsque} \quad h \to 0.$$

qui est appelée la différentiabilité des réalisations du processus.

On suppose que la dérivée du processus aléatoire spatial *X(s)* est :

$$X_{(i)}(s) = \frac{\partial X(s)}{\partial s_i}$$

et que la covariance du processus aléatoire spatial *X(s)* est donnée par :

$$C_{X_{(i)}}(s,s') = \frac{\partial^2 C_X(s,s')}{\partial s_i \partial s_i'}$$

Alors, en utilisant le résultat de la (*proposition 2* du chapitre *I.4.2),* On conclut que :

Le processus $X_{(i)}(s)$ est presque sûrement continu si :

$$C_{X_{(i)}}(s,s+h) - C_{X_{(i)}}(s+h,s) - C_{X(i)}(s,s+h) + C_{X_{(i)}}(s,s) \le \frac{\alpha|h|^{2n}}{\left|\log|h|\right|^{1+\beta}}.$$

Où : $\beta > 2$, pour tout $s,h \in C$.

I.4.4- Intégration Stochastique :

Définition :

Soit $X(S)$ un processus aléatoire spatial. L'intégrale $\tau(w)=\int_U \alpha(w,s)X(s)ds.$ où : $U \subset \Re^n$.

($\alpha(w,s)$: est une fonction continue déterministe bornée).

L'intégrale $\tau(w)$ existe en moyenne quadratique, si le limite de la somme $\tau_m(w)=\sum_{i=1}^{m}\alpha(w,s)X(s_i)\Delta(s_i)$ existe , lorsque $m\to\infty$ aussi, en m.q.

Cette limite est appelée l'intégrale de Riemann du processus $X(S)$.

$\Delta(s_i)\to 0$, $\sum_{i=1}^{m}\Delta(s_i)=U$; $\tau(w)$ et $\tau_m(w)$ sont des variables aléatoires .

On a : $\tau(w)=\lim_{m\to\infty}\tau_m(w)$ et le processus aléatoire spatial $X(s)$ est dit intégrable en moyenne quadratique.

Proposition :

Un processus aléatoire spatial *X(s)* est intégrable en moyenne quadratique au sens Riemann si et seulement si :

$$E[\tau^2(w)]=\int_V\int_V \alpha(w,s)\overline{\alpha(w,s')}C_X(s,s')dsds' < \infty$$

Où : $\overline{\alpha(w,s)}$: est le conjugué du nombre complexe $\alpha(s,w)$, et $V \subset IR^n$

I.5- Les Processus Spatiaux aléatoires Gaussiens :

La famille des processus aléatoires spatiaux gaussiens est très importante pour des raisons multiples, tels que :

- Ils ont des propriétés mathématiques communes, et approximent beaucoup de phénomènes naturels.
- En utilisant le théorème central limite, les processus gaussiens sont la limite d'un grand nombre de processus non gaussiens.

I.5.1 - Définition :

Un processus aléatoire spatial gaussien *X(s)* est un processus aléatoire spatial dont la fonction de densité de probabilité du vecteur aléatoire $X=(X_1,X_2,....,X_m)'$, $\forall m \geq 1$ est donnée

$$\text{Par :} \quad f_X(x)=\frac{1}{(2\pi)^{m/2}|C_X|^{1/2}}\exp\left[-\frac{(x-m_X)'C_X^{-1}(x-m_X)}{2}\right]$$

Où : $X=(X_1,X_2,....,X_m)'$ et $C_X=\{\ C_X(s_i,s_j),\ \ i,j=1,2,...,m\ \}$

I.5.2 – Propriétés des Processus aléatoires Spatiaux Gaussiens

Un processus aléatoire spatial gaussien possède les propriétés suivantes :

- *X(s)* est complètement caractérisé par ses moments d'ordre 1 et 2.
- Si *X*(s) et *Y(s)* sont deux processus aléatoires spatiaux gaussiens alors :

  La densité de probabilité conditionnelle du processus *X* sachant *Y* est aussi gaussienne. Cette densité est donnée par :

$$f_{X/Y}(X/\Psi)=\frac{1}{(2\pi)^{m/2}\left|C_{X/\Psi}\right|^{1/2}}\exp\left[-\frac{(x-m_{X/\Psi})'C_{X/\Psi}^{-1}(x-m_{X/\Psi})}{2}\right] \text{ avec } \left|C_{X/\Psi}\right|=\det\left(C_{X/\Psi}\right)$$

Où : $m_{X/\Psi}=C_{XY}C_Y^{-1}\left[\Psi-m_Y\right]+m_X$ et $C_{X/\Psi}=C_X-C_XC_YC_{XY}^T$

- Le processus aléatoire spatial gaussien *X(s)* est indépendant, si et seulement si il est non corrélé.

I.5.3- Le Théorème Central Limite

Le théorème central limite est, probablement, le théorème le plus utilisé dans la théorie des probabilités. i.e. sous certaines conditions, la distribution de probabilité (typiquement , la distribution d'une somme d'un grand nombre de variables aléatoires indépendantes) peut être approchée par une distribution de probabilité Gaussienne.

Soit $\{X_k\}$, $k=1,2,.....$ Une suite de variables aléatoires indépendantes ayant comme moyennes m_k, et de variances $\sigma_k^2\neq 0,\ et\ \ E\left|X_k-m_k\right|^3<\infty$.

Soit $\left\{Y_n=\dfrac{\sum_{i=1}^{n}(X_k-m_k)}{\sqrt{\sum_{k=1}^{n}\sigma_k^2}}\right\}$ une suite de variables aléatoires avec des distributions de probabilité $\{F_n(\Psi)\}$.

Si $\lim_{n\to\infty}\dfrac{\sqrt{\sum_{i=1}^{n}E\left|X_k-m_k\right|^3}}{\sqrt{\sum_{k=1}^{n}\sigma_k^2}}=0$, alors $\{F_n(\Psi)\}$ tend vers une loi gaussienne *N(0,1)* quand $n\to\infty$

I.6- Hypothèses auxiliaires : [1]

L'étude des processus spatiaux moyennant leurs moments d'ordre 2 exige des hypothèses auxiliaires pour que le modèle purement mathématique du processus aléatoire spatial soit compatible avec le phénomène qu'il décrit et au même temps, que le modèle soit applicable sous certaines hypothèses nécessaires pour l'inférence.

I.6.1- L' homogénéité :

Le processus naturel est modélisé par un processus aléatoire spatial *homogène au sens large*.

Ceci implique que : $m_X(s) = m \quad et \quad C_X(s,s') = C_X(h = s - s')$.

i.e. que sa moyenne est constante et sa covariance dépend toujours de la distance vectorielle entre 2 points de l'espace.

La propriété d'homogénéité du processus aléatoire spatial *X(s)* est considéré comme une fonction à valeurs dans l'espace de *Hilbert* $L_2(\Omega, F, P)$, revient au fait qu'il existe un sous espace linéaire fermé H engendré par la variable aléatoire X dans $L_2(\Omega, F, P)$ un groupe d'opérateurs U_k, tel que : $U_k X(s) = X(s+h)$, où $s,h \in IR^n$.

<u>*Remarques*</u> :

- Dans le cas unidimensionnel, où le processus aléatoire spatial devient un processus aléatoire, l'homogénéité est équivalente à l'hypothèse de stationnarité (dans le cas des séries chronologiques) .

- En général, un processus aléatoire spatial *X(s)* est dit homogène au sens stricte si la distribution de probabilité de la suite $X(s_1), X(s_2), ..., X(s_k)$, pour $k \in IR$ reste la même lorsque tous les points $s_1, s_2, ..., s_k$ soient translatés par un vecteur h$\in IR^n$.

  - L'homogénéité au sens stricte est rarement utilisée dans les situations pratiques. Par la suite, l'homogénéité au sens large d'un processus aléatoire spatial va être appelée ″ Homogénéité ″.

I.6.2- L'isotropie

Un processus naturel est modélisé par un processus aléatoire spatial i*sotropique,* (*au sens large*), dans le cas où le processus aléatoire spatial *X(s)* vérifie :

$E(X(s)) = m$ et $C_X(s,s') = C_X(r = |s - s'|)$

Ceci veut dire que la covariance dépend de la longueur de la distance vectorielle entre deux points de l'espace .

Une troisième hypothèse est essentielle dans le cas où les hypothèses 1 et 2 ne sont pas applicables, i.e. lorsque le processus aléatoire spatial étudié n'est pas homogène.

I.6.3- Processus Spatial Intrinsèque

Un processus naturel est modélisé par un processus aléatoire spatial avec des *incréments homogènes* d'ordre ν noté où processus aléatoire spatial *intrinsèque* d'ordre ν.

Ça veut dire que si le processus aléatoire spatial *X(s)* est non homogène, il existe une transformation linéaire *T*, tel que : *Y(s)=T(X(s)),* qui est un processus homogène.

On peut montrer, dans ce cas, que la fonction de covariance non homogène $C_X(s_i, s_j)$ de *X(s)* est composée d'une partie homogène $K_X(h)$, $h = s_i - s_j$, appelé covariance spatial généralisé d'ordre ν.(*Matheron*, 1973), et une partie polynomiale d'ordre ν en fonction de s_i, s_j et $P_\nu(s_i, s_j)$.

Remarques :

1) Le mot généralisé est employé pour distinguer $K_X(h)$ par rapport aux autres covariances.
2) Un processus aléatoire spatial homogène est aussi un processus intrinsèque d'ordre ν pour une valeur quelconque de ν, mais le contraire est généralement faux.

I.7- Processus Spatiaux Homogènes :

Soit *X(s)* un processus aléatoire spatial homogène.

$\rho_X(h) = \dfrac{C_X(h)}{C_X(0)}$: est la fonction de corrélation homogène correspondante à $C_X(h)$

Alors, les propriétés suivantes sont vraies :

I.7.1- Propriétés des Processus Spatiaux Homogènes :

- Les fonctions de covariance et de corrélation sont des fonctions symétriques, i.e :

  $C_X(h) = C_X(-h)$ et $\rho_X(h) = \rho_X(-h)$.

  De plus, on a : $C_X(h) \le C_X(0)$ *et* $|\rho_X(h)| \le 1$.

- Quand $h \to \infty$, on a : $\lim\limits_{|h|\to+\infty} \dfrac{C_X(h)}{|h|^{(1-n)/2}} = 0$.

- Si $\rho_X(h) \in H_{n,0}$ ($H_{n,0}$: est la classe des fonctions de corrélation qui sont continues partout, sauf peut être à l'origine) .

  Alors : $\rho_X(h) = \alpha\delta(h) + \beta\rho_b(h)$ ……. (*) où : b est une certaine variable spatiale.

- $\delta(h)$: indice de *Kronecker*.

- $\rho_b(h) \in H_{n,c}$ ($H_{n,c}$: est la classe des fonctions de corrélation homogènes dans IR^n et qui sont continues partout).

- $\alpha, \beta \geq 0$.

C'est le cas qu'on appelle " *nugget- effect* " en géostatistique.

L'équation (*) implique que le processus aléatoire spatial *X(s)* avec $\rho_X(h) \in H_{n,0}$ peut être écrit comme : $X(s) = X_\alpha(s) + X_b(s)$

Où : $X_\alpha(s)$ est la composante ' choatic' et $X_b(s)$ est la composante de *X(s)* continue en *m.q*.

- Si $\rho_X(h) \in H_{n,c}$, alors : $\rho_X(h) = E\{\exp[i.w.h]\}$ $(i^2 = -1)$.

Où *w* est un vecteur aléatoire d'ordre n.

$\rho_X(h)$: peut être considérée comme la fonction caractéristique de *w*.

I.7.2- La Géométrie d'un Processus Spatial Homogène :

On va voir que la géométrie d'un processus aléatoire spatial homogène dépend du comportement de la covariance à l'origine (*h=0*).

I.7.2.1- Proposition :

Soit *X(s)* un processus aléatoire spatial homogène.

a) Si $C_X(h)$ est continue au point (*h=0)* , alors il est continu partout dans IR^n , d'où *X(s)* est continu en *m.q*.

b) Si on a : $Cov\left(\frac{\partial^\nu X(s)}{\partial s_{i_1}\partial s_{i_2}....\partial s_{i_p}}, \frac{\partial^\nu X(s)}{\partial s'_{i_1}\partial s'_{i_2}....\partial s'_{i_p}}\right) = \frac{(-1)^\gamma \partial^{2\gamma} C_X(h)}{\partial h_{i_1}^2 \partial h_{i_2}^2 ....\partial h_{i_p}^2}$ existe et elle est finie au point *h=0*, alors la dérivée partielle $\frac{\partial^\gamma X(s)}{\partial s_{i_1}\partial s_{i_2}....\partial s_{i_p}}$ existe en *m.q*.

c) Si $C_X(h)$ n'est pas continue au point *h=0 (nugget- effect*), le processus *X(s)* devient non continu en *m.q* en tout point *s* de IR^n

Un cas particulier de b), $\frac{\partial^\gamma X(s)}{\partial^\nu s_i}$ existe en moyenne quadratique si et seulement si :

$(-1)^\gamma \left(\frac{\partial^{2\gamma} C_X(h)}{\partial h_i^{2\gamma}}\right)$ existe et finie au point *h=0*.

Un résultat important est introduit par le corollaire suivant :

I.7.2.2- Corollaire :

Le processus aléatoire spatial *X(s)* est différentiable en *m.q* (i.e : $\frac{\partial^\gamma X(s)}{\partial^\nu s_i}$ existe en *m.q*)

Si et seulement si : $\frac{\partial^{2\gamma-1} C_X(h)}{\partial h_i^{2\gamma-1}} \Big|_{h=0} = 0.$

I.7.3- Exemples :

- On considère la covariance spatial $C_X(h) = \exp\left[-\frac{|h|^2}{a^2}\right]$ où $a>0$. est continue au point h=0, où toutes ses dérivées sont définies et finie.

  Donc, le processus spatial associé *X(s)* est continu et différentiable à n'importe quel ordre.

- La covariance unidimensionnel : $C_X(h) = \frac{\partial C_X(h)}{\partial h}\Big|_{h=0} \neq 0.$

  Dans ce cas, le processus aléatoire *X(s)* est continu en *m.q* mais il n'est pas différentiable.

I.8- Processus Spatiaux Isotropiques :

I.8.1- Définition :

Un processus aléatoire spatial est dit *isotropique* si sa fonction de covariance s'écrit en fonction de la distance $r = |h|$ seulement.

$$C_X(h) = C_X(r).$$

I.8.2- Exemples :

Les covariances usuelles d'un processus aléatoire spatial *isotropique* sont :

i) $C_X(r) = C_X(0).\exp\left[-\frac{r^2}{a^2}\right]$ où $a>0$. (*covariance gaussienne*).

ii) $C_X(r) = C_X(0).\exp\left[-\frac{r}{a}\right]$ où $a>0$ *(covariance exponentielle).*

iii) $C_X(r) = \begin{cases} C_X(0)\left[1-\frac{3r}{2a}+\frac{r^3}{2a^3}\right] & si\ r\in[0,1] \quad (\text{ covariance sphérique }) \\ 0 \end{cases}$

Remarque :

Les propriétés géométriques d'un processus aléatoire spatial isotropique sont les conséquences immédiates des propriétés d'un processus spatial homogène.
Par exemple, on a vu que *X(s)* est un processus spatial homogène est continu si et seulement si $C_X(h)$ est continu au point *h=0*, devient *X(s)* est un processus spatial isotopique est continu si et seulement si $C_X(r)$ est continu au point r=*0.*

I.9- Processus spatiaux à incréments homogènes

Les incréments de *X(s)* est définie par : $Y_h(s) = X(s+h) - X(s)$ est une extension des hypothèses de stationnarité (dans *IR*) et d'homogénéité (dans IR^n).

On va étudier en détail l'incrément du processus spatial $Y_h(s) = X(s+h) - X(s)$ en supposant que *X(s)* et $Y_h(s)$ sont continues en *m.q*.

On a : $m_{Y_h}(s) = E\left[X(s+h) - X(s)\right] = m_X(s+h) - m_X(s)$. ….(1)

La fonction de covariance est donnée par :

$$C_{Y_h} = E\left[Y_{h_i}(s_i).Y_{h_j}(s_j)\right] = E\left[[X(s_i+h_i) - X(s_i)][X(s_j+h_j) - X(s_j)]\right]$$

$$= D_X\left[s_i, s_j, h_i, h_j\right] \qquad \text{……..(2)}$$

Les équations 1) et 2) établissent les principales dualités entre les processus spatiaux *X(s)* et $Y_h(s)$.

La fonction $D_X\left[s_i, s_j, h_i, h_j\right]$ est appelée *" fonction de structure "* du processus spatial *X(s)* et elle est liée à la fonction de covariance par :

$$D_X(s_i, s_{j;} h_i, h_j) = C_X(s_i + h_i, s_j + h_j) - C_X(s_i + h_i, s_j) - C_X(s_i, s_j + h_j) + C_X(s_i, s_j) \quad \text{…… (3)}$$

On définit l'équation : $\gamma_X(s_i, s_j; h_i, h_j) = \frac{1}{2} D_X(s_i, s_j; h_i, h_j)$ ….(4).

En supposant que : $s_i = s_j = s \quad et \quad h_i = h_j = h.$

L'équation (4) devient : $\gamma_X(s,h) = 1/2.E\left[X(s+h) - X(s)\right]^2$.

γ_X : est appelé *semi-variogramme* de *X(s)* et sa moyenne est constante. Si sa moyenne n'est pas constante, on aura :

$$\gamma_X(s,h) = \frac{1}{2} Var\left[X(s+h) - X(s)\right]^2$$

On a : $2\gamma_X(s_i, s_{j;} h_i, h_j) = \gamma_X(s_i + h_i, s_j + h_j) - \gamma_X(s_i + h_i, s_j) - \gamma_X(s_i, s_j + h_j) + \gamma_X(s_i, s)$

I.9.1- Définition :

Un processus aléatoire spatial *X(s)* est dit à incréments homogènes, si sa moyenne et son semi- variogramme dépendent du vecteur différence *h*, tel que :

$$m_{Y_h} = E(Y_h(s)) = m_X(h) \qquad \text{et} \qquad \gamma_X(s,h) = \frac{1}{2} E\left[Y_h(s)\right]^2 = \frac{1}{2} C_{Y_h}(0) = \gamma_X(h) \qquad \forall h.$$

Remarque : Dans le cas d'un processus spatial homogène *X(s)*, on a :

$$m_X(h) = 0 \quad et \quad \gamma_X(h) = C_X(0) - C_X(h) \qquad \forall h.$$

I.9.2- Proposition :

Soit *X(s)* un processus aléatoire spatial non homogène à incréments homogènes, alors :

$$m_X(h_i + h_j) = m_X(h_i) + m_X(h_j) \qquad \ldots\ldots (5)$$

et $\gamma_X(h_i + h_j) \le \gamma_X(h_i) + \gamma_X(h_j) + 2\sqrt{\gamma_X(h_i)}.\sqrt{\gamma_X(h_j)}$

preuve :

On a : $E\left[X(s+h_i+h_j) - X(s+h_j)\right]^2 = 2\gamma_X(h_i)$

et : $E\left[X(s+h_j) - X(s)\right]^2 = 2\gamma_X(h_j)$

$$E\left[X(s+h_i+h_j) - X(s+h_j)\right]\left[X(s+h_j) - X(s)\right] = \frac{1}{2}\begin{Bmatrix} E\left[X(s+h_i+h_j) - X(s)\right]^2 - E\left[X(s+h_i+h_j) - X(s+h_j)\right]^2 \\ - E\left[X(s+h_j) - X(s)\right]^2 \end{Bmatrix}$$

$$= \gamma_X(h_i + h_j) - \gamma_X(h_i) - \gamma_X(h_j).$$

En appliquant l'inégalité de *Schwartz,* on obtient :

$$\gamma_X(h_i + h_j) - \gamma_X(h_i) - \gamma_X(h_j) \le \sqrt{2.\gamma_X(h_i)}.\sqrt{2.\gamma_X(h_j)}$$

Remarque :

$m_X(h)$ est continu en *h* et la condition de linéarité (équation (5)) entraîne :

$$m_X(h) = E\left[X(s+h) - X(s)\right] = a.h$$

Où : *a* est un vecteur de coefficients de IR^n.

En géostatistique, $m_X(h)$ est appelé *" drift "* du processus aléatoire spatial *X(s).* Si *a=0,* on dit que *X(s)* est *" non drift "*

I.10- Processus Aléatoire Spatial Généralisé :

Dans ce chapitre, les processus aléatoires spatiaux vont être considérés comme continus ou discrets, de même pour l'argument *S* qui peut prendre des valeurs discrètes ou continues.

Soit *Q* un espace linéaire spécifié aux éléments *q* et soit $H_2 = L_2(\Omega, F, P)$ un espace de *Hilbert* de toutes les variables aléatoires $X(q)$ sur *Q*, tel que :

i. Muni un produit scalaire : $\left\langle X(q_1), X(q_2) \right\rangle = E\left(X(q_1).X(q_2)\right) = \int\int x_1 x_2 dF_X(x_1 x_2)$, où : $F_X(x_1, x_2)$: est la distribution de probabilité conjointe des variables aléatoires $X(q_1), X(q_2)$, avec $\|X(q)\|^2 = E|X(q)|^2 < \infty$.

ii. Satisfaisant la condition de linéarité : $X\left(\sum_{i=1}^{N}\lambda_i q_i\right)=\sum_{i=1}^{N}\lambda_i X(q_i)$, pour tout $q_i \in Q$; $\forall\, \lambda_i \in \Re$ (ou $\underline{\subset}$), $i=1,2,...,N$, ($q \in Q \subset \Re^n$.)

Définition :

Soit $L_2(\Omega,F,P)$ un espace de *Hilbert* des variables aléatoires de $\Re^n$, un processus aléatoire spatial généralisé, $X(q)$, est la variable aléatoire donné par $X: Q \to L_2(\Omega,F,P)$.

On considère que le processus aléatoire spatial généralisé continu et que :

$E\left|X(q_n)-X(q)\right|^2 \to 0$ quand $q_n \xrightarrow{n\to\infty} q$.

Les caractéristiques du second ordre du processus aléatoire spatial généralisé sont :

$$m_X(q)=E[X(q)]=\int x dF_X(x) \text{ (Appelée valeur moyenne)}$$
$$\text{et : } C_X(q_1,q_2)=E\left[X(q_1)-m_X(q_1)\right]\ \left[X(q_2)-m_X(q_2)\right].$$

La moyenne et la covariance fonctionnelle doivent être des valeurs réelles et continues dans le sens suivant : $m_X(q_n)\to m_X(q)$ lorsque $q_n \xrightarrow{n\to\infty} q$ et $C_X(q_n,q_n')\to C_X(q,q')$

Lorsque $q_n \xrightarrow{n\to\infty} q$ et $q_n' \xrightarrow{n\to\infty} q'$, on note aussi que $m_X(q)$ est linéaire , i.e. :

$$m_X\left(\sum_{i=1}^{N}\lambda_i q_i\right)=E\left[X\left(\sum_{i=1}^{N}\lambda_i q_i\right)\right]=E\left[\sum_{i=1}^{N}\lambda_i X(q_i)\right]=\sum_{i=1}^{N}\lambda_i m_X(q_i),$$

Alors $m_X(q)$ est une fonction généralisé dans Q.

La covariance fonctionnelle $C_X(q_1,q_2)$ est bilinéaire fonctionnelle sur Q du fait que $X(q)$ est linéaire , de plus $C_X(q,q)=E\left[X(q)-m_X(q)\right]\geq 0$, $C_X(q_1,q_2)$ est fonctionnelle bilinéaire définie non négative.

I.11- Modélisation d'un processus naturel par un processus aléatoire spatial

I.11.1- Introduction

L'étude du comportement spatial des processus naturels en appliquant les concepts et méthodes des processus aléatoires spatiaux constitue une partie importante du programme de la recherche stochastique, plus précisément, la description des caractéristiques dominantes du *processus naturel*, ce qui est réalisé en utilisant :

(a) Les hypothèses de travail fondamentales, qui sont essentielles pour l'implémentation d'une méthode de recherche stochastique.
(b) Un ensemble d'hypothèses auxiliaires, ainsi que quelque relations de dualité, qui lient les processus naturels aux mathématiques des modèles de processus aléatoires.
(c) Une heuristique – un ensemble de règles méthodologiques- pour des modèles de corrélations spatiales en utilisant des procédures expérimentales (dans le cas de fonctions de covariance

et de semi-variogramme ordinaires), et pour des procédures automatiques (lorsqu'on a affaire à des modèles généralisés) .

Une conséquence de l'utilisation de la structure spatiale du processus naturel représentée en terme de processus aléatoire spatial à une signification objective. Les références aux données expérimentales soutiennent les exigences théoriques.

I.11.2- Relations de dualité entre un processus naturel et un processus aléatoire spatial :

La prochaine étape dans le développement de notre modèle est d'examiner comment est ce qu'il peut dériver des conclusions sur le comportement du processus naturel. On commence par l'étude des propriétés mathématiques du modèle de processus aléatoire. En d'autres termes, on établit un ensemble d'hypothèses auxiliaires, qu'on appelle *relations de dualité* entre le comportement du processus naturel et les caractéristiques mathématiques du processus aléatoire spatial associé. D'où le rôle que joue les paramètres associés aux fonctions de corrélation spatiale (Au sens ordinaire ou généralisé).

♦ **Relation de dualité N°1** :

Si la variation d'un semi-variogramme ou de la fonction de covariance se fait suivant différentes directions dans l'espace, le processus aléatoire spatial est non isotropique.
Le processus naturel est dit anisotropique.

♦ **Relation de dualité N°2**

Le rang ε du semi-variogramme ou de la fonction de covariance détermine la zone d'influence des valeurs locales du processus naturel.

♦ **Relation de dualité N°3**

Le comportement de la covariance ou du semi-variogramme pour de grandes distances détermine le degrés d'homogénéité du processus.

♦ **Relation de dualité N°4** :

Le comportement du semi-variogramme ou de la fonction de covariance à l'origine détermine le degrés de régularité dans la variation spatiale du processus.

Cette relation est une conséquence du résultat théorique :
Le comportement du semi-variogramme ou de la fonction de covariance à l'origine détermine la continuité et la dérivabilité en *m.q* du processus aléatoire spatial, ce qui implique la continuité du processus spatial naturel .

♦ **Relation de dualité N°5**

Si on peut montrer que : $$\gamma_X(h) \leq \frac{c|h|^{2n}}{\left|\log|h|\right|^{1+\beta}}$$

Où c est une constante positive et $\beta > 2$, alors le processus aléatoire spatial est continu en *m.q* et par conséquent le processus naturel est spatiallement continu.

I.12- Estimation du variogramme : [2]

Le variogramme est défini par : $2\gamma_X(S_1 - S_2) = \text{var}[X(S_1) - X(S_2)]$,
Certains le définissent comme : $E[X(S_1) - X(S_2)]^2$.

Lorsque le processus $X(.)$ est intrinsèquement stationnaire (de moyenne nulle) , les deux définitions coïncident , cependant, le processus $X(.)$ est représenté comme suit :

$$X(S) = u(S) + \delta(S),$$

Où $\delta(.)$ est un processus stochastique de moyenne nulle est de variogramme $2\gamma_X$, et une moyenne $u(S)$, non constante , alors :

$$E[X(S_1) - X(S_2)]^2 = 2\gamma_X(S_1 - S_2) + [u(S_1) - u(S_2)]^2 \ ;$$

(qui n'est pas en général une fonction de $(S_1 - S_2)$).

Elle doit aussi satisfaire la condition : $\dfrac{E[X(S_1) - X(S_2)]^2}{\|S_1 - S_2\|^2} \to 0$, lorsque $\|S_1 - S_2\| \to \infty$.

