

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Saad Dahlab, Blida
USDB



Faculté des sciences
Département d'informatique

Mémoire de fin d'études pour l'obtention
du diplôme de Master en informatique

Option : Ingénierie de logiciel

Thème

Implémentation des algorithmes génétiques multiobjectifs :

Application pour le problème de sélection des entreprises pour
l'octroi de marchés.

Réalisé par :

M^{lle} AMEUR Khadija

M^{lle} CHENTIR Afef

Encadré par :

M^{me} H.ABED

MA-004-10-1

Membre du jury

M^{me} Benetiti

M^{re} Ferfena

M^{me} Benatouba

Présidente

Examinatrice

Examinatrice

Promotion : 2009-2010

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Saad Dahlab, Blida
USDB

Faculté des sciences
Département d'informatique

Mémoire de fin d'études pour l'obtention
du diplôme de Master en informatique

Option : Ingénierie de logiciel

Thème

Implémentation des algorithmes génétiques multiobjectifs :

Application pour le problème de sélection des entreprises pour
l'octroi de marchés.

Réalisé par :

M^{lle} AMEUR Khadidja

M^{lle} CHENTIR Afef

Encadré par :

M^{me} H.ABED

Promotion : 2009-2010

Dédicaces

A ma chère maman,

A mon cher père,

A ma sœur Fatima,

A mes frères Taki dine , Hamza ,Khaled et Mohamed,

A mon grand père Hajd Khaled,

A tout ma grande famille,

A mon binôme Afef pour son travail, sa patience ainsi qu'à sa famille,

A tout mes amis d'études,

A mes adorables amis Meriem, Lamia, Malika, a toutes les files,

Enfin, que toutes les personnes qui ont participé de près ou de loin à la réalisation de ce travail trouvent ici l'expression de mes vifs sentiments de gratitude.

A toutes ces personnes cette thèse leur est dédiée à titre de modeste remerciement.

Khadija

Dédicaces

A ma très chère mère,

A mon superbe père,

A ma très chère prof Saida,

A mes sœurs Nabila et Kawthar,

A mon frère Hamza,

A mon binôme Khadidja,

A ma chère tante Amina,

A toute personne qui aime Afef et que Afef aime,

Je dédie ce modeste travail.

Afef.

Remerciements

Toute louange est à ALLAH l'unique et le seul... qui nous a guidé vers le chemin du savoir ...donné le pouvoir et la patience à chaque fois qu'il le fallait...

Nous tenons à remercier notre promotrice Mme ABED de nous confier ce merveilleux travail. Nous la remercions énormément pour son support, son intérêt et son savoir-faire

Nous dédions un merci très spécial à Mme BENATCHBA dont les conseils précieux et la documentation nous ont guidé tout au long de ce beau voyage. Qu'elle trouve ici l'expression de notre reconnaissance.

Nous remercions également Mr ZAHAF, Mr DERBALA, Mr BOUDJIT et Mr FERDJANI pour leurs accueils, leurs simplicités et pour la documentation qu'ils ont mis à notre disponibilité.

Que les membres du jury trouvent ici le témoignage de notre reconnaissance pour avoir bien accepté d'évaluer notre travail.

Nous remercions enfin toute personne qui nous a aidé, ou avait l'intention de le faire.

ملخص

عرفت الخوارزميات الجينية انتشارا واسعا منذ بداية استعمالها من طرف غولدبرغ. نظرا لقوتها وسهولة التعامل معها. الدراسة الحالية تسلط الضوء على الخوارزميات الجينية المتعددة الأهداف والتي تشكل امتدادا للخوارزميات الجينية الكلاسيكية في ميدان العضلات المتعددة الأهداف. للتكيف مع هذا النوع من العضلات، تستعمل الخوارزميات الجينية مفاهيم جديدة مثل: الهيمنة ومبدأ باريتو، وكذا تقنيات جديدة مثل التقسيم والتجمع. الهدف من هذه الرسالة هو التعرف على ملامح الخوارزميات الجينية وبشكلٍ أخصّ المتعددة الأهداف منها. سيُجسّد هذا الهدف من خلال طرح خوارزمية جينية متعددة الأهداف موجهة لحلّ مشكلة معاملات المعايير. هذه الأخيرة هي إحدى المشاكل المكوّنة لمشكلة اختيار الشركات لعقد صفقات إنجاز المشاريع. تجسيد العمل الحالي هو برنامج لدعم اتخاذ القرار على مستوى مكاتب الدراسات.

الكلمات الدلالية: الخوارزميات الجينية، الخوارزميات الجينية المتعددة الأهداف، العضلات المتعددة الأهداف، مبدأ باريتو، موازنة المعايير، اختيار الشركات.

Résumé

Depuis leur introduction par Goldberg, les Algorithmes Génétiques (AGs) ont suscité d'une très grande réputation vue leur robustesse et facilité d'emploi. Le travail actuel se focalise sur les Algorithmes Génétiques Multiobjectifs (AGMs) qui ne sont autres que la version multiobjectif des AGs classiques. Afin de passer à l'échelle des problèmes multiobjectifs, ce type d'AGs introduit de nouveaux concepts tels que la notion de dominance et l'optimum de Pareto. Les AGMs utilisent également de nouvelles techniques telles que le niching et le sharing.

Le but de ce travail est d'étudier les AGs et plus particulièrement les AGMs. Ceci est concrétisé par l'élaboration d'un AGM applicable à un problème multiobjectifs très célèbre : la pondération de critères. Ce dernier est un sous-problème du processus de sélection des entreprises. Notre étude est dans le contexte des appels d'offres.

La finalité de cette étude est un logiciel d'aide à la décision exploitable au niveau des bureaux d'études.

Mots clés : *Algorithmes Génétiques, Algorithmes Génétiques Multiobjectifs, Problèmes multiobjectifs, Optimum de Pareto, Pondération de critères, sélection des entreprises.*

Abstract

Since their introduction by Goldberg, the Genetic Algorithm (GA) aroused of a very big reputation seen their robustness and easier use. The current work focuses on the Multiobjective Genetic Algorithms (MGA) which are the multiobjective version of the classic AGs. To pass on the scale of the multiobjective problems, the GA introduces new concepts such as the notion of dominance and the Pareto optimum. They also use new techniques such as the niching and the sharing.

The purpose of this work is to study the GA and more particularly the MGA. This is concretized by the elaboration of an MGA and to apply it then to a very famous multiobjective problem: the level-headedness of criteria. This last one is a sub-problem of the problem of companies' selection for the granting of markets.

The purpose of this study is decision help software exploitable at the level of engineering consulting firms.

Keywords: *Genetic Algorithm, Multiobjective Genetic Algorithms, Multiobjectiv problem, Pareto's optimum, levelheadedness of criteria, companies' selection.*

<u>Introduction Générale</u>	1
<u>CHAPITRE 1 : Les Algorithmes Génétiques Multiobjectifs</u>	4
<u>Introduction</u> :.....	5
1. Généralités et mise en œuvre d'un algorithme génétique classique :	5
2. Les paramètres des algorithmes génétiques	16
3. Schéma général d'un algorithme génétique :	17
4. Les particularités des algorithmes génétiques :.....	18
5. Les Algorithmes Génétiques Multiobjectifs:	19
6. Particularités et difficultés des Algorithmes Génétiques Multiobjectifs :.....	28
<u>Conclusion</u>	33
<u>CHAPITRE 2 : Sélection des fournisseurs dans le contexte d'appels d'offres : Critères, méthodes et risques opérationnels</u>	34
<u>Introduction</u> :.....	35
1. <u>Le problème de sélection de fournisseurs</u> :.....	36
2. <u>Processus de sélection de fournisseurs dans le contexte d'un appel d'offre</u> :.....	37
2.1 <u>Étapes du processus d'appel d'offres</u>	37
2.2 <u>Identification des problématiques spécifiques au processus d'appel d'offres</u> :.....	40
2.3 <u>La problématique de pondération des critères</u> :.....	42
3. <u>Vue décisionnelle du processus de sélection de fournisseurs</u> :.....	44
4. <u>Critères de sélection des fournisseurs</u> :.....	46
5. <u>Méthodes d'évaluation des fournisseurs</u>	47
<u>Conclusion</u>	51
<u>CHAPITRE 3 : Conception : Algorithme Génétique Multiobjectif pour le problème de Sélection des Entreprises</u>	52
<u>Introduction</u> :.....	53
1 <u>Formulation multiobjectif du problème de sélection d'entreprise</u> :.....	54
1.1 <u>Éléments de la formulation proposée</u> :.....	54
1.2 <u>Formulation du problème d'optimisation multiobjectif</u> :	57
2 <u>Application des algorithmes génétiques multiobjectifs pour le problème de sélection d'entreprises pour l'octroi de marchés</u> :	57
2.1 <u>Principe de l'algorithme AGMSE</u> :.....	58

2.2	<u>Eléments de l’algorithme AGMSE:</u>	60
	<u>Conclusion</u>	71
	<u>CHAPITRE 4 : Implémentation et tests expérimentaux</u>	72
	<u>Introduction</u>	73
1.	<u>Présentation du logiciel :</u>	73
1.1	<u>L’environnement MATLAB</u>	73
1.2	<u>Description du logiciel</u>	75
2.	<u>Résultats expérimentaux</u>	78
2.1	<u>Les données du problème de tests :</u>	79
2.2	<u>Les tests :</u>	80
	<u>Conclusion</u>	88
	<u>Conclusion et perspectives</u>	89
	<u>Bibliographie</u>	91

