الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي و البحث العلمي
Ministère de l'enseignement Supérieur et de la recherche scientifique

Université Mohamed Khider Biskra
Faculté des Sciences et de la Technologie
Département de Génie Electrique
Filière : Electronique
Option : Système embarqué

Mémoire de Fin d'Etudes
En vue de l'obtention du diplôme:

MASTER

Thème

Algorithme de recherche par organisme symbiotique : étude et application

Présenté par :
Noureddine Samia

Avis favorable de l'encadreur :
Dr. TOUMI Abida

Avis favorable du Président du Jury :

Dr. ZEHANI Soraya

Cachet et signature

Université Mohamed Khider de Biskra
Faculté des Sciences et de la Technologie
Département de génie électrique

MÉMOIRE DE MASTER

Sciences et Technologies
Electronique
Système Embarqué

Réf. : Entrez la référence du document

Présenté et soutenu par :
Noureddine Samia

Le : lundi 25 juin 2018

Algorithme de recherche par organisme symbiotique : étude et application

---

Jury :

Dr.	Zehani Soraya	MCB	Université de Biskra	Président
Dr.	Toumi Abida	MCA	Université de Biskra	Rapporteur
Mlle.	Medouakh Saadia	MAA	Université de Biskra	Examinateur

Année universitaire : 2017 - 2018

Dédicaces

Je dédie ce travail à ma petite et grande famille

Remerciements

Je tiens premièrement à me prosterner devant Allah le tout puissant de m'avoir donné le courage et la patience pour terminer ce travail.

Ce mémoire a été réalisé avec le soutien et l'assistance de personnes pour lesquelles je saisie cette occasion pour leur exprimer ma gratitude.

Je tiens tout d'abord à exprimer ma profonde reconnaissance à mon encadreur mademoiselle **Toumi Abida** maître de conférences A de m'avoir honoré par sa direction et la proposition de ce projet. A travers elle, j'ai découvert un nouveau domaine de recherche.

Durant ce projet, elle n'a pas cessé de m'orienter avec ces conseils, ces orientations ce qui a permis à ce projet d'être à ce niveau.

Je suis très reconnaissant mademoiselle **Medouakh Saadia** et mademoiselle **Zehani Soraya**, qui ont accepté de m'honorer avec leurs présences après une lecture approfondie pour évaluer mon travail et me fournir des critiques et remarques pertinentes l'améliorant.

Je ne peux pas laisser cette occasion sans remercier vivement la doctorante **Betka Abir** pour sa présence près de moi et ses idées constructives. Qu'elle trouve ici toute ma profonde reconnaissance.

Sommaire

Liste des abréviations

ABC : Artificial Bee Colony Algorithm (L'algorithme de la colonie d'abeilles)

CA : Cultural Algorithm (L'algorithme culturel

DE : Differential Evolution (L'algorithme à évolution différentielle)

DM : Data Mart

DW : Data Warhouse

GA : Genetic Algorithm (algorithme génétique)

GWO: Grey Wolf Optimisation (L'optimisation du loup gris)

MC : Matrice de confusion

NP-difficiles **:** Non déterministe Polynomial-difficiles

OC : Optimisation Combinatoire

PSO : Particle Swarm Optimization (optimisation par essaim de particules)

ROC : Receiver Operating Characteristic

SA : Simulated Annealing (recuit simulé)

SCA : Sine Cosine Algorithm (L'algorithme de Sine Cosine)

SOS : Symbiotic Organisms Search (Algorithme de recherche par les organismes symbiotiques)

SQL : Structured Query Langage (c'est un langage de base de données)

Liste des figures

Chapitre II : Eta de l'art sur le datamining

Chapitre III : Evaluation de la symbiotique

Liste des figures

Liste des tableaux

Chapitre II : Etat de l'art sur le data mining

Chapitre III : Evaluation de la symbiotique

Chapitre I

Etat de l'art sur l'optimisation et les méta heuristiques

I.1 Introduction

La résolution automatique de problème a envahi plusieurs domaines de l'ingénierie et de notre vie sociale. Certain type de problème sont difficile à résoudre par une approche algorithmique car la modélisation de solution ne suit pas un principe mathématique modélisable par des méthodes formelles. A cet effet, les chercheurs se sont orientés vers ce type de problème par besoin et par souci de découverte de solution aux problèmes d'optimisation combinatoire **[1]**. Les problèmes d'optimisation combinatoire sont parfois faciles à définir mais ils sont généralement difficiles à résoudre. Leur particularité réside dans le fait que la plupart de ces problèmes sont de type des problèmes NP-difficiles. Une autre particularité vient aussi du fait que ces problèmes ne possèdent pas à ce jour de solution algorithmique efficace capable de traiter toutes les données **[1]**.

Des travaux de recherche ont été réalisés avec l'apparition de techniques approchées très efficaces appelées métaheuristiques **[1]**.

Les métaheuristiques se basent sur les méthodes de voisinage comme le recuit simulé et la recherche tabou, et les algorithmes évolutifs comme les algorithmes génétiques et les stratégies d'évolution. Ces métaheuristiques ont l'avantage de générer des solutions approchées pour des problèmes d'optimisation classiques de grande taille. Pendant ces dernières années un grand intérêt applicatif concerne les métaheuristiques plus spécialement dans le domaine de la recherche opérationnelle mais surtout en intelligence artificielle **[2]**.

Dans ce chapitre nous présenterons les concepts fondamentaux de l'optimisation combinatoire, ensuite nous détaillerons les différentes métaheurisqtiques. Une description particulières sera donné à la métaheuristque de notre étude à savoir la symbiotique.

I.2 Optimisation combinatoire

L'optimisation combinatoire est un domaine d'actualité et à la pointe de la combinatoire et de l'informatique théorique qui vise à utiliser des techniques combinatoires pour résoudre des problèmes d'optimisation discrets. Un problème d'optimisation discrète cherche à déterminer la meilleure solution possible à partir d'un ensemble fini de possibilités **[1].** Les problèmes d'optimisation combinatoire apparaissent dans une multitude d'applications réelles, telles que le routage, l'affectation, l'ordonnancement, le découpage et l'emballage, la conception de réseau, l'alignement des protéines et bien d'autres domaines d'importance économique, industrielle et scientifique. Les techniques disponibles pour ces problèmes peuvent être grossièrement classées en deux catégories principales : les méthodes exactes et métaheuristiques **[1].**

I.2.1 Définition de l'optimisation combinatoire

L'optimisation combinatoire (OC) est une approche de résolution de problème n'ayant pas de solution algorithmique car la plupart de ces problèmes appartiennent à la classe des problèmes NP-difficiles. L'objectif de l'optimisation combinatoire est de trouver l'optimum c'est-à-dire minimiser ou maximiser une fonction dite fonction coût a une ou plusieurs variables soumises à des contraintes **[1] [2]**.

I.2.2. Résolution d'un problème d'optimisation combinatoire

La résolution d'un problème d'optimisation combinatoire demande l'étude des points suivants **[1][2]** :

- modélisation d'un problème : espace de recherche, solutions ;
- formulation mathématique : fonction objectif, contraintes ;
- application d'une méthode d'optimisation ;
- obtention d'une solution.

I.2.3 Les méthodes d'optimisation combinatoire

Les méthodes d'optimisation peuvent être classées en deux grandes classes de techniques pour la résolution des problèmes **[1][2]**:

- les méthodes exactes ;
- les méthodes approchées ou heuristiques et métaheuristiques.

I.2.3.1 Les méthodes exactes

Les algorithmes exacts sont des techniques exploitées pour chercher au moins une solution optimale d'un problème. La caractéristique principale de ces méthodes exactes vient du fait qu'elles ne sont efficaces que pour les instances de problèmes de petite taille **[1][2].**

Parmi les algorithmes exacts les plus exploités et qui ont fait preuve de robustesse nous citons **[2]**:

- les méthodes de recherche arborescente (Branch & bound),
- la programmation dynamique,
- la programmation linéaire,
- La programmation non linéaire.

❖ **La méthode de branch and bound**

L'objectif de La méthode de branch and bound (en français évaluation et séparation progressive) est d'énumérer les solutions d'une manière intelligente de manière à exploiter certaines propriétés du problème à résoudre pour éliminer des solutions partielles qui ne mènent pas à la solution désirée **[1][2]**.

Ainsi, l'algorithme arrive souvent à donner la solution recherchée en des temps raisonnables. Bien entendu, dans le pire cas, l'algorithme décide toujours sur l'élimination explicite de toutes les solutions du problème, son algorithme comme suit **[2]** :

Début
Placer le nœud début de longueur *0* dans une liste.
Répéter
Si la première branche contient le nœud recherché **alors**
Fin avec succès.
Sinon
- Supprimer la branche de la liste et former des branches nouvelles en étendant la branche supprimée d'une étape.
- Calculer les coûts cumulés des branches et les ajouter dans la liste de telle sorte que la liste soit triée en ordre croissant.
Finsi
Jusqu'à (liste vide ou nœud recherché trouvé)
Fin

❖ **Programmation Dynamique**

La programmation dynamique permet d'obtenir une solution optimale d'un problème par la somme des solutions des sous-problèmes résolus de façon optimale. Il est ainsi recommandé de diviser un problème donné en sous-problèmes et les résoudre un par un, son algorithme donné en quatre étapes **[2]** :

- caractériser la structure d'une solution optimale ;
- définir récursivement la valeur d'une solution optimale ;
- calculer la valeur d'une solution optimale en remontant progressivement jusqu'à l'énoncé du problème initial ;
- construire une solution optimale pour les informations calculées.

❖ **La programmation linéaire**

La programmation linéaire est une technique de l'optimisation pour résoudre de nombreux problèmes économiques et industriels. La programmation linéaire indique la façon de résoudre les problèmes dont la fonction objective et les contraintes sont toutes linéaires [2].

Nombreux problèmes réels de recherche opérationnelle peuvent être formulés comme un problème de PL. A cet effet, un grand nombre d'algorithmes pour la résolution d'autres problèmes d'optimisation sont engendrés à partir de la résolution de problèmes linéaires [2].

I.2.3.2 Les méthodes approchées

Les méthodes de résolution exactes ont pour objectif l'obtention d'une solution dont l'optimalité est garantie. Dans certain cas il est plus intéressant de trouver des solutions de meilleure qualité sans garantie d'optimalité au profit d'un temps de calcul plus réduit **[1][2]**. A cet effet, il est conseillé d'utiliser des méthodes appelées métaheuristiques, adaptées à chaque problème traité, avec cependant l'inconvénient de ne disposer en retour d'aucune information sur la qualité des solutions obtenues **[1].**

Les heuristiques ou les méta-heuristiques exploitent généralement des processus aléatoires dans l'exploration de l'espace de recherche pour faire face à l'explosion combinatoire engendré par l'utilisation des méthodes exactes **[1].** La figure (fig. I.1) suivante donne un panorama des méthodes les plus utilisées **[2].**

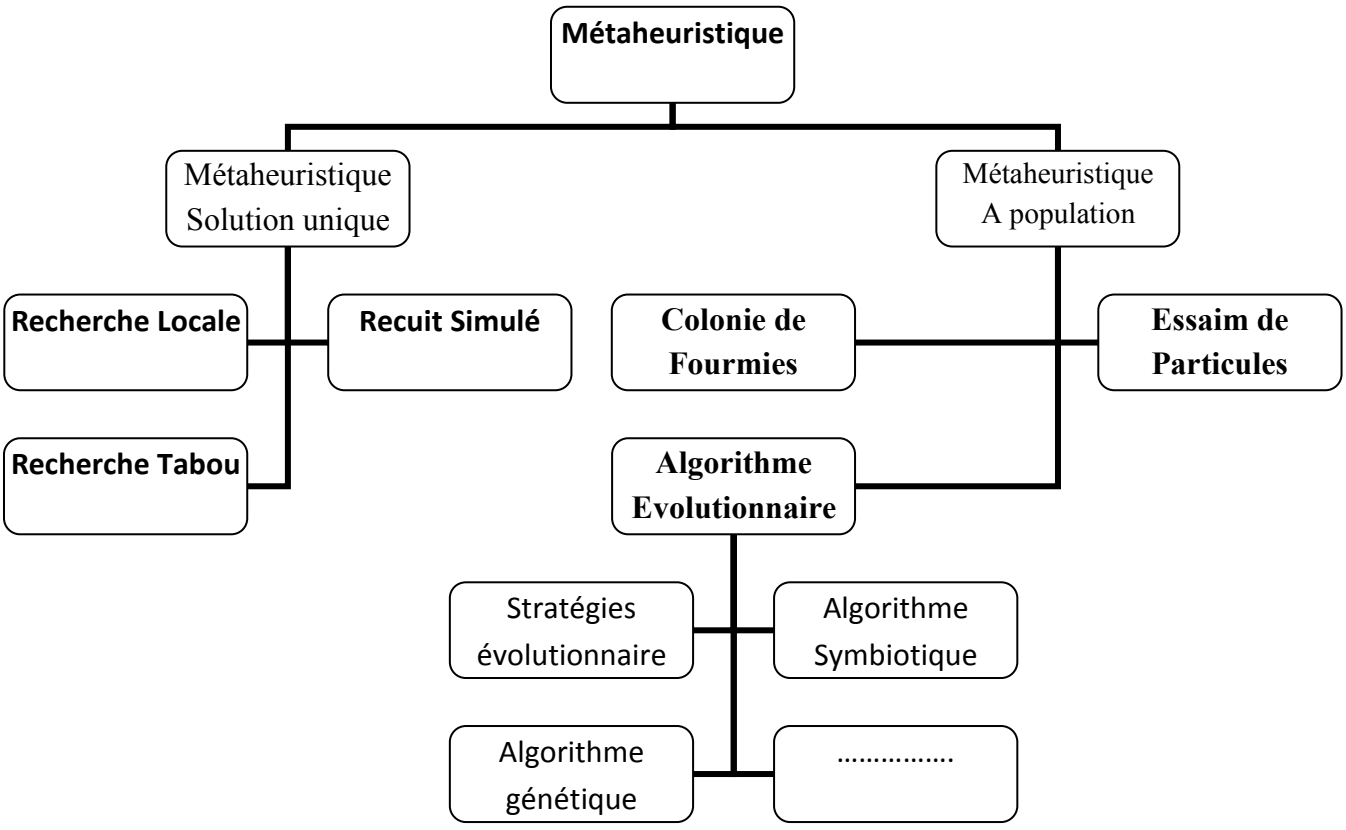


Figure I.1. Classes des méthaheuristiques **[2].**

❖ **Recuit simulé**

Le recuit simulé (SA) a été introduit comme une méthode de recherche locale normale, utilisant une stratégie pour éviter les minima locaux. Cette métaheuristique est basée sur une technique utilisée depuis longtemps par les métallurgistes qui, pour obtenir un alliage sans défaut, faisant alterner les cycles de réchauffage (ou de recuit) et de refroidissement lent des métaux **[6][7]**.

Le processus se poursuit tant que l'énergie du système diminue. Lorsque l'énergie reste stationnaire, une diminution graduelle de la température est effectuée avant de reprendre le processus pour décroitre l'énergie. La recherche s'arrête lorsque les diminutions de température restent inefficaces, généralement quand la valeur de la température tend vers zéro **[7]**. L'algorithme du recuit simulé est comme suit :

1 Définir la fonction objectif (f).

2 Choix des mécanismes de perturbation d'une configuration ΔS.

3 Tirer une configuration aléatoire S.

4 Calculer l'énergie associée à cette configuration E.

5 Initialiser la température (T0).

6 **Tant que** Conditions d'arrêts pas satisfaites **faire**

6.1. **Tant que** l'équilibre thermodynamique pas atteint **faire**

6.1.1. Tirer une nouvelle configuration S'.

6.1.2. Appliquer la règle de Metropolis.

6.1.3. **Si** f (S') < f(S) **Alors**

f_{min} =f(S ')

S_{opt} =S'

Fin de si

Fin tant que

6.2. Décroître la température.

Fin tant que

7 Afficher la solution optimale

En résumé, les principaux paramètres de contrôle sont **[7]** :

- la valeur initiale de la température ;
- la fonction de décroissance de la température ;
- le critère de changement de palier de température ;
- les critères d'arrêt.

❖ **Recherche Tabou**

La recherche tabou est introduite principalement par Glover et Laguna dans **[8]**. L'idée est basée sur le principe suivant : « Garder des traces du passé pour mieux s'orienter dans le futur ». L'idée est d'utiliser une (petite) mémoire (la liste tabou) pour éviter de tomber dans un optimum local et/ou pour éviter de boucler (cycles de petite taille). Dans la liste tabou, on peut garder des configurations, des points ou des régions visitées, ou plus généralement des attributs, qui vont éviter des « mouvements » déjà faits. Le processus de fonctionnement de cette technique est décrit dans ce qui suit **[9]** :

- choisir la taille k de la liste tabou LTABOU
- choisir un nombre d'itération NB
- choisir une solution initiale s
- meilleure_evaluation <− f(s)
- meilleure_solution <− s
- change <− VRAI

Tant que {change = VRAI}
change <− FAUX
Pour {iteration = 1 jusqu'à NB} **faire**
identifier le voisinage N(s)
T(s,k) <- les points de N(s) de la liste LTABOU
N(s,k) <- N(s)-T(s,k)+A(s,k)
trier le voisinage en fonction de la fonction f
s' = élément de N(s,k) tel que f(s') minimum
Si {f(s') < meilleure_evaluation} **alors**
meilleure_solution <− s'
meilleure_evaluation <− f(s')
change = VRAI
Fin si
mettre à jour la liste tabou,
(i.e. ajouter N(s,k) à LTABOU)
s <− s'
Fin pour
Fin tant que

❖ Algorithme Génétique

Les algorithmes génétiques (AG) sont des algorithmes d'optimisation fondés sur les mécanismes de la sélection naturelle et de la génétique. Ils ont été adaptés à l'optimisation par John **[10]**, un autre chercheur en l'occurrence David Goldberg qui a beaucoup travaillé pour améliorer l'aspect formel et fonctionnel **[11]**. L'inspiration de cette technique repose sur la théorie de l'évolution et de la génétique. A cet effet, on utilise le concept d'individu (solution potentielle), population (ensemble de solutions), génotype (une représentation de la solution), gène (une partie du génotype), parent, enfant, reproduction, croisement, mutation, génération, etc.

L'aspect fonctionnel repose sur une approche partant d'une population de solutions potentielles (chromosomes) initiales, arbitrairement choisies, par la suite, une étape d'évaluation consiste à estimer leur performance (Fitness) relative. A partir de ces performances, le processus génère une nouvelle population de solutions potentielles en utilisant des opérateurs évolutionnaires simples : la sélection, le croisement et la mutation. De cela découle que certain individus se reproduisent, d'autres disparaissent et seuls les individus les mieux adaptés sont supposés survivre. Ce cycle est itéré autant de fois que cela nécessite jusqu'à ce qu'une solution satisfaisante soit obtenue. En effet, l'héritage génétique à travers les générations permet à la population d'être adaptée et donc répondre au critère d'optimisation, Un algorithme génétique recherche le ou les extrema d'une fonction définie sur un espace de données. La mise en œuvre de cet algorithme est comme décrite dans le processus suivant **[11]** :

Début : Instance de problème I

1 : Choisir un codage pour représenter une configuration

2 : Choisir la taille de la population

3 : Générer une population

4 : **Répéter**

5 : Sélectionner deux configurations dans la population a Paide d'un opérateur de sélection

6 : Combiner deux configurations parents a l'aide d'un opérateur de croisement pour obtenir un enfant

7 : Appliquer éventuellement un opérateur de mutation a l'enfant

8 : Evaluer la qualité de l'enfant

9 : Insérer l'enfant dans la population

10 : Eliminer éventuellement des configurations de la population avec un opérateur d'élimination

11 : **Jusqu'a** critère d'arrêt

Fin : Meilleur Horaire $S_{meilleur}$ pour l'instance I

❖ L'algorithme de la colonie d'abeilles

L'algorithme de la colonie d'abeilles artificielles (Artificial Bee Colony Algorithm : ABC), proposé par Karaboga en 2005 pour l'optimisation des paramètres réels, est un algorithme d'optimisation récemment introduit et simule le comportement de recherche de colonie d'abeilles pour des problèmes d'optimisation. Pour résoudre les problèmes d'optimisation sous contrainte, une méthode de gestion des contraintes a été incorporée à l'algorithme **[12]**.

Les principales étapes de l'algorithme ABC simulant ces comportements sont les suivantes :

1. Initialiser les positions de la source de nourriture.
2. Chaque abeille employée produit une nouvelle source de nourriture dans son site de source de nourriture et exploite la meilleure source.
3. Chaque abeille regarde une source en fonction de la qualité de sa solution, produit une nouvelle source de nourriture dans le site de la source de nourriture sélectionnée et exploite la meilleure source.
4. Déterminer la source à abandonner et allouer son abeille employée comme éclaireur pour la recherche de nouvelles sources de nourriture.
5. Mémorisez la meilleure source de nourriture trouvée jusqu'ici.
6. Répétez les étapes 2 à 5 jusqu'à ce que le critère d'arrêt soit satisfait.

❖ Optimisation par essaim de particules

L'optimisation par essaim de particules (Particle Swarm Optimization PSO) est une méthode d'optimisation stochastique, pour les fonctions non-linéaires, basée sur la reproduction d'un comportement social. L'origine de cette méthode vient des observations faites lors des simulations informatiques de vols groupés d'oiseaux et de bancs de poissons. Ces simulations ont mis en valeur la capacité des individus d'un groupe en mouvement à conserver une distance optimale entre eux et à suivre un mouvement global par rapport aux mouvements locaux de leur voisinage **[13]**.

Les particules sont les individus et elles se déplacent dans l'hyperespace de recherche en se basant sur des informations limitées **[13]** :

1. Chaque particule est dotée d'une mémoire qui lui permet de mémoriser le meilleur point par lequel elle est déjà passée et elle possède une tendance à retourner vers ce point.

2. Chaque particule est informée du meilleur point connu au sein de son voisinage et elle va tendre à aller vers ce point. Chaque individu utilise donc, non seulement, sa propre mémoire, mais aussi l'information locale sur ses plus proches voisins pour décider de son propre déplacement.

Des règles simples, telles que " aller à la même vitesse que les autres ", " se déplacer dans la même direction " ou encore " rester proche de ses voisins " sont des exemples de comportements qui suffisent à maintenir la cohésion de l'essaim.

Le déplacement d'une particule est influencé par les trois types de comportement :

1. Une composante physique : la particule tend à suivre sa suivre sa propre voie ;

2. Une composante cognitive : la particule tend à revenir vers le meilleur site par lequel elle est déjà passée ;

3. Une composante sociale : la particule tend à se diriger vers le meilleur site déjà atteint par ses voisins.

- **L'algorithme de Sine Cosine**

L'algorithme de Sine Cosine (Sine Cosine Algorithm SCA) est un nouvel algorithme méta-heuristique. Il est basé sur la population et consiste à trouver le processus d'optimisation avec un ensemble de solution aléatoire. Ces solutions aléatoires sont calculées à plusieurs reprises au cours des itérations par une fonction objective **[14].**

La probabilité de trouver des optima globaux est augmentée avec le nombre suffisant de solutions aléatoires **[14]**.

Le SCA utilise un opérateur de mouvement cosinus sinusoïdal pour mettre à jour la position de chaque agent de recherche dans l'espace de solution par rapport à la meilleure solution en utilisant les formules ci-dessous **[14]** :

$$X_i^{t+1} = \begin{cases} X_i^t + r_1 * \sin(r_2) * |r_3 P_i^t - X_i^t|, & r_4 < 0.5 \\ X_i^t + r_1 * \cos(r_2) * |r_3 P_i^t - X_{ii}^t|, & r_4 < 0.5 \end{cases} \qquad \text{……..…(Equ. I.1)}$$

où

- X_i^t est la position de la solution courante en-t dimension en t-iteration,
- r1 / r2 / r3 sont des nombres aléatoires,
- P_i est la position du point de destination dans i-dimension,
- et || indique la valeur absolue.
- r4 est un nombre aléatoire dans [0,1]

❖ L'optimisation de loup gris

L'optimisation du loup gris (Grey Wolf Optimisation GWO) est une technique intelligente de l'essaim développée par Mirjalili et al., 2014, qui imite la hiérarchie de leadership des loups qui sont bien connus pour leur chasse de groupe **[15]**.

Le GWO imite le comportement de chasse et la hiérarchie sociale des loups gris. L'algorithme GWO (Pseudo code**)** est décrit brièvement avec les étapes suivantes **[15]** :

Initialiser la population des loups gris Gi (i = 1, 2, ..., n)
Initialiser a, A, et C
Calculer la fitness de chaque agent de recherche
Gα = le meilleur agent de recherche
Gβ= le deuxième meilleur agent de recherche
Gδ = le troisième meilleur agent de recherche
Tant que (t < Max nombre d'iterations)
Pour chaque agent de recherche
Mettre à jour la position courante de chaque agent de recherche
FinPour
Mettre à jour a, A, et C
Calculer la fitness pour tous agent de recherche
Mettre à jour Gα, Gβ, and Gδ
t=t+1
FinTantQue
retourner Gα

❖ L'algorithme culturel

En tant que système à double hérédité, l'algorithme culturel (cultural algorithm CA) a deux composantes de base : l'espace de la population et l'espace de croyance. Dans chaque génération, les individus de l'espace population sont d'abord évalués avec une fonction de performance **[16]**.

Une fonction d'acceptation est ensuite utilisée pour déterminer quels individus seront autorisés à mettre à jour l'espace de croyance **[16]**.

Les expériences de ces individus choisis sont ensuite ajoutées au contenu de l'espace de croyance via la fonction update. Ensuite, la connaissance de l'espace de croyance est autorisée à influencer la sélection des individus pour la génération suivante de la population à travers la fonction influence **[16]**.

Dans ce qui suit, il est décrit un tel cadre et le pseudo-code de base de l'algorithme culturel **[16]** :

Début
t = 0;
initialise Bt, Pt
Répeter
evaluaer Pt
Mettreàjour(Bt, accept(Pt))
générer(Pt, influence(B t))
t = t + 1;
selectionner Pt à partir Pt- 1
Jusqu'à (terminaison de la condition vérifiée)
Fin

Ici Pt représente la composante Population à l'instant t, et Bt pour l'Espace de Croyance à l'instant t. L'algorithme commence par initialiser à la fois la population et l'espace de croyance. Ensuite, il entre dans la boucle d'évolution jusqu'à ce que la condition de terminaison soit satisfaite.

- **Algorithme à évolution différentielle**

L'algorithme à évolution différentielle (DE pour Diffrential Evolution) est une branche de la programmation évolutive développé par Rainer Storn et Kenneth Price pour l'optimisation problèmes sur des domaines continus. En DE, la valeur de chaque variable est représentée par un nombre réel. Les avantages de DE sont sa structure simple, sa facilité d'utilisation, sa rapidité et sa robustesse **[17]**.

- **Algorithme de recherche par les organismes symbiotiques**

L'algorithme SOS (Symbiotic Organisms Search) est un nouvel algorithme métaheuristique développé par **[18]**. Il a été inspiré par la dépendance basée sur l'interaction observée parmi les organismes de la nature, qui sont connu sous le nom de symbiose.

Comme la plupart métaheuristique basée sur la population algorithmes, SOS a les caractéristiques suivantes **[3] [18]** :

(1) il utilise une population d'organismes qui contient des solutions candidates utilisées pour chercher la solution globale sur l'espace de recherche ;

(2) les opérateurs utilisent les solutions candidates pour guider le processus de la recherche ;

(3) il utilise un mécanisme de sélection pour préserver les meilleures solutions ;

(4) il nécessite la mise en place d'un contrôle commun de paramètres approprié tels que la taille de la population et le nombre maximal d'évaluations.

Cependant, contrairement à la plupart des algorithmes métaheuristiques qui ont des paramètres de contrôle (par exemple, AG a un taux de croisement et de mutation ; le PSO a un poids d'inertie, des facteurs cognitifs et des facteurs sociaux) **[18]**,

SOS ne nécessite aucun paramètre spécifique à l'algorithme. Ceci est considéré comme un avantage sur les algorithmes concurrents parce que SOS n'a besoin d'effectuer le réglage des paramètres **[18]**.

Le réglage incorrect lié à des paramètres spécifiques à l'algorithme peut augmenter le temps de calcul et produire des solutions optimales locales **[18]**.

Au début, pour un écosystème aléatoire une matrice d'écosystème aléatoire (population) est créée, chaque ligne représentant une solution candidate au problème **[18]**.

Le nombre d'organismes dans l'écosystème, la taille dite de l'écosystème est prédéterminé par l'utilisateur **[18]**.

Les rangées dans la matrice sont appelés organismes, chaque organisme virtuel représente une solution candidate au problème ou à l'objectif correspondant [**18]**.

La recherche commence après que l'écosystème initial a été généré.

Pendant le processus de recherche, chaque organisme bénéficie d'une interaction continue avec les trois phases différentes **[18]** :

• **Mutualisme** : un organisme développe une relation bénéfique à lui-même avec un autre - un exemple classique est l'interaction entre les abeilles et les fleurs ;

• **Commensalisme** : Le commensalisme, en biologie, est une forme de relations entre deux organismes d'espèces différentes où un organisme bénéficie de l'autre sans l'affecter- un exemple est la relation entre le poisson rémora et les requins ;

• **Parasitisme** : un organisme développe une relation bénéfique lui-même mais nuit à l'autre, un exemple est le plasmodium parasite, qui utilise sa relation avec le moustique anophèle à transfert entre hôtes humains.

Les trois phases sont adoptées à partir des symbioses les plus courantes utilisées par les organismes pour augmenter leur condition physique et leur avantage de survie à long terme.

Pendant l'interaction, celui qui reçoit un avantage évolue vers un organisme plus apte alors que celui qui est lésé périt.

Les mécanismes de mise à jour du meilleur organisme sont menés après une génération d'organismes a terminé ses trois phases. Les phases sont répétées jusqu'à ce que le critère d'arrêt soit atteint. L'algorithme représentant en résumé les bases étapes de la procédure d'optimisation SOS est comme suit **[18]** :

1 : Initialisation (écosystème initial, ensemble d'écosystèmes taille et itération maximale)

2 : **Pour** (compteur = 1 à l'itération maximale) **Faire**

3 : **Pour** (chaque organisme de l'écosystème) **Faire**

4 : Phase de mutualisme

5 : Phase de commensalisme

6 : Phase de parasitisme

7 : Mettre à jour le meilleur organisme

8 : **Fin pour**

9 : **Fin pour**

I.3 Conclusion

Comme synthèse de ce chapitre, nous pouvons signaler que les approches de résolution de problèmes combinatoires qui n'ont pas de solutions algorithmiques sont maitrisables avec ce panorama de métaheuristiques.

Il est à signaler que certain problèmes sont aussi solvables par une combinaison de ces techniques soit par hybridation séquentielle ou parallèle ou une intégration de l'une dans l'autre.

Une comparaison directe sur un problème est possible mais se prononcer sur un choix d'une méthode par rapport à une autre ne peut se faire que sur des critères de qualité du résultat à obtenir.

Dans le chapitre suivant, nous allons décrire un autre domaine qui fera l'objet de notre étude pratique, il s'agit du datamining.

Chapitre II

Etat de l’art sur le datamining

II.1 Introduction

Il y a plusieurs données qui circulent dans l'industrie de l'information en général et dans tout secteur économique ou administratif. Ces données n'ont aucune valeur cognitive ou économique tant qu'elles ne sont pas converties en informations utiles. Il est intéressant voir nécessaire d'analyser ce grand volume de données et d'en tirer des informations utiles et exploitables.

L'extraction de l'information n'est pas le seul procédé qu'il faut effectuer ; l'exploration de données implique aussi d'autres processus tels que le nettoyage de données, l'intégration de données, la transformation de données, l'exploration de données, l'évaluation de modèles et l'affichage de données.