I.12.1- Estimation par la Méthode des Moments :

Sous l'hypothèse que la moyenne du processus est constante, un estimateur naturel basé sur la méthode des moments (*Matheron* 1962) est :

$$2\hat{\gamma}_X(h) = \frac{1}{|N(h)|}\sum_{N(h)}\left(X(S_i) - X(S_j)\right)^2 , h \in \Re^d \text{ Où : } N(h) = \left\{(S_i, S_j): \quad S_i - S_j = h, \quad i, j = 1, \ldots, n\right\}$$

et $|N(h)|$ est le nombre de couples distincts de $N(h)$.

On notera que $N(-h) \neq N(h)$, et que $2\hat{\gamma}_X(-h) = 2\hat{\gamma}_X(h)$ afin de préserver la propriété théorique du variogramme.

Lorsque les données sont irrégulièrement espacées dans $\Re^d$, on utilisera l'estimateur suivant : $2\gamma_X^+(h(l)) = Moy\left\{\left(X(S_i) - X(S_j)\right)^2 : \quad (i, j) \in N(h), \quad h \in T(h(l))\right\}$.

La région $T(h(l))$ est une certaine région spécifiée (tolérance) dans $\Re^d$ autour de $h(l)$, $l = 1, \ldots, k$ et $Moy\{.\}$: moyenne pondérée des éléments de $\{.\}$.

Les régions de tolérance doivent être assez petites que possibles pour conserver la résolution spatiale, et encore assez grandes pour que l'estimateur $2\gamma_X^+(.)$ soit stable . *Journel* et *Huijbregt*s ont recommandé un nombre de couples distinct $|U\{N(h), \quad h \in T(h(l))\}|$ dans $T\{h(l)\}$ soit proche de 30, on choisit les régions de tolérance de manière à ce que la plupart satisfassent cette condition.

On note que le statisticien peut choisir, soit une stationnarité intrinsèque ou une stationnarité de second ordre. L'estimateur du *covariogramme*, pour des données spatiales est donné par :

$$\hat{C}(h)=\frac{1}{|N(h)|}\sum_{N(h)}\left(X(S_i)-\overline{X}\right)\left(X(S_j)-\overline{X}\right) \text{ Où : } \overline{X}=\sum_{i=1}^{n}\frac{X(S_i)}{n}$$

On note que $2\hat{\gamma}_X(h)\neq 2\left(\hat{C}(0)-\hat{C}(h)\right)$ (l'estimateur ne préserve pas la propriété théorique).

I.12-2 Comparaison de l'estimation du covariogramme et du variogramme :

On sait que dans le cas de la stationnarité du second ordre, il existe une relation simple entre le covariogramme et le variogramme. De plus le variogramme peut être défini dans des cas où l'on peut pas le faire pour un covariogramme, dans ces cas , $\hat{C}(.)$ estime des paramètres non existants.

On supposera que les données sont equi-espacé dans $\Re$ et qu'ils sont représentées par $\{X(S),\quad S=1,\ldots,n\}$.

Les comparaisons sont faciles dans ce cas, par contre les conclusions restent générales.

***Biais* :**

L'estimateur classique du variogramme est sans biais pour $2\gamma_X(.)$lorsque $X(.)$ est strictement stationnaire, cependant lorsque $X(.)$ est stationnaire du seconds ordre, $\hat{C}(.)$ (déjà défini) a un biais égale à $o\left(\frac{1}{n}\right)$: $E\left(\hat{C}(h)\right)=C(h)+o\left(\frac{1}{n}\right)$.
Voir, par exemple, *Fuller* (1976, chapitre 6.2)

Le biais peut contribuer aux erreurs carrées moyennes lorsque *n* est petit.
Le biais obtenu peut être prolongé à un modèle à tendance polynomiale : $X(S)=\sum_{j=0}^{p}B_jS^j+\delta(S)\quad ,S\in\Re$.

En utilisant la méthode des moindres carrées ordinaires est utilisée pour obtenir un estimateur de $B=\left(B_0,B_1,\ldots,B_p\right)'$ i.e. $\hat{B}=\left(Z'Z\right)^{-1}Z'X$, où Z est une matrice *n.(p+1)* et $\hat{\delta}(S)=X(S)-\left(1,S,\ldots,S^p\right)\hat{B},\quad S=1,\ldots,n$ peut être utilisé à la place de $\{X(S),\quad S=1,\ldots,n\}$. Pour obtenir les estimateurs de $2\gamma_X(.)$ et $2\hat{C}(.)$.

I.13- Information et entropie d'un processus aléatoire spatial

On considère un processus aléatoire spatial, et soit $f_X(.)$ une fonction de densité de probabilité multivariée du vecteur aléatoire $X=(X_1,X_2,....,X_m)^T$ de *X(s)* définie aux points $s_1,s_2,....,s_m$.

Il existe différentes définitions du concept d'information. On donne la définition traditionnelle (*Shannon, 1948*), on considère que les variables aléatoires sont continues, mais les définitions correspondantes restent vraies dans le cas des variables discrètes.

L'information contenu dans le vecteur de variables aléatoires $X=(X_1,X_2,....,X_m)^T$ du processus aléatoire spatial *X(s),* est donnée par :

$$Inf(X_1,X_2,...,X_m)=-\log[f_X(X_1,X_2,....,X_m]$$

Où le logarithme peut être pris à n'importe quelle base *b>1*.
Quand le logarithme est pris à la base *b=2*, l'unité de l'entropie est le ″ bit ″ (*binary digit*).
La notion d'entropie est étroitement liée au concept de l'information.
La quantité ; $\varepsilon(X)=E\{Inf(X_1,X_2,....,X_m)\}=E[-\log(f_X(X_1,X_2,....,X_m)]$

$$=-\iint....\int\log[f_X(x_1,x_2,...,x_m)]dx_1dx_2.....dx_m.$$

est appelée entropie de la variable aléatoire *X*.

$\varepsilon(X)$: mesure l'incertitude dans la densité de probabilité $f_X(x_1,x_2,...,x_m)$ à priori, avant l'expérimentation.

Exemple :

L'entropie du vecteur aléatoire gaussien *X* est donné par : $\varepsilon(X)=\log\left[(2\pi e)^m|C_X|\right]^{1/2}$.

I.14- Prédiction : [3]

On suppose que les données sont les observations du processus $X(s)$ aux sites $\{s_1,s_2,...,s_n\}$ de la région D.

Une liste d'objectifs possibles est donnée par la suite .

Soit D_0 une région, tel que $D_0\subset D$.

A1. Pour $t\in D$, la valeur estimée (prédite) du processus $X(t)$ au point *t*.

A2. Prédire l'intégrale $\mathrm{I} = \int_{D_0} X(t)dt$ ou la moyenne $I \mid |D_0|$

A3. Estimer (prédire) l'intégrale de la fonction non linéaire du processus , i.e. $J = \int_{D_0} g(X(t))dt$, un exemple typique est $g(x) = \exp[x]$.

A4. Prédire le maximum du processus sur D_0, ie $\max_{t \in D_0} X(t)$.

A5. Estimer une région incluse dans D, où le processus dépasse un seuil donné ; ie : $D(\lambda) = \left\{ t : X(t) \geq \lambda \right\}$.

Dans chaque problème d'estimation sus - cité, on veut prédire des intégrales, aussi bien qu'on veut prédire des points.

Les prédicteurs généralement utilisés pour "*A1* "sont des prédicteurs linéaires non biaisés. On note $\hat{X}(t)$ le prédicteur de $X(t)$.

On dit que $\hat{X}(t)$ est linéaire s'il est de la forme :

$\hat{X}(t) = \sum_{i=1}^{n} a_i X(s_i)$, pour les coefficients (a_i) dépendants de t.

Dire que $\hat{X}(t)$ est sans biais est équivalents à dire que : $\sum_{i=1}^{n} a_i B_j(s_i) = 0$ tel que B_j sont donnés par l'équation : $E(X(s)) = \mu(s) = \sum_{j=1}^{p} \beta_j B_j(s)$ $1 \leq j \leq p$.

Le meilleur prédicteur linéaire sans biais *(BLUE)* de $X(t)$ (appelé aussi prédicteur *Kriging*) est celui qui minimise la variance de l'erreur de prédiction.

Il est donné par :

$$\hat{X}(t) = \left[v(t)^t \; B(t)^t \right] . \begin{bmatrix} V & F \\ F^t & 0 \end{bmatrix}^{-1} (X) \quad = \quad B(t)^t \hat{\beta} + v(t)^t V^{-1} \left(X - F.\hat{\beta} \right) ;$$

t :désigne la transposée d'une matrice où :

$$v^t(t) = [c(t, s_1), c(t, s_2), ..., c(t, s_n)]$$

$$B^t(t) = [B_1(t), B_2(t), ..., B_p(t)]$$

$$V_{ij} = c(s_i, s_j), \quad 1 \leq i, j \leq n$$

$$F_{ij} = B_j(s_i) \qquad 1 \leq i \leq n; 1 \leq j \leq p$$

$$\hat{\beta} = \left(F^t V^{-1} F \right)^{-1} F^t V^{-1} X$$

$$X = [X(s_1), X(s_2), ..., X(s_n)]$$

On note que $\hat{\beta}$ est l'estimateur des moindres carrés généralisé de β.

L'erreur quadratique moyenne de la prédiction est :

$$MSE(t) = MSE(t|s_1, s_2, ..., s_n) = E\left[\left(X(t) - \hat{X}(t)\right)^2 / X\right] = c(t,t) - \left[v(t)^t \; B(t)^t\right] \begin{bmatrix} V & F \\ F^t & 0 \end{bmatrix}^{-1} \begin{bmatrix} v(t) \\ B(t) \end{bmatrix}.$$

On note que $MSE(t)$ ne dépend pas des valeurs observées *X*, mais dépend toujours des sites d'observations $s_1, s_2, s_3, ..., s_n$ et de la covariance. De plus, on a pas besoin de connaître le vecteur des coefficients de la régression β.

Si $X(t)$ est un processus gaussien, alors, en utilisant la formule ci-dessus, on peut obtenir l'intervalle de prédiction de l'intégrale de $\hat{X}(t)$.

L'intégrale de prédiction à 95% est donné par : $\hat{X}(t) \pm 1.96\sqrt{MSE(t)}$.

On va s'intéresser à la prédiction du processus en un nombre de points supérieur à 1.

La covariance conditionnelle de l'erreur de prédiction entre 2 sites est donnée par :

$$Cov\left(X(t_1) - \hat{X}(t_1), X(t_2) - \hat{X}(t_2) | X\right)$$
$$= \begin{bmatrix} C(t_1,t_1) & C(t_1,t_2) \\ C(t_2,t_1) & C(t_2,t_2) \end{bmatrix} - \begin{bmatrix} v(t_1) & v(t_2) \\ b(t_1) & b(t_2) \end{bmatrix}' \begin{bmatrix} V & F \\ F & 0 \end{bmatrix}^{-1} \begin{bmatrix} v(t_1) & v(t_2) \\ B(t_1) & B(t_2) \end{bmatrix}$$

Pour(**A2**), il n'est pas difficile de montrer que le (*BLUP*) de l'intégrale est : $\hat{I}(D_0) = \int_{D_0} \hat{X}(t)dt$

$\hat{I}$: a comme erreur quadratique moyenne de prédiction :

$$E\left(\left[\int_{D_0} \left\{\hat{X}(t) - X(t)\right\} dt\right]^2 / X\right) = \int_{D_0 X D_0} E\left[\left\{\hat{X}(t) - X(t)\right\}\left\{\hat{X}(s) - X(s)\right\} \mid X\right] dtds .$$

I.15 - L'Espace de Transformation des Processus aléatoires Spatiaux :

I.15.1- Introduction :

L'analyse des problèmes est beaucoup plus facile dans le cas unidimensionnel que dans le cas multivarié, ces problèmes multidimensionnels sont rencontrés en hydrologie et en science de la terre.

Dans ces circonstances, les opérations mathématiques de l'espace des transformations introduit dans ce chapitre, simplifient l'étude du processus physique qui prend place dans un espace multidimensionnel en le convertissant en un espace unidimensionnel . Ceci revient à représenter un processus aléatoire de $\Re^n$ par une combinaison linéaire de processus spatiaux de $\Re$ indépendants.

On définit le concept de transformations spatiales en terme de fonction $f_n(s)$, $s \in \Re^n$.

I.15.2- L'Espace des Transformations :

I.15.2.1- Définition :

Soit $\{H_{n-1};\theta,s.\theta\}$, l'ensemble des hyperplans passants par un point donné de $\Re^n$. Le vecteur unitaire $\theta=(\theta_1,\theta_2,...,\theta_n)$ représente l'orientation de l'hyperplan H_{n-1} et $s.\theta$ est le produit d'*Inner* qui défini sa distance par rapport à l'origine, tel que :

$$\int_{\Re^n}\delta_t(s.\theta)ds=1 \;;\; \delta_t(s.\theta)=0 \text{ si } t\neq s.\theta \qquad (1)$$

L'espace de transformation du type 1 (*ST1*), T_n^1 rend la fonction $f_n(s)$, $s\in\Re^n$ dans l'ensemble des fonctions suivantes : $\hat{f}_{t,\theta}(t)=T_n^1[f_n(s)]=\int_{\Re^n}f_n(s)\delta_t(s.\theta)ds$, (2)

Pour les hyperplans $\{H_{n-1},\theta,s.\theta\}$.
(*ST1*) : est considéré complètement connu pour tout s,θ donnés.

I.15.2.2- Définition :

L'espace des transformations du second genre (ST-2), ψ_1^n : désigne l'ensemble des fonctions $f_{1,\theta}(t)$, définies sur un ensemble des hyperplans $\{H_{n-1},\theta,s\theta\}$ passant par un point donné de $\Re^n$, La fonction : $f_n(s)=\psi_1^n[f_{1,\theta}(t)]=\int_{H_n}U(s,\theta)f_{1,\theta}(s.\theta)d\theta$,

$U(s,\theta)$ est la fonction poids et $H_n\subseteq\Re^n$.

Un cas particulier intéressant de (*ST1*) est lorsque $\theta=\{1,0,...,0\}$, alors (2) devient :

$$\hat{f}_{1,\theta}(s_1)=T_n^1[f_n(s)]=\int_{\Re^n}f_n(s)\delta_{s_1}(s,\theta)ds=\int_U f_n(s_1,s_2,...,s_n)ds_2...ds_n \quad \text{où } U\subset\Re^n.$$

On peut étendre (*ST-1)* et *(ST-2)* au cas de la dimension $H_{(n-k)}$ de l'hyperplan I_{n-k},

$$\hat{f}_{n-k}(s_1,s_2,...,s_{n-k})=T_n^{n-k}[f_n(s)]=\int f_n(s_1,s_2,...,s_n)ds_{n-k+1}.....ds_n .$$

I.15.3- Représentations des processus aléatoire spatiaux :

I.15.3.1- Proposition :

Soit $X_n^*(s)$ un processus aléatoire spatial.

$X_n^*(s)$ admet la représentation linéaire suivante : $X_n^*(s_k)=\sum_{i=1}^{N} a_{ki} X_{1,i}(S_k.\theta_i)$, $a_{ki}\in\Re$

tel que les : $X_{1,i}$ sont des processus aléatoires non carrelés deux à deux à travers les droites θ_i, $i=1,2,...,N$ passants par un point donné, si et seulement si la covariance correspondante admet la représentation linéaire suivante : $C_n^*(h=S_k-S_\lambda)=\sum_{i=1}^{N} a_{ki}a_{\lambda i}c_{1,i}(h,\theta_i)$

Où : $c_{1,i}(h.\theta_i)$ est la covariance du processus aléatoire $X_{1,i}(h.\theta_i)$, i=1,....N

I.15.3.2- Proposition :

Un processus aléatoire spatial $X_n(s), s\in\Re^n$ admet la représentation $X_n(s)=\psi_1^n[X_{1,\theta}(\tau)]$, avec la fonction des poids $a(s,\theta)$, où $\tau=s.\theta$ et $X_{1,\theta}(\tau)$ sont des variables aléatoires non corrélées deux à deux, si et seulement si les covariances correspondantes sont reliées par la relation :

$C_n(h=s_k-s_\lambda)=\psi_1^n(C_{1,\theta}(t))$

Avec les fonctions poids $a(s_k,\theta)$: $a(s_\lambda,\theta)$. Où , t=$h.\theta$.

I.16- Simulation de processus aléatoire spatial :

I.16.1- Simulation sans contraintes :

Les méthodes de simulation discutées dans ce paragraphe sont appelées " sans contraintes ", afin de les distinguer des simulations où les réalisations dérivées sont contraint par les valeurs disponibles du processus naturel.

a) On suppose la méthode de simulation d'un processus aléatoire spatial :

Soit $X(s)$ un processus aléatoire spatial avec une densité de probabilité $f_x(X_1,X_2,...,X_m)$, alors : L'expression générale suivante est vérifiée :

$$f_X(X_1,X_2,...,X_m)=f_X(X_1).f_X(X_2/X_1).....f_X(X_m/X_1,X_2,...,X_m) \quad .....(1).$$

A travers l'équation (1) $X(s)$ peut être simulé comme suit :

Etape 1 *:*

Pour $f_X(X_1,X_2,...,X_m)$ donné, on calcul les densités de probabilités $f_X(X_1), f_x(X_2/X_1), ...$ et $f_X(X_m/X_1,X_2,...X_{m-1})$; Les distributions de probabilités correspondantes peuvent être calculées :

$F_X(X_1)=\int_0^{X_1} f_X(u)du$, $F_X(X_2/X_1)=\int_0^{X_2} f_X(u/X_1)du$.

Etape 2 :

Le système d'équations suivant de distributions marginales et conditionnelles est développé : $F_X(X_1)=V_1$

$$\begin{aligned} &F_X(X_2 / X_1)=V_2 \\ &\quad\vdots \qquad\qquad \ldots\ldots\ldots\ldots\ldots\ldots(2) \\ &F_X(X_m / X_1, X_2, ..., X_{m-1})=V_m \end{aligned}$$

Où les $V_i, i=1,....m$ sont des variables aléatoires indépendantes.

Etape 3 :

Finalement, les réalisations d'un processus aléatoire spatial $X(s)$ peuvent être générée en résolvant le système (2) en fonction de $X_1, X_2, .., X_m$.

Exemple :

Un processus aléatoire spatial avec une distribution de probabilité de *Cauchy* peut être simulé en utilisant la méthode présentée ci-dessus, ceci peut être réalisé en considérant que les distributions de probabilité sont des Cauchy univariés, et les distributions conditionnelles sont des " Student " avec m degrés de liberté.

I.16.2- Simulation avec contrainte :

D'un point de vu intuitif, on voudrait avoir les 2 caractéristiques suivantes :

- Les simulations d'un processus aléatoire spatial respecte les valeurs mesurées aux sites (localités) ; (en supposant bien sur qu'on a pas d'erreur de mesure).
- On exige une implémentation de tel sorte que l'approche conditionnelle, i.e. les réalisations générées du processus aléatoire spatial sont contraint par les données disponibles.

En d'autre termes, on suppose que les valeurs du processus $X(s)$ sont connues aux locations S_i, $i=1,2,\cdots,m$.

Les réalisations sont générées pour un processus aléatoire spatial :

$$X^*(s)=X(s)/X(s_i)=x_i, \quad i=1,2,\cdots,m .$$

La simulation avec contrainte est aussi appelée : simulation conditionnelle. Dans le cas particulier, d'un processus spatial $X^*(s)$ peut être obtenu en utilisant l'expression suivante :

$$X^*(s)=\hat{X}(s)+\varepsilon(s) \ldots..(1)$$

Où : $\hat{X}(s)$ est estimation du processus actuel . (estimateur : *Kriging*)

et $\varepsilon(s)=X'(s)-\hat{X}'(s)$: est l'erreur entre la réalisation non conditionnelle $X(s)$ et son estimateur .

Dans la majorité des applications pratiques, la densité de probabilité multivariée est rarement connue ou peut être estimée sur la base des données disponibles. Par conséquent, on se limite à satisfaire un seul objectif, à savoir, sur la base des statistiques du second ordre.

En particulier, la plupart des simulations en sciences de la terre produisent des réalisations d'un processus aléatoire spatial Gaussien, ces réalisations préservent la moyenne, la covariance, et le semi-variogramme, et la distribution de la distribution de probabilité du processus naturel.

I.16.3- La technique de matrice triangulaire : (Lower upper triangular)

Cette technique produit un processus aléatoire spatial Gaussien.
Soit $C_x(s_i, s_j)$, la covariance d'un processus aléatoire spatial de moyenne nulle, on cherche à simuler *m* points d'une grille.

L'application de l'approche *LU* à résoudre ce type de problème se fait en trois étapes.

Etape 1 :

Pour un modèle de covariance donné $C_x(s_i, s_j)$, $(i,j=1,2,..,m)$, on développe la matrice de covariance correspondante de taille $(m \times m)$ comme suit :

$$C_x = \begin{bmatrix} C_x(s_1, s_2) & \cdots & C_x(s_1, s_m) \\ C_x(s_2, s_1) & \cdots & C_x(s_2, s_m) \\ \vdots & & \\ C_x(s_m, s_1) & \cdots & C_x(s_m, s_m) \end{bmatrix}$$

Etape 2 :

La matrice C_x est une matrice symétrique définie non négative qui peut être décomposée en un produit d'une matrice triangulaire supérieure *L* et une matrice triangulaire inférieure *U* (en utilisant la méthode de *Cholesky*) . i e $C_x = LU$, où : $U = L^t$.

Etape 3 :

On suppose que *V* est un vecteur de variables aléatoires Gaussiennes centrées réduites, on défini le vecteur $X=L.V$ ……(1) Où : $X^t = \left[X(s_1), ..., X(s_m)\right]$.

Les simulations générées par (1) ont une moyenne nulle et une covariance :

$$E\left[(LV)(LV)^t\right] = L.E\left[VV^t\right]U = L.I.C_x .$$

De plus (1) est une combinaison linéaire de variables aléatoires identiquement distribuées, en utilisant le théorème central limite ceci produit une simulation d'un processus aléatoire spatial Gaussien.

Remarque :

Soit $Z(s)$ un processus aléatoire spatial ayant une moyenne non nulle égale à $E[Z(s)]$. Si on a : $Z(s)=X(s)+E(Z(s))$, où $X(s)$ est un processus aléatoire spatial centré, on peut générer les réalisations de $Z(s)$ tel que :

$$Z=LV+M \text{ où : } Z^t=[Z(s_1),...,Z(s_m)] \text{ ; } M^t=[E[Z(s_1)],...,E[Z(s_m)]].$$

M^t : le vecteur moyenne correspondant.

I.16.4- Simulation de Vecteur de Processus Aléatoire Spatial :

b) Méthode « Turning Band » :

Mathématiquement , c'est l'extension d'un processus aléatoire spatial "*scalaire*" au cas vectoriel (technique de simulation) pose certaines difficultés. On considère, par exemple, le cas d'un processus spatial homogène.

Le processus spatial scalaire $X_n(s)$ est remplacé par le vecteur de processus aléatoires spatial : $X_n(s)=[X_1(s),X_2(s),\cdots,X_k(s)]^t$ ……..(1)

La covariance $C_x(s_i,s_j)$ est remplacée par la matrice de covariance

$$C_x(h)=\begin{bmatrix} C_{x_1}(h) & \cdots & C_{x_1,x_k}(h) \\ C_{x_2,x_1}(h) & \cdots & C_{x_2,x_k}(h) \\ \vdots & & \\ C_{x_k,x_1}(h) & & C_{x_k}(h) \end{bmatrix} \text{……..(2)}$$

On aura alors : $X_n(s)=\frac{1}{\sqrt{n}}\sum_{i=1}^{N} X_{1,\theta_i}(s.\theta_i)$ ……..(3) où : X_{1,θ_i} sont des vecteurs de processus aléatoires univariés associés à l'équation (1) à travers l'espace de transformation.

I.17- Echantillonnage Spatial

I.17-1. Considérations générales :

En général, l'échantillonnage spatial vise un arrangement d'observations d'un processus naturel où les sites explorés satisfassent certains objectifs .

Dans la plupart des cas, l'objectif est d'avoir une erreur d'estimation du processus la plus petite possible parmi un nombre d'arrangements pratiques imaginable ; cependant, dans plusieurs autres situations les facteurs additionnels, comme le coût de prélèvement et les contraintes physiques doivent être prix en compte .

En conclusion, les paramètres sont liés à la gestion d'emplacement planifiés. La prise de décision peut exercer une commande importante sur l'efficacité de la conception de l'échantillonnage.

La technique d'échantillonnage doit prendre en compte les caractéristiques de la variabilité spatiale du processus naturel échantillonné. L'analyse de la variabilité spatiale est la clé d'un échantillonnage efficace. Les principaux échantillonnages peuvent être compris en connaissant les hypothèses sur la structure spatiale utilisée. Tous les cas d'échantillonnage n'existent pas lorsque le site n'a pas de variabilité spatiale.

La désignation d'un schéma de collecte de données exige une pure formulation d'au moins deux composantes, ce qui est donné par le domaine et l'objectif de l'échantillonnage. La présence et la forme des frontières et la priorité des sous domaines est d'une importance particulière.

L'augmentation de la taille de l'échantillon ne fournit aucune information supplémentaire, mais c'est lorsqu'il existe un certain degré de variabilité spatiale qui croît que la taille de l'échantillon contribue à l'exactitude de l'estimation.

Les techniques d'échantillonnages classiques sont limitées, pour la caractérisation d'un site pour les raisons suivantes :

- Les techniques classiques ne traitent pas les observations spatialement corrélées.
- Elles ne s'intéressent pas aux problèmes géométriques de recherche et d'identification du modèle.
- Elles s'occupent, essentiellement, d'estimation de fréquences dans les grandes populations, pas avec des décisions.

L'utilisation de l'approche " *processus aléatoire spatial* ", peut être avantageuse pour spécifier les caractéristiques de variabilité spatiale importantes pour désigner le réseau d'échantillonnage, d'analyser et d'évaluer les résultats obtenus.

Tandis que la théorie des méthodes d'échantillonnage qui est discuté dans ce chapitre sont généralement valides pour un processus naturel spatial qui peut être modélisé par un processus aléatoire spatial .

I.17-2 Optimisation et solutions sub-optimales :

En général, une bonne définition au sens mathématique d'un échantillonnage spatial est un problème difficile. Dans certains cas, une solution optimale du problème peut ne pas exister.

D'un point de vu statistique, le problème d'échantillonnage spatial peut être posé comme suit :

Soit $X(s)$ un processus aléatoire spatial , on suppose que $\mathrm{H} = \left\{ s: \quad s = \{s_i \quad i = 1, \cdots N\} \right\}$

(N entier fixe). H est l'espace de tous les ensembles s de points s_i, dans un domaine $U(s)$ de $X(s)$.

On choisit $s^* = \{s_i^* \quad i = 1, \cdots, N\} \in$ H, satisfaisant :

$$\sigma^2(s^*) = \min_{\lambda_i} E\left[\int_{U(s')} f(s)X(s)dS - \sum_{i=1}^{N} \lambda_i X(s_i^*)\right]^2 = \min_{s \in H} \min_{\lambda_i} E\left[\int_{U(s')} f(s)X(s)ds - \sum_{i=1}^{N} \lambda_i X(s_i)\right]^2 \quad \text{.....(1)}$$

où : f est une fonction connue continue sur $U(s) \subset \Re^n$.

Un indice simple d'efficacité spatiale est la variance $\sigma_A^2(s)$ tel que : $\sigma_A^2(s) = E\left[X_U - \hat{X}_U\right]^2$

Où : $X_U = \frac{1}{U}\int_{U(s')} X(s)ds$ est la moyenne du processus, son estimation est donnée par : $\hat{X}_U = \frac{1}{N}\sum_{i=1}^{n} X(s_i)$ où : $s = \{s_i, \quad i = 1,2,..., N\} \in$ H.