LISTE DES FIGURES

Fig.1-1 :	Organigramme général d'un algorithme génétique.....	18
Fig.1-2 :	Exemple de front de Pareto	21
Fig.1-3 :	Principe de l'évaluation de performance des individus dans les AGMs.....	21
Fig.1-4 :	Sélection parallèle dans l'algorithme VEGA.....	23
Fig.1-5 :	Défauts de l'attribution de la performance dans les algorithmes non- Pareto	24
Fig.1-6 :	Tri de la population selon le principe de dominance (Ranking de Pareto)...	24
Fig.1-7 :	Mesure de surpeuplement.....	25
Fig.1-8 :	Principe de NSGA.....	26
Fig.1-9 :	Une fonction multimodale.....	30
Fig.1-10 :	Problèmes treppeurs.....	31
Fig. 2-1 :	Représentation du processus d'aide à la décision.....	45
Fig. 2-2 :	Positionnement approximatif des méthodes de sélection de fournisseurs....	50
Fig. 3-1:	Principe générale de l'AGMSE.....	58
Fig. 3-2:	Pseudo algorithme de l'AGMSE.....	59
Fig. 3-3:	Organigramme général de l'AGMSE.....	59
Fig. 3-4:	Algorithme de Normalisation.....	64
Fig.4-1 :	Le menu de choix principal.....	75
Fig.4-2 :	interface ingénieur du bureau d'étude	76
Fig.4-3 :	Formulaire de données de sélection.....	76
Fig.4-4 :	interface informaticien.....	77
Fig.4-5 :	Menu réglages de l'algorithme génétique.....	77
Fig.4-6 :	Menu détails des résultats.....	78
Fig.4-7 :	l'influence du nombre d'individus sur la qualité de solution.....	82
Fig.4-8:	L'influence du nombre de générations sur la qualité de solution.....	84
Fig.4-9:	L'influence du taux de mutation sur la qualité de solution.....	86
Fig.4-10:	L'influence du taux de croisement sur la qualité de solution.....	87

LISTE DES TABLEAUX

Tab.1-1 : Les AGMs les plus célèbres	28
Tab.2-1 : Synthèse des étapes du processus d'AO.....	38
Tab.2-2 : Synthèse des étapes du processus d'AO	40
Tab.2-3 : Critères de sélection des fournisseurs et leurs poids selon Dickson et Weber.....	46
Tab.2-4 : Critères de sélection des fournisseurs et leurs poids selon Dickson et Weber.....	51
Tab.4-1 : Les intervalles de poids pour chaque critère.....	79
Tab.4-2 : Les détails des offres du problème de teste.....	80
Tab.4-3 : Les valeurs des Epsilons pour chaque critère.....	80
Tab.4-4 : L'influence du nombre d'individus sur la qualité de solution.....	81
Tab.4-5 : Détails des résultats des évaluations suivant le nombre d'individus.....	83
Tab.4-6: L'influence du nombre de générations sur la qualité de solution.....	84
Tab.4-7: L'influence du taux de mutation sur la qualité de solution.....	85
Tab.4-8: L'influence du taux de croisement sur la qualité de solution.....	87

Introduction générale

L'optimisation multiobjectif est à l'heure actuelle, sans aucun doute, un domaine de recherche en plein essor tant du point de vue de la recherche théorique que des applications. En effet, la grande majorité des problèmes issus du monde réel sont de nature multiobjectif. Il est rare en pratique de pouvoir effectuer un choix où une solution est parfaite de tous points de vue. On parle d'optimisation multiobjectif dans les cas complexes où l'on doit optimiser simultanément plusieurs objectifs, souvent contradictoires. Ceci amène à choisir une solution de compromis parmi une multitude de solutions possibles. Il peut s'agir de minimiser un coût de production, d'optimiser le parcours d'un véhicule, d'améliorer les performances d'un circuit électronique, d'affiner un modèle de calcul, de fournir une aide à la décision à des managers,...etc.

Parmi les méthodes de résolution les plus célèbres dans le domaine de l'optimisation multiobjectif, figurent les Algorithmes Génétiques Multiobjectifs (AGMs). Ces derniers sont une adaptation des Algorithmes Génétiques (AGs) classiques à l'optimisation multiobjectif. Les idées générales des AGs classiques sont reprises dans les AGMs avec quelques modifications aux niveaux des opérateurs de reproduction (croisement et mutation). Selon le type d'algorithme adopté, d'autres éventuelles modifications peuvent être ajoutées notamment dans la phase de génération de population initiale.

La notion de multiobjectivité au sein des AGs nécessite l'exploitation de nouveaux aspects mathématiques tels que le concept de « dominance » et le concept du « front de Pareto ». Ces dernières interviennent dans la phase d'évaluation des individus de la population considérée.

L'objectif de ce mémoire est d'étudier les Algorithmes Génétiques Multiobjectifs et de les implémenter. Cette implémentation aura pour application un problème économique très célèbre connu sous le nom de « problème de sélection de fournisseurs ».

Le problème de sélection de fournisseurs est un problème multiobjectif multidisciplinaire très largement étudié dans la littérature. Il s'agit de sélectionner un

fournisseur (de services ou de produits) en considérant plusieurs objectifs, souvent contradictoires, tels que : la minimisation de prix, la maximisation de qualité, la minimisation du délai...ect.

Le problème de sélection de fournisseurs est au même temps un problème complexe composé d'une série de sous-problèmes. Chacun de ces sous-problèmes nécessite des outils théoriques et opérationnels spécifiques, d'où des méthodes de résolutions particulières. La méthode de résolution étudiée dans ce mémoire, la méthode des AGMs, est supposé être applicable dans la phase dite « pondération des critères ». Cette dernière est l'une des phases du processus de sélection les plus délicates et les plus coûteuse en terme de temps. Elle est ainsi du fait qu'elle s'appuie sur la notion d'expertise et de subjectivité.

Notre objectif se résume donc en les deux points suivants :

1. Vérifier la possibilité d'application des AGMs à la phase de pondération des critères. Cette vérification se fait en essayant d'avoir une formulation multiobjectifs adéquate à ce sous problème.
2. En cas de possibilité d'application, adapter les AGMs au sous-problème de pondération des critères d'évaluation en redéfinissant les éléments des AGMs (codage, opérateurs, fonction d'évaluation...ect).

Le travail actuel est validé pratiquement pour le problème de sélection des entreprises pour l'octroi de marchés (un cas particulier du problème de sélection de fournisseur). Les données numériques, les offres de soumissions et leurs évaluations, sont obtenues à partir du bureau d'étude ATRIUM à Blida.

Ce mémoire est composé essentiellement de quatre chapitres :

Dans le premier chapitre destiné au Algorithmes Génétiques Multiobjectifs, on introduit d'abord les Algorithmes Génétiques classiques puis on évoque les principes de base communs à tout les AGs, pour passer ensuite aux aspects spécifiques des AGMs.

Le deuxième chapitre est la synthèse de la recherche bibliographique établit pour le problème multiobjectifs dit « Sélection d'entreprise pour l'octroi de marchés ». On y présente globalement le processus de sélection, et les différentes étapes de ce dernier. Nous verrons que chaque étape requiert des méthodes de résolutions bien appropriées. Ainsi nous pouvons vérifier que la phase de ce processus dans la quelle il serait possible d'appliquer les AGMs est bien la phase de pondération de critères.

Le troisième chapitre constitue la conception détaillée ayant pour but les deux points cités ci-dessus (c.à.d. la vérification de possibilité d'application ensuite l'application elle-même). On détaille en premier lieu la formulation multiobjectif convenable à la phase de pondération des critères. Puis, on présente les éléments de L'AGM que nous proposons pour la résolution de ce problème. L'algorithme ainsi élaboré est intitulé **Algorithme Génétique Multiobjectif pour le problème de Sélection des Entreprises (AGMSE)**.

L'AGMSE nécessite comme tout autre AG un réglage de paramètres et des tests numériques. Ceci fera l'objet du quatrième chapitre.

L'implémentation de l'AGMSE est réalisée sous l'environnement MATLAB dans le but de simplifier les calculs (en exploitant la faculté de MATLAB à gérer rapidement et facilement les calculs matriciels). Le logiciel dédié sera présenté également dans ce dernier chapitre. Ce logiciel qui devra être très simple d'un côté. Et d'un autre côté, il devra servir, accompagné ou non d'autres outils, d'outil d'aide à la décision au sein du bureau d'étude.

CHAPITRE 1

Les Algorithmes Génétiques Multiobjectifs

CHAPITRE 1

Les Algorithmes Génétiques Multiobjectifs

Introduction :

L'optimisation multiobjectif est, sans aucun doute à l'heure actuelle, un domaine de recherche en plein essor tant du point de vue de la recherche théorique que des applications du monde réel. La grande majorité de ces problèmes sont de nature multiobjectif du fait qu'il est rare de pouvoir effectuer un choix où une solution est parfaite à tous points de vue.

Parmi les méthodes les plus étudiées et exploitées dans le domaine de l'optimisation multiobjectif figurent les Algorithmes Génétiques Multiobjectifs (AGMs). Ces derniers représentent la tendance actuelle des Algorithmes Génétiques standards, initialement proposés par John Holland [HOL75].

