

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR
ET DE LA RECHERCHE SCIENTIFIQUE
UNIVERSITE MENTOURI CONSTANTINE
FACULTE DES SCIENCES DE L'INGENIEUR
DEPARTEMENT D'ELECTRONIQUE

N° d'ordre :
Série :

Mémoire
Présenté pour obtenir le diplôme de
Magister En Electronique
Option : contrôle

Par M^{elle} : Terki Amel

THEME

**Analyse des performances des algorithmes génétiques
utilisant différentes techniques d'évolution de la
population**

Devant le jury :

| | | | |
|--------------------------|---------------------------------------|-------------------|---------------------------|
| <u>Président</u> | : M^r K. BELARBI | Professeur | Univ. Constantine. |
| <u>Rapporteur</u> | : M^{elle} N. MANSOURI | Professeur | Univ. Constantine. |
| <u>Examineurs</u> | : M^r A.BENNIA | Professeur | Univ. Constantine. |
| | M^{elle} F. HACHOUF | M.C | Univ. Constantine. |

Soutenu le : / /

SOMMAIRE

| | |
|---|----|
| INTRODUCTION GENERALE----- | 1 |
| <i>CHAPITRE I : LES ALGORITHMES EVOLUTIONNAIRES</i> | |
| I. Introduction ----- | 3 |
| II. Les algorithmes évolutionnaires (AE)----- | 5 |
| III. La boucle évolutionnaire ----- | 8 |
| VI. Application des algorithmes évolutionnaires----- | 10 |
| V. Les algorithmes génétiques (AG)----- | 11 |
| V.1/ Introduction----- | 11 |
| V.2/ Les outils évolutionnaires de base d'un (AG)----- | 12 |
| V.3/ Optimisation par les algorithmes génétique ----- | 13 |
| V.4/ Mécanismes de fonctionnement d'un (AG)----- | 14 |
| 1°/ Initialisation de la population----- | 15 |
| 2°/Codage et décodage des variables----- | 16 |
| 3°/ La fonction d'adaptation----- | 19 |
| 4°/ La sélection des parents----- | 20 |
| 5°/ La recombinaison génétique----- | 23 |
| a)-Le Croisement----- | 23 |
| b)-La mutation----- | 25 |
| 6°/La sélection finale----- | 27 |
| a)-La sélection par descendance----- | 27 |
| b)- La sélection par compétition----- | 27 |
| c)- La sélection de procréation sélective----- | 27 |
| IV Conclusion----- | 28 |
| <i>CHAPITRE II : AMELIORATION DE L'ALGORITHME GENETIQUE</i> | |
| I. Introduction----- | 29 |
| II. Les mécanismes biologiques pouvant assurer la diversité des individus----- | 30 |
| III. La transformation----- | 30 |
| III.1/Principe----- | 30 |
| III.2/Algorithme----- | 32 |
| III.3/Organigramme----- | 32 |
| III.4/Influence des paramètres intervenant dans la transformation des gènes sur les performances de l'algorithme----- | 33 |

| | |
|---|----|
| a)-Influence de la longueur du segment de gènes----- | 33 |
| b)- Influence du taux de transformation----- | 36 |
| VI. La transposition----- | 37 |
| VI.1/Principe----- | 37 |
| VI.2/Construction du transposon----- | 38 |
| VI.3/ Transposition simple----- | 39 |
| VI.4/ Transposition basée sur un tournoi----- | 40 |
| VI.5/ Transposition asexuée----- | 40 |
| VI.6/ Etude de l'influence de la longueur de la séquence du segment de gènes dans la transposition----- | 42 |
| VI.7/ Etude de la convergence des différents algorithmes de la transposition----- | 49 |
| V. Stratégie de la recherche de niche----- | 52 |
| V.1/Introduction----- | 52 |
| V.2/ Recherche des niches----- | 53 |
| V.3/ L'évolution des niches----- | 54 |
| V.4/Algorithme----- | 55 |
| V.5/ Les Niches et l'évolution des individus----- | 56 |
| V.6/ La recherche de niche et l'opérateur de méiose----- | 56 |
| IV. Etude de la convergence des différents algorithmes en utilisant la stratégie de la recherche des niches----- | 58 |
| IIV. Conclusion----- | 62 |

CHAPITRE III : OPTIMISATION DES SYSTHMES BIOLOGIQRUS

NON LINEAIRES

| | |
|---|----|
| I.Introduction----- | 63 |
| II. mécanismes de catalyse enzymatique----- | 63 |
| III. Le complexe enzyme – substrat----- | 65 |
| a / Les substrats----- | 65 |
| b/ Les membranes----- | 66 |
| VI.Optimisation d'un système enzymatique à deux compartiments et un seul substrat --- | 66 |
| V. Optimisation d'un réacteur de d'écoulement de prise----- | 73 |
| IV.Conclusion----- | 78 |
| CONCLUSION GENERAL----- | 79 |

BIBLIOGRAPHIE

INTRODUCTION GENERALE

L'optimisation est l'une des branches les plus importantes des mathématiques appliquées, et de nombreuses recherches, à la fois pratiques et théoriques, lui sont consacrées. Il existe deux grandes approches de l'optimisation. L'une est dite déterministe : les algorithmes de recherche utilisent toujours le même cheminement pour arriver à la solution, et on peut donc déterminer à l'avance les étapes de la recherche, L'autre est aléatoire : pour des conditions initiales données, l'algorithme ne suivra pas le même cheminement pour aller vers la solution, et peut même proposer différentes solutions. C'est vers cette seconde approche, que va s'orienter notre travail, et plus particulièrement vers un type bien précis d'algorithme de recherche aléatoire, les algorithmes du type évolutionnaires.

Les algorithmes évolutionnaires représentent un outil important pour la résolution des problèmes d'optimisation. D'ailleurs, ils sont de plus en plus utilisés dans de multiples domaines. Ils sont faciles à mettre en œuvre et fournissent d'excellentes performances à de faibles coûts.

Les algorithmes génétiques font partie de cette famille, ils permettent d'explorer des domaines possédant de très nombreuses solutions, et leur efficacité pratique a été prouvée bien avant que les résultats de convergence théorique soient établis. Toutefois le choix des nombreux opérateurs génétiques, intervenant dans la mise en place de l'algorithme reste à l'appréciation de l'utilisateur, c'est pour cela qu'un domaine de recherche très actif est consacré à l'étude de ces derniers et à la mise en place de nouvelles techniques. Surtout que l'utilisation de ces algorithmes est souvent coûteuse en temps de calculs et les performances de ces algorithmes dépendent beaucoup des différents opérateurs génétiques.

Dans le cadre de ce mémoire, nous nous sommes intéressés à l'étude d'algorithmes génétiques utilisant de nouveaux mécanismes inspirées de processus biologiques, tels que la transformation, la transposition génétique et la stratégie de la

recherche de niche et leur incidence sur leurs performances de l'algorithmes d'optimisation.

Pour confirmer l'efficacité de ces nouveaux mécanismes, nous nous sommes intéressés aux problèmes d'optimisation des systèmes dynamiques non linéaires modélisés par des équations du type équations différentielles.

Le travail est présenté en trois chapitres:

- Le premier chapitre présente d'une façon générale l'état de l'art des algorithmes évolutionnaires. Dans ce chapitre nous commençons par une description détaillée des algorithmes évolutionnaires. Un historique des différents courants de recherche conclura cette partie. Par la suite nous introduirons les différentes étapes de l'algorithme génétique simple et leurs mécanismes de fonctionnement.
- Le deuxième chapitre présente une description d'un algorithme génétique utilisant la transformation et la transposition génétique au lieu de l'opérateur de croisement. La stratégie de la recherche de niche pour les problèmes de grande dimension est également abordée. Pour cela, on étudiera en premier lieu, les différents paramètres qui influenceront sur la transformation et la transposition avec ces différentes formes. On terminera ce chapitre par une étude comparative des différents algorithmes étudiés.
- Le troisième chapitre est consacré à l'application de ces différents algorithmes aux problèmes d'optimisation des systèmes dynamiques, tel que les systèmes biologiques.

I. Introduction

La biologie et l'observation des phénomènes biologiques ont toujours été une source d'inspiration très riche pour les scientifiques. Des approches, telles que les réseaux de neurones, la vie et l'évolution artificielle représentent de bons exemples.

L'idée d'évolution est très ancienne, plusieurs auteurs ont proposé des interprétations évolutionnistes du monde, à base de phénomènes d'adaptation au milieu et de lutte pour la vie. A cet effet des courants différents et variés ont marqué la recherche durant les quarante dernières années, et sont toujours le sujet de débats parfois très passionnés. Historiquement [1], les premières expériences en évolution artificielle proposées par **Friedberg** à la fin des années cinquante. L'idée consistait à faire évoluer des programmes informatiques, écrits en langage machine suivant un processus similaire au processus de l'évolution naturelle. Des parties de ces programmes étaient mutées par des variations aléatoires puis sélectionnées pour constituer de nouveaux programmes. Seulement, contrairement aux programmes de départ les nouveaux programmes ne marchaient pas toujours. A la suite de ces premières expériences plusieurs chercheurs ont commencé à s'intéresser au sujet, et certains d'autre eux ont tenté d'adopter ce principe dans les années 60, qui consiste à s'inspirer des mécanismes de l'évolution naturelle et à utiliser le concept de populations d'individus ou solutions pour résoudre des problèmes du monde réel. Toute fois la faible puissance des machines de l'époque et des connaissances de la génétique naturelle n'a pas permis d'avoir de résultats convaincants. La mise en oeuvre relativement aisée de ces algorithmes ainsi que les nombreux succès qu'elles ont obtenus ont contribué à leur développement spectaculaire ces vingt dernières années. Ce développement porte à la fois sur leur diffusion dans de très nombreux domaines d'application ainsi que sur l'étude et l'exploration des mécanismes d'évolution eux-mêmes. De nombreuses applications de ces méthodes concernent le domaine des systèmes de production (Pierreval, 2003) et notamment le domaine de l'ordonnancement (Portmann, 2001) [1, 2].

L'évolution naturelle a permis de créer des systèmes biologiques très complexes adaptés à de nombreuses conditions. Ses mécanismes reposent sur le principe de compétition entre les individus. Selon la théorie de Charles **Darwin**, les espèces naturelles sont en constante compétition pour survivre et partager des ressources généralement limitées. Les individus les plus adaptés auront plus facilement accès à ces ressources, ils survivront plus longtemps et auront plus de chances de se reproduire et de transmettre leur héritage génétique à leurs descendants [3].

L'évolution naturelle représente donc une méthode de recherche dans un espace vaste pour permettre aux organismes biologiques de survivre et de se reproduire. Simulant cette propriété, l'évolution artificielle a été entre prise pour résoudre des problèmes d'optimisation généralement très difficiles à résoudre par des méthodes d'optimisation classique et pour les quels la recherche de l'optimum se fait dans des espaces de grande dimension. L'ensemble des méthodes développées et proposées forment ce que l'on désigne de nos jours par les algorithmes évolutionnaires [4].

Pour résumer, Lerman et Ngouenet distinguent 4 principaux points qui font la différence fondamentale entre ces algorithmes et les autres méthodes d'optimisation classiques dans la recherche de l'optimum [5]:

1. Les algorithmes évolutionnaires utilisent un codage des paramètres, et non les paramètres eux mêmes.
2. Les algorithmes évolutionnaires travaillent sur une population de points, au lieu d'un point unique.
3. Les algorithmes évolutionnaires n'utilisent que les valeurs de la fonction étudiée, pas sa dérivée, ou une autre connaissance auxiliaire.
4. Les algorithmes évolutionnaires utilisent des règles de transition probabilistes, et non déterministes.

La simplicité de leurs mécanismes, la facilité de leur mise en application et leur efficacité même pour des problèmes complexes ont conduit à un nombre croissants de travaux dans des différents domaines.

II. Les algorithmes évolutionnaires (AE)

Les algorithmes évolutionnaires sont des méthodes stochastiques d'optimisation globale basées sur la théorie Darwinienne de l'évolution des espèces biologiques, ils utilisent à la fois les principes de la survie des individus les mieux adaptés et ceux de la propagation du patrimoine génétique qui s'inspirent des mécanismes de sélection naturelle et des phénomènes génétiques tel que des mécanismes d'évolution de la nature : croisements, mutations, sélections, etc. Pour utiliser ces algorithmes, il faut disposer d'une population d'individus. Chaque individu dispose d'une chaîne chromosomique qui dirige son comportement. Cette chaîne s'apparente à l'ADN dans les organismes vivants. Comme dans les systèmes naturels, des croisements sont réalisés périodiquement et permettent à l'algorithme de créer la génération suivante d'individus, ainsi des mutations sont aussi effectuées. Ces mutations évitent à l'ensemble de la population de converger vers une solution qui ne serait pas optimale. Il existe plusieurs types de ces algorithmes mais l'idée essentielle est la même : simuler l'évolution d'une population dans un espace de recherche à l'aide de trois opérateurs: sélection, croisement, mutation [6, 7].

Malgré la simplicité du processus évolutionnaire, fabriquer un algorithme évolutionnaire efficace est une tâche difficile, car les processus évolutionnaires sont très sensibles aux choix algorithmiques et paramétriques. L'expérience a prouvé que les réussites les plus importantes sont fondées sur une très bonne connaissance du problème à traiter, et une compréhension délicate des mécanismes évolutionnaires.

On distingue quatre grandes familles d'algorithmes évolutionnaires comme indiqué par la figure 1-2 [2,8].

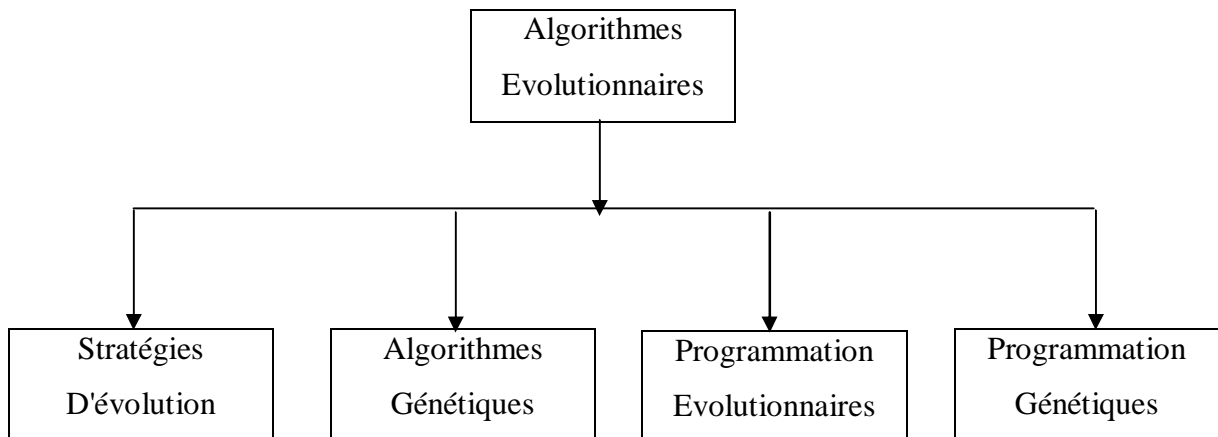


Figure 1-1: Différentes classes d'algorithmes évolutionnaires

- **Les Algorithmes Génétiques : (AG)**, développés à l'Université de Michigan (USA) par (J. Holland 1975, Goldberg 1989, L. Davis 1991 et Michalewicz 1992). Ils ont été imaginés comme outils de modélisation de l'adaptation. C'est les plus connus des algorithmes évolutionnaires; ils favorisent l'utilisation du croisement comme principal opérateur de recherche. Il utilise cependant la mutation avec un faible pourcentage de probabilité, et une méthode de sélection de type probabiliste dans laquelle la probabilité de sélection est proportionnelle au fonction d'adaptation de l'individu. La représentation des individus génotype, qui est à l'origine de type binaire, a été par la suite développée à de nombreuses autres formes de représentation [2, 8].
- **Les Stratégies d'Evolution : (SE)**, développés par (Rechenberg et H.P. Schwefel, 1965, Berlin). Elles ont été développées pour résoudre des problèmes d'optimisation à variables réelles posés au milieu industriels et pour les quels n'existe pas de fonction objectif analytique; le contexte étant l'optimisation paramétrique. Ce sont les meilleurs algorithmes pour les problèmes purement numériques. Ce modèle de stratégies d'évolution utilise le principe de mutation sur les réels du modèle de la programmation évolutive avec un taux de mutation

plus grand. Cette augmentation peut être interprétée par le fait que si la proportion de mutation réussie est élevée, l'espace de recherche exploré est limité autour d'un optimum local; il faut donc diversifier la population en augmentant le taux de mutation. Ces approches utilisent un opérateur de sélection de type déterministe, les solutions dont la fonction d'adaptation est mauvaise sont éliminées de la population. En outre, dans le modèle originel, les populations des parents et de leurs descendants sont généralement de taille différente [2, 8].

- **Programmation Evolutionnaire : (PE)**, développés par (L.J. Fogel, 1964 et D.B. Fogel, 1991, 1995, Californie, USA). Ce modèle évolutionnaire accentue l'utilisation de la mutation et n'utilise pas dans sa version originale la recombinaison des individus par croisement. Développé à l'origine pour l'évolution des automates à état fini, ce modèle est souvent appliqué à la résolution de problèmes d'optimisation à variables réelles dans de espaces de recherche très variés. L'idée consiste à faire subir des mutations importantes aux mauvais individus et des mutations faibles aux bons individus. L'opérateur de sélection est de type probabiliste. Il est à noter que la représentation des individus n'a pas une forme spécifique de génome telle que dans une représentation linéaire de type chaîne binaire par exemple [2,8].
- **Programmation Génétique : (PG)**, développés par (J. Koza, 1990, Californie, USA). Apparue initialement comme une extension du modèle d'apprentissage des algorithmes génétiques, ils sont devenus une branche à part entière (conférence, journal, ...). La (PG) permet de générer des fonctions informatiques à partir des principes évolutionnaires, la population est un ensemble de codes de base de programmes informatiques. La spécificité des (PG) est l'espace de recherche, Les individus formant une population sont donc des programmes candidats à la résolution d'un problème. Ces programmes sont exprimés sous la forme d'arbres sur lesquels les opérateurs génétiques produisent des transformations en vue d'obtenir un programme qui satisfaisant la résolution du problème choisi. Les (PG) cherchent à atteindre un des vieux rêves des programmeurs, faire écrire le programme par un autre programme [2, 8].

Ces différentes classes d'algorithmes évolutionnaires ne diffèrent que sur les détails d'implantation des opérateurs et sur les procédures de sélection et remplacement de la population. Malgré que leur but soit différent à l'origine, ils sont maintenant surtout utilisés pour résoudre des problèmes d'optimisations.

Les méthodes les plus répandues sont les algorithmes génétiques (**AG**). Globalement, les différences entre ces méthodes résident dans la stratégie de résolution (**Bäck93, Park94, Fogel94**). Les (**AG**) sont considérés comme des méthodes de résolution "ascendantes", c'est-à-dire que la solution optimale peut être obtenue progressivement en assemblant des parties optimales des solutions partielles, les opérateurs de recombinaison jouent alors un rôle essentiel. Les stratégies évolutionnistes et la programmation évolutionniste sont vues comme des méthodes "descendantes", dans lesquelles l'environnement agit comme une pression pour faire apparaître la solution optimale. Les opérateurs de recombinaison y ont un rôle secondaire (**Fogel94**) [9].

Les premières applications étaient marquées par l'utilisation de chaînes binaires de longueur fixe pour les AG, des vecteurs de réels pour les stratégies évolutionnistes, un automate à état fini pour la programmation évolutionnaire et des codes de programme pour la programmation génétique. Ceci jusqu'au début des années 80 où les recherches étaient principalement théoriques avec peu d'applications réelles (**Goldberg89**). Ensuite, après de nombreuses investigations, chaque nouvelle application apportait une nouvelle perspective à l'amélioration de la théorie [9].

Dans l'état actuel des recherches, les algorithmes évolutionnistes sont encore à l'état de développement sur les fondements théoriques, notamment dans la résolution de problèmes d'optimisation et d'apprentissage.

III. La boucle évolutionnaire

Le cycle principal d'un algorithme évolutionnaire présenté en figure1-3 consiste à sélectionner des individus parents, à partir d'une population de départ, qui en se reproduisant par des opérations de "croisement" et "mutation" engendrent de nouveaux descendants. Une nouvelle population est ainsi constituée par "remplacement" de

l'ensemble ou d'une partie de la population de départ. La répétition de ce cycle de base produit donc une succession de générations de solutions, jusqu'à satisfaction d'un critère donné de fin de cycle. Ces opérations sont répétées en boucle, souvent sous forme de "générations", jusqu'à ce que la population converge. Si cette boucle est correctement calibrée, le processus stochastique associé converge vers la solution désirée (l'optimum global d'une fonction par exemple) [2, 4].

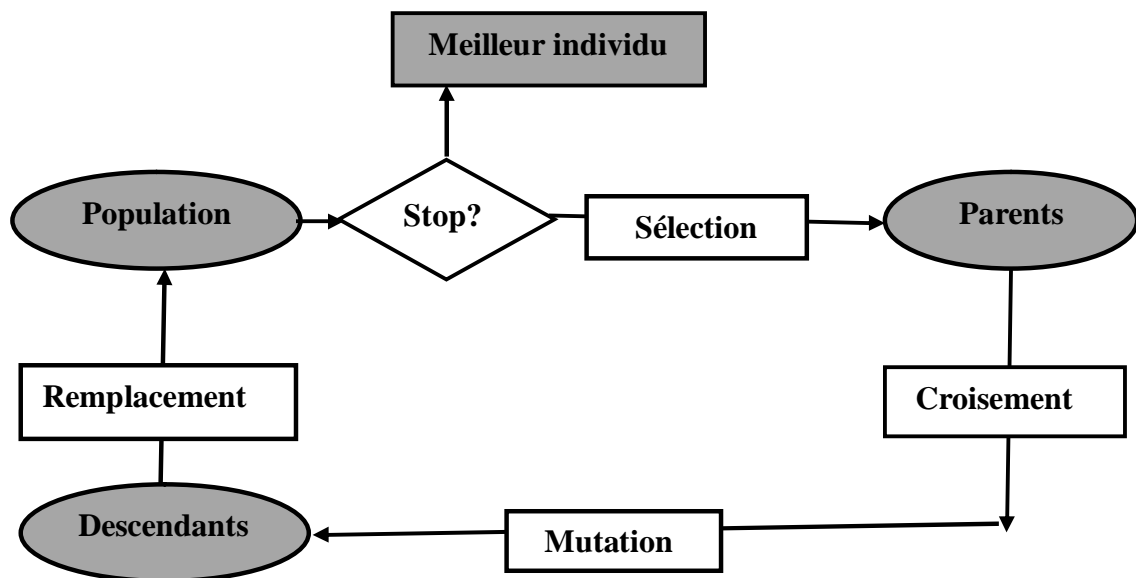


Figure 1-2 : Cycle d'un algorithme évolutionnaire

Les opérateurs intervenant dans l'évolution peuvent être décrits simplement, mais leurs applications sont plus complexes. Car ils dépendent du choix d'un certain nombre de paramètres dont la valeur a une incidence directe sur les performances de l'algorithme évolutionnaire. Le problème essentiel réside justement dans le fait de savoir comment adapter ces paramètres pour que les opérateurs répondent aux spécificités du problème, d'où la nécessité de bien comprendre ces mécanismes d'évolution pour adapter et régler efficacement les différentes composantes de ces algorithmes [5].

VI. Application des algorithmes évolutionnaires

Les algorithmes évolutionnaires sont des algorithmes d'optimisations stochastiques, qui consistent à faire évoluer artificiellement un ensemble de solutions potentielles à un problème donné. Cette évolution qui a pour but la découverte des optima de la fonction à optimiser. Le principe général de ces algorithmes, consiste donc à faire évoluer artificiellement un ensemble de solutions potentielles de manière à favoriser les meilleures solutions; ce qui correspond à la recherche du maximum d'une fonction. Aujourd'hui les algorithmes évolutionnaires sont très utilisés pour la résolution de problème d'optimisation ils sont reconnus comme des outils très efficaces, plus particulièrement lorsque les problèmes à résoudre sont compliqués et que la fonction à optimiser n'est pas régulière ou que ses dérivées soient inaccessibles, mal conditionnées ou complexes à calculer, et que les méthode d'optimisation classiques soient totalement inadaptés [1, 3].

Le succès pratique de ces méthodes vient du fait qu'elles représentent des outils d'optimisation adaptés à des fonctions, des problèmes difficiles, complexes, irréguliers. Le mécanisme évolutionnaire a cependant un coût calculatoire (c'est une méthode itérative, une recherche par tâtonnement, aléatoire dirigé) qui peut devenir important. Ces méthodes sont en fait complémentaires des méthodes d'optimisation plus standard comme les méthodes d'optimisation déterministes qui font le plus souvent des hypothèses de régularité sur les fonctions à optimiser [4].

Malgré l'apparente simplicité d'un processus évolutionnaire, fabriquer un algorithme évolutionnaire efficace est une tâche difficile, car les processus évolutionnaires sont très sensibles aux choix algorithmiques et paramétriques, aux représentations du problème notamment. L'expérience prouve que les grandes réussites sont fondées sur une très bonne connaissance du problème à traiter, sur beaucoup de créativité, et sur une bonne compréhension des mécanismes évolutionnaires [4].

Le champ d'application des algorithmes évolutionnaires est en fait très large, il va des applications réelles complexes comme le contrôle du flux de pipelines de gaz, le

routage aérien ou la planification des trajectoires de robots, à des problèmes plus théoriques et combinatoires, en modélisation économique, en finance, en commande de processus et pour l'apprentissage. On peut cependant dire que l'intérêt et l'efficacité des algorithmes évolutionnaires ont été globalement prouvés d'un point de vue théorique [7].