Dans ce cas, l'objectif de l'échantillonnage est de choisir un ensemble $S' \in$ H tel que la variance de l'estimateur σ_A^2, soit minimale : $\sigma_A^2(s^*) = \min_{S \in H} \sigma_A^2(s)$ .....(2)

On note que $\sigma_A^2(s)$ peut être écrite comme $E(.) = E_x\left\{E_{sp}[.]\right\}$ où : E_x est la moyenne relativement au processus aléatoire spatial et E_{sp} : relativement à l'échantillonnage des points aléatoire sur le domaine *U(s).*

Pour $f(s) = \frac{1}{U}$ et $\lambda_i = \frac{1}{N}$, les expressions (1) et (2) coïncident.

L'efficacité de l'échantillonnage spatial peut être évalué par différents indices d'efficacité, comme l'estimation de l'erreur moyenne sur un domaine *U.*

$\sigma_A^2 = \frac{1}{K}\sum_{k=1}^{K} \sigma_x^2(s_k)$: où K est le nombres de points estimés dans le domaine U.

s_k : les points (en général d'une grille rectangulaire).

$\sigma_x^2(s_k)$ sont les points (estimation d'erreurs).

Par exemple : $\sigma_x^2(s_k) = \min_{\lambda_{k_i}} E[X(s_k) - \sum_{i=1}^{m} \lambda_{k_i} X(s_i)]^2$ où : $(m \le N \le K)$

Remarque : Les différents indices peuvent nous mener vers des échantillonnages différents car les critères d'échantillonnage qu'on utilise dépendent de l'objectif de l'exploration du site; par exemple en science de la terre, le critère (2) est souvent utilisé.

I.17-3 Notions importantes d'échantillonnage :

L'application des méthodes d'estimation spatiale dans l'échantillonnage spatial, utilisent trois notions, l'échantillonnage de point (*Sampling pattern*), l'échantillonnage de la densité et l'estimation de voisinage.

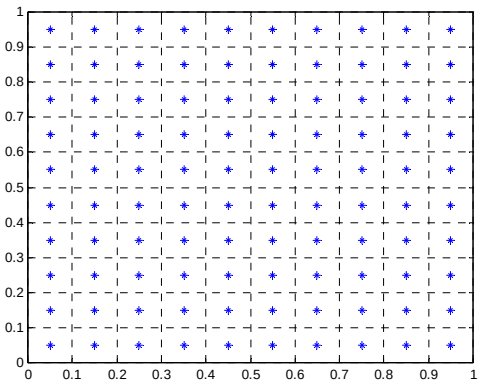
1) **Echantillonnage de points (Sampling Pattern)** :

Dépend de la configuration des observations dans l'espace. Les plus importants « Sampling pattern » bi-dimensionnels sont :

- *L'échantillonnage de points réguliers* : Tel que le modèle hexagonal, carré et triangulaire, par convention, les noms sont assignés sur la base du partitionnement du domaine d'échantillonnage *U*. Dans ces modèles le point d'échantillonnage est le centroïde du voronoï polygone. (voir Fig 1).

- *L'échantillonnage de points stratifiés* : Comme l'hexagonal et le carré, les points sont obtenus en divisant le domaine d'échantillonnage de manière exclusive (ex : polygones) et sélectionner un point aléatoirement pour chaque partition.

- *Echantillonnage de points aléatoires* : D'autres groupes d'échantillonnage de points sont utilisés comme le rectangulaire (voir Oléa, 1984)



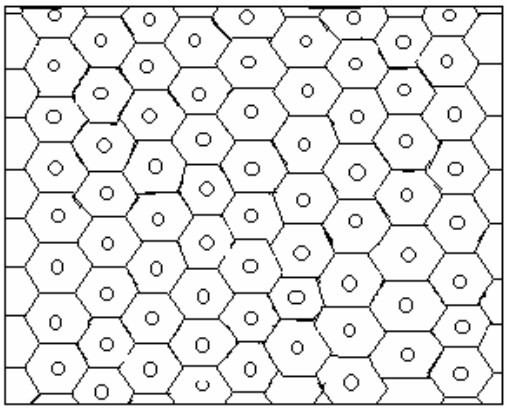


Fig 1 : a) Plan d'échantillonnage Carré **b) Plan d'échantillonnage Hexagonal**

2) Echantillonnage de la densité θ : est le nombre d'observations par unité du domaine d'échantillonnage.

3) L'estimation du voisinage m : C'est le facteur échantillonnage qui apparaît dans le nombre minimum de points nécessaires pour l'estimation du système à résoudre.

I.17.4 - Compléments : [5]

Récemment, un échantillonnage adapté a été développé en incluant une information (Thompson et Seber, 1996), en permettant une modification du schéma si un certain critère est vérifié, tel que la présence d'un phénomène rare.

Une autre méthodologie " *Survey Sampling* " a été plus récemment utilisée, en permettant un arrangement général (Müller, 1998). Premièrement sans contraintes, après en incluant une sorte de contraintes. Dans ces études, la fonction objectif est formulée telle quelle soit approximée par les différentes configurations des données (les plus précises) et qui vont avec les considérations théoriques et pratiques.

I.17.4.1 – Evaluation des plans :

L'évaluation des plans en collectant des données de tel manière qu'un objectif quantitatif soit satisfait. On considère le critère noté $\phi(A/S)$, où la dépendance des données collectées est indiquée par S sur un domaine A.

- A : Peut être une forme arbitraire.
- $|S|$:Le nombre d'éléments de S (déterminé à l'avance).

Un critère est estimé par la fonction *fitness* $\hat{\phi}(A/S)$ qui est un estimateur de $E(\phi(A/S))$.
Pour l'évaluer, l'espérance est remplacée par la valeur moyenne d'un ensemble N_e de points d'évaluations.

$$E(\phi(A/S)) = \sum_{i=1}^{Ne_e} \hat{\Phi}\ (A/S) \quad ..... \ (1)$$

Pour un *écart type Kriging* S_{oK} comme mesure de qualité de la carte.

$\phi(A/S)$ est égale à la moyenne des *écart type Kriging* dans le domaine A donné par l'échantillon S *et* $\phi(x_i/S)$ est égale à l'écart type évalués en N_e localités tests.

Le critère appliqué par (Stein et Ettema) est le critère de Warrick et Myers (1987), qui distribue n(n-1)/2 couples de points d'échantillon également sur les classes distantes pour les calculs du variogramme $\sum_{i=1}^{n_c} (\xi_i - \xi^*_i)^2$ …. (2)

Où n_c: le nombre de distance classes pour le variogramme.

- ξ^*_i : Le nombre exigé de couples de points pour la ième distance classe.

- *W.M* : Critère qui dépend seulement des distances entre les points de l'échantillon, donc la fonction *fitness* peut être donc facilement calculée à partir de l'équation (1).

La distribution idéale est définie comme une distribution égale au nombre de couples sur une distance classe donnée.

I.17.4.2 – Echantillonnage adaptatif :

L'échantillonnage adaptatif (Thompson et Seber, 1996) est relativement une nouvelle stratégie où l'information recueillie durant la collection des données est utilisée. Un plan d'échantillonnage adaptatif n'est pas entièrement désigné en avance, mais se fait durant l'évaluation.

Le plan lui même consiste en un ensemble de blocs contigus A_i , $i=1,....N$ de même taille.

Chaque bloc est divisé en un nombre égale au nombre de parcelles non recouvertes μij, $j=1,2,...,M$.

Le voisinage de la parcelle $\delta\mu ij$ est contigu par l'entourage, ce qui est représenté par la parcelle elle même plus ses quatre adjacents dépendant toujours de la proximité physique.

I.17.5 - Algorithmes de recherche :

Une partie importante lors de la conception de l'échantillonnage est le choix d'un ensemble de points qui satisfassent un critère d'optimalité, c'est ce qu'on appelle le problème de recherche " *Search problem* ".

Dans la plupart des cas, il est difficile (ou bien impossible de déterminer des solutions optimales), donc on se restreint à des solutions sous- optimales. Les algorithmes de recherche jouent un rôle important dans les méthodes d'échantillonnage locales .

Dans $\Re^2$, la minimisation des indices $\sigma^2(s)=\sigma_A^2(s)$ ou $\sigma_M^2(s)$, peut se faire à l'aide d'algorithmes calculatoires qui donnent des résultats sous-optimaux rapidement. Tous les algorithmes de recherche qu'on va considérer supposent que le nombre N de points échantillonnés est fixe. Cependant, les N points doivent être sélectionnés d'un domaine U fini, défini de manière précise dans l'espace.

Les algorithmes qui connaissent le plus de succès sont « annealing algorithms », un important ingrédient de ces algorithmes est la probabilité π_k (*Métropolis et al, 1953, Geman 1984), (Sacks et Shiller, 1988),*

Avec $\pi_k = \min\left\{1, \exp\left[-\gamma_k\left(\sigma^2(s_{k+1}) - \sigma'^2(s_k)\right)\right]\right\}$ où $\gamma_k = \frac{\log k}{\gamma}$ $(k \geq 2)$

et γ un paramètre à spécifier et la probabilité

$$\pi_k = \begin{cases} 1 & si \quad \sigma^2(s_{k+1}) - \sigma^2(s_k) \leq 0 \\ 0.5 & \sin on \end{cases},$$

" *Search algorithm* " donné par l'algorithme (I) est basé sur une méthode proposée par *Sacks* et *Schiller* (1988).On peut l'utiliser dans $\Re^2$, lorsque le nombre de points de l'échantillon est fixe (on le prend égale à N).

Cet algorithme dépend d'un certain nombre de paramètres : n_0, α, δ et M, qui doivent être spécifiés (l'expérience joue un rôle important). En pratique, cet algorithme est performant lorsque les valeurs prises par les paramètres sont : $\alpha=0.01$, $\delta=[0.3, 0.5]$ et $M \in [100,250]$.

L'algorithme (I) reste valide lorsque la minimisation de l'indice d'efficacité est remplacé par un autre critère d'optimalité, par exemple : Echantillonnage par le critère du maximum d'entropie.

Moins sophistiqué, tout de même, utilisable en pratique, les algorithmes de recherche : " *greedy algorithm* " et " *Sequentiel exchange algorithm* " sont utilisés lorsque l'on veut sélectionner un échantillon de taille *N* parmi *M* échantillons candidats dans un domaine *U* suivant un certain critère d'optimalité. Ces localités sont connues à priori. Le nombre de combinaisons possibles de *N* points de l'échantillon parmi *M* est d'ordre M^N.

I.17.5.1- Algorithme (I) " Annealing Search "

Étape (1) :

On commence par un ensemble S_0 de *N* points (échantillon) prient aléatoirement dans un domaine *U* d'un processus naturel, et soit π_0=0.7 .

Étape (2) :

à l'étape *k+1*, on sélectionne un point S' aléatoirement de l'ensemble *U*-S_k , et on cherche un autre point $S''\in S_k$ satisfaisant : $\sigma^2\left(S_k \cup S' - S''\right) = \underset{S\in S_k}{Min}\left(S_k \cup S' - S\right)$ si $\sigma^2\left(S_k \cup S' - S''\right) < \sigma^2\left(S_k\right)$, on prend $S_{k+1} = S_k \cup S' - S''$, sinon, on prend $S_{k+1} = S_k \cup S' - S$ avec une probabilité π_k et $S_{k+1} = S_k$ avec une probabilité $1-\pi_k$.

Étape (3) :

Dans le cas où : $S_{k+1}=S_k$, on prend un point S^* aléatoirement du domaine $U-S_k-S'$ et on répète l'étape (2) en utilisant S^* au lieu de S' .

Étape (4) :

Refaire l'étape (3), si nécessaire (τ_0 fois), $\tau_0 \le K - N$ (*K* : le nombre de points estimés et *N* le nombre de points échantillonnés. Si après τ_0 épreuves, il n'y a pas de mouvement enregistré, on prend $S_{k+1} = S_k$ et $\pi_{k+1} = \min\left\{1, \frac{\pi_k}{\left(1-\delta'\right)}\right\}$. Changement à l'étape *k+2*, retour à l'étape (2).

Étape (5) :

Soit $$\pi_{k+1}\begin{cases}\left(1-\delta\right)\pi_k & si \quad \sigma^2\left(S_{k+1}\right) \le \left(1-\alpha\right)\min\limits_{i\le k}\sigma^2\left(S_i\right) \\ \pi_k & Sinon\end{cases}$$

Étape (6) :

La procédure d'échantillonnage est achevé après *M* épreuves si π_k change.

I.17.5-2 : Algorithme (II) "Greedy Algorithm" :

Cet algorithme est très simple, et implique *N*M* combinaisons possibles d'échantillons de taille *N* parmi *M* sites candidats, l'échantillon n'est pas optimal mais il peut être utile pour une première utilisation en pratique.

Greedy Search Algorithm *:*

Etape (1) :

on sélectionne un seul point (en respectant le critère d'optimalité) parmi les *M* sites candidats.

Etape (2) :

On choisit un second point parmi *(M-1)* ,sans prendre en compte le premier point choisis.

Etape (3) :

On continu ainsi jusqu'à avoir sélectionner les *N* points.

I.17.5-3: Algorithme (III) :"Sequential Exchange Search Algorithm "

Etape (1) :

On commence avec un ensemble initial S_0 de *N* points parmi les *M* points possibles (" *greedy algorithm* " peut nous donner le S_0).

Etape (2) :

On laisse *(N-1)* échantillons de l'ensemble S_0 fixes et on trouve le point optimal pour le $N^{ième}$ échantillon, ce qui donne un nouveau échantillon S_1 .

Etape (3) :

Faire la même chose pour les *(N-1)* points restants. Soit S_{N-1} le dernier ensemble de sites échantillonnés , après *N* points utilisés.

Etape (4) :

Répéter les étapes de (1) à (3) en commençant par S_{N-1} au lieu de S_0 après *k* répétitions, l'algorithme converge vers l'échantillon optimal (là où aucune autre modification n'est possible).

Dans notre cas, on va utilisé un autre algorithme de recherche qui est l'A.G.S (*algorithme génétique spatial*). Boukhetala et Mehassouel ([19],[35]) ont prouvé l'efficacité de cet algorithme ainsi que sa supériorité par rapport au recuit simulé dans le cas de l'estimation d'une intégrale double à l'aide d'un nombre fini de points soigneusement choisis de manière à minimiser l'erreur quadratique moyenne entre l'intégrale double et son estimateur; On proposera un AGS adapté au problème de l'échantillonnage spatial optimal sur un domaine continu et borné.

II- Les Algorithmes Génétiques :

II.1- Préambule :

Les Algorithmes Génétiques (*AG*) représentent une famille assez riche et très intéressante d'algorithmes d'optimisation stochastique fondées sur les mécanismes de la sélection naturelle et de la génétique. Les champs d'application sont fort diversifiés.
On les retrouve aussi bien en théorie des graphes qu'en compression d'images numérisées ou encore en programmation automatique . Les raisons de ce nombre d'applications sont claires. Leur principe est d'opérer une recherche stochastique sur un important espace à travers un ensemble “ une population ” de pseudo -solutions. Ces algorithmes sont simples et très performants dans leur recherche d'amélioration. De plus, ils ne sont pas limités par des hypothèses contraignantes sur le domaine d'exploration. Ainsi, le mathématicien abordant le sujet n'a guère à se préoccuper de la continuité et de la différentiabilité de la fonction à optimiser.

Les algorithmes génétiques différent fondamentalement des autres méthodes d’optimisation selon quatre axes principaux:

1. Les *AG* utilisent un codage des paramètres, et non les paramètres eux mêmes.

2. Les *AG* travaillent sur une population de points, au lieu d'un point unique.

3. Les *AG* n'utilisent que les valeurs de la fonction étudiée, pas sa dérivée, ou une autre connaissance auxiliaire.

4. Les *AG* utilisent des règles de transition probabilistes, et non déterministes.

Afin de bien appréhender la dynamique des algorithmes génétiques, introduisons les principaux éléments du jargon utilisé dans la littérature.

- *Individu ou Chromosome* : une solution potentielle.
- *Population* : un ensemble de chromosomes ou de points de l'espace de recherche.
- *Environnement*: l'espace de recherche;
- *Fitness ou Fonction d'évaluation* : la fonction - positive - que l'on cherche à maximiser.

Ce petit lexique, est fait avec une prudente analogie avec la biologie, a naturellement une simple fonction métaphorique. Leur emploi n’obéit qu'à un souci pragmatique. Nous les utilisons dans l'introduction de l'algorithme génétique standard, considéré désormais par maintes auteurs sous le nom d' *Algorithme Génétique Canonique* (*AGC*), défini par *Holland* [10. 1975] et redynamisé par *Goldberg* [11. 1989].

II.2 Algorithme Génétique Canonique

Avant d'examiner les mécanismes et la puissance d'un Algorithme Génétique Canonique (*AGC*), présentons clairement l'objectif visé. Le problème de l'*AGC* peut être résumé de la manière suivante : On dispose d'une fonction *f*, non constante, définie sur le cube logique $\{0,1\}^N$, N est un entier naturel à valeurs dans IR_+^*, et on se fixe pour objectif de localiser l'ensemble f^* des maxima globaux de *f*, ou à défaut, de trouver le plus rapidement et le plus efficacement possible des points sous-optimaux.

Formellement, il s'agit d'une optimisation sans contrainte du type :

$$Max\left\{ f(x) \mid x \in \{0,1\}^N \right\}$$

Très souvent, comme dans notre problème, on s'intéresse plutôt aux minima d'une fonction *f*; et, dans ce contexte, les résultats suivants sont utilisés.

1)- ***Optimisation des fonctions*** :

Soit *f* une fonction réelle d'une ou de plusieurs variables. Les problèmes d'optimisation suivants sont équivalents:

- $\underset{x}{Min}\, f(x)$
- $Max_x(-f(x))$

De plus, la fonction optimisée par un *AG* (f) doit avoir des valeurs positives dans l'espace de recherche. Dans le cas contraire, il est nécessaire d'ajouter aux valeurs de *f* une constante positive adéquate *C* conformément à l'équivalence des problèmes d'optimisation suivants:

2)

a) $Max_x f(x)$

b) $Max_x(f(x)+C),\ \ C \in IR$

En définitive, l'objectif de l'*AG* est de régler les différents paramètres d'un problème donné pour obtenir la plus grande valeur du fitness f possible.

II.2-1 Philosophie des AGC :

L'organigramme [*figure 1*] nous montre clairement le fonctionnement de l'*AGC* et indique ses différents opérateurs de base. Ces opérateurs sont inspirés directement des mécanismes de la sélection naturelle et des phénomènes génétiques. Ainsi, le principe de l'*AGC* - ce qui d'ailleurs justifié son nom - est de faire évoluer une population de façon à adapter les individus à l'environnement en place, comme le ferait un ensemble d'êtres vivants. Ce principe s'apparente convenablement à celui de la théorie de *Darwin* sur l'évolution, selon laquelle " la vie est une compétition et seuls les mieux adaptés survivent et se reproduisent ".

Techniquement, une nouvelle génération s'obtient généralement au bout d'un cycle [figure1] composée de trois opérateurs standards : *reproduction*, *croisement*, et *mutation*.

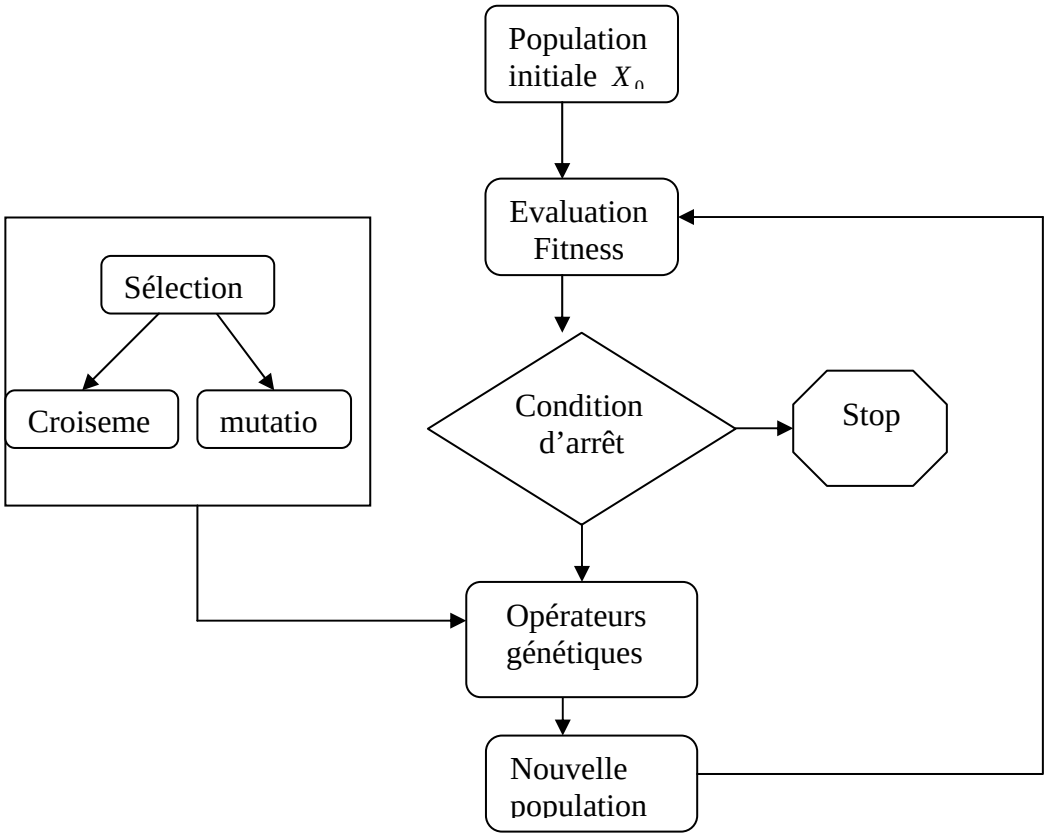


Figure 2 – Organigramme de l'AG

a) Reproduction :

L'opérateur de reproduction est directement inspiré de la sélection naturelle, il consiste à dupliquer chaque individu de la population proportionnellement à sa valeur d'adaptation. Il s'agit, en quelque sorte, de réaliser autant de tirages avec remise - à la manière de la technique du *Bootstrap* - qu'il y a d'éléments dans la population. Chaque élément étant tiré avec une probabilité liée à sa valeur d'évaluation que le statisticien peut bien interpréter comme des probabilités de survie. Ainsi, les individus ayant un grand *fitness* ,ont plus de chance d'être sélectionnés pour la génération suivante. On parle alors de *sélection proportionnelle*.

L'opérateur de reproduction peut être mise en œuvre sous forme algorithmique de différentes façons. La plus facile est de créer une *roue de loterie biaisée* [figure 3] pour laquelle chaque individu de la population courante occupe une section de la roue proportionnelle à son adaptation.

Pour réaliser la reproduction, il suffit de faire tourner la roue biaisée ainsi définie autant de fois qu'on a besoin de descendant. Le groupe ainsi obtenu constitue l'ébauche de la nouvelle génération qui est ensuite soumise à d'autres opérateurs génétiques.

Etant donné un fitness positif (*f*) et une taille de la population *n*, une telle reproduction peut être décrite schématiquement par le pseudo code suivant:

Algorithme : *roue biaisée*

Début

Evaluer le fitness f_i de chaque individu X_i

Calculer *F* le fitness total de la population

$F = \sum_{i=1}^{n} f_i$

Pour i :=1 :n

$p_i = f_i / F$ probabilité de sélection de X_i

$q_i = 0$

Pour j :=1 :i

$q_i = q_i + pj$

fpour

pour i :=1 :n

$\tilde{p}$ réel tiré dans [0,1]

Si $\tilde{p} < q_1$

Sélectionner X_1

Sinon Sélectionner X_k, $2 \le k \le n$ tel que $q_{k-1} < \tilde{p} < q_k$

Fsi

Fpour

Fin

L'inconvénient majeur de ce type de reproduction vient du fait qu'elle peut favoriser la domination d'un individu ne représentant pas véritablement le meilleur. De plus, elle peut aussi engendrer une perte de diversité par la domination d'un super- individu. Ce qui ne donne pas une "bonne" population. Pour pallier cet inconvénient, on préfère très souvent des méthodes de reproduction plus justes qui n'autorisent en aucun cas la naissance d'un *super-individu*.

Une solution de compromis peut éviter les inconvénients de l'utilisation de la sélection proportionnelle est de procéder par un changement d'échelle de la fonction d'adéquation : On s'arrange pour que la fonction d'adéquation minimale et la fonction d'adéquation maximale aient des valeurs données en adoptant un changement d'échelle statique ou dynamique, linéaire ou exponentielle ; évidemment, le changement d'échelle est indispensable quand les valeurs de la fonction d'adéquation sont négatives puisque la sélection ne peut admettre des probabilités négatives.

Une autre solution est l'utilisation de la *sélection par rangement* : le principe de cette méthode consiste à ranger les individus de la population de la génération courante par ordre croissant de leurs évaluations *(fitness*), (par ordre décroissant de leurs évaluation pour un problème de maximisation). Puis attribuer à tout individu i, une probabilité p_i de participation dans la génération suivante, selon son rang, telle que :

$$p_i = \frac{\Phi}{n} - (r_i - 1) \times \frac{(2\Phi - 2)}{n(n-1)}$$

Tel que : le nombre moyen d'enfants de l'individu i est $n \times p_i$ Où :

n : taille de la population

r_i : rang de l'individu i

Φ : est la pression de la sélection, *Φ∈[1,2]* qui représente le nombre moyen d'individus enfants du meilleur individu.

Cette méthode de sélection évite les inconvénients de la sélection proportionnelle et représente un changement d'échelle dynamique.

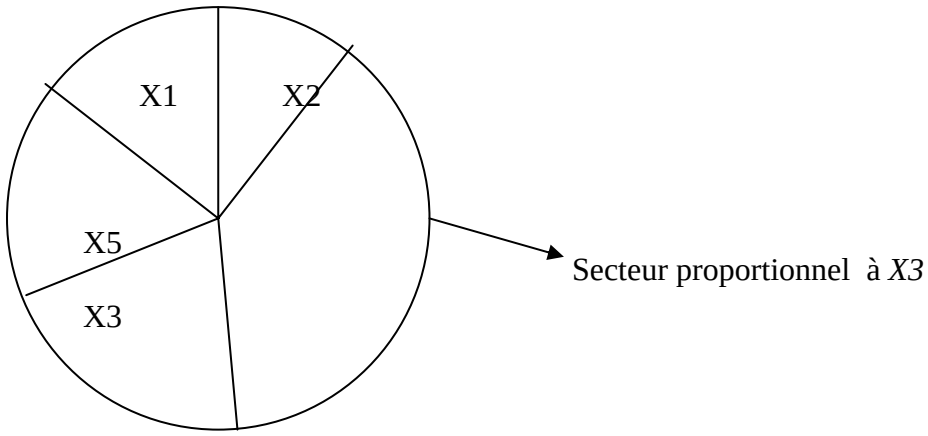


Figure 3- Roue de loterie biaisée

Après la reproduction, un *croisement* permet de générer les nouveaux individus.

b) Croisement : Relativement à deux individus " parents ", considérés à cette occasion comme des vecteurs, la dynamique de l'opérateur de croisement, inspiré directement du processus de multiplication des chromosomes humains, consiste à générer deux nouveaux individus " enfant " de la façon suivante :

On choisit au hasard 2 *parents* dans la population intermédiaire P' (ayant subit la reproduction) et un site de croisement (un nombre de 1 à l , l est la longueur du chromosome),soit *k* l'indice du site, On obtiendra alors deux enfants en prenant d'une part les *k* premiers gènes du premier parent et les (*l-k*) derniers gènes du deuxième parent. On note que l'opérateur de croisement est appliquée avec une probabilité p_c. Cette méthode est appelée méthode de croisement à un site.

Il existe d'autres types de croisements comme le croisement à deux sites où on échange les segments ainsi délimités entre les deux parents. L'avantage de cet opérateur est de parcourir plus rapidement l'ensemble des solutions en permettant de modifier plus facilement les allèles (bits) des deux bouts d'un *chromosome*.