Le présent chapitre a pour but d'exposer les Algorithmes Génétiques Multiobjectifs. Etant une version des AGs standard, les AGs multiobjectifs reprennent plusieurs notions de ces derniers avec d'éventuelles modifications. Pour cela, nous commençons par une présentation des AGs standards¹ y compris leurs mises en œuvre et leurs éléments. Nous passerons par la suite aux particularités des AGMs vis-à-vis des AGs standards. Nous résumons à la fin les variantes des AGMs les plus connues.

1. Généralités et mise en œuvre d'un algorithme génétique classique :

1.1 Généralités sur les algorithmes génétiques classiques :

Les Algorithmes Génétiques (AGs) sont des méthodes d'optimisation stochastiques très largement utilisés.

¹Cette présentation nous est essentielle puisque nous ne sommes pas assez familiarisés avec les AGs standards.

Un algorithme génétique reproduit l'évolution naturelle d'organismes vivants, génération après génération, en respectant les phénomènes d'hérédité et la loi de survie énoncée par Darwin². Dans une population, ce sont les individus les mieux adaptés au milieu qui survivront et pourront donner une descendance.

Dès les années cinquante, plusieurs biologistes ont simulé des structures biologiques sur ordinateur. Dans la décennie soixante, John Holland [HOL75] détermina une analogie entre un individu dans une population d'individus et une solution d'un problème d'optimisation combinatoire dans un ensemble de solutions. L'ouvrage de Goldberg [GLOD89] introduit l'application des AGs sur des problèmes concrets.

Un individu est caractérisé par une structure de données qui représente une solution. La force d'un individu, encore appelée fitness, exprime la qualité de la solution correspondante. Les opérateurs génétiques de croisement et de mutation agissent sur les structures de données associées aux individus. Ils permettent de parcourir l'espace de solutions du problème. La population correspond à l'ensemble courant de solutions. Le renouvellement de cette dernière, autrement dit la création d'une nouvelle génération, est obtenu par itération de l'AG, créant de nouveaux individus et détruisant d'autres. L'exécution d'un tel algorithme doit conduire, à partir d'une population initiale, après de nombreuses générations, à une population où les individus sont tous forts (un ensemble de bonnes solutions).

1.2 Mise en œuvre d'un algorithme génétique :

Un AG recherche le ou les extrema d'une fonction définie sur un espace de données.

Pour l'utiliser, on doit disposer des éléments suivants :

- Un codage approprié des solutions,
- Un moyen pour obtenir la population initiale,
- Une fonction d'évaluation mesurant la fitness d'un individu,
- Un mode de sélection des individus à reproduire,
- Des opérateurs génétiques adaptés au problème,
- Des valeurs pour les paramètres de l'algorithme.

² Théorie *contradictoire* avec la réalité de la création des êtres vivants : Nous ne retenons dans notre travail que l'aspect *purement mathématique* de cette dite « théorie ».

1.2.1 Codage :

Le codage s'inspire des chromosomes rencontrés dans la nature, qui est une chaîne de gènes. Ils existent plusieurs types de codages, nous citons parmi eux les deux les plus utilisés : le codage binaire et le codage réel.

a) Codage binaire :

C'est le codage initialement retenu par John Holland. Dans ce codage, les gènes sont représentés par des 0 et des 1.

Définition 1. *Séquence/Chromosome/Individu / gène (Codage binaire)*

Nous appelons une séquence (chaîne, chromosome, individu) de longueur $L(A)$ une séquence $A = \{a_1, a_2, \dots, a_l\}$ avec $\forall i \in \{1, \dots, l\}, a_i \in V = \{0, 1\}$.

Diverses stratégies de codage binaire peuvent être utilisées [BOU06], nous citons :

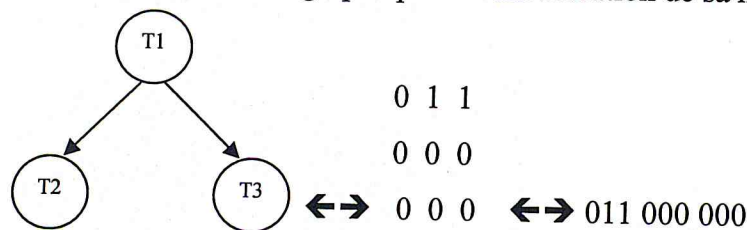
- La représentation sur 5 bits d'un entier appartenant à $[0, 31]$ par son code dans le système binaire.

$$3 \leftrightarrow 00011 ; 6 \leftrightarrow 00110 ; 9 \leftrightarrow 01001$$

- La représentation d'un vecteur de trois variables entières dans $[0, 31]$ par la concaténation de leurs codes binaires.

$$(3, 6, 9) \leftrightarrow 00011 00110 01001$$

- La représentation d'un graphe par la concaténation de sa matrice d'incidence.



Ce codage binaire présente l'avantage d'être très simple à utiliser. Les opérateurs des AGs, que nous présentons par la suite, sont souvent assez simples à mettre en place dans le cas de ce codage. Malgré tout, quelques inconvénients existent (voir à ce sujet Michalewicz [MIC92] pour des exemples concrets) :

- Il peut être difficile d'adapter ce codage à certains problèmes. La représentation binaire traditionnelle utilisée pour les algorithmes génétiques crée des difficultés pour les problèmes d'optimisation de grandes dimensions à haute précision numérique. Par exemple, si l'on doit utiliser 100 variables appartenant au

domaine $[-500, 500]$ et si l'on a besoin d'une précision de l'ordre de 10^{-6} , on doit utiliser des chromosomes d'une longueur de $l=3000$. Cela génère en retour un espace de recherche de l'ordre de 10^{1000} . Pour de tels problèmes, les algorithmes génétiques basés sur des représentations binaires ont de faibles performances, comme le montre Michalewicz et al. [MIC94].

- D'après le même auteur, ce codage peut créer une convergence mais non l'obtention de la valeur optimale.
- Suivant le problème, l'exécution de l'algorithme peut être coûteuse en temps.
- Le croisement et la mutation peuvent être inadaptés (pour la création, par exemple, d'individus n'appartenant pas à l'espace de recherche).

Une des améliorations majeures consiste alors à se servir directement de nombres réels, d'où : **le codage réel**.

b) Le codage réel :

Les inconvénients du codage binaire soulevés ci-dessus ont permis d'introduire un nouveau type de codage : **le codage réel**. Il s'agit de se servir directement des nombres réels se rapportant au problème traité. Les résultats donnés dans [MIC92], [MIC94] montrent que la représentation réelle aboutit souvent à une meilleure précision et qu'en règle générale le gain en termes de temps de calcul (CPU) est important. Ils en concluent qu'une représentation plus *naturelle* du problème offre des solutions plus efficaces.

En utilisant le codage réel, l'individu n'est alors plus qu'un nombre réel dans l'espace des valeurs permises : $A=\{a\}$, $a \in D \subset \mathbb{R}$. L'opérateur de sélection reste identique à celui de la roulette biaisée ou du tournoi. En revanche, les auteurs utilisent de nouveaux opérateurs de croisement et de mutation (voir [MIC92] [MIC94]) pour une présentation détaillée de ces opérateurs à codage réel et voir [VAL04] pour une étude de ces opérateurs dans le cadre d'un jeu répété inflation-chômage) [VAL04].

1.2.2 Population initiale :

Dans un algorithme de recherche par voisinage, le choix des solutions initiales de bonne qualité accélère la convergence vers l'optimum.

On peut obtenir la population initiale par :

- Génération aléatoire, permettant d'avoir des solutions de diverses caractéristiques.
- Des heuristiques, donnant des individus « fort » mais qui risquent de faire converger l'algorithme vers des optima locaux [DEL92].
- Un mélange de solutions heuristiques et aléatoires, permettant d'avoir à la fois des individus « fort » et une diversité.
- Duplication et évolution d'une seule solution, lorsque le problème est fortement contraint et l'obtention de plusieurs solutions est difficile. On prend une solution et on modifie par mutation et croisement.

1.2.3 Evaluation

L'évaluation consiste à mesurer la « force » de chaque individu « fitness » de la population.

Définition 2. *Performance -fitness- d'une séquence* Nous représentons la performance d'une séquence A par une fonction à valeurs positives, $f : A \rightarrow f(A) \in \mathbb{R}^+$.

Si le problème est de maximisation, la fonction objective peut être utilisée comme mesure. En cas de minimisation, on utilise une fonction qui varie inversement avec la fonction objective. Soit le problème,

$$\min_x C(x)$$

On peut mesurer la fitness d'un individu correspondant à une solution x_i par :

- $f(X_i) = UB - C(X_i)$ telle que UB est une borne supérieure de $C(X)$.
- $f(X_i) = \frac{1}{C(X_i)}$
- $f(X_i) = \frac{mx - C(X_i)}{mx - mn}$,

tels que mx et mn sont respectivement le mauvais et le meilleur des coûts de la population courante.

Aucune condition particulière n'est requise pour la fonction objective, il suffit qu'elle retourne des valeurs numériques comparables. La performance de l'AG peut être sensible au choix de la mesure de la fitness comme on le verra dans notre conception.