A la terminaison de ces processus, il devient possible d'utiliser cette information dans de plusieurs applications comme la détection de la fraude, l'analyse du marché, le contrôle de la production, l'exploration scientifique, etc **[4]**.

Le Datamining représente la procédure d'extraction d'informations à partir d'énormes stock de données. En d'autres termes, le datamining est défini aussi comme l'exploration de données pour en extraire les connaissances [**19].**

II.2 Les différents modèles d'entrepôts de données pour le datamining

L'importance de présenter et de définir les deux concepts qui sont le data warehouse et le data Mart vient du fait que le premier représente l'endroit de stockage de données (entrepôt, grand dépôt), le deuxième concept représente une partie réduite en espace géographique du premier.

II.2.1 Le data warehouse

A cause de la concurrence dans le marché associé aux exigences spécifiques des clients et la rapidité du cycle de vie des produits, cela a poussé les producteurs à se comporter vis-à-vis du marché de manière anticipative et prédictive. Ils sont de plus en plus obligés à répondre aux attentes et bien connaitre leur clientèle. La connaissance de son processus de production, du comportement de ses clients, de ses fournisseurs est importante pour une survie de l'entreprise, car cela permetra de prévenir la situation sur les périodes à venir **[19].**

Actuellement, les entreprises disposent un stock de données assez grand. La raison de ce stockage vient du fait que l'infrastructure et les machines sont de faibles capacités alors que la quantité de données à traiter augmente énormément **[19]**.

Ces réservoirs de connaissance doivent être explorés pour interpréter le sens et de détecter les relations entre données, il faut aussi extraire des modèles expliquant leur comportement **[19].**

Pour cette raison, les chercheurs ont développés le modèle de grande quantité d'information appelé data warehouse (DW), qui est une manière de stockage regroupant, sous une forme homogène, toutes les données de l'entreprise sur une longue période. Cette approche a mis à la disposition aux décideurs des nouvelles perspectives en termes d'extraction de connaissances à travers le processus de datamining **[19].**

II.2.1.1 Définition et historique du data warehouse

Le data warehouse (DW) est un rassemblement de données concernant un domaine qui sont intégrées, non volatiles et historisées, représentées de manières à faciliter un traitement automatique comme l'aide à la décision **[19]**.

Le concept de data warehouse (entrepôt de données) a été proposé pour la première fois en 1990. Plusieurs chercheurs disaient que cela n'est autre que la transformation d'un concept déjà ancien qui est l'infocentre **[19]**.

Mais l'économie actuelle les entreprises sont en face à une concurrence de plus en plus pertinente, des clients de plus en plus exigeants, dans un contexte organisationnel de plus en plus complexe et dynamique **[19]**.

Pour faire face aux nouveaux défis économiques, l'entreprise doit anticiper et prédire son fonctionnement avec ses acteurs. L'anticipation ou la prédiction ne peut être efficace qu'en se basant sur de l'information pertinente. Cette information est à la disposition de toute entreprise qui possède d'un capital de données gérées par ses systèmes opérationnels et qui peut en acquérir d'autres auprès de fournisseurs externes. Malheureusement, les données sont devenues surabondantes, non organisées dans une vision futuriste décisionnelle et distribuées dans de plusieurs systèmes hétérogènes **[19].**

Ces données représentent un gisement d'informations, pour cette raison il devient vital de rassembler et d'homogénéiser les données pour analyser les indicateurs et les facteurs pertinents pour faciliter les prises de décisions postérieures **[19].**

Pour répondre à ces besoins, le nouveau rôle de la science de données est de définir et d'intégrer une architecture qui sera le fondement des applications décisionnelles, il s'agit alors du data warehouse (DW) **[19].**

II.2.1.2 Intérêt d'un data warehouse

L'objectif d'une entreprise à travers la construction d'un système décisionnel est améliorer sa performance. Pour cela, elle doit décider et anticiper ses actions dans le marché par rapport à l'information disponible et capitaliser sur ses expériences **[19]**.

Depuis plusieurs années, une importante masse d'informations est stockée dans les entreprises. Les systèmes d'information ont pour mission de garder la trace d'événements de manière fiable et sans erreurs **[19].**

En même temps, les entreprises sont devenues conscient de la valeur du capital informationnel qu'elle possède. Au delà de ce que le traitement automatique leur apporte en terme fonctionnel, elles donnent plus d'intérêt à ce qu'elle pourrait apporter au niveau du contenu informationnel **[19].**

Cette nouvelle conception de porter de l'importance et de voir un système d'information comme l'importante dimension pour se placer dans le devant de la scène économique et être en mesure de se comparer de manière qualitative. Cette situation appelée aussi veille stratégique est devenue un enjeu économique de première catégorie pour les entreprises et qui est devenue une question de survie **[19].**

Il est donc essentiel qu'une entreprise prend en charge son devenir, cela repose d'une part sur l'estimation exacte de l'entrée financière suite aux placements dans le marché. Ces entrées sont estimées avec une vision des chutes des rentes monétaires au lieu de voir ce que les investissements ont donné comme valeur gagnée. Ceci peut être apprécier à travers une étude sur les produits achetés, les périodes de consommations, le nombre acheté etc. Cette étude permettra aux décideurs de dresser une carte de consommation quotidienne, hebdomadaire, mensuelle et même annuelle **[19].**

II.2.2 Le Data Mart

Le data mart est une vision restreinte du modèle du data warehouse. La raison de son apparition réside sur le désire d'un client de s'adresser vers une entité de vente réduite au lieu de s'orienter vers un entrepôt de données. La problématique d'un data warehouse vient du fait qu'il inclus une grande variété de produit avec des données extrêmement complexe **[19]**.

Pour cela, les chercheurs ont proposé le modèle du Data Mart (DM) qui a pour objectif de réduire cette complexité surtout technique et informatique en prenant le client comme le point de départ de toute transaction commerciale **[19]**.

L'avantage d'un DM revient à sa conception qui est réduite en complexité et d'un valeur économique moins couteuse par rapport à un data warehouse. Il est beaucoup plus destiné à une communauté urbaine petite qu'un entrepôt de données dont le rôle est d'alimenter parfois un pays **[19].**

Le DM est donc une vision restreinte d'un espace géographique commercial destiné à répondre aux besoins directe d'une population d'une localité. Le rôle d'un DM est similaire que le DW, la différence réside donc sur une vision restreinte en matière d'acheminement de produit et d'aide à la décision **[19].**

Le DM est représenté conceptuellement comme une simple petite base de données contenant un nombre réduite de fichiers et de données, il est manipulable par des langage d'interrogation de base de données comme SQL.

Pour mettre en claire les différences majeures entre le DW et le DM, nous dressons dans le tableau suivant les principales caractéristiques entre ces deux concepts **[19].**

Tableau II.1 Différence entre data warehouse et data mart

	Data Warehouse	**Data Mart**
Cible utilisateur	Toute l'entreprise	Département
Implication du service informatique	Elevée	Faible ou moyen
Base de données d'entreprise	SQL type serveur	SQL milieu de gamme, bases multidimensionnelles
Modèles de données	A l'échelle de l'entreprise	Département
Champ applicatif	Multi sujets, neutre	Quelques sujets, spécifique
Sources de données	Multiples	Quelques unes
Stockage	Base de données	Plusieurs bases distribuées
Taille	Centaine de GO et plus	Une à 2 dizaines de GO
Temps de mise en place	9 à 18 mois pour les 3 étapes	6 à 12 mois (installation en plusieurs étapes)
Coût	> 6 millions d'euro	500.000 à 3 millions d'euro
Matériel	Unix	NT, petit serveur Unix

II.3 Généralités sur le datamining

La croissance industrielle rapide a engendré d'énorme quantité de données, collectées et stockées dans de vastes et nombreux dépôts de données qui a dépassait beaucoup notre capacité de compréhension sans outils puissants **(Figure II.1) [20]**. Par conséquent, les données collectées dans les référentiels de données volumineux deviennent des «tombes de données» - archives de données qui sont rarement visités. Par conséquent, des décisions importantes sont souvent prises sur les données riches en informations stockées dans des référentiels de données, mais plutôt sur une décision intuitive, tout simplement parce que le décideur n'a pas les outils pour extraire les précieuses connaissances intégrées dans les vastes quantités de données **[20]**.

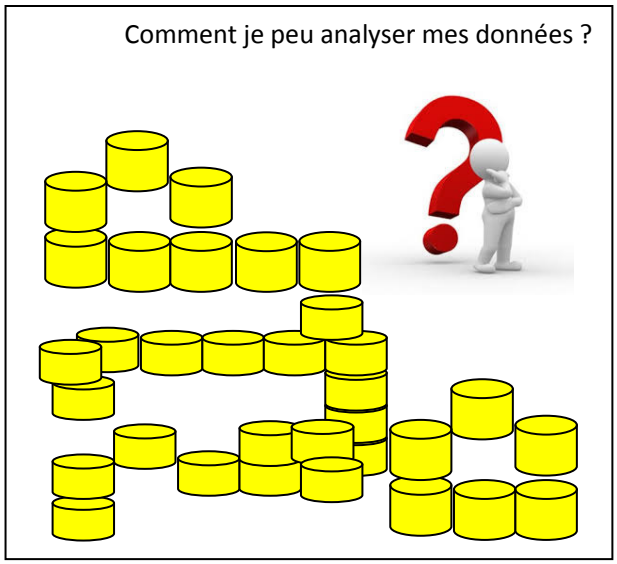


Figure II.1 Nous sommes riches en données, mais pauvres en informations [20].

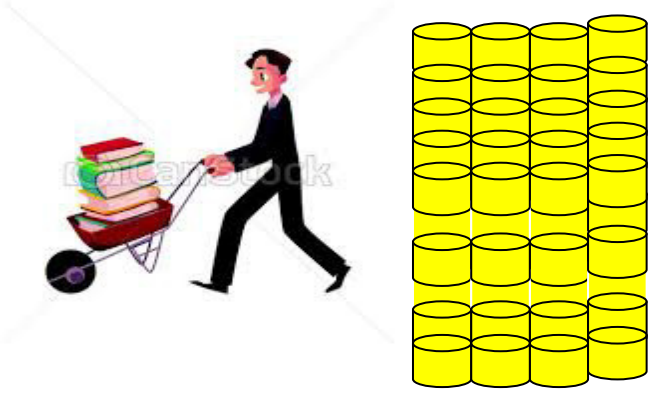


Figure II.2 Extraction de données: recherche de connaissances (Modèles intéressants) dans les données [20].

La fouille de données est une phase dans le processus de découverte des connaissances. C'est une substance essentielle qui intervient dans le processus de découverte de connaissance.

Ce processus est décrit en détail dans la Figure II.3. Il est sous la forme d'un cycle d'étapes s'exécutant dans un ordre itératif contenant les étapes suivantes **[20]:**

1-Collecte des données : objectif de cette étape est combiné plusieurs sources de données, parfois avec hétérogénéité dans une seule base de données **[20] [21]**.

2-Nettoyage des données : dans cette phase on s'intéresse à la régularisation des données qui consiste à éliminer les attributs ont des valeurs invalides ou sans valeurs. Dans le contexte, cela est appelé le bruit (données non conforment) **[20] [22]**.

3-Sélection des données : cela consiste à choisir de la base de données dans le cadre d'une tâche du datamining que les données les plus intéressantes **[20] [23]**.

4-Transformation des données : c'est une opération qui a pour objectif la transformation des structures des attributs pour être adéquates à la procédure d'extraction des informations **[20] [24]**.

5-Extraction des informations (Datamining): l'objectif de cette phase est l'utilisation de quelques algorithmes du Data Mining sur les données produites par l'étape précédente (*Knowledge Discovery in Databases*, ou KDD) **[20] [24]**.

6-Visualisation des données : l'utilisation des techniques de visualisation (histogramme, camembert, arbre, visualisation 3D) pour exploration interactive de données (la découverte des modèles de données) **[20] [24]**.

7-Evaluation des modèles : l'identification des modèles strictement intéressants en se basant sur des mesures données **[20] [24]**.

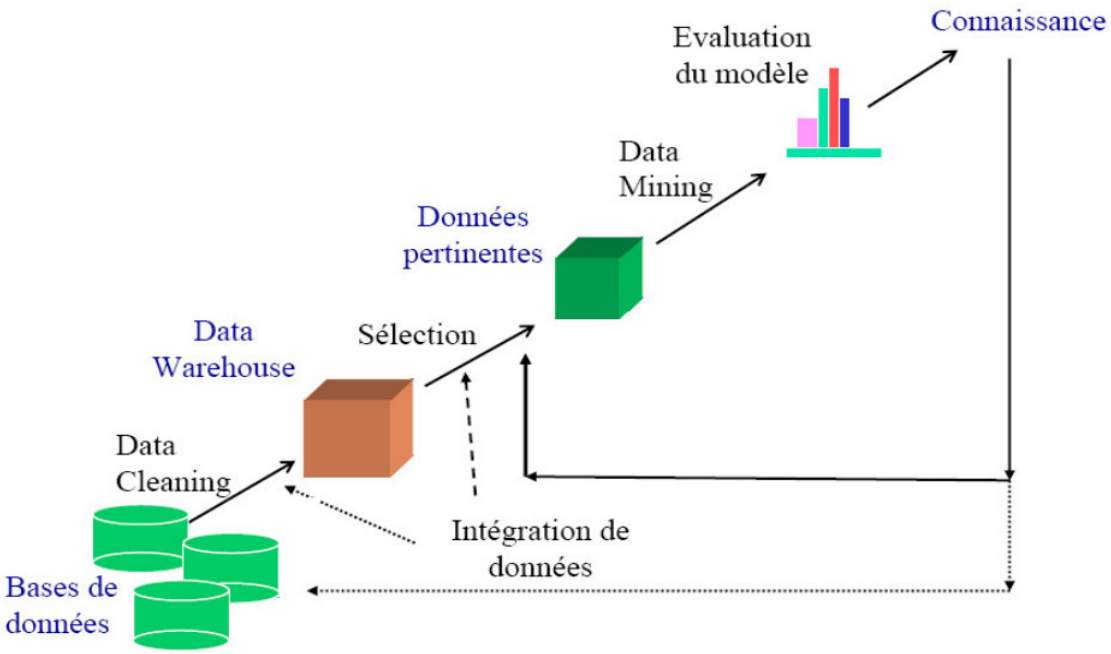


Figure II.3 Les étapes du processus de l'extraction de connaissances **[20]**.

II.4 Les techniques du datamining

Dans le domaine du datamining plusieurs approches sont exploitées. La valeur scientifique de ces approches a eu un impact très apprécié à cause de leur grande utilisation en particulier avec l'exploitation de grand volume de données.

D'autre part, ces approches ont fait preuve de maturité applicative suite à la mise à la disposition d'environnement graphique mettant en valeur les résultats obtenus. Ces résultats sont nettement et facilement exploités par les décideurs pour la prise de décision.

A cet effet, plusieurs modèle de sont exploités qui sont **[25]** :

Réseaux neuraux artificiels – c'est un formalisme modélisant le système neuronal humain dans le processus d'apprentissage **[25]**.

Arbres de décision – ce sont des structures en forme d'arbre qui représente les ensembles de décisions. Ces décisions génèrent des règles pour la classification d'un ensemble de données **[25]**.

Algorithmes génétiques – c'est un formalisme inspiré de la génétique humaine, il est utilisé pour l'optimisation en exploitant les trois principales étapes à savoir : sélection, croisement et mutation **[25]**.

Le voisin le plus proche – c'est une approche qui réalise une classification d'une ensemble de données en sous classes sur la base du principe « le plus proche voisin » (en valeur, etc...) **[25]**.

II.5 Les taches du datamining

Le genre de connaissances à extraire fait référence au type de fonction à exploiter. La *fonction descriptive* exploite des propriétés générales des données dans la base de données. L'exploration de données concerne la forme de modèles pouvant être extraits. Sur la base du type de données à extraire, nous trouvons deux catégories des fonctions impliquées dans le datamining qui influencent l'objectif final décisionnel. Beaucoup de problèmes scientifiques, socio-économiques peuvent être exprimés en termes de modèles à obtenir, ils sont regroupés comme suit sous six formes possibles **[19][20]** :

- La classification
- L'estimation
- La prédiction
- Le groupement par similitude
- L'analyse des clusters
- La description
- L'optimisation.

Une première classification de ses approches de datamining donne une première classification des trois premières sous l'angle de technique supervisée. Cette catégorie utilise un ensemble de données pour les transformer en un modèle représentant une variable particulière considérer comme objectif suite à l'exploitation de ses mêmes données. En e qui concerne la deuxième classe de méthodes basée sur la similitude et l'analyse des clusters, ces deux groupes sont de type non-supervisées ayant comme objectif la définition et la recherche d'une relation entre toutes les variables **[19].**

La description appartient à ces deux catégories de tâche, elle est vue comme une tâche supervisée et non-supervisée en même temps **[19].**

II.5.1 La classification

Le processus de classification est un mécanisme qui comme dès le jeune âge de l'être l'humain. Il le pratique à travers des questions pour comprendre l'appartenance d'un objet, d'un animal d'une personne à sa catégorie ou à sa famille **[19]**.

La définition couramment exploitée est:

« La classification consiste à examiner des caractéristiques d'un élément nouvellement présenté afin de l'affecter à une classe d'un ensemble prédéfini. » **[21].**

La classification permet de créer des classes d'individus ou d'objet pour l'exploiter dans le processus d'apprentissage, comme classe nous avons : homme / femme, oui / non, rouge / vert / bleu, ..

Par **exemple** [**20]** :

- ✓ la classe d'oiseau contient : canari, pigeon, moineau
- ✓ la classe poisson : sardine, requin, baleine

Les techniques les plus appropriées à la classification sont :

- Les arbres de décision,
- Le raisonnement basé sur la mémoire,
- Eventuellement l'analyse des liens.

II.5.2 L'estimation

Contrairement à la classification, le résultat d'une estimation permet d'obtenir une variable continue. Celle-ci est obtenue par une ou plusieurs fonctions combinant les données en entrée. Le résultat d'une estimation permet de procéder aux classifications grâce à un barème **[26].**

Exemple [26] : on peut estimer le revenu d'un ménage selon divers critères (type de véhicule et nombre, profession ou catégorie socioprofessionnelle, type d'habitation, etc.).

Il sera ensuite possible de définir des tranches de revenus pour classifier les individus.

Un des intérêts de l'estimation est de pouvoir ordonner les résultats pour ne retenir si on le désire que les n meilleures valeurs. Cette technique sera souvent utilisée en marketing, combinée à d'autres, pour proposer des offres aux meilleurs clients potentiels. Enfin, il est facile de mesurer la position d'un élément dans sa classe si celui ci a été estimé, ce qui peut être particulièrement important pour les cas limitrophes. La technique la plus appropriée à l'estimation est : le réseau de neurones.

II.5.3 La prédiction

La prédiction ressemble à la classification et à l'estimation mais dans une échelle temporelle différente. Tout comme les tâches précédentes, elle s'appuie sur le passé et le présent mais son résultat se situe dans un futur généralement précisé. La seule méthode pour mesurer la qualité de la prédiction

est d'attendre. L'analyse de Régression est généralement utilisée pour la prédiction. La prédiction peut aussi être utilisée pour l'identification de tendances de distribution basées sur des données disponibles **[20][26]**. Les techniques les plus appropriées à la prédiction sont **[20][26]**:

- L'analyse du panier de la ménagère
- Le raisonnement basé sur la mémoire
- Les arbres de décision
- les réseaux de neurones

II.5.4 Le regroupement par similitudes

Le regroupement par similitudes consiste à grouper les éléments qui vont naturellement ensembles. La technique la plus appropriée au regroupement par similitudes est l'analyse du panier de la ménagère **[26]**.

Les modèles fréquents sont les modèles qui se produisent fréquemment dans les données transactionnelles. Voici la liste des types de motifs fréquents **[26]**:

- ✓ *Ensemble d'articles fréquent* - Il s'agit d'un ensemble d'articles qui apparaissent fréquemment ensemble, par exemple, du lait et du pain.
- ✓ *Sous-séquence fréquente*: une séquence de motifs fréquents, comme l'achat d'une caméra, est suivie d'une carte mémoire.
- ✓ *Sous-structure fréquente* - La sous-structure fait référence à différentes formes structurelles, telles que des graphiques, des arbres ou des treillis, qui peuvent être combinées avec des ensembles d'items ou des sous-séquences.

II.5.5 L'analyse des clusters

Le cluster fait référence à un groupe d'objets similaires. L'analyse de cluster fait référence à la formation d'un groupe d'objets très similaires les uns aux autres mais très différents des objets des autres clusters. En d'autres termes, l'analyse des clusters consiste à segmenter une population hétérogène en sous populations homogènes. Contrairement à la classification, les sous populations ne sont pas préétablis. La technique la plus appropriée à la clustérisassions est l'analyse des clusters **[26]**.

II.5.6 La description

C'est souvent l'une des premières tâches demandées à un outil de Datamining. On lui demande de décrire les données d'une base complexe.

Cela engendre souvent une exploitation supplémentaire en vue de fournir des explications. La technique la plus appropriée à la description est l'analyse du panier de la ménagère **[25]**.

II.5.7 L'optimisation

Pour résoudre de nombreux problèmes, il est courant pour chaque solution potentielle d'y associer une fonction d'évaluation **[25]**.

Le but de l'optimisation est de maximiser ou minimiser cette fonction. Quelques spécialistes considèrent que ce type de problème ne relève pas du Datamining **[25]**.

La technique la plus appropriée à l'optimisation est le réseau de neurones **[25]**.

II.6 Domaines d'application du DATAMINING

Le processus de l'exploration de données est un processus d'extraction d'informations à partir d'énormes ensembles de données.

Les informations ou les connaissances ainsi extraites peuvent être utilisées pour l'une des applications suivantes **[20] [27]** :

- analyse et gestion du marché ;
- analyse corporative et gestion des risques ;
- détection de fraude ;
- la télécommunication ;
- bio-informatique et la biotechnologie.

En plus de ces domaines ; il aussi possible d'exploiter l'exploration de données dans les domaines du contrôle de la production, les réseaux sociaux pour l'analyse des opinions, la fidélisation des clients, de l'exploration scientifique, du sport, de l'astrologie et de l'Internet **[20] [27]**.

II.6.1 Analyse et gestion du marché

Voici la liste des différents domaines du marché où l'exploration de données est utilisée **[20] [27]** :

- **Profilage client** : dans ce domaine, l'exploration de données permet de déterminer quel type de personne achète quel type de produits.

- **Identification des besoins des clients** : l'exploration de données très utile pour connaitre exactement les meilleurs produits pour différents clients. Son principe est basé sur la prédiction pour trouver les facteurs susceptibles d'attirer de nouveaux clients.

- **Analyse croisée des marchés** : l'exploration de données effectue des associations / corrélations entre les ventes de produits.

- **Target Marketing** : l'exploration de données permet de déterminer des groupes de clients modèles qui partagent les mêmes caractéristiques telles que les intérêts, les habitudes de dépenses, les revenus, etc.

- **Détermination du modèle d'achat du client** : l'exploration de données aide à déterminer le modèle d'achat du client.

- **Fournir des informations récapitulatives** : l'exploration de données est capable de mettre à la disposition des décideurs différents rapports récapitulatifs multidimensionnels.

II.6.2 Analyse corporative et gestion des risques

L'exploration de données est utilisée dans les domaines suivants du secteur des entreprises **[20] [27]** :

- **Planification financière et évaluation des actifs :** il s'agit de l'analyse et de la prévision des flux de trésorerie et de l'analyse des réclamations éventuelles pour évaluer les actifs.

- **Planification des ressources :** pour ce type de traitement, l'objectif est faire une synthèse qui permettra de comparer les ressources et les dépenses.

- **Concurrence :** l'objectif dans ce cas d'utilisation consiste à surveiller les concurrents et les orientations du marché.

II.6.3 Détection de fraude

L'exploration de données fait l'objet d'utilisation dans les domaines des services de cartes de crédit bancaire et des télécommunications pour détecter les fraudes.

Dans le domaine de la téléphonie mobile, il est très efficace pour vérifier les appels frauduleux, il aide à trouver la destination de l'appel, la durée de l'appel, l'heure du jour ou de la semaine, etc. **[20] [27]**.

II.6.4 Le datamining dans la télécommunication

Dans le domaine de la télécommunication la technique de datamining est nécessaire pour beaucoup de taches, comme exemple **[20][27] :**

- Analyse des achats de services de télécommunications.
- Prédiction de modèles d'appels téléphoniques.
- Gestion des ressources et de trafic réseau.
- Automatisation de la gestion du réseau et de la maintenance pour le diagnostique et la prise en charge des problèmes de transmission du réseau.

II.6.5 Le data mining dans la bio-informatique et la biotechnologie

La tendance actuelle dans le domaine de la résolution de problèmes complexes s'oriente vers le monde des vivants. La bio-informatique est un axe de recherche qui s'intéresse à inspirer des solutions ou des algorithmes pour ce type de problèmes. Cette approche de résolution est basée sur des principes de la biologie couplée avec la technologie d'informations.

Certaines applications du datamining portent sur **[20][27]** :

- la prédiction les structures de différentes protéines.
- la détermination de la complexité des structures de plusieurs médicaments.

II.7 Travaux à base de la symbiotique pour le datamining

Dans ce paragraphe nous allons établir une présentation des quelques travaux exploitant la technique de symbiotique pour le datamining.

II.7.1 Algorithme d'optimisation de recherche à base d'organismes symbiotiques hybrides pour la planification de tâches dans le cloud computing

Le cloud computing a attiré une attention considérable de la communauté de la recherche en raison du taux de migration rapide des services de technologie de l'information vers son domaine. Les progrès de la technologie de virtualisation ont rendu le cloud computing très populaire grâce au déploiement plus facile des services d'application **[28].**

Les tâches sont soumises à des centres de données cloud pour être traitées au fur et à mesure. La planification des tâches est l'un des défis de recherche importants dans l'environnement de cloud computing. Il a été démontré que la formulation actuelle des problèmes d'ordonnancement des tâches

est NP-complet, par conséquent trouver la solution exacte en particulier pour les grandes tailles de problèmes est intraitable. La fonctionnalité hétérogène et dynamique des ressources cloud rend la planification optimale des tâches non triviale **[28].**

Par conséquent, des algorithmes efficaces d'ordonnancement des tâches sont nécessaires pour une utilisation optimale des ressources. Symbiotic Organisms Search (SOS) a été montré pour fonctionner de manière compétitive avec Particle Swarm Optimization (PSO). Le but de cette étude est d'optimiser la planification des tâches dans l'environnement de cloud computing sur la base d'un SOS à recuit simulé (SA) afin d'améliorer le taux de convergence et la qualité de la solution de SOS. L'algorithme SOS a une forte capacité d'exploration globale et utilise moins de paramètres **[28].**

II.7.2 Une nouvelle recherche sur les organismes symbiotiques à objectifs multiples pour un problème de compromis temps-coût-utilisation du travail

Cette recherche présente un nouvel algorithme d'optimisation multiple MOSOS. MOSOS est appliqué pour résoudre le problème de compromis de décalage temps-coût-utilisation. La performance du modèle est démontrée dans les résultats expérimentaux. Test statistique trouvé MOSOS pour fournir de meilleures solutions par rapport aux autres méthodes **[29]**.

Les quarts de travail multiples sont couramment utilisés dans les projets de construction pour répondre aux exigences du projet. Néanmoins, les quarts de soir et de nuit augmentent le risque d'événements indésirables et doivent donc être utilisés dans toute la mesure du possible. L'optimisation du compromis entre la durée du projet (temps), le coût du projet et l'utilisation des horaires de travail du soir et de nuit, tout en conservant toute la logique du travail et les contraintes de disponibilité des ressources est nécessaire pour améliorer le succès global du projet.

Dans cette étude, une nouvelle approche appelée "recherche multi-objectif sur les organismes symbiotiques" (MOSOS) pour résoudre plusieurs problèmes de travail est introduite **[29]**.

L'algorithme MOSOS est de nouvelles techniques d'optimisation multi-objectifs basées sur des méta-heuristiques inspirées des stratégies d'interaction symbiotique que les organismes utilisent pour survivre dans l'écosystème **[29]**.

II.7.3 Planification des réseaux de brouillard avec optimisation Knapsack by Symbiotic Organismes Recherche

Internet des objets comme un concept utilise un capteur sans fil sur les réseaux qui ont des limitations de puissance, de stockage et de retard lors du traitement et de l'envoi de données dans le

nuage. Brouillard informatique est une extension des services de cloud pour réduire la latence et le trafic **[30]**.

La planification est la question NP-difficile dans l'informatique du brouillard. en raison de la proximité des capteurs et des nuages qui sont capables de traitement de puissance et sont bénéfiques pour les algorithmes de gestion des ressources. Ce travail propose un ordonnancement basé sur le knapsack optimisé par l'algorithme SOS simulée dans iFogsim (logiciel de simulation) **[30]**.

II.8 Conclusion

Le Datamining et devenu de nos jour un enjeu scientifique et économique de grande importance.

Son objectif consiste à extraire d'un entrepôt de données (Data Warehouse) deux types de connaissances : l'une, explicative des résultats obtenus par l'analyse multidimensionnelle ou explicative d'hypothèses relatives au contenu informationnel du data warehouse, l'autre, nouvelle, porteuse éventuellement de nouvelles possibilités d'action.

Le terme « mining » fait aussi référence à d'autre type d'analyse comme le texte-mining, l'image mining…

Dans ce chapitre nous avons dressé un état de l'art portant sur les concepts en étroite liaison avec le datamining, les approches existantes pour sa mise en œuvre ainsi que les étapes composant le processus lui-même.

Nous avons présenté aussi des travaux mettant en valeur l'intérêt du couplage entre le datamining et l'approche principale de notre mémoire à savoir la symbiotique.

Dans le prochain chapitre nous présenterons notre approche de modélisation du datamining à base de la symbiotique comme solution d'optimisation du processus.

Chapitre III : Evaluation de la symbiotique

III.1 Introduction

De nos jours le volume de données personnel et d'entreprise est en nette progression fulgurante. L'intérêt primordial dans le traitement de ses données pour une prise de décision devient un enjeu vital notamment en matière d'extraction de connaissances utiles.

Pour cette optimisation d'extraction de connaissances, nous nous sommes orienté vers une métaheuristique très efficace pour la résolution du problème d'optimisation, il s'agit de la symbiotique qui fait l'objet de notre principale contribution.