-Croisement multi-sites : Plus généralement, un croisement multi-site qui consiste à choisir au hasard K sites de croisement, et interchanger les parties correspondantes des deux parents choisis aléatoirement.

Schématiquement, un opérateur de croisement à 4 sites de croisement, est représenté par la figure suivante.

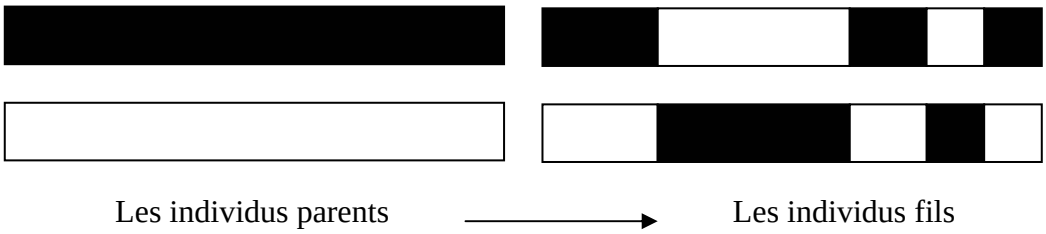


Figure 4- L'opérateur de croisement à 4 sites

- Croisement uniforme :

Soit r_i un nombre appartenant à l'intervalle [0,1].
L'application d'une opération de *croisement uniforme* sur les deux individus parents *A* et *B* :
$A=(a_1,a_2,....,a_i,....,a_n)'$, $B=(b_1,b_2,....,b_i,.....,b_n)'$. donne les deux individus fils A',B' :
$A'=(a_1',a_2',....,a_i',....,a_n')'$, $B'=(b_1',b_2',....,b_i',.....,b_n')'$.
Tel que :

$$a_i'=\begin{cases} a_i, & r_i \le 0.5 \\ b_i & r_i > 0.5 \end{cases} \qquad b_i'=\begin{cases} a_i & r_i > 0.5 \\ b_i & r_i \le 0.5. \end{cases}$$

c) Mutation :

L'opération de mutation consiste à modifier de manière aléatoire, avec une probabilité généralement faible, la valeur d'une composante de l'individu.
Dans le cas d'un codage binaire de l'*AG*, un booléen $\omega \in \{0,1\}$ choisi aléatoirement est remplacé par son inverse $1-\omega$.Techniquement, la combinaison de la reproduction et du croisement sont suffisantes pour assurer l'évolution de la population, mais il peut arriver que certaines informations essentielles- composantes dans le cas des vecteurs- disparaissent de la population. Le rôle premier de la mutation est donc de pallier à cet inconvénient. Prise isolément, la mutation constitue une exploration aléatoire dans l'espace des chaînes. Mais une utilisation parcimonieuse avec la reproduction et le croisement constitue une police d'assurance protégeant de la perte prématurée de composantes importantes.

Remarque :

Un rapprochement plus profond des *AG* à la biologie a permis de mettre en évidence d'autres opérateurs génétiques et d'autres mécanismes de reproduction. On peut citer entre autre, l'inversion, la combinaison de l'inversion et du croisement.

Ces opérateurs ont été utilisés dans plusieurs contextes. Cependant, les trois opérateurs déjà cités dans cet partie ont fait preuve de leur simplicité de mise en œuvre algorithmique et de leur efficacité pour résoudre un nombre important de problèmes d'optimisation par les *AG*.

II.2.2- Les paramètres :

Les opérateurs de l'*AG* sont guidés par un certain nombre de paramètres généralement fixés à l'avance et dont dépend très fortement la "bonne" convergence de l'algorithme. Présentons brièvement les principaux paramètres de l'*AG*.

- La taille de la population et la longueur du codage de chaque individu. Il est conseillé de prendre comme taille de la population la valeur correspondant à la longueur du codage des individus. En effet, si la taille de la population est très grande, l'évaluation de tous les individus de la population peut s'avérer trop longue. Par contre, si elle est très petite, l'algorithme peut converger trop rapidement.

- La probabilité de croisement p_c. Elle dépend de la forme de la fonction d'évaluation. Son choix est généralement expérimental et sa valeur est très souvent prise entre 0.5 et 0.9. Plus elle est élevée, plus la population subit des changements importants. Ainsi, la convergence est très rapide si le taux de croisement est proche de 1.

- la probabilité de mutation p_m . Le taux de mutation est généralement faible, le risque d'un taux élevé étant de modifier les meilleurs individus et ainsi, de s'éloigner de l'*optimalité*.

Il ressort clairement des divers détails donnés sur les paramètres de l'*AG* que la fixation des différentes valeurs numériques des précédents paramètres, reste une tâche assez complexe.
En fait, la délicate gestion des différents paramètres de l'*AG* n'est qu'une traduction de la difficile maîtrise de ceux-ci au niveau moléculaire.

II.2.3- Génération d'une nouvelle population :

Nous avons décrit un ensemble suffisant d'opérateurs standards dont se sert le programmeur pour générer les nouveaux individus à partir des descendants de la population courante. Ces opérateurs peuvent constituer une étape directe ou indirecte dans le protocole de création d'une nouvelle génération d'individus. Ainsi, la littérature des *AG* offre une collection fort variée de stratégies dont il convient ici d'en citer quelques unes:

1) *La nouvelle population est composée uniquement des enfants* : On laisse alors disparaître tous les individus de la population courante. L'inconvénient majeur de cette approche est le risque de perdre le meilleur individu lorsqu'on ne le conserve pas automatiquement dans la nouvelle population.

2) Les enfants remplacent d'une façon régulière les individus les moins forts de la génération courante.

3) La nouvelle génération est constituée des *n* meilleurs individus de la population intermédiaire formée des enfants et des parents.

1) Il peut être intéressant de garder intact le meilleur individu de la population lors du passage d'une génération à une autre. La stratégie *élitiste* concrétise cette idée, en remplaçant le meilleur individu d'une population dans la population suivante et ce, sans aucune altération (mutation), [11. 89], cet individu reste toutefois candidat pour la phase de reproduction habituelle.

II.3- Le théorème fondamental des A.G

Les Algorithmes génétiques font intervenir constamment des opérateurs aléatoires. Cependant, l'influence de l'arbitraire du choix est bien moindre qu'on peut avoir tendance à le croire. Afin de bien comprendre le processus de fond – appelé théorème fondamental, qui guide l'algorithme génétique, introduisons la notion de schéma.

II.3.1 Les schémata :

Une bonne partie des études menées sur les *AG* repose sur leur approche par l'information globale. Ainsi, plusieurs auteurs se sont intéressés aux similarités entre les individus d'une population. La formalisation de cette recherche de similarités entre les individus utilise des méta -symboles ou des sous ensembles particuliers de l'espace de recherche appelé *schéma* .

II.3.1.1- Définition :

Soit $X_1, X_2, ..., X_n$ une séquence de longueur *n* sur l'alphabet binaire {0,1,*} où * : est une indéterminée booléenne susceptible de prendre l'une des deux valeurs logiques 0 ou 1. Une *chaîne binaire* $b = b_1 b_2 .....b_n$ est dite *instance* du motif *X* si et seulement si :

$$\forall i \in \{1,2,...,n\} \quad b_i = X_i \quad ou \quad b_i \in \{0,1\} \quad si \quad X_i = *.$$

II.3.1.2- Définition :

L'ensemble des chaînes binaires de longueur *n* qui sont des instances d'un motif spécifié *S* de longueur *n* est appelée schéma .

Par exemple, le schéma *S = 10*10*0* représente l'ensemble des individus suivants, classés dans l'ordre lexicographique.

S 1 = 1001000
S 2 = 1001010
S 3 = 1011000
S 4 = 1011010

Relativement à la notion de schéma, introduisons deux caractéristiques couramment utilisées dans la théorie des *AG*.

- L'ordre d'un schéma *S,* noté *o(S*), correspond au nombre de positions instanciées dans le schéma *S.*

- La longueur de définition , notée *l(S),* désigne la distance entre la première et la dernière position instanciée du schéma *S.*

Ainsi, dans l'exemple ci-dessus S = 10* 10 * 0 a pour ordre *o(S)* = 5, et pour longueur de définition *l(S)* =7.

II.3.2- Théorème des schémata :

Nous considérons un *AG canonique* opérant sur une fonction f définie sur le cube logique $B = \{0,1\}^N$ (N entier naturel) à valeurs dans IR_+^*. Désignons par P une population d'individus - chaînes binaires de longueur N. , on associe à B la suite des valeurs : $\left\{ f(\omega) \mid \omega \in B \right\}$

II.3.2.1- Définition :

L'utilité d'un schéma S vis à vis de la population P se mesure par l'expression:

$$U(P,S) = \frac{1}{|S \cap P|} \sum_{\omega \in S \cap P} f(\omega)$$

Où $|S \cap P|$: désigne le cardinal de l'ensemble $S \cap P$.

Désignons par $(P_t), t \in IN$ les différentes populations successives obtenues par l'*AGC*.
Le théorème fondamental des algorithmes génétiques indique que la fréquence relative de présence de représentants dans la population P_t appartenant à un schéma S, a d'autant plus tendance à augmenter (dans la population P_{t+1}) que l'utilité de S est meilleure. Formellement, ce théorème se traduit par l'inégalité suivante [11. 1975].

II.3.2.2- Théorème :(des schémata) :

Si $(P_t), t \in IN$ désigne les populations successives obtenues par l'*AGC*, alors :

$$E([S \cap P_{t+1}]) \geq |S \cap P_t| \frac{U(P_t,S)}{U(P_t,B)} (1 - c(P_t,S) - m(P_t,S)).$$

Où : E désigne l'espérance mathématique et *c(.)* (resp. *m(.)*) approximent les probabilités qu'une instance de S soit détruite - n'apparaisse pas dans la génération suivante - par l'opérateur de croisement (resp. *mutation*).

Ce résultat peut se formuler en utilisant les différents paramètres de l'*AGC* et les caractéristiques du schéma S. Pour le croisement, nous nous limiterons au croisement à un seul site. En effet, la probabilité qu'un schéma S de taille N soit détruit - n'est plus de représentant dans la génération - est quantifiée par : $\frac{l(S)}{(N-1)}$

Où : *l(S)* est la longueur de définition du schéma S. Ainsi, un schéma ayant une grande longueur de définition a plus de chance d'être détruit par l'opérateur de croisement.

Le croisement étant appliqué avec une probabilité p_c, la survie du schéma S après croisement simple (un seul site de croisement) est alors minorée par la probabilité :

$$1 - \left(p_c \frac{l(S)}{(N-1)} \right)$$

D'autre part, un chromosome du schéma peut subir des changements sous l'effet de la *mutation*. Le schéma restera inchangé si aucune des *o(S)* positions instanciées ne subit l'influence de l'opérateur de mutation.

Ainsi, la destruction du schéma *S* par une mutation de probabilité(*petite*) p_m se traduit par la probabilité $(1-p_m)^{o(S)} \approx 1-o(S)p_m$

En définitive nous avons : $c(P_t,S) = p_c \dfrac{l(S)}{(N-1)}$ *et* $m(P_t,S) = o(S)p_m$

Dans ces conditions, l'inégalité du théorème des schémata se présente sous la forme :

$$E([S \cap P_{t+1}]) \geq |S \cap P_t| \frac{U(P_t,S)}{U(P_t,B)} \left(1 - p_c \frac{l(S)}{(N-1)} - o(S)p_m\right)$$

A travers cette reformulation, on peut remarquer que les *building blocks* - schémata de courte longueur de définition et de faible ordre - auront tendance à se multiplier considérablement dans la population. Ainsi, la bonne convergence de l'*AG* reposerait sur la capacité de recombiner ces building blocks par croisement pour obtenir des chromosomes de plus en plus performants. L'utilité relative des schémata $U(P_t,S)$ devient alors un aspect fondamental de la mesure de performance d'un algorithme génétique.

II.3.3- Parallélisme Implicite :

Lorsqu'on regarde l'*AG* comme un méga- opérateur sur l'espace des schémata, une des conséquences pratiques du théorème des schémata est le phénomène appelé *parallélisme implicite* des opérations. En fait, chaque évaluation d'un individu fournit une information sur les schémata qu'il représente. Une population de *m* chromosomes engendre entre 2^N *et* $m \times 2^N$ schémata. Le nombre de schémata traités est en moyenne de l'ordre de m^3. Ainsi, plus l'espace de recherche est large , plus il est intéressant est l'emploi d'un *AG* par rapport à d'autres technique [12. 1990].

II.3.4- Codage des Solutions :

Dans un Algorithme génétique, il est nécessaire, compte tenu de la complexité des objets manipulés, de travailler à un niveau synthétique, notamment pour les deux opérations fondamentales que sont le croisement et la mutation. Les solutions sont alors représentées sous forme de codes qui tiennent compte, au mieux, de la nature du problème posé. Pour notre problème (*optimisation d'une fonction dans un espace continu et borné*) nous codons, comme nous le verrons plus loin, une configuration directement par un vecteur de réels. Cependant, les premières générations des *AG*, en particulier l'*AGC*, utilisaient essentiellement
l'alphabet binaire conformément aux travaux de *Holland* [10. 1975] qui montre, avec la théorie des schémata, que le code optimal pour un *AG* est un code ayant un alphabet le plus petit possible. Dans ce contexte, le décodage d'une chaîne binaire $S = (s_0, s_1, ..., s_n)$ en un nombre réel $x \in [a, b]$ est directe et complètement déterminée par les deux opérations suivantes :

(1) La séquence binaire *S* est convertie en base 10 :

$$(s_0, s_1, ..., s_n)_2 = \left(\sum_{i=0}^{n} s_i 10^i\right)_{10} = x'$$

(2) le nombre réel *x* correspondant est:

$$x = a + x' \frac{b-a}{2^{n+1}-1}$$

Ces opérations sont effectuées un grand nombre de fois (à chaque évaluation d'un individu) au cours de l'évolution de l'algorithme. Par conséquent, le choix d'un code doit tenir compte de la complexité algorithmique du processus de *codage/décodage* dont la lourdeur peut ralentir les calculs et influencer considérablement la convergence de l'algorithme tout entier.

II.4- Convergence de l'AGC :

La convergence vers une solution optimale d'un algorithme est un but toujours recherché et rarement atteint. La question se pose de façon fondamentale pour les Algorithmes génétiques.

Dans cette partie, les propriétés des *chaînes de Markov* sont exploitées pour étudier la convergence de l'Algorithme génétique . Nous adoptons l'approche récente de Gunter [13. 1994] dans laquelle la séquence des populations produites par l'*AG* est considérée comme une chaîne de Markov homogène. La propriété fondamentale d'un système *Markovien* est l'absence de mémoire : l'évolution future du système ne dépend que de l'instant présent et non de son évolution passée. Toutefois il existe d'autres approches plus ou moins convaincantes. En particulier, le parallélisme implicite et le théorème des schémata ont servi d'outils de l'analyse de la convergence des *AG*. Malheureusement, les arguments combinatoires très immédiats utilisés dans cette analyse ont paru peu rigoureux [14. 1991].

II.4.1- chaînes de Markov finies :

Dans cette partie, nous rappelons les principaux résultats relatifs aux processus Markoviens dont nous chercherons à montrer l'intérêt pour la convergence des Algorithmes génétiques dans le paragraphe suivant.

Nous considérons une chaîne de Markov homogène d'espace fini d'états *E* de cardinal *r* complètement définie par la donnée d'une part, par ses probabilités initiales

$q_i = P(X_0 = E_i), 1 \le i \le r$ et d'autre part, par sa matrice de transition $P = (p_{ij})\ \ 1 \le i, j \le r.$

$$P = \begin{pmatrix} p_{11} & p_{12} & \cdots\cdots & p_{1r} \\ p_{21} & p_{22} & \cdots\cdots & p_{2r} \\ \vdots & \vdots & \vdots & \vdots \\ p_{r1} & p_{r2} & \cdots & p_{rr} \end{pmatrix}$$

II.4.1.1- Définition :

Un état *i* est dit apériodique si : $PGCD\left\{k / p_{ii}^{k} > 0\right\} = 1.$

Où : p_{ii}^{k} - coefficient de *P* multiplié par lui même *k* fois- est la probabilité conditionnelle de revenir à l'état *i* en *k* étapes.

II.4.1.2- Définition :

Une chaîne de Markov irréductible - tout état peut être atteint à partir de tout autre état - et ayant tous ses états apériodiques est dite ergodique.

Nous donnons ci-après une liste de propriétés matricielles fondamentales partagées par les matrices de transition. La matrice $M = (a_{ij})$ référencée dans les énoncés est supposée carrée et d'ordre n.

II.4.1.3- Définition (Propriétés des matrices de transition) :

(1) *M* est dite *positive (M >0)* si $a_{ij} > 0 \quad \forall i, j \in \{1,2,....n\}$.

(2) *M* est dite *non négative (M ≥ 0)* si $a_{ij} \geq 0 \quad \forall i, j \in \{1,2,...,n\}$

Une matrice *M non négative* est dite :

(3) *primitive*, s'il existe un entier naturel *k* tel que M^k soit positive. En particulier, toute matrice positive est une matrice primitive;

(4) *Réductible* si elle peut être mis sous la forme (*U* et *V* sont des matrices carrées) :

$$\begin{pmatrix} U & 0 \\ R & V \end{pmatrix}$$

En appliquant les mêmes permutations sur les lignes et les colonnes de la matrice *M*. Une matrice réductible n'est donc pas primitive.

(5) *Stochastique* si : $\forall i, \sum_{k=1}^{r} a_{ik} = 1$

En particulier toutes les matrices de transitions sont stochastiques et tout produit de matrices stochastiques donne une matrice stochastique. Une matrice *M* stochastique est dite :

(6) Stable si toutes ses lignes sont identiques.

(7) Positive par colonne si elle a au moins un élément positif par colonne.

II.4.1.4- Lemme :

Soient $A = (a_{ij}), B = (b_{ij}), C = (c_{ij}) \quad \forall i, j \in \{1,2,...,n\}$ trois matrices stochastiques avec *B* positive et *C* positive par colonne. Alors, *ABC* est une matrice positive.

II.4.1.5- Théorème : [15. 1980]

Pour toute matrice *M* stochastique primitive, le produit matriciel M^k converge, lorsque $k \to \infty$, vers une matrice stochastique positive et stable $M^\infty = I' q^\infty$

Où $I = (1,\ldots,1)$ et $q^\infty = q^0 \lim_{k\to\infty} M^k = q_0 M^\infty$ est sans élément nul, unique, et indépendante de la distribution initiale q_0.

II.4.1.6- Théorème: [15. 1980] :

Soit une matrice $M = \begin{pmatrix} U & 0 \\ R & V \end{pmatrix}$ stochastique et réductible, où la matrice carrée U est d'ordre l stochastique primitive et R; $V \neq \infty$. Alors :

$$M^{\infty} = \lim_{k\to\infty} M^{k} = \lim_{k\to\infty} \begin{pmatrix} U^{k} & 0 \\ \sum_{i=0}^{k-1} V^{i} R U^{k-i} & V^{k} \end{pmatrix} = \begin{pmatrix} U^{\infty} & 0 \\ R^{\infty} & 0 \end{pmatrix} \qquad (\text{I})$$

(I) est une matrice stochastique stable avec $M^{\infty} = I' q^{\infty}$, unique et indépendante de la distribution initiale. q_{∞} satisfait : $q_i^{\infty} > 0, \forall 1 \le i \le l \quad et \quad q_i^{\infty} = 0, \forall l \le i \le n.$

II.4.2 Analyse de l'*AG* par les Chaînes de Markov

Le comportement de l'Algorithme Génétique optimisant une fonction *f* définie sur le cube logique $B = [0,1]^N$ tel que N est un entier naturel à valeurs dans IR_+^* est modélisé, dans ce paragraphe, par une chaîne de Markov homogène finie pour laquelle l'espace d'états est donné par E = $B^{N.l}$ où *N* désigne la taille d'un chromosome et *l* l'effectif de la population.

Chaque état est alors constitué rigoureusement d'une population et d'une seule. Dans ces conditions, nous pouvons désigner par π_k^i la projection qui retourne pour l'état *i* le *k* ième chromosome de taille l . Cette fonctionnalité nous permet d'identifier tous les individus d'une population - état.

Nous considérons une décomposition de la matrice *P* de transition de l'*AG* en un produit naturel de matrices stochastiques *P = C * M* S* où : *C ,M* et *S* décrivent les transitions intermédiaires issues respectivement du croisement, de la mutation et de la reproduction-sélection. Cette décomposition est toujours possible tel que nous l'indique la propriété (II.4.1.3). Ceci nous conduit au résultat suivant :

II.4.2.1- Théorème [13. 1994] :

Soit un *AG* avec une probabilité de croisement $p_c \in [0,1]$, une probabilité de mutation $p_m \in [0,1]$, une sélection proportionnelle et une matrice de transition *P*. Alors *P* est primitive.

Preuve :

L'opérateur de croisement peut être vu comme une fonction aléatoire définie sur *E* à valeurs dans *E* . Chaque état est donc lié d'une façon probabiliste à un autre état. Ainsi, *C* est une matrice stochastique. Ce raisonnement tient pour *M* et *S*. La probabilité qu'un état *i* devienne l'état j après une mutation est donnée par : $m_{ij} = p_m^{H_{ij}} (1 - p_m)^{N.l - H_{ij}} > 0 \quad \forall i, j \in E$

où H_{ij} désigne la distance de *Hamming* entre les représentations binaires des états *i* et *j*. *M* est donc positive.

La probabilité p_s^{ii} qu'une sélection ne perturbe pas un état généré par la mutation est bornée par : $p_s^{ii} \ge \dfrac{f(\pi_1^i) \times f(\pi_2^i) \times ..... \times f(\pi_{N.l}^i)}{(f(\pi_1^i) + ... + f(\pi_{N.l}^i))^{N.l}}$, pour tout état $i \in E$, ainsi *S* est positive par colonne.

Ce qui indique que dans ces conditions $P = C*M*S$ est positive. Or toute matrice positive est primitive, donc P est primitive.

II.4.2.2- Corollaire :

L'*AG* avec les paramètres du théorème précédent est une *chaîne de Markov ergodique.*

Preuve : Conséquence immédiate des théorèmes (II-4.1.5) et (II-4.1.6).

La distribution initiale n'a donc pas d'effet sur le comportement limite de la chaîne de Markov. Ainsi l'initialisation de l'algorithme génétique canonique peut être arbitraire et l'opération de reproduction- sélection peut - théoriquement - être omise, puisque la distribution limite restera inchangée. Par contre, la propriété d'ergocité a des conséquences considérables sur le comportement asymptotique de l'*AG*. Précisons dans quel sens nous entendons la convergence de l'*AG*.

L'objectif d'un *AG* est de localiser l'ensemble $f^* = \max\left\{f(x) / x \in \{0,1\}^N\right\}$ des maxima globaux de f. En gardant les notations précédentes, désignons par $X_t = \left\{ f(\pi_k^t(i)) \quad tq \quad k = 1,..N \right\}$.

Une suite de variables aléatoires représentant les meilleurs fitness dans la population *i* au temps *t*. Alors, l'*AG* converge vers l'optimum global si et seulement si : $\lim_{t\to\infty} P\left\{X_t = f^*\right\} = 1$.

Ceci nous conduit à un résultat qui fait encore couler beaucoup d'encre.

II.4.2.3- Théorème :

Tout *AG* avec une probabilité de croisement $p_c \in [0,1]$, une probabilité de mutation $p_m \in [0,1]$, et une sélection proportionnelle ne converge pas vers l'optimum global.

Preuve :

Soit un état $i \in E$ tel que $\max\left\{f(x) / x \in \{0,1\}^N < f^*\right\}$ et la probabilité p_i^t que l'*AG* soit à l'état *i* au temps *t* . On a clairement, $p(X_t \neq f^*) \geq p_i^t \Leftrightarrow \quad p(X_t = f^*) \leq 1 - p_i^t$

D'après le théorème (II-4.1.5), la probabilité que l'*AG* soit à l'état *i* converge vers $p_i^\infty > 0$. Par conséquent :

$$\lim_{t\to\infty} P(X_t = f^*) \leq 1 - p_i^\infty < 1$$

Dans la pratique, nous ne nous conformerons pas strictement au modèle aléatoire de la sélection proportionnelle; puisque la meilleure solution obtenue à une étape de l'*AG* est gardée pour l'étape suivante. Ainsi, la solution globale est forcement visitée après un nombre fini, certes très élevé, de transitions. Cette technique est appelée couramment élitisme [11. 1989].

II.5- Adaptation des AG au cas Spatial

II.5.1- Préambule

Dans cette partie, nous exploitons la souplesse des *AG*, décrit dans le paragraphe précédent, pour développer de nouveaux algorithmes adaptés au problèmes spatiaux, particulièrement à l'échantillonnage spatial optimal sur la base d'optimisation d'un certain critère, dont la représentation des individus (chromosomes) de la population et les opérateurs génétiques prennent en considération les caractéristiques du domaine d'échantillonnage.

Ce problème est mathématiquement représentée - indépendamment des approches (solution classique, moindres carrés ou maximum de vraisemblance) - par l'équation fondamentale de l'optimisation numérique sans contrainte en dimension finie [21, 1981] [15, 1984]

$$f_0 = \min\left\{ f(x) / x \in IR^k \right\}$$

où *f* désigne la fonction coût adoptée et *k* la dimension de la configuration recherchée.

Nous commencerons par décrire au paragraphe (II-5.2) le fitness et l'espace de recherche de nos algorithmes. C'est au paragraphe (II-5.3)que nous présenterons notre codage qui a un caractère très synthétique et géométrique. Les opérateurs de croisement et de mutation adaptés à nos individus seront ensuite présentés au paragraphe (II-5.4). Le paragraphe (II-5.5) sera réservé à la description des algorithmes que nous proposons.

II.5.2 Fitness et espace de recherche

Nous supposons la donnée d'une fonction objectif positive, à minimiser, *f* quelconque .
Si nous notons par *X* un individu de l'*AG*, le fitness, quantité positive est donnée, par :

$$\tilde{f}_i = -f(X_i) + \max_{1 \le i \le n} \left\{ f(X_i) \right\} + \varepsilon$$

Où ε est une constante positive très petite. L'ajout de cette constante permet d'attribuer une probabilité de sélection non nulle à l'individu le plus faible.
L'espace de recherche, d'un *AG* classique, dans un problème d'optimisation d'une fonction de *n* variables est une boule hypercubique $[a_1, b_1] \times [a_2, b_2] \times .... \times [a_n, b_n] \in IR^n$. Ce qui suppose - pour une bonne efficacité numérique - une identification a priori de la zone solution.

Par ailleurs, notre problème possède un ensemble de solutions pouvant décrire un cône. En effet, si *Y* est une configuration optimale, la configuration αY est aussi une configuration optimale. En d'autres termes, la fonction objective est invariante par toute transformation homothétique

Dans ces conditions, une boule de l'espace de représentation - de dimension deux ou trois - munie d'une subdivision discrète est suffisante comme espace de recherche.

Ainsi, nous considérons pour l'espace de recherche, dans le cas d'une représentation planaire, un carré de côté égal à l'unité.

On a déjà noté que de nombreuses versions des algorithmes génétiques originaux ne se basent plus sur un codage binaire des paramètres à optimiser, mais travaillent directement sur les paramètres eux mêmes.

Ces versions d'algorithmes génétiques, que nous appelons codés réels offrent généralement l'avantage d'être mieux adaptés aux problèmes d'optimisation numériques continus car elles accélèrent la recherche et rendent plus aisé le développement de méthodes hybrides avec des méthodes plus classiques.

Cependant, les algorithmes codés réels nécessitent le développement d'opérateurs spéciaux (mutation, croisement, etc…) faits sur mesure en fonction de l'application.

En général, les algorithmes codés réels sont rapides, plus précis(surtout pour des larges domaines qui requièrent une représentation trop longue avec les algorithmes génétiques discrets) ; ils sont plus consistant d'une génération à une autre, ce qui est du au fait que les opérateurs agissent directement sur le phénotype et non pas sur le génotype. En outre, ils permettent d'éviter certains problèmes dus au caractère artificiel du codage binaire.