1.2.4 Sélection

L'opérateur de sélection est chargé de définir quels seront les individus d'une population qui vont être dupliqués dans la population suivante et vont servir de parents (application de l'opérateur de croisement).

Soit n le nombre d'individus de la population courante, on doit en sélectionner $n/2$ (l'opérateur de croisement nous permet de repasser à n individus).

Cet opérateur est peut-être le plus important puisqu'il 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 reliée à son efficacité relative au sein de la population.

A l'inverse d'autres techniques d'optimisation, les AGs ne requièrent pas d'hypothèse particulière sur la régularité de la fonction objectif. L'algorithme génétique n'utilise notamment pas ses dérivées successives, ce qui rend très vaste son domaine d'application. Aucune hypothèse sur la continuité n'est non plus requise. Néanmoins, dans la pratique, les algorithmes génétiques sont sensibles à la régularité des fonctions qu'ils optimisent.

Le peu d'hypothèses requises permet de traiter des problèmes très complexes. La fonction à optimiser peut ainsi être le résultat d'une simulation.

La sélection permet d'identifier statistiquement les meilleurs individus d'une population et d'éliminer les mauvais. On trouve dans la littérature un nombre important de principes de sélection plus ou moins adaptés aux problèmes qu'ils traitent.

Les quatre types de sélections les plus cités dans la bibliographie sont : [SOU04]

a) *La loterie biaisée ou roulette wheel :*

Cette méthode est la plus connue et la plus utilisée. Avec cette méthode chaque individu a une chance d'être sélectionné proportionnelle à sa performance. Donc plus les individus sont adaptés au problème, plus ils ont de chances d'être sélectionnés. Pour utiliser l'image de la "roue du forain", chaque individu se voit attribuer un secteur dont l'angle est proportionnel à son adaptation, sa "fitness".

On fait tourner la roue et quand elle cesse de tourner on sélectionne l'individu correspondant au secteur désigné par une sorte de "curseur", curseur qui pointe sur un secteur particulier de celle-ci après qu'elle se soit arrêté de tourner.

Cette méthode, bien que largement répandue, a pas mal d'inconvénients :

- Une forte variance : Il n'est pas impossible que sur n sélections successives destinées à désigner les parents d'une nouvelle génération, la quasi-totalité des n individus sélectionnés soient des individus ayant une fitness vraiment mauvaise. Par conséquent, aucun individu à forte fitness ne fasse partie des parents de la nouvelle génération.

Ce phénomène est très dommageable car cela va complètement à l'encontre du principe des AGs qui veut que les meilleurs individus soient sélectionnés de manière à converger vers une solution la plus optimale possible.

- Une perte de diversité : ce qui signifie une domination écrasante d'un individu "localement supérieur". Prenons l'exemple d'un individu ayant une fitness très élevée par rapport au reste de la population, disons dix fois supérieure, il n'est pas impossible qu'après quelques générations successives on se retrouve avec une population ne contenant que des copies de cet individu. Le problème est que cet individu avait une fitness très élevée, mais que cette fitness était toute relative, elle était très élevée mais seulement en comparaison des autres individus. On se retrouve donc face au problème de convergence prématurée vers un optimum local. Il existe certaines techniques pour essayer de limiter ce phénomène, comme par exemple le "scaling", qui consiste à effectuer un changement d'échelle de manière à augmenter ou diminuer de manière forcée la fitness d'un individu par rapport à un autre selon leur écart de fitness.

Malgré tout, il est conseillé d'opter plutôt pour une autre méthode de sélection.

b) La méthode élitiste :

Cette méthode consiste à sélectionner les n individus dont on a besoin pour la génération suivante en prenant les n meilleurs individus de la population courante après l'avoir triée de manière décroissante selon la fitness de ses individus.

Cette méthode est encore pire que celle de la loterie biaisée dans le sens où elle amènera à une convergence prématurée encore plus rapidement et surtout de manière encore plus sûre que la méthode de sélection de la loterie biaisée ; en effet, la pression de la sélection est trop forte, la variance nulle et la diversité inexistante, du moins le peu de diversité qu'il pourrait y avoir ne résultera pas de la sélection mais plutôt du croisement et des mutations.

Là aussi, il faut opter pour une autre méthode de sélection.

c) La sélection par tournois :

Cette méthode est celle avec laquelle on obtient les résultats les plus satisfaisants.

Le principe de cette méthode est le suivant : on effectue un tirage avec remise de deux individus de la population courante, et on les fait "combattre". Celui qui a la fitness la plus élevée l'emporte avec une probabilité p comprise entre 0.5 et 1. On répète ce processus n fois de manière à obtenir les n individus de la population suivante qui serviront de parents.

La variance de cette méthode est élevée et le fait d'augmenter ou de diminuer la valeur de p permet respectivement de diminuer ou d'augmenter la pression de la sélection.

d) *La sélection universelle stochastique :*

Cette méthode semble être très peu utilisée et qui, en plus, possède une variance faible. Nous n'entrerons donc pas dans les détails, on se contentera d'exposer sa mise en œuvre : On prend l'image d'un segment découpé en autant de sous-segments qu'il y a d'individus. Les individus sélectionnés sont désignés par un ensemble de points équidistants.

Dans toutes ces méthodes, la diversité de la population doit être entretenue au cours des générations afin de parcourir le plus largement possible l'espace des solutions. C'est le rôle des opérateurs de croisement et de mutation. Les premiers sont des opérateurs sexués, deux parents sont nécessaires pour générer deux enfants, les seconds sont des opérateurs asexués qui n'ont besoin que d'un parent pour générer un enfant.

1.2.5 Croisement

Bien que de nombreuses variantes de ces opérateurs existent, nous présentons ici celles qui sont, le plus couramment, utilisées :

a) *Le croisement à un point :*

C'est le croisement le plus simple. Si n est la longueur du chromosome, on choisit aléatoirement une position s entre 1 et $n-1$. On coupe les deux parents à cette position et on les recolle en les croisant.

$$\text{parent 1} = B_1 B_2 B_3 \dots B_s B_{s+1} \dots B_n$$

$$\text{parent 2} = C_1 C_2 C_3 \dots C_s C_{s+1} \dots C_n$$

$$\text{fils 1} = B_1 B_2 B_3 \dots B_s C_{s+1} \dots C_n$$

$$\text{fils 2} = C_1 C_2 C_3 \dots C_s B_{s+1} \dots B_n$$

b) *Le croisement à k points :*

Le croisement à k points est une généralisation du croisement à un point. On choisit aléatoirement k positions entre 1 et n, créant ainsi K+1 morceaux. Le 1^{er} fils (respectivement le 2^{ème} fils) est la concaténation des morceaux d'ordre pair (impaire) du 1^{er} parent et des morceaux d'ordre impaire (paire) du 2^{ème} parent.

Soient $k = 2$ et i, j les points de croisement générés,

$$\text{parent 1} = B_1 B_2 B_3 \dots B_i B_{i+1} \dots B_j B_{j+1} \dots B_n$$

$$\text{parent 2} = C_1 C_2 C_3 \dots C_i C_{i+1} \dots C_j C_{j+1} \dots C_n$$

$$\text{fils 1} = B_1 B_2 B_3 \dots B_i C_i C_{i+1} \dots C_j B_{j+1} \dots B_n$$

$$\text{fils 2} = C_1 C_2 C_3 \dots C_i C_{i+1} B_{i+1} \dots B_j C_{j+1} \dots C_n$$

c) *Le croisement uniforme :*

Avec ce croisement, chaque gène de chaque fils est celui de l'un des deux parents avec la même probabilité. Pratiquement, on utilise un « masque de croisement », qui est un mot binaire de même longueur que les individus.

$$\text{parent 1} = B_1 B_2 B_3 B_4 B_5$$

$$\text{parent 2} = C_1 C_2 C_3 C_4 C_5$$

On génère deux fois cinq valeurs aléatoire dans $\{0, 1\}$, soient $(0,1,1,0,1)$ et $(1,0,0,1,1)$.

Le 0 indique qu'on prend le gène du 1^{er} parent et le 1 indique qu'on prend celui du 2^{ème} parent. On obtient :

$$\text{fils 1} = B_1 C_2 C_3 B_4 C_5$$

$$\text{fils 2} = C_1 B_2 B_3 C_4 C_5$$

d) *Pseudo algorithme : [OUM00]*

Soit **Pop** un vecteur représentant les individus de la population, **N** sa taille, **Pc** la probabilité du croisement et **Nc** le nombre de couples qui subissent un croisement, l'opérateur de croisement peut être mis sous forme algorithmique comme suit:

Croisement (Pc, Pop) :

- $N_c = \text{ENT}[P_c * N/2]$ avec $\text{ENT}[r]$ représente la partie entière d'un nombre réel r
- **Répéter** N_c fois:
 - Choisir deux indices i et j dans $[1, N]$ avec une distribution uniforme tels que $i \neq j$
 - Croiser $[\text{indi}, \text{indj}]$ Selon le type de croisement choisi
 - Rajouter les fils résultant en fin de Pop

1.2.6 Mutation

Une mutation est une perturbation introduite pour modifier une solution individuellement, par exemple, en cas de codage binaire, la transformation d'un 0 en un 1 ou inversement. En général, on décide de muter une solution avec une probabilité assez faible. Le but de mutation est d'introduire un élément de diversification et d'innovation.