V. Les algorithmes génétiques (AG)

V.1/ Introduction

Les algorithmes génétiques (**AG**), ont été initialement développés par John Holland (1975) ses collègues et ses étudiants, à l'université du Michigan dans deux buts principaux[10]:

1. Mettre en évidence et expliquer rigoureusement les processus d'adaptation des systèmes naturels.
2. Concevoir des systèmes artificiels qui possèdent les propriétés des systèmes naturels.

Leurs champs d'application sont très vastes. Outre l'économie, ils sont utilisés pour l'optimisation de fonctions numériques difficiles (discontinues, multimodales, bruitées...), traitement d'image (alignement de photos satellites, reconnaissance de suspects...), optimisation d'emplois du temps, optimisation de design, contrôle de systèmes industriels, apprentissage des réseaux de neurones etc. La raison de ce grand nombre d'application est claire c'est la simplicité de leurs mécanismes, la facilité de leur mise en application et leur efficacité même pour des problèmes complexes, les (**AG**) peuvent être utilisés pour contrôler un système évoluant dans le temps (chaîne de production, centrale nucléaire...) car la population peut s'adapter à des conditions changeantes [10].

De plus, les (**AG**) utilisent deux stratégies importantes pour trouver une solution ou un ensemble de solutions. Ces stratégies sont : l'*exploration* et l'*exploitation*. Elles permettent de trouver le maximum global (solution du problème) du fait qu'elles sont complémentaires. Si l'exploration investigate l'ensemble des solutions de l'espace de

recherche, la phase d'exploitation quant à elle se sert de la connaissance des solutions pour aider à trouver de meilleures solutions. La combinaison de ces deux stratégies peut être tout à fait efficace [8].

V.2/ Les outils évolutionnaires de base d'un (AG)

Les (AG), sont basés sur trois éléments principaux : la *sélection*, le *croisement* et la *mutation*. Dans la littérature il s'agit d'*opérateurs* de reproduction. Leur principe est simple, comporte trois phases :

- 1- la genèse (l'initialisation aléatoire d'individus pour former la population de la première génération).
- 2- la reproduction (l'évolution des individus de la génération courante vers la suivante) :
 - la sélection des individus reproducteurs.
 - le croisement génétique de ces individus pour la création de nouveaux individus.
 - la mutation de certains individus pour que la création génétique ne s'affaiblisse pas.
 - l'évaluation des individus par le calcul de leur fonction d'adaptation.
- 3- Recherche de l'individu le plus adapté selon les critères souhaités. La solution sera représentée par le meilleur individu de la dernière génération.

Pour résumer, L'AG est fondé sur [12]:

- Une représentation chromosomique des solutions du problème.
- Une méthode pour générer une population initiale de solutions.
- Une fonction d'évaluation (fitness) pour classer les solutions en fonction de leurs dispositions.
- Des opérateurs génétiques qui définissent la manière dont les caractéristiques génétiques des parents sont transmis aux enfants.
- Les valeurs des paramètres utilisés par l'AG.

V.3/ Optimisation par les algorithmes génétique

Les (AG), utilisent un vocabulaire similaire à celui de la génétique, cependant, les processus auxquels ils font référence sont beaucoup plus complexes. En imitant ce principe, les algorithmes génétiques appliqués à un problème d'optimisation font évoluer un ensemble de solutions utilisent un mécanisme de sélection naturelle. Ainsi, les AG ne se basent pas sur un *individu*, mais sur une *population* d'individus qui vont évoluer de génération en génération pour obtenir un résultat se rapprochant de la solution optimale.

Pour un problème d'optimisation donné, un individu représente un point de l'espace d'état ou une solution possible du problème donné il est composé d'un ou plusieurs chromosomes. Les chromosomes sont eux-mêmes constitués de gènes qui contiennent les caractères héréditaires de l'individu. A chaque individu est attribué un "*fitness*" qui mesure la qualité de la solution qu'il représente, souvent c'est la valeur de la fonction à optimiser. Ensuite, une nouvelle population des solutions possibles est produite en sélectionnant les parents parmi les meilleurs de la "*génération*" actuelle pour effectuer des *croisements* et des *mutations* [12].

La sélection a pour but de favoriser les meilleurs éléments de la population, tandis que le croisement et la mutation assurent une exploration efficace de l'espace d'état. Les meilleurs individus d'une génération vont créer une nouvelle génération plus adaptée au problème dont la nouvelle population contient une plus grande proportion de caractéristiques des meilleurs individus de la génération précédente.

L'organigramme fonctionnel présenté dans la figure 1-3, illustre la structure générale de l'algorithme génétique. Nous détaillerons dans la suite les diverses phases qui le constituent et les mécanismes associés à chacune d'entre elles.

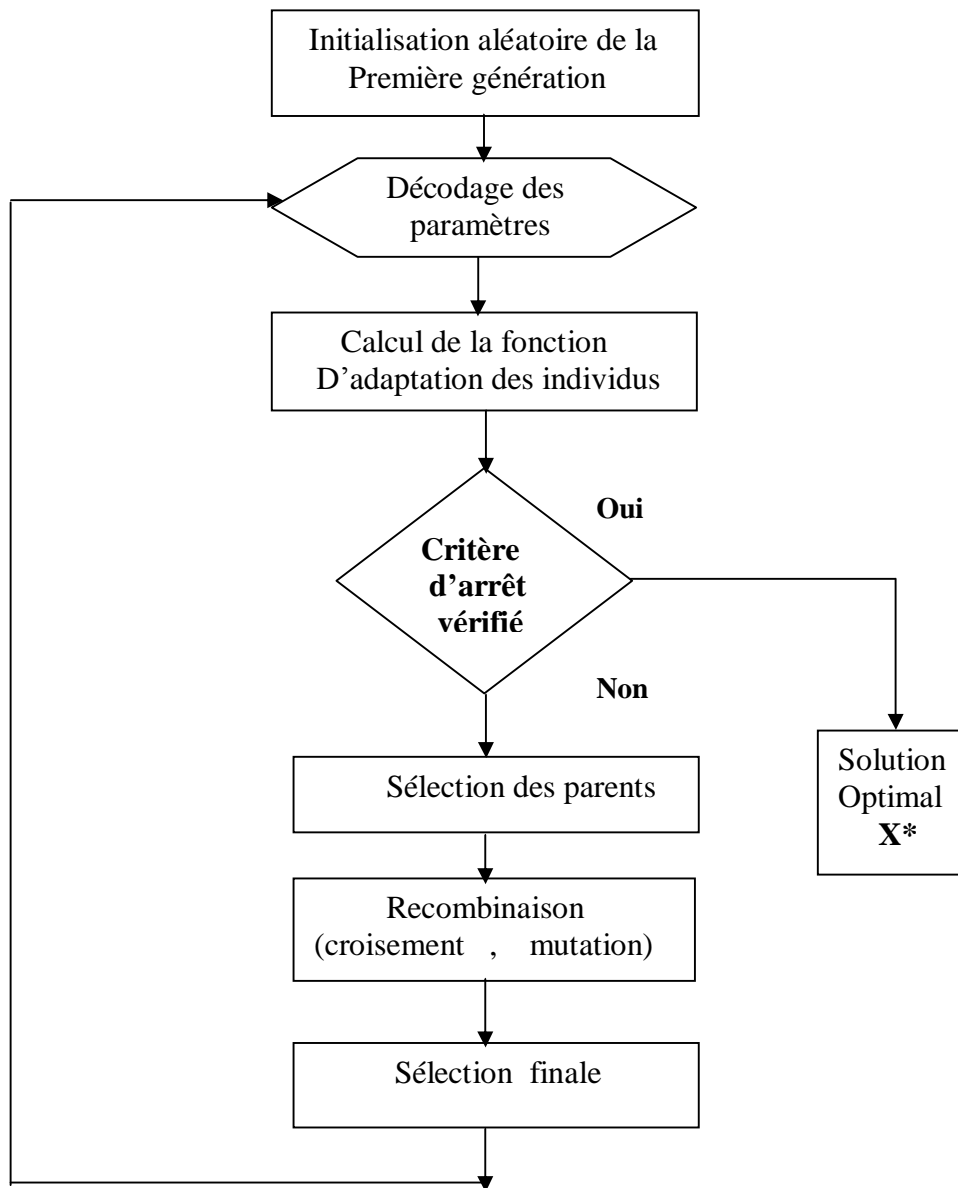


Figure 1-3 : Organigramme général d'un algorithme génétique

V.4/ Mécanismes de fonctionnement d'un (AG)

Les différentes étapes de fonctionnement des (AG) se résument à qui suit :

1. **Initialisation** : Une population initiale de N individus est générée aléatoirement.
2. **Évaluation** : Chaque individu est décodé, puis évalué.
3. **Sélection**: Création d'une nouvelle population par l'utilisation d'une méthode de sélection appropriée.
4. **Recombinaison**: croisement et mutation au sein de la nouvelle population.
5. **Retour** à la phase d'évaluation jusqu'à la vérification du critère d'arrêt de l'algorithme.

La mise en œuvre des algorithmes génétiques nécessite donc plusieurs étapes. L'idée fondamentale est que: la population choisie contient potentiellement la solution, ou plutôt la meilleure solution, à un problème donné. Cette solution n'est pas exprimée car la combinaison génétique sur laquelle elle repose est dispersée chez plusieurs individus. Ce n'est que par l'association de ces combinaisons génétiques au cours de la reproduction que la solution pourra s'exprimer. Lors de la reproduction et de la recombinaison génétique associée, un individu hérite, par hasard, d'un des gènes de chacun de ses parents. L'originalité des mécanismes repose en particulier sur le fait qu'il n'a pas considéré les seules mutations comme source d'évolution mais aussi et surtout les phénomènes de croisement. C'est en croisant les solutions potentielles existant que l'on peut se rapprocher de l'optimum [11].

1°/ Initialisation de la population

Comme dans tout problème d'optimisation, une connaissance de ``bons" points de départ conditionne la rapidité de la convergence vers l'optimum. Si la position de l'optimum dans l'espace d'état est totalement inconnue, il est naturel de générer aléatoirement des individus en faisant des tirages uniformes dans chacun des domaines associés aux composantes de l'espace d'état, en veillant à ce que les individus produits respectent les contraintes [11].

La génération de la population initiale peut se faire en prenant des individus régulièrement répartis dans l'espace. Néanmoins, une initialisation aléatoire est plus simple à réaliser. Les valeurs $N(x_i)$ des gènes est alors tirées au hasard selon une distribution uniforme.

Le choix de la population initiale peut conditionner fortement la rapidité de l'algorithme. Il doit être capable de produire une population d'individus non homogène qui servira de base pour les générations futures, et capable de rendre plus ou moins rapide la convergence vers l'optimum global.

Dans le cas où l'on ne connaît rien du problème à résoudre, il est essentiel que la population initiale soit assez bien répartie sur tout le domaine de recherche. Une population trop petite évoluera probablement vers un optimum local intéressant alors qu'une population trop grande sera inutile car le temps de convergence sera excessif. La taille de la population doit être choisie de façon à réaliser un bon compromis entre temps de calcul et qualité du résultat

2°/Codage et décodage des variables

Le codage est une partie très importante des algorithmes génétiques. Il permet de représenter l'individu sous la forme d'un chromosome. Ce chromosome est constitué de gènes qui prennent des valeurs dans un alphabet binaire ou non. Certains auteurs n'hésitent pas à faire le parallèle avec la biologie et parlent de **génotype** en ce qui concerne la représentation binaire d'un individu, et de **phénotype** pour ce qui est de sa valeur réelle correspondante dans l'espace de recherche [11].

Le choix du codage est délicat. Il doit permettre de coder toutes les solutions et permettre la mise en œuvre des opérateurs de reproduction. C'est ainsi que le bon déroulement des algorithmes génétiques sera assuré. Plusieurs type de codages sont utilisés, on citera à titre d'exemple: codage réel, codage binaire, Gray.

Codage binaire

Goldberg et Holland ont démontré qu'il est idéal de représenter le chromosome en une chaîne binaire. C'est pourquoi les AG utilisent généralement cette représentation Les individus sont représentés sous forme de chaînes de bits contenant toute l'information nécessaire à la description d'un point dans l'espace. Ce type de codage a pour intérêt de permettre la création d'opérateurs de croisement et de mutation simples [11].

A chaque variable d'optimisation x_i correspond un *gène*. Un *chromosome* sera donc un ensemble de gènes. Chaque point est représenté par un *individu* doté d'un *génotype* constitué d'un ou de plusieurs chromosomes. La *population* est un ensemble de N individus qui vont évoluer d'une génération à une autre. Du point de vue informatique, un gène est un entier long de K bits. Un chromosome est un tableau de gènes. Un individu est un tableau de chromosomes. La population est un tableau d'individus [11].

Dans le codage binaire le gène est codé par un caractère binaire, 0 ou 1. C'est le plus courant et celui qui a été employé lors de la première application des algorithmes génétiques. Un des avantages du codage binaire est que l'on peut ainsi facilement coder toutes sortes d'objets : des réels, des entiers, des valeurs booléennes, des chaînes de caractères... Cela nécessite simplement l'usage de fonctions de codage et décodage pour passer d'une représentation à l'autre [12].

Pour chaque paramètre x_i situé dans l'intervalle $[x_{i \min}, x_{i \max}]$, on associe une chaîne binaire $b_0 b_1 \dots b_{l_{xi}}$ définie sur l_{xi} bits. A cette chaîne correspond une valeur entière naturelle [12]:

$$N(x_i) = \sum_{i=1}^{l_{xi}-1} 2^{l_{xi}-i-1} \cdot b_i$$

Le paramètre réel x_i de l'espace de recherche relatif à $N(x_i)$ est obtenu par interpolation linéaire.

$$x_i = x_{i \min} + \frac{x_{i \max} - x_{i \min}}{2^{l_{xi}} - 1} \cdot N(x_i)$$

La longueur totale du chromosome est donnée par.

$$L = \sum_{i=1}^m l_{xi}$$

m : le nombre des paramètres.

Exemples

Chromosome A:

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
|---|---|---|---|---|---|---|---|

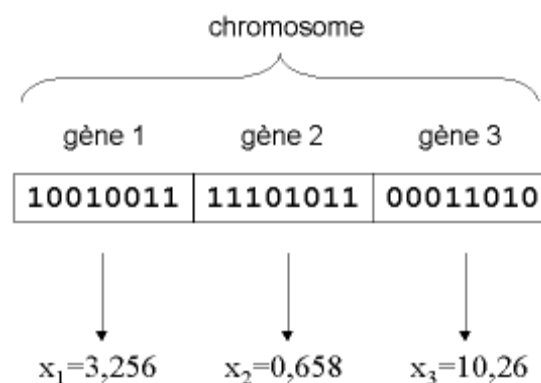
Chromosome B:

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
|---|---|---|---|---|---|---|---|

Codage réel

Davis, Janikow et Michalewicz ont effectué une comparaison entre la représentation binaire et la représentation réelle. Ces auteurs ont trouvé que la représentation réelle donne de meilleurs résultats d'après leur problème à résoudre. Dans ce codage le génome est un vecteur réel et l'espace de recherche est un sous-ensemble de \mathbb{R} . Cette représentation est aujourd'hui très utilisée dans les problèmes d'optimisation car dans de nombreuses applications du monde réel, ces problèmes sont naturellement formulés sous forme paramétrique i.e. Les premiers travaux qui ont utilisé ce type de représentation ont été ceux de Rechenberg et de Schwefel quand ils ont introduit les stratégies d'évolution [1,12].

Le codage réel peut-être utile notamment dans le cas où l'on recherche le maximum d'une fonction réelle.

Exemples

Codage de Gray

Dans le cas d'un codage binaire on utilise souvent la "distance de Hamming" entre les codage binaires de deux nombres réels proches, comme mesure de la dissimilarité entre deux éléments de population, cette mesure compte les différences de bits de même rang de ces deux séquences. Et c'est là que le codage binaire commence à montrer ses limites. En effet, deux éléments voisins en terme de distance de Hamming ne codent pas nécessairement deux éléments proches dans l'espace de recherche. Cet inconvénient peut être évité en utilisant un "codage de Gray" : le codage de Gray est un codage qui a comme propriété que entre un élément n et un élément $n+1$, donc voisin dans l'espace de recherche, un seul bit diffère [13].

3° La fonction d'adaptation

Pour calculer le coût d'un point de l'espace de recherche, on utilise une fonction d'évaluation ou d'adaptation (**F**). L'évaluation d'un individu ne dépendant pas de celle des autres individus, le résultat fournit par la fonction d'évaluation va permettre de sélectionner ou de refuser un individu pour ne garder que les individus ayant le meilleur coût en fonction de la population courante : c'est le rôle de la fonction (**F**). Cette procédure permet de s'assurer que les individus performants seront conservés, alors que les individus peu adaptés seront progressivement éliminés [13].

La fonction d'adaptation, associe une valeur pour chaque individu. Cette valeur à pour but d'évaluer le degré d'adaptation d'un individu à son environnement. Les individus peuvent être aussi comparés entre eux. Cette fonction, propre au problème, est souvent simple à formuler lorsqu'il y a peu de paramètres. Au contraire, lorsqu'il y a beaucoup de paramètres ou lorsqu'ils sont corrélés, elle est plus difficile à définir. Dans ce cas, la fonction devient une somme pondérée de plusieurs fonctions.

La fonction d'adaptation doit exprimer le plus fidèlement possible, la problématique posée sous forme mathématique. Sa définition peut être simplement analytique, ou elle peut éventuellement faire appel au jugement de l'utilisateur. En raison

de l'analogie avec la théorie de l'évolution (survie des individus les mieux adaptés à leur environnement), les algorithmes génétiques sont naturellement formulés en terme de maximisation. Ils servent donc à déterminer le maximum d'une fonction F Réelle à une ou plusieurs variables. Le problème d'optimisation sur l'espace de recherche E est formulé comme suit [13]:

$$\underset{x \in E}{\text{Max}} F(x)$$

Si le problème à résoudre est un problème de minimisation d'une fonction J .

$$\underset{x \in E}{\text{Min}} J(x)$$

Ceci équivaut au problème de maximisation de $F(x)$, que l'on définit comme suit:

$$F(x) = \frac{1}{1+J(x)}$$

Le choix de F n'est pas unique, mais cette transformation est la plus utilisée dans la littérature.

4°/ La sélection des parents

La sélection des parents a pour but de deviner les individus de la population courante qui seront autorisés à se reproduire (les "**parents**"). La sélection est fondée sur la qualité des individus, estimée à l'aide de fonction d'adaptation. Cette opération est peut-être la plus importante puisqu'elle permet aux individus d'une population de survivre, de se reproduire ou de mourir. En règle générale, la probabilité de survie d'un individu sera directement liée à son efficacité relative au sein de la population. Il existe plusieurs méthodes pour la reproduction. , on citera à titre d'exemple:

- La sélection par roulette ou proportionnelle
- La sélection par tournoi.

- La sélection à reste stochastique.
- La sélection stochastique à reste stochastique.
- La sélection par le rang.
- La sélection uniforme.

Parmi ces différents types de sélection la méthode la plus connue et la plus utilisée reste la roulette biaisée, proposée par Goldberg (1989).

La sélection par roulette

La phase de sélection spécifie les individus de la population qui doivent survivre. La méthode de base, appelée roue de loterie attribue à chaque individu une probabilité de survie proportionnelle à son adaptation dans la population. Lors de la phase de sélection, les individus sont sélectionnés aléatoirement en respectant les probabilités p_i associées pour former la population de la nouvelle génération.

Cette méthode consiste à dupliquer chaque individu de la population proportionnellement à son milieu. Ainsi, les individus ayant la plus grande valeur de fitness auront plus de chance d'être choisis. Dans une population de N individus, la fonction de sélection est la suivante [12]:

$$P_s(x_i) = \frac{F(x_i)}{\sum_{j=1}^N F(x_j)}$$

En utilisant cette probabilité de reproduction, on peut créer une roue de loterie biaisée. Chaque individu de la population occupe une section de la roue proportionnellement à son adaptation et qui indique aléatoirement quel individu peut se reproduire. Cette méthode n'assure pas la sélection des meilleurs individus et peut être une cause de la convergence prématurée

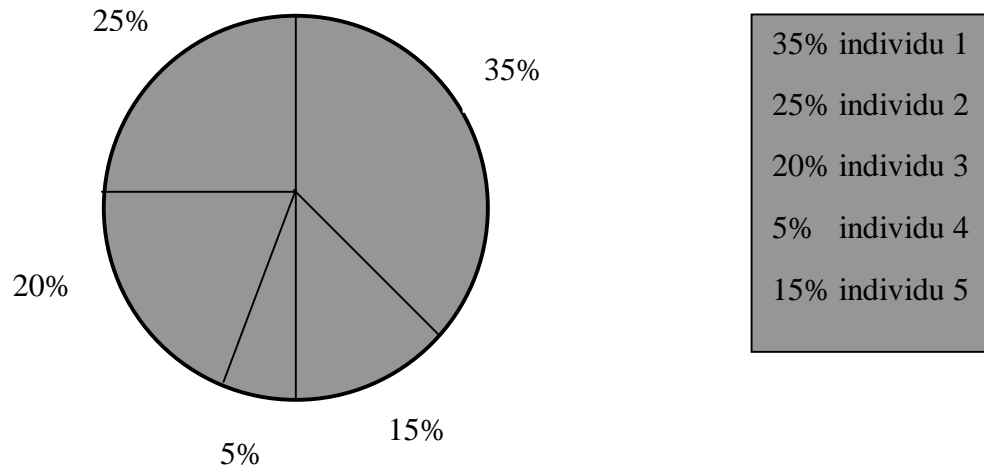


Figure 1-4 : Exemple de sélection par roulette

La Sélection par tournoi

C'est la méthode la plus facile à mettre en œuvre. Cette technique utilise la méthode de la roulette biaisée pour sélectionner deux individus. On récupère celui dont la valeur de la fonction d'adaptation est la plus grande. Cette méthode choisit toujours une valeur de la fonction d'adaptation plus élevée par rapport à la technique de la roulette biaisée [12].

La sélection uniforme

C'est une technique très simple qui consiste à sélectionner un individu aléatoirement de la population P . La probabilité P_i pour qu'un individu soit sélectionné est définie par [12] :

$$P_i = \frac{1}{\text{taille pop}}$$

La sélection par le rang

Cette méthode est très semblable au tirage à la roulette, sauf que les cases de la roulette ne sont plus proportionnelles à la fitness des individus, mais à leur rang dans la population. Le meilleur individu a le rang le plus élevé, le dernier a un rang de 1 [12].

La sélection stochastique

Contrairement aux méthodes déterministes, les méthodes stochastiques associent aux individus une probabilité de sélection, généralement fonction croissante de leur fonction d'adaptation [12].

5°/ La recombinaison génétique

Pour créer un nouvel individu à partir des meilleures solutions précédemment sélectionnées, il est nécessaire de procéder à la combinaison des gènes des parents pris de manière aléatoire et d'après la théorie de l'évolution, pour que la génération suivante soit plus adaptée au problème et plus performante on doit combiner les meilleurs individus de la population actuelle. Une étape d'identification et de sélection de ces meilleurs individus est donc nécessaire pour que chaque individu ait une chance proportionnelle à son adaptation de devenir parent [11].

On distingue deux opérateurs principaux : Le croisement et La mutation qui permette d'explorer l'ensemble des solutions possibles. Ces opérations sont appliquées aléatoirement, à l'aide de deux paramètres, la probabilité de croisement et la probabilité de mutation. Ces probabilités sont des paramètres très importants, qui influent de façon considérable sur la convergence.

a)-Le Croisement

Le phénomène de croisement est une propriété naturelle de l'ADN. C'est par analogie qu'ont été conçus les opérateurs de croisement dans les (AG). Le croisement combine les gènes des deux individus parents pour donner deux nouveaux chromosomes d'individus enfants (descendants) possédant des caractéristiques issues des deux parents. La zone de croisement est généralement choisie aléatoirement dans les chromosomes. Les méthodes de croisement sont liées au codage mais leur principe est identique. Il a pour but d'enrichir la diversité de la population en manipulant la structure des chromosomes, il

favorise l'exploration de l'espace de recherche et permet d'explorer l'ensemble des solutions possibles. Classiquement, les croisements sont envisagés avec deux parents et génèrent deux enfants. Dans un groupe de parents arbitrairement choisis dans la population chaque paire dans la population formée va subir le croisement avec une probabilité P_{cross} [11].

De nombreux types de croisement existent dans la littérature. Ils préservent plus ou moins l'identité génétique des parents et permettent un déplacement dans tout l'espace des solutions le type de croisement le plus simple est le croisement à un site

Croisement à un site

Il consiste à échanger les gènes de chacun des parents de longueur l en vérifiant la probabilité P_c . Le site de croisement S doit être choisi entre 1 et $(l - 1)$. Le changement va se faire entre le site sélectionné et la position finale l des deux chaînes comme le montre la figure 1-5.