II.5.3- Représentation Chromosomique: codage des individus

Classiquement pour l'emploi d'un *AG* en optimisation, le chromosome est directement la concaténation des codes binaires des composantes du vecteur représentant un point de l'espace de recherche. Dans notre problème, une solution "l'échantillon optimal" dans un espace de dimension $k < n$ est une matrice $X_{n\times k}$ de réels, de la forme suivante:

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1k} \\ \vdots & & & \vdots \\ \vdots & & & \vdots \\ x_{n1} & x_{n2} & & x_{nk} \end{bmatrix}$$

Ainsi, une population de solutions dans l'espace de recherche est un ensemble de matrices réelles $M(n\times k)$. Nous adoptons un code directement lié a la matrice, pour l'individu i, par le vecteur réel de taille $n\times k$ $\quad X_i=(x^i_{11},....,x^i_{1k},x^i_{21},...,x^i_{2k},.....,x^i_{n1},....,x^i_{nk})$

Où les x^i_{ij} sont des réels et le sous-ensemble $x^i_{j1}, x^i_{j2},....,x^i_{jk}$ correspond aux coordonnées de la *jème* entité traitée.

L'intérêt majeur, par rapport au codage binaire, de ce type de codage réside dans sa capacité réductrice de la complexité. En effet, le décodage - porté directement sur les indices - est quasi- immédiat et ne nécessite aucunement une opération de codage/décodage semblable à celle que nous avons présentée au paragraphe (II.3.4).

Nous restreignons, par la suite, nos analyses sur un espace de dimension 2, ce qui n'altère en rien la perspicacité des résultats.

II.5.4- Croisement et Mutation

II.5.4.1- Croisement

Etant donné deux configurations codées $X_1 = (x^1_1,.....,x^1_{nk})$ *et* $X_2 = (x^2_1,....,x^2_{nk})$

un croisement classique, que nous appellerons par la suite *croisement global* produit des nouvelles configurations suivant le principe énoncé au paragraphe(II.2.1). Le croisement uniforme correspond, ici, à interchanger un sous ensemble aléatoire de sommets [figure 4], alors qu'une opération de mutation concernant une même configuration correspond à un déplacement "adéquat " d'un sommet choisi aléatoirement. Par exemple, en considérant les individus parents de la figure 4:

X = (x1, x2, x3, x4, x5, x6, x7) .

$= (x1_1, x1_2, x2_1, x2_2, x3_1, x3_2, x4_1, x4_2, x5_1, x5_2, x6_1, x6_2, x7_1, x7_2))$

Y = (xa, xb, xc, xd, xe, xf, xg)

$= (xa_1, xa_2, xb_1, xb_2, xc_1, xc_2, xd_1, xd_2, xe_1, xe_2, xf_1, xf_2, xg_1, xg_2)$

Un croisement uniforme qui porte sur les sites - indices - 3, 4, 7, 8, 11 et 12 produit deux individus enfants:

$X(x1, xa_1, xb_1, xd_2, x5_1, xf_1, x7_1)$

$Y' = (x_a, x_2, x_c, x_4, x_e, x_6, x_g)$

$= (xa_1, xa_2, x2_1, x2_2, xc_1, xc_2, x4_1, x4_2, xe_1, xe_2, x6_1, x6_2, xg_1, xg_2)$

Une difficulté émerge lorsqu'on applique, conformément à la théorie des *AG*, les opérateurs que nous venons de décrire avec une probabilité faible pour la mutation.

Le capital génétique - l'ensemble des composantes de tous les vecteurs - des différentes générations reste pratiquement inchangé et diffère très peu de celui de la population initiale.

Ceci s'explique par le fait que le croisement a finalement pour simple rôle de changer les indices des composantes. Une première idée a été d'enchérir la probabilité de mutation pour maintenir un diversité suffisant et ainsi préserver l'amélioration.

Les travaux de *De Jong* démontrent clairement que cette méthode ne règle pas le problème. Une augmentation incontrôlée du taux de mutation peut dégrader la performance de l'algorithme génétique. Nous avons donc introduit un nouveau opérateur dit de croisement local qui agit localement sur des couples de composantes des deux individus à croiser. Ce croisement génératrice de nouvelles composantes se justifie par le résultat suivant:

II.5.4.2- Théorème (Propriété sur le codage binaire) :

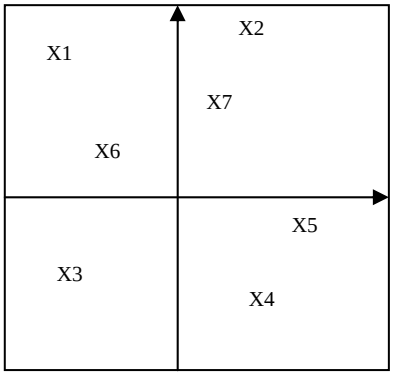
Soient C_1 *et* C_2 deux individus binaires d'un *AG*. Si C_1' *et* C_2' sont deux individus obtenus par un croisement multi- site de C_1 *et* C_2, l'égalité suivante est vérifiée :

$C_1 = 1011100010101$ et $C_2 = 0010100100111$

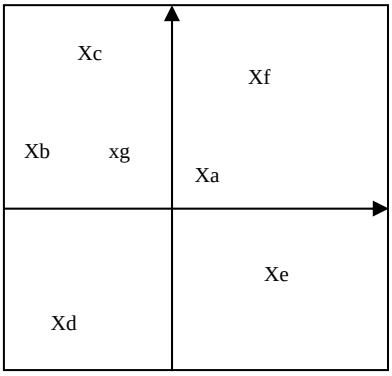
Un *croisement multi-site* (3,7,10) donne :

$C_1' = 0011100100101$ et $C_2' = 1010100010111$

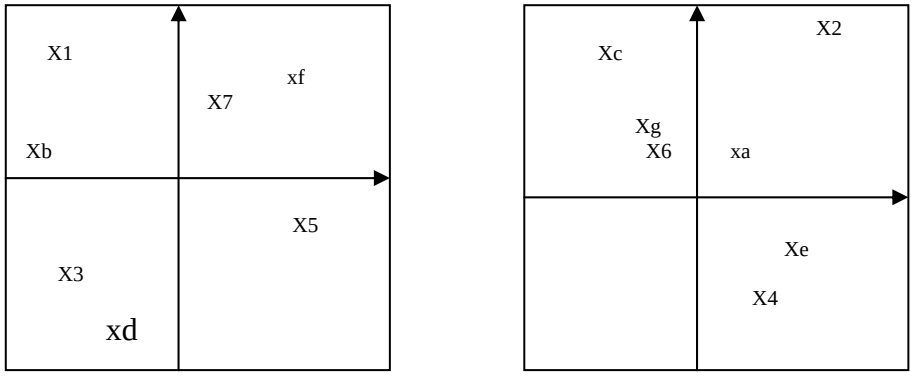
et nous avons bien $C_1 + C_2 = C_1' + C_2' = 1110000111100$



Parent 1



parent 2



Enfant1 **enfant2**

Fig. 4-Croisement de deux configuration planaires

II.5.4.3- Croisement local :

Relativement à deux individus X_1 *et* X_2 le croisement local s'effectue comme suit:

On choisit aléatoirement, dans l'esprit des *AG*, des couples (x_i^1, x_i^2) de composantes d'un couple d'individus parents . L'une des composantes, x_i^1 par exemple, du couple ainsi sélectionné est remplacée par un réel x_1', tiré au hasard dans *[a,b]* où :

$$a = \max(x_i^1, x_i^2) - \min(x_i^1, x_i^2)$$

$$b = \min(x_i^1 + x_i^2, Sup)$$

et la composante x_i^2 est substituée, conformément à (II-5.4.2), par $x_i^1 + x_i^2 - x_1'$

Sup, Max(.) et *min*(.) désignent respectivement la borne supérieure de l'espace réel de recherche, les fonctions maximum et minimum usuelles.

En définitive notre croisement dit local est pour notre problème, une adaptation très intéressante du croisement *multi-site* classique sur des chromosomes produits par concaténation des codes binaires.

II.5.4.4- Mutation :

La mutation d'un gène (site) dans notre problème d'échantillonnage spatial dans un domaine borné $D \subset IR^2$ se fait comme suit :

Soit S_k le site identifié par ses coordonnées (x_k, y_k).On note S_k' : le gène muté connu par ses coordonnées (x_k', y_k').

En supposant que $(x_k, y_k) \in [a,b] \times [c,d]$.

La mutation se fait à travers l'opérateur de mutation qu'on définit comme suit :

$$\begin{cases} x_k' = x_k \quad \text{Sgn} \quad U \\ y_k' = y_k \quad \text{Sgn} \quad V \end{cases}$$

Tel que : $U \in \begin{cases} U_{[x_k,b]} & si \quad x_k \geq \dfrac{a+b}{2} \\ U_{[a,x_k]} & si \quad x_k < \dfrac{a+b}{2} \end{cases}$ $V \in \begin{cases} U_{[y_k,d]} & si \quad y_k \geq \dfrac{c+d}{2} \\ U_{[c,y_k]} & si \quad y_k < \dfrac{c+d}{2} \end{cases}$

Et $U_{[a,b]}$: uniforme sur *[a,b]*.

$$\text{Sgn} = \begin{cases} (+) & si \quad x_k = a \quad ou \quad y_k = c \\ (-) & si \quad x_k = b \quad ou \quad y_k = d \\ (+,-) & avec \quad une \quad proba \quad \dfrac{1}{2}. \end{cases}$$

II.5.5- Critères de terminaison pour les algorithmes génétiques

Un premier critère, simple, est de stopper la procédure lorsque tous les individus sont identiques, en d'autres mots, quand la population s'est regroupée à l'intérieur d'une sphère de rayon donné.

Un autre critère consiste à tester l'amélioration de la fonction d'adéquation du meilleur individu au cours des générations successives. Cependant , le premier critère peut conduire à des temps de recherche excessifs, tandis que le second peut ne peut pas être satisfaisant pour les fonctions caractérisées par un relief présentant des plateaux.

Dans notre étude, nous avons décider d'arrêter la recherche après un nombre fixe d'itérations. Ce nombre a été choisis après quelque essais sur la fonction d'évaluation. Ceci représente une solution de compromis entre les contraintes de convergence de la population, le temps de calcul et la précision.

II.6 Description de l'Algorithme

L'algorithme génétique élitiste ci-dessous calcule une matrice solution *X* du problème de minimisation de la fonction objective *f*.

Le schéma de base de cet algorithme destiné à optimiser directement la fonction objective, se déroule de façon récursive comme suit :

f : fitness ou fonction d'évaluation,

n : taille de la population,

c_μ : fonction de croisement. Cette fonction peut être une composée de différents types de croisement.

m_υ :fonction de mutation.

Début
$t \leftarrow 0$
μ probabilité de croisement
υ probabilité de mutation
X(t) population initiale
Evaluer *X*(t)
Répéter

$f(X_m(t)) = \min_{1 \le i \le n} f(X_i(t))$
Si $X_m(t)$ est "bon" **STOP**
$t \leftarrow t+1$
Générer *Y (t)* par *reproduction - sélection*

$Y'(t) = c_\mu[Y(t)]$ *croisement*
$Y''(t) = m\upsilon[Y'(t)]$ *mutation*
jusqu'à " *bonne solution* "
Fin

III- Quantification Vectorielle :

III-1. Introduction :

Nous pourrions simplifier en désignant les quantificateurs comme les convertisseurs analogiques- numériques utilisés dans les appareils électroniques tels que les dispositifs de mesure, les systèmes d'enregistrement et ceux de communication. En effet, ces derniers ne manipulent que des données numériques ayant été mises en forme par un quantificateur.

Si nous nous plaçons plus précisément dans le domaine du codage de sources, en plus d'une représentation du signal, souple et adaptée aux traitements, la quantification permet une compression supplémentaire de l'information ceci au prix d'une perte que l'on peut contrôler.

Cette technique se prête donc particulièrement au codage d'images qu'il faut comprimer pour les transmettre ou les archiver.

On rappelle que l'efficacité de la technique de quantification a été prouvé dans le cas de l'échantillonnage asymptotique par *Benhenni*, de plus, *K-Boukhetala* et *Z-Habib* ont appliqué la technique de quantification sur l'échantillonnage optimal dans le cas univarié, sur les erreurs d'observations corrélées.

Afin de généraliser les études précédentes, Dans notre cas, la quantification va être appliquée aux erreurs d'observation spatialement corrélées, au cas de l'échantillonnage spatial optimal.

III.2- La quantification vectorielle (QV) :

III.2.1- Définitions :

La quantification consiste en l'approximation d'un signal d'amplitude continue par un signal d'amplitude discrète.

La *quantification* vectorielle **[22]** (notée **QV**) consiste alors à représenter tout vecteur x de dimension *k* par un autre vecteur y_i de même dimension mais ce dernier appartenant à un ensemble fini *D* de *L* vecteurs. Les y_i sont appelées les vecteurs représentants, les vecteurs de reproduction ou les code vecteurs. *D* est appelé le dictionnaire ou le catalogue des formes.

Il n'y a rien de mystérieux à considérer des espaces de grandes dimensions, pour mieux appréhender les raisonnements il suffit de s'avoir que tout s'organise autour des coordonnées des vecteurs, qu'il n'y a pas lieu de s'imposer une représentation mentale géométrique. Pour l'illustrer nous précisons qu'un vecteur x de l'espace IR^k est simplement un vecteur colonne constitué de *k* nombres réels x_i : $X = (x_1, x_2, ..., x_k)^T$, et que par exemple, une sphère entièrement caractérisée par son centre $u = (u_1, u_2, ..., u_n)^T$ son rayon ρ est constitué des points dont les coordonnées satisfont : $\sum_{i=1}^{k} (x_i - u_i)^2 = \rho^2$.

Un *Quantificateur Vectoriel* de dimension *k* et de taille *L* peut être défini mathématiquement comme une application *Q* de IR^k vers *D* :

$$Q : IR^k \rightarrow D$$
$$x \rightarrow Q(x) = y_i \quad \text{avec } D = \{y_i \in IR^k, i = 1,2,...., L\}$$

Cette application *Q* détermine implicitement une partition de l'espace source IR^k en *L* régions C_i. Ces régions encore appelées *classes* ou régions de *Voronoï* sont déterminées par :

$$C_i = \{x \in IR^k, Q(x) = y_i\}$$

Les conditions suivantes sont satisfaites :

$$\bigcup_{i=1}^{L} C_i = IR^k \quad ; \quad C_i \cap C_j = \varnothing, \quad i \neq j, \quad \begin{matrix} \forall i = 1,2,..., L \\ \forall j = 1,2,.....L \end{matrix} \quad .$$

Remarque :

Dans le but d'alléger les notations, l'abréviation **QV** est utilisée pour désigner selon le contexte, soit un quantificateur vectoriel, soit la quantification vectorielle.

III.2.2- QV appliquée à la compression de données :

III.2.2.1 La compression :

La compression ou codage de données vise à diminuer la quantité des éléments binaires nécessaire à la représentation de l'information contenue dans le signal à transmettre ou à archiver. Cette diminution peut autoriser ou non la perte d'information **[23].**

Principe : Soit $\{x_n\}$ l'ensemble des échantillons du signal source :

- **Coder** : c'est déterminer une loi de codage *C* telle que, $C : \{x_n\} \rightarrow \{y_n\}$.
- **Décoder** : c'est définir une loi de décodage *D* telle que, $D : \{y_n\} \rightarrow \{\hat{x}_n\}$.

Si $D^{-1} = C$, il s'agit d'un code sans perte d'information dit *réversible* (codage statistique ou entropique).

Si $D^{-1} \neq C$, il s'agit d'un code avec perte d'information dit *irréversible.*

Une bonne compression est réalisée si on réunit :

- La loi *C* minimisant le nombre d'éléments binaires y_n à transmettre,
- La loi *D* assurant une bonne reconstruction du signal source $\{x_n\}$ par les $\{\hat{x}_n\}$.

III.2.2.2- QV appliquée au codage :

La quantification Vectorielle offre la combinaison des opérations de codage et de décodage.
En effet, considérons *I* l'ensemble des *L* index des vecteurs de reproduction y_i du dictionnaire *: I* $= \{1,2,....,L\}$

Alors la loi de codage est déterminée par :

$$\boldsymbol{C}: \begin{array}{ccc} IR^k & \rightarrow & I \\ x & \rightarrow & C(x) = i \end{array}$$

Le processus de décodage est lui défini par : $\boldsymbol{D}: \; I \rightarrow \; D$

$$i \rightarrow \; D(i) = y_i$$

Finalement nous obtenons : $Q = \boldsymbol{DoC}$.
Un **QV** se décompose donc en deux applications : un codeur et un décodeur.

III.2.2.2.1- Le codeur :

Le rôle du codeur consiste, pour tout vecteur x du signal d'entrée, à rechercher dans le dictionnaire *D* le code-vecteur y_i le plus proche de *x.*

Nous introduisons ici la définition générale de la métrique L_p , où la norme d'un vecteur *x* de dimension *k* est : $L_p(x) = \|x\|^p = \sum_{j=1}^{k} |x_i|^p$

La distance entre 2 vecteurs x_1 *et* x_2 est alors : $\|x_1 - x_2\|^p = d_p(x_1, x_2) = \sum_{j=1}^{k} |x_{1j} - x_{2j}|^p$.

La notion de proximité que nous avons mise en place est la *distance euclidienne*:

$d(x, y_i) = \sum_{j=1}^{k} (x_j - y_{ij})^2$.

C'est donc le cas particulier de L_p où *p = 2*. Cette distance mesure la distorsion entre les vecteurs x et y_i .

Les régions de *voronoï* sont alors données par :

$C_i = \{ x \in IR^k \setminus Q(x) = y_i, si, d(x, y_i) \le d(x, y_j), \forall j \ne i \}$

Chaque région contient un ensemble de vecteurs de IR^k et tous les vecteurs *x* qui appartiennent à C_i sont représentés par le même vecteur y_i du dictionnaire.

La compression d'information est réalisée à ce niveau car c'est uniquement l'index du représentant y_i minimisant le critère de distorsion qui sera transmit ou stocké au lieu d'un vecteur. Cette opération, qui perd de l'information, fait que la **QV** est une *opération irréversible*, le signal original ne pourra plus être restitué tel quel.

La quantité d'information requise pour représenter le vecteur source est donnée par *R* le *débit binaire* en *bits* par composante du vecteur, ou *bits par dimension*, ou encore *bits par échantillon:* $R = \frac{1}{k} \log_2 L$

III.2.2.2.2- Le décodeur :

Le décodeur est considéré comme un récepteur voué à la reconstruction du vecteur source, pour cela il dispose d'une réplique du dictionnaire qu'il consulte afin de restituer le code vecteur correspondant à l'index qu'il reçoit.
Le décodeur réalise donc l'opération de décompression.

III.2.2.2.3- Principe de la QV :

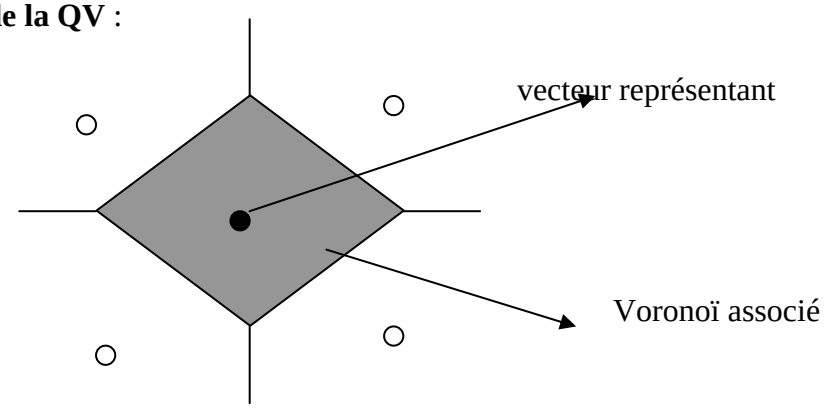


Fig. 1 - Principe de la quantification vectorielle

III.2.2.2.4- Evaluation des performances du système :

On peut déterminer les performances d'un système de quantification en calculant la distorsion moyenne entre l'entrée et la sortie.
Un système sera d'autant plus performant que sa distorsion moyenne sera faible. D'où l'intérêt de minimiser la distorsion moyenne.

Précisément la mesure de distorsion doit réunir 3 conditions essentielles **.** En effet cette fonction réelle positive doit être :

- simple à calculer

- utilisable par un algorithme de minimisation

- significative telle qu'une grande distorsion implique une mauvaise qualité de la restitution subjective du signal (et inversement).

Le choix d'une erreur quadratique moyenne (la mesure de distorsion moyenne la plus répandue qui s'interprète comme la puissance de l'erreur de quantification) permet de vérifier les 2 premières conditions, cependant elle ne caractérise que pauvrement la dégradation qualitative faite au signal. La troisième condition serait parfaitement obtenue au prix de l'introduction d'une pondération spychovisuelle adaptée au codage des séquences d'images.
Cependant nous n'avons pas encore mis en place cette pondération.

III.3- La quantification scalaire (QS) :

III.3.1-La quantification scalaire uniforme :

La quantification scalaire **[22]** notée (**QS**) est une forme particulière de la (**QV**) où la dimension des vecteurs est égale à un.

Afin d'introduire différentes notions de bruits rencontrées en quantification, nous allons présenter le plus simple des quantificateurs : le quantificateur scalaire uniforme à débit fixe.

Ce type de quantificateur est entièrement déterminé par :

- Les $L+1$ niveaux de décisions: $x_0, x_1, ..., x_L$, qui partitionnent en L intervalles égaux de l'axe des réels *IR* et déterminent le pas de quantification.

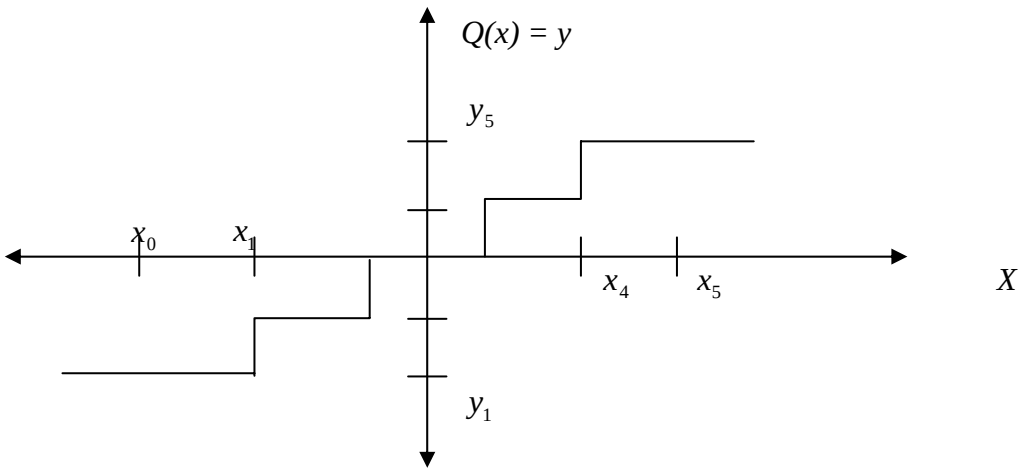
- Les L valeurs de reproduction: $y_1, y_2, ..., y_L$, qui sont les centres de masse de chacun des intervalles de décision.

Le débit de ce quantificateur est donc donné par $R = \log_2 L$ [bit \ échantillon].

Il apparaît deux sortes d'erreurs ou bruits de quantification :

- Le *bruit granulaire* qui se produit lorsque la valeur d'entrée x se situe dans l'une des cellules $[x_i, x_{i+1}]$. l'erreur résultante qui est la différence entre x et $Q(x)$ peut être majorée par un demi pas de quantification.

- *Le bruit de surcharge* ou de dépassement qui se produit lorsque la valeur d'entrée se situe hors de l'intervalle $[x_0, x_L]$. La valeur de reproduction est alors soit y_1 soit y_L, et l'erreur résultante supérieure à un demi pas de quantification.



III.3.2- Le QS optimal :

Le QS optimal est celui qui minimise, pour une source donnée et un débit fixe, l'erreur moyenne de reconstruction due aux bruits de quantification et de surcharge. Les niveaux de reconstruction sont donc répartis en tenant compte de la densité de probabilité de la variable à quantifier. Intuitivement nous comprenons que la concentration des niveaux de reconstruction est plus importante dans la zone de l'espace où la densité de probabilité des vecteurs à quantifier est plus élevée. En fait ce problème d'optimisation à une solution unique et le **QS** obtenu est connu sous le nom de quantificateur de *Lloyd-Max* . **[24]**

III.4- Eléments de la théorie de l'information :

III.4.1- Entropie : *les références bibliographiques relatives à cette partie sont* : **[25],[26]**

Considérons une source à temps discret et ergodique *{x(n)}, n*= $0 \pm 1 \pm 2,...$
Le flux des symboles *x(n)* forme une séquence aléatoire *{X(n)}, n*= $0 \pm 1 \pm 2,...$ dont les réalisations x_k appartiennent à l'alphabet de la source $A_k = \{x_k / k = 1,2,....K\}$.
Si *K* est fini, la source est dite à amplitude discrète, si *K* est infini, la source est à amplitude continue.

La source est *sans mémoire* si les échantillons successifs sont statistiquement indépendants.

III.4.1.1 Source à amplitude discrète et sans mémoire :

Soit P_k la probabilité d'occurrence du symbole x_k : $P_k = \Pr\{X(n) = x_k\}$.

Alors une appréciation numérique de l'information reçue est donnée par : $I(x_k) = -\log_2 P_k$ [en bits].

La quantité d'information moyenne reçue ou entropie est alors [en bits/échantillon] :

$$H(X) = E(I(X)) = -\sum_{i=1}^{K} P_k . \log_2 P_k \qquad et \qquad 0 \le H(X) \le \log_2 K.$$

On a :

- Si *H(x) = 0,* la source est totalement *prédictible*.
- Si *H(x)* = $\log_2 K$, la source est *non prédictible*, les symboles sont équiprobables.

$\log_2 K$: mesure la capacité de l'alphabet.

Les *redondances* de la source sont appréciées en calculant *R(X)* = $\log_2 K$ - *H(X)* .

III.4.1.2- Source à amplitude discrète avec mémoire :

Dans ce cas, il existe des dépendances statistiques entre les échantillons successifs de la source.

Soit *x = x(n) =(x(n), x(n + 1),…., x(n + N- 1))*^{T} , un vecteur constitué de N échantillons successifs de la source.

Le vecteur précédant est caractérisé par sa probabilité jointe *P(x)* qui est indépendante du temps si nous considérons une source stationnaire.

Alors *x* est une réalisation du vecteur aléatoire $X = (X(n),X(n+1),\ldots,X(n+N-1))^T$.

Soit l'entropie des vecteurs aléatoires [en bits/échantillon]:

$$H_N(X) = \frac{1}{N}.E(-\log_2 P(X)) = \frac{-1}{N}.\sum\sum.....\sum_X P(X).\log_2 P(X).$$

On a :

$$H(X) = \lim_{N\to\infty} H_N(X).$$

On considère aussi l'entropie conditionnelle d'un symbole à l'instant *n* étant donnés, les *(N - 1)* symboles précédents : *H (X(n)/ X(n - 1),X(n- 2),…., X(n- N + 1))*

Il est démontré que :

$$H(X(n)/\ \ X(n-1), X(n-2),....., X(n-N+1)) \le H_N(X)$$

et que $$\lim_{N\to\infty} H(X(n)/\ \ X(n-1), X(n-2),....., X(n-N+1)) = H_N(X)$$

Les deux fonctions $H_N(X)$ et $H(X(n)/\ \ X(n-1), X(n-2),....., X(n-N+1))$ sont décroissantes avec N.