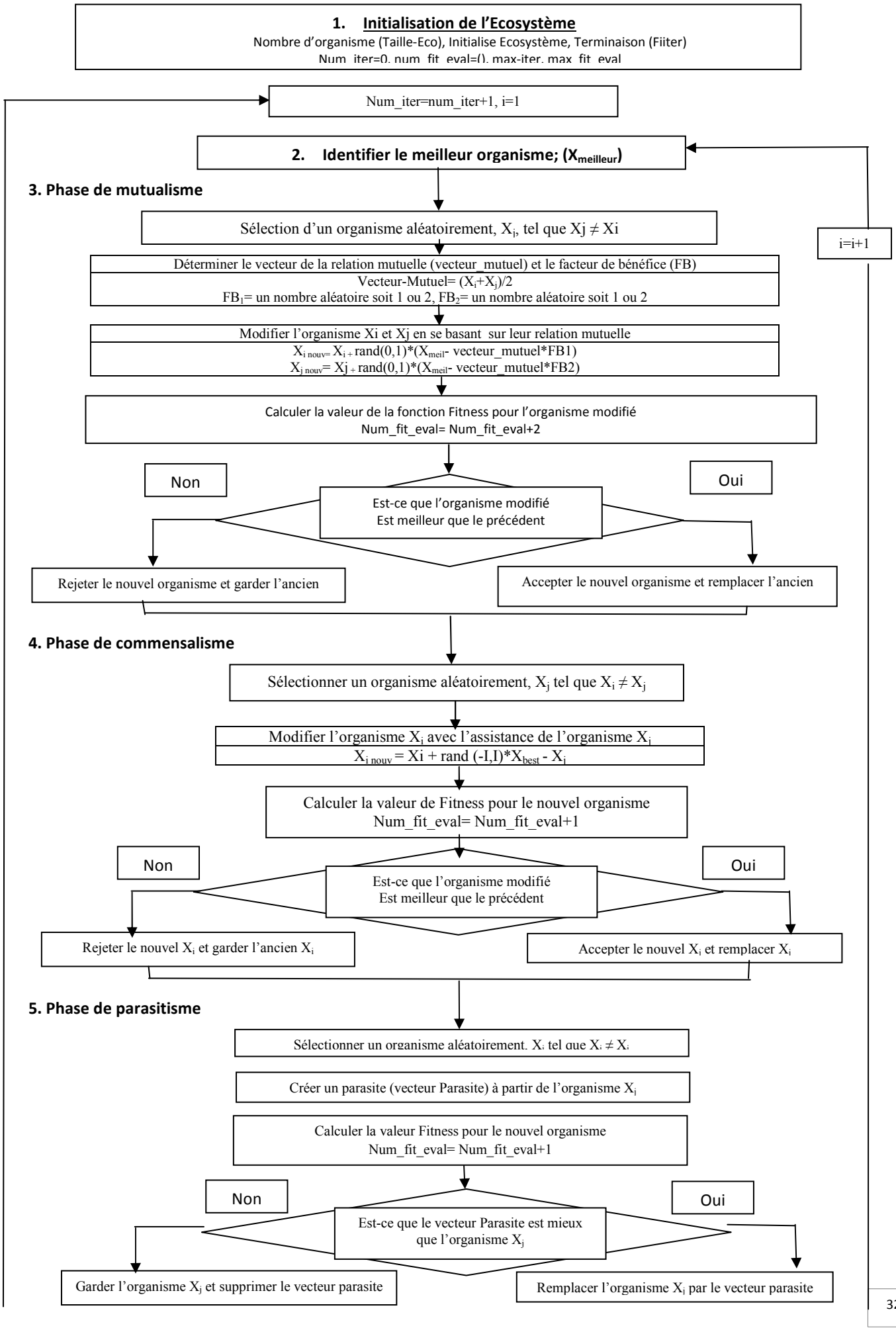
A cet effet, un système de datamining commence à s'imposer pour répondre aux besoins d'une décision qui oriente et optimise l'avenir ou le future d'entreprise.

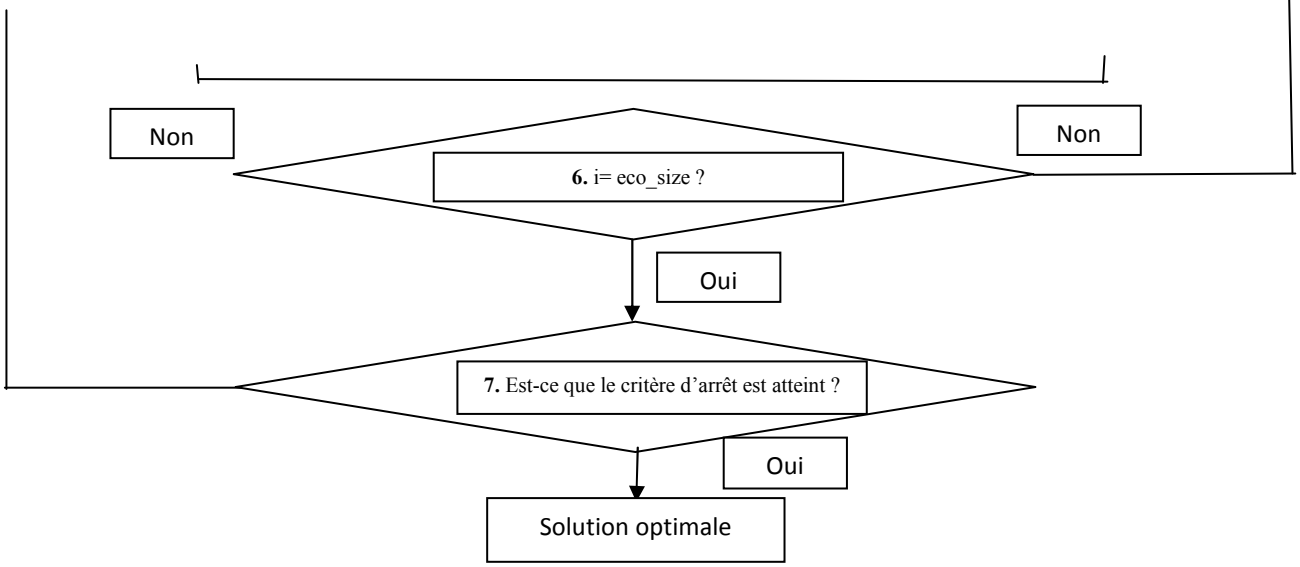
L'objectif de ce chapitre est de développer d'abord une modélisation conceptuelle de l'approche symbiotique en détaillant son fonctionnement et ses différents composants, par la suite nous évaluons notre proposition par des expériences.

III.2 Test de fonctionnement de la symbiotique

La symbiotique possède un fonctionnement spéciale par rapport aux autres métaheurisqtiques. Cette spécificité réside dans le type de l'interaction entre les entités définissant le comportement global du système.

Dans ce qui suit, nous présentons l'organigramme fonctionnel de la technique symbiotique.


1. Initialisation de l'Ecosystème
Nombre d'organisme (Taille-Eco), Initialise Ecosystème, Terminaison (Fiiter)
Num_iter=0, num_fit_eval=(), max-iter, max_fit_eval
Num_iter=num_iter+1, i=1
2. Identifier le meilleur organisme; (Xmeilleur)
3. Phase de mutualisme
Sélection d'un organisme aléatoirement, Xj, tel que Xj ≠ Xi
i=i+1
Déterminer le vecteur de la relation mutuelle (vecteur_mutuel) et le facteur de bénéfice (FB)
Vecteur-Mutuel= (Xi+Xj)/2
FB1= un nombre aléatoire soit 1 ou 2, FB2= un nombre aléatoire soit 1 ou 2
Modifier l'organisme Xi et Xj en se basant sur leur relation mutuelle
Xi nouv= Xi + rand(0,1)*(Xmeil- vecteur_mutuel*FB1)
Xj nouv= Xj + rand(0,1)*(Xmeil- vecteur_mutuel*FB2)
Calculer la valeur de la fonction Fitness pour l'organisme modifié
Num_fit_eval= Num_fit_eval+2
Non
Est-ce que l'organisme modifié
Est meilleur que le précédent
Oui
Rejeter le nouvel organisme et garder l'ancien
Accepter le nouvel organisme et remplacer l'ancien
4. Phase de commensalisme
Sélectionner un organisme aléatoirement, Xj tel que Xi ≠ Xj
Modifier l'organisme Xi avec l'assistance de l'organisme Xj
Xi nouv = Xi + rand (-I,I)*Xbest - Xj
Calculer la valeur de Fitness pour le nouvel organisme
Num_fit_eval= Num_fit_eval+1
Non
Est-ce que l'organisme modifié
Est meilleur que le précédent
Oui
Rejeter le nouvel Xi et garder l'ancien Xi
Accepter le nouvel Xi et remplacer Xi
5. Phase de parasitisme
Sélectionner un organisme aléatoirement, Xj tel que Xi ≠ Xj
Créer un parasite (vecteur Parasite) à partir de l'organisme Xi
Calculer la valeur Fitness pour le nouvel organisme
Num_fit_eval= Num_fit_eval+1
Non
Est-ce que le vecteur Parasite est mieux
que l'organisme Xj
Oui
Garder l'organisme Xj et supprimer le vecteur parasite
Remplacer l'organisme Xi par le vecteur parasite



Organigramme de la symbiotique [31]

III.2.1 Tableau des expressions des fonctions de test utilisées

Dans le tableau III.1 nous dressons une liste des différentes fonctions utilisées pour le teste de fonctionnement.

Cette liste de fonction nous a permis de vérifier la robustesse et la performance de la solution algorithmique proposée concernant la métaheuristique symbiotique.

L'ensemble des fonctions est divisée en 3 catégories à savoir :

- fonction unimodale à haute dimension : F1 au F7 permet de tester l'exploitation des méthodes pour l'obtention des meilleures solutions ;
- fonction multimodale à haute dimension : F8 au F13
- et fonction multimodale à faible dimension : F14 au F23

Pour tester l'exploration des méthodes de l'espace de recherche, nous avons trois classes de fonctions.

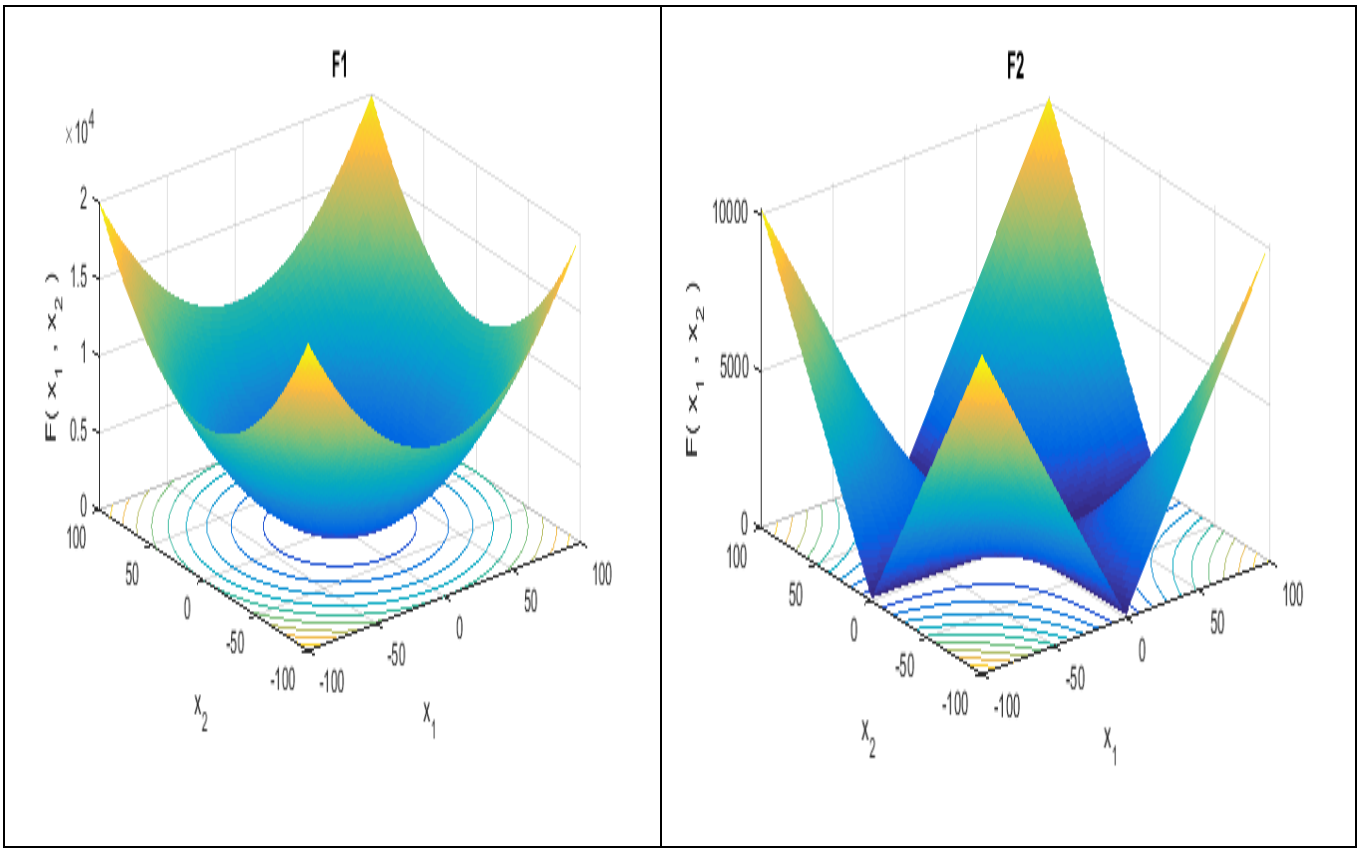
Le Tableau III.1 Liste des fonctions de test utilisées **[18][32]**

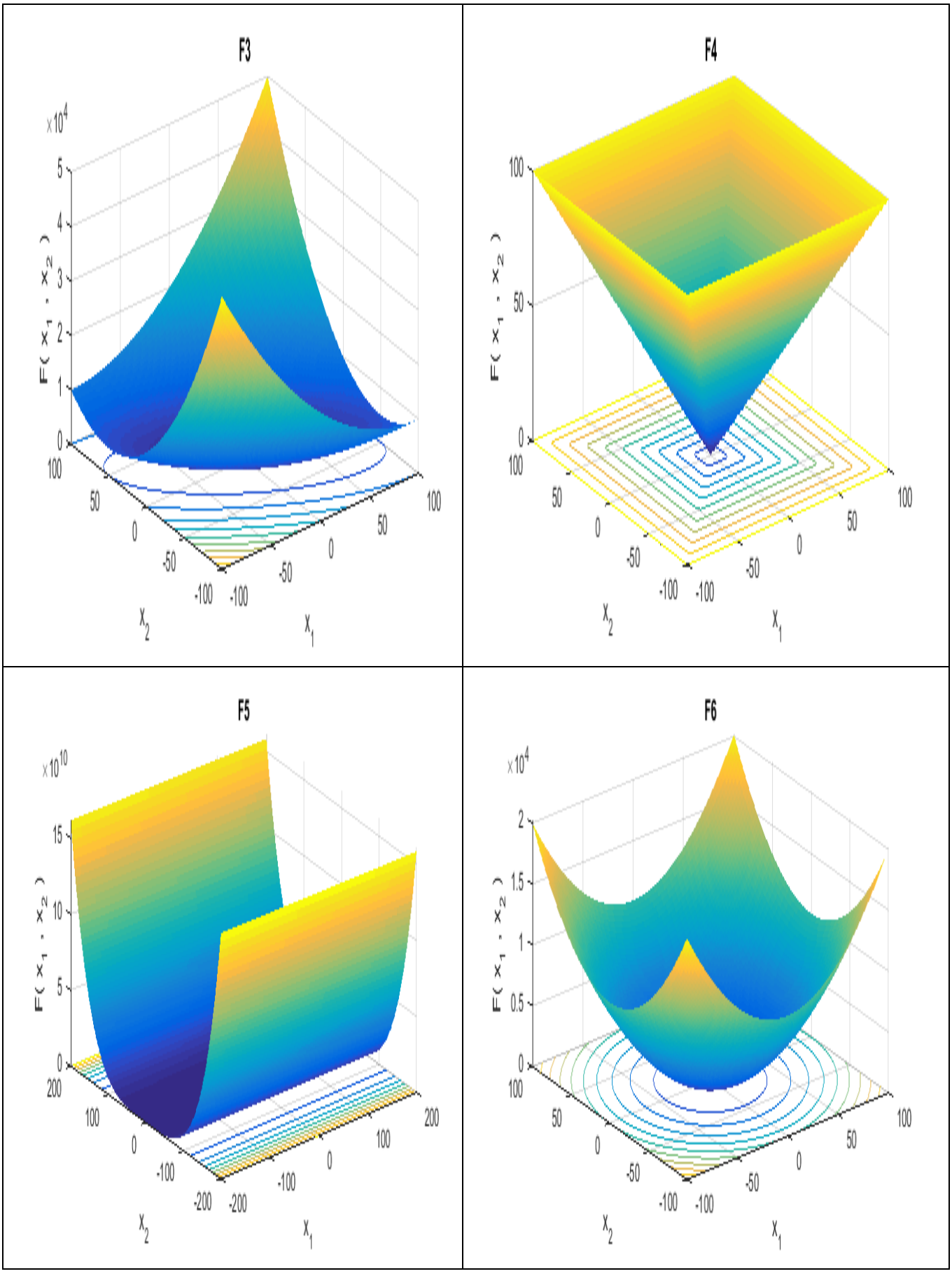
Fonctions de Teste		**Dimension**	**Intervalle**	**Maximum Iteration**
Fonctions unimodales à haute dimension	$F_{01} = \sum_{i=1}^{n} x_i^2$	30	[−100, +100]	500
	$F_{02} = \sum_{i=1}^{n} \lvert x_i \rvert + \prod_{i=1}^{n} \lvert x_i \rvert$	30	[−10, +10]	950
	$F_{03} = \sum_{i=1}^{n} \left(\sum_{j=1}^{i} x_j \right)^2$	30	[−100, +100]	500
	$F_{04} = max_i \{ \lvert x_i \rvert, 1 \le i \le D \}$	30	[−100, +100]	1000
	$F_{05} = \sum_{i=1}^{D-1} \left[100 (x_{i+1} - x_i^2)^2 + (x_i - 1)^2 \right]$	30	[−30, +30]	8000
	$F_{06} = \sum_{i=1}^{D} ([x_i + 0.5])^2$	30	[−100, +100]	15
	$F_{07} = \sum_{i=1}^{D} i x_i^4 + random\ [0,1)$	30	[−1.28, +1.28]	1500
Fonctions multimodales à haute dimension	$F_{08} = \sum_{i=1}^{D} -x_i \sin\left(\sqrt{\lvert x_i \rvert} \right)$	30	[−500, +500]	1500
	$F_{09} = \sum_{i=1}^{D} [x_i^2 - 10 \cos(2\pi x_i) + 10]$	30	[−5.12, +5.12]	40
	$F_{10} = -20 \exp\left(-0.2 \sqrt{\frac{1}{D} \sum_{i=1}^{D} x_i^2} \right) - \exp\left(\frac{1}{D} \sum_{i=1}^{D} \cos(2\pi x_i) \right) + 20 + e$	30	[−32, +32]	60
	$F_{11} = \frac{1}{4000} \sum_{i=1}^{D} x_i^2 - \prod_{i=1}^{D} \cos\left(\frac{x_i}{\sqrt{i}} \right) + 1$	30	[−600, +600]	70
	$F_{12} = \frac{\pi}{D} \left\{ 10 \sin^2(\pi y_i) + \sum_{i=1}^{D-1} (y_i - 1)^2 [1 + 10 \sin^2(\pi y_i + 1)] + (yD - 1)^2 + \sum_{i=1}^{D} u(x_i, 10, 100, 4) \right\}$ $y_i = 1 + \frac{x_i + 1}{4}$, $u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & x_i > a \\ 0 & -a < x_i < a \\ k(-x_i - a)^m & x_i < -a \end{cases}$	30	[−50, +50]	2000
	$F_{13} = 0.1 \left\{ 10 \sin^2(\pi y_i) + \sum_{i=1}^{D-1} (y_i - 1)^2 [1 + 10 \sin^2(\pi y_i + 1)] + (yD - 1)^2 + \sum_{i=1}^{D} u(x_i, 10, 100, 4) \right\}$	30	[−50, +50]	2000
	$F_{14} = \left[\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^{2} (x_i - a_{ij})^6} \right]^{-1}$	2	[−65.53, +65.53]	150
	$F_{15} = \sum_{i=1}^{11} \left[a_i - \frac{x_1 (b_i^2 + b_i x_i)}{b_i^2 + b_i x_2 + x_4} \right]^2$	4	[−5, +5]	400
	$F_{16} = 4x_1^2 - 2.1x_1^4 + \frac{1}{3} x_1^6 + x_1 x_2 - 4x_2^2 + 4x_2^4$	2	[−5, +5]	200
	$F_{17} = \left(x_2 - \frac{5.1}{4\pi^2} x_1^2 + \frac{5}{\pi} x_1 - 6 \right)^2 + 10 \left(1 - \frac{1}{8\pi} \right) \cos x_1 + 10$	2	[5, +10] × [0, +15	180

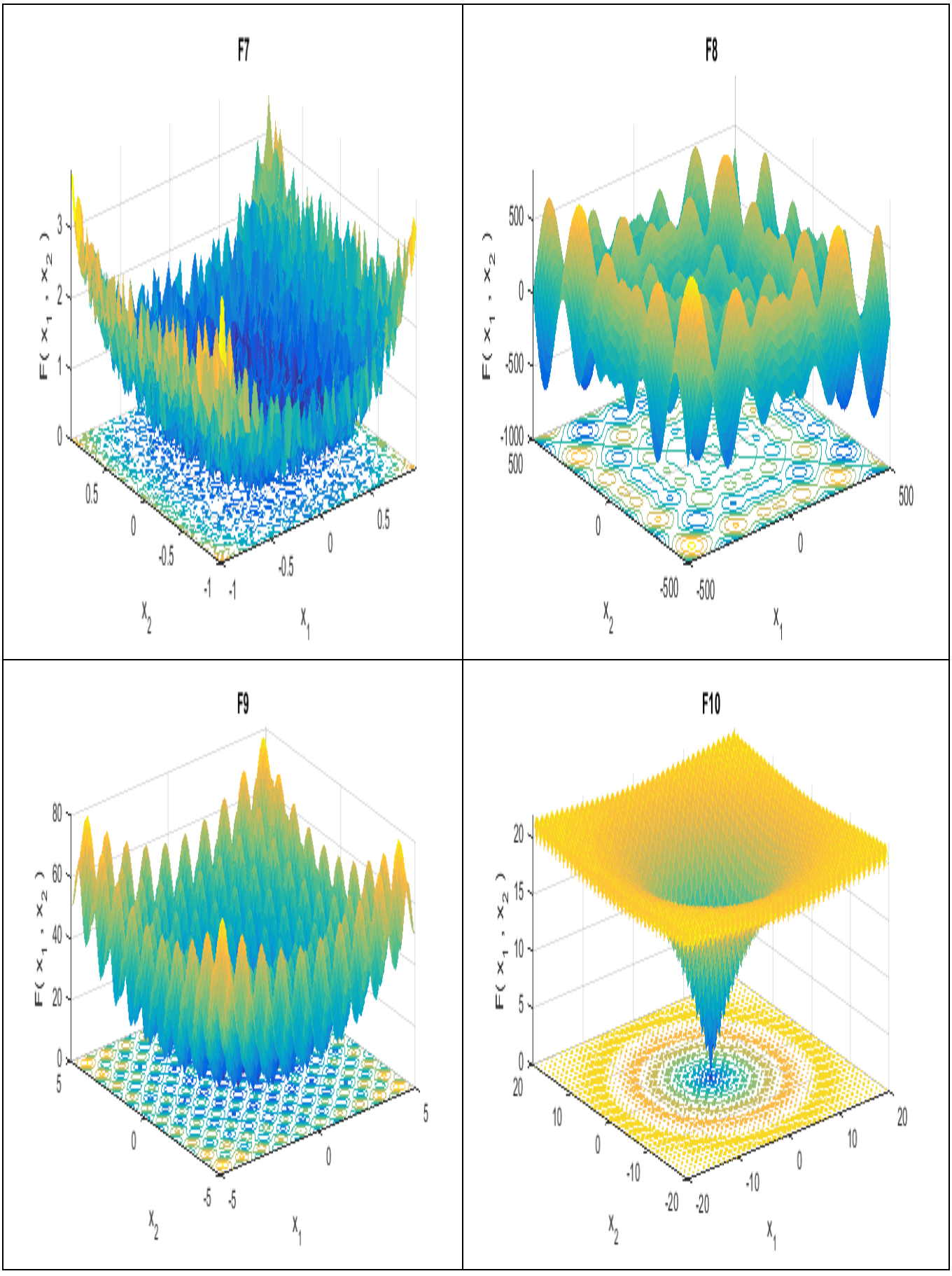
Fonctions multimodales à petite dimension	$F_{18} = [1+(x_1+x_2+1)^2(19-14x_1+3x_1^2-14x_2+6x_1x_2+3x_2^2)] \times [30+(2x_1-3x_2)^2(18-32$	2	[−5,+5]	200
	$F_{19} = -\sum_{i=1}^{4} c_i exp\left(-\sum_{j=1}^{3} a_{ij}(x_j - p_{ij})^2\right)$	3	[0,+1]	100
	$F_{20} = -\sum_{i=1}^{4} c_i exp\left(-\sum_{j=1}^{6} a_{ij}(x_j - p_{ij})^2\right)$	6	[0,+1]	250
	$F_{21} = -\sum_{i=1}^{5} [(X-a_i)(X-a_i)^T + c_i]^{-1}$	4	[0,+10]	200
	$F_{22} = -\sum_{i=1}^{7} [(X-a_i)(X-a_i)^T + c_i]^{-1}$	4	[0,+10]	200
	$F_{23} = -\sum_{i=1}^{10} [(X-a_i)(X-a_i)^T + c_i]^{-1}$	4	[0,+10]	200

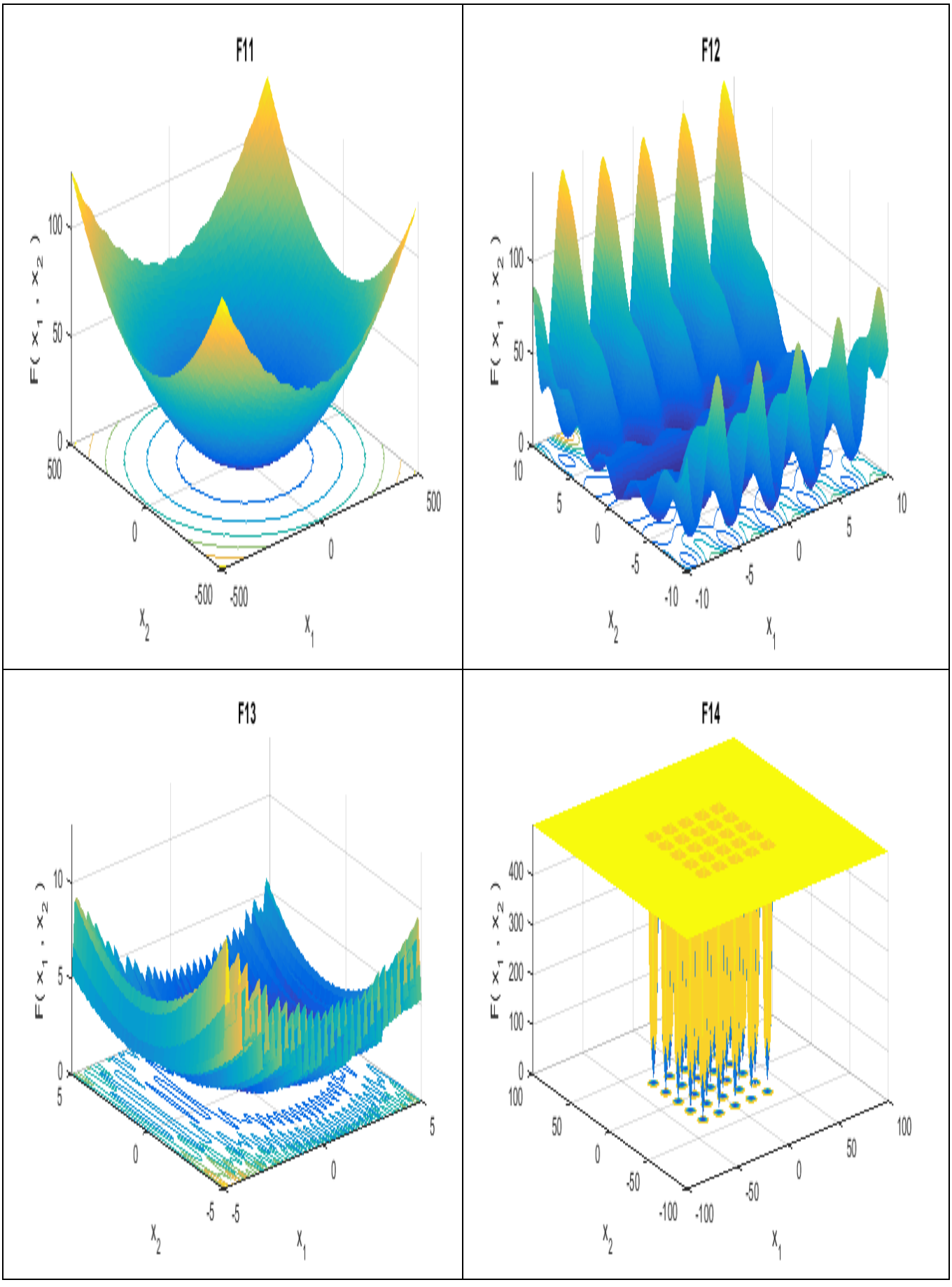
III.2.2 Trace 3D de chaque fonction

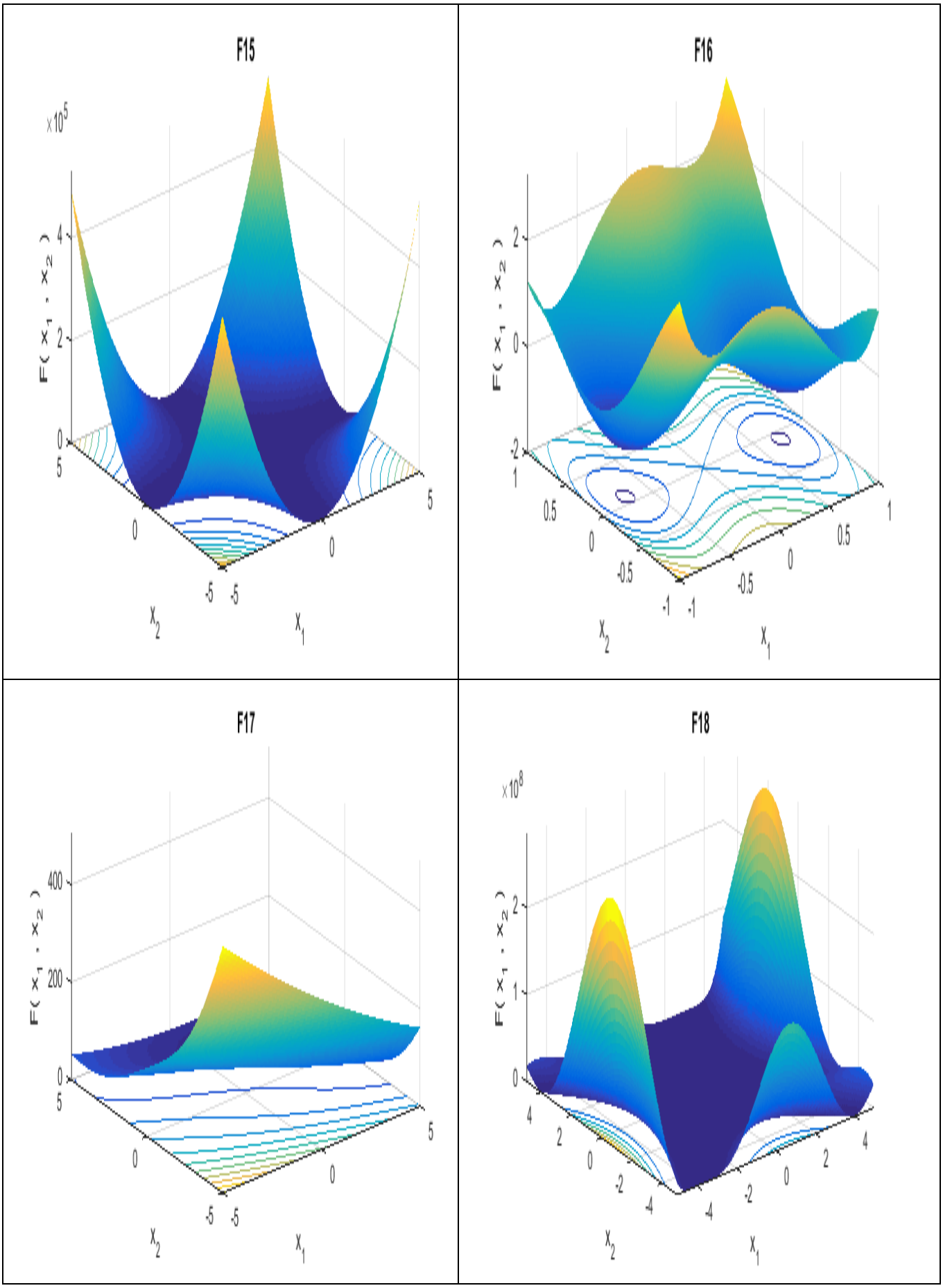
Pour bien présenter notre évaluation, nous dressons dans cette patrie les aspects des fonctions en 3D.

