

République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

**UNIVERSITÉ MOHAMED KHIDER, BISKRA**

FACULTÉ des SCIENCES EXACTES et des SCIENCES de la NATURE et de la VIE

**DÉPARTEMENT DE MATHÉMATIQUES**



Mémoire présenté en vue de l'obtention du Diplôme :

**MASTER en Mathématiques**

Option : **Analyse**

Par

**KHerroub Roumaïssa**

Titre :

**Sur les méthodes métaheuristiques**

Membres du Comité d'Examen :

Dr. <b>ZERROUG</b> Abdelhamid.....	UMKB	Président
Dr. <b>GHOUDJMIS</b> Fatiha.....	UMKB	Encadreur
Dr. <b>ZERROUG</b> Abdelhamid.....	UMKB	Examineur

Juin 2018

## DÉDICACE

Je dédie ce humble travail .....

## REMERCIEMENTS

.....

# Table des matières

<b>Remerciements</b>	<b>ii</b>
<b>Table des matières</b>	<b>iii</b>
<b>Liste des figures</b>	<b>v</b>
<b>Introduction</b>	<b>1</b>
<b>1 Généralités sur L'optimisation</b>	<b>3</b>
1.1 Modélisation . . . . .	3
1.1.1 La démarche de modélisation : . . . . .	4
1.1.2 Quelques caractéristiques de la fonction objectif . . . . .	7
1.2 Transformation du problème . . . . .	8
1.2.1 Les techniques de résolution des problèmes d'optimisation . . . . .	8
1.3 Classification des problèmes d'optimisation . . . . .	9
1.3.1 Optimisation sans contrainte . . . . .	9
1.3.2 Optimisation avec contrainte . . . . .	9
1.4 Métaheuristique . . . . .	9
1.4.1 Quelques méthodes métaheuristiques . . . . .	10

<b>2</b>	<b>L’algorithme de pollinisation de fleur</b>	<b>12</b>
2.1	Pollinisation de fleur dans la nature . . . . .	12
2.1.1	Pollinisation spontanée . . . . .	13
2.1.2	Pollinisation croisée . . . . .	13
2.2	L’algorithme de pollinisation de fleur . . . . .	14
2.2.1	Algorithme de fleur . . . . .	14
2.2.2	Les règles utilisées par l’algorithme de pollinisation de fleur . . . . .	14
2.2.3	Les étapes de l’algorithme : pollinisation de fleur . . . . .	15
2.2.4	Le pseudo-code de l’algorithme de pollinisation de fleur . . . . .	18
2.2.5	Domaines d’application de l’algorithme de pollinisation de fleur . . . . .	19
<b>3</b>	<b>Application sous Matlab</b>	<b>21</b>
3.1	Le code Matlab de l’algorithme de pollinisation de fleur . . . . .	21
	<b>Conclusion</b>	<b>28</b>
	<b>Bibliographie</b>	<b>28</b>
	<b>Annexe A : Notations</b>	<b>31</b>

# Table des figures

2.1	Pollinisation . . . . .	13
-----	-------------------------	----

# Introduction

Beaucoup des problèmes réels d'optimisation sont très complexes et stimulants pour résoudre, et ils sont employés dans plusieurs applications. Et pour résoudre ces problèmes il faut utiliser des algorithmes d'optimisation, et il y a aucune garantie qu'on peut trouver la solution optimale. La nature avait résolu beaucoup de problèmes pour des milliards d'années, et beaucoup des systèmes biologiques ont montré une fascination et une efficacité remarquable dans la résolution des problèmes. Au-dessus. Les dernières décennies, les algorithmes d'optimisation ont été appliqués dans un nombre étendu des problèmes difficiles. Récemment un nouvel algorithme inspiré par la nature, nommé l'algorithme de pollinisation de fleur, est développé par des scientifiques.

Des problèmes d'optimisation de technologie sont souvent résolus par l'utilisation des algorithmes métaheuristiques. L'algorithme de pollinisation de fleur (FPA) a été alors prolongé pour traiter l'optimisation à objectifs multiples.

L'objectif de ce mémoire est d'étudier une moderne méthode métaheuristique «L'algorithme de pollinisation de fleur»

►Ce travail est divisé en trois chapitres :

- Le **premier chapitre**, donne quelques Généralités sur L'optimisation :(L'optimisation, Modèle mathématique, Quelques caractéristiques de la fonction objectif, Quelques méthodes métaheuristiques...).

- Le **deuxième chapitre**, présente L'algorithme de pollinisation de fleur : (les

règles, les étapes et domaine d'application de L'algorithme de pollinisation de fleur...).

•Le **dernier chapitre** : est consacré à L'application de L'algorithme de pollinisation de fleur sous matlab.

Enfin, ce manuscrit se terminera par une conclusion générale.



# Chapitre 1

## Généralités sur L'optimisation

L'optimisation est l'étude de la planification efficace et de résoudre des problèmes de conception utilise les outils mathématiques. Les techniques d'optimisation traditionnelles ont placé beaucoup de limitations tandis que développer les modèles de recherches mathématiques et opérationnelles. Les solutions modèles dans des algorithmes d'optimisation traditionnels dépendent la plupart du temps du type de la fonction objectif et des contraintes.

**Définition 1.0.1** (*Optimum*) : (du latin, le meilleur.) Etat, degré de développement de quelque chose jugé le plus favorable au regard de circonstances données.

### 1.1 Modélisation

**Définition 1.1.1** (*Modèle mathématique*) On appelle un modèle mathématique toute représentation mathématique d'un phénomène, économique, humain, etc., réalisée afin de pouvoir mieux l'étudier.