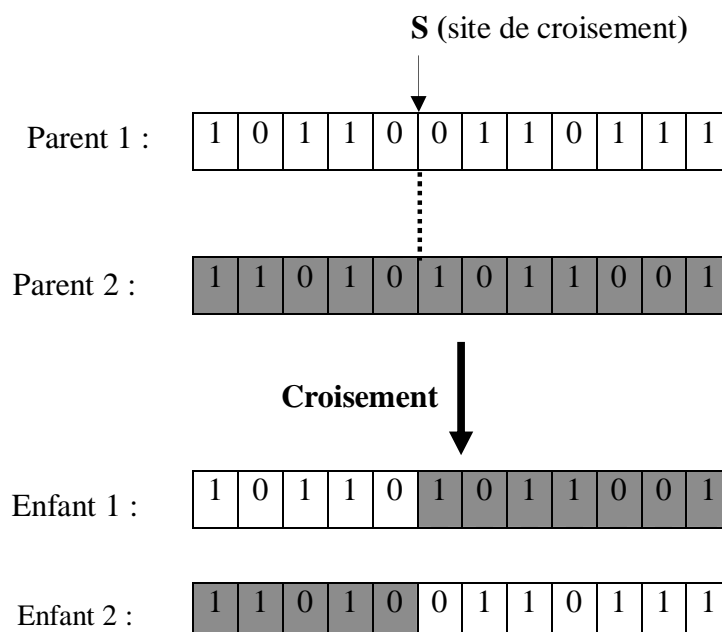


Figure 1-5 : Croisement à un site

Croisement à k sites

On choisit au hasard k points de croisements successifs. Cet opérateur généralement considéré comme plus efficace que le précédent. Le changement va se faire entre deux sites successifs des deux chaînes comme le montre la figure 1-6.

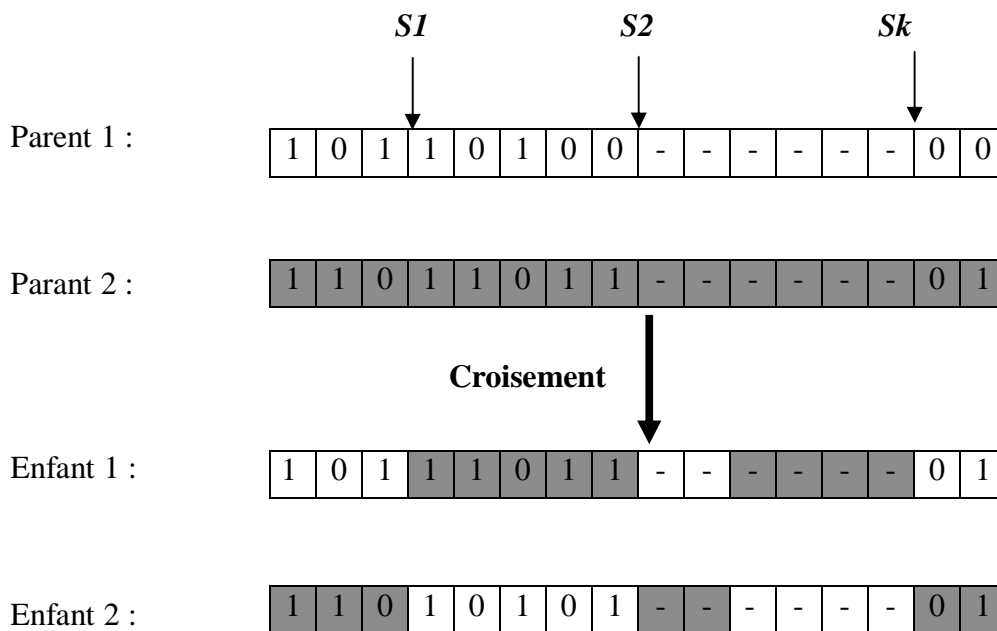


Figure 1-6 : Croisement à k sites

b)-La mutation

La mutation prend une place de plus en plus importante dans les algorithmes génétiques, alors qu'il y a encore quelques années son rôle était encore considéré comme accessoire. Comme les individus les mieux adaptés sont les plus susceptibles d'être choisis lors de la sélection, la perte de certains gènes est inévitable avec le temps. La mutation est l'opérateur qui permet d'éviter la dégénérescence de la population. Cette dégénérescence peut se traduire par une convergence des individus vers un optimum local, d'où l'importance de la mutation. Ce phénomène génétique d'apparition de

"mutants" est rare mais permet d'expliquer les changements dans la morphologie des espèces, toujours dans le sens d'une meilleure adaptation au milieu naturel.

Classiquement, la mutation modifie aléatoirement, un petit nombre de gènes, avec un faible taux de probabilité, ceci revient à modifier aléatoirement la valeur d'un paramètre du dispositif. Les individus de la population issus du croisement vont ensuite subir un processus de mutation avec une probabilité P_{mut} qui est exécuté bit à bit. Comme pour le croisement, la mutation dépend du problème posé, la principale différence se situe dans le taux de mutation qui est généralement faible et se situe entre 0.5% et 1% de la population totale. Ce taux faible permet d'éviter une dispersion aléatoire de la population et n'entraîne que quelques modifications sur un nombre limité d'individus[13].

La mutation a pour rôle de maintenir une certaine diversité dans la population et protège les individus contre une perte des informations essentielle contenues dans leurs gènes. Elles permettent d'assurer une recherche aussi bien globale que locale et garantit la convergence vers l'optimum.

Comme pour les croisements, de nombreuses méthodes de mutation ont été développées dans la littérature mais l'une des plus efficaces est celle qui consiste à muter chaque paramètre de la fonction à optimiser avec une probabilité dépendant des informations contenues dans les gènes des individus. Dans le cas du codage binaire, chaque bit est remplacé selon une probabilité P_{mut} par son inverse. C'est ce qu'illustre la figure 1-7.

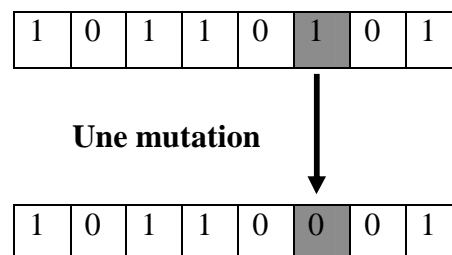


Figure 1-7 : mutation dans un chromosome.

6°/La sélection finale

Cette étape consiste à garder seulement les solutions les plus intéressantes, tout en maintenant une population assez grande et assez diversifiée. C'est pourquoi la taille de la population doit rester la même d'une génération à l'autre.

La sélection revient à choisir les meilleurs individus pour former la nouvelle génération, c'est à dire éliminer N individu parmi les $2N$ individus (N parents et N enfants) pour cela plusieurs méthodes sont proposées [13].

a)-La sélection par descendance

Dans cette méthode, on garde toujours les enfants, quelque soit leur adaptation la population de la nouvelle génération est obtenue par descendance ; les enfants remplaçant automatiquement leurs parents.

L'inconvénient de cette sélection est que l'on risque de voir disparaître les caractéristiques génétiques des parents les mieux adaptés si elles n'ont pas été totalement transmises lors de la recombinaison génétique.

b)- La sélection par compétition

Une compétition a lieu entre parents et enfants pour déterminer ceux qui feront partie de la génération suivante. Les enfants sont insérés dans la population si et seulement si leurs performances sont supérieures à celles de leurs parents, à rang équivalent.

c)- La sélection de procréation sélective

On garde les N meilleurs individus parmi la population intermédiaire de parents et d'enfants.

IV Conclusion

Dans ce chapitre, nous avons donné des généralités sur les algorithmes évolutionnaires. En premier lieu, nous avons introduit un rappel sur l'évolution des algorithmes évolutionnaires. Puis nous avons fait un rappel sur les différents types de ces algorithmes et leurs domaines d'application. Ainsi que ce chapitre nous a permis d'avoir une vue générale sur les concepts des AG, leurs applications et leurs mécanismes de fonctionnement. Une de ces applications est l'optimisation. Nous pouvons conclure que les AG sont des algorithmes simples de conception et peuvent résoudre des problèmes assez complexes. La résolution de ces problèmes est obtenue grâce aux opérateurs de reproduction.

L'objectif de ce chapitre était de donner un aperçu rapide des différentes notions utilisées dans les méthodes d'optimisation évolutionnaires fondées sur le principe de la sélection naturelle afin de faciliter la lecture de la suite de ce mémoire ou nous allons présenter une amélioration à ces méthodes et plus précisément sur les algorithmes génétiques ou nous allons introduire des nouveaux mécanismes au lieu du croisement appelé **Transformation** et **Transposition**, et une nouvelle stratégie appelée la stratégie des niches, permettant les utilisateurs à mieux les maîtriser, pour cela, nous avons introduit à la fin de ce chapitre les mécanismes de fonctionnement des algorithmes proposés et nous avons expliqué brièvement les procédures de fonctionnement où elles vont être mieux expliquées en introduisant les différents résultats trouvés.

I. Introduction

Comme présenté, dans le chapitre précédent, les algorithmes génétiques représentent une alternative intéressante aux techniques classiques d'optimisation. Leur robustesse et leur fiabilité ont été largement démontrées au travers de nombreux travaux et articles publiés. Toutefois l'efficacité de ces algorithmes dépend fortement du choix des opérateurs génétiques intervenant lors du processus de diversification de la population au cours des générations et dans l'exploration de l'espace des solutions.

L'un des aspects fondamentaux d'un algorithme génétique réside dans la diversité de la population. Une perte de cette diversité peut impliquer une convergence prématurée de l'algorithme vers un optimum local. En effet, les solutions convergent vers une région précise de l'espace de recherche et l'algorithme n'explore plus les autres zones. Il est également vrai qu'une population homogène est moins adaptée aux changements de l'environnement. Dans les algorithmes génétiques classiques c'est l'opérateur de croisement qui assure la diversité de la population. Pour cette raison, il doit être basé sur une méthode de sélection assez forte [10].

Les performances des algorithmes se trouvent souvent pénalisées par le caractère très aléatoire des opérateurs de croisement et de mutation. Pour remédier à ce phénomène, certains évolutionnistes proposent de nouvelles techniques qui permettent de favoriser les croisements ou les mutations [12].

Dans la réalité la diversité des espèces génétiques est obtenue grâce à la participation de différents mécanismes et stratégies faisant intervenir des processus tel que l'insertion, la duplication ou le mouvement des gènes. D'ailleurs récemment des techniques inspirées de processus biologique autres que le croisement ont été proposées dans la littérature [12].

Dans le cadre de ce mémoire nous nous sommes intéressés en particulier à celles appelées "transformation" et "transposition". Ces dernières représentent une alternative à l'opérateur de croisement classique. Dans ce chapitre, nous décrirons également la stratégie de la recherche de niche. Cette méthode est basée sur l'approche génétique, et elle est caractérisée par une recherche révolutionnaire à deux niveaux ; au niveau des niches et au niveau des individus.

II. Les mécanismes biologiques pouvant assurer la diversité des individus

Dans la nature, la diversité génétique des individus est assurée grâce à plusieurs mécanismes. Nous citerons à titre d'exemples des opérations telles que l'insertion, la duplication, la reproduction ou le mouvement des gènes.

Chacune de ces opérations fait intervenir un ou plusieurs processus qui d'une façon ou d'une autre produisent des changements dans le génome de l'espèce et permet une diversité génétique. Plusieurs mécanismes biologiques peuvent reproduire ces phénomènes; on s'intéressera en particulier à: la transformation, la transposition, la transduction, etc.

La transformation: Dans une transformation de nouveaux gènes prélevés sur des cellules mortes, sont incorporés dans le chromosome de l'individu au travers de son environnement immédiat.

La transduction: Des nouveaux gènes prélevés accidentellement sur un porteur, sont transmis au nouvel individu par l'intermédiaire d'un verus.

La transposition: Elle se caractérise par la présence d'unités génétiques mobiles qui se déplacent soit avec le génome soit elles se dupliquent elles même avant de se déplacer pour s'insérer ailleurs.

La mise en œuvre de chacun de ces mécanismes dans un algorithme nécessite la mise en place d'un ensemble de procédures. Dans ce mémoire, nous nous sommes intéressés à la transformation et à la transposition.

III. La transformation

III.1/Principe

Le principe de la transformation est tiré d'un phénomène naturel lié au comportement de certaines bactéries. En effet, il existe dans la nature, des bactéries qui après avoir absorbé des fragments d'ADN à partir de leur environnement immédiat sont capables de le réintégrer dans leur matériel génétique propre. La réintégration de ces fragments d'ADN peut leur conférer certains avantages. D'après la littérature ces fragments d'ADN ou gène segment ont plus d'effet si elles sont prises à partir de bactéries mortes[14].

Simoès et **Costa** ont proposé d'utiliser ce mécanisme, en tant qu'opérateur génétique dans les algorithmes génétiques, ce qui a pour effet de favoriser la diversité et éviter la convergence prématurée. L'efficacité de la transformation dépend de deux paramètres : "le choix de la longueur du segment de gènes et le pourcentage d'individus choisis pour être transformés" [14].

Dans un algorithme génétique qui utilise la transformation, on dispose d'une population et d'une corbeille des segments de gènes créer aléatoirement. Des individus pris aléatoirement parmi la population, subiront la transformation.

Cette transformation se produit comme suit :

Un point de transformation est choisi sur le chromosome de chaque individu, ensuite un segment de gènes de l'individu est remplacé par un segment de gènes pris dans la corbeille. Après chaque transformation, la corbeille des segments de gènes est mise à jour en réinjectant les gènes pris sur les individus qui ont été transformés et des nouveaux gènes créés aléatoirement. Le taux des individus qui doivent subir la transformation ainsi que la longueur du segment de gènes sont deux paramètres importants dont il faut prendre en compte dans la mise en œuvre de l'algorithme. La population devant subir la transformation est sélectionnée par une des méthodes de sélection énumérées dans le chapitre précédent.

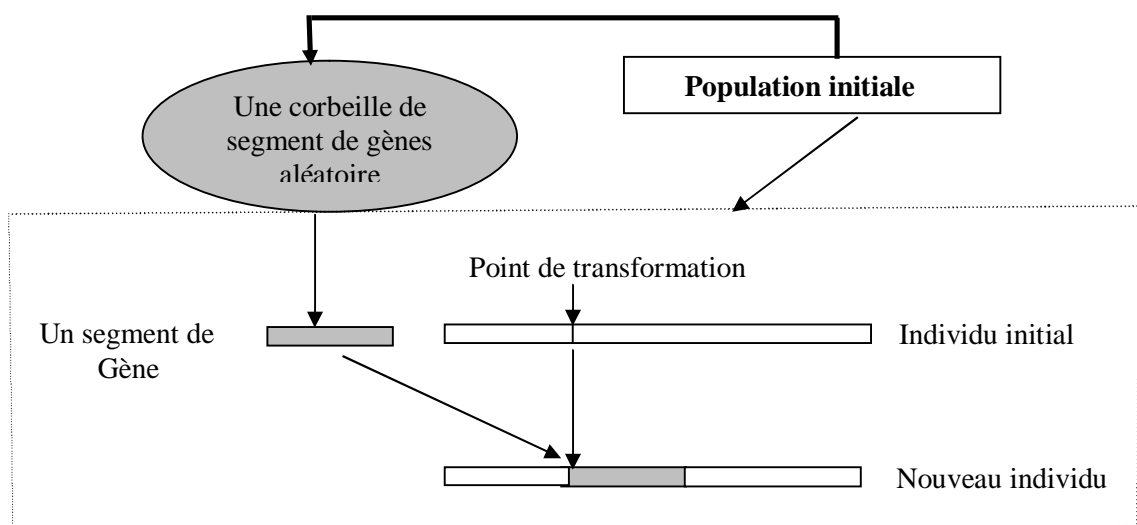


Figure 2-1: Mécanisme de la transformation

III.2/Algorithme

1. $P(t)$ =génération de la première population
2. $S(t)$ =génération initial d'un groupe de segments de gènes
3. Tant que le critère d'arrêt n'est pas vérifié faire:
 - a. Evaluer $P(t)$.
 - b. Sélectionner les individus a partir de $P(t)$, les individus devant subir la transformation.
 - c. Appliquer la transformation aux individus sélectionnés, on obtient la population $P'(t)$.
 - d. Mutation des nouveaux individus $\longrightarrow P''(t)$ la nouvelle population.
 - e. Mise à jour de $S(t)$ en utilisant des segments récupérés sur les individus de $P(t)$ et des nouveaux segments créés aléatoires.
 - f. Remplacer $P(t)$ par $P''(t)$.

III.3/Organigramme

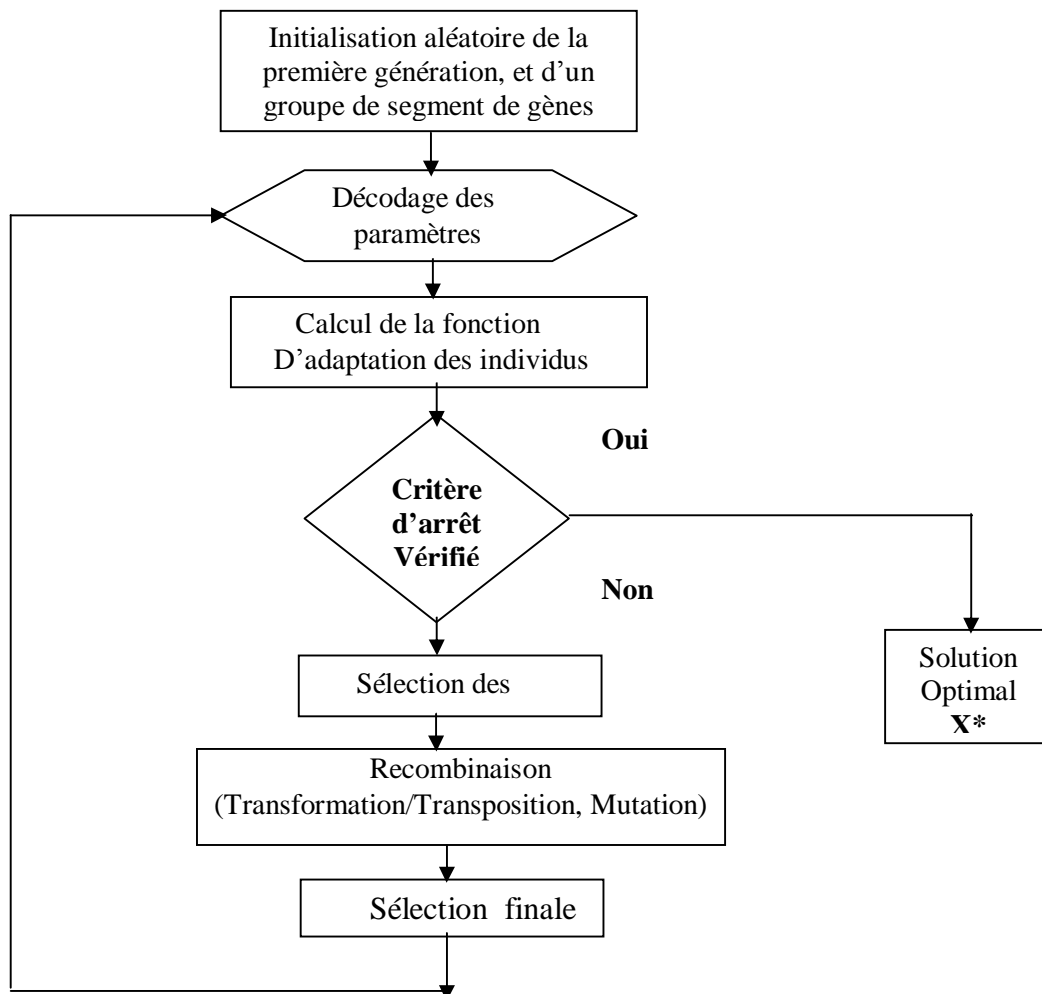


Figure 2-2 : Organigramme général de l'algorithme génétique modifié

III.4/Influence des paramètres intervenant dans la transformation des gènes sur les performances de l'algorithme

Le choix de certains paramètres intervenant dans la transformation des gènes est un paramètre très important pour la convergence de l'algorithme. On s'est surtout intéressés à l'influence de la longueur du gène segment et à la proportion de la population transformée. Pour cela, une étude préalable a été nécessaire. Les tests ont été faits sur une seule fonction, en l'occurrence la fonction *Rosenbrok* à 5 et à 10 variables.

Notre intérêt s'est limité à l'étude d'une seule fonction, car notre but était juste de tester le fonctionnement d'un tel algorithme.

a)-Influence de la longueur du segment de gènes

Pour les tests, le taux de transformation a été fixé à 70%. La longueur du chromosome a été fixée à 60 pour le cas de la fonction à 5 variables et à 100 pour la fonction à 10 variables. Les longueurs des segments de gènes (**PG**) utilisés pour le premier cas sont : 5, 15, 25 et 50, et pour le deuxième cas : 10, 20, 40 et 70. Les résultats obtenus dans chaque cas, sont indiqués successivement dans les tableaux 2-1, 2-2 et les figures 2-3, 2-4.

Tableau 2-1

| PG | N_géné | F1_min | X1min | X2min | X3min | X4min | X5min |
|----|--------|------------|---------|---------|---------|---------|---------|
| 5 | 22 | 1.0742e-05 | 1.0015 | 1.0029 | 1.0059 | 1.0026 | 1.0039 |
| 15 | 62 | 0.0056786 | 0.99465 | 0.98974 | 0.97597 | 0.95297 | 0.9044 |
| 35 | 100 | 0.022141 | 1.001 | 0.99604 | 0.98806 | 0.98176 | 0.97509 |
| 50 | 100 | 0.050247 | 0.99919 | 1.0179 | 1.0347 | 1.0736 | 1.1591 |

N_géné : c'est le nombre de générations où l'algorithme atteint la valeur minimale.

Tableau 2-2

| PG | N_géné | F1_min | X1min | X2min | X3min | X4min | X5min |
|----|--------|----------|---------|---------|---------|---------|--------|
| 5 | 100 | 0.093485 | 0.99853 | 1.0053 | 1.0064 | 1.001 | 1.0056 |
| 10 | 98 | 0.05334 | 1.0067 | 0.99333 | 0.9685 | 0.98418 | 1.0136 |
| 35 | 100 | 0.18477 | 0.98799 | 0.99905 | 0.99282 | 1.0001 | 1.0139 |
| 70 | 100 | 1.2913 | 1.0107 | 1.0385 | 1.0373 | 1.0347 | 1.03 |
| | | | X6min | X7min | X8min | X9min | X10min |
| | | | 1.0367 | 1.0616 | 1.0641 | 1.0799 | 1.1538 |
| | | | 1.0024 | 1.024 | 1.0508 | 1.0935 | 1.1925 |
| | | | 1.0088 | 1.0053 | 1.0115 | 1.0069 | 1.0035 |
| | | | 1.0053 | 1.0319 | 1.0063 | 1.0189 | 1.016 |

Dans le premier cas ($i=5$), les résultats obtenus montrent que pour des segments de gènes de longueur 35 et 50 bits, l'optimum obtenu n'est pas satisfaisant. La valeur de l'optimum obtenu pour un segment de gènes de longueur 15 bits est meilleure mais pour un nombre de génération plus élevé. Le meilleur résultat est obtenu pour PG égale 5 avec un nombre de génération très réduit.

Dans le deuxième cas ($i=10$), les résultats montrent que pour des segments de gènes de longueur 5, 35 et 70 bits l'optimum obtenu n'est pas satisfaisant. La meilleure valeur est obtenue pour PG égale 10.

Il en ressort donc que les résultats dépendent du rapport entre la longueur du segment de gènes et celle du chromosome. On remarque, aussi que l'augmentation de la longueur du segment de gènes dans le premier cas diminue la convergence de l'algorithme, alors que dans le deuxième cas avec un segment de gène de 5 bits on n'a pas atteint l'optimum. Un choix approprié est donc nécessaire.