La mutation peut créer des chromosomes difficiles à avoir avec le croisement. Elle correspond à une petite perturbation de la solution. On cite les trois variantes suivantes :

Modifier la valeur d'un gène : le gène de position 3 est modifié de B3 en B5.

parent = $B_1B_2B_3B_4B_5 \rightarrow$ fils = $B_1B_2B_5B_4B_5$

Permuter deux gènes consécutifs : les gènes de position 3 et 4 sont échangés.

parent = $B_1B_2B_3B_4B_5 \rightarrow$ fils = $B_1B_2B_4B_3B_5$

Permuter deux gènes quelconques : les gènes de position 1 et 4 sont échangés

parent = $B_1B_2B_3B_4B_5 \rightarrow$ fils = $B_4B_2B_3B_1B_5$

Inverser l'ordre d'une sous chaîne du chromosome : la sous chaîne $B_2B_3B_4B_5$ est inversée.

parent = $B_1B_2B_3B_4B_5 \rightarrow$ fils = $B_1B_5B_4B_3B_2$

Pseudo algorithme : [OUM00]

Si Pop représente une population donnée et sa taille. L'algorithme suivant peut résumer le principe de la mutation :

Mutation (Pop, Pm)

- **Pour** chaque individu i de Pop tel que $i = 1, N$ faire
 - **Pour** chaque bit de l'individu i faire
 - Générer un nombre réel r aléatoirement sur l'intervalle $[0,1]$
 - Si $r < Pm$: inverser le bit
 - **FinPour**
- **FinPour**

1.2.7 Condition d'arrêt :

Comme toute procédure itérative, un AG s'arrête dès qu'une certaine condition d'arrêt est vérifiée. Plusieurs conditions peuvent être définies, citons par exemple :

Arrêt après un nombre de génération fixé à priori : C'est la plus utilisée lorsqu'un impératif de temps de calcul est imposé ou lorsque la performance de l'algorithme est à tester. Un nombre de générations est produit signifie qu'un pourcentage de l'espace des solutions est exploré.

Arrêt lorsque la population cesse d'évoluer en solution ou n'évolue plus rapidement : Cela s'interprète par l'atteinte d'un minimum local ou lorsque la population devient homogène. On peut penser que la population se situe à proximité de solutions optimums. Ce critère permet d'arrêter l'algorithme quand la plupart des gains ont été obtenus. L'inconvénient de cette stratégie est qu'il n'est pas possible de prévoir la durée d'exécution.

Une combinaison des deux conditions : On arrête lorsqu'on évolue plus dans la qualité de la solution ou bien lorsqu'on atteint un nombre maximum de générations.

2. Les paramètres des algorithmes génétiques

De la même manière que pour les autres méthodes stochastiques, l'efficacité des algorithmes génétiques dépend fortement de leurs paramètres. Nous présentons brièvement les principaux paramètres de ces algorithmes comme suit :

2.1 *Le taux de mutation (P_m) :*

C'est la probabilité avec laquelle on applique l'opérateur de mutation sur un individu de la population. Si cette probabilité est très petite, plusieurs individus pouvant être optimaux ne seront jamais générés. La population risque donc de ne pas être diversifiée. Dans le cas contraire, il y aura une grande perturbation des individus générés et la recherche devient purement aléatoire. Ainsi, les individus générés perdent leurs ressemblances avec leurs parents. L'algorithme va donc perdre sa capacité d'apprentissage à partir de l'historique de la recherche [TAL95]. Des études empiriques conseillent d'utiliser un taux de mutation qui vaut 0.01 [GOL94]. Les taux de mutation sont également faibles dans les populations naturelles, ce qui nous conduit à penser que la mutation est considérée comme un mécanisme d'adaptation secondaire pour les AG.

2.2 *Le taux de croisement (P_c):*

C'est la probabilité avec laquelle un ensemble d'individus est choisi pour leurs appliquer l'opérateur de croisement. Son choix est généralement expérimental et sa valeur est très souvent prise entre 0.5 et 0.9 [LER00]. Plus le taux est élevé, plus la population subit d'importants changements. Ainsi, la convergence va être très rapide si le taux de croisement est proche de 1.

2.3 *Le taux de renouvellement (Γ) [DAV91] :*

Ce paramètre désigne le pourcentage de la population qui doit être remplacé par leurs enfants dans la génération suivante. Si ce taux vaut 1, toute la population doit être remplacée.

2.4 *Le taux de diversité :*

Il indique le degré de ressemblance des individus dans la population. Preux [PRE94] a proposé une formule mathématique pour la mesure de la diversité de la population qui est la suivante:

$$\{\Delta_H = \sum \Delta_H(X_i, Y_j) / (X_i, Y_j) \in P^2, i > j\}$$

P : Désigne une population

$\Delta_H(X_i, Y_j)$: La distance de Hamming entre deux individus (X_i, Y_j)

Remarque :

Pour le bon fonctionnement des algorithmes génétiques, il faut bien choisir les valeurs de ses paramètres cités précédemment ce qui n'est pas toujours évident. Dejong [DEJ75] a effectué une série d'analyses des paramètres pour l'optimisation de cinq fonctions successives. Cette étude a suggéré, pour avoir de bonnes performances, de choisir :

- Une probabilité du croisement élevée.
- Une probabilité de mutation faible (Inversement proportionnelle à la taille de la population).
- Une population de taille modérée.

Un problème important est le codage des données comme le dit Goldberg lui-même : « le codage des données est un art, et de cet art dépend le succès ou l'échec de la tentative » [ALL94].

2.5 La taille de la population N :

Il est conseillé de prendre comme taille de la population la valeur correspondante à la longueur du codage des individus [LER00]. En effet, si la taille de la population est très grande, l'évaluation de tous les individus de la population peut s'avérer trop long. Par contre, si, elle est très petite, l'algorithme peut converger trop rapidement, l'algorithme n'arrivera peut être pas à l'optimum globale. Goldberg a évalué cette taille à $N=30$.

3. Schéma général d'un algorithme génétique :

A un niveau très général, le fonctionnement d'un AG est basé sur les phases suivantes :

Etape 0 : Définir un codage du problème et les paramètres de l'algorithme;

Etape 1 : $t = 0$, créer une population initiale $P(0) = x_1, x_2, x_3, \dots, x_p$;

Etape 2 : Calculer la force $f(x_i)$ de chaque individu $x_i, i = 1..p$;

Etape 3 : Sélectionner 'p' individus de $P(t)$; les ranger dans un ensemble $S(t)$;

Etape 4 : Pour chaque paire d'individus de $S(t)$:

Appliquer l'opérateur de croisement avec une probabilité p_c ;

Appliquer l'opérateur de mutation avec une probabilité p_m ;

Etape 5 : $t = t+1$, $P(t) = S(t)$,

Si critère d'arrêt atteint alors Stop,

Sinon : aller à l'étape 2 ;

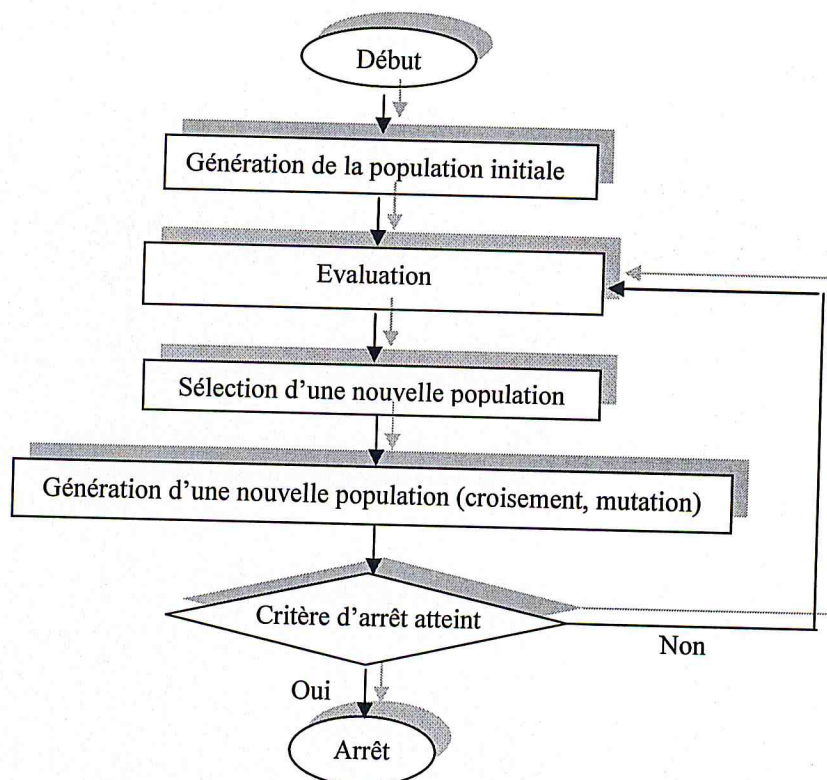


Fig 1.1 : Organigramme général d'un algorithme génétique

4. Les particularités des algorithmes génétiques : [BOU06]

Les AGs ont retenu l'attention de nombreux chercheurs et praticiens de l'optimisation en raison de leurs particularités attrayantes ci-dessous mentionnées :

- Ils travaillent sur un codage des paramètres du problème et non sur les paramètres eux-mêmes.
- Ils effectuent la recherche d'un optimum à partir d'une population et non d'un point unique.