La démarche de modélisation est à la fois passionnante et difficile. En effet aucune recette universelle n'existe et le nombre de modélisations possibles pour un problème donné n'a

d'autres limites que l'imagination du modélisateur. Cependant, il est essentiel de bien maîtriser les outils d'optimisation, et d'en connaître les hypothèses sous-jacentes afin de développer le modèle adéquat pour l'analyse visée.

Le besoin découle directement du besoin d'organiser. Optimiser consiste à identifier une configuration optimale, ou un optimum d'un système, au sens large du terme.

Dans le cadre d'une démarche scientifique, cette définition nécessite quelques précisions.

### 1.1.1 La démarche de modélisation :

La démarche de modélisation consiste en trois étapes :

1) L'identification des **variables de décision** : c'est-à-dire les composantes du système concerné sur lesquelles il est possible d'agir afin d'améliorer son fonctionnement. En général, si ces variables sont au nombre de  $n$ , elles sont représentées par un vecteur (colonne)

de  $\mathbb{R}^n$ , souvent noté  $x = (x_1 \dots x_n)$ , c'est-à-dire  $x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$ .

En pratique, cette étape est probablement la plus compliquée et la plus importante. La plus compliquée, car seules l'expérience en modélisation et une bonne connaissance du problème concret guident les choix. La plus importante, car tout le reste du processus en dépend. Un choix inapproprié de variables de décision peut générer un problème d'optimisation trop compliqué et impossible à résoudre.

2) La description de la méthode : qui permet d'évaluer l'état du système concerné, étant donné une configuration des variables de décision.

Nous supposons que le modélisateur est capable d'identifier une formule, une fonction, fournissant une mesure de l'état du système, mesure qu'il désire rendre la plus petite possible ou la plus grande possible. Cette fonction, appelée **fonction objectif**, est notée  $f$  et la mesure précitée obtenue pour les variables de décision  $x$  est un nombre réel noté

$f(x)$ .

3) La description mathématique des circonstances, ou **contraintes** : précisant les valeurs que les variables de décision peuvent prendre.

**Exemple 1.1.1** (*Swisscom*)

L'entreprise Swisscom voudrait installer une antenne doit se trouver le plus près possible de chaque client, en donnant la priorité aux meilleurs d'entre eux. Cependant, afin d'éviter les proliférations d'antennes à une distance inférieure à  $10km$  des deux autres, situées respectivement aux coordonnées  $[-5, 10]$  et  $[5, 0]$  les coordonnées sont exprimées en Kilomètres à partir du siège central swisscom. Pour chaque client, Swisscom connaît sa situation géographique, ainsi que Le nombre d'heures de communication que le client compte utiliser par mois. Ces données sont reprises dans le tableau 1.1 suivant. A quel endroit l'entreprise swisscom doit-elle installer sa nouvelle antenne ?

Client	Coord	Heures
1	$[5, 10]$	200
2	$[10, 5]$	150
3	$[0, 12]$	200
4	$[12, 0]$	300

TAB. 1.1 – Données sur les clients de swisscom

**-Variable de décision**

Swisscom doit identifier l'emplacement idéal pour son antenne, c'est-à-dire les coordonnées de cet emplacement. Nous définissons deux variables de décision  $x_1$  et  $x_2$  représentant ces coordonnées dans un système de référence donné.

**Fonction objectif**

La distance  $d_i(x_1, x_2)$  entre un client  $i$  situé aux coordonnées  $(a, b)$  et l'antenne est donnée par

$$d_i(x_1, x_2) = \sqrt{(x_1 - a)^2 + (x_2 - b)^2}$$

Afin de tenir compte du temps de communication, nous mesurons la somme des distances pondérées par les heures d'utilisation :

$$\begin{aligned} f(x_1, x_2) &= 200d_1(x_1, x_2) + 150d_2(x_1, x_2) + 200d_3(x_1, x_2) + 300d_4(x_1, x_2) \\ &= 200\sqrt{(x_1 - 5)^2 + (x_2 - 10)^2} \\ &\quad + 150\sqrt{(x_1 - 10)^2 + (x_2 - 5)^2} \\ &\quad + 200\sqrt{x_1^2 + (x_2 - 12)^2} \\ &\quad + 300\sqrt{(x_1 - 12)^2 + x_2^2} \end{aligned}$$

### Contraintes

Les contraintes sur les distances entre les antennes peuvent s'écrire :

$$\sqrt{(x_1 + 5)^2 + (x_2 - 10)^2} \geq 10$$

et

$$\sqrt{(x_1 - 5)^2 + x_2^2} \geq 10$$

Nous pouvons rassembler les différentes étapes de modélisation pour obtenir le problème d'optimisation suivant :

$$\begin{aligned} \min_{(x_1, x_2)} f(x_1, x_2) &= 200\sqrt{(x_1 - 5)^2 + (x_2 - 10)^2} \\ &\quad + 150\sqrt{(x_1 - 10)^2 + (x_2 - 5)^2} \\ &\quad + 200\sqrt{x_1^2 + (x_2 - 12)^2} \\ &\quad + 300\sqrt{(x_1 - 12)^2 + x_2^2} \end{aligned}$$

sous contraintes

$$\sqrt{(x_1 + 5)^2 + (x_2 - 10)^2} \geq 10$$
$$\sqrt{(x_1 - 5)^2 + (x_2 - 10)^2} \geq 10$$

### 1.1.2 Quelques caractéristiques de la fonction objectif

Plusieurs concepts permettent de caractériser la fonction objectif.

Il est important d'identifier les caractéristiques de la fonction objectif car chaque algorithme d'optimisation est basé sur des hypothèses spécifiques. Lorsque les hypothèses d'un algorithme ne sont pas vérifiées pour un problème donné, il n'y a aucune garantie que l'algorithme puisse être utilisé pour résoudre ce problème.