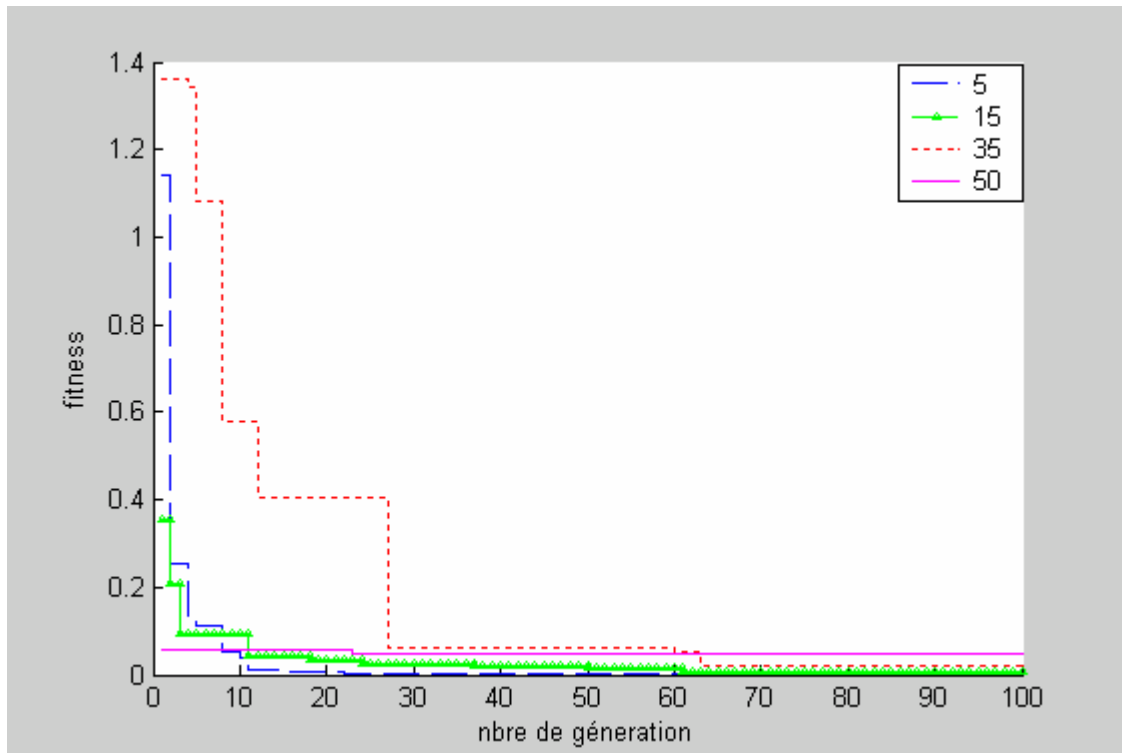


Figure 2-3: Evaluation de la fonction d'adaptation pour différentes longueur de gènes

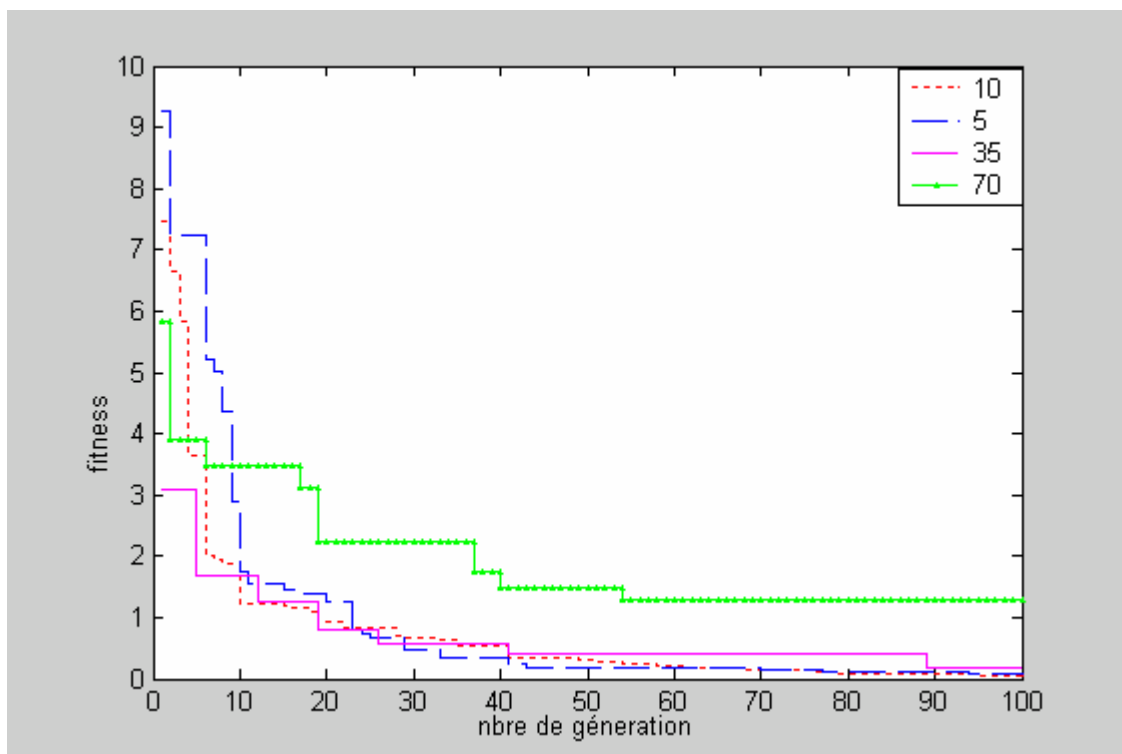


Figure 2-4: Evaluation de la fonction d'adaptation pour différentes longueur de gènes

b)- Influence du taux de transformation

Pour l'étude de l'influence du taux de transformation, la longueur du gène segment est fixée à 5 bits et la longueur du chromosome à 75 pour un nombres de variables ($i = 5$). L'algorithme AGT a été ensuite, exécuté pour déterminer le minimum de la fonction *Rosenbrock* avec un taux de population transformée égale successivement à 30%, 50%, 70%, 90%. Le tableau 2-3, et la figure 2-5 indiquent les différents résultats obtenus.

Tableau 2-3

| T% | N_géné | F1_min | X1min | X2min | X3min | X4min | X5min |
|----|--------|------------|---------|---------|---------|---------|---------|
| 30 | 100 | 0.028141 | 0.9971 | 0.99652 | 0.98706 | 0.98146 | 0.98509 |
| 50 | 52 | 0.0023612 | 1.002 | 1.0034 | 1.0024 | 1.0032 | 1.007 |
| 70 | 25 | 0.00059401 | 0.99792 | 0.99609 | 0.99403 | 0.98895 | 0.97803 |
| 90 | 15 | 1.0863e-05 | 1.001 | 1.001 | 1.003 | 1.002 | 1.002 |

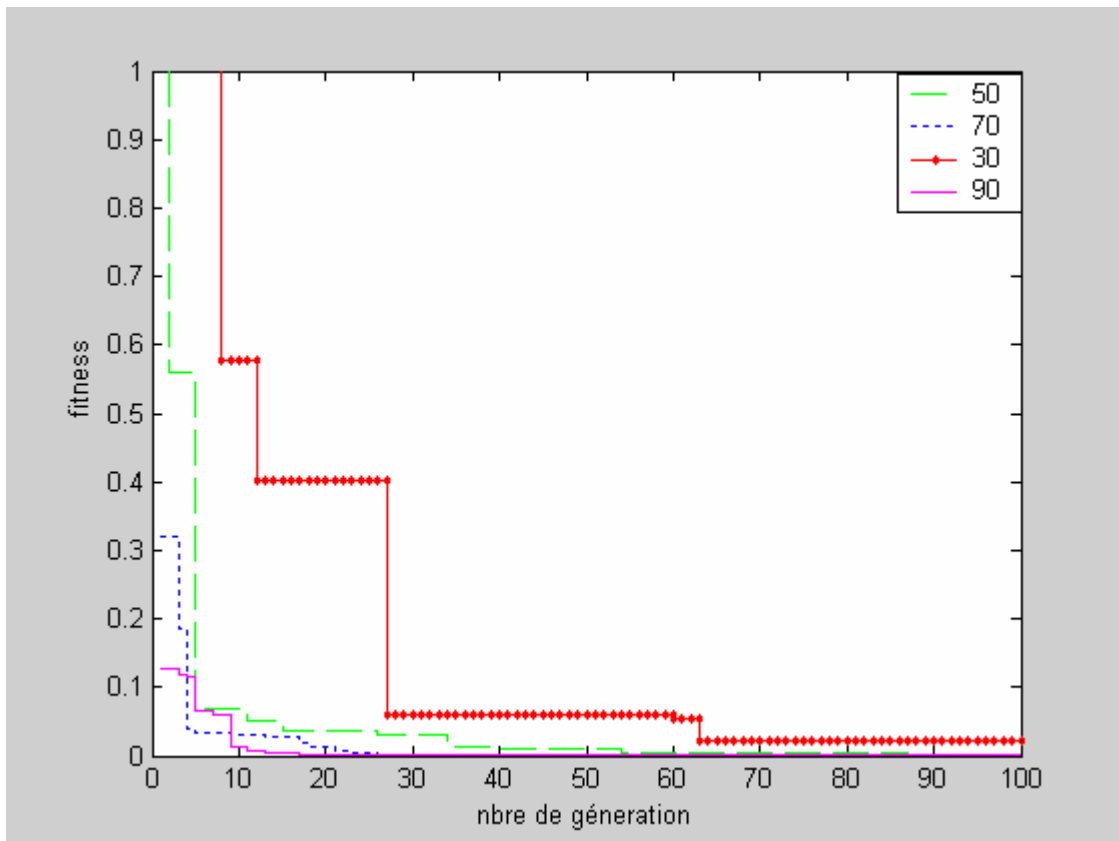


Figure 2-5: Evolution de la fonction d'adaptation pour des différentes taux de population transformée

D'après ces résultats, on peut remarquer que l'utilisation d'un taux de transformation entre 50% et 90% améliore la convergence de l'algorithme, c'est à dire, la valeur minimale de la fonction est obtenue en un temps minimal.

VI. La transposition

VI.1/Principe

La transposition a été découverte pour la première fois par Barbara McClintock dans les années 50 (quand la structure d'ADN n'était pas encore complètement comprise).

La transposition est caractérisée par la présence d'unités génétiques mobiles à l'intérieur du génome. Ces unités mobiles sont appelées "transposon", également connu comme des gènes sauteurs et qui peuvent être constituées d'un ou de plusieurs gènes. Le transposon a longtemps été considéré comme une sorte de caractère anormal, mais en 1983 après que Barbara McClintock ait obtenu le prix Nobel, le rôle des transposons dans l'évolution a été reconnu. Par exemple, les modifications génétiques causées par le transposon sont responsables de l'augmentation des cancers chez l'être humain, ou de la résistance aux antibiotiques dans les bactéries [15].

Le déplacement du transposon peut avoir lieu dans le même chromosome ou dans deux chromosomes différents. Certains transposons se déplacent d'un emplacement sur le chromosome à un nouveau point du même chromosome ou vers un autre chromosome. D'autres transposons laissent une copie. Le point de l'insertion du transposon peut être choisi au hasard, mais il y a des transposons qui montrent une préférence régionale si ils sont inséré dans le même gène. Il existe plusieurs formes de transposition [16] ;

- Transposition simple.
- Transposition basée sur un tournoi.
- Transposition asexuée.

VI.2/Construction du transposon

Si on suppose que **CL** est la longueur du chromosome et **FSL** est la longueur d'une séquence d'un segment appartenant à un gène du chromosome. La méthode de la construction du transposon et la suivante [13,14] :

1. Choisir au hasard un gène (gène **T**) entre **0** et **CL** pour construire le transposon.
2. Les gènes **FSL** immédiatement avant le gène **T** formeront le premier segment.
3. Le second segment doit être identique ou l'inverse du premier, il est choisi à partir du gène **T** sur toute la longueur du chromosome jusqu'à **CL**.
4. Le transposon est formé de tous les gènes entre le gène **T** et le dernier gène du second segment qui se déplace toujours avec le transposon.

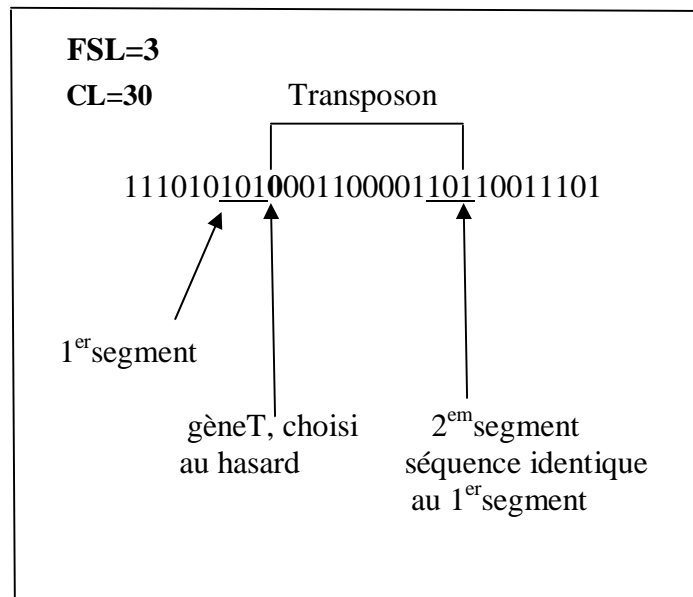


Figure 2-6 : Construction du transposon

VI.3/ Transposition simple

La première forme de la transposition proposée par Simões et al. (1999) a été inspiré directement de la biologie. Après la sélection de deux parents et la construction du transposon sur l'un des d'eux ; le point d'insertion du transposon est défini chez le deuxième parent. Au delà de ce point, le même montant de la matière génétique est échangé entre les deux parents sélectionnés. Le transposon est reconnu par une première séquence d'un segment de gène choisi au hasard sur le premier chromosome, et une deuxième séquence identique ou l'inverse de la première sur le même chromosome, sa longueur est fixe [16].

Le point de l'insertion est défini sur le deuxième chromosome, si une séquence identique ou inverse du segment est trouvée, il sera le premier gène après cette séquence. Après cela, le mouvement du transposon se produit et le même nombre de gènes, de la même longueur du transposon, seront échangés entre les deux parents.

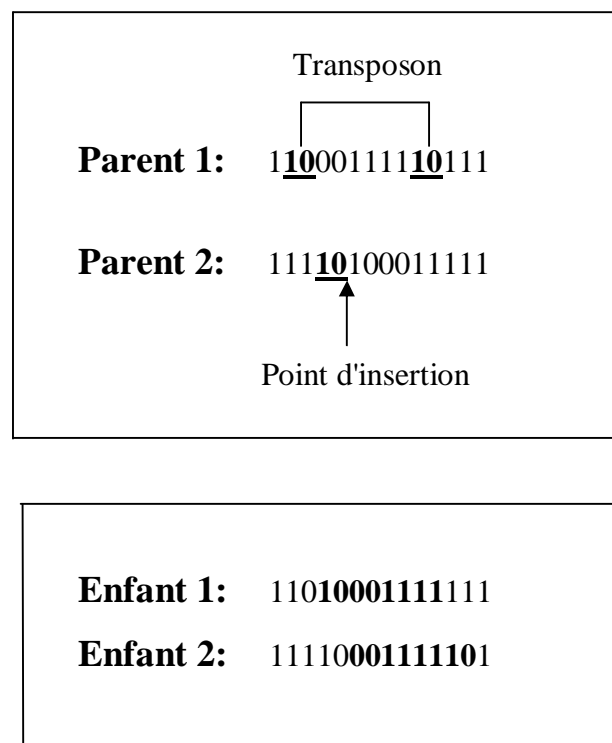


Figure 2-7 : Transposition simple

VI.4/ Transposition basée sur un tournoi

Les deux parents sélectionnés deviennent des concurrents dans un tournoi de dimension deux. Le transposon sera cherché dans le chromosome du vainqueur et le point d'insertion sera localisé chez le parent perdant. Cet individu sera changé, par l'insertion du transposon qui remplace le même nombre de gènes après le point de l'insertion [16].

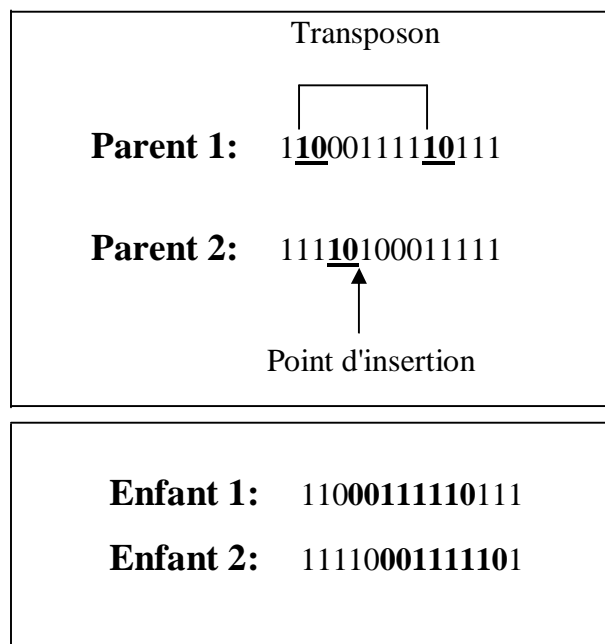


Figure 2-8 : Transposition basée sur un tournoi

VI.5/ Transposition asexuée

Dans la nature le mécanisme de la transposition peut se produire aussi dans le même chromosome. C'est la multiplication asexuée ou végétative, qui s'effectue sans l'intermédiaire de cellules reproductrices. Par conséquent, **Simões et al** (2000) ont rendu effectif une forme de transposition asexuée qui a gardé le principe de fonctionnement du processus biologique. Dans ce cas, la recherche entière du segment, du transposon et le point de l'insertion se produit dans le même individu.

Après avoir sélectionné un individu pour la reproduction, la transposition asexuée sera appliquée. Le début du transposon sera choisi au hasard par un gène **T** sélectionné chez un l'individu. Après une séquence d'un segment de gène de longueur **FSL** pris avant

le gène T, le premier segment sera créé. La recherche du second segment commence après le gène T et s'arrête quand une séquence égale ou inverse sera trouvée. Les gènes voisins du gène T et le dernier gène du second segment constituent le transposon.

Le point d'insertion du transposon est cherché dans le même chromosome, ce processus commence dans le premier gène après le second segment. Ce point est déterminé si une séquence égale ou inverse du segment est trouvée dans le même chromosome. Quand le point de l'insertion est trouvé, le transposon est extrait de sa place originale est intégré après le point de l'insertion [16].

Tout le processus peut être illustré dans l'algorithme suivant:

1/ La sélection du chromosome:

0000111111101111111101010

2/ La construction du transposon :

0000111111101111111101010
 ↑
 gène T (au hasard)

3/ Trouver le Point d'insertion:

00001111111011111111101010
 ↑
 Point d'insertion

4/ Extraction du transposon :

00001 11111111101010
 1111110

5/ L'intégration du transposon:

0000111111111110111111101010

6/ Chromosome obtenu:

000011111111111011111101010

VI.6/ Etude de l'influence de la longueur de la séquence du segment de gènes dans la transposition

Le choix de la longueur de la séquence du gène segment intervenant dans la transposition des gènes est un paramètre très important pour la convergence de l'algorithme. On s'est surtout intéressés à l'influence de cette longueur dans les différents types de transposition. Pour cela, une étude préalable a été nécessaire sur trois algorithmes, AGPS, AGPT et AGPA. Les tests ont été faits sur la fonction *Rosenbrouk* à 3 variables.

Dans le premier cas la longueur du chromosome "CL" a été fixé à 24 pour les trois algorithmes et dans le deuxième cas à 72. La population initiale "N" est fixée à 100 pour les deux cas ainsi que la probabilité de mutation à 0.03. Les algorithmes AGPS, AGPT et AGPA ont été exécutés avec un même nombre de génération. Les longueurs de la séquence du segment de gènes "FSL" utilisés sont: 3, 5, 7 et 9. Les résultats obtenus dans chaque cas, sont indiqués successivement dans les tableaux 2-4 à 2-9, et les figures 2-9 à 2-14.

a)-Transposition simple (AGPS)

1° Cas: CL=24

Tableau 2-4

| FSL | F1_max | X1min | X2min | X3min |
|-----|---------|---------|---------|---------|
| 3 | 0.99678 | 0.97683 | 0.9522 | 0.90645 |
| 5 | 0.9295 | 0.87059 | 0.76353 | 0.58235 |
| 7 | 0.85762 | 0.80544 | 0.64712 | 0.41205 |
| 9 | 0.82954 | 0.80625 | 0.675 | 0.45663 |

2° Cas: CL=72

Tableau 2-5

| FSL | F1_max | X1min | X2min | X3min |
|-----|---------|---------|---------|---------|
| 3 | 0.71451 | 0.675 | 0.46256 | 0.21566 |
| 5 | 0.7749 | 0.73242 | 0.53535 | 0.28125 |
| 7 | 0.96811 | 0.91624 | 0.83906 | 0.70783 |
| 9 | 0.90328 | 0.83906 | 0.70782 | 0.7909 |

b) -Transposition basée sur un tournoi (AGPT)***1° Cas: CL=24******Tableau 2-6***

| FSL | F1_max | X1min | X2min | X3min |
|-----|---------|---------|---------|---------|
| 3 | 0.99693 | 1.0216 | 1.0408 | 1.082 |
| 5 | 0.97638 | 1.0467 | 1.0101 | 1.023 |
| 7 | 0.88608 | 0.83908 | 0.69657 | 0.47812 |
| 9 | 0.8027 | 0.7635 | 0.57412 | 0.32706 |

2° Cas: CL=72***Tableau 2-7***

| FSL | F1_max | X1min | X2min | X3min |
|-----|---------|---------|---------|---------|
| 3 | 0.78263 | 0.73652 | 0.54375 | 0.29565 |
| 5 | 0.86925 | 0.80788 | 0.65395 | 0.42659 |
| 7 | 0.89825 | 0.84021 | 0.70499 | 0.49453 |
| 9 | 0.99908 | 0.98437 | 0.97504 | 0.95086 |

c) -Transposition asexuée (AGPA)***1° Cas: CL=24******Tableau 2-8***

| FSL | F1_max | X1min | X2min | X3min |
|-----|---------|---------|---------|---------|
| 3 | 0.99906 | 1.0024 | 1.0024 | 1.0106 |
| 5 | 0.9742 | 0.92824 | 0.86235 | 0.73882 |
| 7 | 0.88332 | 0.92 | 0.87882 | 0.78 |
| 9 | 0.67337 | 0.64824 | 0.40941 | 0.17059 |

2° Cas: CL=72***Tableau 2-9***

| FSL | F1_max | X1min | X2min | X3min |
|-----|---------|---------|---------|---------|
| 3 | 0.83297 | 0.78156 | 0.60937 | 0.37152 |
| 5 | 0.86925 | 0.80788 | 0.65395 | 0.42659 |
| 7 | 0.9652 | 0.91292 | 0.83261 | 0.69128 |
| 9 | 0.97438 | 0.93751 | 0.87187 | 0.75695 |

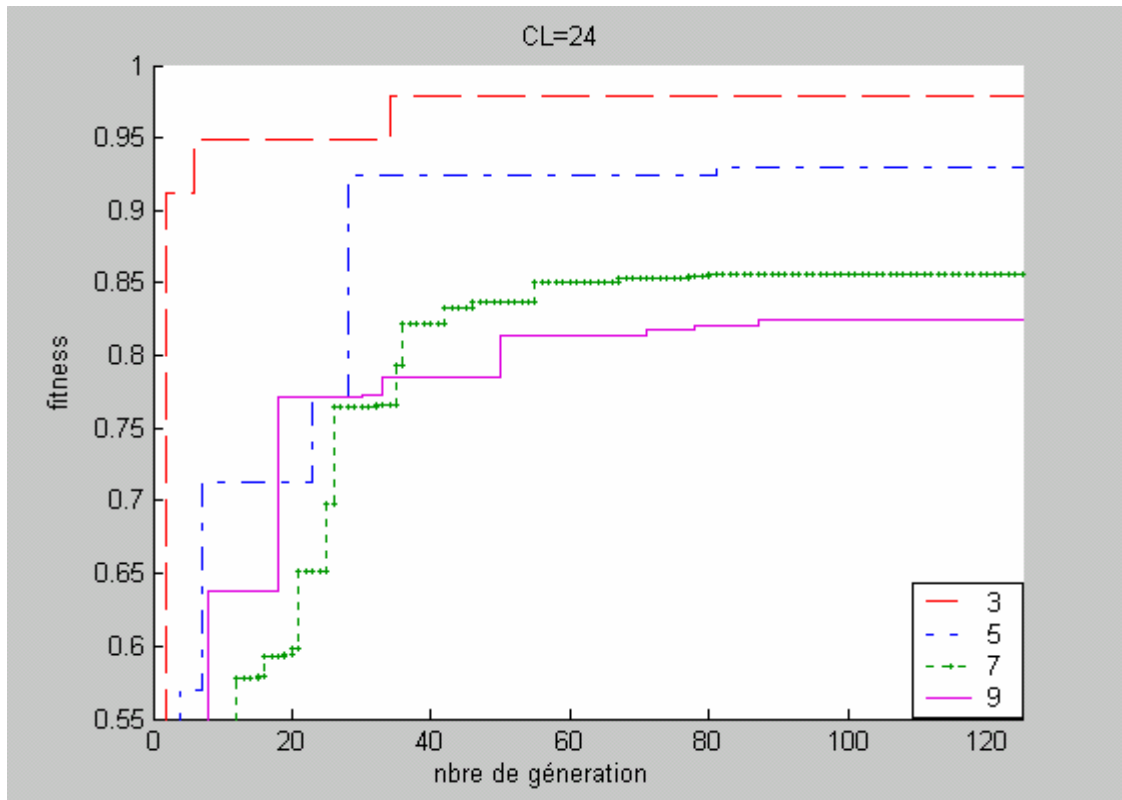
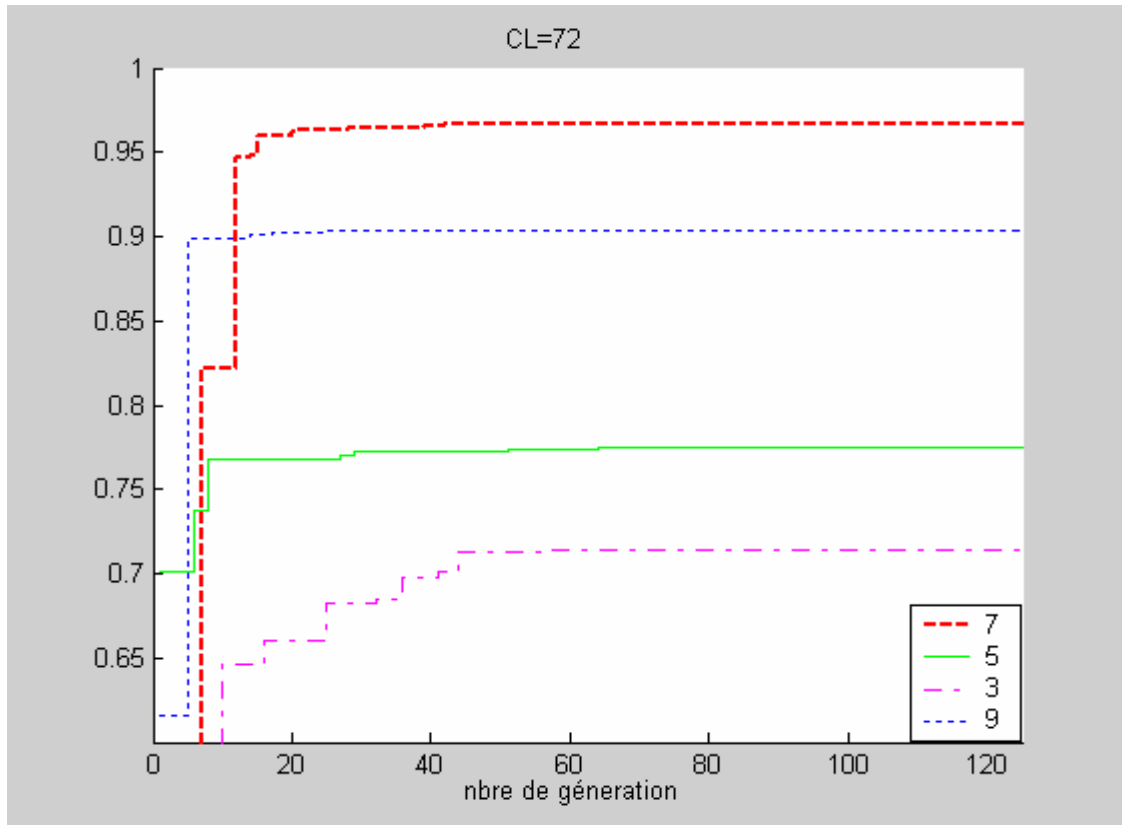
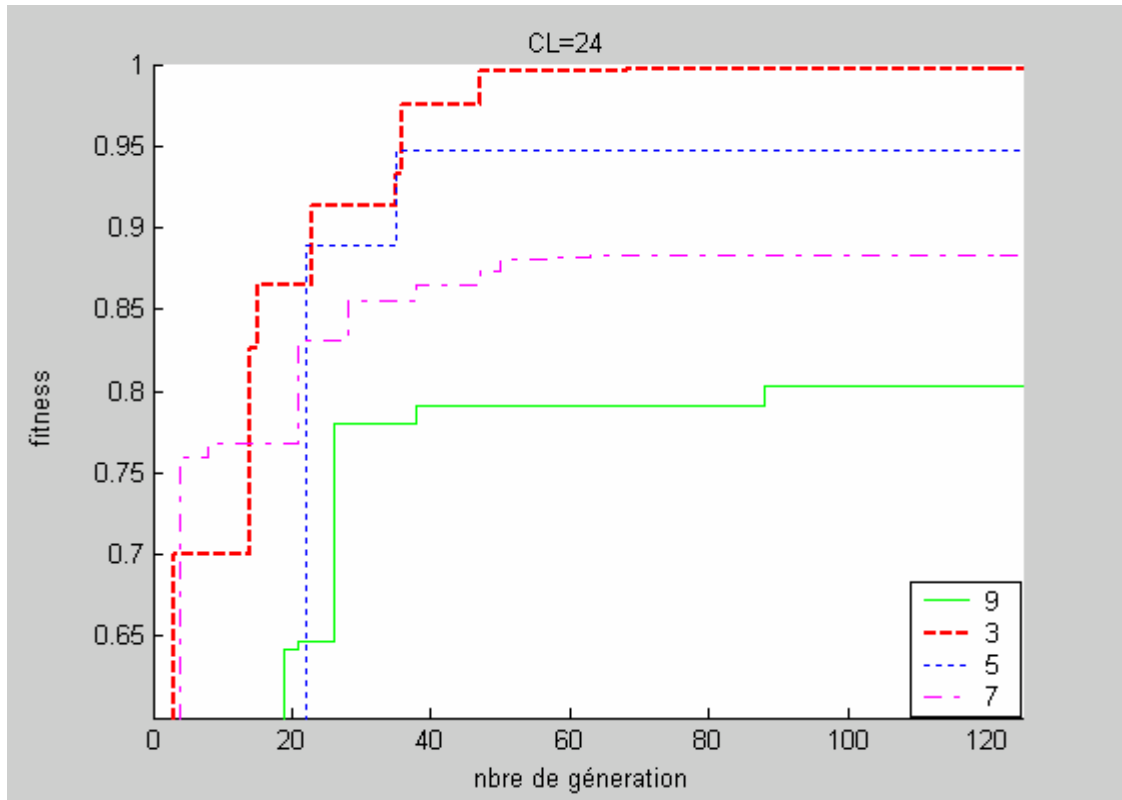