- Ils utilisent :
 - Les informations apportées par la fonction objectif elle-même et non par ses fonctions dérivées ou son gradient. La fonction objectif peut être le résultat d'une optimisation.
 - Des règles de transition probabilistes qui leur permettent de sortir d'un optimum local.
- Le peu d'hypothèses requises leur permet de traiter des problèmes très complexes.
- Leur parallélisme intrinsèque augmente leurs possibilités pratiques.
- Ils sont attrayants pour l'utilisateur car ils ne réclament pas en général des connaissances très pointues en optimisation.

5. Les Algorithmes Génétiques Multiobjectifs:

Les Algorithmes Génétiques Multiobjectifs (notés AGMs) sont la version des AGs destinée à la résolution des problèmes multiobjectifs. En effet, ces AGs se sont montrés très efficaces face à de tels problèmes.

La section suivante a pour but de présenter les AGMs, pour cela nous commençons d'abord par définir les deux piliers théoriques des AGMs qui sont : l'optimisation multiobjectif et la dominance de Pareto.

5.1 Optimisation multiobjectif :

L'optimisation multiobjectif est, sans aucun doute à l'heure actuelle, un domaine de recherche en plein essor tant du point de vue de la recherche théorique que des applications du monde réel. La grande majorité de ces problèmes sont de nature multiobjectif du fait qu'il est rare de pouvoir effectuer un choix où une solution est parfaite à tous points de vue. On parle d'optimisation multiobjectif dans les cas complexes où l'on doit optimiser simultanément plusieurs objectifs, souvent contradictoires. Ceci amène à choisir une solution de compromis, Pareto optimal, parmi une multitude de solutions possibles.

Nous définissons dans ce qui suit ce qu'est un problème d'optimisation multiobjectif.

Définition d'un Problème d'optimisation multiobjectif : [GRA08]

Un problème d'optimisation multiobjectif est défini par un triplet (X, F, g) qui consiste à minimiser $F(\mathbf{x})$ pour $\mathbf{x} \in X$ sachant que $g(\mathbf{x}) \leq 0$.

a) *Les variables de décision :*

Dans un problème d'optimisation, les m variables de décision sont les valeurs à choisir. On notera ces variables par x_i tel que $i \in \{1, \dots, m\}$. Le vecteur X de m variables de décision est représenté par $X = (x_1, x_2, \dots, x_m)$.

b) *Les fonctions objectifs :*

Les n fonctions objectifs à optimiser sont notées f_i , $i \in \{1, \dots, n\}$ et le vecteur f de n objectifs pour $\mathbf{x} \in X$ est représenté par : $f(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_n(\mathbf{x}))$

c) *Les contraintes :*

Dans un problème d'optimisation multiobjectif, il est très fréquent de limiter les valeurs que peuvent prendre les variables de décision. Ces restrictions se traduisent par les g contraintes. Elles sont notées $g_i(\mathbf{x})$, $i \in \{1, \dots, p\}$.

5.2 Domination et front de Pareto [GRA08]

Soit $U = (u_1, u_2, \dots, u_m)$ et $V = (v_1, v_2, \dots, v_m)$ deux vecteurs de décision.

- V domine U (et on le note $V \prec U$) pour un problème de *minimisation* si et seulement si :

$$\begin{cases} \forall i \in [1, n]: f_i(V) \leq f_i(U) \\ \exists i \in [1, n]: f_i(V) < f_i(U) \end{cases}$$

- Le front de Pareto FP^* est l'ensemble des vecteurs de décision qui ne sont pas dominés.

$$FP^* = \{x \in X \mid \nexists x' \in X \text{ tq } x' \text{ domine } x\}$$

Le front de Pareto est l'ensemble des solutions de compromis. Une solution est dite « Pareto optimale » si elle appartient au front de Pareto.

Sur la figure 1-2, les points A et B sont deux points du front de Pareto : A ne domine pas B, B ne domine pas A, mais tous les deux dominent le point C.

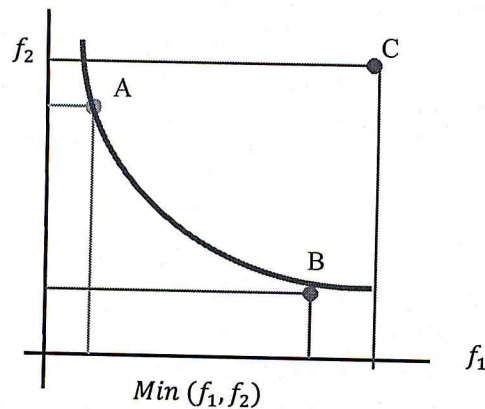


Fig.1-2 Exemple de front de Pareto [GRA08]

5.3 Adaptation des algorithmes génétiques à l'optimisation multiobjectif : [BEN05]

Comme nous l'avons vu précédemment, lors de l'application d'un AG, l'algorithme manipule une population des solutions au lieu d'une seule, comme dans le cas de l'utilisation de la plupart des autres méthodes d'optimisation non évolutionnaires.

Cette différence donne aux AGs un avantage crucial quand il s'agit de trouver de multiples solutions Pareto-optimales.

Mais, afin d'adapter les AGs mono-objectif à la recherche d'un ensemble de solutions de Pareto, certaines modifications doivent être apportés à ces algorithmes. En effet, le but poursuivi consiste à trouver non simplement plusieurs solutions de Pareto, mais, une bonne approximation de l'ensemble de Pareto du problème considéré. En termes d'AGs, il faut diriger la pression de sélection vers cet ensemble tout en préservant la diversité entre les individus non dominés (voir la figure 1-3).

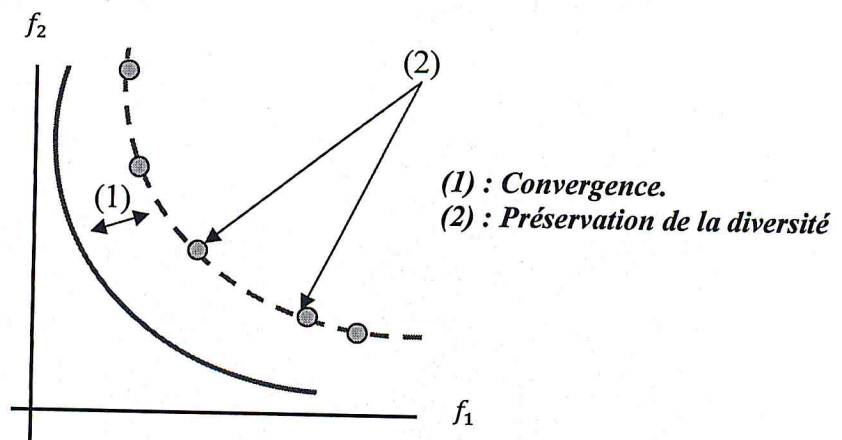


Fig.1-3 Principe de l'évaluation de performance des individus dans les AGMs [BEN05]

Récemment, un grand nombre de travaux menés sur l'application des algorithmes évolutionnaires aux problèmes d'optimisation multiobjectif ont présenté des résultats prometteurs. Dans cette section, nous illustrons un certain nombre d'AGMs décrits dans la littérature. Cependant, nous distinguons deux groupes d'algorithmes : des algorithmes qui se basent sur le concept de dominance de Pareto et d'autres qui n'utilisent pas ce concept (dits non-Pareto).

Dans ce qui suit, nous définissons chaque approche en donnant un exemple pour chacune d'elles. Nous récapitulons ensuite les aspects les plus importants des autres variantes d'AGMs.

5.3.1 Algorithmes non Pareto

Comme l'indique leur nom, ces algorithmes ne tiennent pas en compte le concept de dominance de Pareto lors de la comparaison des différents individus. En général, les méthodes dites non agrégées et non Pareto possèdent un processus de recherche qui traite séparément les objectifs.

Parmi ces algorithmes nous détaillerons, à titre d'exemple, l'algorithme VEGA.

Vector Evaluated Genetic Algorithm (VEGA) [SHA85]

Cet algorithme est basé sur une idée très simple : à chaque génération, la population est aléatoirement divisée en autant de sous-population (de tailles égales) qu'il y a d'objectifs, ensuite, chaque sous-population est évaluée selon l'une des fonctions objectifs. L'opérateur de sélection est ensuite restreint à chaque sous-population.