**Définition 1.1.2** (*Fonction convexe*) Une fonction  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  est dite convexe si pour tout  $x, y \in \mathbb{R}^n$  et pour tout  $\lambda \in [0, 1]$ , on a

$$f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y)$$

**Exemple 1.1.2** La fonction  $x \rightarrow |x|$  est convexe sur  $\mathbb{R}$  car si  $\lambda \in [0, 1]$ , alors :

$$|\lambda x + (1 - \lambda)y| \leq \lambda|x| + (1 - \lambda)|y|$$

**Définition 1.1.3** (*Fonction strictement convexe*) Une fonction  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  est dite strictement convexe si pour tout  $x, y \in \mathbb{R}^n$ ,  $x \neq y$ , et pour tout  $\lambda \in ]0, 1[$ , on a

$$f(\lambda x + (1 - \lambda)y) < \lambda f(x) + (1 - \lambda)f(y)$$

**Exemple 1.1.3** La fonction  $f$  définie sur  $\mathbb{R}$  par  $f(x) = x^2$  est strictement convexe sur  $\mathbb{R}$  car si  $\lambda \in ]0, 1[$  et  $x \neq y$  alors :

$$\begin{aligned} f(\lambda x + (1 - \lambda) y) &= \\ \lambda^2 x^2 + 2\lambda(1 - \lambda)xy + (1 - \lambda)^2 y^2 &< \lambda^2 x^2 + \lambda(1 - \lambda)(x^2 + y^2) + (1 - \lambda)^2 y^2 \\ &= \lambda x^2 + (1 - \lambda) y^2 \text{ (en utilisant } 2xy < x^2 + y^2 \text{)}. \end{aligned}$$

**Définition 1.1.4** (*Fonction concave*) Une fonction  $f : \mathbb{R}^n \rightarrow \mathbb{R}$  est dite concave  $-f$  est une fonction convexe, c'est-à-dire si pour tout  $x, y \in \mathbb{R}^n$  et pour tout  $\lambda \in [0, 1]$ , on a

$$f(\lambda x + (1 - \lambda) y) \geq \lambda f(x) + (1 - \lambda) f(y)$$

## 1.2 Transformation du problème

L'étape de modélisation étant terminée, nous ne sommes pas encore au bout de nos peines. En effet, il existe un nombre illimité de façons d'écrire mathématiquement un problème donné. Dès lors, les algorithmes et logiciels imposent une formulation particulière sur laquelle ils se basent pour résoudre le problème. Nous allons voir les techniques permettant de nous plier à leurs exigences.

### 1.2.1 Les techniques de résolution des problèmes d'optimisation

#### Minimum local

Appelons  $Y = \{x \in \mathbb{R}^n / h(x) = 0, g(x) \leq 0 \text{ et } x \in X\}$  l'ensemble des vecteurs vérifiant toutes contraintes. Le vecteur  $x^* \in Y$  est appelé minimum local du problème s'il existe  $\varepsilon > 0$  tel que  $f(x^*) \leq f(x), \forall x \in Y$  tel que  $\|x - x^*\| < \varepsilon$ .

## Minimum global

Appelons  $Y = \{x \in \mathbb{R}^n / h(x) = 0, g(x) \leq 0 \text{ et } x \in X\}$  l'ensemble des vecteurs vérifiant toutes contraintes. Le vecteur  $x^* \in Y$  est appelé minimum global du problème si  $f(x^*) \leq f(x), \forall x \in Y$ .

## 1.3 Classification des problèmes d'optimisation

### 1.3.1 Optimisation sans contrainte

On appelle problème d'optimisation de minimisation sans contrainte, le problème suivant :

$$\min_{x \in \mathbb{R}^n} f(x) \quad f : \mathbb{R}^n \rightarrow \mathbb{R}$$

### 1.3.2 Optimisation avec contrainte

La fonction  $f(x)$  est une contrainte, c'est-à-dire sujette à une autre fonction qui limite ses variables, qui se traduit par le fait que le vecteur de variables de décision doit appartenir a un domaine donné  $D \in \mathbb{R}^n$ , on écrit :

$$\left\{ \begin{array}{l} \min f(x), x = (x_1, x_2, x_3 \dots x_n) \\ g_m(x) \leq 0, m = 1, 2, \dots n_g \\ k_m(x) = 0, m = n_g + 1, \dots, n_g + n_k, \forall x \in \mathbb{R}^n \end{array} \right.$$

avec  $n_g$  et  $n_k$  est le nombre des contraintes d'inégalités respectivement (d'égalités).

## 1.4 Métaheuristique

**Définition 1.4.1** (*métaheuristique*) Le mot métaheuristique est dérivé de la composition de deux mots grecs :

- heuristique qui vient du verbe heuriskein et qui signifie ‘trouver’.
- méta qui est un préfixe signifiant ‘au-delà’, ‘dans un niveau supérieur’.

**Les métaheuristiques** sont un ensemble d’algorithmes d’optimisation visant à résoudre les problèmes d’optimisation difficiles. Elles sont souvent inspirées par des systèmes naturels.

**Propriété 1.4.1** *Les propriétés fondamentales des métaheuristiques sont les suivantes :*

- Les métaheuristiques sont des stratégies qui permettent de guider la recherche d’une solution optimale.
- Le but visé par les métaheuristiques est d’explorer l’espace de recherche efficacement afin de déterminer des solutions (presque) optimales.
- Les techniques qui constituent des algorithmes de type métaheuristique vont de la simple procédure de recherche locale à des processus d’apprentissage complexes.
- Les métaheuristiques sont en général non-déterministes et ne donnent aucune garantie d’optimalité.
- Les métaheuristiques peuvent contenir des mécanismes qui permettent d’éviter d’être bloqué dans des régions de l’espace de recherche.