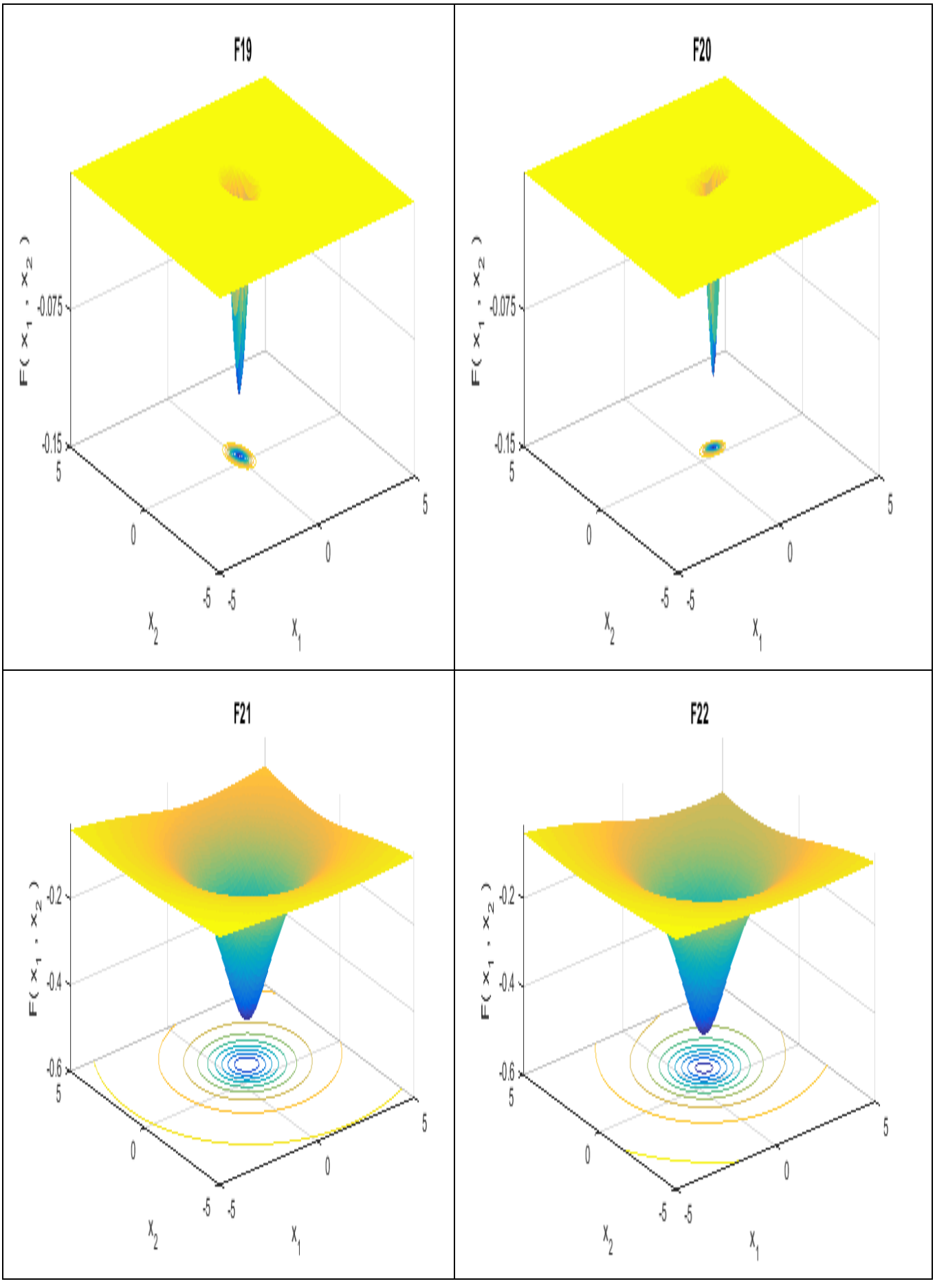



F3
F4
F5
F6
F(x1 , x2)
X1
X2


F7
F(x1 , x2)
X2
X1
F8
F(x1 , x2)
X2
X1
F9
F(x1 , x2)
X2
X1
F10
F(x1 , x2)
X2
X1


F11
F12
F13
F14
F(x1 , x2)
X1
X2


F15
F16
F17
F18
F(x1 , x2)
X1
X2


F19
F20
F21
F22
F(x1 , x2)
-0.075
-0.15
-0.2
-0.4
-0.6
5
0
-5
X2
X1

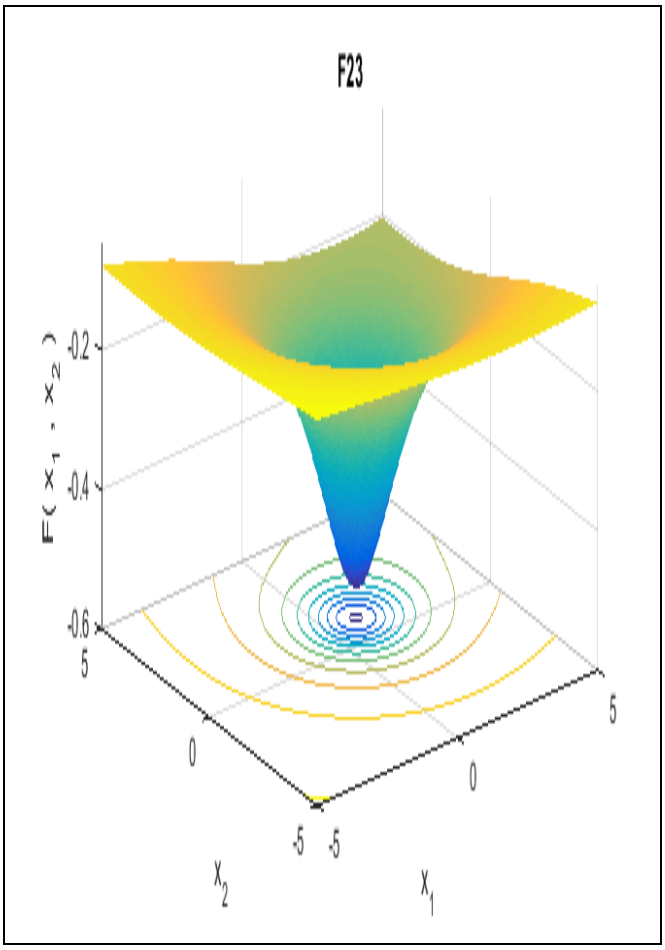


Figure III.1 Liste des figures des courbes 3D pour 23 fonctions **[33][34]**

III.2.3 Tableau comparatif des résultats

Pour avoir une bonne appréciation de notre solution, nous avons comparé la méthode symbiotique (SOS) avec d'autres méthodes : PSO, CA, ABC, GWO. Ces résultats sont montrés dans le Tableau III.2. Pour l'évaluation de chaque fonction, nous avons utilisés trois variables qui sont la moyenne, l'écart type et le classement de chaque méthode. Ce classement est obtenu sur la base de la meilleure moyenne des valeurs des fonctions objectives et les meilleurs écarts type par rapport à la valeur optimale de la fonction. Dans la catégorie unidimensionnelle la méthode SOS donne du F1 au F4 de très bons résultats par rapport aux autres méthodes sauf pour la fonction F5 pour laquelle SOS a donné une solution différente de l'optimum de même pour les autres méthodes. Pour la catégorie des multidimensionnelles (petite et haute), notre méthode a donné pour certaines valeurs des résultats exactes (optimales), pour d'autres, le résultat le plus proches et quelque uns des solutions différents.

Nous avons constaté que notre méthode est efficace dans l'exploitation c'est-à-dire elle génère les meilleures solutions, mais pour l'exploration (F8…F23) donne des solutions entre bonnes et acceptables.

Tableau III.2 Résultat d'évaluation des méthodes

Fonctions		**Optimum**	**PSO**	**ABC**	**CA**	**SCA**	**GWO**	**SOS**
F1	Moy	0	0.3970	0.1336	1.2946e-04	1.2728	8.1334e-41	**0**
	EcTy		0.3209	0.0324	1.3350e-04	3.3294	1.1758e-40	0
	Class		5	4	3	6	2	1
F2	Moy	0	0.0352	0.1752	59.2268	2.3977e-06	8.0399e-47	**0**
	EcTy		0.1624	0.0990	20.9776	6.6258e-06	8.3453e-47	0
	Class		4	5	6	3	2	1
F3	Moy	0	1.6452e+003	8.4519e+03	2.6431e+04	3.8001e+03	2.8663e-11	**0**
	EcTy		640.9846	1.7992e+03	4.6212e+03	3.0736e+03	7.8706e-11	0
	Class		3	5	6	4	2	1
F4	Moy	0	2.1999	12.3781	3.1395	5.3804	1.3831e-21	**0**
	EcTy		1.7124	1.6148	3.2499	5.9093	2.8568e-21	0
	Class		3	6	4	5	2	1
F5	Moy	0	33.0197	27.8865	48.5678	27.0538	**25.8985**	28.8209
	EcTy		31.9313	0.4019	115.9528	0.7684	**0.7736**	0.0334
	Class		5	3	6	2	**1**	4
F6	Moy	0	2.7682e+004	3.3316e+04	1.5645e+04	3.1148e+04	1.2398e+03	**2.9951**
	EcTy		4.9476e+003	5.1635e+03	3.3031e+03	5.9109e+03	330.2118	**0.7141**
	Class		4	6	3	5	2	1
F7	Moy	0	0.0251	0.0311	0.0701	0.0086	**1.9404e-04**	1.6e-03
	EcTy		0.0170	0.0079	0.0177	0.0085	**9.6654e-05**	2.2e-03
	Class		4	5	6	3	**1**	2
F8	Moy	-418.9829 * n	-9.9460e+003	**-9.1745e+32**	-1.0400e+04	-4.3341e+03	-6.3427e+03	-6.5835e+03
	EcTy		514.9218	**2.1229e+33**	669.8460	285.6626	818.9499	666.3787
	Class		3	**1**	2	6	5	4
F9	Moy	0	275.2287	308.3371	316.9494	231.5441	47.6089	**0**
	EcTy		17.2402	20.9358	18.6367	47.0853	15.1017	**0**
	Class		4	5	6	3	2	1
F10	Moy	8.8818e-16	13.4428	16.4168	11.1855	16.7991	0.0844	**8.8818e-16**
	EcTy		0.7387	0.7822	0.8800	3.8739	0.0292	**0**
	Class		4	5	3	6	2	1
F11	Moy	0	43.5010	57.5828	18.1817	36.2411	0.0848	**0**
	EcTy		11.9431	9.5749	4.6308	15.6611	0.0495	**0**
	Class		5	6	3	4	2	1
F12	Moy	1.5705e-032	0.0456	0.5025	4.1989	0.3651	0.0120	**0.1628**
	EcTy		0.0902	0.1092	5.9838	0.0549	0.0119	**0.0616**
	Class		2	5	6	4	1	3
F13	Moy	0	**0.0040**	0.1013	0.0062	2.0343	0.2046	2.6935
	EcTy		**0.0095**	0.0132	0.0096	0.1694	0.1377	0.6036
	Class		1	3	2	5	4	6
F14	Moy	0.998004	0.9980	1.0297	1.1561	1.7920	3.3878	**0.9980**
	EcTy		4.3578e-013	0.1571	0.7905	0.9918	3.8337	**0**
	Class		2	3	4	5	6	1
F15	Moy	0.0003075	0.0036	0.0010	0.0044	9.5601e-04	0.0019	**3.4411e-04**
	EcTy		0.0075	3.3967e-05	0.0122	3.9461e-04	0.0056	**1.8314e-04**
	Class		5	3	6	2	4	1
F16	Moy	-1.0316285	-1.0316	-1.0315	-1.0316	-1.0316	-1.0316	**-1.0316**
	EcTy		7.0682e-013	7.6299e-05	6.2476e-16	5.2826e-05	1.6968e-08	**6.7987e-16**
	Class		3	6	2	5	4	1
F17	Moy	0.398	0.3979	0.3980	0.3979	0.3999	0.3979	**0.3979**
	EcTy		1.0120e-011	6.9784e-05	0	0.0017	1.5976e-06	**0**
	Class		2	4	1	5	3	1
F18	Moy	3	3.0000	3.0001	3.0000	3.0000	3.0000	**3.0000**
	EcTy		1.8965e-012	1.8513e-04	1.3628e-15	9.3354e-05	2.1140e-05	**0.5368e-16**
	Class		3	6	2	5	4	**1**
F19	Moy	-3.86	-3.8628	-3.8625	**-3.8628**	-3.8537	-3.8612	-3.8628
	EcTy		1.0583e-007	2.3838e-04	**1.8467e-15**	0.0037	0.0025	2.2662e-15
	Class		3	4	**1**	6	5	2
F20	Moy	-3.32	-3.2602	-3.2515	-3.2792	-3.0135	-3.2608	**-3.2839**
	EcTy		0.0606	0.0403	0.0582	0.2057	0.0732	**0.0566**
	Class		4	5	2	6	3	**1**
F21	Moy	-10.1532	-6.9258	**-10.1018**	-7.7627	-3.2472	-9.9496	-8.7258
	EcTy		3.0358	**0.1039**	3.5566	1.6226	1.0105	2.3362
	Class		5	**1**	4	6	2	3
F22	Moy	-10.4029	-7.8961	-10.3587	-7.5721	-4.5168	**-10.4009**	-8.2768
	EcTy		3.4918	0.0345	3.5908	1.4180	**0.0011**	2.6576
	Class		4	2	5	6	**1**	3
F23	Moy	-10.5364	-8.5370	**-10.4930**	-7.7253	-4.4472	-10.3185	-9.0222
	EcTy		3.3054	**0.0230**	3.6025	0.9653	1.0813	2.4782
	Class		4	**1**	5	6	2	3
Moyenne Classe			3.5652	4.0869	3.8260	4.6956	2.8695	**1.9130**
Classement global			3	5	4	6	2	**1**

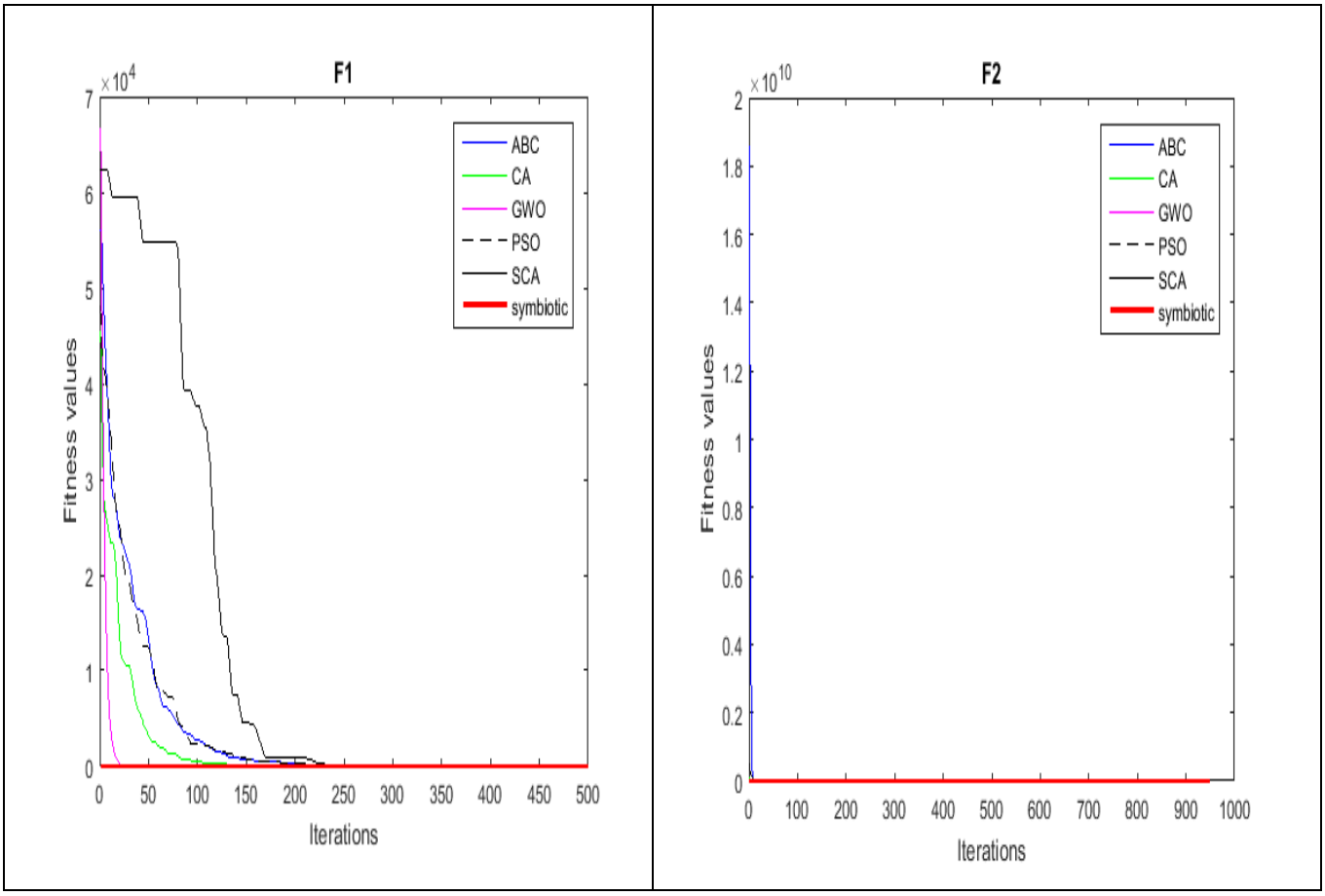
III.2.4 Courbe de Fitness

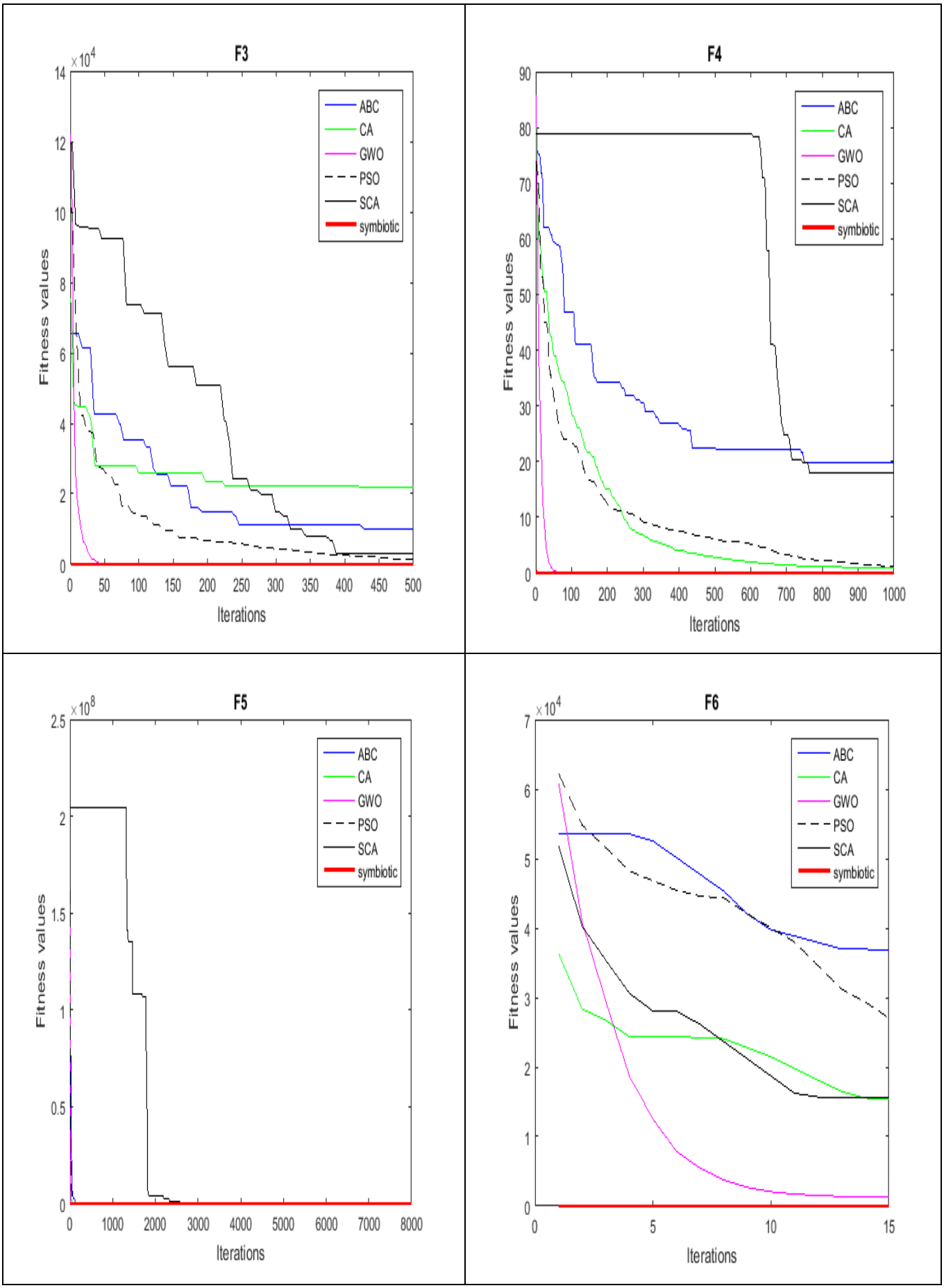
Pour plus de performance de cette métaheuristique, on présente les tracés des courbes des fonctions Fitness de teste. La figure III.2 montre les courbes de convergence obtenues par six algorithmes sur 23 fonctions.

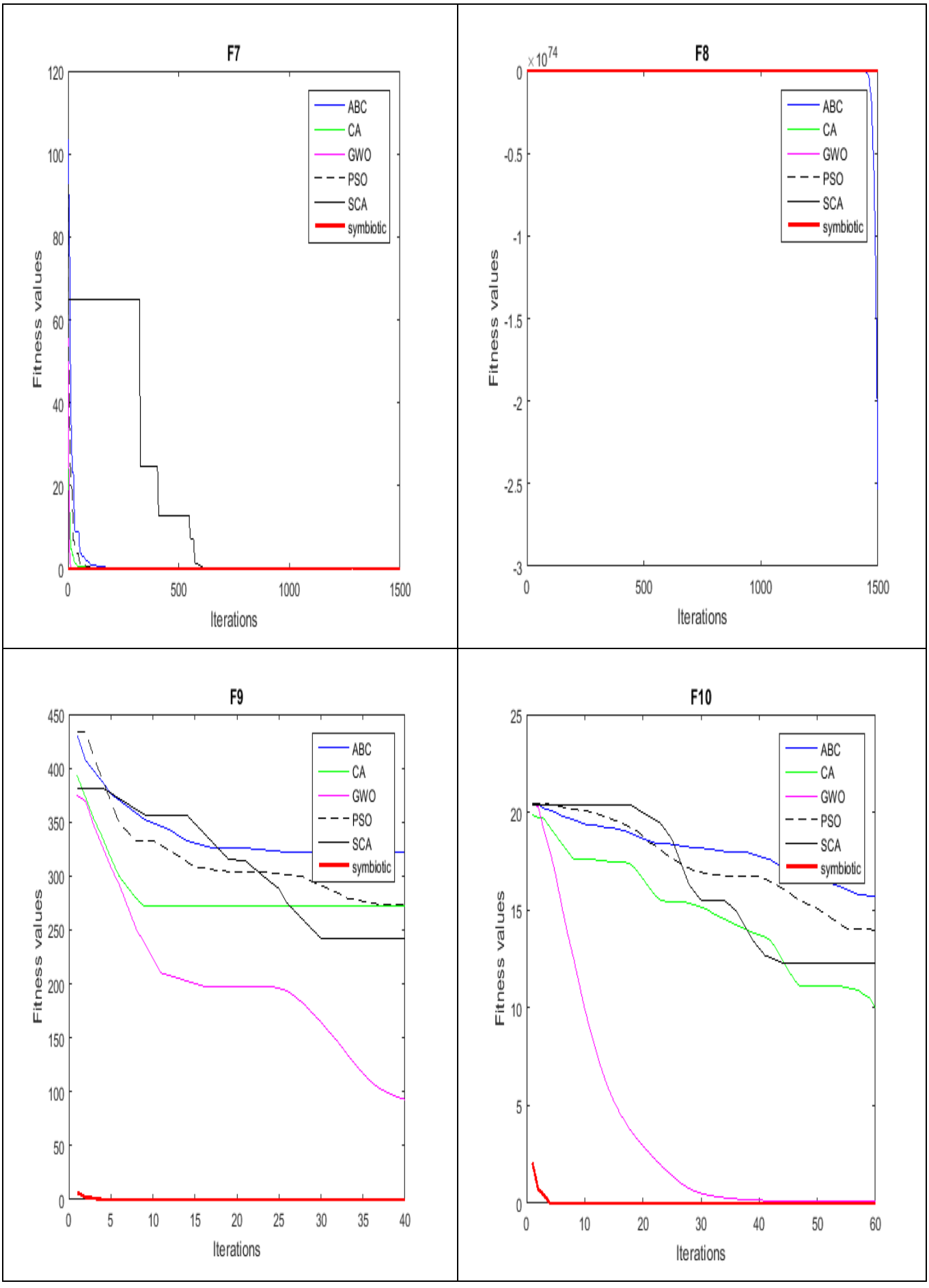
Les valeurs indiquées sur ces figurent représentent une comparaison des résultats de fitness des 23 fonctions selon SOS, PSO, ABC, CA et SCA.

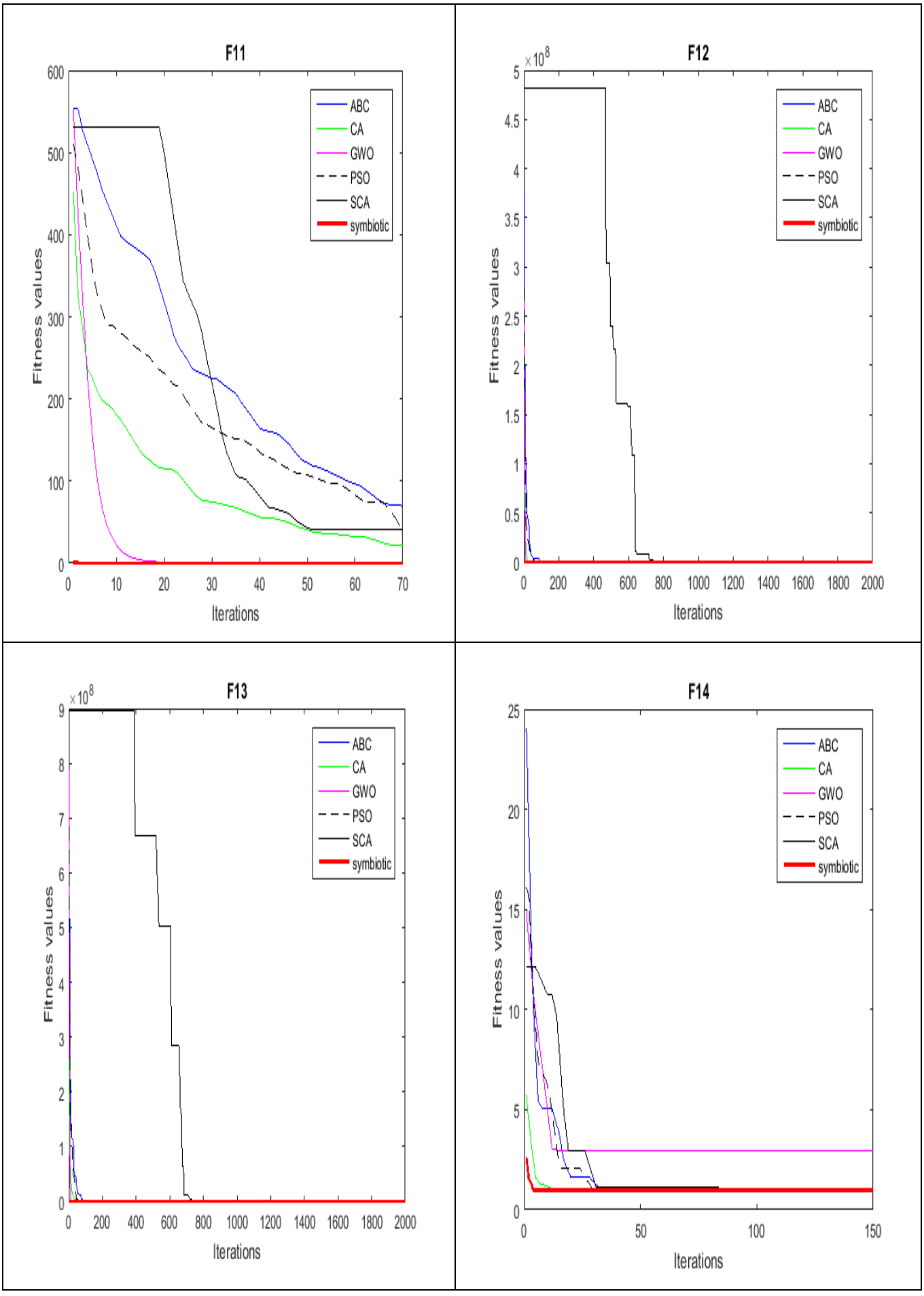
A partir de ces figures, il est clair que notre méthode possède une meilleure précision de recherche que les cinq autres méthodes, cette figure montre aussi que l'algorithme SOS a une vitesse de convergence plus rapide.