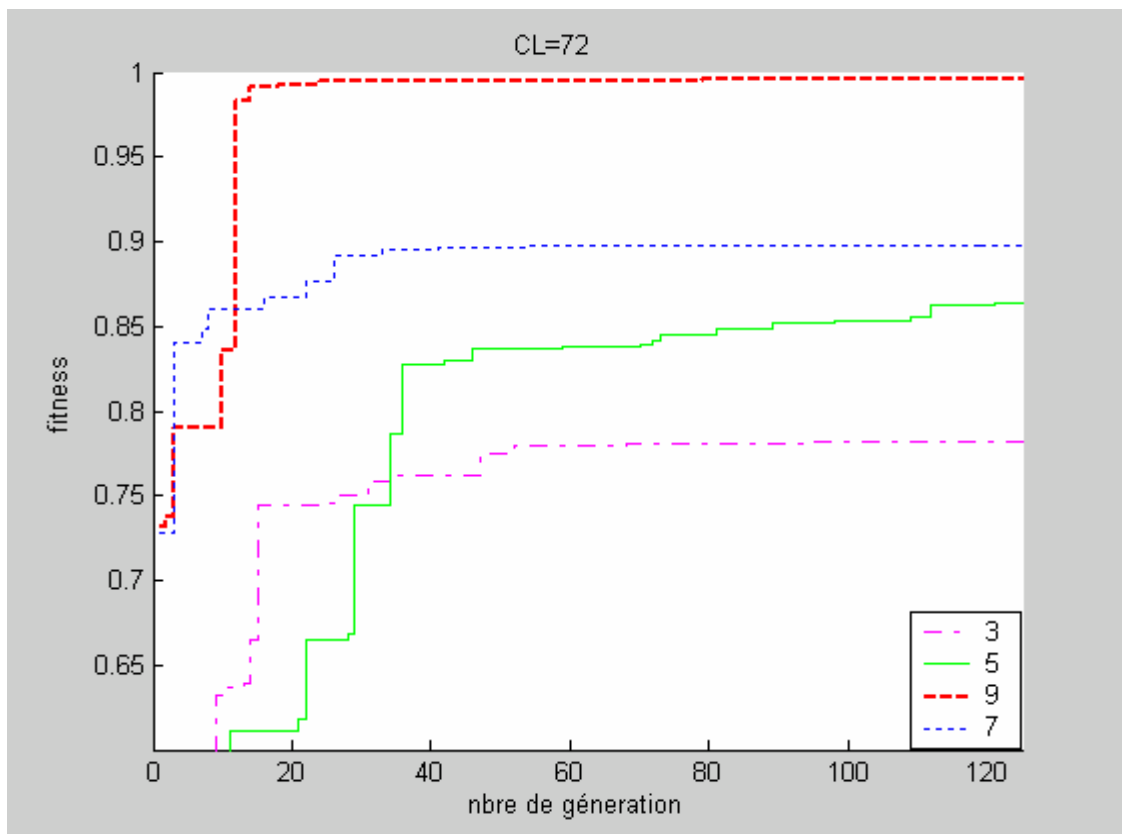
Figure 2-9: Evaluation de la fonction d'adaptation pour différentes longueur de séquence des gènes pour un "AGPS"



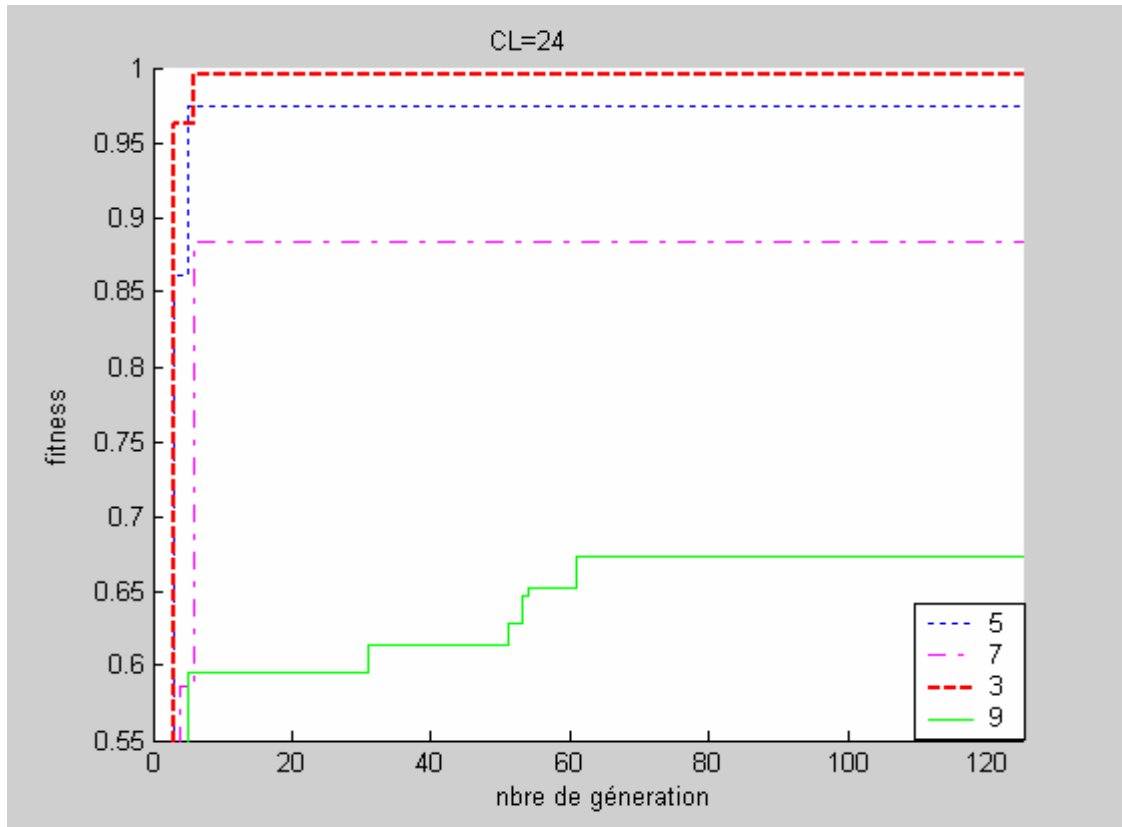
Figures 2-10: Evaluation de la fonction d'adaptation pour différentes longueur de séquence des gènes pour un "AGPS"



Figures 2-11: Evaluation de la fonction d'adaptation pour différentes longueur de séquence des gènes pour un "AGPT"



Figures 2-12: Evaluation de la fonction d'adaptation pour différentes longueur de séquence des gènes pour un "AGPT"



Figures 2-13: Evaluation de la fonction d'adaptation pour différentes longueur de séquence des gènes pour un "AGPA"

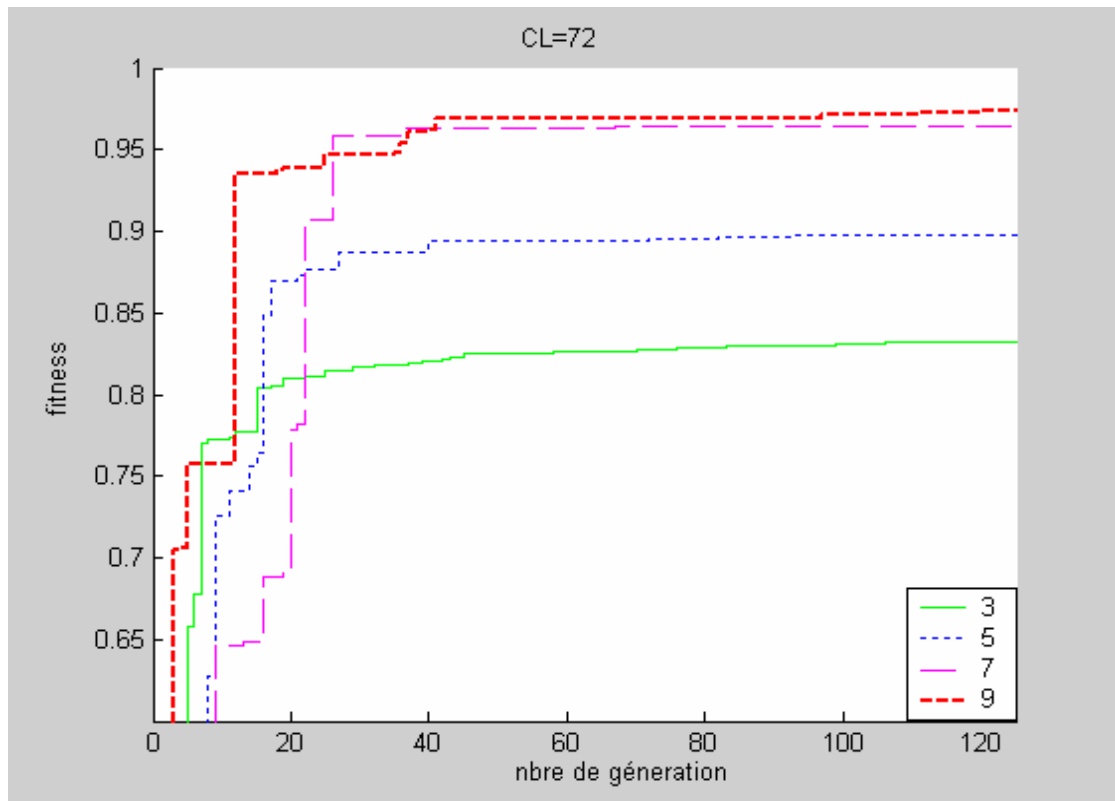


Figure 2-14: Evaluation de la fonction d'adaptation pour différentes longueur de séquence des gènes pour un "AGPA"

Les résultats obtenus dans cette partie montrent que pour des séquences de gène segment de longueur 7 et 9 bits et une longueur de chromosome égale à 24 bits l'optimum obtenu n'est pas satisfaisant dans les différentes formes de transposition soit: AGPS, AGPT ou AGPA. Mais pour des séquences de gène segment de longueur 3 et 5 bits les valeurs de l'optimum obtenu sont meilleures dans les trois formes, alors que pour une longueur de chromosome égale à 60 bits. Les meilleures valeurs sont obtenues pour FSL égale à 7 et 9 bits.

Les différents résultats montrent également que, le choix des séquences "FSL" doit être fait à travers la longueur du chromosome. On peut voir que, quand la longueur du chromosome est petite les valeurs obtenus sont semblables pour les même longueurs de séquence, alors que lorsque on utilise des chromosomes plus importants il faut choisir des longueurs de séquences plus élevés. La longueur de la séquence "FSL" dépend de la longueur du chromosome, il faut donc savoir comment choisir cette longueur pour atteindre plus rapidement la valeur de l'optimum.

VI.7/ Etude de la convergence des différents algorithmes de la transposition

1° Choix des paramètres génétiques

L'étude comparative des performances des différents algorithmes proposés a nécessité des exécutions sur les fonctions tests. La minimisation de ces fonctions en utilisant les algorithmes proposés nécessite un bon choix des paramètres génétiques : le nombre de générations maximal, le nombre d'individus, la longueur du chromosome, et la probabilité de mutation (Pmut).

En plus, de ces paramètres on a précisé les valeurs des paramètres caractéristiques propres aux algorithmes AGPS, AGPT, AGPA et AGS; à savoir la longueur de la séquence du gène segment "FSL". Le tableau 2.10 illustre les fonctions étudiées avec leurs domaines de recherche et donne les valeurs des paramètres génétiques utilisés pour la simulation.

Tableau 2-10

| | <i>f 1</i> | <i>f 2</i> |
|--|------------|------------|
| Longueur du chromosome | 96 | 100 |
| Taille de la population (N) | 100 | 100 |
| Nombre de variable (Xi) | 3 | 5 |
| Nombre de génération max. | 100 | 100 |
| Pmut | 0.09 | 0.09 |
| Intervalle | [-2 2] | [-2 2] |
| Longueur de la séquence du gène segment (FSL) | 6 | 6 |

2°/Etude comparative des performances des différents algorithmes

Les algorithmes (AGPS, AGPT, AGPA aussi qu'un algorithme génétique simple AGS) ont été successivement appliqués sur les deux fonctions, en utilisant les paramètres génétiques mentionnés précédemment dans le tableau 2-10.

Les résultats obtenus après plusieurs tests sont représentés sur les figures 2-15, 2-16. Ces figures montrent la variation de la fonction d'adaptation par rapport au nombre de générations pour les différents algorithmes et pour chacune des fonctions à minimiser.

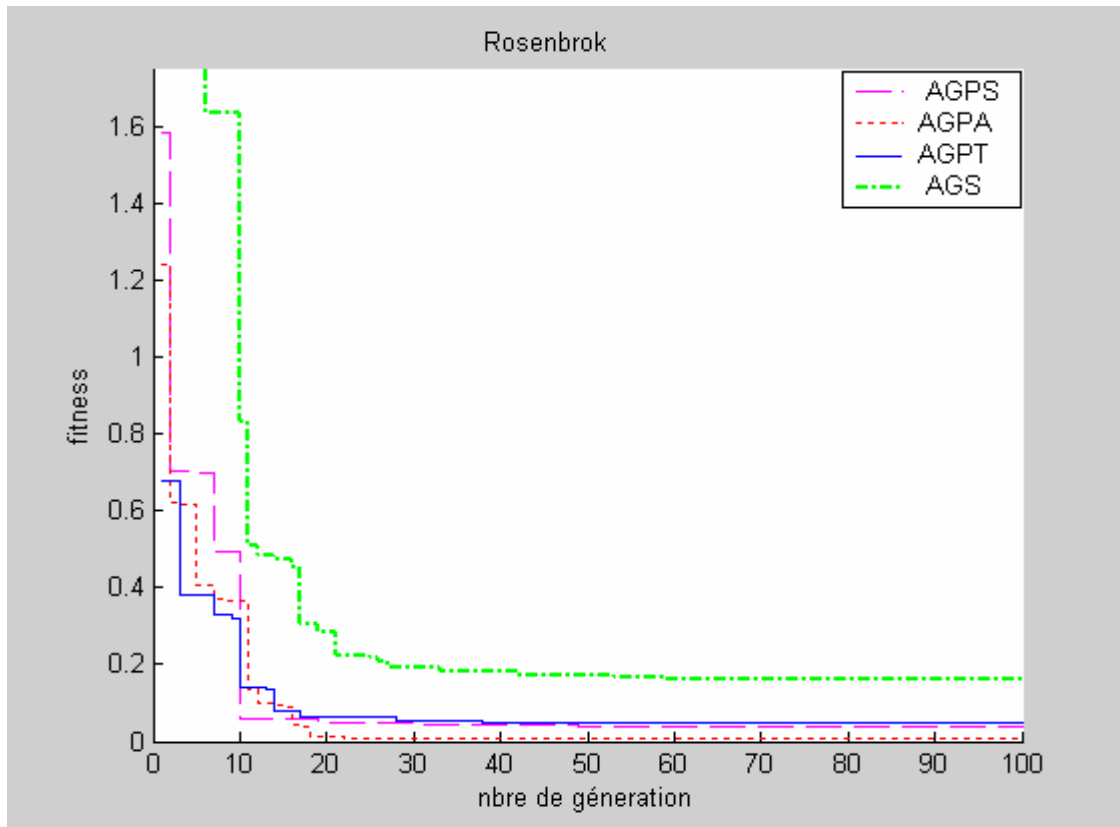


Figure 2-15: Evaluation de la fonction f_1 au cours des générations

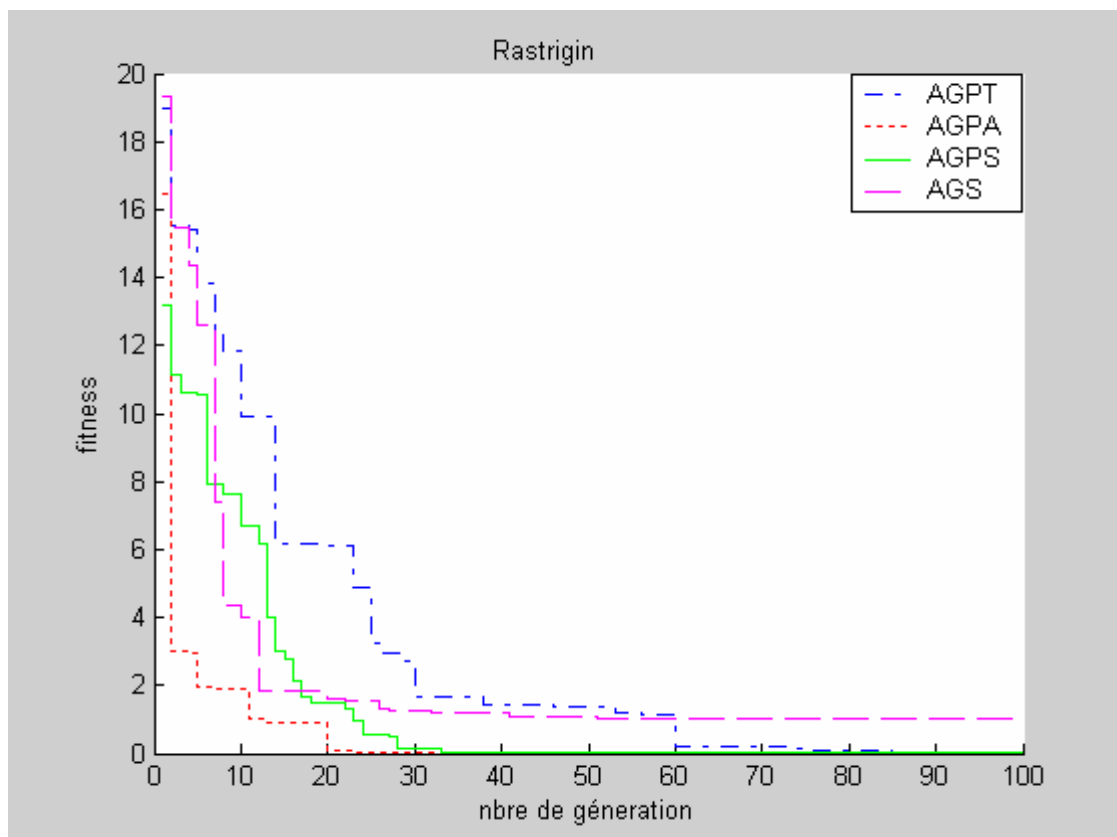


Figure 2-16: Evaluation de la fonction f_2 au cours des générations

D'après les courbes obtenues, on remarque que pour les deux fonctions, la valeur minimale est atteinte en utilisant l'algorithme AGPA et ce la en un nombre minimal de générations.

L'algorithme AGPT permet d'obtenir des résultats acceptables et qui sont très proches que ceux trouvés en utilisant l'AGPS. Avec l'algorithme AGS, la valeur minimale exacte n'est pas atteinte même après plusieurs générations. Ce dernier peut converger vers la solution mais au prix d'un nombre assez important de générations.

Donc, on peut dire que l'utilisation de la forme asexuée de la transposition améliore la convergence des algorithmes génétiques simples et réduit le temps de calcul.

Pour mieux comparer les résultats obtenus, on a proposé de déterminer la valeur minimale et le nombre de générations nécessaires pour atteindre l'optimum de chaque fonction. Ces résultats sont illustrés dans le tableau 2-11 pour les trois formes d'algorithmes de transposition proposés.

Tableau 2-11

| | | AGSP | AGTP | AGPA | AGS |
|------------|-------------|--------------|------------|-------------|------------|
| <i>f</i> 1 | N_géné | 48 | 50 | 22 | 100 |
| | F1_min | 0.035567 | 0.047928 | 0.0079867 | 0.15894 |
| | X1 | 0.91277 | 0.89823 | 1.0392 | 0.80625 |
| | X2 | 0.83238 | 0.80625 | 1.0808 | 0.65455 |
| | X3 | 0.69231 | 0.65001 | 1.1693 | 0.42892 |
| <i>f</i> 2 | N_géné | 32 | 72 | 22 | 100 |
| | F2_min | 2.6921e-007 | 0.0033783 | 4.556e-007 | 0.99619 |
| | X1 | 2.0981e-005 | 0.00068092 | 3.7193e-005 | -0.99764 |
| | X2 | -1.3351e-005 | -0.0031948 | 3.7193e-005 | 0.014098 |
| | X3 | 2.0981e-005 | -0.0019703 | 2.861e-006 | 0.014564 |
| | X4 | 1.3351e-005 | 0.001688 | 3.7193e-005 | 0.00048733 |
| X5 | 1.7166e-005 | 0.0016956 | 2.861e-006 | 0.0022535 | |

Résultats de l'étude statistiques

Pour confirmer l'efficacité des différentes techniques proposées, une étude statistique a été faite sur les différents algorithmes étudiés précédemment. Les résultats obtenus, sont indiqués dans le tableau 2-12.

Tableau 2-12

| | AGSP | AGTP | AGPA | AGS |
|-----------------------------|-------------|-------------|-------------|------------|
| Nombre d'exécutions | 50 | 50 | 50 | 50 |
| Taux de réussite (%) | 90 | 92 | 96 | 52 |

Ces résultats confirment le fait que l'utilisation de la transposition avec ces différentes formes donne de meilleurs résultats par rapport à l'algorithme génétique simple. On remarque aussi que le minimum des deux fonctions est atteint pendant les premières générations de l'AGPA (de 20 à 22 générations). Pour les deux autres algorithmes, l'AGPT et l'AGPS, on remarque quelque ressemblance dans leur comportement.

Dans ces deux situations la performance de la transposition est légèrement inférieure à celle de la forme asexuée.

L'utilisation de la transposition donne de bons résultats car on obtient la valeur minimale désirée pour chacune des fonctions. On peut conclure que l'importance de la transposition se trouve dans " des bonnes " solutions qui savent que l'échange des gènes est contrôlé par un contexte défini par les séquences des gènes segment.