### 1.4.1 Quelques méthodes métaheuristiques

#### **Algorithmes Génétiques :**

Les algorithmes génétiques appartiennent à la famille des algorithmes évolutionnistes. Leur but est d’obtenir une solution approchée à un problème d’optimisation. La solution optimale est cherchée à partir d’une population de solutions en utilisant des processus aléatoires. La recherche de la meilleure solution est effectuée en créant une nouvelle génération de solutions par application successive, à la population courante, de trois opérateurs : la



sélection, le croisement, et la mutation. Ces opérations sont répétées jusqu'à ce qu'un critère d'arrêt soit atteint.

**Algorithme de colonie d'abeille(ABC) :**

L'algorithme d'ABC est principalement appliqué dans des problèmes continus d'optimisation. Il simule le comportement de danse de remuer pendant lequel un essaim des abeilles exécute le procédé de fouiller des abeilles. L'algorithme d'ABC a une meilleure exécution dans la fonction problème d'optimisation. L'ABC peut produire à une solution plus optimale et est ainsi plus efficace que les autres méthodes dans plusieurs problèmes d'optimisation. À l'ABC, la position d'une source de nourriture représente une solution possible au problème, et la quantité de nectar d'une source de nourriture correspond à la qualité (forme physique) de la solution. L'ABC commence par les solutions aléatoires et essaye de trouver de meilleures solutions en recherchant les voisinages des meilleures solutions courantes. Les solutions sont représentées comme sources de nourriture qui chacune sont associées à une abeille est utilisée. Une égalité de nombre d'abeilles de spectateur chacun choisissent une de ces sources de nourriture à exploiter basé sur la qualité ou la forme physique, utilise le choix de roulette-roue. Spectateur et les abeilles sont utilisées essayent d'identifier de meilleures sources de nourriture dans le voisinage de leur courant source de nourriture en perturbant une dimension aléatoirement choisie de leur position de source de nourriture vers une autre source aléatoirement choisie de nourriture.

# Chapitre 2

## L'algorithme de pollinisation de fleur

La nature a inspiré des algorithmes. Les chercheurs avaient exploré de nouvelles manières de résoudre des problèmes d'optimisation nécessités à l'inefficacité de traditionnel algorithmes d'optimisation particulièrement pour résoudre de grands et compliqués problèmes. La popularité du Nature-Inspirée des algorithmes sont principalement influencés par la capacité des systèmes biologiques de s'ajuster effectivement sur fréquemment variable spirée des algorithmes environnement. L'intelligence d'essaim a été de grand intérêt pour beaucoup de scientifiques de recherches dans la dernière décennie.

### 2.1 Pollinisation de fleur dans la nature

La reproduction aux usines se produit par l'union des gamètes. Les grains de pollen produits par les gamètes masculins et des ovules soutenus par les gamètes femelles sont produits par différentes pièces et il est essentiel que le pollen doive être transféré au stigmate pour l'union. Ce processus de transfert et de dépôt des grains de pollen d'anthère au stigmate de la fleur est pollinisation. Le processus de la pollinisation est la plupart du temps facilité par un agent. La pollinisation est un résultat de la fertilisation et c'est nécessité dans l'agriculture pour produire les fruits et les graines.

Il y a deux types de pollinisation :

### 2.1.1 Pollinisation spontanée

Quand le pollen d'une fleur pollinise la même fleur ou fleurs de la même espèce, le processus s'appelle la Pollinisation spontanée. Il se produit quand une fleur contient les gamètes masculins et femelles.

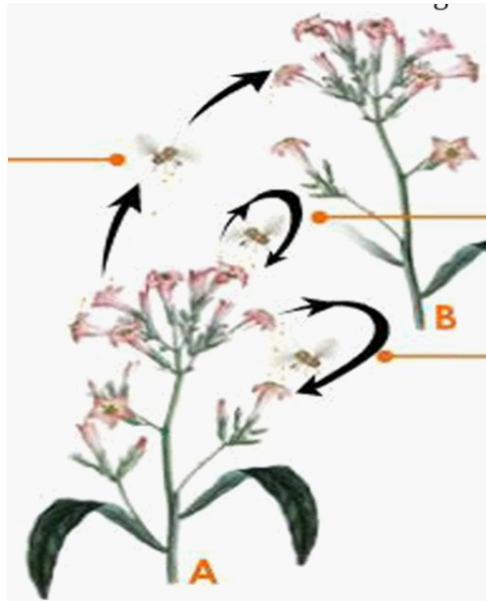


FIG. 2.1 – Pollinisation

### 2.1.2 Pollinisation croisée

La pollinisation en travers se produit quand des grains de pollen sont déplacés à une fleur d'une autre espèce. Le processus de la pollinisation en travers se produit avec l'aide des agents abiotiques ou biotiques tels que des insectes, des oiseaux, des escargots, des bates et d'autres animaux comme pollinisateurs.

La pollinisation abiotique est un processus où la pollinisation se produit sans participation des agents externes. Seulement environ 10% de la chute d'usines dans cette catégorie.

Le processus de la pollinisation qui exige les pollinisateurs externes est connu comme Pollinisation biotique pour déplacer le pollen de l'anthère au stigmate. Les insectes jouent un rôle important comme pollinisateurs. La pollinisation d'insecte se produit aux usines avec les pétales colorés et l'odeur forte qui attirent des abeilles de miel, mites, coléoptères, guêpes, fourmis et papillons. Les insectes sont attirés aux fleurs dues à la disponibilité du nectar, pollen comestible et quand l'insecte se repose sur la fleur, les grains de pollen collent au corps. Quand l'insecte visite une autre fleur, le pollen est transféré au stigmate facilitant la pollinisation. La pollinisation est également facilitée par des vertébrés comme des oiseaux et des chattes. Les fleurs pollinies par des chauves-souris ont la plupart du temps les pétales colorés blancs et l'odeur forte. Les oiseaux pollinies habituellement des fleurs avec pétales rouges et sans odeur.