De plus, pour deux sources ayant deux alphabets identiques et une même probabilité pour les symboles, il est prouvé que :

(H(X) / source avec mémoire*) ≤ (H(X)* / source sans mémoire*).*

III.4.1.3- Entropie d'un dictionnaire :

remarque : Le dictionnaire généré pour atteindre une distorsion moyenne donnée, est un exemple de source à amplitude discrète.

En gardant les mêmes notations que celles du paragraphe (III-2.2) : chaque vecteur est de taille *k* est associé un index *i* parmi les *L* index de l'alphabet I.

L'entropie de ce dictionnaire est calculée par formule : $H(D) = \frac{1}{k}.\sum_{i=1}^{L} P_i(-\log_2 P_i).$

- Avec : P_i : probabilité de sélectionner le vecteur y_i du dictionnaire D.
- $(-\log_2 P_i)$: l'appréciation numérique sur l'incertitude de sélectionner y_i, l'unité d'information étant le bit.

III.4.1.4- Source à amplitude continue et sans mémoire :

Soit {*X(n)}*une source stationnaire et sans mémoire. Nous considérons aussi qu'elle est centrée.
Soit $P_x(x)$ sa fonction de densité, $\sigma_x^2 = E(X^2(n))$ sa variance et $R_{xx} = \sigma_x^2.\delta(k)$ sa fonction d'autocorrélation (*δ(k)* : étant l'indice de *Kronecker*).
Le signal a une entropie absolue infinie, en effet $P(x_k) = 0 \Rightarrow H(X) = +\infty.$
C'est pourquoi on introduit *l'entropie différentielle* (qui peut être positive, négative ou nulle en fonction de l'amplitude de la source) :

$$h(X) = E(-\log_2 P_x(X)) = -\int_{-\infty}^{+\infty} P_x(x).\log_2 P_x(x)dx$$

Il est démontré que l'entropie différentielle est maximale si la source suit une loi Gaussienne N $(0, \sigma_x^2)$. Dans ce cas :

$$P_x(x) = \frac{1}{\sqrt{2\pi\sigma}}.e^{-1/2(x^2/\sigma_x^2)} \quad \text{et} \quad h(X)_G = \log_2 \sqrt{2.\pi.e.\sigma_x^2}$$

Il est aussi pratique de définir la puissance *entropique* qui traduit la répartition de l'information par unité de variance du signal :

$$Q = \frac{1}{2.\pi.e}.2^{2.h(X)}$$

Pour une source gaussienne $Q = \sigma_x^2$ et pour une non gaussienne $Q < \sigma_x^2$.

III.4.1.5- Source à amplitude continue avec mémoire :

Nous considérons cette fois un vecteur x constitué de N échantillons successifs de la source. x est décrit par sa fonction de densité de probabilité jointe $P_x(x)$ et on définit l'entropie différentielle par :

$$h(X) = \lim_{N \to \infty} h_N(X)$$

avec $$h_N(X) = \frac{1}{N}.E(-\log_2 P_x(x)) = -\frac{1}{N}\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}......\int_{-\infty}^{+\infty} P_x(x).\log_2 P_x(x)dx$$

De façon générale :

(*h(X)* = source avec mémoire) < (*h(X)* / source sans mémoire) $\leq \log_2(2.\pi.e.\sigma_x^2)$.

La borne supérieure est atteinte dans le cas d'une source gaussienne, les entropies inférieures à cette valeur sont dues aux redondances de la source.

La puissance entropique d'une source gaussienne "colorée" (c.a.d avec mémoire) est : $Q = \gamma_x^2.\sigma_x^2$.

où γ_x^2 : est *la mesure de platitude du spectre*. Nous avons : $0 \leq \gamma_x^2 \leq 1$.

Si $\gamma_x^2 = 1$, on retrouve le cas d'une *gaussienne* sans mémoire.

Si $\gamma_x^2 < 1$, la source présente une mémoire, le signal est plus ou moins *prédictible*.

L'inégalité suivante est alors prouvée :

(Q / source avec mémoire) < (Q =source sans mémoire) $\leq \sigma_x^2$.

III.5- Supériorité de la quantification vectorielle sur celle scalaire :

La théorie de l'information annonce donc d'une façon générale que de meilleures résultats en codage sont obtenus si l'on quantifie des vecteurs plutôt que des scalaires. En contre partie une complexité des systèmes ainsi que des délais de calcul plus importants sont nécessaires. Ce gain de la *QV* sur la *QS* est notamment dû à l'exploitation de la mémoire de la source (c.a.d des corrélations existant entre les coordonnées des vecteurs).
Cependant *Zador* a théoriquement prouvé la supériorité de la *QV* sur la *QS* même si les échantillons de la source sont statistiquement indépendants.

Résultats de Zador :

Nous rappelons les notations du (III-2.2) :

- *x* est le vecteur d'entrée de dimension *k*, soit *p(x)* sa fonction de densité de probabilité.
- le quantificateur choisit parmi *L* vecteurs de reproduction le y_i le plus "proche" de *x*.
- C_i est le voronoï associé à y_i, l'ensemble des C_i recouvre IR^k et ces cellules sont disjointes entre elles.
- la notion de proximité (soit l'amplitude de l'erreur introduite par la quantification) est mesurée par la distance euclidienne *d(x; y_i)*.
- *l'erreur quadratique moyenne* [par dimension] est :

$$E = \frac{1}{k} \int_{IR^k} d(x, y_i).P(x)dx = \frac{1}{k} \sum_{i=1}^{L} \int_{C_i} d(x, y_i).P(x)dx$$

Le problème est : pour *k*, *L*, et *p(x)* donnés nous recherchons le quantificateur optimal (c.a.d les y_i) qui minimisent *E*, tel que *E(k, L, p)* = $\min_{y_i} E$

E(k, L, p) est l'erreur minimale que l'on peut atteindre.
P.L. Zador a montré qu'il est possible de réduire E en quantifiant des vecteurs de grandes dimensions **[27].**

Exactement il a prouvé que : $\lim_{L\to\infty} L^{2/k}.E(k, L, p) = G_k \left(\int_{IR^k} P(x)^{k/k+2} dx \right)^{k+2/k}$

La formule précédente peut aussi réécrite (sachant $L = 2^{R.k}$, *R* étant le débit binaire) :

$$\lim_{L\to\infty} E(k, L, p) = 2^{-2.R}.G_k \left(\int_{IR^k} P(x)^{k/k+2} dx \right)^{\frac{k+2}{k}}$$

La répartition optimale des vecteurs de reproduction est donc celle proportionnelle à $p(x)^{\frac{k}{k+2}}$

Le paramètre G_k ne dépend que de la dimension *k*.

Zador a montré que : $\frac{1}{(k+2).\pi}.\Gamma(\frac{k}{2}+1)^{\frac{2}{k}} \le G_k \le \frac{1}{k.\pi}.\Gamma(\frac{k}{2}+1)^{\frac{2}{k}}.\Gamma(1+\frac{2}{k}).$

Donc, si $k \to +\infty$: $G_k \to \frac{1}{2.\pi.e} \approx 0.05855.$

Interprétation et calcul de G_k :

$$G_k = \frac{\lim_{L\to+\infty} L^{2/k}.E(k,L,p)}{(\int_{IR^k} p(x)^{\frac{k}{k+2}} dx)^{\frac{k+2}{k}}}$$

On considère le cas simple où x est uniformément distribué sur une grande région de IR^k. Alors sur cet espace :

$$p(x) = \frac{1}{\sum_{i=1}^{L}\int_{C_i} dx}$$

E est minimale si chaque y_i est au centre de son voronoï. En considérant le cas asymptotique, on aura : $\mathrm{E} = \frac{1}{k}.\frac{\sum_{i=1}^{L}\int_{C_i} d(x,y_i)dx}{\sum_{i=1}^{L}\int_{C_i} dx}$

Nous calculons : $\left(\int_{IR^k} p(x)^{\frac{k}{k+2}} dx\right)^{\frac{k+2}{k}} = \left(\sum_{i=1}^{L}\int_{C_i} p(x)^{\frac{k}{k+2}} dx\right)^{\frac{k+2}{k}} = \left(\sum_{i=1}^{L}\int_{C_i} dx\right)^{\frac{2}{k}}$

Alors : $G_k = \lim_{L\to+\infty} \frac{E(k,L,p)}{(\frac{1}{L}\sum_{i=1}^{L}\int_{C_i} dx)^{\frac{2}{k}}}$

G_k s'interprète donc comme le rapport de l'erreur quadratique moyenne [par dimension] en considérant la quantification optimale d'une source uniforme et des conditions asymptotiques, sur un facteur qui rend G_k sans dimension.

G_k est alors tabulé pour une densité uniforme de la source.

En notant $\int_{C_i} dx = Vol(C_i)$:, alors : $G_k = \frac{1}{k}.\frac{\frac{1}{L}\sum_{i=1}^{L}\int_{C_i} d(x,y_i)dx}{\left(\frac{1}{L}\sum_{i=1}^{L} Vol(C_i)\right)^{1+\frac{2}{k}}}$

La source considérée étant uniforme, on adopte le cas où les vecteurs de reproduction sont les points d'un réseau régulier. Les voronoï sont alors congrues au même polytope II, alors :

$$G_k(II) = \frac{1}{k}\frac{\int_{II} d(x,0)dx}{Vol(II)^{1+\frac{2}{k}}} = \frac{1}{k}\frac{\int_{II} x^T.x \quad dx}{Vol(C_0)^{\frac{2}{k}+1}}$$

On retrouve le moment d'ordre 2 de II.

Par exemple si *k = 1*, les représentants sont uniformément distribués sur la droite réelle, on construit le réseau tel que les voronoï sont des intervalles de longueur 1, II est l'intervalle $\left[\frac{-1}{2}.+\frac{1}{2}\right]$ (c'est le réseau Z). On obtient :

$$G_1(II) = \frac{\int_{-1/2}^{+1/2} x^2 dx}{\int_{-1/2}^{1/2} dx} = \frac{1}{12} \approx 0.08333$$

Ce résultat est classique en quantification : si une variable uniforme est quantifiée scalairement, l'erreur quadratique moyenne est *1/12* ($> \frac{1}{2.\pi.e}$). Zador a donc montré que cette erreur peut être réduite en utilisant des quantificateurs de dimensions spatiales élevées. Déjà, pour *k = 2*, on trouve en utilisant le réseau régulier hexagonal G2(II) = $\frac{5}{36.\sqrt{3}} \approx 0{,}080175.$

III.6 QV optimale :

III.6.1- Approche intuitive :

Comme nous l'avons expliqué, quantifier consiste à répartir dans un espace de dimension fixée un nombre déterminé de représentants, ce nombre étant fonction du débit alloué au quantificateur. L'efficacité du quantificateur se mesure alors à la qualité de restitution du signal source qui doit être la plus fidèle possible (c.a.d avec une erreur de reconstruction minimale).

La **QV** se révèle supérieure à la **QS** car elle autorise :

- L'utilisation meilleure de la mémoire de la source (explication des corrélations existants entre les coordonnées des vecteurs).
- Une liberté pour la répartition dans l'espace des représentants.

III.6.2- Quantificateur optimal- Définitions - Théorème :

Pour une distribution statistique donnée de la source est un débit fixé :

- Le *quantificateur globalement optimal* est celui qui minimise la distorsion moyenne.
- Un *quantificateur localement optimal* a un dictionnaire qui peut être légèrement perturbé sans que la distorsion moyenne n'augmente.

La théorie qui établit la supériorité de la **QV** sur la **QS** n'est pas constructive, elle ne décrit pas la façon de concevoir le dictionnaire localement optimal. Seuls des propriétés suffisantes sont connues qui permettent de construire des dictionnaires localement optimaux.
Le quantificateur se décompose en 2 applications : un codeur et un décodeur. Le quantificateur (localement) optimal est alors celui réunissant : **[28]**

- Le *codeur optimal* (pour un dictionnaire fixé), celui-ci respecte la "régle du plus proche voisin" que nous avons déjà décrit : $\forall j, d(x, y_i) < d(x, y_j) \Rightarrow C(x) = y_i$

cette régle détermine la répartition de l'espace IR^k en voronoï.

- Le *décodeur optimal* (pour une partition de IR^k donnée), le vecteur représentant y_i doit minimiser la distorsion associée C_i, y_i est donc le centroïde de cette cellule : $y_i = cent(C_i)$.
- Une troisième condition est nécessaire : il faut que la probabilité d'avoir un vecteur à coder à la même distance de deux représentants soit nulle, sinon ce vecteur source est affecté à l'un des 2 représentants, la partition optimale de l'espace n'est plus et donc la condition de décodeur optimal est devenue impossible, si les vecteurs source sont à amplitude continue, cette troisième condition est toujours vérifiée.

III.6.3- Algorithme de Lloyd Généralisé :

III.6.3.1- Description :

Les 3 conditions précédantes conduisent à la conception d'un algorithme qui réalise, à partir d'une séquence d'apprentissage représentative de la statistique de la source à coder, la construction d'un algorithme (localement) optimal.

Cet algorithme de classification appelé algorithme des " *K moyen* " est l'extension au cas vectoriel de l'algorithme de *Lloyd-Max* du cas scalaire.

Il s'agit d'un algorithme d'optimisation itératif opérant à partir d'un dictionnaire initial.

A chaque itération (dite itération de Lloyd), 2 opérations distinctes sont appliquées :

- Une *classification* suivant la règle de codage optimal.
- Une *optimisation* suivant la règle de décodage optimal.

Chaque itération de Lloyd, en modifiant localement le dictionnaire, réduit ou laisse inchangé la distorsion moyenne. L'algorithme converge en un nombre fini d'itérations vers le minimum local le plus proche correspondant au dictionnaire initial. Le choix de ce dernier est donc capital.

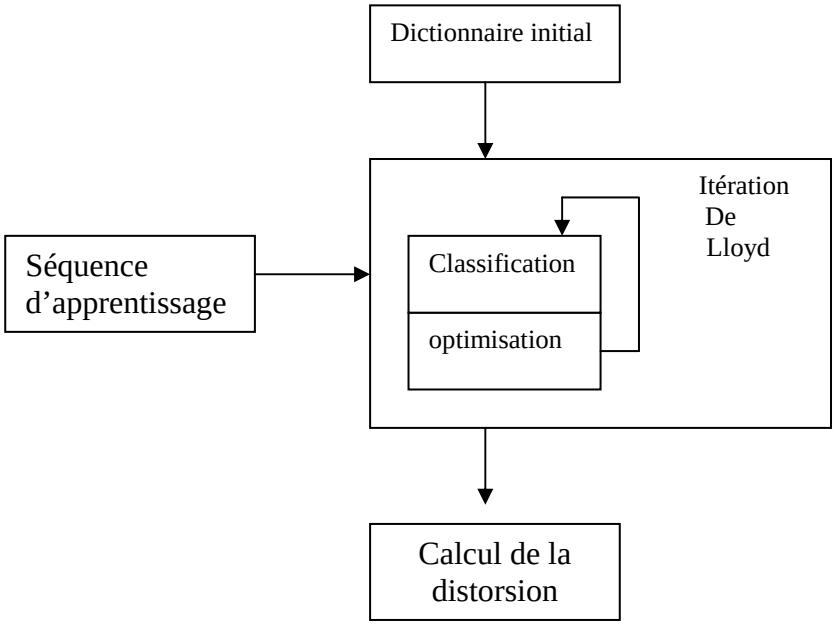


Fig.3_ Schéma du fonctionnement de l'algorithme de Lloyd

III.6.3.2 - Choix du dictionnaire initial :

Le choix de ce dictionnaire initial est essentiel car il conditionne les résultats finaux de l'algorithme. Plusieurs méthodes ont été proposées pour le déterminer.

III.6.3.2.1 Initialisation Aléatoire :

Le dictionnaire le plus simple est celui qui contient les *L* premiers vecteurs de la suite d'apprentissage ou *L* vecteurs extraits aléatoirement de cette suite. Ces vecteurs peuvent bien sûr ne pas être du tout représentatifs de la suite d'apprentissage et on aboutit à des résultats très médiocres.

III.6.3.2.2- L'Algorithme à Seuil :

Au lieu de prendre *L* vecteurs aléatoirement, on fixe une distance minimale entre les éléments du dictionnaire initial. Cette méthode permet d'obtenir une meilleure représentativité que dans le cas précédent mais n'est toujours pas satisfaisante, le seuil étant souvent difficile à déterminer puisque dépendant de la complexité de la séquence d'apprentissage.

III.6.3.2.3 Méthode par Dichotomie Vectorielle :

Cette méthode référencée comme étant l'algorithme de **LBG** combine à l'itération de Lloyd une technique de "splitting". Celle-ci consiste à découper chaque vecteur représentant y_i en 2 nouveaux vecteurs $y_i + \varepsilon$ *et* $y_i - \varepsilon$ (ε étant un vecteur de perturbation de faible énergie), avant d'appliquer au nouveau dictionnaire obtenu les itérations de Lloyd. Le dictionnaire initial est alors le centroïde de la séquence d'apprentissage, puis l'algorithme génère une succession de dictionnaires (à chaque boucle le nombre de vecteurs de reproduction est multiplié par 2).

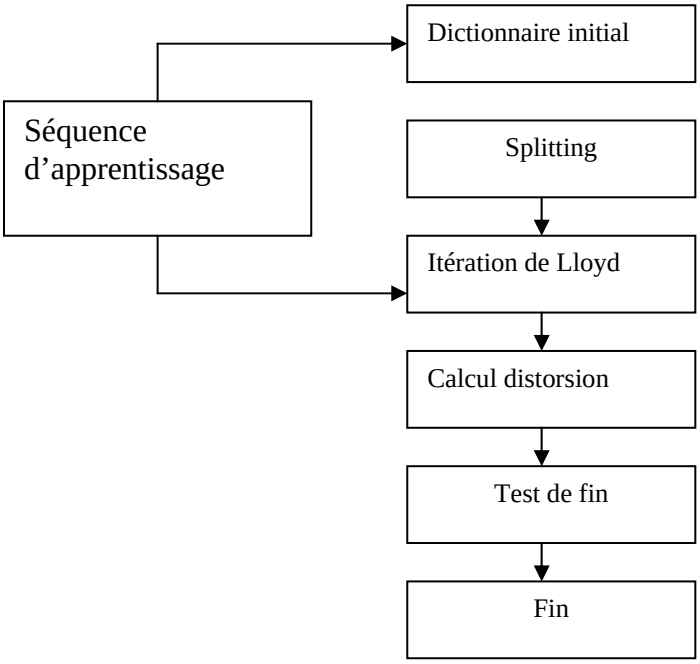


Fig 4- Schéma de fonctionnement de l'algorithme LBG

III.6.3.2.4- Complexité :

III.6.3.2.4.1 Complexité de la Construction du Dictionnaire :

La complexité de l'algorithme est fonction :

- des paramètres du dictionnaire (la dimension de l'espace *k*, le nombre de vecteurs représentants *L*).
- des paramètres de la séquence d'apprentissage (le nombre de vecteurs *M*).

Lors de la construction du dictionnaire, la phase de classification de l'itération de Lloyd est la plus coûteuse en terme calculatoire. En effet chaque vecteur source doit être comparer à chacun des *L* représentants, il y a autant de calculs de distorsion dans IR^k et cela à chaque itération. Ainsi si *n* itérations sont exécutées (en considérant la métrique euclidienne) :

- le nombre de multiplications effectuées est $N_X = k.L.M.n.$

- le nombre d'addition effectuées est : $N_+ = (2.k - 1).L.M.n.$
- le nombre de comparaisons effectuées est $N_c = (L - 1).M.n.$

La phase d'optimisation (calcul des nouveaux centroïdes des classes) nécessite aussi $k.(M - L).n$ additions et $k.L.n$ multiplications.

Enfin $k.(L + M) + M$ emplacements mémoires sont nécessaires pour stocker les vecteurs de la séquence d'apprentissage, leurs classes (auquel chacun appartient) et les vecteurs du dictionnaire.

III.6.3.2.4.2 Complexité de la Recherche au sein du Dictionnaire :

Le dictionnaire obtenu ne possède aucune structure topologique particulière facilitant le codage. Ainsi pour trouver le vecteur représentant correspondant au vecteur de la source à coder, L calculs de distorsion sont nécessaires. Cette procédure de *recherche exhaustive* au sein du dictionnaire a donc une complexité d'ordre $L = 2^{R.k}$, (R étant le débit binaire). En détaillant, coder un vecteur nécessite :

- $L.k$ multiplications.
- $(2.k - 1).L$ additions
- $(L - 1)$ comparaisons

Or de bonnes performances ne sont atteintes qu'à a débit élevé (pour de petites dimensions). ou inversement avec de grandes dimensions (pour des débits faibles).

Ce bilan rédhibitoire a conduit à envisager de nouvelles méthodes de quantification vectorielle permettant un codage avec des coûts calculatoires moindres. D'une façon générale ces méthodes consistent à imposer au dictionnaire *des contraintes structurelles* afin de simplifier le codage (rapidité de décision), cependant les résultats obtenus sont toujours sous optimaux (le représentant choisit n'est pas toujours le plus proche voisin du vecteur à coder, cependant il en est proche en moyenne) : il s'agit alors de faire un compromis.

III.6.4- QV à l'aide des Réseaux Réguliers de Points : (QV en treillis) :

III.6.4.1- Treillis Meilleurs Quantificateurs :

La cellule de voronoi C_i associé au point y_i du réseau régulier Λ de la région de l'espace constituée des points x qui sont aussi proches de y_i que d'un autre point du réseau.

$$C_i = \left\{ x \in IR^k \ / \ \ d(x, y_i) \le d(x, y_j) \ \ \forall i \neq j \right\}$$

Les y_i sont donc séparés par des hyperplans et les régions obtenues sont des polytopes convexes dont l'union forme IR^k, aux sommets des intersections entre ces hyperplans se situent les trous du réseau. Les sommets d'un voronoi appartiennent aux hyperplans médiateurs entre les y_i, ce sont les points x pour lesquels les distances aux y_i sont des maxima locaux.

Le réseau étant régulier, tous les voronoi sont congrus au même polytope et ont le même volume égal à celui de la région fondamentale : vol(II) = vol(C_i) = vol(C_0) = $\int_{C_0} (\det \Lambda)^{1/2}$.

Ils ont aussi le même moment d'ordre 2 (ou moment d'inertie) normalisé :
$G_k(II) = G_k(C_i) = G_k(C_0)$

$$= \frac{1}{k} \frac{\int_{C_i} d(x, y_i)dx}{Vol(C_0)^{1+\frac{2}{k}}} = \qquad \frac{1}{k} \frac{\int_{C_0} d(x,0).dx}{Vol(C_0)^{1+\frac{2}{k}}} = \qquad \frac{\int_{C_0} \|x\|^2.dx}{Vol(C_0)^{1+\frac{2}{k}}}$$

On retrouve G_k le coefficient de " Zador " (Voir chapitre III-5).

Remarque :Le treillis Λ meilleur quantificateur est donc celui pour lequel $G_k(II)$ est minimal.

Là encore le problème est sans solution globale pour $k \geq 2,$ et on ne considère que les réseaux réguliers.

Conway et *Sloane* ont développé pour les treillis $Z^k \quad (k \geq 1), A_k \quad (k \geq 1), E_6, E_7, E_8, \Lambda_{16}$ et leurs réciproques, des algorithmes de quantification rapide.(Voir [29]).

$Z^k \ (k \geq 1)$ sont les réseaux cubiques. (c'est le réseau utilisé dans notre application).

Pour un réseau cubique Z^k, on a : $Z^k = \left\{ y = (y_1, y_2, ..., y_k)^T \ / \ y_i \in Z \right\}$

C'est l'ensemble des points de IR^k dont les coordonnées sont des entiers.

Soit $x = (x_1, x_2, ..., x_k) \in IR^k$ le vecteur à quantifier, on désire lui associer le vecteur de reproduction $y \in IR^k$ le plus proche.

Soit *f* la fonction qui, appliqué à un réel x_i , nous rend l'entier le plus proche. Dans le cas où x_i est équidistant de 2 entiers, *f* nous rend l'entier ayant la valeur absolue la plus petite (c.a.d si $x_i = l_i + 0.5 \mid l_i \in Z \quad alors, \quad f(x_i) = l_i)$, on choisit ainsi le vecteurs représentant ayant la plus petite énergie. On obtient : $y = f(x)$

$$= \left(f(x_1), f(x_2), ..., f(x_k) \right)^T$$

On effectue donc une quantification scalaire uniforme de pas unité sur chaque coordonnée du vecteur source.

Ce type de quantification sera celui qu'on va utiliser, du fait de sa simplicité du point de vu programmation, de plus, l'opérateur de quantification va être injecté dans l' algorithme génétique dont l'implémentation est déjà complexe.

Dans l'application qui va suivre, la quantification va être appliquée aux erreurs d'observation spatiallement corrélées afin de créer des perturbations permettant l'accélération de notre algorithme génétique.

IV- Application

Une première étape est de tester la performance des opérateurs génétiques proposés au chapitre (III.5), dans la résolution d'un problème d'échantillonnage spatial optimal sur la base d'optimisation d'un certain critère.

Par souci de continuité du travail réalisé jusqu'à présent, on a repris l'application de [19], afin d'améliorer la qualité (précision) des résultats obtenus à partir des méthodes d'optimisation utilisées.

IV.1-Description du problème :

Le but est l'installation d'un réseau pluviométrique optimal, qui permet d'observer une quantité de pluie donnée *M* sur une région $D \subseteq IR^3$ avec un nombre réduit de stations (*k*) et par conséquent on aura un coût réduit.

On a : $D = \left\{ (x, y, z) \mid x \in [540,800],\ \ y \in [150,400],\ \ z \in [456,1250] \right\}$.

Les limites du domaine D ont été fixé par les géographes pour des considérations appropriés.

- *s=(x, y, z)* : coordonnées du site pluviométrique dans la région *D*.
- *x, y* sont exprimés en kilomètres et *z* en mètres.

z : représente l'attitude du site pluviométrique.

Une analyse statistique (*Régression multiple*) sur la quantité de pluie qui tombe annuellement sur la zone *D* a donnée le résultat suivant :

Soit Q_{S_i} : la quantité de pluie qui tombe sur un site S_i avec $S_i = (x_i, y_i, z_i)$

$$Q_{S_i} = \langle \tilde{\alpha}, S_i \rangle + d + m.\varepsilon(S_i) \qquad \text{(Exprimé en millimètre).}$$

$\tilde{\alpha} = \begin{pmatrix} a \\ b \\ c \end{pmatrix}$, *m* : est un paramètre réel qu'on a pris égale à 1.

Les paramètres *a, b, c, d* sont estimés par la méthode des moindres carrés ordinaires :

$$\hat{a} = 60.05.10^{-3} \qquad \hat{b} = 0.701 \qquad \hat{c} = 0.29 \quad \hat{d} = -94.44.$$

Avec le coefficient de détermination $R^2 = 0.89.$

Les erreurs d'observation seront modélisés par *un processus aléatoire spatial* $\varepsilon_s(\omega)$, où $s \in D$ et ω sont des événements élémentaires de l'espace d'état $\Omega, i.e.\ \ \omega \in \Omega$.

$\varepsilon_s(\omega)$: est défini sur un espace de probabilité (Ω, F, P) et *F* une σ – algèbre de sous ensembles de Ω et *P* est la mesure de probabilité.

Le modèle du *processus aléatoire spatial* prend en considération les caractéristiques de variabilité, d'incertitude et d'hétérogénéité associé au processus de la pluviométrie.

L' analyse stochastique, la variabilité spatiale, se réduit au problème de construction à l'aide d'une moyenne μ_ε et covariance $C(\varepsilon_i, \varepsilon_j)$.