V. Stratégie de la recherche de niche**V.1/Introduction**

Dans la nature, les ressources telles que l'eau et la nourriture présentes dans un écosystème sont limitées et doivent être partagées par toutes les espèces vivantes dans cet écosystème. S'il y a abondance de ressources les espèces prospèrent et se multiplient, s'il y a un manque de ces ressources le nombre d'individus des espèces diminue et seuls les plus aptes survivent. Inspiré du principe des niches écologiques et de l'équilibre que tente de maintenir la nature, John Holland a proposé le principe du nichage dans les algorithmes

évolutionnaires. Ici les niches sont caractérisées par un ensemble d'individus, présentant certaines ressemblances, qui partagent une ressource commune "la fonction d'adaptation". Parallèlement, les optima de la fonction objective peuvent être vus comme les endroits où les niches se formeront. Le but du nichage est donc de faire recouvrir équitablement ces endroits par la population [4].

Dans la sélection génétique la pression de la sélection, va favoriser une certaine ressemblance entre les individus de la population. Cette ressemblance se traduit directement par une perte de diversité dans la population, et la convergence des individus vers une région de l'espace qui n'est pas favorable. Elle va attirer une grande partie de la population et réduire les chances d'atteindre l'optimum global. Pour éviter cette convergence prématurée et maintenir la diversité génétique de la population on a introduit le mécanisme de nichage. Pour cela le concept des niches écologiques qui définit une région de l'espace autour d'une solution considérée, a été utilisé. Cette technique est apparue pour remédier certaines insuffisances des AG classiques [4].

Dans la nature les espèces appartenant à des niches écologiques différentes évoluent de façons indépendantes, et ne sont pas en compétition pour les mêmes ressources. Récemment des méthodes de nichage utilisant plusieurs sous populations évoluant parallèlement; ici la population globale est divisée en plusieurs sous populations et chacune évolue indépendamment du reste, permettant ainsi l'exploration des différentes parties de l'espace de recherche. Occasionnellement, les différentes populations échangent des individus, "migration", c'est le seul moyen de communication entre populations [4].

V.2/ Recherche des niches

L'optimisation génétique basée sur la recherche de niches est une approche qui est caractérisée par une recherche évolutionnaire à deux niveaux : Au niveau des l'individus, et au niveau des niches.

La recherche de niche est une nouvelle version d'algorithmes génétiques. Elle est inspirée des opérations de reproduction présentes dans la nature au niveau cellulaire et moléculaire, mais avec plus de précision que la combinaison proposé dans un algorithme génétique simple. En particulier, elle propose un nouveau procédé de méiose au niveau de la reproduction génétique.

La méiose est la division de la cellule aboutissant à la réduction de moitié du nombre des chromosomes. Se produisant au moment de la formation des cellules reproductrices, elle est la source principale de la diversité dans la nature pour la reproduction sexuelle [17].

La population globale d'individus (l'écosystème) est assemblée dans différents groupes; chacun de ces groupes occupe une place particulière. Dans l'approche des niches chaque niche fournit son propre ensemble de paramètres qui est arbitrairement choisi. Enfin la sélection et la reproduction agissent non seulement au niveau des individus mais également au niveau des niches, les individus doivent concourir afin de survivre parmi leur ensemble. Les niches doivent également concourir entre elles pour les ressources dans l'environnement.

La sélection des niches signifie qu'à chaque génération les niches qui sont mal adapté peuvent disparaître, et celles dont l'adaptation est bonne restent. La force des paramètres de commande de l'algorithme est faite par une approche évolutionnaire contrairement à ce qui se produit dans les systèmes évolutionnaire ES et dans la programmation évolutionnaire PE [17].

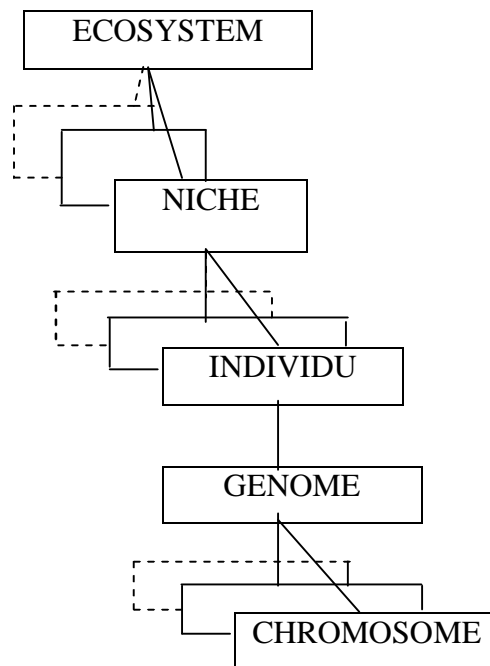


Figure 2-17: Structure de l'environnement des niches

V.3/ L'évolution des niches

Ce qui caractérise une niche dans un écosystème, c'est son ensemble d'individus et les paramètres de l'environnement où la niche est localisée (le facteur de reproduction). Les paramètres sont imposés extérieurement et aléatoirement par l'écosystème, rien ne dirige les niches vers les meilleurs paramètres de reproduction.

Les niches dont les paramètres sont bons sont susceptibles de s'adapter plus rapidement dans l'environnement, et par conséquent restent vivantes, mais celles dont les paramètres sont mauvais sont moins susceptibles de s'adapter et auront tendance à disparaître. Les paramètres de reproduction imposés aux niches sont : la probabilité de la mutation des gènes, la probabilité de croisement dans le chromosome, et le facteur de sélectivité [14].

L'évolution aux niveaux des niches est basée sur les opérateurs suivants [17]:

Sélection des niches : les niches mal adaptées vont disparaître, tandis que les meilleures restent.

Recombinaison des niches : les niches qui disparaissent sont remplacées par de nouvelles niches.

La mutation des niches : les paramètres qui caractérisent chaque niche sont aléatoirement changés (mutés). Ils vont subir une mutation à chaque génération. Cet opérateur est inspiré par le fait que les paramètres équivalents changent dans la vie normale sans interruption.

L'algorithme suivant explique l'opération globale et le mécanisme de la recherche de niche.

V.4/Algorithme

Initialisation de l'écosystème:

1. Création d'un nombre donné de niches.
2. Initialisation aléatoire des paramètres qui caractérisent chaque niche (Probabilité de mutation et de croisement).
3. Initialisation aléatoire de la population initiale de chaque niche.
4. Evaluation des niches: évaluation de la fitness de chaque niche, cette évaluation est basée sur le rapport entre la moyenne des fitness de chaque niche et la meilleurs fitness. On appelle ce rapport "facteur d'évaluation globale de la niche".
5. Début du processus d'évolution :
6. Sélection des meilleures niches; elles survivront à la prochaine génération.
7. Sélection des mauvaises niches; elles peuvent disparaître.
8. Création de nouvelles niches par la recombinaison de meilleures niches avec de nouvelles niches créées aléatoirement. Elles remplaceront les niches disparues.
9. Initialisation aléatoire des paramètres de chaque nouvelle niche.
10. Evaluation des nouvelles niches : le nouveau écosystème et la sélection des meilleures niches
11. La mutation de chaque niche avec une certaine probabilité mise aléatoirement dans chaque niche.
12. Création d'une nouvelle génération des individus dans chacune des niches

V.5/ Les Niches et l'évolution des individus

A l'intérieur de chaque niche, les individus sont soumis à la concurrence entre eux. La sélection et la reproduction à l'intérieur d'une niche suivent les règles suivantes :

1. Initialisation de la sous population.
2. Sélection de la sous population pour la création des enfants.
3. début du processus d'évolution :
4. Sélection des parents avec une méthode de sélection choisie.
5. Recombinaison génétique utilisant l'opérateur de méiose.
6. Mutation génétique avec une probabilité aléatoire mise dans chacune des niches.
7. Evaluation de la nouvelles génération (enfants) dans chaque niche.

V.6/ La recherche de niche et l'opérateur de méiose

La recherche de niche est une nouvelle méthode d'optimisation globale possible conçue et mise en application pour essayer de mettre en parallèle le plus étroitement les opérateurs de recherche avec leurs équivalents biologiques. Les opérateurs de mutation et de croisement sur la structure génétique sont complétées par un procédé de méiose qui est absent dans un algorithme génétique simple (AGS). L'opérateur de méiose est employé pour le traitement des problèmes de grande dimension. Dans l'approche standard consiste à un AGS, et pour un problème d'optimisation multidimensionnelle assembler toutes les variables dans un seul chromosome, perdant ainsi toute l'indépendance des variables. Avec l'opérateur de méiose, cette indépendance est garantie car à chaque variable est associée à un chromosome particulier, comme c'est indiqué dans la figure 2-18. Le but de la recherche de niche est de produire la meilleure solution dans un intervalle de temps très réduit, afin d'obtenir des solutions plus précises. Le fait que les différentes niches conduisent davantage à faire des recherches dans les différentes régions de l'espace de recherche, augmente fortement la probabilité d'éviter des optimums locaux [17].

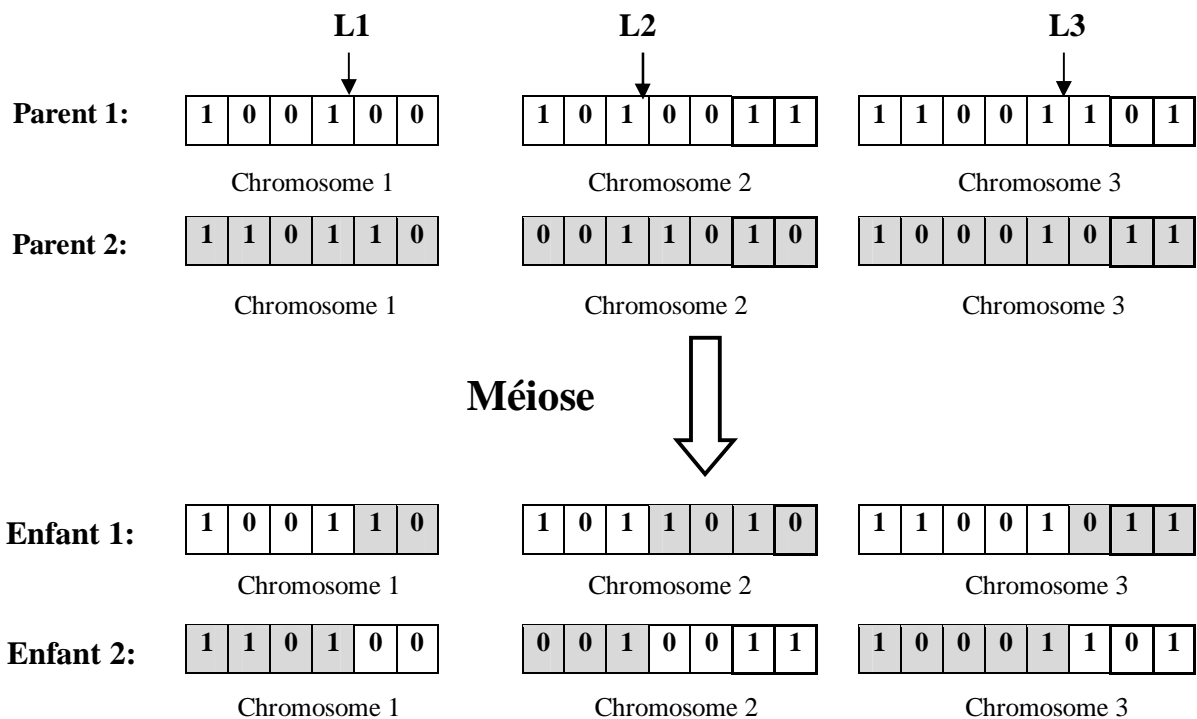


Figure 2-18 : Opérateur de méiose

L1, L2, L3: site de croisement.

IV. Etude de la convergence des différents algorithmes en utilisant la stratégie de la recherche des niches

Pour cette partie, nous allons présenter les différents résultats obtenus pour chacune des procédures présentées précédemment. Pour cela, nous avons utilisé les deux fonctions tests à 10 variables : du tableau (2-13).

- La fonction Rosenbrock.
 - La fonction Rastrigin.
1. **AGS**: Algorithme génétique simple utilisant une sélection proportionnelle, un croisement à un site, une mutation simple pour chaque paramètre du chromosome et la sélection des individus de chaque nouvelle génération par la sélection de procréation sélective.
 2. **AGT** : Algorithme génétique utilisant la transformation des gènes, une sélection simple et une mutation entre chaque paramètre.
 3. **AGPS** : Algorithme génétique utilisant la transposition simple des gènes, une sélection simple et une mutation entre chaque paramètre.
 4. **AGPT** : Algorithme génétique utilisant une transposition des gènes basé sur un tournoi, une sélection simple et une mutation entre chaque paramètre.
 5. **AGPA** : Algorithme génétique utilisant une transposition des gènes asexuée, une sélection simple et une mutation entre chaque paramètre.
 6. **AGM** : Algorithme génétique simple utilisant l'opérateur de méiose une sélection simple et une mutation entre chaque paramètre.

Tableau 2-13

| Nom | Fonction | F_min | X_min | Intervalle |
|---------------------------------|---|-------|-----------|------------|
| <i>Rosenbrock</i> <i>f 1</i> | $f(x_i / i=1,n) = 100 \times (x_i^2 - x_{i-1})^2 + (1 - x_i)^2$ | 0 | $X_i = 1$ | [-2 2] |
| <i>Rastrigin</i> <i>f 2</i> | $f(x_i / i=1,n) = n * 10 + \left(\sum_{i=1}^n (x_i^2 - 10 * \cos(2\pi x_i)) \right)$ | 0 | $X_i = 0$ | [-2 2] |

Les paramètres génétiques utilisés sont ceux indiqués dans le tableau 2-14.

Tableaux 2-14

| | <i>f 1</i> | <i>f 2</i> |
|---|------------|------------|
| Longueur du chromosome | 120 | 120 |
| Taille de la population (N) | 100 | 100 |
| Nombre de variable (Xi) | 10 | 10 |
| Nombre de génération max. | 50 | 50 |
| Pmut | 0.09 | 0.09 |
| Pcroi | 0.9 | 0.9 |
| Longueur de la séquence du gène segment (FSL) | 6 | 6 |
| Taux de transformation (T%) | 70% | 70% |
| Longueur du gène segment (PG) | 10 | 10 |
| Intervalle | [-5 5] | [-5 5] |

Les algorithmes (AGT, AGPS, AGPT, AGPA, et AGM) ont été successivement appliqués sur les deux fonctions, en utilisant la stratégie de la recherche des niches.

Les résultats obtenus après plusieurs tests sont représentés sur les figures 2-19 et 2-20. Ces figures montrent la variation de la fonction d'adaptation par rapport au nombre de générations pour les différents algorithmes étudiés en même temps et pour chacune des fonctions à minimiser. Tous les résultats sont illustrés dans le tableau 2-16.

Résultats de l'étude statistiques

Les résultats obtenus de l'étude statistiques, sont indiqués dans le tableau 2-15.

Tableau 2-15

| | AGS | AGM | AGT | AGPS | AGPT | AGPA |
|----------------------|-----|-----|-----|------|------|------|
| Nombre d'exécutions | 50 | 50 | 50 | 50 | 50 | 50 |
| Taux de réussite (%) | 50 | 84 | 96 | 88 | 88 | 92 |

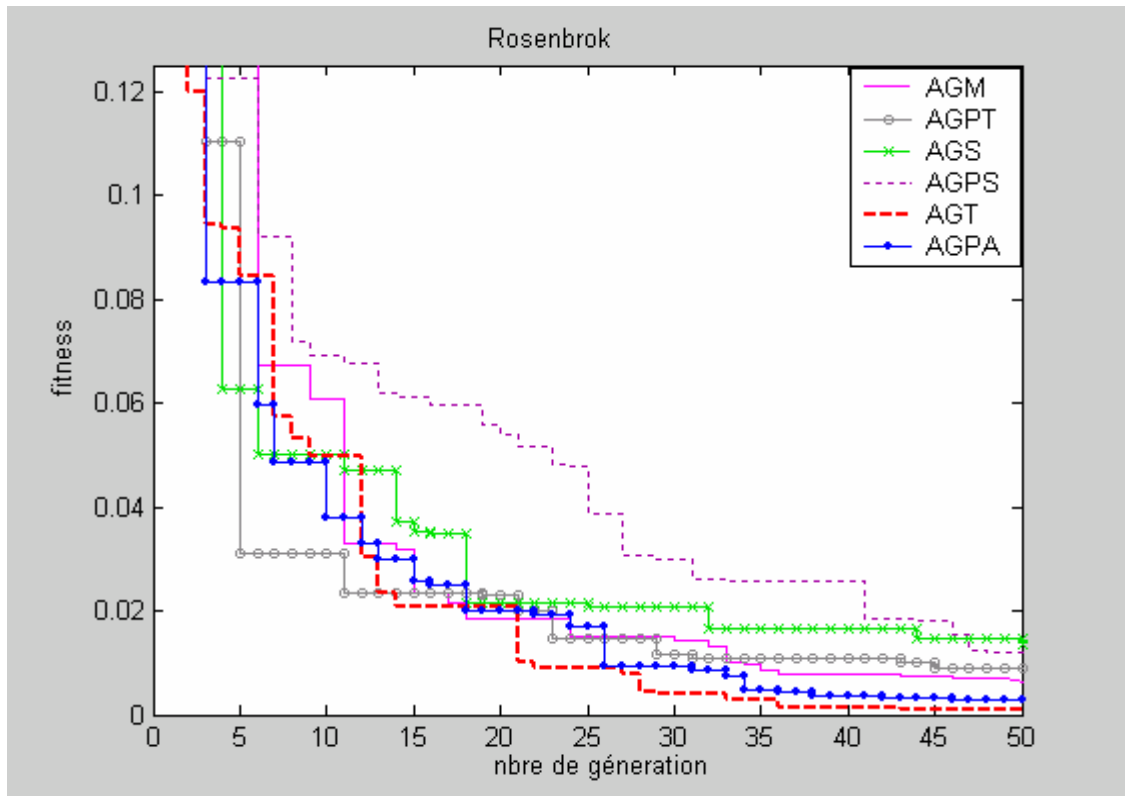


Figure 2-19: Evolution de la fonction f_1 au cours des générations

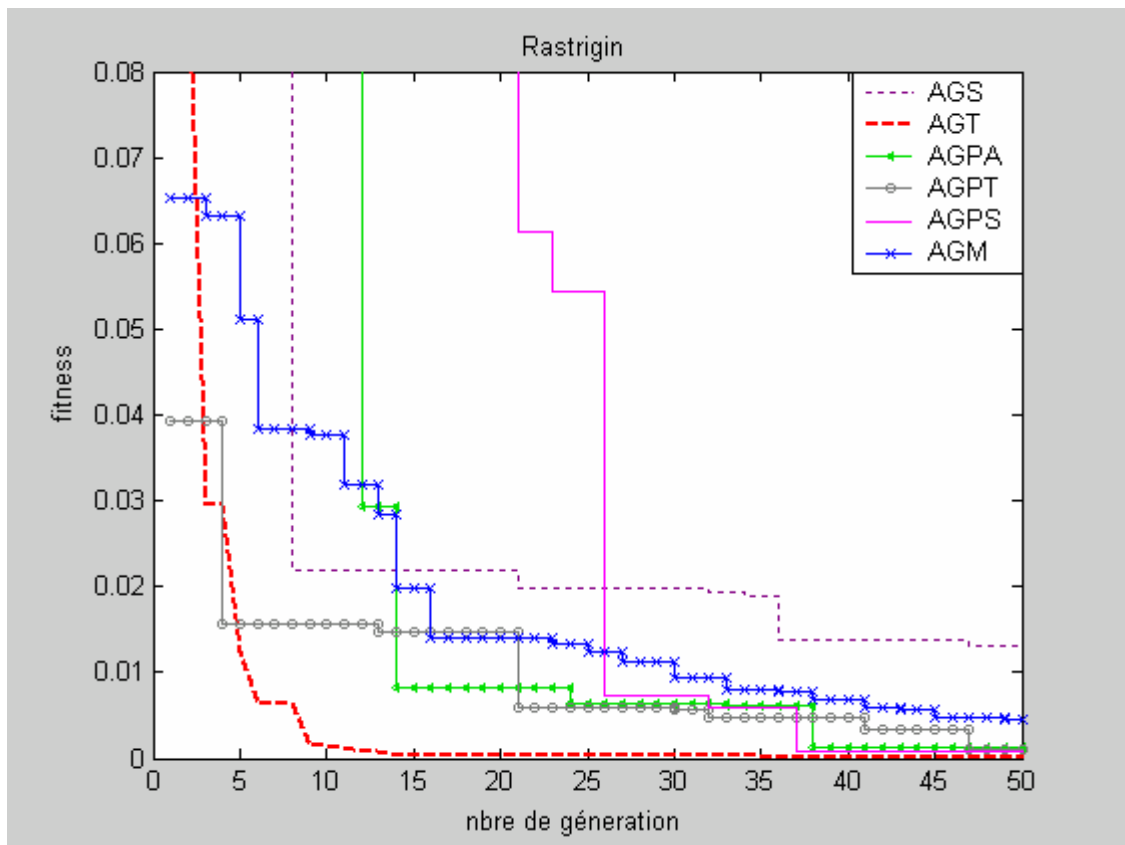


Figure 2-20: Evaluation de la fonction f_2 au cours des générations

Tableaux 2-16

| | | AGS | AGM | AGT | AGPS | AGPT | AGPA |
|-----------|---------------|------------|------------|------------|------------|------------|------------|
| <i>f1</i> | N_géné | 50 | 48 | 36 | 48 | 45 | 39 |
| | F1_min | 0.010986 | 0.0045224 | 0.00086816 | 0.0028938 | 0.0026566 | 0.0023559 |
| | X1 | 0.99989 | 0.99841 | 1.001 | 1.0007 | 1.002 | 0.99932 |
| | X2 | 0.99608 | 0.99864 | 1.001 | 0.99902 | 1.0014 | 0.99852 |
| | X3 | 0.99585 | 0.99856 | 1.001 | 0.99966 | 1.001 | 0.99944 |
| | X4 | 0.99825 | 0.99734 | 1.001 | 1.0009 | 1.0006 | 1.0002 |
| | X5 | 1.0012 | 0.99608 | 1.001 | 0.99928 | 0.99974 | 1.001 |
| | X6 | 0.99947 | 0.99734 | 1.001 | 0.99905 | 0.99837 | 1.001 |
| | X7 | 1.0007 | 0.99726 | 1.001 | 1.0003 | 0.99886 | 0.99867 |
| | X8 | 1.0025 | 0.99612 | 1.001 | 0.99955 | 0.99856 | 0.996 |
| | X9 | 1.0059 | 0.99124 | 1.001 | 1.0012 | 0.99768 | 0.99124 |
| | X10 | 1.0095 | 0.98228 | 1.0012 | 1.0033 | 0.99776 | 0.98247 |
| <i>f2</i> | N_géné | 50 | 45 | 35 | 45 | 47 | 38 |
| | F2_min | 0.083197 | 0.003352 | 1.0742e-05 | 0.00052703 | 0.0008142 | 0.0002831 |
| | X1 | 6.4109e-04 | -0.0041097 | 9.9597-09 | 8.8682e-06 | 7.8882e-05 | 9.41097-06 |
| | X2 | 7.3681e-04 | 8.3871e-05 | 9.9595e-09 | 8.9782-05 | 7.8882-05 | 8.2471e-07 |
| | X3 | 7.2471e-04 | 8.3871e-05 | 8.2694e-09 | 9.8546e-06 | 9.8546e-05 | 8.2471e-07 |
| | X4 | 7.2471e-04 | 8.3871e-05 | 8.269e-09 | 8.3991e-07 | 8.3991e-05 | 8.2471e-07 |
| | X5 | 7.2151e-04 | 8.3871e-05 | 9.2481e-09 | 8.8886e-07 | 8.8886e-05 | 8.2471e-07 |
| | X6 | 7.2597e-04 | 8.3871e-05 | 8.4897-09 | 7.41097-06 | 8.41097-06 | 8.9682e-06 |
| | X7 | 8.2389e-05 | 8.3871e-05 | 8.4899e-09 | 8.2779e-07 | 8.6779e-06 | 8.9682-06 |
| | X8 | 8.2389e-04 | 7.6246e-06 | 9.2874e-09 | 8.2474e-07 | 8.2974e-06 | 9.8246e-06 |
| | X9 | 8.2389e-04 | 8.3871e-05 | 9.2874e-08 | 8.2471e-06 | 8.2371e-06 | 8.3991e-07 |
| | X10 | 8.2471e-05 | 7.6246e-06 | 8.2471e-08 | 8.2471e-06 | 8.2481e-05 | 8.9846e-06 |

D'après les courbes obtenues, on remarque que pour les deux fonctions, la valeur minimale est atteinte en utilisant l'algorithme AGT et ce la en un temps minimal c'est à dire un nombre minimal de générations.

L'algorithme utilisant la transformation des gènes a donné de meilleurs résultats, et il est toujours suivi par l'AGPA. On remarque aussi que l'AGPS et l'AGPT donnent des résultats qui sont très proches. Ainsi que, l'utilisation de l'AGM améliore la convergence de l'algorithme et donne de bon résultat.

Pour mieux comparer les résultats obtenus dans cette épreuve, on a proposé de déterminer la valeur minimale et le nombre de générations nécessaires pour atteindre l'optimum de chaque fonction. Ces résultats sont illustrés dans le tableau 2-16 pour les six formes d'algorithmes proposés.

D'après l'étude statistique et les résultats du tableau 2-16, on peut confirmer que l'utilisation de la transformation et la transposition dans la stratégie de la recherche de niche, donne de meilleurs résultats par rapport à l'algorithme génétique simple. On voit aussi que le minimum des deux fonctions est atteint pendant les premières générations de l'AGT et l'AGPA. Pour les deux autres algorithmes, l'AGPT et l'AGPS, on remarque quelque ressemblance dans leur comportement. Dans tous les cas la performance de la transformation et la transposition est prédominante.