## **2.2 L'algorithme de pollinisation de fleur**

### **2.2.1 Algorithme de fleur**

Le processus de pollinisation des plantes à fleurs a inspiré Xin-She Yang pour développer un algorithme de pollinisation des fleurs (FPA)2012.

### **2.2.2 Les règles utilisées par l'algorithme de pollinisation de fleur**

Pour plus de facilité, les quatre règles données ci-dessous sont utilisées :

1. La pollinisation croisée et biotique est considérée comme processus de pollinisation globale, et les pollinisateurs de pollen-transport se déplacent d'une manière qui obéit des vols de Lévy.
2. Pour la pollinisation locale, la pollinisation spontanée et abiotique est employée.

3. Les pollinisateurs, comme des insectes développent une fidélité de fleur, qui est comparable à la possibilité de reproduction proportionnelle à l'assortiment de deux fleurs impliquées.

4. L'interaction ou la commutation de la pollinisation locale et de la pollinisation globale est commandée par une probabilité  $p \in [0, 1]$  de commutateur, avec une légère polarisation vers la pollinisation locale.

### 2.2.3 Les étapes de l'algorithme : pollinisation de fleur

#### Pollinisation global et Pollinisation local

**La première étape :** Le processus global de pollinisation et le processus local de pollinisation sont deux étapes principales dans l'algorithme de pollinisation de fleur. Dans le processus global de pollinisation, des pollens de fleur sont portés par pollinisateurs, en employant la règle 1 et 3. Ces pollens peuvent voyager au-dessus d'une distance parce que les pollinisateurs peuvent voler et se déplacer à plus grande portée. Cette pollinisation globale peut être représentée mathématiquement par :

$$x_i^{t+1} = x_i^t + \gamma L(\lambda) (g^* - x_i^t)$$

Avec  $x_i^t$  le pollen  $i$  ou le vecteur  $x_i$  de solution à l'itération  $t$  et  $g^*$  est la meilleure solution optimal trouvée parmi toutes les solutions à l'itération usuel.

Ici le  $\lambda$  est un facteur de cadrage pour contrôler la taille d'étape. Avec  $\gamma$  le paramètre qui correspond à la force de la pollinisation, qui est essentiellement également une taille d'étape.

Par conséquent le paramètre est également la taille d'étape qui correspond à la force de la pollinisateurs peuvent être voyagés au-dessus d'une distance. Ici, le vol de Lévy peut être employé pour imiter cette caractéristique de déplacement. Donc  $L > 0$  exemplifie Lévy

distribution :

$$L \sim \frac{\lambda \Gamma(\lambda) \sin(\pi\lambda/2)}{\pi} \frac{1}{s^{1+\lambda}} \quad (s \gg s_0 > 0)$$

Ici,  $\Gamma(\lambda)$  est la fonction gamma et le lévy standard la distribution est valide pour de langues étapes  $s > 0$ . Cependant dans la théorie la taille  $s_0$  critique devrait être suffisamment grand,  $s_0 = 0.1$  ou même  $s_0 = 0.01$  peut être employé dans la pratique.

Ici, le  $\sim$  de notation signifie pour dessiner les nombres aléatoires qui obéissent la distribution du côté droit.

Par conséquent, la règle 2 et la règle 3 qui sont fondamentalement pour la pollinisation locale est représentée comme :

$$x_i^{t+1} = x_i^t + \varepsilon (x_j^t - x_k^t)$$

**La deuxième étape :** Ici,  $x_j^t$  et  $x_k^t$  sont des pollens de différentes fleurs de la même espèce. L'équation imite la constance de fleur dans des voisinages limités. En assumant mathématiquement si  $x_j^t$  et  $x_k^t$  venez des mêmes espèces ou choisi parmi la même population, ceci devient d'une manière équivalente une marche aléatoire locale, si un graphique peut être tiré  $\varepsilon$  d'une distribution uniforme dans  $[0, 1]$ .

- En principe, les activités de pollinisation de fleur peuvent se produire à toutes les balances, locales et globales. Mais en réalité, des pièces rapportées de fleur ou les fleurs adjacentes dans le voisinage pas aussi lointain sont plus probables pour être pollinies par le pollen local de fleur que ceux lointaines. Afin d'imiter ce dispositif, nous peut effectivement employer une probabilité de commutateur (règle 4) ou commutateur de probabilité  $p$  de proximité entre pollinisation globale commune à la pollinisation locale intensive. Pour commencer, nous pouvons employer une naïve valeur de  $p = 0.5$  comme une valeur initiale.

Une étude paramétrique a montré que  $p = 0.8$  peut fonctionner mieux pour la plupart des applications. Les études de préliminaire suggèrent que l'algorithme de fleur soit très

efficace, et a été prolongé à l'optimisation à objectifs multiples.

**Remarque 2.2.1** (*Processus de Lévy*)

Soit  $X = (X_t, t \geq 0)$  un processus stochastique défini sur un espace de probabilité  $(\Omega, F, P)$ .

On dit que  $X$  est à accroissements indépendants si, pour tout  $n \in \mathbb{N}^*$  et pour tous  $0 \leq t_1 \leq \dots \leq +\infty$ , les variables aléatoires

$$\left\{ X_{t_{j+1}} - X_{t_j}, 1 \leq j \leq n - 1 \right\}$$

sont indépendantes. On dit que  $X$  est accroissement stationnaire. Si pour tous  $t, s > 0$  la variable aléatoire  $X_{t+s} - X_s$  a même loi que  $X_t - X_0$ .