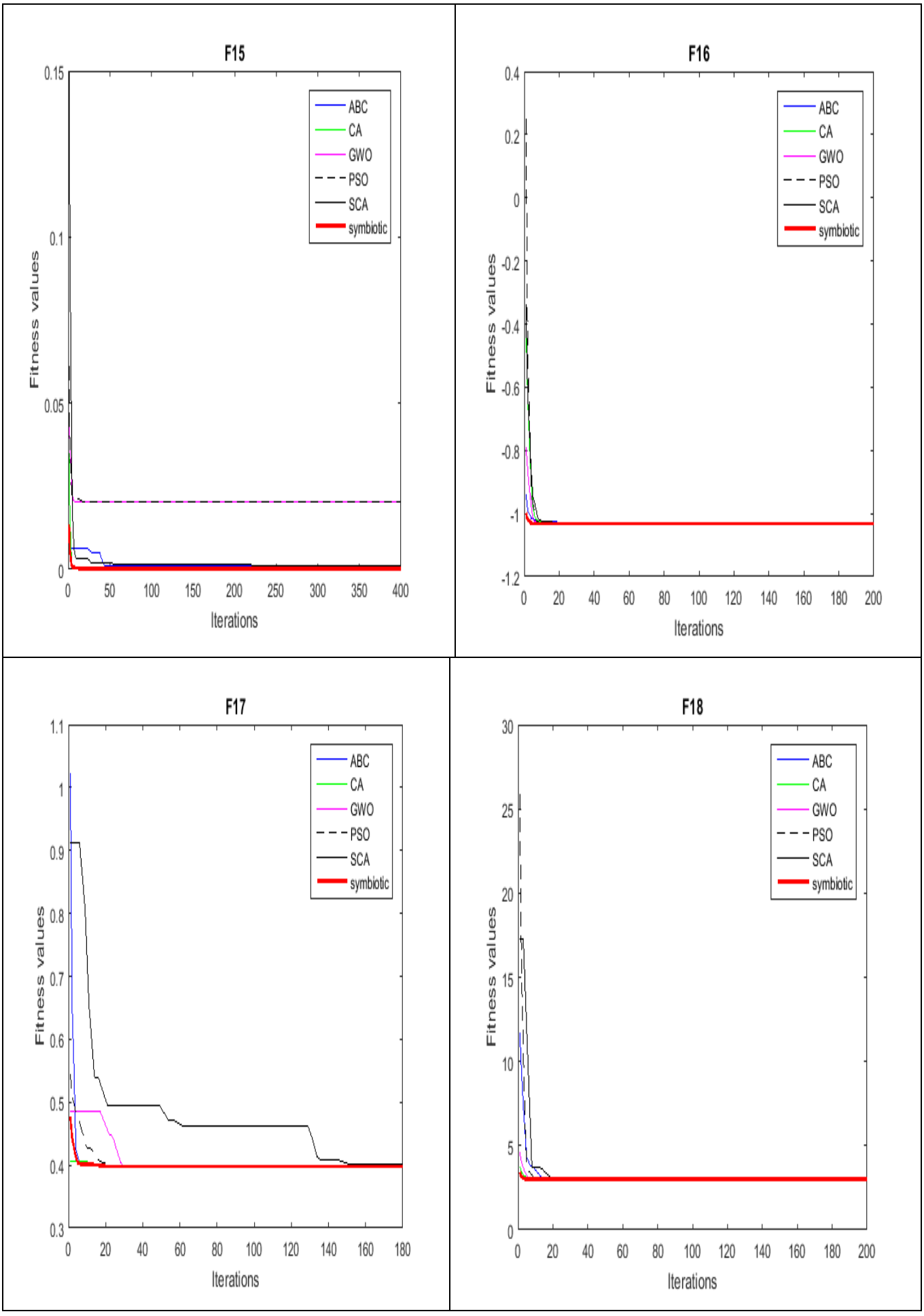
Cela prouve que SOS possède une forte robustesse et une bonne stabilité assurant une nette convergence par rapport aux autres méthodes.

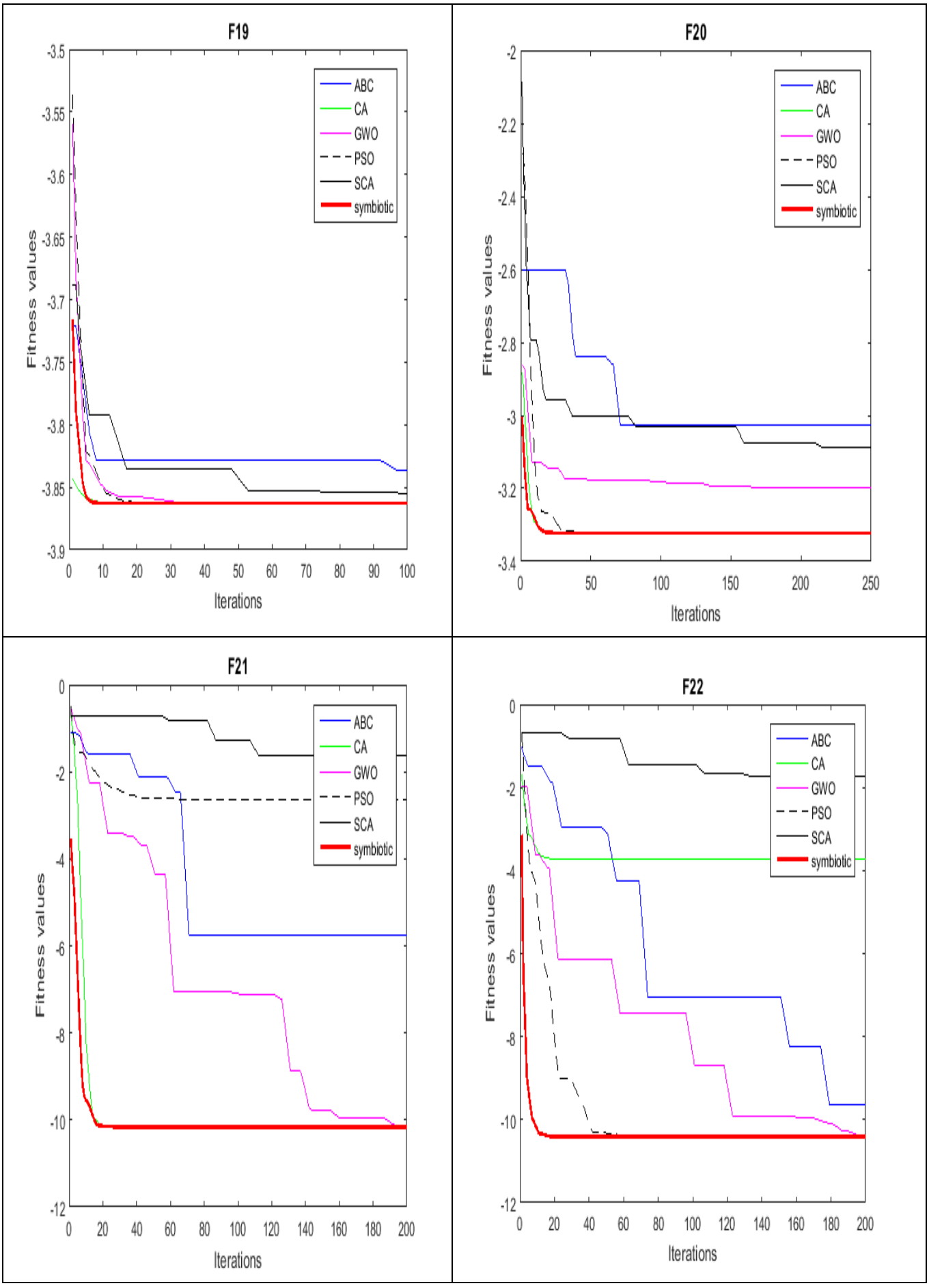



F3
×10^4
Fitness values
Iterations
ABC
CA
GWO
PSO
SCA
symbiotic
F4
Fitness values
Iterations
ABC
CA
GWO
PSO
SCA
symbiotic
F5
×10^8
Fitness values
Iterations
ABC
CA
GWO
PSO
SCA
symbiotic
F6
×10^4
Fitness values
Iterations
ABC
CA
GWO
PSO
SCA
symbiotic


F7
Fitness values
Iterations
ABC
CA
GWO
PSO
SCA
symbiotic
F8
×10^74
Fitness values
Iterations
ABC
CA
GWO
PSO
SCA
symbiotic
F9
Fitness values
Iterations
ABC
CA
GWO
PSO
SCA
symbiotic
F10
Fitness values
Iterations
ABC
CA
GWO
PSO
SCA
symbiotic


F11
Fitness values
Iterations
ABC
CA
GWO
PSO
SCA
symbiotic
F12
×10^8
Fitness values
Iterations
ABC
CA
GWO
PSO
SCA
symbiotic
F13
×10^8
Fitness values
Iterations
ABC
CA
GWO
PSO
SCA
symbiotic
F14
Fitness values
Iterations
ABC
CA
GWO
PSO
SCA
symbiotic


F15
Fitness values
Iterations
ABC
CA
GWO
PSO
SCA
symbiotic
F16
Fitness values
Iterations
ABC
CA
GWO
PSO
SCA
symbiotic
F17
Fitness values
Iterations
ABC
CA
GWO
PSO
SCA
symbiotic
F18
Fitness values
Iterations
ABC
CA
GWO
PSO
SCA
symbiotic


F19
Fitness values
Iterations
ABC
CA
GWO
PSO
SCA
symbiotic
F20
Fitness values
Iterations
ABC
CA
GWO
PSO
SCA
symbiotic
F21
Fitness values
Iterations
ABC
CA
GWO
PSO
SCA
symbiotic
F22
Fitness values
Iterations
ABC
CA
GWO
PSO
SCA
symbiotic

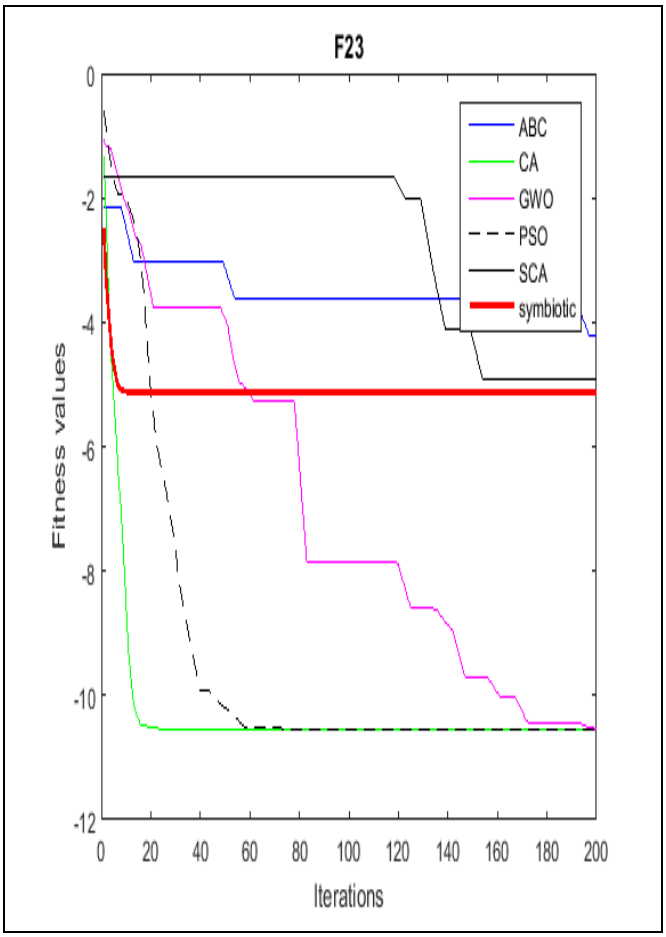


Figure III.2 Liste des figures des courbes de Fitness pour 23 fonctions

III.2.5 Analyse de variation ANOVA

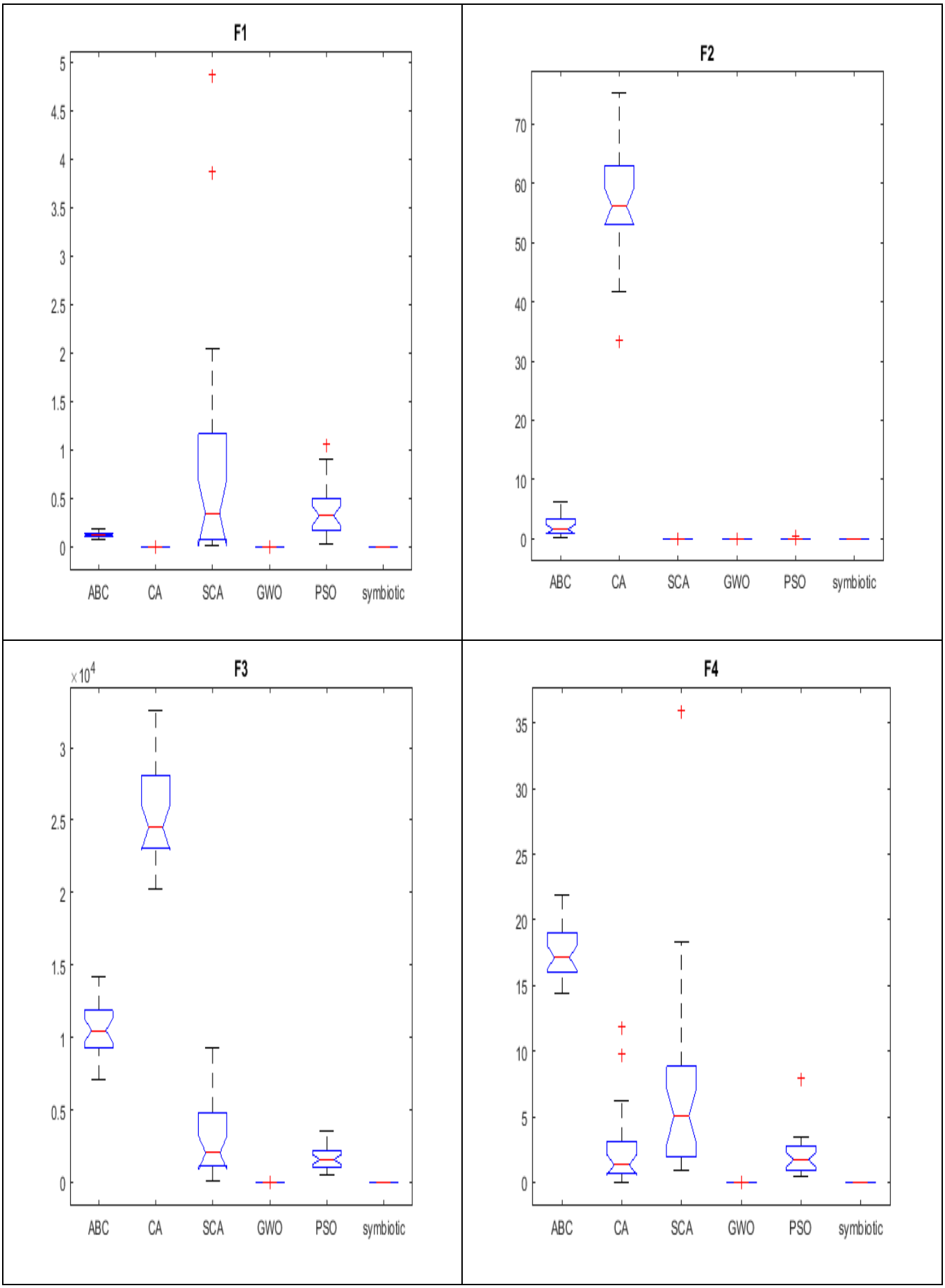
Pour expliciter notre évaluation de la méthode proposée par rapport aux autres algorithmes de comparaison, nous avons utilisés le modèle de la Figure III.3 « Test ANOVA » qui est une représentation schématique de la distribution d'une variable.

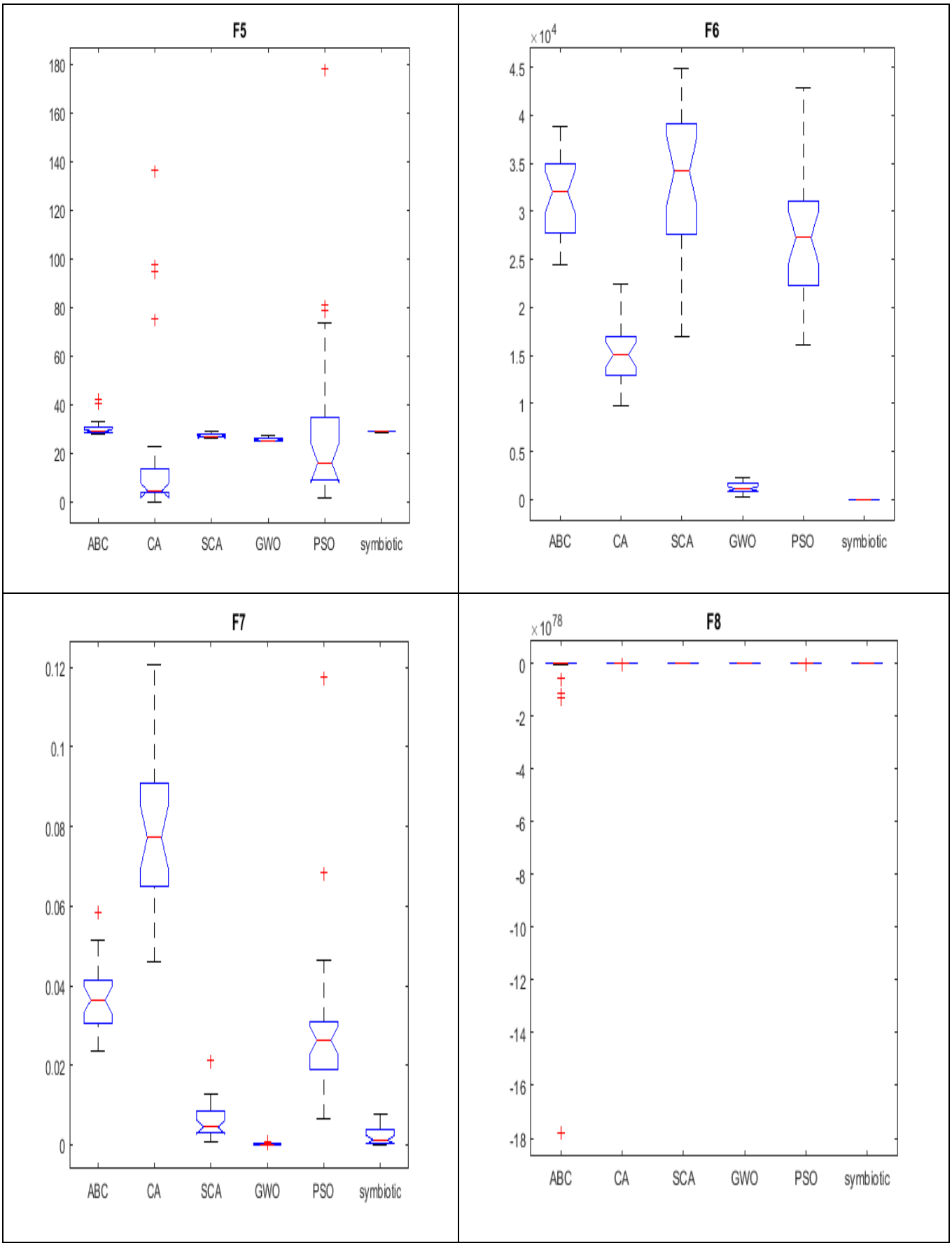
Le test Anova inventée en 1977 par John Tukey, appelée aussi diagramme en boîte, boîte de Tukey ou box-and-whisker plot en anglais, est utilisée pour représenter le schéma essentiel d'une série statistique quantitative.

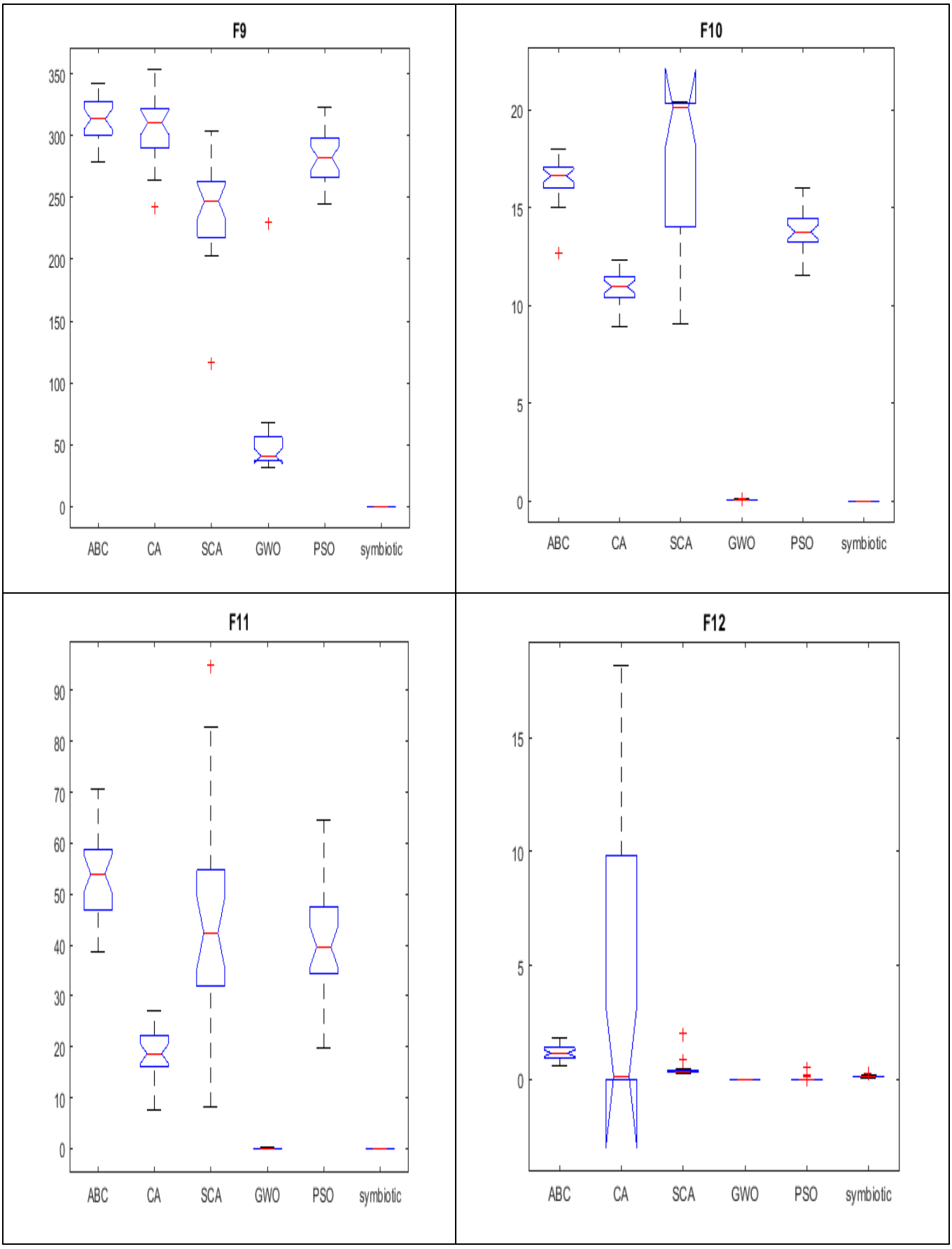
Le test Anova qui on effectué sur les 23 fonctions de test donne le résultat de 25 exécution indépendantes de notre méthode.

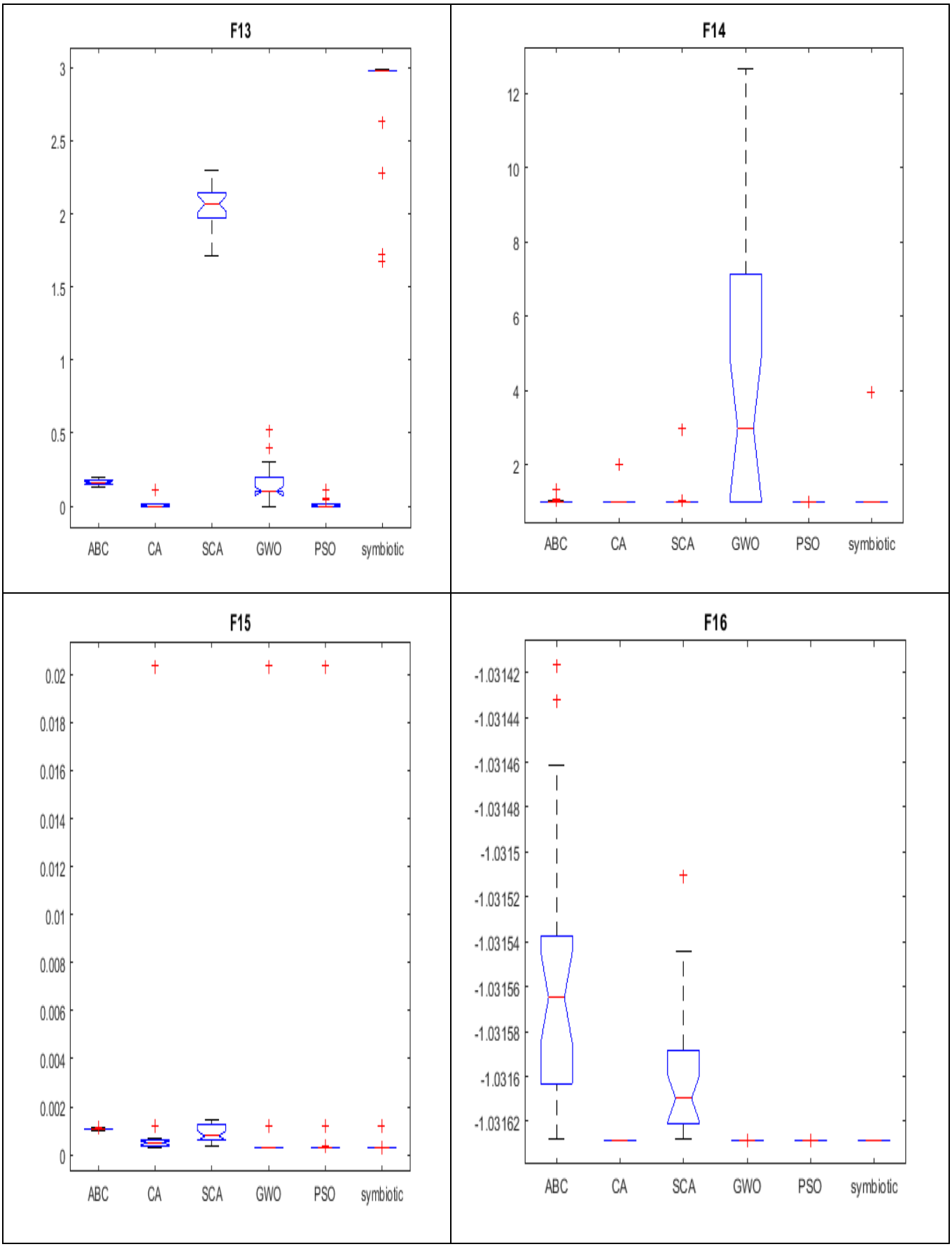
Les formes dans les figures obtenues par ce test qui sont des boites, représentent les variations des solutions optimales entre les valeurs minimales et maximales, centrée par la moyenne qui est la solution optimale de la fonction sujette de test. Nous remarquons que pour SOS les fonctions de F1 et F19 notre méthode est très stable cela prouve qu'il génère une bonne performance avec une nette précision. Dans les derniers cas l'algorithme SOS se comporte de manière acceptable.

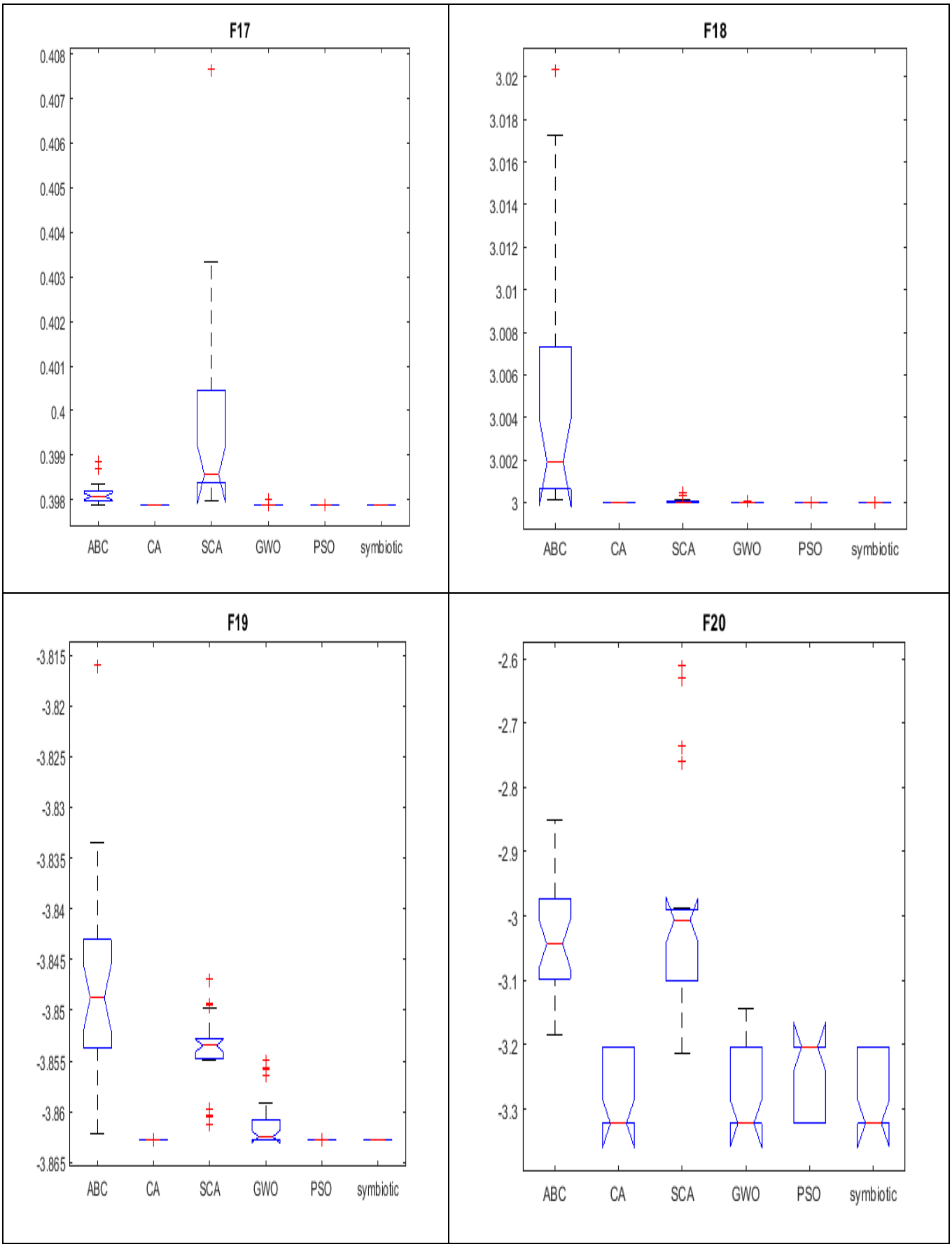
Ces résultats montrent que cet algorithme est plus adéquat pour les problèmes de grandes dimensions.