IV. Conclusion

Nous avons vu précédemment quelques méthodes qui permettent d'empêcher la convergence prématurée, par exemple des opérateurs d'exploration telle que la transformation des gènes et la transposition avec toutes ces formes. Le choix des paramètres employés dans la transformation et la transposition est un aspect important dans la mise en œuvre de l'algorithme. Certains paramètres comme la longueur du gène et le taux de transformation ainsi que la longueur de la séquence du segment de gènes pour la transposition, ont une grande influence sur la convergence vers l'optimum des fonctions.

Pour maintenir la diversité génétique à long terme les méthodes de nichage ont été introduites. La recherche de niche est caractérisée par une recherche évolutionnaire qui dirigés les individus et les niches en évoluant vers l'optimum global. Le but d'une telle méthode est s'augmenter la précision de la recherche de l'optimum global quand ce dernier est localisé dans des régions étroites.

I. Introduction

La biologie moderne a fait faire un saut extraordinaire dans la compréhension des mécanismes biologiques intervenant dans le monde du vivant. Cette connaissance ouvre actuellement un champ d'investigation très intéressant qui permet de mettre en œuvre les méthodes expérimentales et théoriques pour l'observation et la modélisation des objets biologiques, de l'échelle microscopique aux systèmes intégrés.

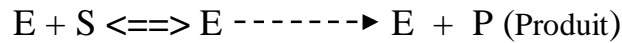
Pour tester l'efficacité des différents algorithmes génétiques proposés, on s'est intéressé à l'étude de deux systèmes dynamiques non linéaires; le premier représentant un système enzymatique à deux compartiments et un seul substrat, le deuxième représentant un réacteur d'écoulement de prise.

II. mécanismes de catalyse enzymatique

La compréhension des mécanismes catalytiques de systèmes enzymatiques originaux en relation avec leur fonction au sein des cellules représente un des défis majeurs de la biologie moderne.

Les organismes vivants sont le siège d'un grand nombre de réactions biochimiques très diverses. Ces réactions s'effectuent dans des conditions où, normalement, elles ne pourraient se faire. Si elles ont lieu, c'est parce qu'elles sont catalysées par des macromolécules biologiques; **les enzymes**. De plus les enzymes en tant que seuls catalyseurs des réactions chimiques qui se déroulent dans les organismes vivants, ils sont caractérisés par une très haute spécificité de reconnaissance des molécules sur lesquelles elles agissent [18, 19].

Au début du 20^e siècle, de gros travaux furent entrepris pour purifier des enzymes et surtout de décrire leur activité catalytique en termes mathématiques. En 1902 V. Henri et Adrian Brown suggérèrent indépendamment que la formation d'un complexe enzyme - substrat est un intermédiaire obligatoire de la réaction enzymatique, ils redécouvrirent l'équation mathématique reliant l'effet de la concentration du substrat à la vitesse de catalyse. Le point important est que l'obtention de cette équation repose sur l'hypothèse qu'il s'établit un équilibre rapide entre les concentrations de l'enzyme, du substrat et du complexe enzyme - substrat [20, 21]:



Le mécanisme de catalyse des réactions enzyme substrats se schématise comme suit:



La plupart des réactions enzymatiques impliquent 2 substrats, le deuxième substrat peut être, soit réaction d'hydrolyse ou un donneur ou un accepteur d'électrons. La réaction à 2 substrats (S1, S2) et 2 produits (P1, P2) s'exprime par la relation:



Les principaux mécanismes de catalyse enzymatique à 2 substrats sont:

- **Mécanismes de catalyse aléatoires:** les deux substrats, S1 et S2, se fixent de manière aléatoire sur l'enzyme libre E, c'est-à-dire qu'il n'y a pas de fixation privilégiée de l'un ou l'autre des deux substrats.
- **Mécanismes de catalyse ordonnés :** Dans un tel système, l'ordre de fixation est obligatoire; le substrat S1 doit se fixer à l'enzyme libre E avant le substrat S2. En d'autres termes, le substrat S2 ne peut se fixer qu'au complexe ES1.



III. Le complexe enzyme - substrat

La formation initiale d'un complexe enzyme - substrat (E-S), fût suggérée d'après les observations suivantes [22]:

- Le haut degré de spécificité de la reconnaissance d'un substrat par le site actif d'une enzyme. Cette reconnaissance résulte d'une très forte complémentarité des structures du substrat et de l'enzyme qui le fixe, comme le sont une clé et la serrure dans laquelle elle entre. par exemple, certains composés qui ressemblent chimiquement à un substrat mais qui ont des groupements moins volumineux ne sont pas catalysés, bien qu'ils doivent encore mieux s'insérer dans le site actif
- La forme de la courbe dite de saturation : vitesse initiale de la réaction enzymatique en fonction de la concentration en substrat ($V_i = f([S])$).
- Le fait que les substrats protègent souvent les enzymes de l'inactivation.

a / Les substrats

Un substrat et un produit d'une réaction enzymatique sont caractérisés par des liaisons chimiques. Au cours d'une réaction, des échanges d'énergie avec le milieu environnant ont lieu : certaines liaisons du substrat sont rompues en absorbant de l'énergie et les liaisons du produit sont formées en libérant de l'énergie. Lors de l'absorption de cette énergie, la vitesse des molécules de substrats augmente et donc la fréquence et le nombre de collisions entre molécules de substrats et d'enzyme augmentent aussi. Parallèlement, l'augmentation de l'agitation thermique fragilise les liaisons qui sont donc plus faciles à rompre. Seule l'intervention d'un catalyseur biologique, une enzyme, augmente la vitesse de la réaction en abaissant l'énergie d'activation, les substrats franchissent plus facilement la barrière d'activation.

La vitesse d'une réaction enzymatique dépend de l'efficacité de la rencontre entre les molécules qui vont former l'état de transition. Cette efficacité dépend à la fois[23] :

- De l'orientation des molécules l'une par rapport à l'autre
- De leur énergie : cette énergie minimale requise est l'énergie d'activation.

b/ Les membranes

En biologie cellulaire, une membrane est la structure qui sépare deux compartiments cellulaires et définit ainsi un intérieur et un extérieur. On distingue la membrane plasmique qui délimite l'intérieur de l'extérieur de la cellule et les membranes intracellulaires des organites. Les membranes sont constituées de phospholipides organisés en bicouche et de protéines membranaires "flottant" dans cette mosaïque fluide.

Elle possède beaucoup de fonctions, les membranes [23]:

- sont des barrières sélectivement perméables séparant les milieux internes et externes de la cellule et des organites et permettant des échanges contrôlés entre les deux compartiments (Transport membranaire).
- comportent sur leurs surfaces externes des récepteurs spécifiques à certaines molécules (signalisation hormonale)
- sont le sein d'activités biochimiques de par la présence d'enzymes

Fondamentalement, une membrane est une double couche de lipides, chaque couche ayant une face hydrophile orientée vers l'extérieur de la membrane et une face hydrophobe orientée vers l'intérieur de la membrane. Toutefois, les membranes comportent une part importante de glucides et de protéines. Leur composition exacte est très variable d'un type cellulaire à l'autre. De plus, cette composition n'est généralement pas symétrique [23].

VI. Optimisation d'un système enzymatique à deux compartiments et un seul substrat

Un système enzymatique à 2 compartiments peut être schématisé par la représentation suivante [24]:

| | | | | | |
|----|--|-------|-------|--|-------|
| S0 | | S1, E | S2, E | | S0+ m |
|----|--|-------|-------|--|-------|

Les réactions enzymatiques ont lieu dans chacun des compartiments grâce à la présence d'une enzyme **E**. Et les deux compartiments sont séparés par une membrane, à travers laquelle diffuse le substrat.

- S_1 et S_2 représentent respectivement, les concentrations du substrat à l'intérieur, des compartiments 1 et 2.
- S_0 c'est la concentration du substrat extérieur au compartiment 1.
- $S_0 + m$ c'est la concentration du substrat extérieur au compartiment 2. m est assimilé à une perturbation.

D'après Kernévez [24], ce procédé peut être modélisé par le système d'équations différentielles suivant:

$$\frac{ds_1}{dt} = (S_0 - S_1) + (S_2 - s_1) - rR(S_1)$$

$$\frac{ds_2}{dt} = (S_0 + m - S_2) + (S_1 - s_2) - rR(S_2)$$

Avec: $R(S) = \frac{S}{1 + S + S^2}$

L'action enzymatique dans ces compartiments est définie par [24]:

$$W = 100(R(S_1) + R(S_2))$$

Dans la première partie de cette étude, on s'intéresse à déterminer l'activité **W** maximale, pour le système à l'équilibre et différentes valeurs de m .

Le problème revient donc à résoudre:

$$MaxW = Max100(R(S_1) + R(S_2))$$

Pour déterminer l'activité maximale en utilisant les algorithmes génétiques, le problème revient à maximiser la fonction d'adaptation. Deux cas sont considérés:

- 1° Cas: on considère la concentration du substrat extérieur S_0 comme variable.
- 2° Cas: on considère la concentration du substrat extérieur S_0 et la perturbation m comme variables.

1° Cas: S_0 comme variable

Afin de connaître l'influence du paramètre m sur le comportement du système, les tests ont été menés en attribuant trois valeurs différentes au paramètre m . Les différentes valeurs de la perturbation m qui vont s'ajouter à la concentration du substrat extérieur S_0 , sont:

- Le premier cas; $m=1$.
- Le deuxième cas; $m=5$.
- Le troisième cas; $m=0$.

Le problème est ensuite résolu par chacun des algorithmes décrits précédemment. Les paramètres génétiques utilisés sont ceux indiqués dans le tableau 3-1. Les résultats obtenus pour chaque cas, sont représentés dans les figures 3-1, 3-2 et 3-3.

Tableau 3-1

| | |
|------------------------------------|---------|
| Longueur du chromosome | 32 |
| Taille de la population (N) | 100 |
| Nombre de variable (1) | s_0 |
| Intervalle (S_0) | [10 50] |
| Pmut | 0.03 |
| Pcroi | 0.9 |
| Longueur du segment de gènes (FSL) | 4 |
| Taux de transformation (T%) | 80% |
| Longueur du gène segment (PG) | 5 |
| Nombre de génération max. | 50 |

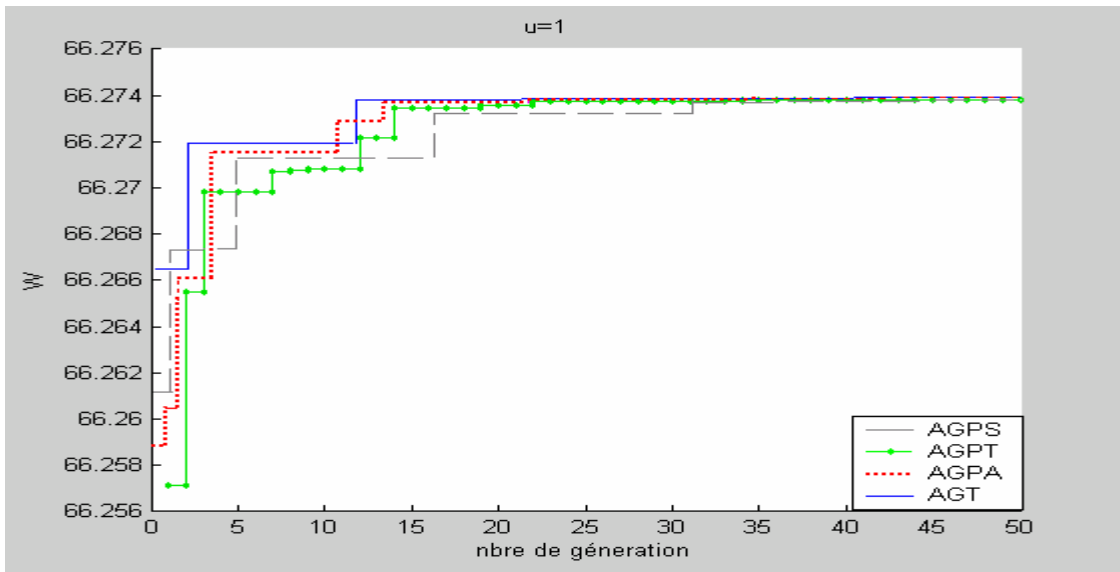


Figure 3-1: Evolution de l'activité W au cours des générations avec ($m=1$)

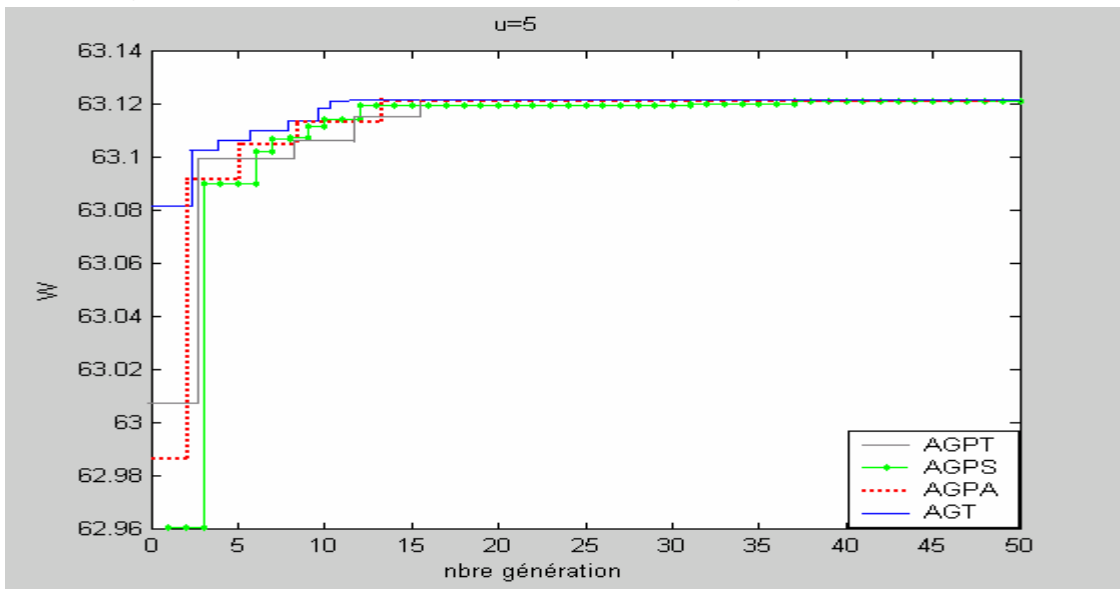


Figure 3-2: Evolution de l'activité W au cours des générations avec ($m=5$)

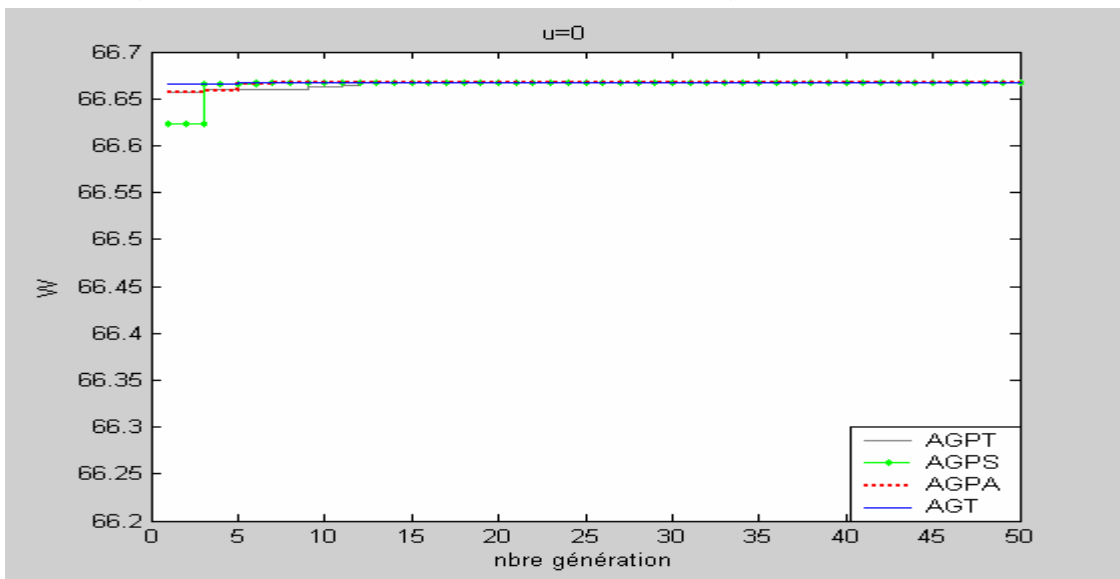


Figure 3-3: Evolution de l'activité W au cours des générations avec ($m=0$)

Tableau 3-2

| | | AGT | AGPS | AGPT | AGPA |
|---------|--------|---------|---------|---------|---------|
| $m = 1$ | N_géné | 12 | 39 | 36 | 22 |
| | W | 66.2741 | 66.2741 | 66.2740 | 66.2741 |
| | S0 | 33.580 | 33.580 | 33.581 | 33.580 |
| | S1 | 0.83366 | 0.83366 | 0.83566 | 0.83366 |
| | S2 | 1.0559 | 1.0559 | 1.0599 | 1.0559 |
| $m = 5$ | N_géné | 12 | 37 | 16 | 14 |
| | W | 63.1201 | 63.1201 | 63.1201 | 63.1201 |
| | S0 | 29.8832 | 29.8832 | 29.8832 | 29.8832 |
| | S1 | 0.5578 | 0.5578 | 0.5578 | 0.5578 |
| | S2 | 1.0691 | 1.0691 | 1.0691 | 1.0691 |
| $m = 0$ | N_géné | 4 | 6 | 13 | 5 |
| | W | 66.667 | 66.667 | 66.667 | 66.667 |
| | S0 | 34.36 | 34.344 | 34.344 | 34.361 |
| | S1 | 0.99989 | 0.9809 | 0.9809 | 1.001 |
| | S2 | 0.99989 | 0.9809 | 0.9809 | 1.001 |

Pour le cas $m=1$ et d'après l'analyse des résultats obtenus pour les différents algorithmes, il apparaît que pour un nombre de générations et d'exécutions minimales avec l'AGT, l'activité maximale W égale à 66.2741 avec une concentration extérieure S0 égale à 33.2741. La même valeur de l'activité est atteinte avec les différentes formes de la transposition, mais avec un nombre de générations et d'exécutions plus élevés.

Si on augmente la perturbation à 5, on remarque que dans un nombre de générations et d'exécutions très petit avec l'AGT l'activité maximale, est plus petite que dans le cas précédent; ce qui est normale puisque la perturbation est plus importante. Avec les autres formes d'algorithmes de transposition; la même valeur maximale de l'activité est atteinte dans un nombre de générations plus élevées. Si on annule la perturbation on remarque que l'activité maximale W égale à 66.667, est plus élevé que dans les deux cas précédent et elle est atteinte dans les premières générations .

Les résultats obtenus avec ces deux exemples mettent en évidence l'influence du paramètre m sur l'activité W .

Pour trouver la valeur de m pour laquelle on a la plus grande activité, on doit donc considéré ce paramètre comme inconnu dans le problème d'optimisation

2°Cas; (S0 et U) comme variables

Dans cette partie on s'est intéressés à l'influence du paramètre m sur l'activité W , pour cette raison on a considéré la perturbation comme un paramètre inconnu avec la concentration extérieure S_0 . A cet effet, et afin de trouver de meilleur solution, il a fallu faire plusieurs tests pour choisir et déterminer les paramètres génétiques qui vont être utilisés pour arriver a des résultats satisfaisants. Le tableau 3-3 indique les paramètres génétiques choisis dans les différents essais. Les résultats obtenus pour chaque cas, sont représentés dans le tableau 3-4, alors que les figures 3-4 et 3-5 représentent les courbes obtenus des deux cas.

Tableau 3-3

| | 1°Cas | 2°Cas |
|---|------------------------------|------------------------------|
| Longueur du chromosome | 40 | 32 |
| Taille de la population (N) | 100 | 50 |
| Nombre de variable (2) | s_0, m | s_0, m |
| Intervalles (s_0) et (m) | $S_0=[10 \ 50], m=[-1 \ 10]$ | $S_0=[10 \ 50], m=[-1 \ 10]$ |
| Longueur du segment de gènes (FSL) | 5 | 4 |
| Taux de transformation (T%) | 80% | 80% |
| Longueur du gène segment (PG) | 6 | 5 |

Tableau 3-4

| 1°Cas | AGT | AGPS | AGPT | AGPA |
|---------------|-----------|---------|---------|---------|
| N_géné | 50 | 50 | 50 | 50 |
| W | 66.609 | 64.23 | 63.223 | 63.983 |
| m | 0.38608 | 3.6901 | 5.232 | 4.0344 |
| S0 | 34.227 | 31.107 | 29.719 | 30.871 |
| S1 | 0.70444 | 0.62031 | 0.56404 | 0.60744 |
| S2 | 1.0665 | 1.0623 | 1.1429 | 1.0914 |
| 2°Cas | AGT | AGPS | AGPT | AGPA |
| N_géné | 28 | 50 | 50 | 50 |
| W | 66.666 | 60.804 | 62.515 | 65.817 |
| m | 0.414e-07 | 7.2332 | 5.7074 | 1.7162 |
| S0 | 34.333 | 27.482 | 29.211 | 33.005 |
| S1 | 0.99983 | 0.45969 | 0.51714 | 0.76472 |
| S2 | 0.99981 | 0.94205 | 0.79646 | 1.0789 |

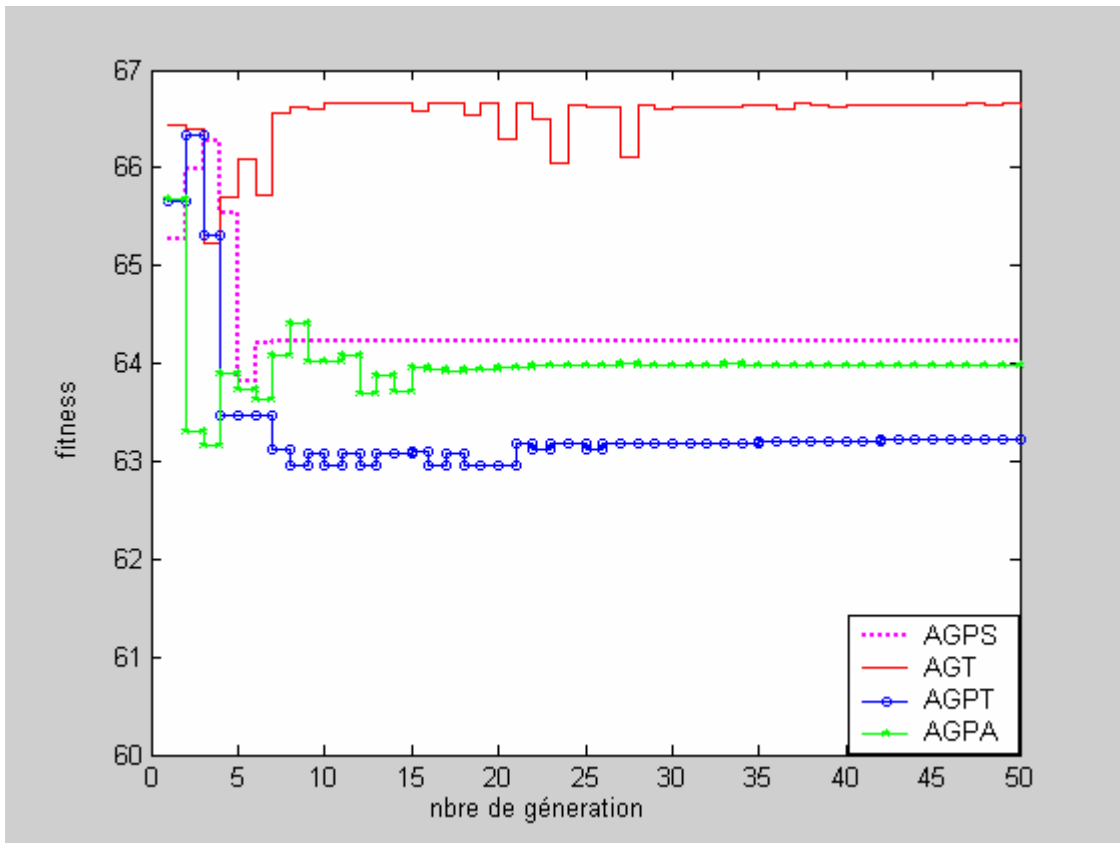


Figure 3-4: Evolution de l'activité W au cours des générations

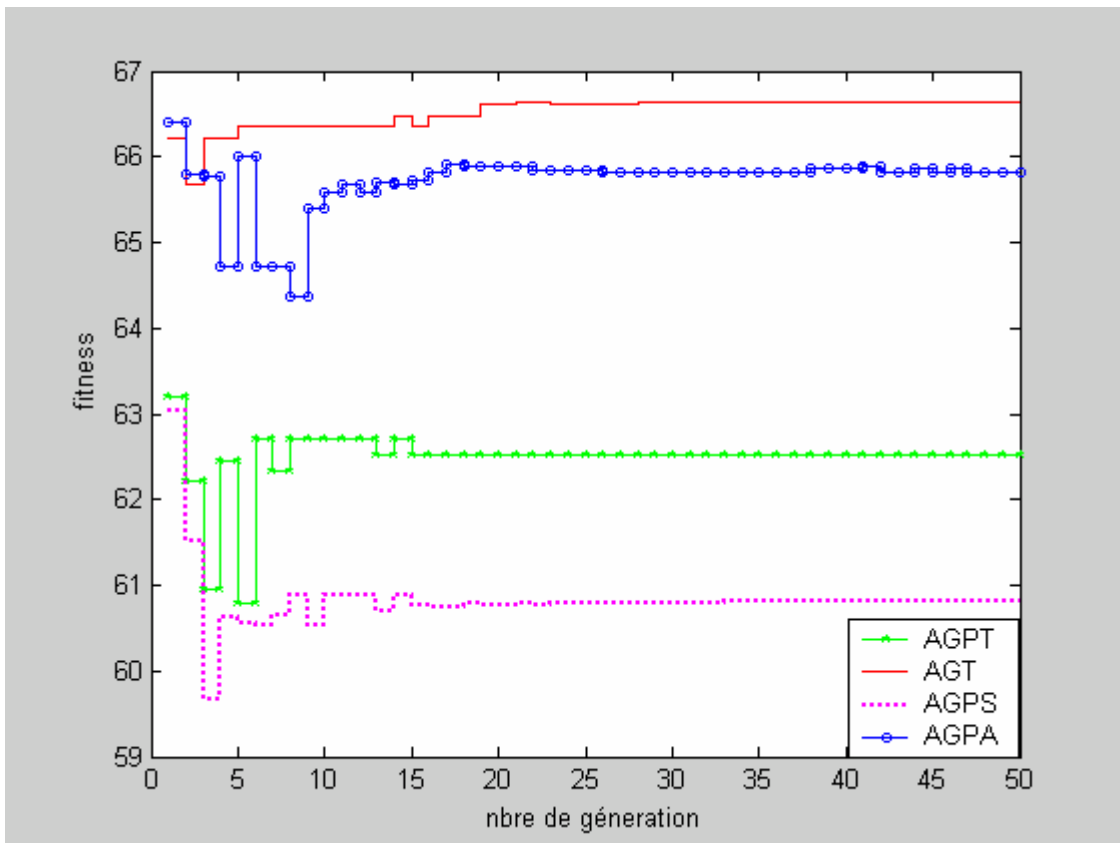


Figure 3-5: Evolution de l'activité W au cours des générations

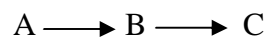
D'après les courbes, on remarque que dans les deux cas la valeur maximale de l'activité est atteinte avec l'AGT dans la première exécution et dans les premières générations. On ressort que l'activité maximale est $W= 66.667$, elle correspond à une solution symétrique $S1=S2= 1$ avec $S0=34.36$ et une perturbation nulle $m=0$.