Algorithme : une itération de VEGA

- 1-Diviser la population en M sous-populations
 - 2-Sélection des individus pour chaque sous-population selon leurs performances pour un seul de ces M objectifs. Les mettre dans S.
 - 3- Appliquer les opérateurs génétiques sur les individus de S.
-
-

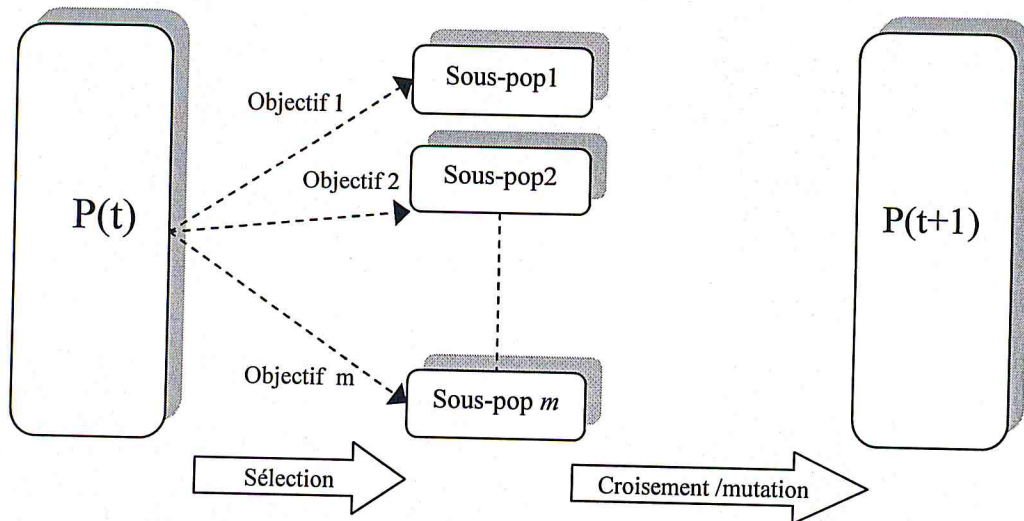


Fig. 1-4 Sélection parallèle dans l'algorithme VEGA [BEN05]

Discussion :

La méthode VEGA a tendance à créer des sous-populations dont les meilleurs individus sont spécialisés pour un objectif particulier. L'évolution de la population favorise l'apparition des espèces. En effet, comme la méthode de sélection ne tient compte que d'un seul objectif, elle privilégie les individus qui obtiennent une bonne performance pour cet objectif. Dès lors ces individus ne seront sélectionnés que lorsqu'on effectuera la sélection sur cet objectif. Les individus que Schaffer appelle les individus "milieu", parce qu'ayant une performance générale acceptable mais ne possédant aucun critère fort, vont être éliminés car ils ne seront sélectionnés dans aucune sous-population. Cette disparition entraîne la spécialisation des individus pour chaque objectif. Ce résultat est contraire au but initial de la méthode qui était de trouver un compromis entre les différents critères.

Schaffer propose deux heuristiques pour améliorer sa méthode :

- La première est un croisement restreint qui ajoute une préférence pour sélectionner les parents non dominés. Cette méthode a tendance à éviter la disparition des individus "milieu" mais elle a tendance également à accentuer la convergence.
- La seconde encourage le croisement entre individus spécialisés sur des objectifs différents. Mais les effets sont identiques à la première heuristique.

Remarque :

Les algorithmes non-Pareto sont souvent simples et faciles à mettre en œuvre. Mais, ces méthodes répartissent la population sur les extrema du front de Pareto (voir figure 1-5). En effet, l'évaluation mono-objectif des sous-populations ne permet pas la comparaison des diverses solutions multiobjectif disponibles à chaque itération en vue d'en exploiter les caractéristiques.

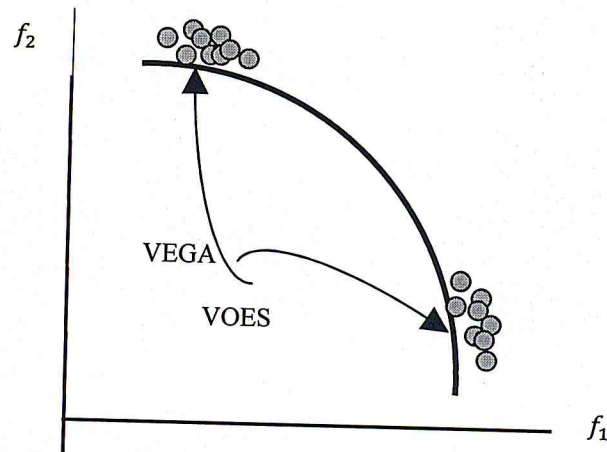


Fig.1-5 Défauts de l'attribution de la performance dans les algorithmes non-Pareto [BEN05]

5.3.2 Algorithmes basé sur la dominance de Pareto :

Contrairement aux méthodes précédente, ces d'algorithmes sont basés essentiellement sur la notion de dominance de Pareto. Dans cette section, nous détaillons l'un de ces algorithmes : l'algorithme NSGA-II (Elitist Nondominated Sorting Genetic Algorithm) [DEB00]. D'autres méthodes du même type seront présentées brièvement dans la section (5.4).

a) Principe générale de NSGA-II:

Dans cet algorithme, le double objectif convergence-diversité est atteint, d'une part, par l'utilisation d'un schéma du calcul de la performance qui préfère les solutions non dominées et, d'autre part, par l'application d'une technique de mesure de densité des solutions du même front non-dominé [DEB00].

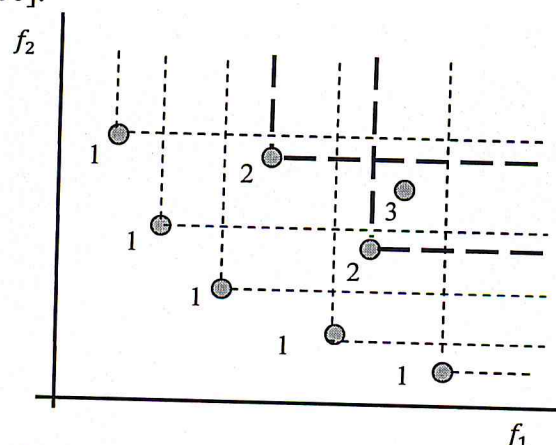


Fig.1-6 Tri de la population selon le principe de dominance (Ranking de Pareto) [BEN05]

La procédure utilisée par l'algorithme NSGA-II [DEB00] est basée sur la notion de tournoi, et compare deux individus comme suit :

- On compare tout d'abord leur rang de Pareto, défini itérativement de la manière suivante (voir Figure 1-6) : les individus non-dominés de la population courante sont dits de rang 1. Ils sont retirés de la population, et la procédure est itérée pour obtenir les rangs 2, 3, ..etc. L'individu avec le rang le plus faible est préféré à l'autre. Cependant, même ainsi relaxée, la comparaison au sens de Pareto est un ordre partiel, et on fait appel en cas d'ex-æquo à un critère basé sur la diversité.

La distance de surpeuplement : la distance de peuplement est définie comme suit (voir Figure 1-7) :

- pour chaque objectif i , on ordonne la population par valeurs croissantes de l'objectif.
- pour chaque individu p , on définit d_i , distance de peuplement partielle selon l'objectif i comme la somme des distances de p à ses deux plus proches voisins dans la liste ordonnée.
- la distance de peuplement totale D_p est donnée par la somme, sur l'ensemble des objectifs, des distances partielles.

On choisit alors celui des deux individus qui a la distance de peuplement la plus faible, i.e., qui est le plus isolé dans l'espace des objectifs.

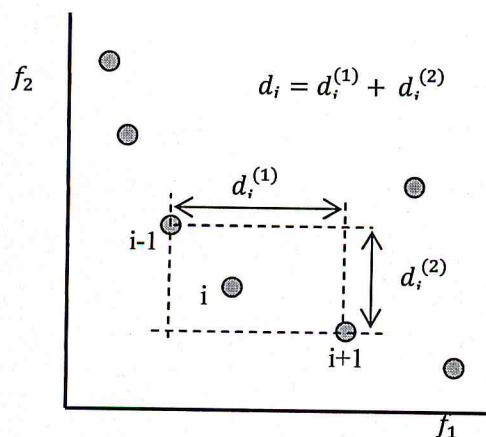


Fig.1-7 Mesure de surpeuplement [RIC06]

b) Algorithme NSGA-II:[BEN05]

Dans NSGA-II, la population des enfants $Q(t)$ est d'abord créée à partir de la population des parents $P(t)$. Ensuite, elles sont réunies en ensemble $R(t) = P(t) \cup Q(t)$, qui est trié selon le principe de dominance : l'ensemble R est divisé en un nombre de classes distinctes R_j ainsi : tous les individus non-dominés de R appartiennent l'ensemble R_1 ; ensuite, tous les éléments

non-dominés de $R \setminus R_1$ sont placés dans l'ensemble R_2 et ainsi de suite jusqu'à trier toute la population (voir figure 1-5). Ainsi nous obtenons r sous-ensembles tels que :

$$R = \bigcup_{j=1}^r R_j \text{ et } R_1 \prec R_2 \prec \dots \prec R_r$$

Notons qu'entre deux solutions de la même classe, aucune ne peut être considérée meilleure de l'autre compte tenu de tous les objectifs du problème. Le nombre total de classes, noté r , dépend de la population R .

Quand toute la population est triée, La population suivante $P(t+1)$ est remplie par les solutions des sous-ensembles non-dominé de $R(t)$ l'un après l'autre en commençant, évidemment, par le premier front. Pour choisir les solutions qui vont survivre du front dont seulement une partie peut être placée dans la population suivante, une mesure de la densité des solutions dans l'espace de critères, dite distance de surpeuplement (ou crowding distance), est utilisée (voir figure 1-8).