F1
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
ABC
CA
SCA
GWO
PSO
symbiotic
F2
70
60
50
40
30
20
10
0
ABC
CA
SCA
GWO
PSO
symbiotic
F3
x 10^4
3
2.5
2
1.5
1
0.5
0
ABC
CA
SCA
GWO
PSO
symbiotic
F4
35
30
25
20
15
10
5
0
ABC
CA
SCA
GWO
PSO
symbiotic


F5
180
160
140
120
100
80
60
40
20
0
ABC
CA
SCA
GWO
PSO
symbiotic
F6
x 10^4
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
ABC
CA
SCA
GWO
PSO
symbiotic
F7
0.12
0.1
0.08
0.06
0.04
0.02
0
ABC
CA
SCA
GWO
PSO
symbiotic
F8
x 10^78
0
-2
-4
-6
-8
-10
-12
-14
-16
-18
ABC
CA
SCA
GWO
PSO
symbiotic


F9
350
300
250
200
150
100
50
0
ABC
CA
SCA
GWO
PSO
symbiotic
F10
20
15
10
5
0
ABC
CA
SCA
GWO
PSO
symbiotic
F11
90
80
70
60
50
40
30
20
10
0
ABC
CA
SCA
GWO
PSO
symbiotic
F12
15
10
5
0
ABC
CA
SCA
GWO
PSO
symbiotic


F13
3
2.5
2
1.5
1
0.5
0
ABC
CA
SCA
GWO
PSO
symbiotic
F14
12
10
8
6
4
2
ABC
CA
SCA
GWO
PSO
symbiotic
F15
0.02
0.018
0.016
0.014
0.012
0.01
0.008
0.006
0.004
0.002
0
ABC
CA
SCA
GWO
PSO
symbiotic
F16
-1.03142
-1.03144
-1.03146
-1.03148
-1.0315
-1.03152
-1.03154
-1.03156
-1.03158
-1.0316
-1.03162
ABC
CA
SCA
GWO
PSO
symbiotic


F17
0.408
0.407
0.406
0.405
0.404
0.403
0.402
0.401
0.4
0.399
0.398
ABC
CA
SCA
GWO
PSO
symbiotic
F18
3.02
3.018
3.016
3.014
3.012
3.01
3.008
3.006
3.004
3.002
3
ABC
CA
SCA
GWO
PSO
symbiotic
F19
-3.815
-3.82
-3.825
-3.83
-3.835
-3.84
-3.845
-3.85
-3.855
-3.86
-3.865
ABC
CA
SCA
GWO
PSO
symbiotic
F20
-2.6
-2.7
-2.8
-2.9
-3
-3.1
-3.2
-3.3
ABC
CA
SCA
GWO
PSO
symbiotic

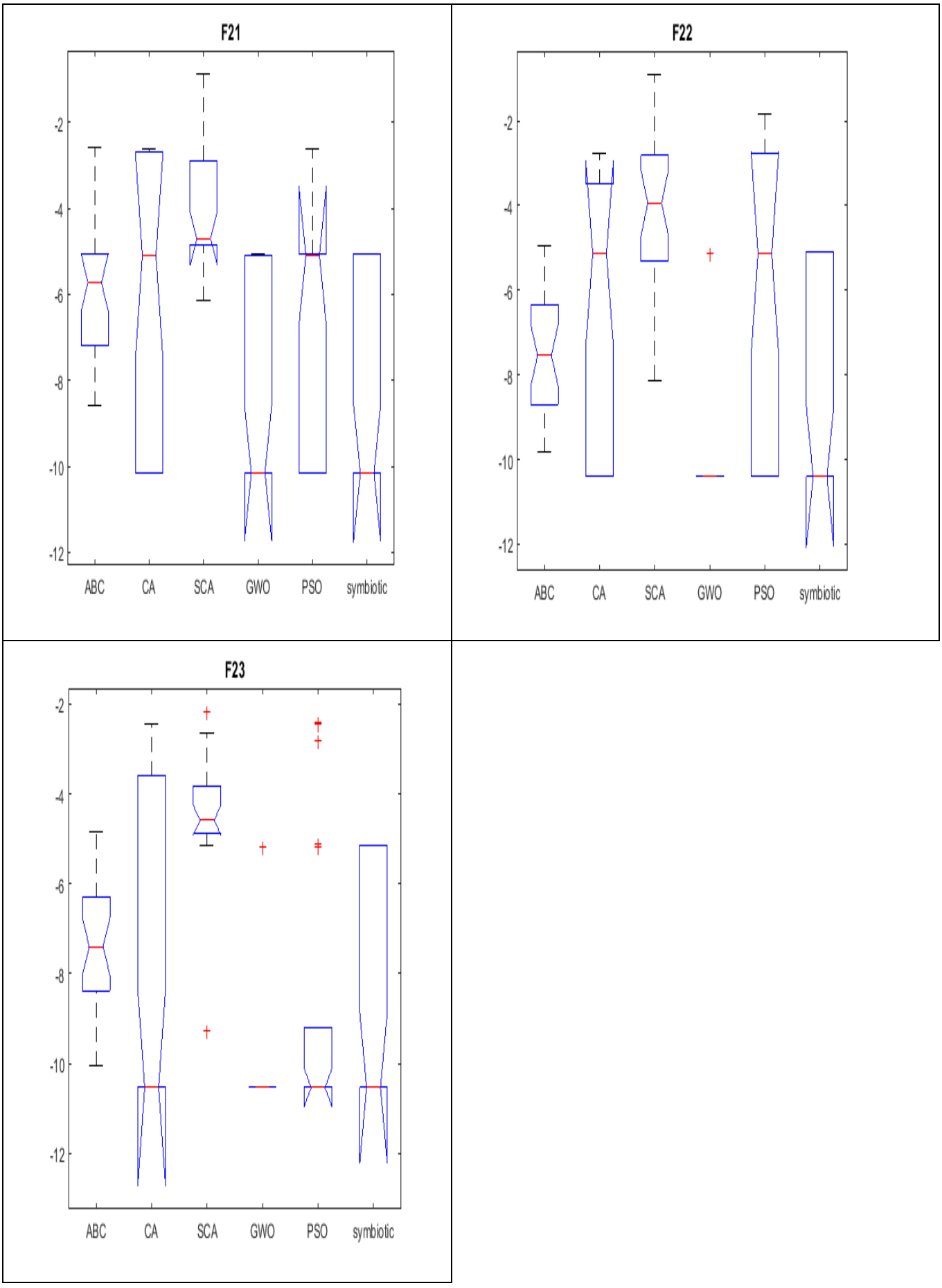


Figure III.3 Variation Anova pour 23 fonctions

III.3 Evaluation de la symbiotique pour le datamining

Dans ce qui suit nous allons faire une évaluation de la symbiotique pour le mécanisme de clustering automatique dans le datamining.

Dans nos essais, nous avons exploités l'algorithme SOS pour une classification supervisée et pour avoir une bonne appréciation, nous avons en même temps appliqué les algorithmes GA, PSO, et DE sur le même dataset.

III.3.1 La fonction objective (fitness)

La sélection d'une fonction objective constitue une force motrice moyenne derrière toute technique évolutive.

Suite à l'étude de la fonction fitness de **[35]**, nous avons constaté qu'elle est complexe et difficile à calculer.

A cet effet, suite à notre besoin nous avons suggéré la fonction de fitness qui est basée sur la distance minimale entre les instances et les centres de classes proposés aléatoirement. Ainsi la fonction de fitness est donnée par la formule suivante :

$$F = \sum_{I=1}^{n} \sum_{j=1}^{k} (dmin(X_i, ck_j)$$

dmin: distance minimale entre chaque instance et les centres de classes.
X : matrice de données.

ck : les centres des classes.

k : nombre de classes.

n : nombre d'instances de la base de données.

III.3.2 Les bases de données utilisées

Dans les expérimentations suivantes, nous avons exploités les bases de données du domaine publique avec différente complexité sachant que : n (nombre d'instance), d (nombre de caractéristique) et k (nombre de classes) **[36]** :

III.3.2.1 Complexité moyenne

Dans le cas d'une complexité moyenne nous avons utilisé la base de données cancer du sein. Cette base contient 2 classes : le type bénin avec 446 instances et le type malin avec 237 instances.

Sa complexité provient du rapprochement entre quelques unes de ses caractéristiques. Les indicateurs de cette base sont : (n=683, d=9, k=2).

III.3.2.2 Complexité plus que la moyenne

Pour tester le comportement exact de l'algorithme SOS nous avons exploité la base de données IRIS qui est de complexité plus que la moyenne.

Cette base contient 3 classes : SETOSA, VERSICOLOR, VIRGINICA et chaque classe contient 50 instances.

Ces trois classes sont des types de fleurs, il est à noter que la complexité apparait sur les caractéristiques de deux classes qui se ressemblent, les données de la base IRIS sont : (n= 150, d=4, k= 3).

III.3.2.3 Complexité élevée

Cette base est de complexité élevée car les paramètres discriminatoires sont superposés ou possèdent une interaction.

La base est partagée en deux catégories, la première concerne les donneurs de sang à un jour donné et à une heure précise pendant un mois.

Les statistiques obtenues indiquent que 24% (179 instances) sont des donneurs et 76 % (567 instances) sont non donneurs. Les valeurs des paramètres de cette base sont : (n=748, d=5, k=2).

III.3.3 Stratégie de simulation

La stratégie expérimentale que nous avons développée repose sur un choix aléatoire des centres des classes.

Ensuite, la fonction fitness évalue les solutions intermédiaires entre eux pour déterminer la meilleure qui sont sous la forme suivante :

$X=\{x_1, x_2,\ldots\ldots\ldots\ldots..x_n\}$ tel que x_i est une solution matricielle.

III.3.4 Résultats d'expérimentation

Nous avons utilisés 200 itérations pour l'exécution des quatre méthodes. Nous avons deux visions d'interprétations des résultats, l'une sera décrite d'après les courbes de fitness et les histogrammes de classification, la seconde sera présentée d'après la matrice de confusion.

III.3.4.1 Courbes de la fonction Fitness

Dans ce qui suit, nous avons évalué la fonction fitness pour les quatre algorithmes sur les trois bases décrites précédemment.

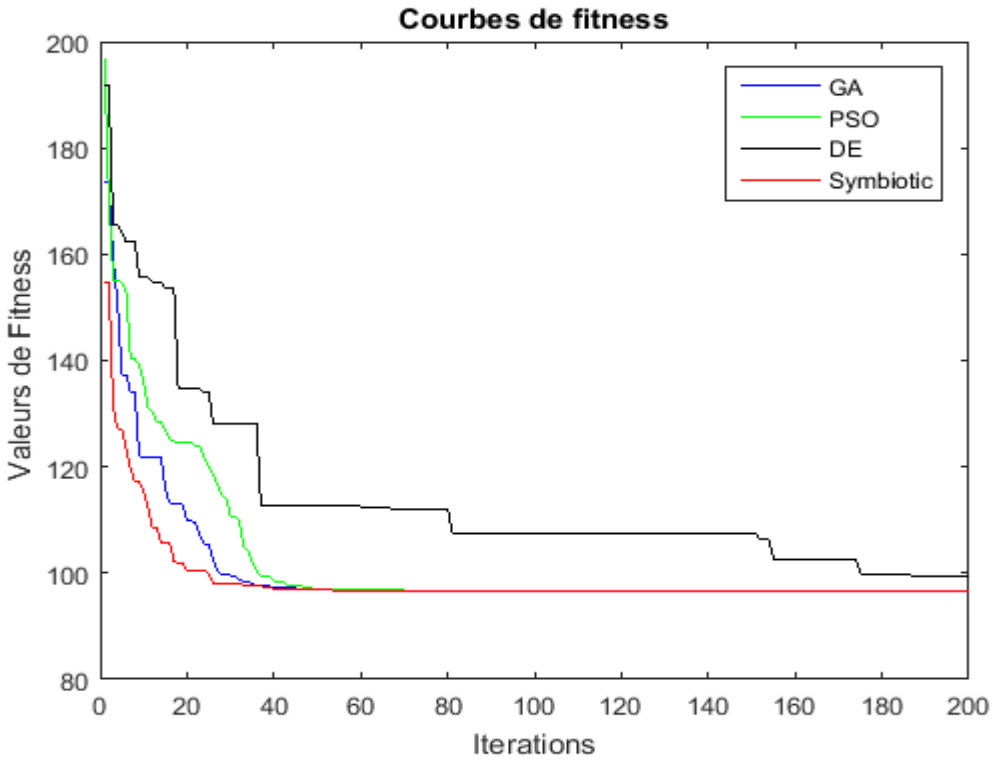


Figure III.4 Comparaison sur la base IRIS

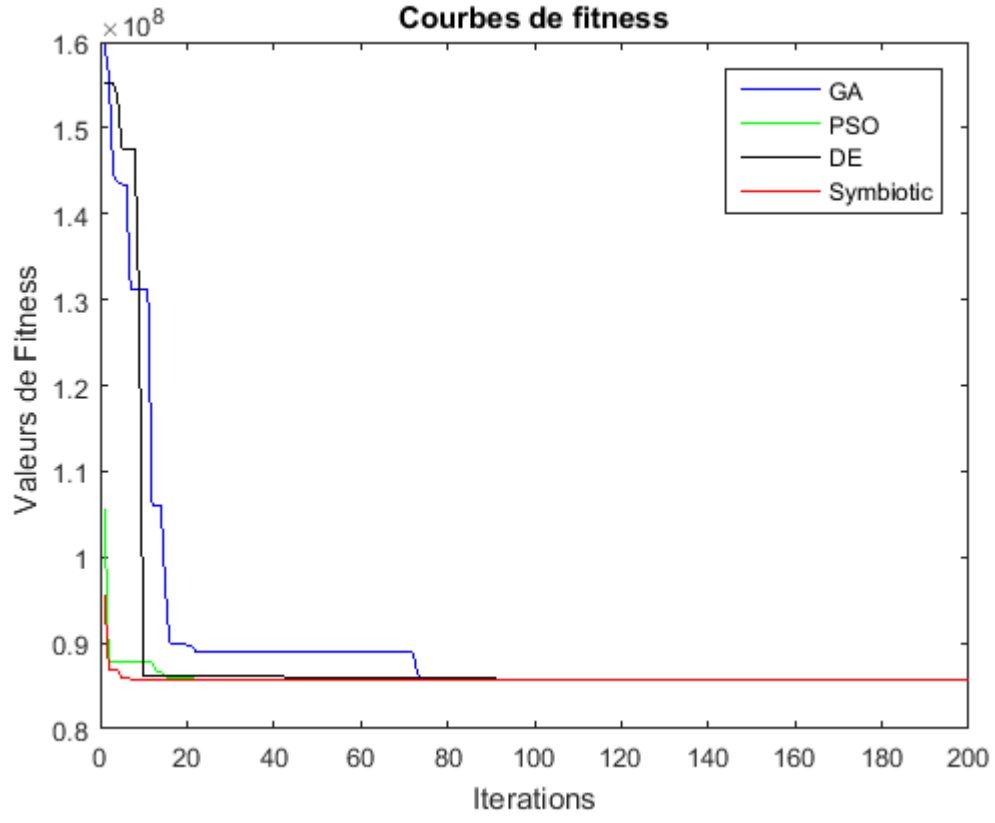


Figure III.5 Comparaison sur la base cancer du sein

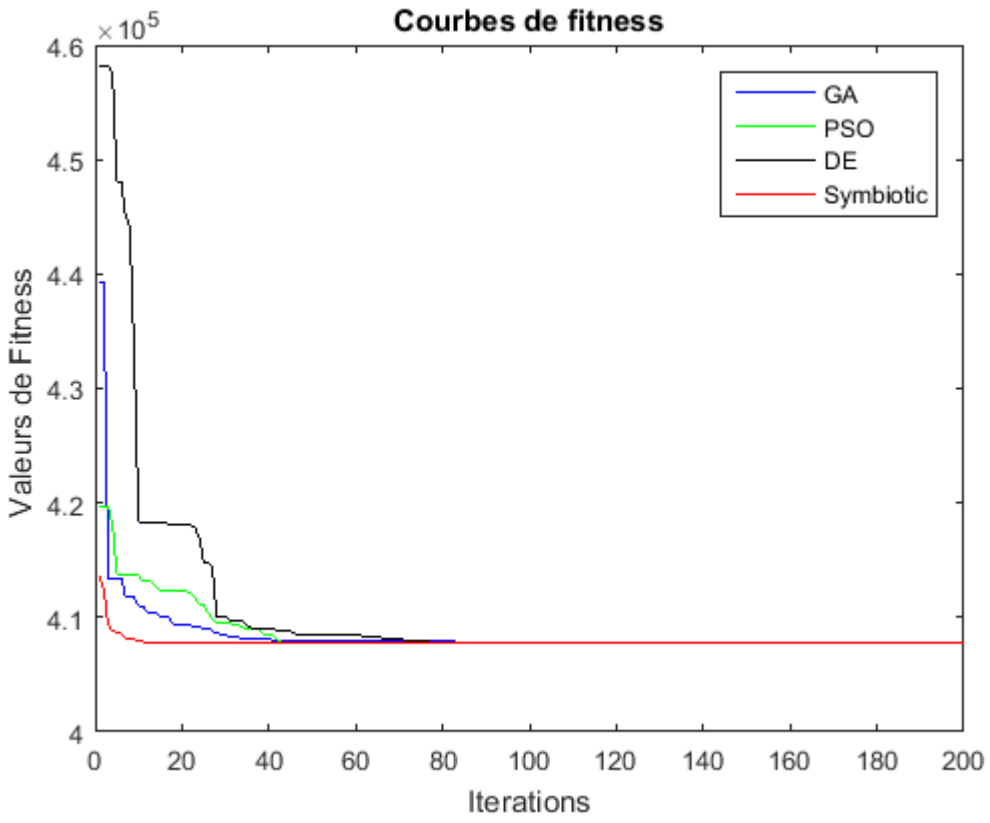


Figure III.6 Comparaison sur la base de donneur de sang d'un mois

Les figurent précédentes qui représentent les variations de la fonction fitness, montrent visuellement que la méthode SOS est plus performante et convergence de manière plus rapide par rapport aux autres méthodes.

Cela prouve la bonne qualité de l'évaluation de solution de SOS par rapport aux autres métaheuristiques utilisées.

III.3.4.2 Histogramme de classification

Après exécution des algorithmes sur les trois bases de données, nous avons représenté les données classifiées sous forme d'histogrammes pour avoir une visibilité significative dans un souci de bonne comparaison entre les méthodes.