La meilleure valeur est celle du deuxième cas dont on a utilisé un nombre de population plus petits, alors que pour les autres formes d'algorithmes de transposition, on remarque que les résultats obtenus sont similaires; dans les premières générations la valeur de l'activité est supérieure que celle trouvée dans les dernières générations, car la solution est stationnée dans un optimum local dont elle n'a pas pu converger vers l'optimum global, ce qui permet d'en déduire que la convergence est guidée par un certain nombre de paramètres fixés à l'avance. La valeur de ces paramètres influence la réussite ou non d'un algorithme génétique, parmi ces paramètres sont le nombre de variables et la taille de la population.

Si la taille de la population, est trop grande le temps de calcul de l'algorithme peut être très important, et si la taille est trop petite, il peut converger trop rapidement vers un mauvais optimum, car il est important de savoir que lorsque la diversité génétique devient très faible, il y a très peu de chances pour qu'elle augmente à nouveau. Et si cela se produit trop tôt, la convergence a lieu vers un optimum local

V. Optimisation d'un réacteur de d'écoulement de prise

Les réactions dans un réacteur tubulaire sont souvent représentées par:



Présentant le rapport d'Arrhenius, les concentrations X_1 et X_2 de A et de B respectivement sont données par les équations suivantes [25]:

$$\frac{dx_1}{dx_2} = -K_1 x_1 e^{-E_1/RT}$$

$$\frac{dx_2}{dt} = K_1 x_1 e^{-E_1/RT} - K_2 x_2 e^{-E_2/RT}$$

Le problème est de maximiser la concentration de B dans le jet de produit en manoeuvrant le profil de température $T(x)$. Le temps de séjour est maintenu à huit minutes, et les autres paramètres sont : $E_1=75$ kJ/mol, $E_2=125$ kJ/mol et $R=8.31$ J/mol.k.

Les valeurs des facteurs pré exponentiels K_1 et K_2 ont été arbitrairement choisies [25]:

$$K_1=1.0 \cdot 10^8 \text{ min}^{-1}$$

$$K_2= 4.0 \cdot 10^{15} \text{ min}^{-1}$$

Pour résoudre ce problème et déterminer la concentration maximale X_2 de B, et dont le but est de tester et confirmer l'efficacité des différents algorithmes génétiques décrits précédemment, Deux cas sont considérés:

- 1° Cas: on prend en considération la température T comme variable pour déterminer la concentration maximale X_2 .
- 2° Cas: on considère la température T et les deux facteurs K_1 et K_2 comme variables pour déterminer la concentration maximale X_2 .

1° Cas: T comme variable

Dans notre étude, on a utilisé les différents algorithmes génétiques pour déterminer la concentration maximale X_2 de B, Les paramètres génétiques utilisés pour ce cas sont indiqués dans le tableau 3-5. Les résultats, sont représentés dans le tableau 3-6, alors que la figure 3-6 représente les courbes obtenues.

Tableau 3-5

| | |
|---|----------|
| Longueur du chromosome | 32 |
| Taille de la population (N) | 100 |
| Nombre de variable (I) | T |
| Intervalle (T) | [10 200] |
| Longueur de la séquence du gène segment (FSL) | 5 |
| Taux de transformation (T%) | 80% |
| Longueur du gène segment (PG) | 5 |
| Nombre de génération max. | 50 |

Tableau 3-6

| | AGT | AGPS | AGPT | AGPA |
|---------------|---------------------|---------------------|---------------------|---------------------|
| N_géné | 2 | 9 | 10 | 4 |
| X2 | 0.17065 | 0.17065 | 0.17065 | 0.17065 |
| T | 83.868 | 83.868 | 83.868 | 83.868 |
| K1 | $1.0 \cdot 10^8$ | $1.0 \cdot 10^8$ | $1.0 \cdot 10^8$ | $1.0 \cdot 10^8$ |
| K2 | $4.0 \cdot 10^{15}$ | $4.0 \cdot 10^{15}$ | $4.0 \cdot 10^{15}$ | $4.0 \cdot 10^{15}$ |

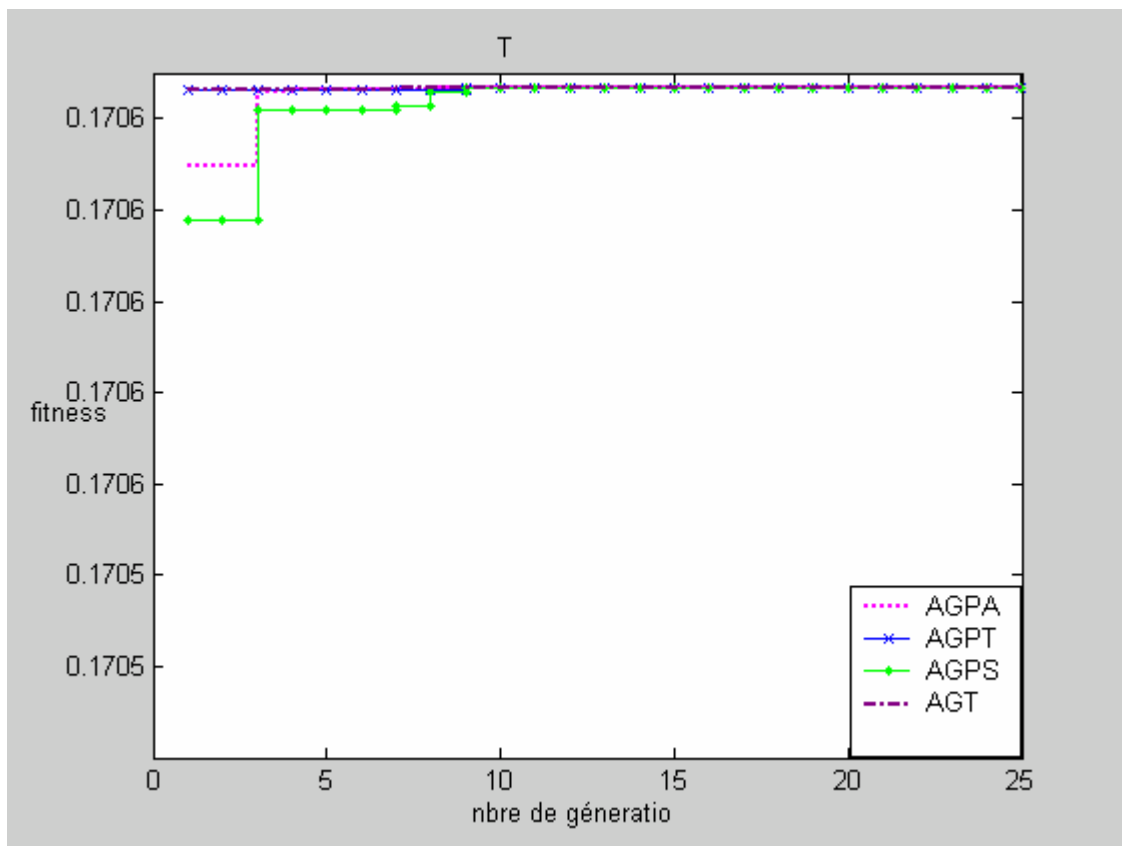


Figure 3-6: Evolution de la concentration X2 au cours des générations

Les résultats du tableau 3-6 montrent que la concentration maximale X2 égale à 0.17065 dans une température égale à 83.868, cette valeur est atteinte avec les différentes formes d'algorithmes, on remarque que dans un nombre de générations minimales la valeur maximale est atteinte avec l'AGT qui est suivis par l'AGPA. La même valeur de la concentration est atteinte avec les autres formes de transposition; l'AGPT et l'AGPS mais avec un nombre de générations plus élevées.

2°Cas; (T, K1 et K2) comme variables

Pour confirmer l'efficacité des différentes formes d'algorithmes on a augmenté le nombre de variable pour trouver la valeur maximale de la concentration et la température correspondante, on a donc considéré trois paramètres comme inconnus; la température T et les deux facteurs K1 et K2.

Les paramètres génétiques utilisés pour ce deuxième cas, sont indiqués dans le tableau 3-7. Les résultats obtenus, sont représentés dans le tableau 3-8, alors que la figure 3-7 représente les courbes obtenues.

Tableau 3-7

| | |
|--|--|
| Longueur du chromosome | 60 |
| Taille de la population (N) | 100 |
| Nombre de variable (3) | T, K1, K2 |
| Intervalle (T) | [10 200] |
| Intervalle (K1) | [10 ⁷ 10 ⁹] |
| Intervalle (K2) | [4*10 ¹⁴ 4*10 ¹⁶] |
| Longueur de la séquence du gène segment (FSL) | 6 |
| Taux de transformation (T%) | 80% |
| Longueur du gène segment (PG) | 6 |
| Nombre de génération max. | 100 |

Tableau 3-8

| | AGT | AGPS | AGPT | AGPA |
|---------------|-------------|-------------|-------------|-------------|
| N_géné | 24 | 48 | 40 | 26 |
| X2 | 0.6538 | 0.65144 | 0.65265 | 0.65378 |
| T | 77.89 | 79.473 | 80.044 | 80.044 |
| K1 | 8.2279e+08 | 9.9782e+08 | 9.913e+08 | 9.9952e+08 |
| K2 | 8.6418e+014 | 4.2901e+014 | 4.0967e+014 | 4e+014 |

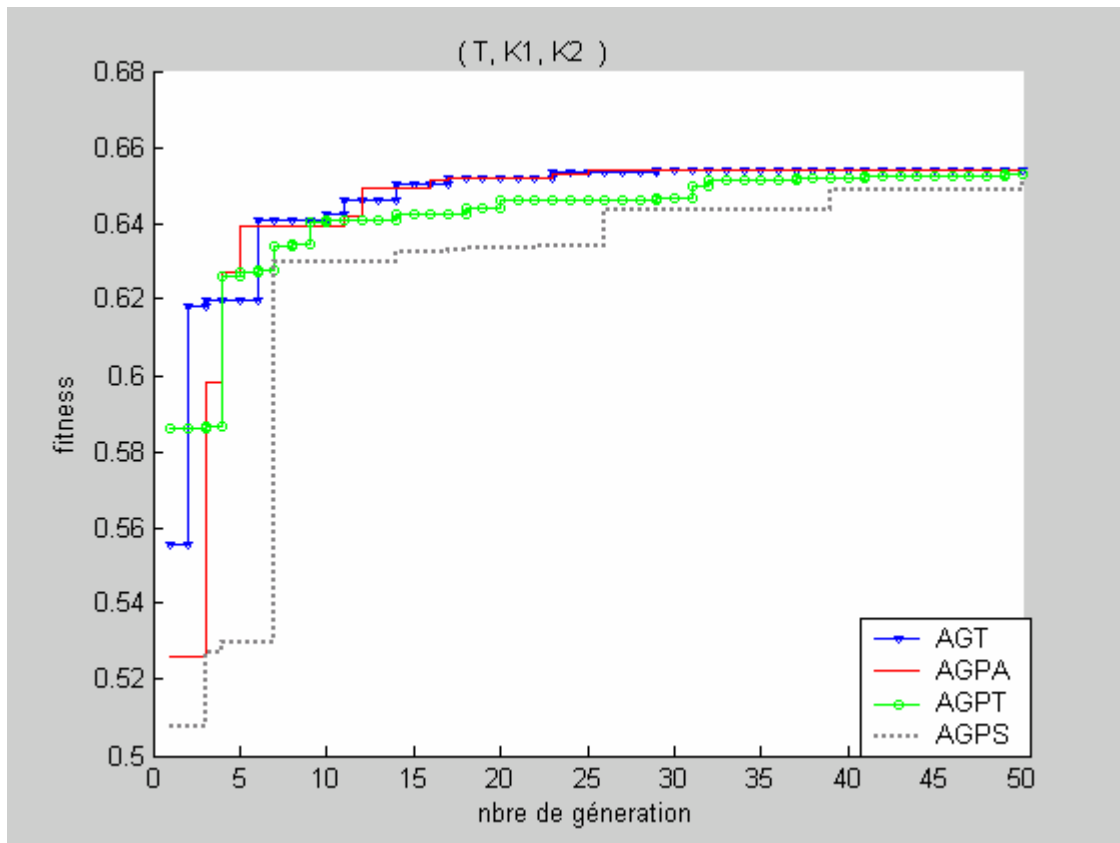


Figure 3-7: Evolution de la concentration X2 au cours des générations

D'après les courbes de la figure 3-7 et le tableau 3-8, on remarque que la valeur maximale de la concentration X2 est égale à 0.6538 dans une température égale à 77.89 avec $K1=8.2279e+08$ et $K2= 8.6418e+014$, cette valeur est atteinte avec l'AGT dans la première exécution mais avec un nombre de générations plus élevées que dans le premier cas. L'AGT est toujours suivi par l'AGPA, ainsi que l'AGPT et l'AGPS, on a toujours le même comportement, ils tiennent plus de générations pour atteindre l'optimum.

Les résultats obtenus avec ces deux cas mettent en évidence l'influence de certains paramètres sur la réussite de l'algorithme. On ressort que le nombre de variables utilisées influe sur l'efficacité de l'algorithme.

IV. Conclusion

On examinant tous les résultats obtenus, on peut conclure que l'utilisation de l'algorithme de la transformation de gènes a prouvé son efficacité, à condition de faire un choix adéquat du problème et cette opération nécessite en générale, plusieurs tests au préalable. Les nouveaux opérateurs génétiques tel que la transformation et la transposition préservent la diversité dans la population pendant toutes les générations et l'algorithme évolutionnaire avec la transformation et la transposition en général donne de meilleures solutions que l'algorithme génétique standard.

A partir de ces résultats on peut déduire que la transformation permet une grande diversité dans la population et évite la convergence prématuré ce qui permet une évolution sur une période plus longue. Afin d'augmenter l'exécution de la transformation et réaliser de meilleures solutions, l'étude à montré que le choix des paramètres tels que: la taille de la population, le nombre de variable et de génération, ont une grande influence sur la convergence de l'algorithme.

Pour conclure, nous pouvons dire que l'utilisation des algorithmes génétiques pour les problèmes d'optimisation donne de bons résultats et présentent donc, une nouvelle stratégie efficace pour l'optimisation des systèmes dynamiques non linéaires, mais elle demande un bon choix des paramètres génétiques.

CONCLUSION GENERALE

La résolution d'un problème d'optimisation consiste à explorer un espace de recherche afin de maximiser ou minimiser une fonction donnée. Les algorithmes génétiques sont des procédures assez robustes pour résoudre un problème d'optimisation. Néanmoins elles présentent certaines limites et difficultés qui influent fortement sur la convergence de l'algorithme. Ces difficultés reposent sur le choix des bons paramètres tels que : la taille de la population, le nombre de génération, les probabilités de croisement et de mutation et les méthodes des opérateurs de reproduction. Ces paramètres dépendent du problème à résoudre et nécessitent quelque fois plusieurs essais préalables pour les fixer.

Les complexités relatives de l'espace de recherche et de la fonction à maximiser conduisent à utiliser des algorithmes de résolutions assez forts. L'usage d'un algorithme génétique utilisant de nouveaux mécanismes biologiques tel que la transformation, la transposition génétique, et la stratégie de la recherche de niche est adapté à une exploration rapide et globale d'un espace de recherche de taille importante. Ces algorithmes génétiques peuvent constituer une alternative intéressante lorsque les algorithmes génétiques simples ne parviennent pas à fournir efficacement des résultats fiables.

Les techniques proposées ont été testées sur des fonctions multivariées dans le but de voir l'efficacité de chaque opérateur utilisé dans l'algorithme, en suite on les a appliqués sur des systèmes dynamiques. Leurs performances ont été comparées avec celles d'un algorithme génétique simple. L'application de ces algorithmes génétiques sur les différentes fonctions, a donné des améliorations sur la convergence, surtout la transformation génétique qui nous a donné une grande performance sur la convergence et la diversité dans la population.

Pour augmenter les performances de la transformation et obtenir de meilleures solutions, un travail préalable sur l'étude paramétrique, a été fait. Ce qui nous a permis de conclure quant à leur influence sur la convergence de l'algorithme.

BIBLIOGRAPHIE

- [1]: **A.Boumaza**, "Introduction de techniques d'évolution artificielle en vision tridimensionnelle et en robotique mobile", Thèse doctorat, Université Rene Descartes– Paris 2004.
- [2]: **R. Dupas**, "Amélioration de performance des systèmes de production:apport des algorithmes évolutionnistes aux problèmes d'ordonnancement cycliques et flexibles", Université d'Artois 2004.
- [3]: **S. Bernard**, "Algorithmes Evolutionnaires", Edition addison-Wesley, Paris 2003.
- [4]: **E. Lutton**, "Darwinisme artificiel ", INRIA - Rocquencourt - Equipe Complex - Projet Fractales, 2004.
- [5]: **T.Vallée et M.Yildizoglu**. "Présentation des algorithmes génétiques et de leurs applications en économie", Université de Nantes, 2001.
- [6]: **S.Baudot-Roux**. "Algorithmes évolutionnaires, hybridation" cf. PARCFD'99, Surveys on Mathematics 2000, Eurogen 97, 98, 99.
- [7]: **P. Lucidarme**. "Apprentissage et adaptation pour des ensembles de robots réactifs coopérants". Proc. ICRA'02, Washington, 2002.
- [8]: **A. Nabonne**. "Algorithmes évolutionnaires et problèmes inverses", chapitre 8, juin 2004.
- [9]: **D.Francisci** "Algorithmes Evolutionnaires et Optimisation Multi-objectifs en Data Mining" projet MECOSI, rapport de recherche I3S/RR-2002.
- [10]: **D.Beasley et R. Martin**, "An Overview of Genetic Algorithms":Part 2, department of Computing Mathematics, University of Wales College of Cardiff, CF2 4YN, 1993.
- [11]: **J.PhilippeRennard**, "Genetic Algorithm Viewer: Démonstration d'un algorithme génétique", www.rennad.org/alif, Avril 2000.
- [12]: **N.Benahmed**, "Optimisation de réseaux de neurones pour la reconnaissance de chiffres manuscrits isolés: sélection et pondération des primitives par algorithme génétique". Université du Québec ,2002.
- [13]: **T.Vallé et M.Yildizoğlu** "Présentation des algorithmes génétiques et de leurs applications en économie", Université de Nantes et Université Montesquieu Bordeaux IV 2001.
- [14]:**A.Simoes and E.Costa**: "Parametric Study to Enhance Genetic Algorithm's, Performance When Using Transformation "**GECCO'02**, 9-13Juillet 2002, New York. pp 196-202
- [15]: **A.Simoes and E.Costa**. "Transposition: A Biologically Inspired Mechanism to Use with Genetic Algorithms"(ICANNGA 99), Portoroz, Slovenia, 6-9 April, 1999, pp. 178-186.

- [16]: **A. Simoes and E.Costa.** "Using Genetic Algorithms with Sexual or Asexual Transposition" (CEC 2000), San Diego, USA, July 2000.
- [17]: **J. Pedro Pedroso.** "Niche Search : an Evolutionary Algorithm for Global Optimisation", Niche-ppsn.ps.gz. Vol 1141, Germany 1995.
- [18]: **D. Diderot,** "Physique et systèmes biologiques", UFR de Physique - Université Paris 7 p7p11physbio.in2p3.fr 2005.
- [19]: **V. Nancy,** "Simulation informatique des systèmes enzymatiques : de la structure à la fonction", Projet de Recherche, cbt-nancy.fr. Université Henri Poincaré , No. 7565 2004.
- [20]: **J. Michel Fayard,** "Biologie des Systèmes et Modélisation Cellulaire" INSA de Lyon, Université Claude Bernard Lyon1, Janvier 2003.
- [21]: **J. Ricard,** "Complexité, émergence, information et causalité dans les systèmes biologiques" Crackling noise. Nature 410, 241-250, 2001.
- [22]: **J. Jaspard** "Les réactions enzymatiques à 2 substrats : hypothèse du quasi-équilibre", univ-angers.fr/Page2/COURS/4EnzymologieLicence/4DeuxSUB..
- [23]: **M.F. Ruiz-López , J.L. Rivail** "Combined Quantum Mechanics and Molecular Mechanics Approaches to Chemical and Biochemical Reactivity", Vol. 1, pp. 437-448, Wiley & sons, 1998.
- [24]: **N. Mansouri,** "Identification des paramètres dans les problèmes non linéaires". Thèse de doctorat, Université Constantine, 1999.
- [25]: **Q.T. Pham,** "Dynamic optimization of chemical engineering processes by evolutionary method", school of Chemical Engineering and Industrial Chemistry, University of New South Wales, Sydney 2052, Australia. 2003.

ملخص

إن هذا العمل يمثل دراسة حول الخوارزميات الجينية وقد استعملنا تقنيات جديدة مستوحاة من ظواهر بيولوجية منها التحول والانعكاس الجيني وإستراتيجية البحث عن الأعشاش
إن الهدف من هذا العمل هو القيام بدراسة ومقارنة مختلف هذه الخوارزميات ولقد تطرقنا بصفة خاصة إلى تطبيق هذه الخوارزميات في الأنظمة الديناميكية للوصول إلى النتائج الجيدة
إن استعمال الخوارزميات الجينية البسيطة أدى إلى الحصول على نتائج غير جيدة وللحصول على نتائج مرضية استوجب استعمال تقنيات جديدة ولقد طبقت على دوال متعددة المتغيرات وعلى أنظمة ديناميكية مما أدى إلى تحسين في التقارب وكان مستوى النتائج جيد خاصة عند استعمال التحول الجيني

RESUME

Dans le cadre de ce mémoire nous nous sommes intéressés à l'étude des algorithmes génétiques utilisant de nouvelles techniques inspirées de processus biologiques, en particulier la transformation, la transposition génétique et la stratégie de la recherche de niche. Le but de notre travail est de faire une étude comparative de leurs performances. Nous nous sommes intéressés en particulier, à la mise en œuvre de ces algorithmes dans les problèmes d'optimisation des systèmes dynamiques.

L'utilisation d'un algorithme génétique simple nous a permis de constater que les résultats obtenus par ce dernier dans certain cas sont très médiocres. Pour améliorer ces résultats, l'utilisation d'un algorithme génétique simple avec une bonne technique s'avère être la meilleure solution.

Les techniques proposées ont été testées sur des fonctions multivariées dans le but de voir l'efficacité de chaque opérateur utilisé dans l'algorithme. Leurs performances ont été comparées avec celles d'un algorithme génétique simple.

L'application de ces algorithmes sur les différentes fonctions et sur les systèmes dynamiques, a donné des améliorations sur la convergence et les taux de réussites sont plus élevés, surtout la transformation génétique.

ABSTRACT

In this work, we are interested in the study of the genetic algorithms using of new techniques inspired of biological processes, in particular the transformation, the genetic transposition and the strategy of the niche search. The goal of our work is to make a comparative study their performances. Us summons itself in particular interested, with the implementation of these algorithms in the problems of optimization of the dynamic systems.

The use of a simple genetic algorithms enabled us to note that the obtained by this last in certain case are very poor. To improve these results, the use of a simple genetic algorithm with a good technique proves to be the best solution.

The techniques suggested are tested on multivariable functions with an aim of seeing the effectiveness of each operator used in the algorithm. Their performances are compared with those of a simple genetic algorithm.

The application of these algorithms on the various functions and the dynamic systems, gave improvements on convergence and the rates of successes are raised, especially the genetic transformation.