On suppose que le processus *aléatoire* spatial $\varepsilon(.)$ modélise les erreurs associées à chaque site, qui est supposé un *processus aléatoire spatial homogène et gaussien.*

Remarque :

Déterminer les sites optimaux revient à trouver l'échantillon optimal de taille fixé *k* en prenant comme critère d'optimalité, la minimisation de l'erreur quadratique moyenne :

$E(\overline{Q} - M)^2$ avec $\overline{Q}$: quantité de pluie moyenne issue des *k* sites choisis.

i.e, $\overline{Q} = \frac{1}{k}\sum_{i=1}^{k} Q_{S_i}$

M: Quantité de pluie ciblée.

Mathématiquement, cela peut être formulé comme suit

IV.2- Formulation Mathématique du Problème :

Soit $S = \left\{ s_i \in D \subset IR^3, i = 1...k \right\}$, l'échantillon de sites,de taille *k.*

On note que $|S| = k \le n$ tel que *n*: représente le nombre de points existants sur le domaine *D.*

n: peut être fini ou infini.

Le critère d'optimalité est donné par :

$$\sigma^2(S^*/\varepsilon) = \underset{S \in D}{Min} \left\{ E\left(\sum_{i=1}^{k} \frac{Q(s_i)}{k} - M \right)^2 \right\}$$

k : Taille de l'échantillon *S.*

S : L'ensemble des échantillons de taille *k.*

Q_{S_i} La quantité de pluie observée sur le site S_i.

$s_i^* = \left(x_i^*, y_i^*, z_i^*\right)$ $i = 1,...,k.$:les coordonnées de l'échantillon optimal de points spatiaux, S^*

$\varepsilon = (\varepsilon(s_i))_{i=1,...k}$: Vecteur des erreurs d'observation commises sur chaque site s_i.

E(.) : L'espérance mathématique définie par rapport à la loi de ε, qui est supposée être une variable aléatoire spatiale gaussienne.

IV.3-Forme Explicite de la Fonction Objectif :

On a : $\sigma^2(S^*;\varepsilon) = \underset{S\in D}{Min}\, E\left[\frac{1}{k}\left(\hat{a}\sum_{i=1}^{k} x_i + \hat{b}\sum_{i=1}^{k} y_i + \hat{c}\sum_{i=1}^{k} z_i + k.\hat{d} + m.\sum_{i=1}^{k}\varepsilon(s_i)\right) - M\right]^2$.

Ce qui donne : $\sigma^2(S^*;\varepsilon) = Min\left\{\frac{1}{k^2}\left[(Cst)^2 + m^2.\sum_{i=1}^{k}\sum_{i=1}^{k} Cov(\varepsilon(s_i),\varepsilon(s_j))\right]\right\}$.....(I)

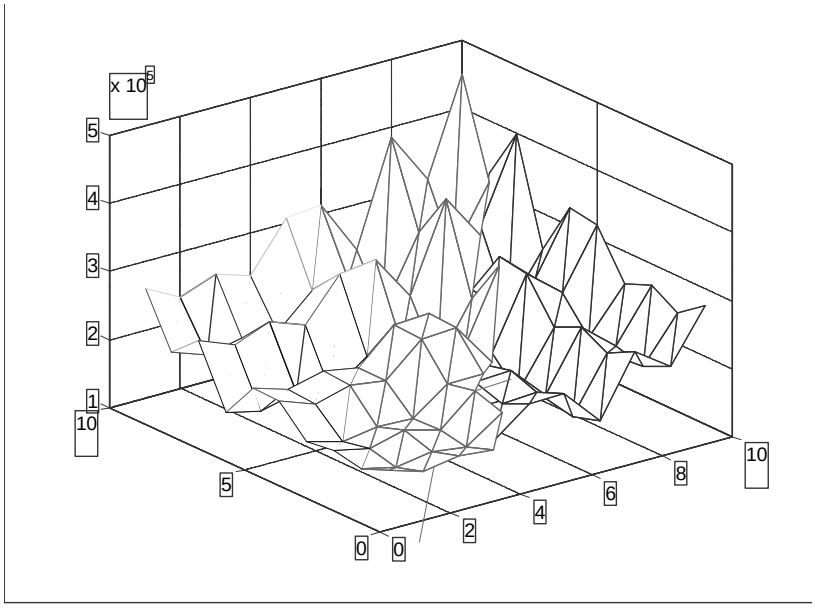
Tel que : $\{s_i \in S : i = 1,...k\}$ *et* $\{s_j \in S : j = 1,...k\}$

$$Cst = \hat{a}\sum_{i=1}^{k} x_i + \hat{b}\sum_{i=1}^{k} y_i + \hat{c}\sum_{i=1}^{k} z_i + k.\hat{d} - k.M$$

IV.4- Résolution du problème de minimisation de (I) :

Le problème *d'échantillonnage spatial optimal* sur $D \subset IR^3$, en minimisant (I) est un problème d'optimisation sur un domaine continu et borné . Il est connu que les algorithmes génétiques dans leur version codés réels représentent une méthode efficace dans ce cas [16].

IV.4.1- Résolution du problème d'optimisation :



De plus, la forme de la fonction objective (I), permet de favoriser les algorithmes génétiques par rapport aux autres méthodes d'optimisation, car on remarque que le graphe de la fonction objectif conserve une certaine structure, que nous pouvons appeler " Structure Combinative".

Une structure combinative est caractérisée par le fait que les minima locaux (ou maxima) ne sont pas disposés n'importe comment, mais qu'au contraire ils contiennent une information sur la position des autres optimum et en particulier sur l'optimum global. (condition de reconstructibilité par recombinaison).(voir [16] page 80).

Ce genre de problème est traité comme des problèmes discrets avec une perte de précision inévitable, c'est pour ça qu'il faut veiller à une discrétisation correcte de l'espace de recherche. De plus, il faut prendre soin de trouver une représentation qui permette la recherche, l'isolation (implicite) et la recombinaison des blocs de construction.
(Un bon codage et de bons opérateurs permettent d'améliorer une solution).

Le codage des individus se fait comme suit :

Soit n : Le nombre d'individus candidats (échantillons spatiaux de taille k).

INCORPORER Equation.3 Le INCORPORER Equation.3 individu i=1,…,n..

Tel que: INCORPORER Equation.3

INCORPORER Equation.3 est représenté par un vecteur de réels de taille INCORPORER Equation.3 :

INCORPORER Equation.3 INCORPORER Equation.3

$= (s_1; s_2; .....; s_k)^{(i)}$. *i=1,2,…,n.*

Le codage qu'on vient de décrire permet d'effectuer les deux types de croisement définis au chapitre (II-5.4), i.e. *le croisement global et le croisement local*. Le premier permet d'interchanger des points de IR^3 entre deux individus, par contre le deuxième permet un déplacement adéquat d'un sommet choisit aléatoirement dans la région *D*.

Soit $X^{(i)}$ *et* $X^{(j)}$ deux individus parents.

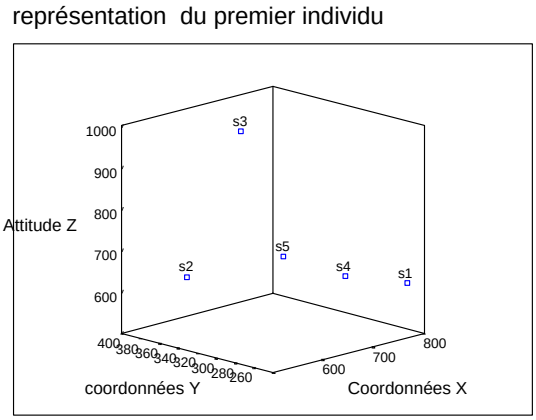
$X^{(i)} = (x_1, y_1, z_1, x_2, y_2, z_2, x_3, y_3, z_3, ..., x_k, y_k, z_k)^{(i)}$ qu'on notera : $(s_1, s_2, s_3, ..., s_k)$

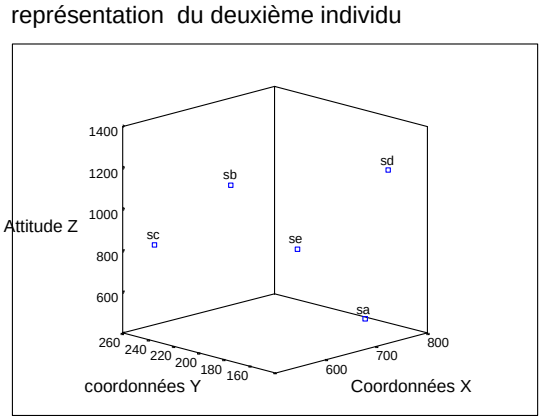
$X^{(j)} = (x_1, y_1, z_1, x_2, y_2, z_2, x_3, y_3, z_3, ..., x_k, y_k, z_k)^{(j)}$ qu'on notera : $(s_a, s_b, s_c, ...$) .

Un croisement global entre les individus parents $X^{(i)}$ *et* $X^{(j)}$ aux sites 1, 3, 6, 8

permet d'interchanger les points $s_1 = (x_1, y_1, z_1)$ et s_a , ainsi que les points $s_3 = (x_1, x_2, x_3)$ *et* s_c .

Ceci peut être schématisé comme suit pour *k=5*:





Au cours de notre application, on a travaillé avec les paramètres suivants :

- M : La quantité moyenne de pluie ciblée : 337.9.
- k : le nombre de sites spatiaux (stations pluviométriques) : 5.
- n : le nombre d'individus (*Chromosomes*) : 10.
- Sélection par rangement (*Pression de sélection* : 1.4).
- Croisements : Les deux types de croisements sus- cités

Avec $P_{CG} = 0.7,$ $P_{CL} = 0.05.$ (P_{CG} : Probabilité de croisement global)

(P_{CL} : Probabilité de croisement local).

- P_m =0.002 (probabilité de mutation)
- Les erreurs sur les sites sont supposées spatiallement corrélées par une fonction de covariogramme gaussiènne donnée par :

$$C(S_i, S_j) = \exp(-\theta.\|S_i - S_j\|^2) \qquad \theta = 0.05.$$

- ***Stratégie***: Remplacement total des parents par les fils d'une génération à une autre.

- ***Critère d'arrêt*** : Après plusieurs expérimentations, on a choisi d'arrêter l'algorithme après 200 itérations.

On note que la fixation des probabilités de croisement local et global est purement expérimentale. Les probabilités proposées sont obtenus après plusieurs simulations, afin d'avoir de bons résultats.

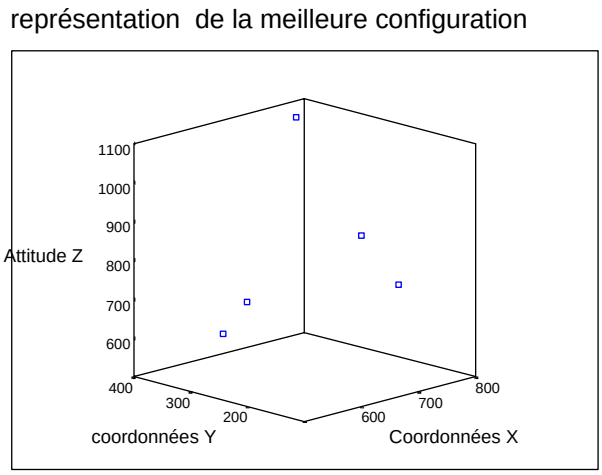
On fait remarquer aussi que la population initiale, i.e. la solution initiale à été prise aléatoirement dans le domaine fermé borné *D*.

L'application de l'algorithme génétique spatial précédent donne de résultats très encourageant.

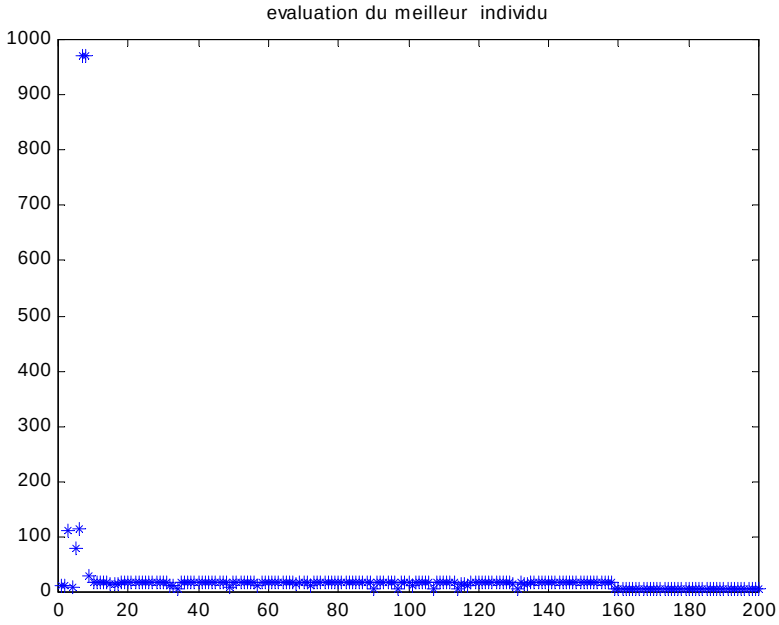
On trouve que la quantité de pluie observée sur un réseau de 5 stations pluviométriques est égale à : 337.56, avec une erreur d'observation égale à 0.341, ce qui est excellent.

Les coordonnées des sites optimaux obtenus à partir de l'application de notre *AGS* sont données par la matrice X : $X = \begin{pmatrix} 769.3 & 205.2 & 709.9 \\ 593.9 & 242.3 & 737.9 \\ 641.9 & 386.2 & 562.7 \\ 771.3 & 387.3 & 1069.3 \\ 703.4 & 205.1 & 860.9 \end{pmatrix}$

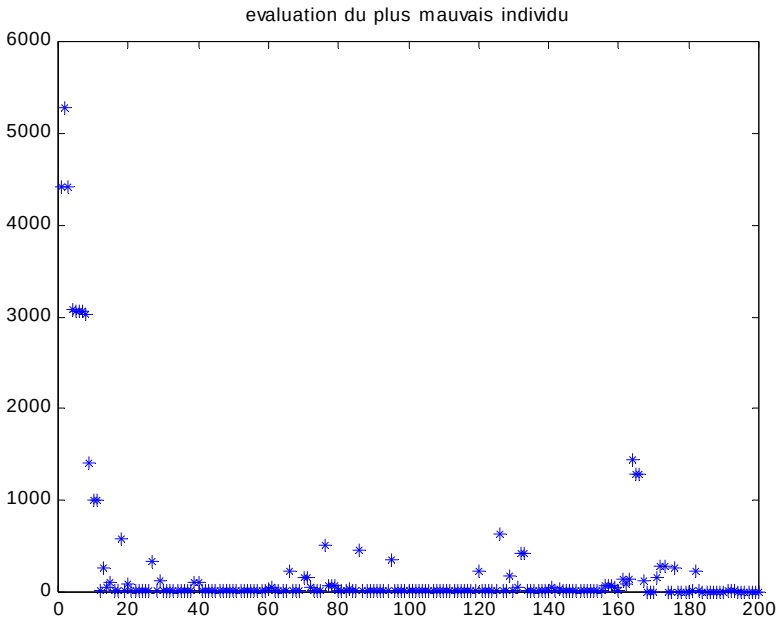
On précise que les coordonnées du *jème* site spatiale du ième individu, sont données par la *jème* ligne de la matrice $X^{(i)}$, *(j=1,…,k).* Ce qui peut être représenté comme suit :



On remarque que l'*AGS* utilisé améliore la qualité du résultat obtenu dans [19]. Cela peut être expliqué par l'introduction d'un nouveau codage des individus tout a fait adéquat au problème d'optimisation spatiale sur un domaine continu et borné; de plus, au deux types de croisement définis (Croisement global et Croisement Local) qui permettent une exploration rapide de l'espace dans le sens défini.



Le graphe ci-dessus montre que la qualité du meilleur individu s'améliore d'une génération à une autre, jusqu'à convergence, où la population est formée d'un seul individu.



Nombre d'itérations

Le graphe d'évaluation du plus mauvais individu, i.e. celui qui à la plus grande évaluation parmi les individus d'une génération donnée (problème de minimisation), montre que la qualité de ce dernier s'améliore d'une itération à une autre ; ce qui revient à dire que la qualité de la population s'améliore toute entière.

L'erreur absolue obtenue ainsi que les graphiques d'évolution du meilleur et du plus mauvais individu montrent que l'algorithme spatial a donné de très bons résultats; seulement on note que ces résultats ont été obtenus après plusieurs simulations. D'où la nécessité de chercher des paramètres optimaux qui donnent de bons résultats avec une bonne précision.

La recherche des paramètres optimaux P_{CG} *et* P_{CL}, sont résumés dans le tableau suivant : (Chaque simulation avec des paramètres fixés est répétée 15 fois).

P_{CG}	P_{CL}	**Erreur absolue** **Err=**$\lvert \overline{Q} - M \rvert$	**Moyenne et écart type**	**I.D.C (Err) (5 %)**
0.9	0.01	4.43 0.153 4.22 1.55 0.32 0.27 7.06 10.34 11.18 0.26 4.43 0.153 4.22 1.55 0.32	$\bar{x} = 3.38 \quad \sigma = 3.75$	[- 3.39, 11.37]
0.9	0.05	6.39 8.54 4.09 0.041 1.48 8.16 2.64 6.29 0.52 10.48 1.66 1.04 0.68 6.015 7.55	$\bar{x} = 4.36 \quad \sigma = 3.037$	[-1.59, 10.31]
0.9	0.1	4.56 11.66 1.26 1.072 4.41 1.38 19.47 3.04 4.36 0.35 13.44 3.59 2.41 1.80 7.87	$\bar{x} = 5.37 \quad \sigma = 5.28$	[4.97, 15.71]
0.9	0.2	1.89 44.32 7.63 7.61 13.33 1.85 21.87 1.90 9.00 4.36 3.46 15.12 3.77 14.76 8.02	$\bar{x} = 9.94 \quad \sigma = 8.7$	[7.11, 26.99]
P_{CG}	P_{CL}	**Erreur absolue**	**Moyenne et**	**I.D.C (Err)**

		Err=| $\overline{Q}$ -M |	**écart type**	**(5 %)**
0.9	0.3	11.54 21.09 0.48 5.96 19.20 22.67 1.32 12.31 7.37 4.21 5.95 23.76 2.65 11.95 5.23	$\bar{x}=10.41 \quad \sigma=7.71$	[4.70, 25.52]
0.9	0.4	8.03 0.257 4.09 12.13 3.99 5.64 7.00 1.43 5.33 11.69 2.29 7.40 19.31 5.74 4.51	$\bar{x}=6.58 \quad \sigma=4.67.$	[2.57, 15.73]
0.9	0.5	22.46 14.67 19.98 9.60 6.29 26.13 3.89 4.76 0.85 11.91 13.66 2.35 39.20 38.01 4.60	$\bar{x}=14.55 \quad \sigma=11.9$	[8.77, 37.87]
0.8	0.01	7.15 3.54 1.36 5.34 11.33 12.82 8.37 5.33 17.21 3.71 9.96 7.85 8.61 2.76 1.047	$\bar{x}=7.72 \quad \sigma=4.10$	[0.315, 15.75]
0.8	0.05	35.12 19.9 5.65 6.43 13.25 1.256 2.75 13.89 4.07 12.59 12.42 9.73 0.401 0.143 3.98	$\bar{x}=9.43 \quad \sigma=8.88$	[7.97, 26.83]
0.8	0.1	6.71 0.89 7.34 15.81 9.23 0.256 15.79 11.52 8.25 1.024 0.8415 8.7 3.91 2.61 .202 7	$\bar{x}=6.80 \quad \sigma=5.09$	[3.17, 16.77]
0.8	0.2	3.58 32.14 23.28 22.29 20.27 11.99 22 14.11 12.47 3.68 25.67 0.15 1.08 3.42 8.00	$\bar{x}=13.6 \quad \sigma=9.82$	[5.64, 32.84]
0.8	0.3	22.39 0.04 2.45 13.88 1.225 1.25 0.527 30.28 7.01 1.77 5.64 35.55 6.71 1.99 16.49 10.58	$\bar{x}=9.83 \quad \sigma=11.07$	[11.86, 31.52]
0.8	0.4	22.11 23.50 1.75 7.76 3.83 3.6 7.05 10.37 6.47 11.87 14.01 5.24 12.25 16.2	$\bar{x}=10.54 \quad \sigma=6.29$	[1.78, 22.86]
0.8	0.5	0.148 3.48 26.63 12.69 10.07 4.78 4.35 10.07 10.88 4.17 4.75 13.58 10.62 1.76 3.69	$\bar{x}=8.15 \quad \sigma=6.4$	[4.39, 20.69]
0.7	0.01	0.09 7.20 5.59 1.168 6.385 6.70 14.90 2.76 6.91 10.82 28.08 0.32 4.59 28.60 13.92	$\bar{x}=9.16 \quad \sigma=8.65$	[7.79, 26.11]
0.7	0.05	2.85 1.50 6.76 4.20 3.41 1.42 0.341 3.16 4.83 2.66 3.16 13.76 9.89 3.36 9.24	$\bar{x}=4.69 \quad \sigma=3.58$	[-2.32, 11.70]
0.7	0.1	2.03 17.54 5.86 8.49 5.35 5.10 3.61 12.09 4.04 6.82 0.38 1.31 9.09 12.74 7.05	$\bar{x}=6.76 \quad \sigma=4.5$	[2.06, 15.58]
0.7	0.2	9.21 1.87 14.29 11.03 3.98 6.16 21.94 11.15 4.34 1.04 12.83 1.82 17.59 1.74 24.31	$\bar{x}=9.55 \quad \sigma=7.28$	[4.71, 23.81]
0.7	0.3	4.561 15.48 27.44 17.17 21.45 11.87 23.51 3.94 1.49 3.60 18.30 , 3.23, 24.99 12.28 1.823	$\bar{x}=12.741 \quad \sigma=8.8$	[4.64, 30.12]
0.7	0.4	14.77 17.68 13.245 37.65 10.04 3.74 12.41 8.87 11.17 6.53 0.269 5.62 12.17 26.84 4.002	$\bar{x}=12.32 \quad \sigma=9.21$	[5.73, 30.37]
0.7	0.5	31.25 29.425 12.59 14.414 26.545 2.68 11.25 1.25 3.034 11.25 6.45 2.64 9.40 21.88 12.33	$\bar{x}=13.06 \quad \sigma=9.64$	[5.83, 31.95]
P_{CG}	P_{CL}	**Erreur absolue**	**Moyenne et**	**I.D.C (Err)**

		Err=$\lvert \overline{Q} - M \rvert$	**écart type**	**(5 %)**
0.6	0.01	8.43 2.70 1.28 4.39 3.87 8.43 2.70 1.28 4.39 3.78 3.28 5.50 7.91 0.56 1.69	$\overline{x} = 4.01 \quad \sigma = 2.49$	[0.87, 8.89]
0.6	0.05	5.14 13.71 0.132 2.76 4.48 0.77 3.39 4.67 0.48 3.66 5.69 3.40 5.17 1.71 8.10	$\overline{x} = 4.21 \quad \sigma = 3.28$	[2.21, 10.63]
0.6	0.1	22.52 6.99 8.85 8.04 11 0.23 5.57 0.5 12.69 7.89 0.86 15.04 3.47 7.53 5.71	$\overline{x} = 7.92 \quad \sigma = 5.69$	[3.23, 19.07]
0.6	0.2	9.40 19.61 12.41 0.45 1.028 1.75 7.71 16.88 0.01 27.14 13.40 31.40 3.55 13.22 20.45	$\overline{x} = 11.91 \quad \sigma = 9.55$	[6.80, 30.62]
0.6	0.3	12.71 1.45 4.23 4.16 2.64 4.67 7.59 1.035 44.66 53.48 7.24 0.68 26.14 4.76 2.48	$\overline{x} = 11.86 \quad \sigma = 15.9$	[19.30, 43.02]
0.6	0.4	0.66 11.01 2.11 10.20 6.76 0.41 0.039 1.40 0.62 15.85 8.26 1.29 11.52 1.93 3.22	$\overline{x} = 5.01 \quad \sigma = 4.95$	[4.69, 14.17]
0.6	0.5	3.57 11.87 13.81 17.81 5.69 23.09 40.93 11.42 0.36 35.048 7.59 17.64 9.89 28.21 35.79	$\overline{x} = 17.51 \quad \sigma = 12.1$	[6.22, 41.24]

Le tableau précédent montre que *l'A.G.S* proposé donne de bons résultats pour $PcG = 0.9 \quad P_{cL} = 0.01$. On suggère ces paramètres aux utilisateurs de l'algorithme génétique spatial pour la résolution d'un problème d'échantillonnage spatial optimal.

L'erreur absolue $\lvert \overline{Q} - M \rvert$ représente la différence entre la valeur moyenne ciblée par le réseau pluviométrique et la valeur fournie par les sites optimaux fournis par *l'A.G.S* après convergence.

IV.4.2- Résolution du Problème Combinatoire :

L'algorithme génétique spatial avec les opérateurs proposés suppose que l'évaluation de (I) est connu en chaque site du domaine *D (* Ce qui est vérifié pour l'application de détermination des sites pluviométriques). En g
?? ??

?? ??

onnage représentent les centres des voronoi.

Pour notre application, $D \subset IR^3$, le modèle utilisé pour le partitionnement du domaine d'échantillonnage est le modèle cubique, c'est à dire les points de l'échantillon vont être pris parmi ceux qui sont centre des cubes. Autrement dit, que le partitionnement du domaine d'échantillonnage va convertir le problème, d'un problème d'optimisation stochastique en un problème combinatoire très compliqué.

Si on suppose que le pas de la discrétisation régulière sur le domaine *D* est (*pas =10 Km*) pour les directions (*X,Y*) et (*pas =10 m*) pour la direction *Z* (*Attitude*).

Le problème est de déterminer *k* points parmi $(24 \times 25 \times 79 = 47400)$ qui minimisent (I).

Or on a C_{47400}^{k} échantillons candidats possibles.

Pour *k =5*, on aura : 1.99×10^{21} échantillons possibles (Ce qui est énorme).

Une recherche exhaustive peut mener à des journées de calculs. Ce genre de problèmes représente le cadre idéal d'évolution des Algorithmes génétiques. Cela va être démontré par l'application qui va suivre :

Au cours de cette application, on a utilisé les mêmes paramètres utilisés lors de la première application.

On note que dans ce cas, les points (*gènes*) des individus (*Chromosomes*) initiaux ont été pris aléatoirement parmi ceux qui sont candidats à l'échantillonnage. De plus les opérateurs génétiques doivent mener à des points parmi ceux qui sont candidats à l'échantillonnage.

Pour cela, on doit adapter l'opérateur de *croisement Local* au cas considéré.
On propose l'adaptation suivante :

Si on considère que le *croisement Local* se fait entre deux individus X_1 *et* X_2 .

On choisi un ensemble de couples de composantes (x_i^1, x_i^2) du couple d'individus (X_1, X_2). La première composante est remplacée par $x_i^{1'}$ parmi les points compris entre *[a,b]*

$$\text{Tel que :} \quad \begin{cases} x_i^{1'} = P_E\left(\dfrac{(b-a)}{pas} \times U\right) \times pas + pas/2; \\ x_i^{2'} = x_i^1 + x_i^2 - x_i^{1'} \end{cases}$$

$$\text{Avec :} \quad \begin{aligned} a &= \max(x_i^1, x_i^2) - \min(x_i^1, x_i^2) \\ b &= \min(x_i^1 + x_i^2, Sup) \end{aligned}$$

P_E : Partie entière.

U : Entier appartenant à *[0,1]*.