**Remarque 2.2.2** (*Loi gamma*)

a) **Fonction gamma**

On appelle fonction gamma ou intégrale de gamma, l'intégrale récurrente :

$$\Gamma(r) = \int_0^{+\infty} x^{r-1} e^{-x} dx \quad x \in \mathbb{R}, r > 0$$

b) **Loi gamma**

Une variable aléatoire absolument continue  $x$  suit une loi gamma de paramètres  $\lambda$  et  $r$  ( $\lambda > 0, r > 0$ ) si sa fonction densité s'écrit :

$$f(x) = \begin{cases} \frac{\lambda^r}{\Gamma(r)} e^{-\lambda x} x^{r-1} & \text{si } x \geq 0 \\ 0 & \text{sinon} \end{cases}$$

**Remarque 2.2.3** (*Loi bêta*)

La distribution appelée Béta de paramètres  $a, b$  ( $a > 0$ ), ( $b > 0$ ) que l'on notera  $B(a, b)$  s'écrit :

$$f(x) = \frac{1}{B(a, b)} x^{a-1} (1-x)^{b-1} \quad \text{si } x \in ]0, 1[$$

## 2.2.4 Le pseudo-code de l'algorithme de pollinisation de fleur

### Algorithm

1. Objective min or max  $f(x)$ ,  $x = (x_1, x_2, \dots, x_d)^t$
2. Initialize a population of  $n$  flowers/pollen gametes with random solutions
3. Find the best solution  $g^*$  in the initial population
4. Define a switch probability  $p \in [0, 1]$
5. **while** ( $t < MaxGeneration$ )
6.     **for**  $i = 1 : n$  (all  $n$  flowers in the population)
7.         **if**  $rand < p$
8.             Draw a (d-dimensional) step vector  $L$  which obeys a Lévy distribution
9.             Global pollination via  $x_i^{t+1} = x_i^t + \gamma L(\lambda)(g^* - x_i^t)$
10.         **else**
11.             Draw  $\varepsilon$  from a uniform distribution in  $[0, 1]$
12.             Randomly choose  $j$  and  $k$  among all the solutions
13.             Do local pollination via  $x_i^{t+1} = x_i^t + \varepsilon(x_j^t - x_k^t)$
14.         **end if**
15.         Evaluate new solutions
16.         if new solutions are better, update them in the population
17.     **end for**
18. Find the current best solution



## 2.2.5 Domaines d'application de l'algorithme de pollinisation de fleur

L'algorithme de pollinisation de fleur est un algorithme d'optimisation efficace avec une étendue des applications large.

◆**Génie chimique** : FPA est utilisé largement dans ce domaine par exemple pour l'identification de paramètre dans la modélisation d'équilibre liquide-liquide de la nourriture dans les systèmes thermodynamiques. Ou pour l'exécution des calculs de point critique en fluides à plusieurs éléments de réservoir dans l'industrie pétrolière.

◆**Génie civil** : L'optimisation des structures de botte est le principe fondamental le plus connu d'application de la technique de la construction. L'algorithme de pollinisation de fleur est utilisé pour l'optimisation de classement par taille des armatures planaires et de l'espace près proposition d'un processus de manipulation pour des contraintes d'effort et de déplacement. FPA est un algorithme concurrentiel pour des structures de botte selon la comparaison de méthodes précédemment développées. Généralement, la pièce de charpente de botte est groupée afin de raccourcir l'effort d'optimisation et l'empêchement pour emprisonner des gens du pays optimums. Et les recherches ont étudié l'optimisation de classement par taille des structures de botte sans grouper les membres en employant FPA.

◆**Génie mécanique** : Plusieurs problèmes de base d'optimisation de construction mécanique tels que la vitesse le réducteur, le train d'engrenage, la conception ressort de la compression de tension et le récipient à pression ont étudié en utilisant deux algorithmes hybrides de FPA combinés avec la grenouille saut de la recherche locale du recuit simulé. Conception optima également étudiée d'échangeur de chaleur, tête en bloc ondulée la conception optima, optimum a soudé la conception de faisceau et le contrôleur de PID accordant pour réponse d'étape donc ont utilisé FPA pour la stabilisation du rotatoire le système inversé de pendule et la méthode réduit au maximum la perte se sont pro-

duits parce que d'un retard de temps. Dans la production et l'industrie, l'optimisation du multi passage en machine des paramètres de rotation ont été faits en employant un FPA amélioré.

◆**Technologie électronique et de communication** : Des méthodes de métaheuristique ont été souvent utilisées dans la radio systèmes de communication ont développé un FPA basé la méthode dans la position annule des antennes réseau circulaires et des besoins de méthode basés par FPA moins de temps de calcul comparant à l'algorithme génétique. FPA mis en application à la synthèse de l'antenne réseau circulaire pour le niveau latéral de lobe commande de taille d'ouverture. Par l'utilisation Cauchy a basé la pollinisation globale, gens du pays augmentés la pollinisation et la probabilité dynamique de commutateur, FPA ont été améliorées pour résoudre la synthèse de modèle des réseaux d'antennes linéaires.

◆**De l'informatique** : Dans la compression d'image, FPA a été utilisé afin de diminuer la complexité de recherche de l'assortiment entre le bloc de gamme et le bloc de domaine. On utilisé FPA pour le seuillage à multi niveaux d'image problème. On résolu un problème racial planaire de graphique en employant a variante de FPA qui emploie des stratégies avides locales telles que l'opérateur local d'échange et opérateur local d'inverse de secondaire-ordre. A utilisé FPA sur processus d'optimisation du problème assorti de forme basé sur le potentiel atomique le modèle assorti et les approches précédentes sont surpassés. Donc présenté une approche rétinienne automatisée de segmentation de vaisseaux sanguins utilisant FPA dans lequel recherche le groupement optimum de l'image rétinienne faisceaux compacts sous quelques contraintes.