- **Application sur la base IRIS**

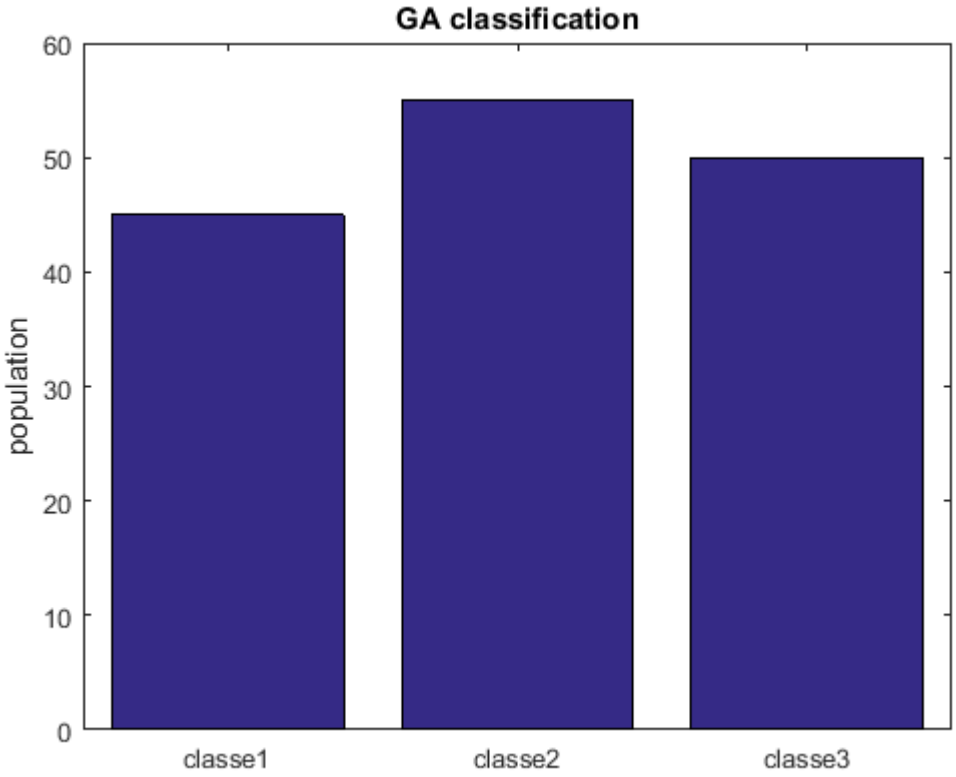


Figure III.7 Classification avec l'algorithme AG sur la base IRIS

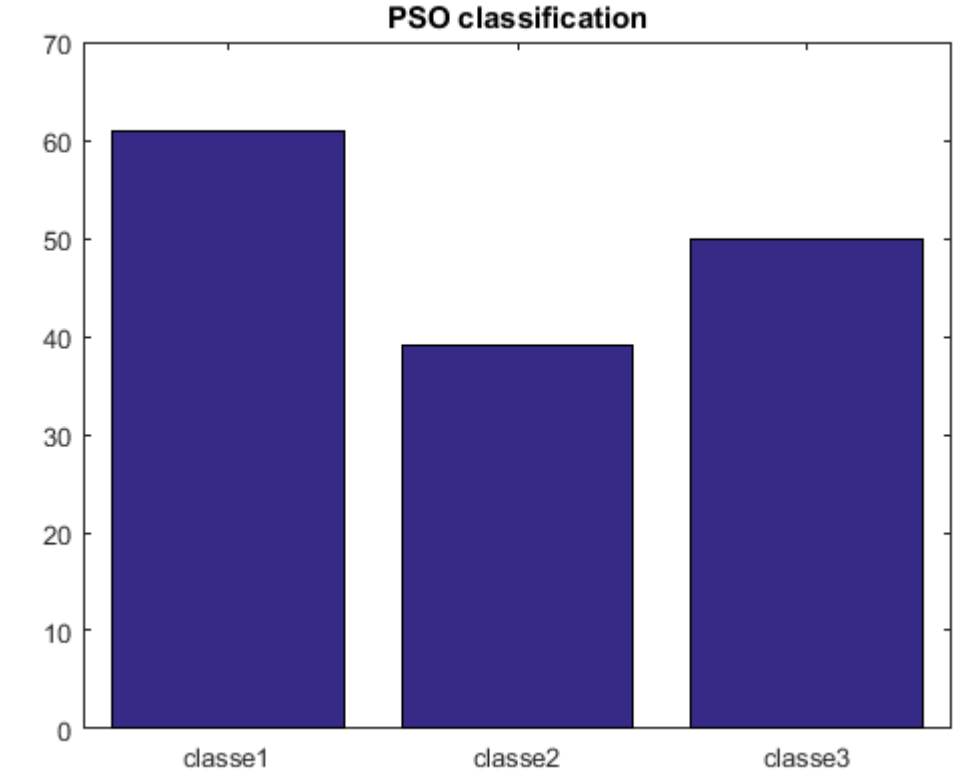


Figure III.8 Classification avec l'algorithme PSO sur la base IRIS

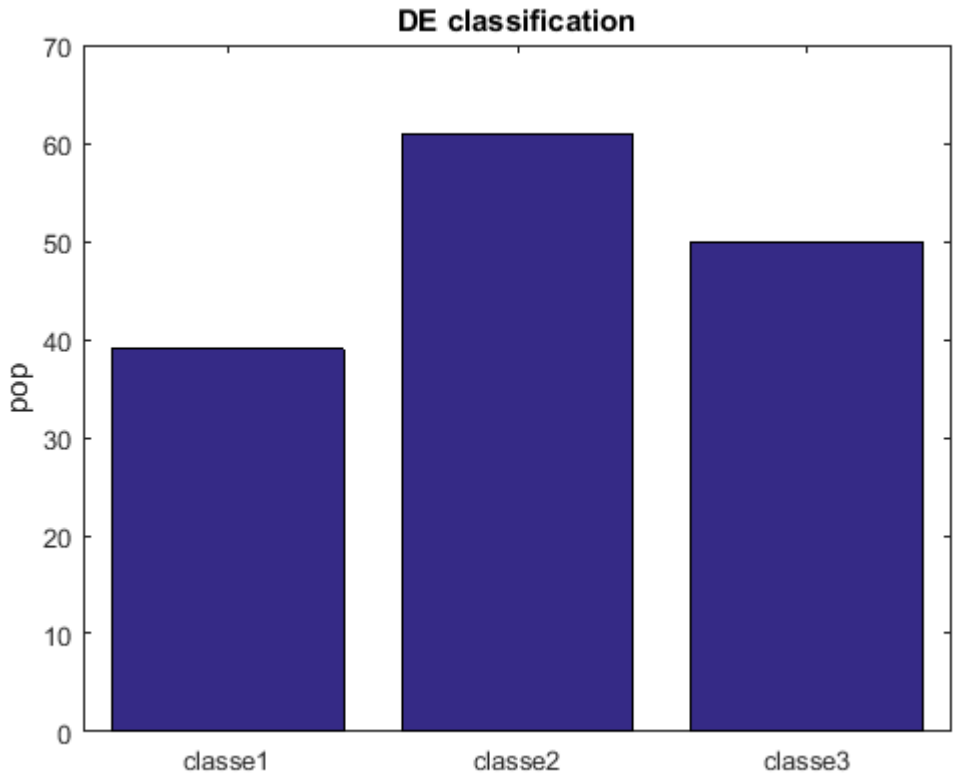


Figure III.9 Classification avec l'algorithme DE sur la base IRIS

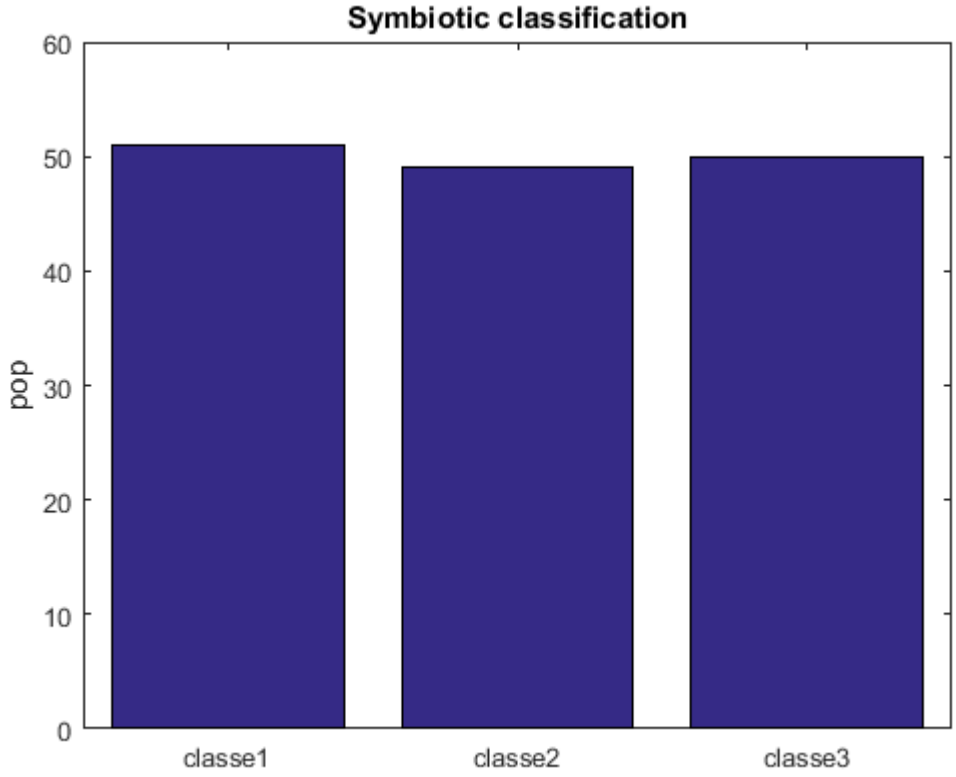


Figure III.10 Classification avec l'algorithme SOS sur la base IRIS

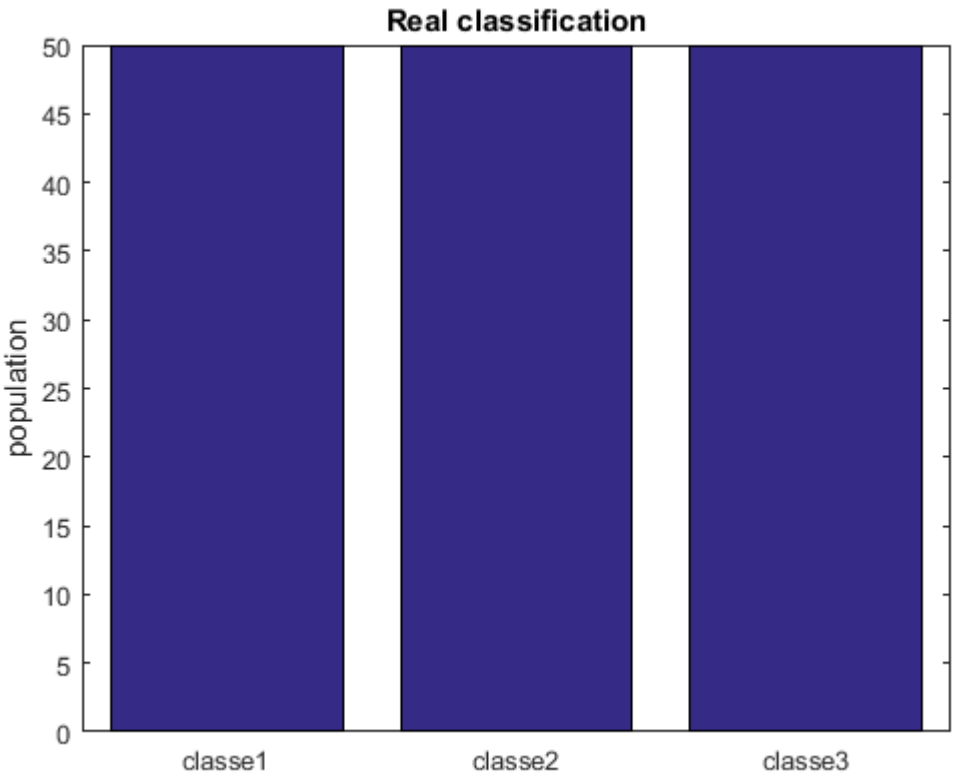


Figure III.11 Histogramme de classification réel d'IRIS

- **Application sur la base cancer du sein**

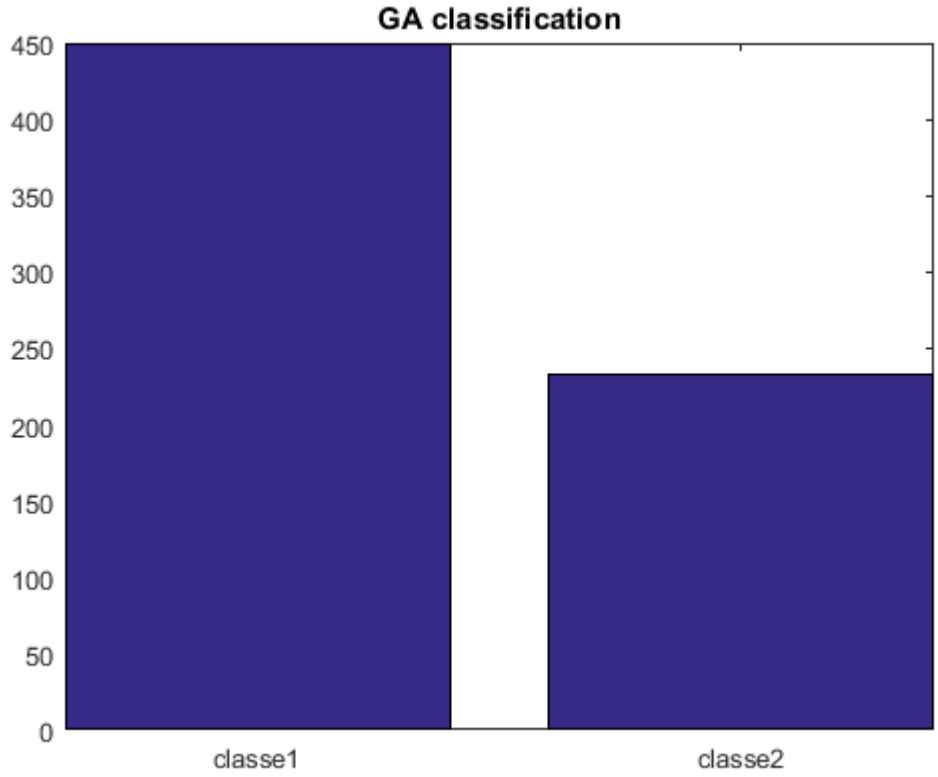


Figure III.12 Classification avec l'algorithme GA sur la base cancer du sein

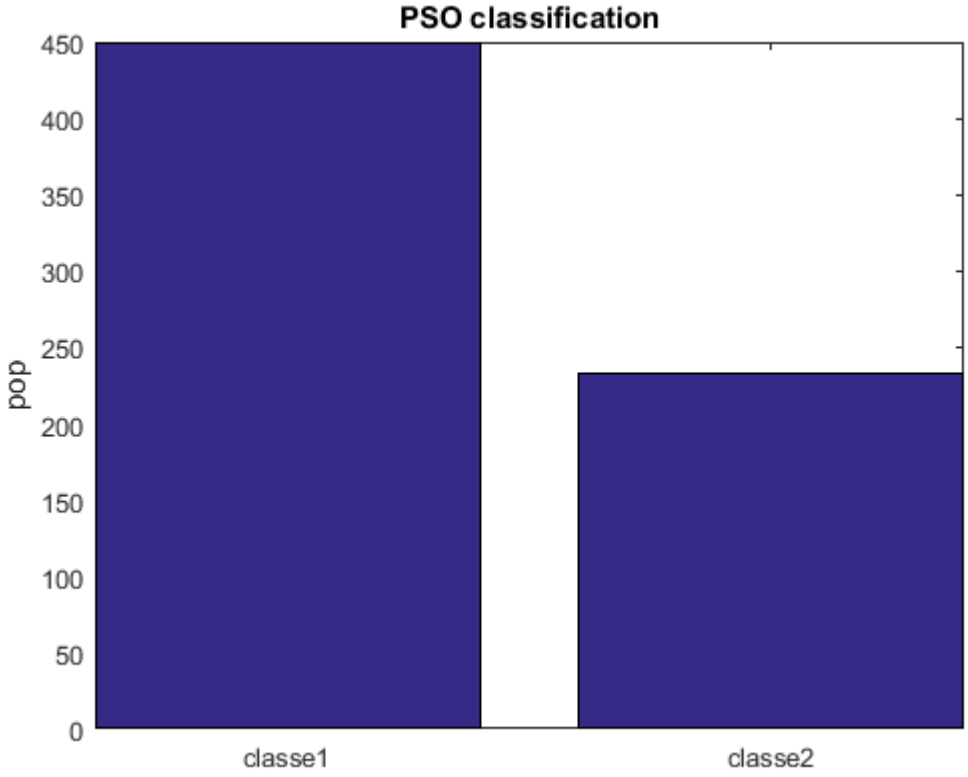


Figure III.13 Classification avec l'algorithme PSO sur la base cancer du sein

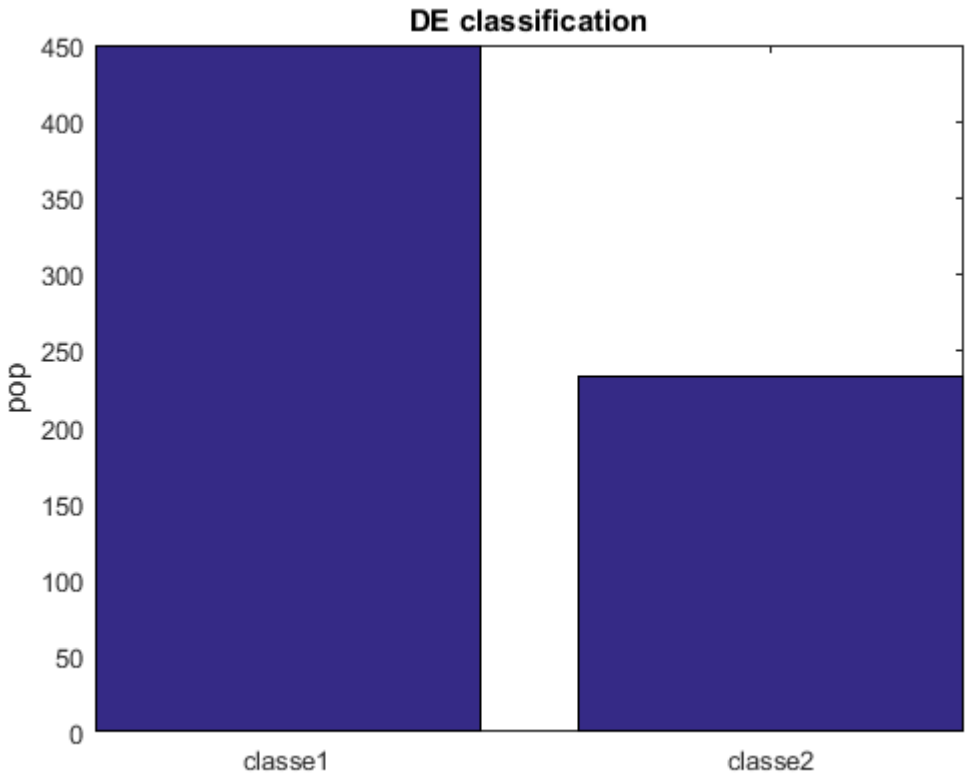


Figure III.14 Classification avec l'algorithme DE sur la base cancer du sein

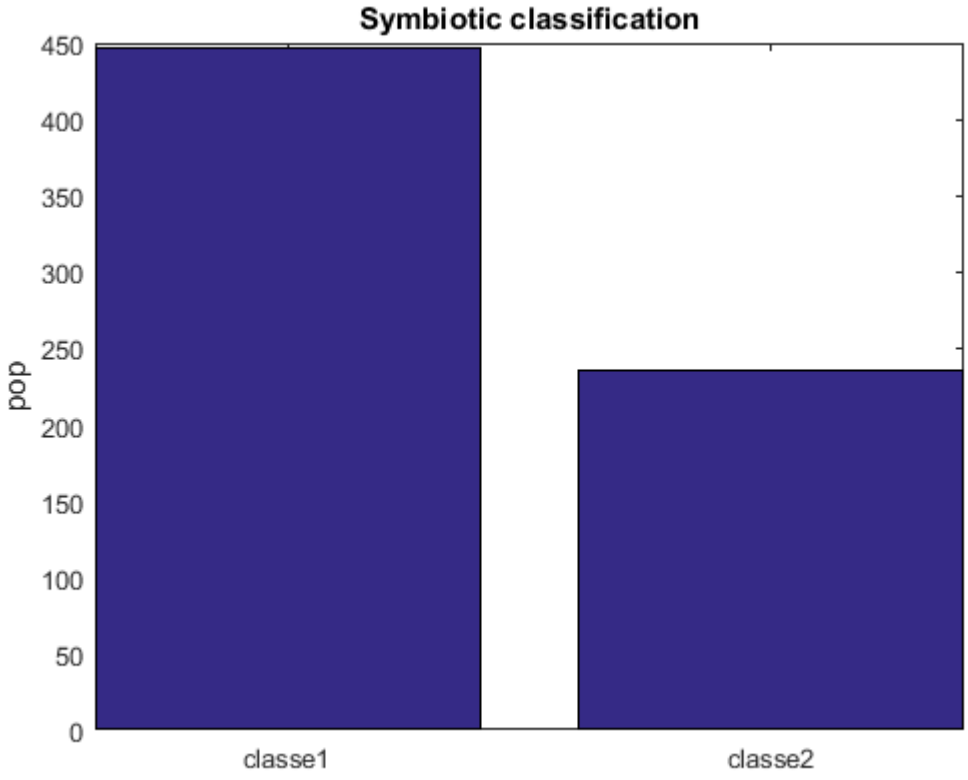


Figure III.15 Classification avec l'algorithme SOS sur la base cancer du sein

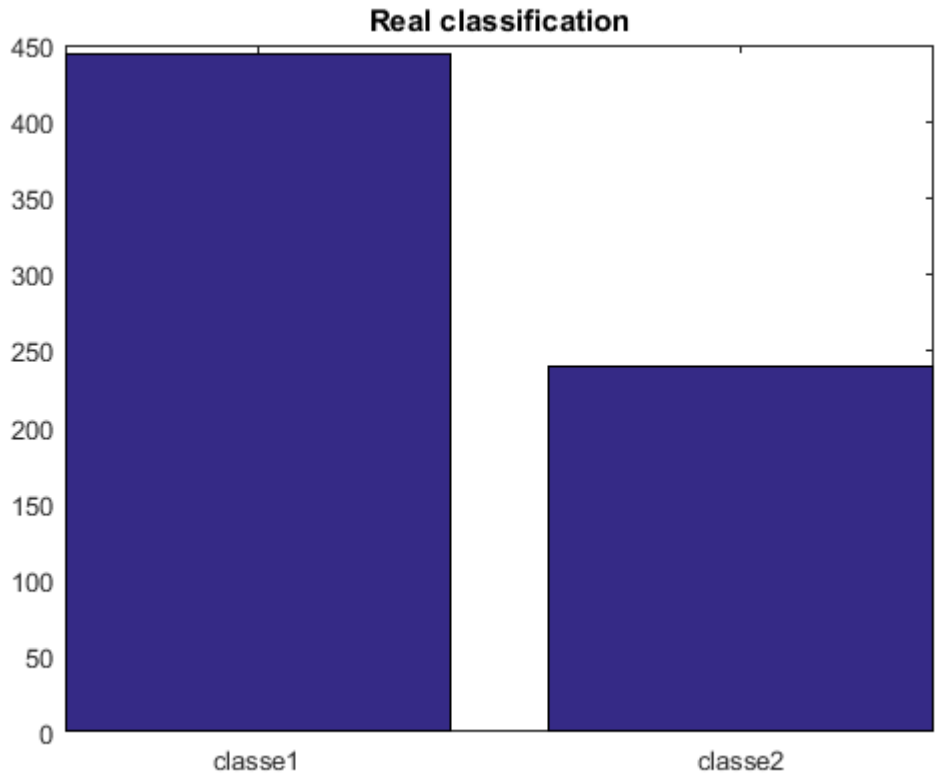


Figure III.16 Histogramme de classification réelle sur la base CANCER DE SEIN

- **Application sur la base de donneur du sang**

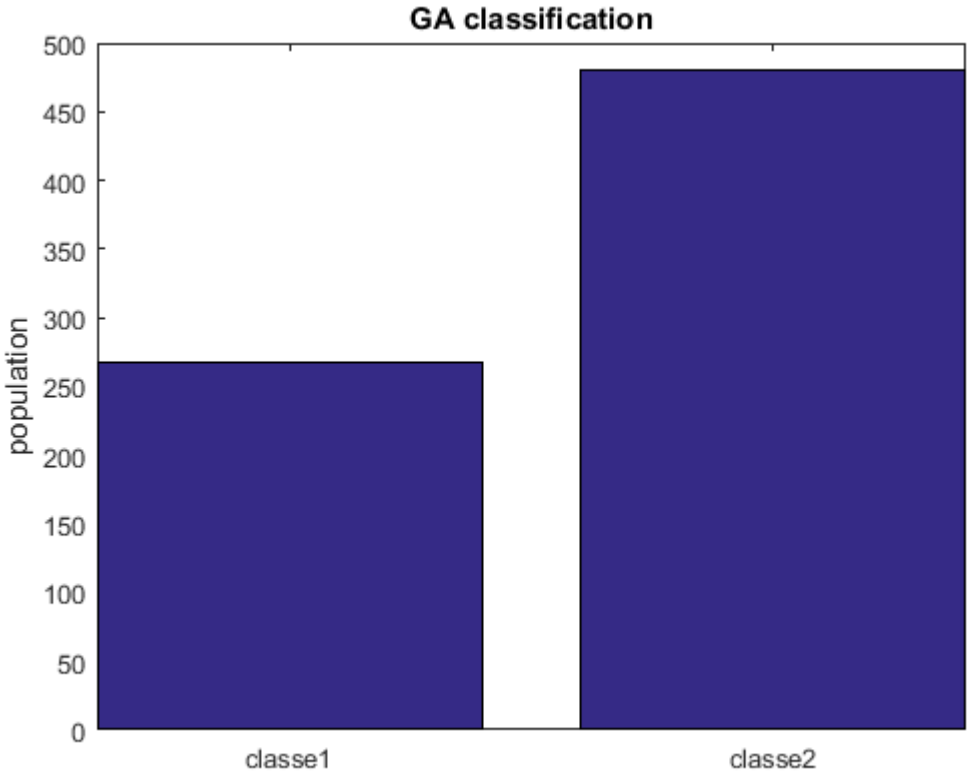


Figure III.17 Classification avec l'algorithme GA sur la base de donneur du sang

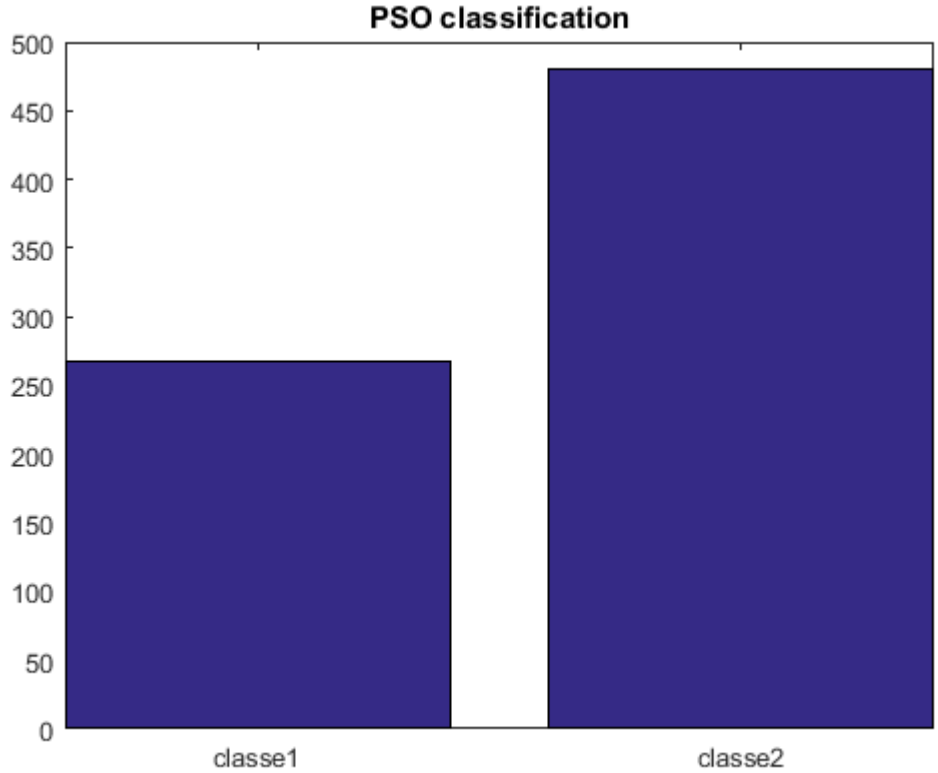


Figure III.18 Classification avec l'algorithme PSO sur la base de donneur du sang

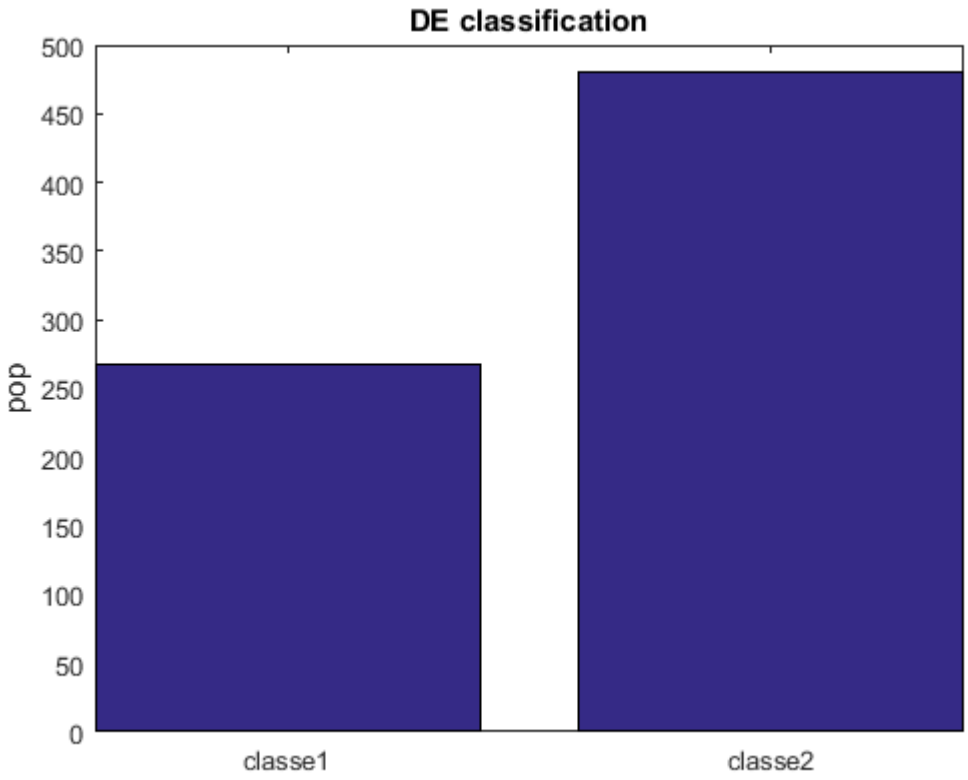


Figure III.19 Classification avec l'algorithme DE sur la base de donneur du sang

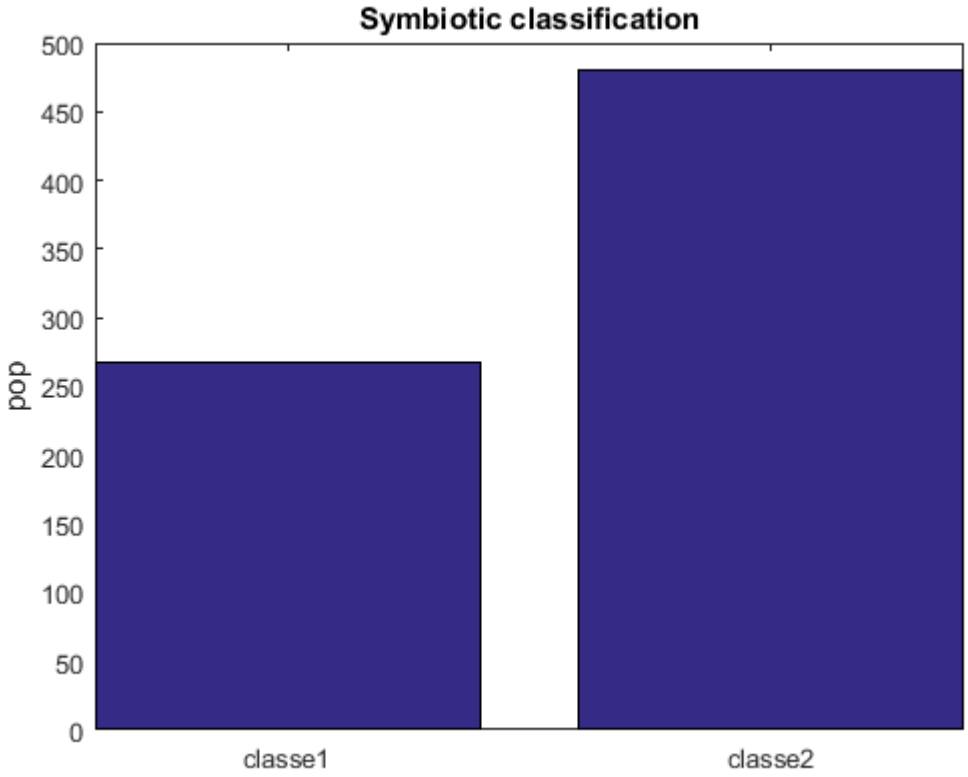


Figure III.20 Classification avec l'algorithme SOS sur la base des donneurs du sang

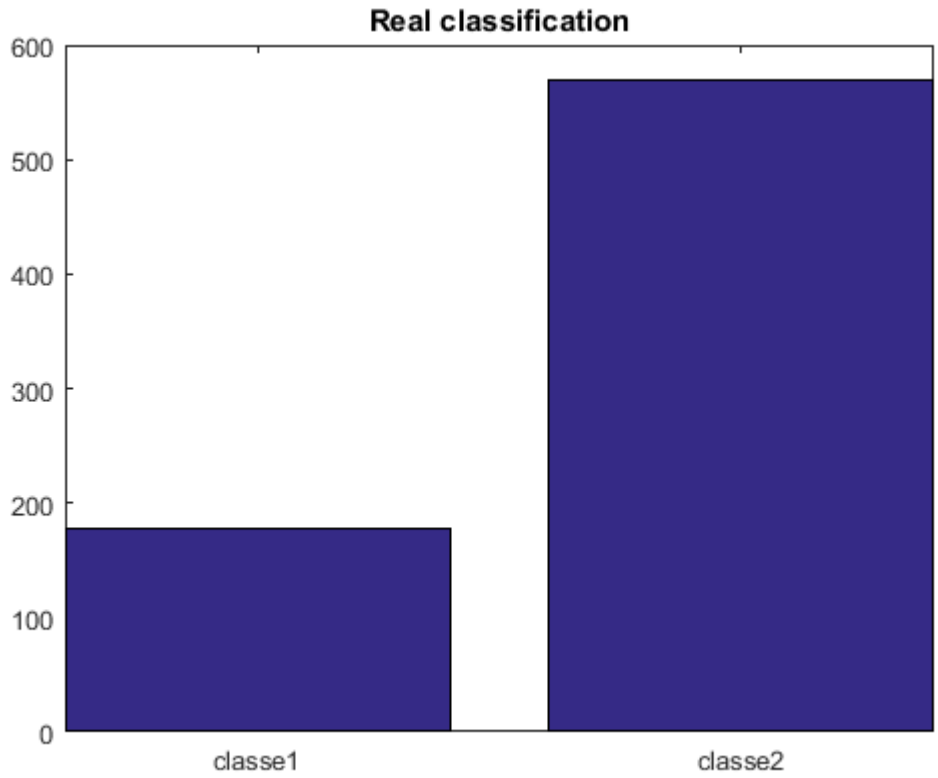


Figure. III.21 Histogramme de classification réelle de la base des donneurs du sang

La synthèse avec analyse de données nous donne le tableau récapitulatif suivant :

Tableau III.3 Répartition des nombres des instances de chaque classe par les quatre algorithmes

Les nombres d'instances dans chaque classe					
		GA	**PSO**	**DE**	**SOS**
IRIS	**Class1 n=50**	**45**	**39**	**39**	**51**
	Class2 n=50	**55**	**61**	**61**	**49**
	Class3 n=50	**50**	**50**	**50**	**50**
Cancer du sein	**Class1 n=446**	**450**	**450**	**450**	**448**
	Class2 n=237	**233**	**233**	**233**	**235**
Donneurs du sang	**Class1 n=179 (24%)**	**267**	**267**	**267**	**267**
	Class2 n=567 (76%)	**481**	**481**	**481**	**481**

III.3.5 Interprétation des résultats

A partir des résultats présentés sur les différents histogrammes de classification précédents, il est visible que (voit Tableau III.3) :

1- **Sur la base de données IRIS**

L'histogramme de SOS est presque identique au résultat réel cela signifier que SOS à donné une très bonne classification car un peu éléments sont mal classés, par contre les autres méthodes génèrent plus de nombres mal classés.

2- **Sur la base de données CANCER**

Le même constat de classification pour cette base avec les algorithmes de test, SOS fait apparaitre les meilleurs clustering par rapport aux trois autres car elle donne seulement quelques éléments mal classés malgré la complexité de cette base de données.

3- **Sur la base de données des donneurs de sang d'un mois**

Après l'exécution des quatre algorithmes sur cette base, nous avons constaté que les résultats sont similaires avec plusieurs éléments mal classés, cela est causé par la complexité de cette base car il y a un chevauchement entre les dates et les heures des donneurs de sang.

III.3.5.1 Matrice de confusion

Pour évaluer la qualité de la classification de SOS et les autres approches, nous avons utilisé la matrice de confusion (MC). La diagonale correspond toujours à la répartition des instances dans la classe correspondante **[37].**

- **Le taux d'exactitude (TEXA) ou de précision** : il correspond à l'ensemble des observations de bonne classification son expression est **[37]** :

  $TEXA=\sum diag(MC)/\sum_{ij} MC_{ij}$

  Tel que :

  $\sum diag(MC)$: c'est la somme des diagonales de la matrice de confusion des instances bien classées

  $\sum_{ij} MC_{ij}$: c'est la somme de tous les coefficients de la matrice de confusion

- **Le taux d'erreur (TERR)** : le taux d'erreur global correspond à la proportion des observations mal classées qui dépendent de la matrice de confusion, son expression est **[37]**:

  $TERR= 1 - \sum diag(MC)/\sum_{ij} MC_{ij}$

Le tableau III.4 les résultats de la matrice de confusion pour les trois bases de données utilisées par les quatre métaheuristiques.

	Le taux d'exactitude ou de précision en %			
	GA	**PSO**	**DE**	**SOS**
Base de données IRIS	40 .66	90	44.66	**90**
Base de données CANCER DU SEIN	96.48	95.31	95.75	**96.48**
Base de données DONNEURS DE SANG	65,106	65,106	65,106	**65,106**

III.3.5.2 Les courbes ROC

Pour bien visualiser les performances des modèles utilisés, nous avons tracés les courbes ROC qui utilisent la matrice de confusion pour la comparaison des modèles avec la classification réelle en faisant varier le seuil de 1 à 0 pour chaque cas **[38].**

Les courbes ROC pour la base de données IRIS

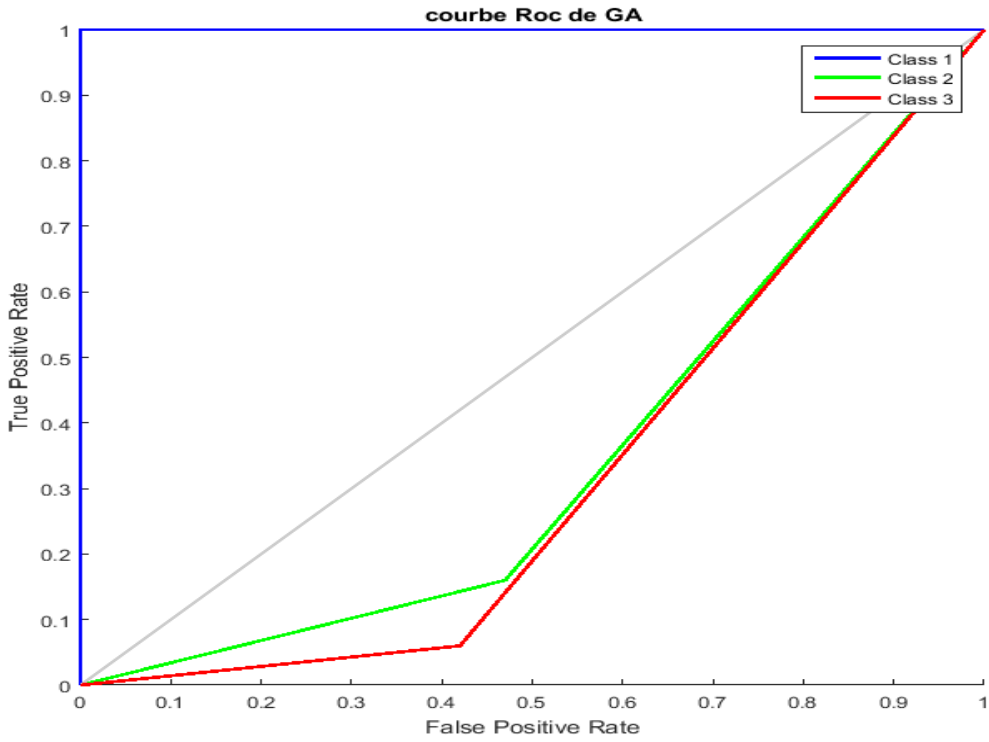


Figure III.22 La courbe ROC selon les trois classes avec GA pour la base de données IRIS

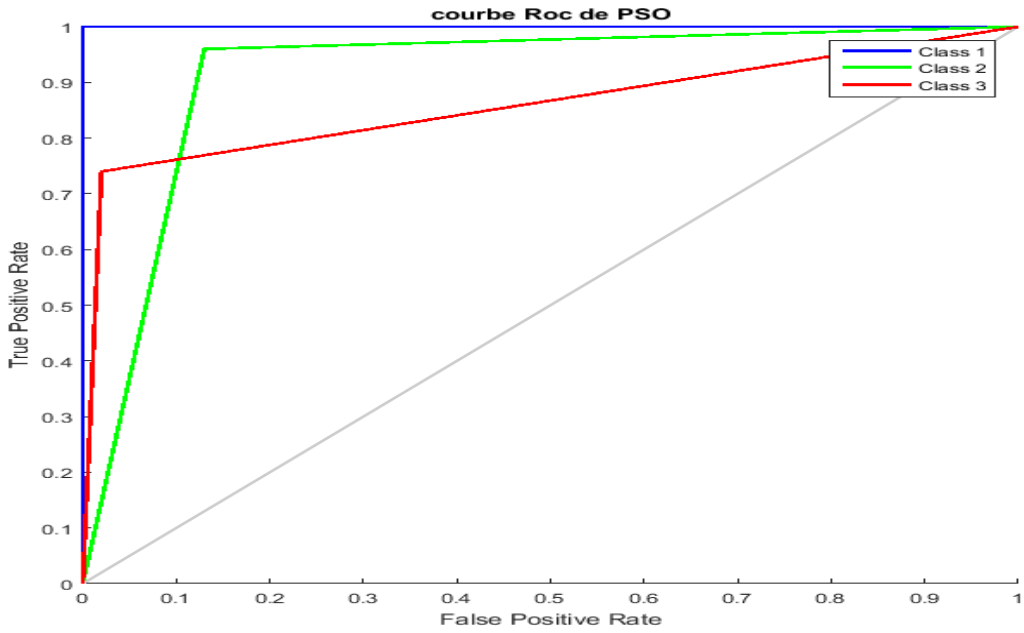


Figure III.23 La courbe ROC selon les trois classes avec PSO pour la base de données IRIS

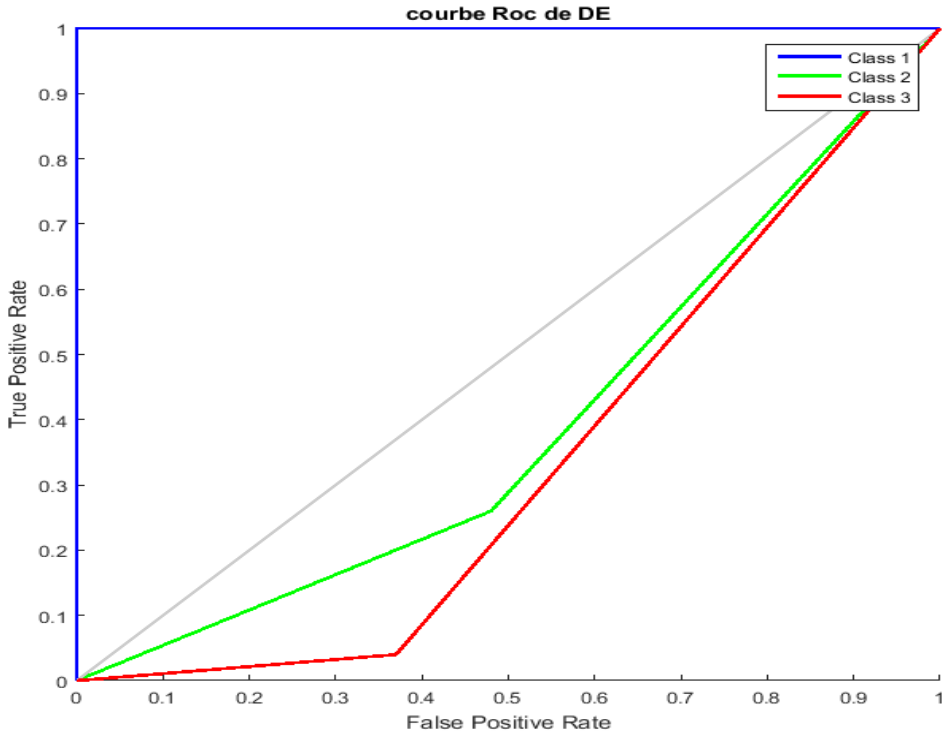


Figure III.24 La courbe ROC selon les trois classes avec DE pour la base de données IRIS