Lors de la recherche de l'échantillon optimal qui minimise (I), l'échantillon peut contenir deux points identiques – car le problème est un problème combinatoire - . Pour cela, afin de garder la diversité de la population, on introduit un *opérateur de diversité* qui est défini comme suit :

Cela est réalisé en augmentant le gène ou en le diminuant comme suit :

$$\begin{cases} x_i' = x_i \quad \text{Sgn} \quad (pas) \\ y_i' = y_i \quad \text{Sgn} \quad (pas) \\ z_i' = z_i \quad \text{Sgn} \quad (pas) \end{cases}$$

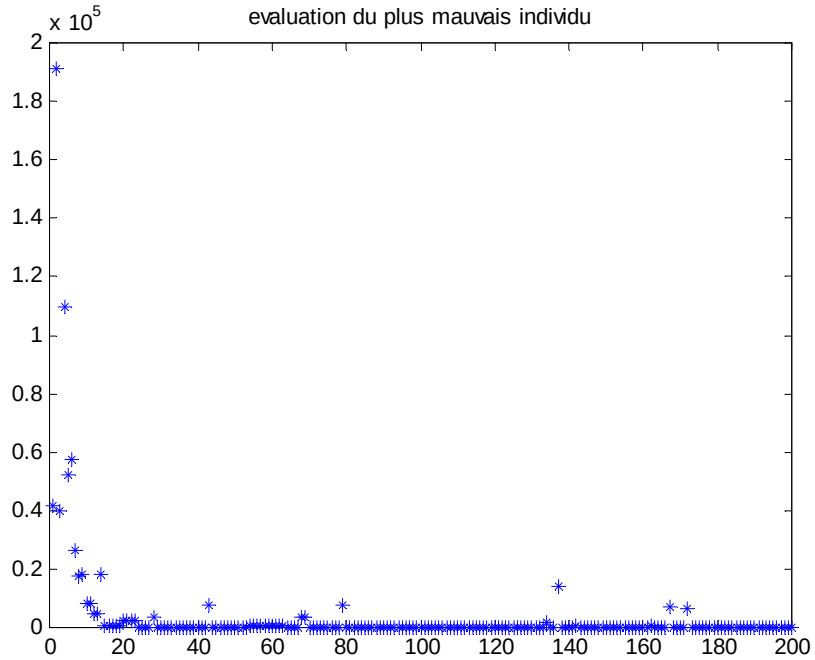
- *pas* : pas de la discrétisation.
- *Sgn* : signe , peut prendre (+, -).

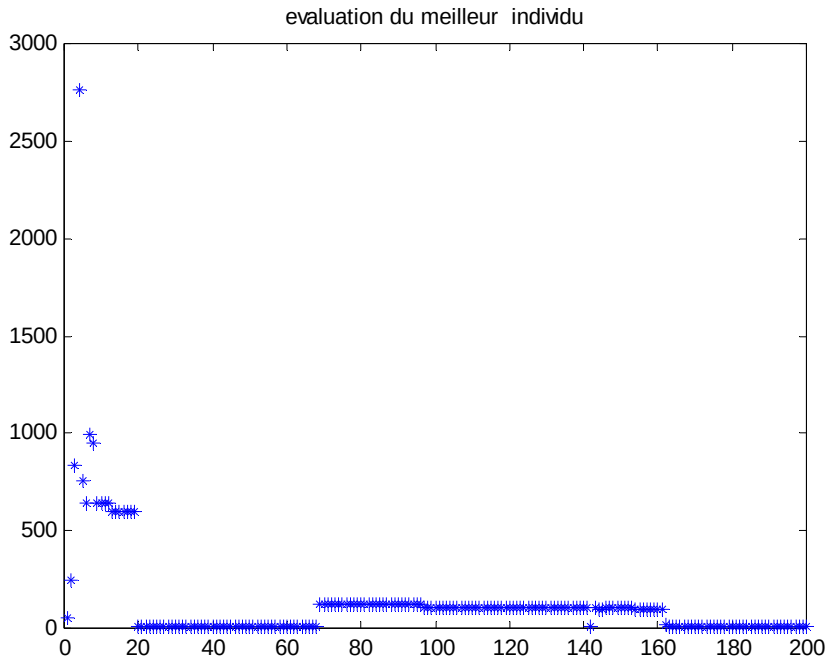
Sgn : est défini comme suit :

$$\text{Sgn} = \begin{cases} + & si \quad (x_i = 545, \quad ou \quad y_i = 155 \quad ou \quad z_i = 455) \\ - & si \quad (x_i = 795, \quad ou \quad y_i = 395 \quad ou \quad z_i = 1245) \\ (+,-) & avec \quad une \quad probabilité \quad \frac{1}{2}. \end{cases}$$

Ce qui permet d'éviter le problème des frontières qui peut se poser.

Les graphes de variation du meilleur individu et celui du plus mauvais individu montrent que la solution s'améliore d'une itération à une autre, de manière très impressionnante lors des premières itérations avant de se stabiliser lors des dernières itérations. Ce qui montre que l'algorithme génétique spatial converge à partir d'une très mauvaise solution.

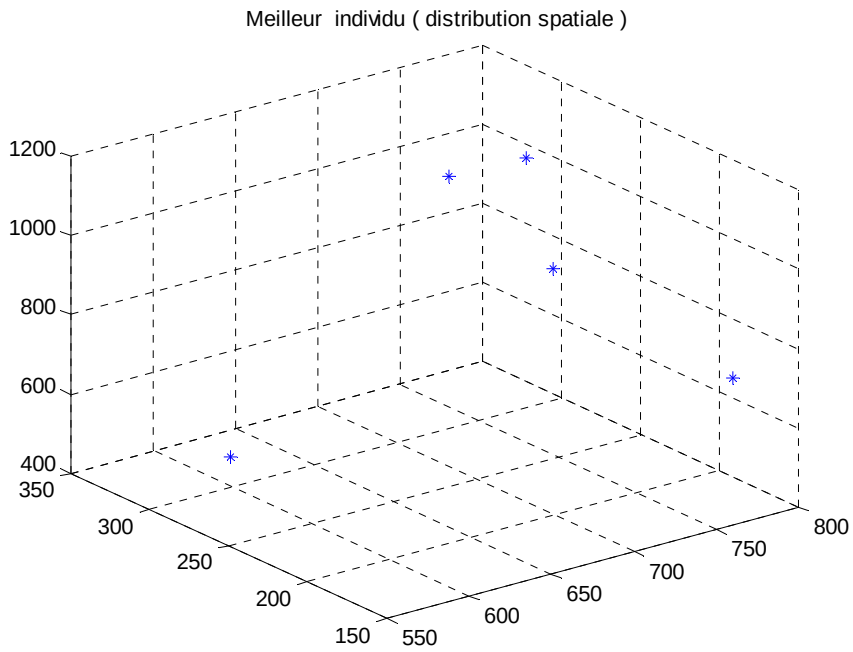




Nombre d'itérations

Après 200 itérations, on remarque que la convergence est atteinte, et les individus de la population deviennent identiques.

La configuration spatiale optimale est donnée par la figure suivante :



La solution optimale est donnée par : $XG = \begin{bmatrix} 685 & 185 & 1065 \\ 775 & 345 & 905 \\ 565 & 265 & 580 \\ 755 & 275 & 1105 \\ 765 & 155 & 755 \end{bmatrix}$

ERR=0.62.

Les résultats de notre *A.G.S* pour les paramètres optimaux, en supposant que les points candidats à l'échantillonnage font partie d'un réseau régulier (centres de cubes) ; sont représentés dans le tableau suivant :

L'exécution du programme s'est faite 80 fois afin d'avoir un intervalle de confiance.

P_{CG}	P_{CL}	**Erreur absolue** $\lvert \overline{Q} - M \rvert$	**Moyenne et écart type**
0.9	0.01	4.43 3.98 0.52 1.81 2.02 1.4 0.31 1.26 0.76 0.11 5.10 2.41 3.46 9.07 1.81 0.65 1.34 0.41 2.19 0.00 0.94 0.75 5.17 2.80 7.14 2.79 0.19 1.78 0.03 0.19 8.28 0.63 1.44 7.59 8.84 1.42 2.14 0.58 4.09 2.46 7.54 1.42 4.62 2.02 4.14 0.39 11.85 0.11 0.26 1.11 0.56 1.81 2.39 0.84 2.01 0.01 0.98 1.58 7.71 3.28 1.07 2.16 0.07 0.21 0.95 4.56 0.92 0.03 0.43 1.22 1.46 13.74 7.81 1.27 5.51 1.64 0.90 5.31 1.18 0.99	$\overline{x} = 2.97 \quad \sigma = 3.57$ **I.D.C**= [4.02, 9.96]

Conclusions :

Même si l'utilisation de *L'A.G.S* sans contraintes géométriques donne de bons résultats, son utilisation n'est pas toujours possible. La discrétisation de l'espace en un réseau régulier de points semble plus pratique (car on a pas toujours la valeur de la fonction d'évaluation en chaque point de l'espace), De plus ça donne, en général, de meilleurs résultats. Cela peut s'expliquer intuitivement par une meilleure exploration de l'espace. (dans un cube donné, on ne peut pas avoir plus d'un point, et par conséquent on a plus de chance de parcourir l'espace le plus rapidement possible).

On remarque que l'algorithme génétique spatial donne de bons résultats pour les paramètres suivants:

- *M* : La quantité de pluie ciblée : 337.9.
- *k* : le nombre de sites spatiaux (stations pluviométriques) : 5.
- *n* : le nombre d'individus (*Chromosomes*) : 10.
- Sélection par rangement (*Pression de sélection* : 1.4).
- Croisements : Les deux types de croisements sus- cités

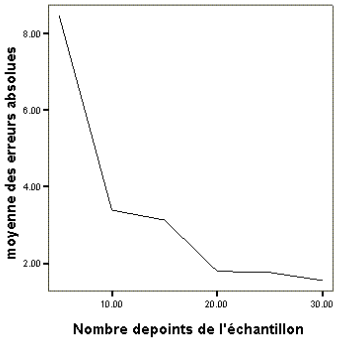
Avec $P_{CG} = 0.9, \quad P_{CL} = 0.01$ (P_{CG} : Probabilité de croisement global)

(: Probabilité de croisement local).

- [ ] =0.002 (probabilité de mutation

Les bons résultats nous ont incités à se demander si l'on peut encore améliorer la qualité de notre algorithme en augmentant le nombre de points de l'échantillon, i.e. augmenter sa précision.

Le graphe suivant résume la moyenne ainsi que l'écart type obtenu pour différents nombre de points d'échantillons pris afin de résoudre notre problème combinatoire.

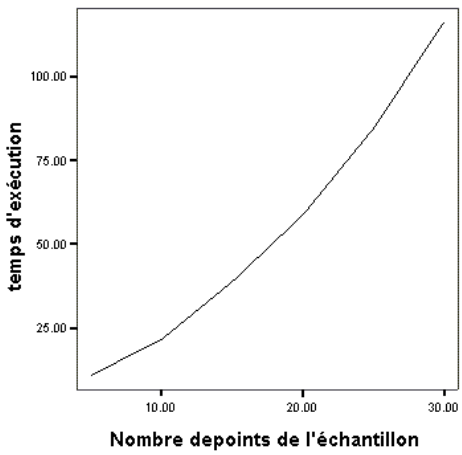


Dot/Lines show Means

Les graphe des moyennes ainsi que celui des écarts type des erreurs absolues en fonction du nombre de points utilisées lors de l'échantillonnage spatial (*k*) montrent clairement que les performances de l'*A.G.S* évoluent en fonction de (*k*). (La précision de l'algorithme augmente si le nombre de sites pluviométriques augmente).

Seulement, il faut prendre en compte la durée d'exécution de l'algorithme en fonction du nombre de gènes (sites spatiaux).i.e qu'on est parfois obligé de faire un compromis entre la précision de notre algorithme et le temps d'exécution de ce dernier.

Evidemment, on s'attendrait à ce que l'on trouve que la durée d'exécution évolue en fonction du nombre de points échantillonnés, ce qui est confirmé expérimentalement.



Dot/Lines show Means

IV.4.3- Utilisation d'un Opérateur de Quantification :

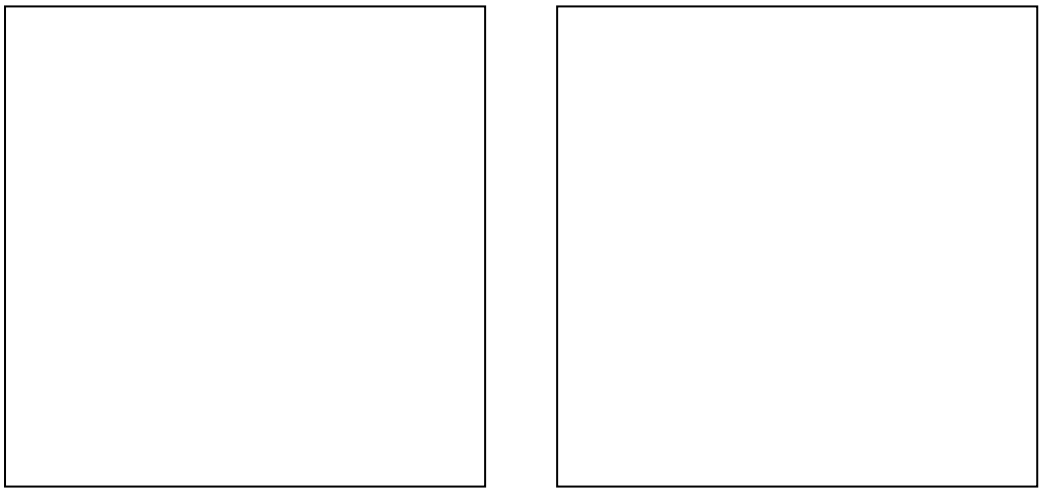
IV.4.3.1 – Introduction :

On a déjà montré que les algorithmes génétiques spatiaux (***A.G.S***) prennent du temps pour converger lorsque la courbe (le relief) qui représente la fonction objectif est lisse, et donne de bons résultats en un temps record, lorsque la courbe contient des pics, surtout si ces derniers sont distribués de manière plus au moins régulière.

Une idée est d'utiliser la quantification afin de créer des perturbations qui seront bénéfiques dans notre cas, i.e. l'optimisation d'une fonction objectif sur un domaine continu et borné, car ça permet aux opérateurs génétiques (notamment le croisement) d'être plus rapide.

Vu la spécificité du problème, la quantification va être appliquée aux erreurs d'observations aux sites spatiallement corrélés (car la variabilité spatiale réside dans les erreurs affectés aux sites d'observations).

On rappelle que les erreurs ont été simulé suivant une gaussienne *N(0,1)*, en utilisant la méthode de triangularisation (voir chapitre (I.16.4)).



Avant quantification

Après quantification

IV.4.3.2- Adaptation de la Quantification à notre Problème :

L'application de la quantification aux erreurs d'observation spatiallement corrélés $\varepsilon(S_i)$,

Cela nous a amené a utilisé une quantification qui se fait à l'aide de réseaux réguliers de points : (*QV en treillis*) qui sont des algorithmes de quantification rapides.

- $Z^k (k \geq 2)$: les réseaux cubiques sont ceux utilisés dans notre application.

Quantifier une erreur d'observation $\varepsilon(s_i)$ sur un site S_i revient à lui associer le vecteur de reproduction le plus proche $H(\varepsilon(s_i)) = \hat{\varepsilon}(s_i) \in IR^3$. Cela revient à effectuer une quantification scalaire uniforme de pas uniforme sur chaque coordonnée de $\varepsilon(s_i), i = 1,...k$.

En utilisant le modèle mathématique (*), la fonction objectif sera de la forme :

$$\sigma^2(S^*;\varepsilon) = Min\left\{\frac{1}{k^2}\left[(Cst)^2 + m^2.\sum_{i=1}^{k}\sum_{i=1}^{k} Cov\left(\hat{\varepsilon}(s_i),\hat{\varepsilon}(s_j)\right)\right]\right\}.....(II)$$

La quantification vectorielle va être utilisée à chaque itération de notre (*A.G.S*), ce qui revient à considérer la quantification comme un opérateur génétique qui est introduit essentiellement afin d'accélérer la vitesse de convergence de *l'A.G.S* proposé.

L'introduction d'un nouveau opérateur (Quantification) engendre automatiquement un temps d'exécution supplémentaire de notre *A.G.S* à chaque itération, ce dernier est négligeable devant la vitesse de notre algorithme en quantifiant les erreurs.

Autrement dit, l'utilisation de *l'A.G.S* sans quantification des erreurs nécessite un nombre d'itérations supérieur au cas où l'on utilise la quantification, ce qui permet de récupérer le temps supplémentaire de l'exécution lors de l'opération de quantification des erreurs (En diminuant le nombre d'itérations nécessaires pour la convergence de l'algorithme).

IV.4.3.3- Résultats et Commentaires :

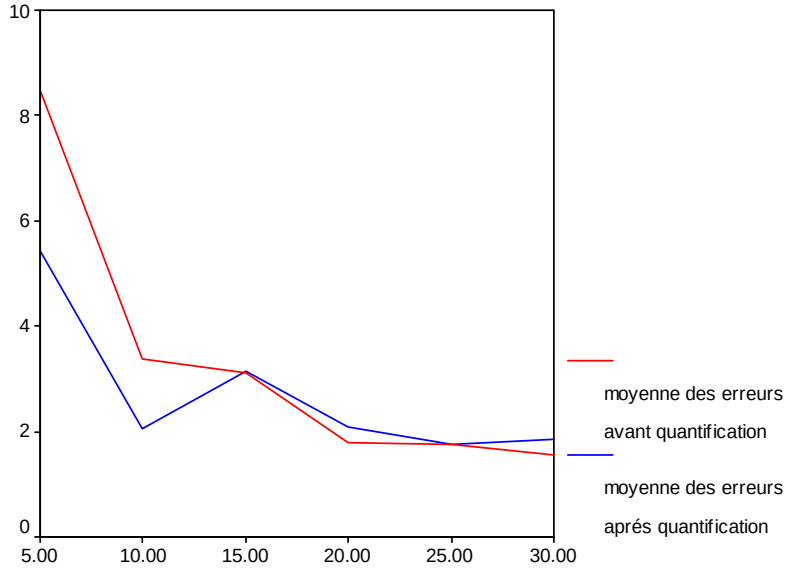
L'application de l'algorithme génétique spatial en quantifiant les erreurs d'observations, toujours en utilisant les mêmes paramètres que dans les applications précédentes donne les résultats suivants: *(k=5*)

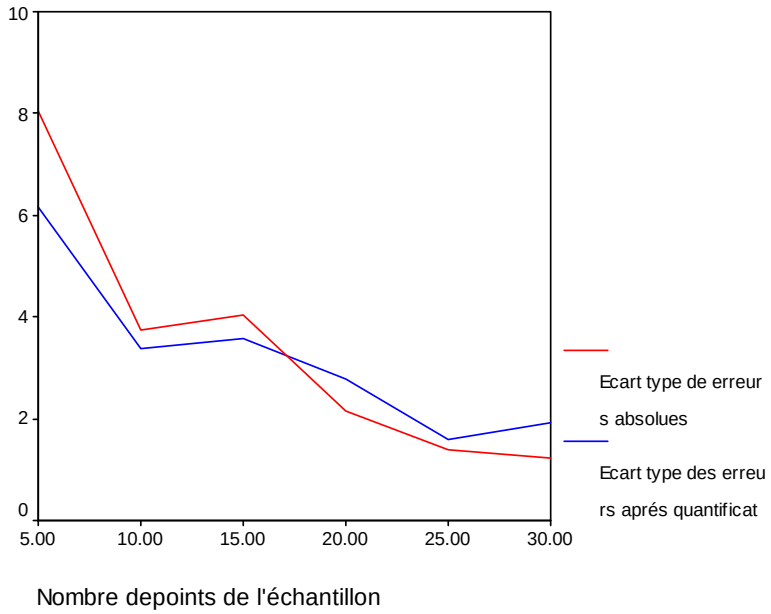
P_{CG}	P_{CL}	**Erreur absolue** $\lvert\overline{Q} - M\rvert$	**Moyenne et écart type**
0.9	0.01	0.35 2.093 1.31 0.22 2.57 0.7 0.56 0.26 0.43 0.61 0.56 2.38 5.58 0.93 5.53 2.70 0.32 0.80 0.441 0.47 6.63 1.95 1.03 1.17 1.67	$\bar{x} = 1.68 \quad \sigma = 2.10$ **I.D.C**= [2.43, 5.79]

On remarque que la quantification (l'opérateur de quantification) améliore nettement les résultats obtenus auparavant.

Comme on la fait pour dans le cas où l'on a pas d'opérateur de quantification, on se demanderait si les performance de notre algorithme génétique sont liés au nombre de points échantillonnés (nombre de sites pluviométriques choisis en avance).

Les résultats obtenus sont résumés dans les graphiques suivants :





On remarque que la quantification des erreurs d'observations spatiallement corrélées améliore nettement la qualité des résultats obtenus pour un nombre de points échantillonnées réduit. Ce résultat confirme le fait que la quantification permet de créer des perturbations au niveau du graphe de la fonction objective qui ont un effet positif sur la vitesse de convergence ainsi sur la qualité des résultats obtenus.

Seulement la quantification peut donner de mauvais résultats lorsque le nombre de sites spatiaux: $k \to \infty$. Intuitivement, on peut expliquer ce résultat par le fait que l'Algorithme Génétique avec les opérateurs déjà définis sans quantification des erreurs donne de très bons

résultats pour $k \to \infty$. Or dans ce cas l'ajout d'un quelconque opérateur génétique (y compris celui de la quantification) ne fera que dégrader la nature des résultats obtenus.

IV.5- Implémentation :

L'algorithme génétique proposé à été implémenté en utilisant le logiciel de programmation 'MATLAB' qui offre d'immenses avantages, notamment, lorsqu'il s'agit de faire des simulations et lorsqu'on travaille avec des entités matricielles.

Conclusion Générale

Le cas de l'échantillonnage spatial dans un domaine compact avec des erreurs corrélées, est considéré. Afin d'améliorer les études précédentes et de montrer que les algorithmes génétiques se présentent efficaces pour le problème, nous avons développé des opérateurs génétiques, appelés opérateur de croisement local et global ainsi qu'un opérateur de mutation, tous adaptés au cas de la conception d'un échantillonnage optimal spatial. Ce qui nous donne un algorithme génétique nouveau, permettant une accélération de la recherche spatiale, une meilleure exploration de l'espace de recherche et d'éviter une convergence prématurée. De plus, nous confirmons l'effet de la quantification sur l'échantillonnage dans le cas spatial et nous remarquons que ça a un effet positif lorsque la taille de l'échantillon est réduite, et un effet négatif sinon.

En perspective, nous envisageons encore une autre amélioration de notre algorithme et de faire une extension au cas de processus spatio-temporelles.

Références bibliographique :

[1] George Christakos. Random Field Models in Earth sciences.
Copyright © 1992 by Academic Press, *INC.*

[2] Cressie. N 1991, Statistics for Spatial data, J.Wiley, New York.

[3] Denis D.Cox Lawrence, H.Ccx, and Katherine B. Ensor
Technical Report 38, October 1995

[4] R.Dipley, Statistical inference for spatial data.

[5] Alfred Stein, Christien Ettema. An overview of spatial sampligprocedures and experimental design of spatial studies for ecosystem
comparisons.
Agriculture, Ecosystems and Environment 94(2003) 31-47

[6] Karim Benhenni and Stamatis Cambanis. The effect of quantization on the performance of sampling design.
IEEE transactions on information theory Vol 44, N°5, Septembre 1998.

[7] Bayesien analysis of regression models with spatially correlated errors and missing observations.
Man-SukOh, Dong Wan Shin, Han joon Kim.

[8] M.C.Bueso and J.M. Angulo. Spatial Sampling Design Based on Stochastique Complexity Journal of Multivariate Analysis 71,94-110 (1999)

[9] Sabyasachi Basu and Gregory C.Reinsel .
Regression Models with Spatially Correlated Errors. Journal of the American Statistical Association March 1994, Vol89, N°425, Theory and methods.

[10] J.H.Holland. Adaptation in Natural and Artificial Systems. PHD thesis, University of Michigan, 1975.

[11] D. Goldberg. Genetic Algorithms in Search, Optimization and Machine Learning . Addison- Wesley, 1989.

[12] Y. Davidor. Genetic algorithms and robotics. An heuristic strategy for optimization Word Scientific:1,1990.

[13]: R. Guenter. Convergence Analysis of Canonical Genetic Algorithms. IEEE Transactions on Neural Network, 5(1):96-101, January 1994.

[14] T.E .Davis and J.C. Principe. A Simulated Annealling like convergence theory for simple Genetic Algorithm Proceeding of the Fourth International Conference on Genetic Algorithm. 174 (182, July 1991).

[15] J.B. Kruskal and M. Wish. Multidimensional Scaling. Sage Publication, 1984.

[16] Jean- Michel Renders. Algorithmes Génétique et Réseaux de Neurones ; Paris 1995.

[17] Charles A. Anderson, Kathryn F.Jones, Jennifer Ryan. A Tow - Dimensional Genetic Algorithm for the Ising Problem. University of Colorado at Denver, Denver, USA.

[18] I.C.Lerman- R.Fngouenet. Algorithmes Génétiques Séquentiels et Parallèles pour une Représentation Affine des Proximités.
Rapport de recherche N°2570 , INRIA.

[19] Mehassouel Nadjib. Conception et réalisation d'échantillonnage spatial optimal appliqué à un problème de localisation. Thèse de Magister.

[20] Habib Fatma Zohra. Echantillonnage optimal avec erreurs d'arrondis appliqué à un problème de pharmacocinétique. Thèse de Magister.

[21] M. Reynolds S. Schiffman and W. Young. Introduction to Multidimensional Scaling, Theory, Methods and Application. Academic press.1981.

[22] R.M. Gray Vector Quantization *IEEE, ASSP* Magazine page 4-29 April 1984

[23] A.N. Netravali and B.G Haskell- Digital pictures : Representation and Compression- Plenum press. New York 1988.

[24] J.Max- Quantizing for minimum Distorsion- IEEE transactions of information theory IT-6: 7-12, March 1960.

[25] C.E.Slepion (ed.)-Key papers in the Development of Information Theory-
IEEE Press, New York 1979.

[26] C.E.Shanon-Coding theorems for a discrete source with fidelity criterion.
IRE National convention Record, part4, page 142-163, 1959.

[27] P. Zador asymptotic quantization error of continuous signals and their quantization dimension.- IEEE Transactions on information Theory, IT-28: 139-149, March 1982.

[28] Y.Linde, A Buzo and R.M.Gray-An Algorithm for Vector Quantizer Design.
IEEE Transactions on communucation. 28 : 84-95, 1980.

[29] J.H.Conway and N.J.A.Sloane-Voronoi Regions of lattices, Second Moments of polytopes and Quantization- IEEE transacyion on information theory, IT-28 : N°2, March 1982.

[30] M.C.Bueso et J.M.Angulo-Spatial Sampling Design Based on Stochastic Compexity-Journal of Multivariate Analysis 71,94-110 (1999).

[31] George Christakos and Bart R.Killam.Sampling Design for contaminant Level Using Anneling Search Algorithms – Water Resources Research, vol29, N°12, Pages 4063-4076, December 1993.

[32] Dong Wan Shin, Seuck Heun Song. Asymptotic efficiency of OLSE for Polynomial Regression Models with Spatially Correlated errors.
Statistics & Probability Letters 47(2000) 1-10

[33] K-Boukhetala and Z-Habib. Finite Sampling Design with Quantization for a Pharmacokinetic Problem. The 6th World Multi-Conference and informatics, Orlando, Florida, U.S.A pages 167-170. July 14-18 (2002).

[34] K-Boukhetala, K-Benhenni, et S. Benamara (1996). Optimal Samplig for Estimating Integral with Correlated Measurement Error. Pages 161-162. Compstat 96.

[35] k- Boukhetala et N. Mehassouel (2000) . Optimal Sampling Methods for Estimating Problems Based on Correlated Observations. Congrés international des mathématiques, Pays bas Aoùt 2000.

[36] K- Boukhetala and Sadoun Ait Kaci- Finite Spatial Sampling Design and Quantization. Compstat 2004 Symposium-Physica Verlag, Spinger 2004.

[37] Sadoun Ait Kaci Azzou et K- Boukhetala. Sur un Algorithme Génétique Spatial, pour Conception d'Echantillonnage Spatial Optimal Fini avec quantification .
MSS 2004.