◆**D'autres applications de technologie** : FPA a été modifié et utilisé pour les problèmes discrets de l'établissement du programme de projet qui est contraint par des ressources. Selon les résultats, FPA a surpassé plusieurs algorithmes métaheuristiques classiques.

# Chapitre 3

## Application sous Matlab

Matlab est un système interactif et convivial de calcul numérique et de visualisation graphique destiné aux ingénieurs et scientifiques.

On utilise logiciel matlab pour donner les résultats de l'algorithme de pollinisation de fleur.

La fonction objectif utilise dans ce programme est Rosenbrock's 3D.

### 3.1 Le code Matlab de l'algorithme de pollinisation de fleur

```
function[best, fmin, N_iter] = fpa_demo(para)
% Default parameters
if nargin < 1,
    para = [20 0.8];
end
n = para(1);      % Population size, typically 10 to 25
p = para(2);     % probability switch
```

```

% Iteration parameters

N_iter = 2000;      % Total number of iterations

% Dimension of the search variables

d = 9;

%Lb=-2*ones(1,d);

%Ub=2*ones(1,d);

Lb = 0 * ones(1, d);

Ub = 3 * ones(1, d);

% Initialize the population/solutions

for i = 1 : n,

    Sol(i, :) = Lb + (Ub - Lb) .* rand(1, d);

    Fitness(i) = Fun(Sol(i, :));

end

% Find the current best

[fmin, I] = min(Fitness);

best = Sol(I, :);

S = Sol;

% Start the iterations – Flower Algorithm

for t = 1 : N_iter,

    % Loop over all bats/solutions

    for i = 1 : n,

        % Pollens are carried by insects and thus can move in

        % large scale, large distance.
    
```

```

        % This L should replace by Levy flights

        % Formula :  $x_{i^{t+1}} = x_{i^t} + L(x_{i^t} - gbest)$ 

if rand > p,

%% L=rand;

L = Levy(d);

dS = L. * (Sol(i, :) - best);

S(i, :) = Sol(i, :) + dS;

% Check if the simple limits/bounds are OK

S(i, :) = simplebounds(S(i, :), Lb, Ub);

% If not, then local pollenation of neighbor flowers

else

epsilon = rand;

    % Find random flowers in the neighbourhood

    JK = randperm(n);

    % As they are random, the first two entries also random

    % If the flower are the same or similar species, then

    % they can be pollenated, otherwise, no action.

    % Formula :  $x_{i^{t+1}} + epsilon * (x_{j^t} - x_{k^t})$ 

    S(i, :) = S(i, :) + epsilon * (Sol(JK(1), :) - Sol(JK(2), :));

    % Check if the simple limits/bounds are OK

    S(i, :) = simplebounds(S(i, :), Lb, Ub);

end

% Evaluate new solutions

```

```

Fnew = Fun(S(i,:));

% If fitness improves (better solutions found), update then
    if (Fnew <= Fitness(i)),
        Sol(i,:) = S(i,:);
        Fitness(i) = Fnew;
    end

% Update the current global best
    if Fnew <= fmin,
        best = S(i,:);
        fmin = Fnew;
    end

end

% Display results every 100 iterations
    if round(t/100) == t/100,
        best
        fmin
    end

end

% Output/display
disp(['Total number of evaluations : t, num2str(N_iter * n)]);
disp(['Best solution = t, num2str(best), fmin = t, num2str(fmin)']);

% Application of simple constraints
functions = simplebounds(s, Lb, Ub)

```

```

% Apply the lower bound

ns_tmp = s;

I = ns_tmp < Lb;

ns_tmp(I) = Lb(I);

% Apply the upper bounds

J = ns_tmp > Ub;

ns_tmp(J) = Ub(J);

% Update this new move

s = ns_tmp;

% Draw n Levy flight sample

function L = Levy(d)

% Levy exponent and coefficient

% Xin-She Yang, Nature-Inspired Optimization Algorithms, Elsevier, (2014).

beta = 3/2;

sigma = (gamma(1 + beta) * sin(pi * beta/2))/(gamma((1 + beta)/2) * beta * 2^((beta -
1)/2))^(1/beta);

u = randn(1, d) * sigma;

v = randn(1, d);

step = u./abs(v).^(1/beta);

L = 0.01 * step;

% Objective function and here we used Rosenbrock's 3D function

function z = Fun(u)

z = (1 - u(1))^2 + 100 * (u(2) - u(1)^2)^2 + 100 * (u(3) - u(2)^2)^2;

```

La meilleur solution donne par la fonction Rosenbrock's  $3D$  est :

1.0000 1.0000 1.0000 1.8826 1.0154 2.9399 2.5581 3.0000 0 2.3386



# Conclusion

Dans ce mémoire, on a présenté une étude d'une méthode métaheuristique de l'algorithme de pollinisation de fleur.

Dans le premier chapitre de ce travail, nous avons évoqué d'abord quelques généralités sur l'optimisation. Par la suite nous sommes intéressés par quelques méthodes métaheuristicques. Nous exposons les généralités, définitions, exemples, et explication d'un modèle mathématique et quelques caractéristiques de la fonction objectif. Le deuxième chapitre est consacré à L'algorithme de pollinisation de fleur qui est caractérisé par les quatre règles et leurs étapes et le pseudo-code de cet algorithme. Et on a parlé du domaine d'application de l'algorithme de pollinisation de fleur. Enfin, et d'après le dernier chapitre qui est intéressé à l'application sous Matlab de l'algorithme de pollinisation de fleur on conclure l'efficacité de cet algorithme dans la résolution des plusieurs problèmes.