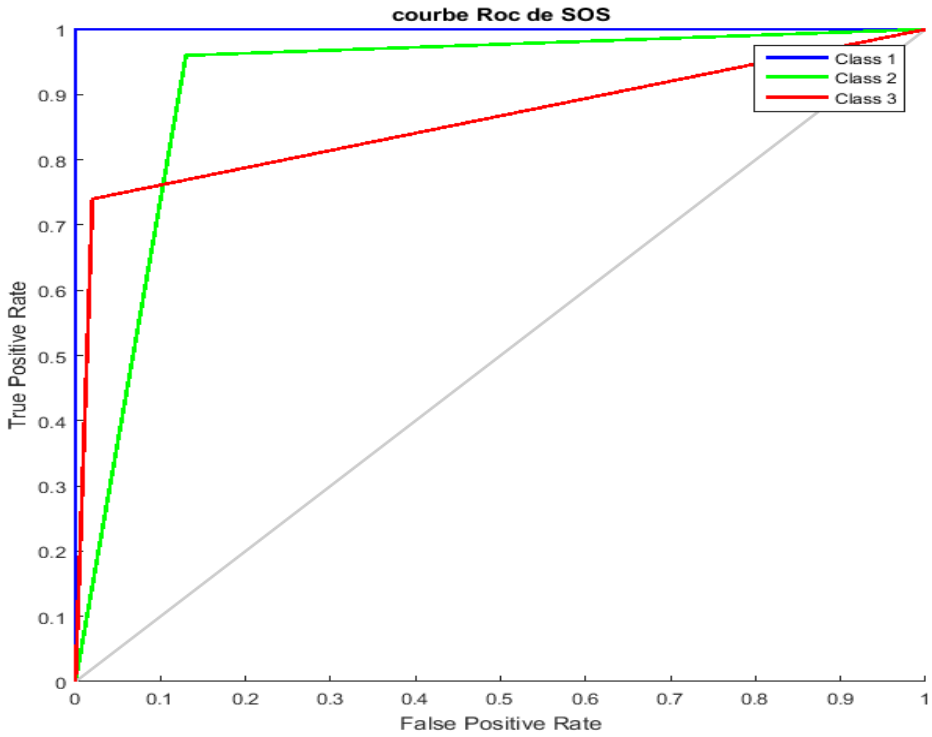


Figure III.25 La courbe ROC selon les trois classes avec SOS pour la base de données IRIS

Les courbes Roc pour la base de données Cancer du sein

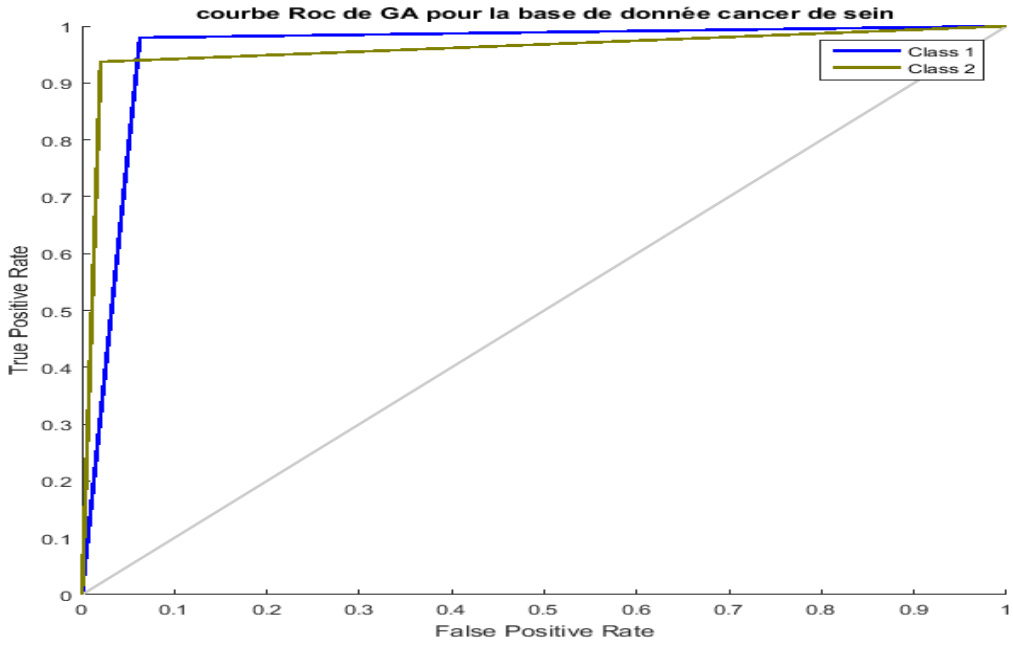


Figure III.26 La courbe ROC selon les deux classes avec GA pour la base de données Cancer du sein

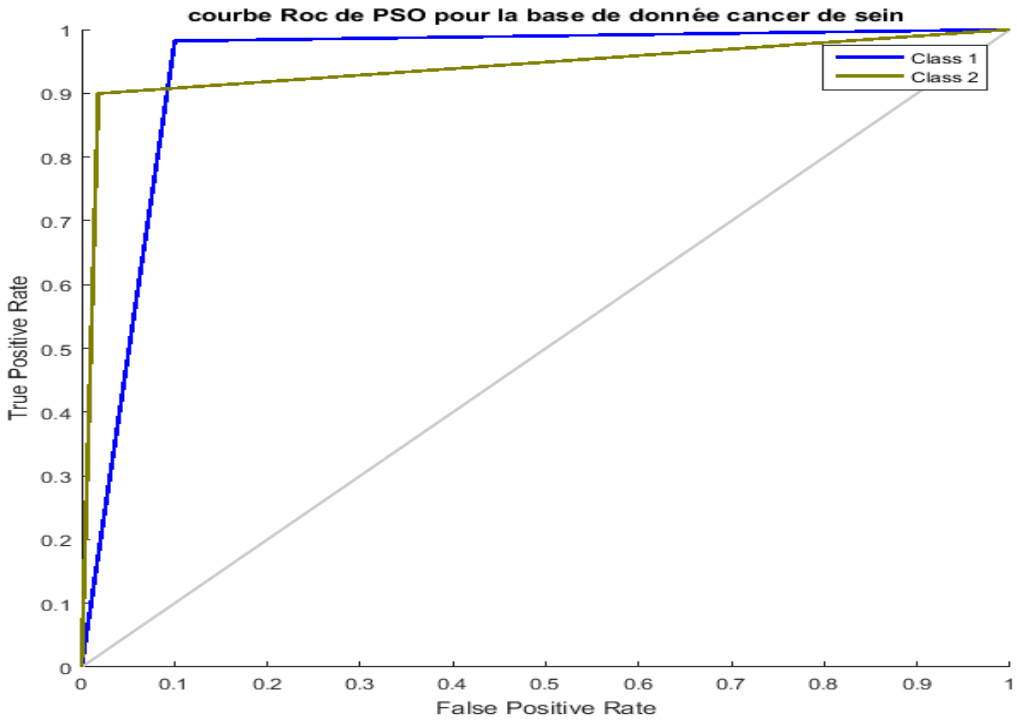


Figure III.27 La courbe ROC selon les deux classes avec PSO pour la base de données Cancer du sein

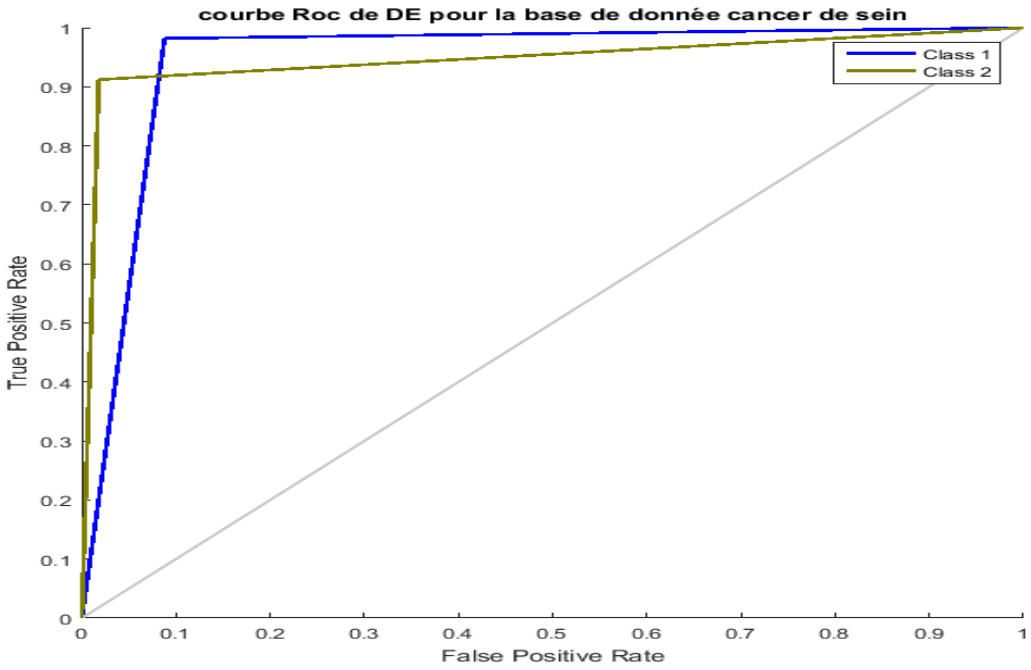


Figure III.28 La courbe ROC selon les deux classes avec DE pour la base de données Cancer du sein

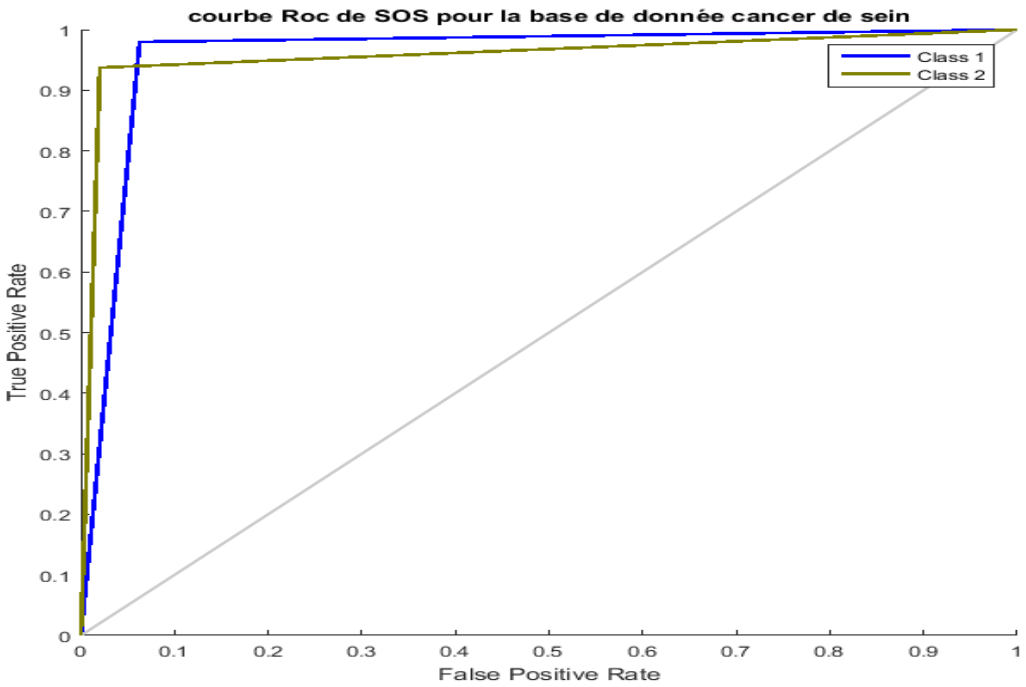


Figure III.29 La courbe ROC selon les deux classes avec SOS pour la base de données Cancer du sein

Les courbes ROC pour la base de données Donneurs de Sang

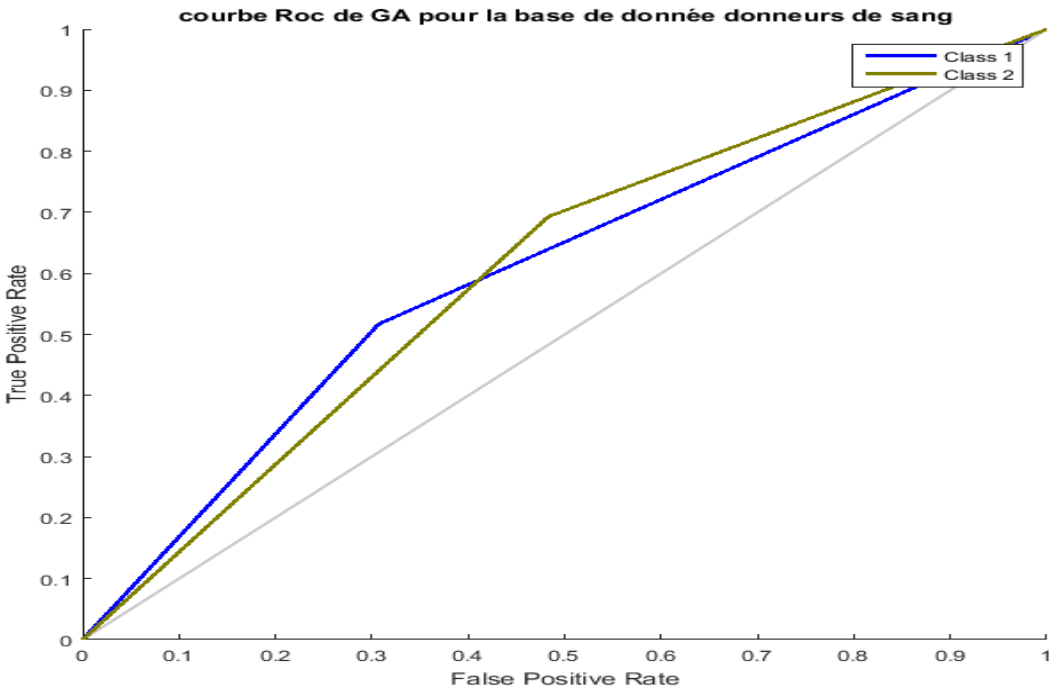


Figure III.30 La courbe ROC selon les deux classes avec GA pour la base de données Donneurs de Sang

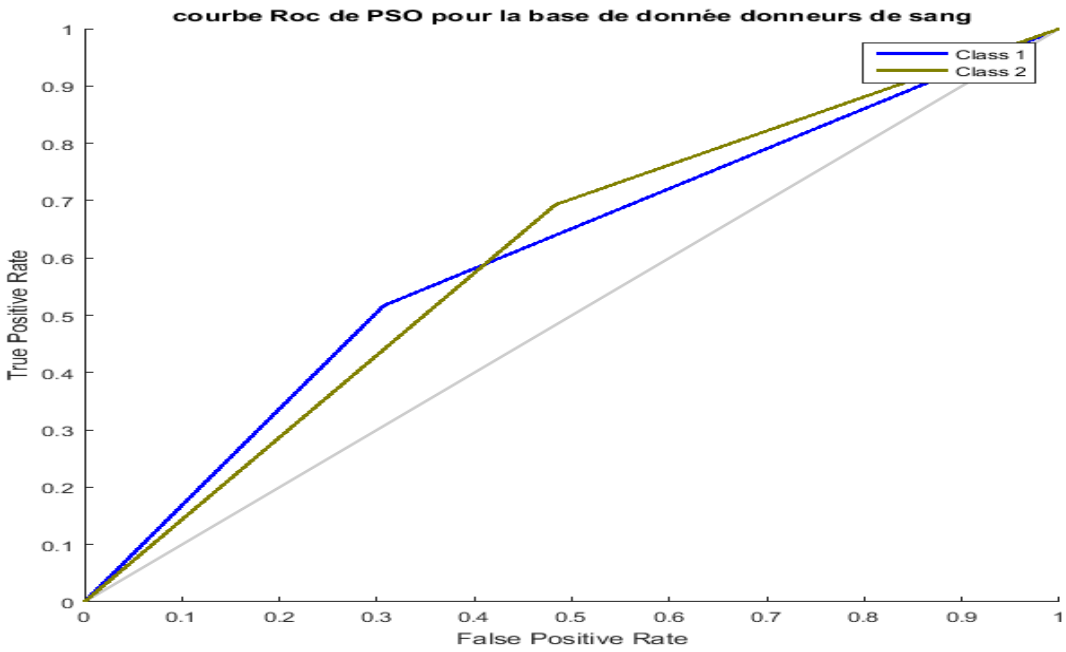


Figure III.31 La courbe ROC selon les deux classes avec PSO pour la base de données Donneurs de Sang

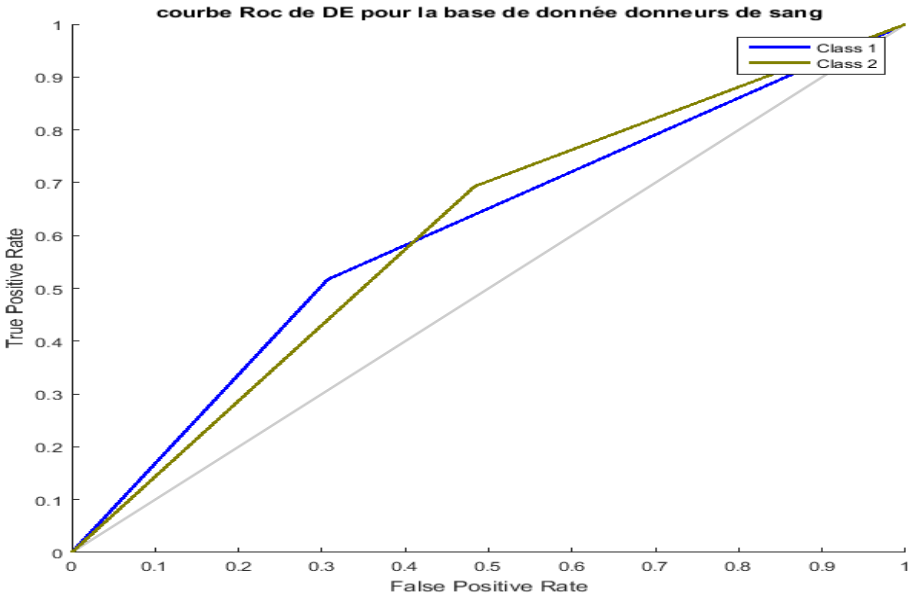


Figure III.32 La courbe ROC selon les deux classes avec DE pour la base de données Donneurs de Sang

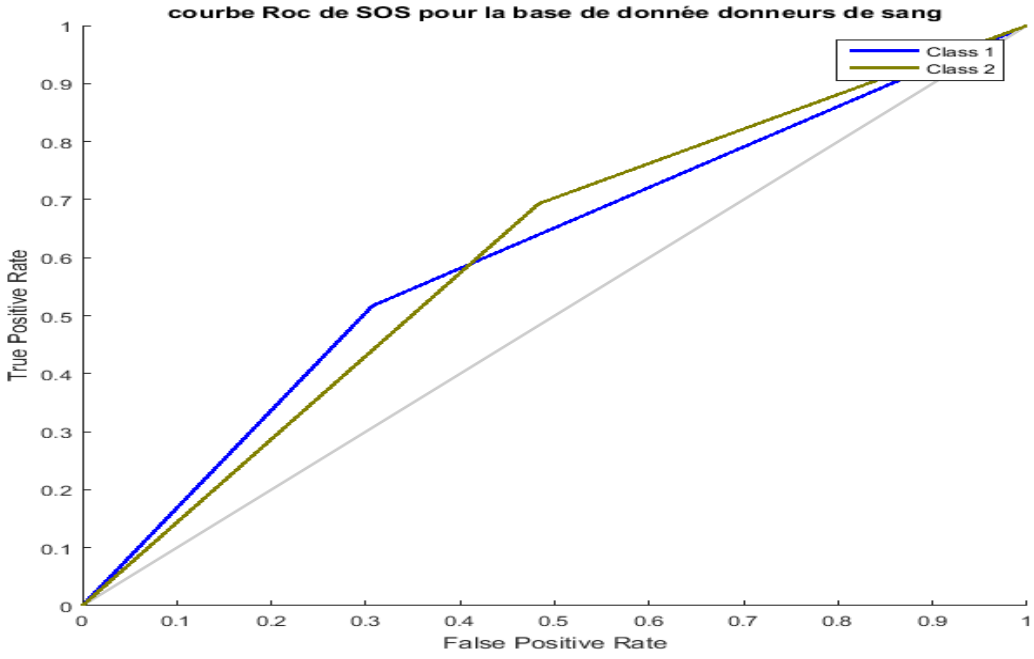


Figure III.33 La courbe ROC selon les deux classes avec SOS pour la base de données Donneurs de Sang

- ✓ **Base de données IRIS**
  - ❖ **L'interprétation de la matrice de confusion**

Nous constatons d'après le tableau III.4 que l'algorithme SOS donne les meilleurs résultats qui sont de l'ordre de 90% de même que le PSO. Cette particularité vient du fait que SOS donne dès la première exécution 90% et les mêmes résultats dans toutes les exécutions, par contre pour les autres algorithmes les résultats varient entre 40% et 90%. Cela est à cause de la fonction de fitness de SOS qui converge rapidement.

  - ❖ **L'interprétation de la courbe ROC**

La courbe ROC effectue une évaluation plus détaillée pour chaque classe de la base, l'algorithme SOS effectue une très parfaite classification pour une classe alors que pour les deux autres SOS a donné une bonne classification de données semblable à PSO, alors que les autres algorithmes AG et DE donnent un médiocre résultat.

- ✓ **Base de données Cancer du sein**
  - ❖ **L'interprétation de la matrice de confusion**

Pour cette base, et à partir du tableau III.4 nous remarquons que les quatre algorithmes donnent des résultats proches à SOS qui a 96.49% et AG avec 96.48%, DE avec 95.75% et PSO qui a 95.31%. Nous signalons que SOS donne toujours dès le départ le meilleur résultat et une convergence rapide de sa fonction de fitness par rapport aux autres.

  - ❖ **L'interprétation de la courbe ROC**

Notre algorithme SOS ainsi que AG donnent une très bonne classification mais les autres algorithmes DE et PSO sont moins significatif.

- ✓ **Donneurs de sang**
  - ❖ **L'interprétation de la matrice de confusion**

Pour cette base, nous constatons que les quatre algorithmes ont donné les mêmes résultats qui sont égal à 65.106%, sachant que SOS a eu une convergence rapide de sa fonction de fitness par rapport aux autres. Ce résultat moyen est la cause de la complexité de cette base.

❖ **L'interprétation de la courbe ROC**

Pour cette qui est difficile à manipuler car ces données sont chevauchées et la base est rarement utilisée, les quatre algorithmes ont donné la même courbe ROC qui représente une classification assez bonne (> la droite diagonale).

III.4 Conclusion

Dans ce chapitre nous avons commencé par une présentation de l'approche symbiotique à travers son aspect fonctionnel sous forme d'organigramme pour bien illustrer les détails de son implémentation.

Par la suite, nous avons explicités les différentes fonctions de fitness nécessaires pour l'évaluation des différentes techniques qui nous ont servis comme modèle de comparaison, il s'agit de l'algorithme PSO, l'algorithme ABC et aussi l'algorithme CA suivi du SCA et GWO.

A travers les expérimentations utilisées, il est clair que la méthode symbiotique possède par rapport aux autres méthodes un aspect d'exploitation très grand lui permettant d'avoir la valeur exacte dans la majorité des cas pratiques réalisés.

Par contre, dans le comportement exploratoire, elle semble moins forte pour certaine fonctions de type multimodale à petite dimension.

Par ailleurs, on note que cet algorithme est plus performant et donne de meilleurs résultats pour les problèmes de grande dimension.

Pour mettre en valeur la robustesse de l'algorithme SOS, le domaine de valorisation et d'expérimentation de ce dernier est le datamining.

D'après les résultats de la matrice de confusion nous constatons que l'algorithme SOS converge rapidement vers la solution optimale et reste stable en cette valeur contrairement aux autres algorithmes.

Bien que SOS fait ses premier pas et que juste sa version de base a été utilisée, contrairement aux autres algorithmes AG, PSO et DE avec leurs versions évolutives, les résultats obtenus montre que SOS est l'algorithme le premier en convergence

Nous signalons et insistant en conséquence que l'algorithme SOS donne un résultat de datamining assez consistant car l'interprétation significative des résultats est très facile à être exploiter pour prendre une décision.

L'extraction de connaissances au sens de datamining est sémantiquement clairement sur ces résultats en matière de lecture et d'analyse de données surtout sur les histogrammes qui montrent de très bons résultats de classification, dans notre projet cette classification est de type supervisée.

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي و البحث العلمي
Ministère de l'enseignement Supérieur et de la Recherche scientifique

Université Mohamed Khider Biskra
Faculté des Sciences et de la Technologie
Département de Génie Electrique
Filière : Electronique
Option : Système embarqué

Thème :

Algorithme de recherche par organisme symbiotique : étude et application

Proposé par : TOUMI ABIDA
Dirigé par : TOUMI ABIDA

RESUMES (Français et Arabe)

Dans le domaine de la résolution de problèmes d'optimisation combinatoires, plusieurs approches bio-inspirées ont fait preuve de capacité et de robustesse en matière de résolution de cette catégorie de problèmes. Elle sont un ensemble de techniques dites métaheuristiques. Certaines de ces métaheuristiques sont parfois très utiles en l'absence de solution algorithmique pour une solution approchée et optimisée via un ensemble de contraintes et une fonction à optimiser. Dans ce projet, nous nous intéressons à l'utilisation d'une nouvelle métaheuristique inspirée par un phénomène naturel à savoir la symbiotique appelée en anglais Symbiotic

Organisms Search (SOS). C'est un algorithme qui simule le comportement interactif parmi les organismes dans la nature.

Dans ce projet, nous avons évalué SOS avec d'autres métaheuristiques comme l'algorithme d'essaim de particules, l'algorithme génétique, l'algorithme à évolution différentielle, l'algorithme de Sine Cosine et l'algorithme culturel, nous avons constaté que SOS est excellent. Pour montrer que SOS est le meilleur pour la classification, nous l'avons appliqué sur le champ datamining.

Mots clés : symbiotic (symbiotique), symbiose, datamining, métaheuristique, optimisation, classification.

الملخص

في مجال حل المشكلة التحسين الاندماجي ، أدت النهج المستوحاة من الظواهر الطبيعية إلى ظهور عائلة من metaheuristiques بعض منها مفيدة للغاية للحصول على الحل التقريبي والأمثل عن طريق مجموعة من الشروط على عملية ووظيفة التحسين أو التقليل . في هذا المشروع، نحن مهتمون باستخدام metaheuristique جديدة مستوحاة من ظاهرة طبيعية وهي التعايش و التكافل بين الكائنات الحية المتشابهة التي نرمز لها ب (SOS) .

هذه الخوارزمية تحاكي السلوك التفاعلي بين الكائنات الحية في الطبيعة. و لقد قمنا بتقييم SOS مع غيرها من عائلة metaheuristiques مثل خوارزمية سرب الجسيمات ، الخوارزمية الجينية ، خوارزمية التطور التفاضلي ، خوارزمية جيب التمام ، والخوارزمية الثقافية ووجدناها ممتازة .لإظهار أن SOS هو الأفضل للتصنيف ، قمنا بتطبيقه على حقل تحليل البيانات.

الكلمات الدالة : تكافلية، تكافل، تحليل البيانات، métaheuristique ، تحسينات ، تصنيف.

Résumé

Dans le domaine de la résolution de problèmes d'optimisation combinatoires, plusieurs approches bio-inspirées ont fait preuve de capacité et de robustesse en matière de résolution de cette catégorie de problèmes. Elles sont un ensemble de techniques dites métaheuristiques. Certaines de ces métaheuristiques sont parfois très utiles en l'absence de solution algorithmique pour une solution approchée et optimisée via un ensemble de contraintes et une fonction à optimiser. Dans ce projet, nous nous intéressons à l'utilisation d'une nouvelle métaheuristique inspirée par un phénomène naturel à savoir la symbiotique appelée en anglais Symbiotic Organisms Search (SOS). C'est un algorithme qui simule le comportement interactif parmi les organismes dans la nature.

Dans ce projet, nous avons évalué SOS avec d'autres métaheuristiques comme l'algorithme d'essaim de particules, l'algorithme génétique, l'algorithme à évolution différentielle, l'algorithme de Sine Cosine et l'algorithme culturel, nous avons constaté que SOS est excellent. Pour montrer que SOS est le meilleur pour la classification, nous l'avons appliqué sur le champ datamining.

Mots clés : symbiotic (symbiotique), symbiose, datamining, métaheuristique, optimisation, classification

الملخص

في مجال حل المشكلة التحسين الاندماجي ، أدت النهج المستوحاة من الظواهر الطبيعية إلى ظهور عائلة من metaheuristiques بعض منها مفيدة للغاية للحصول على الحل التقريبي والأمثل عن طريق مجموعة من الشروط على عملية ووظيفة التحسين أو التقليل . في هذا المشروع، نحن مهتمون باستخدام metaheuristique جديدة مستوحاة من ظاهرة طبيعية وهي التعايش و التكافل بين الكائنات الحية المتشابهة التي نرمز لها ب (SOS).

هذه الخوارزمية تحاكي السلوك التفاعلي بين الكائنات الحية في الطبيعة. و لقد قمنا بتقييم SOS مع غيرها من عائلة metaheuristiques مثل خوارزمية سرب الجسيمات ، الخوارزمية الجينية ، خوارزمية التطور التفاضلي ، خوارزمية جيب التمام ، والخوارزمية الثقافية ووجدناها ممتازة .لإظهار أن SOS هو الأفضل للتصنيف ، قمنا بتطبيقه على حقل تحليل البيانات.

الكلمات الدالة : تكافلية، تكافل، تحليل البيانات، métaheuristique ، تحسينات ، تصنيف.

CONCLUSION GENERALE

En conclusion de ce projet qui consistait à une modélisation d'une approche pour la résolution de problèmes d'optimisation combinatoire à savoir le datamining, nous dressons une synthèse comportant les différentes parties développées.

Dans un premier temps, notre étude s'est portée sur la rédaction d'un état de l'art sur les méthodes dans la littérature permettant de résoudre les problèmes d'optimisation difficiles à savoir les métaheuristiques.

Dans cette partie, un regard particulier a concerné un nouvel algorithme métaheuristique appelé Symbiotic Organisms Search (SOS) inspiré des interactions biologiques entre les organismes dans un écosystème.

SOS est un modèle naturel qui utilise trois stratégies : mutualisme, commensalisme et parasitisme.

Dans la seconde partie de ce mémoire, nous nous sommes intéressés à la description de notre champ d'application, il s'agit du datamining.

L'intérêt porté sur ce domaine est justifié d'une part par les insuffisances constatées à travers l'application des techniques d'optimisations existantes, d'autre part, le processus de datamining est un type de problème d'optimisation combinatoire.

Nous signalons aussi que le datamining est présent dans beaucoup de domaine technologique comme les réseaux sociaux, l'industrie, la santé…

Pour la mise en valeur de nos choix théoriques basés sur l'exploitation de l'algorithme SOS pour le datamining, nous avons consacrés la troisième partie de ce mémoire aux points suivants :

- Evaluation de SOS sur des fonctions de test ainsi que sa comparaison avec d'autres métaheuristiques. Le constat fait preuve de la capacité de SOS de générer des solutions avec une qualité significativement meilleure aux méthodes concurrentes.
- Application de SOS dans le domaine de datamining en utilisant trois bases de données public de différentes complexités. Les résultats obtenus démontrent que SOS est encore performant pour la classification des occurrences.

A la fin de cette conclusion générale, nous signalons que le nouvel algorithme SOS est robuste, il est facile à mettre en œuvre. Nous notons aussi sa capacité de résoudre divers problèmes d'optimisation numérique malgré l'utilisation de moins de paramètres de contrôle que les algorithmes concurrents.

Nous proposons également des perspectives à la fin de ce travail qui nous semble très prometteuses et qui sont comme suit :

- amélioration de SOS pour le traitement de sa phase d'exploitation ;
- dans le datamining nous pourrons exploiter l'approche non supervisée ;
- application de notre proposition dans d'autres domaines comme textemining, l'analyse et la classification de courrier électronique, l'analyse de données dans les réseaux sociaux ;
- enfin l'hybridation de SOS avec d'autres métaheuristiques.