# Bibliographie

- [1] Michel Bierlaire. Introduction à l'optimisation différentiable. Presses polytechniques et universitaires romandes.
- [2] Kamalam Balas ubramani, Karnan Marcus. A Study on Flower Pollination Algorithm and Its Applications. International Journal of Application or Innovation in Engineering and Management(IJAIEM), Volume 3, Issue11, November 2014.
- [3] I.H. Osman and G. Laporte, Metaheuristics : a bibliography. Annals of Operations Research 63, , 1996.
- [4] Rhodes Rémi.(18 novembre 2010). Processus de Lévy et calcul stochastique.
- [5] Nazmus Sakib, Md. Wasi Ul Kabir, Mohammad Shafiul Alam, Md Subbir Rahman. A Comparative Study of Flower Pollination Algorithm And Bat Algorithm on Continuous Optimization Problems. International Journal of Applied Information Systems (IJ AIS) – ISSN Foundation of Computer Science FCS, New York, USA Volume 7–No.9, September 2014 – [www.ijais.org](http://www.ijais.org).
- [6] Sidi Mohamed Douiri, Souad Elbernoussi, Halima Lakhbab. Cours des Méthodes de Résolution Exacte Heuristiques et Métaheuristiques, Master Codes, Cryptographie Et Sécurité De L'information .
- [7] T. Stützle. Local Search Algorithms for Combinatorial Problems – Analysis, Algorithms and New Applications. DISKI – Dissertationen zur Künstlichen Intelligenz. Infix, Sankt Augustin, Germany (1999).

- [8] K.Redjda.(12/2004) Cour probabilités, office des publications universitaires.
- [9] Souhil Mouassa. Optimisation de l'écoulement de puissance par une méthode méta-heuristique (technique des abeilles) en présence d'une source renouvelable (éolienne) et des dispositifs FACTS.De Magister En Electrotechnique.Universite Ferhat Abbass Setif .19/06/2012.
- [10] Site web [http ://www.mathworks.com/](http://www.mathworks.com/)
- [11] Merzougui, A., Labeled, N., Hasseine, A., Bonilla-Petriciolet, A., Laiadi, D., & Bacha, O. (2016). Parameter Identification in Liquid-Liquid Equilibrium Modeling of Food-Related Thermodynamic Systems Using Flower Pollination Algorithms. *The Open Chemical Engineering Journal*, 10(1).
- [12] Shehata, M. N., Fateen, S. E. K., & Bonilla -Petriciolet, A. (2016). Critical point calculations of multi-component reservoir fluids using nature-inspired metaheuristic algorithms. *Fluid Phase Equilibria*.
- [13] Bekdaş, G., Nigdeli, S. M., & Yang, X. S. (2015). Sizing optimization of truss structures using flower pollination algorithm. *Applied Soft Computing*.
- [14] Bekdas G, Nigdeli SM, Yang XS. Size Optimization of Truss Structures Employing Flower Pollination Algorithm without Grouping Structural Members. *International Journal of Theoretical and Applied Mechanics*, 2017.
- [15] Kavirayani, S., & Kumar, G. V. (2017). Flower Pollination for Rotary Inverted Pendulum Stabilization with Delay. *Telkomnika*, 15(1).
- [16] Xin-She Yang.Flower pollination algorithm for global optimization, in : *Unconventional Computation and Natural Computation 2012, Lecture Notes in Computer Science*, Vol. 7445, pp.(2012).
- [17] Chakravarthy, V. V. S. S. S., & Rao, P. M. (2015, February). On the convergence characteristics of flower pollination algorithm for circular array synthesis. In *Electronics and Communication Systems (ICECS), 2015 2nd International Conference. IEEE*.

- [18] Chakravarthy, V. V. S. S. S., Chowdary, P. S., Rao, P. M., & Panda, G. Synthesis of Circular Array Antenna for Sidelobe Level and Aperture Size Control using Flower Pollination Algorithm. *International Journal of Antennas and Propagation*, Volume 2015, Article ID 819712.
- [19] Aylin Ece Kayabekir, Gebraail Bekdaş, Melih Nigdeli and Xin-She Yang. 2018 A Comprehensive Review of the Flower Pollination Algorithm for Solving Engineering Problems.
- [20] Ke-Lin Du, M.N.S Swamy. 2010 *Search and Optimization by Metaheuristic and Algorithms Inspried by Nature*.

## Annexe A : Notations

**Matlab** : est un logiciel de calcul numérique produit par Math Works.

Il est disponible sur plusieurs plateformes contient également une interface graphique puissante, ainsi qu'une grande variété d'algorithmes scientifiques.

On peut écrire des fonctions dans le Matlab que l'on peut ensuite appeler depuis est un fichier contenant des instructions Matlab appelé un script.

**La fonction objectif Rosenbrock's** : Dans l'optimisation mathématique, la fonction du Resrnbrock's est une fonction non convexe, présentée par Howard H.Rosenbrock 1960, qui est employé en tant que problème de test de performance pour des algorithmes d'optimisation. la fonction est définie par :

$$f(x, y) = (a - x)^2 + b(y - x^2)^2$$

Elle a un minimum global à  $(x, y) = (a, a^2)$ , là où  $f(x, y) = 0$ . Habituellement ces paramètres sont placés tels que  $a = 1$  et  $b = 100$ . Seulement dans le cas insignifiant où  $a = 0$  dans la fonction symétrique et le minimum à l'origine.