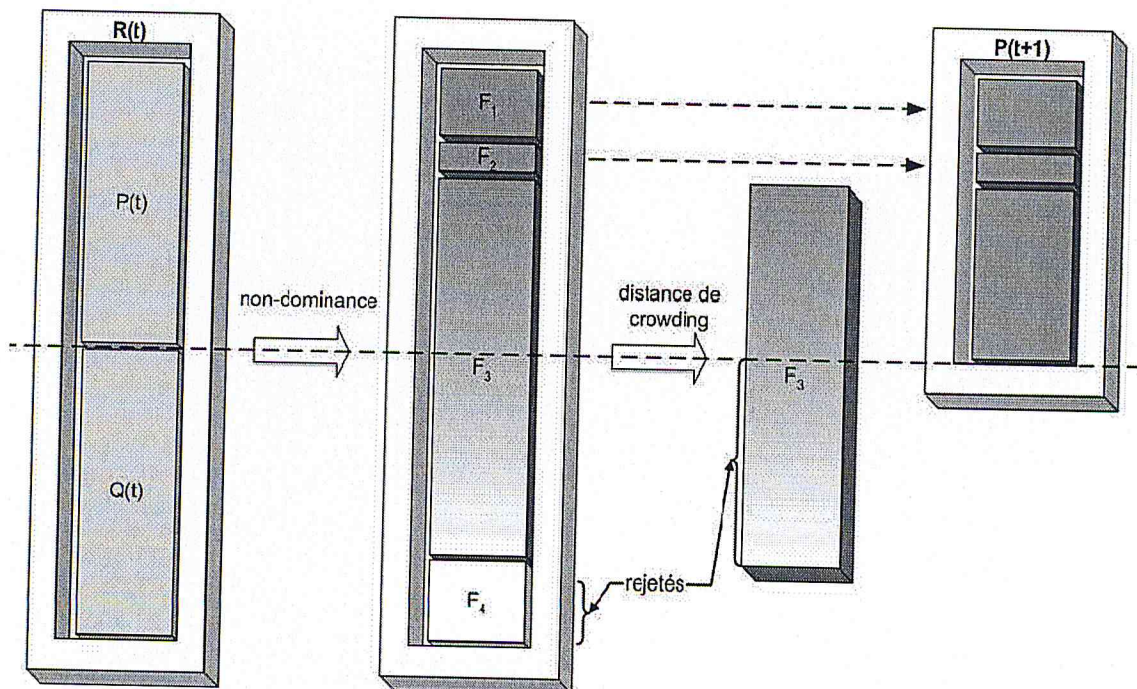


Fig.1-8 Principe de NSGA [BEN05]

Algorithme : une itération de NSGA-II

1. Créer la population des enfants $Q(t)$ à partir de la population des parents.
2. Réunir les populations des parents et des enfants $R(t) = P(t) \cup Q(t)$ et trier l'ensemble résultant en sous-ensembles F_i tels que :

$$R_i = \bigcup_{i=1}^r F_i \text{ et } F_1 \prec F_2 \prec \dots \prec F_r$$

3. Initialiser :
 - La nouvelle population $P_{t+1} = \emptyset$.
 - Le compteur des sous-ensembles non-dominés $i = 1$.
 4. Tant que $|P_{t+1}| + |F_i| < N$ faire :
 - $P_{t+1} \leftarrow P_{t+1} \cup F_i$
 - $i \leftarrow i + 1$
 5. Ordonner l'ensemble selon les « distance de surpeuplement » (crowding distances) et inclure les $N - |P_{t+1}|$ solutions ayant les valeurs de distance les plus grande
-
-

Remarque :

Les méthodes basées sur la dominance de Pareto sont tous des algorithmes élitistes. Ce qui explique le fait que leurs performances sont nettement supérieures à celles des autres algorithmes existants dans la littérature. [BEN05]

5.4 Récapitulation des variantes des AGMs

Selon le type de problème d'optimisation et son application, diverses variantes des AGs sont proposées dans la littérature. Le tableau suivant récapitule les particularités des AGMs les plus cités dans la littérature [KON06] :

Tab.1-1 : les AGMs les plus célèbres [KON06]

Algorithme	Action de fitness	Mécanisme de diversification	Élitisme	Population externe	Avantages	Inconvénients
VEGA [SHA85]	Chaque sous-population est évaluée par rapport à un objectif.	Non	Non	Non	Premier AGM implémenté (simple)	tend à converger vers les extrema de chaque objectif
MOGA [FON93]	Rang de Pareto	Sharing/Niching	Non	Non	Simple extension d'un AG mono objectif	Convergence lente. Problème lié à la taille de la niche.
WBGA [HAJ92]	Pondération moyenne des objectifs normalisés	Niching Pondération prédéfinie	Non	Non	Simple extension d'un AG mono objectif	Difficulté avec les fonctions objectif non convexes
NPGA [HOR94]	Aucune action de fitness, Sélection par tournoi	Nombre de niches comme un jeu décisif dans la sélection par tournoi	Non	Non	Très simple processus de sélection (tournois)	Problème dépendant de la taille de la niche. Paramètres additionnel pour la sélection par tournois
RWGA [MUR95]	Pondération moyenne des objectifs normalisés	Poids affectés aléatoirement	Oui	Oui	Implémentation simple et efficace	Difficulté avec les fonctions objectif non convexes
PESA [COR00]	Aucune action de fitness,	Densité à base de cellules	Élitisme pur	Oui	Facile à implémenter Efficace en termes de calculs	La performance dépend de la taille des cellules. Requiert des informations antérieurs sur l'espace de recherche
PAES [COR99]	Dominance de Pareto est utilisée pour remplacer un parent si un descendant domine	Densité à base de cellules comme un jeu décisif entre les parents et les descendants.	Oui	Oui	Mutation aléatoire Utilisation de la méthode « hill climbing » Facile à implémenter Efficace en termes de calculs	Non basée sur la population La performance dépend de la taille des cellules.
NSGA [SRI94]	Rangement basé sur le tri de non dominance	Sharing/Niching	Non	Non	Convergence rapide	Problème dépendant de la taille de la niche.
NSGA-II [DEB00]	Rangement basé sur le tri de non dominance	Distance de surpeuplement (Crowding distance)	Oui	Non	Paramètre unique (N) Bien testé Efficace	Les distances de rétraction fonctionnent uniquement sur l'espace de recherche
SPEA [ZIT99]	Rangement basé sur l'archive des solutions non-dominées	Clustering pour tronquer la population	Oui	Oui	Bien testé Pas de paramètres pour le clustering	Algorithme de clustering compliqué
SPEA-2 [ZIT01]	Force de domination	Densité basée sur le k plus proche voisin	Oui	Oui	SPEA évolué Assure la préservation des points extremum	Fitness et densité couteuses en temps de calculs
RDGA [LUY03a]	Le problème est réduit en un problème bi-objectif (le rang de solution et la densité)	Densité à base de régions cellules taboues	Oui	Oui	Mise à jour de cellules dynamique Robuste par rapport au nombre d'objectifs	Plus difficile à implémenter que les autres

6. Particularités et difficultés des Algorithmes Génétiques Multiobjectifs : [BER01]

Dans cette section, nous présentons tout d'abord les difficultés rencontrées par le processus d'optimisation des AGMs, puis nous discutons de la difficulté de mise en œuvre de ces méthodes. Nous mettrons l'accent sur les points suivants : le nombre de paramètres et le critère d'arrêt de l'algorithme.

6.1 Difficultés des méthodes Algorithmes Génétiques Multiobjectifs :

Come nous l'avons déjà énoncé dans la section 5.3, un processus d'optimisation multiobjectif doit résoudre les deux tâches suivantes :

- guider le processus de recherche vers la frontière de Pareto,
- maintenir une diversité des solutions pour assurer une bonne répartition sur la frontière de Pareto. (voir figure 1-3 section 5.3)

L'accomplissement de ces tâches est très délicat car les difficultés rencontrées dans un problème multiobjectif sont identiques à celles d'un problème simple objectif mais elles sont amplifiées par la présence d'objectifs dépendants les uns des autres.

6.1.1 Guider le processus de recherche vers la frontière de Pareto :

Le processus de recherche est souvent ralenti ou totalement dérivé par des fonctions possédant une des caractéristiques suivantes : multimodalité, isolation d'un optimum ou tromperie.

a) La multimodalité

Une fonction est dite multimodale si elle possède plusieurs d'optima globaux (Figure 1-9). Dès lors, chaque optimum exerce sur les individus d'une population une attraction différente qui peut piéger le processus de convergence de l'algorithme. Ce problème peut être évité en utilisant une technique de répartition des individus de type sharing ou crowding [MAH95].



Fig.1-9 Une fonction multimodale [BER01]

b) L'isolation d'un optimum

Il existe des problèmes dans lesquels un optimum peut être entouré de grandes zones pratiquement plates. Cet optimum se trouve alors isolé car l'espace de recherche qui l'entoure ne peut pas guider vers lui les individus de la population.

Pour toutes les méthodes présentées ci-dessus, il est très difficile de garantir au décideur que ce type d'optimum peut être trouvé. Par contre les méthodes utilisant une population externe comme archive [BER01] semblent plus aptes à maintenir un optimum isolé que celles qui n'en utilisent pas.

c) La tromperie

Un problème est trompeur (Figure 1-10) lorsqu'il guide la convergence vers une zone non optimale de la fonction.

Dans les deux problèmes ci-dessus, nous voyons que la probabilité de choisir aléatoirement un point dans une zone sous-optimale est très grande. Donc, dans un premier temps, les méthodes ont une tendance à converger vers des optima locaux. Par la suite, le processus de mutation entretient ce phénomène de tromperie.

Pour éviter cette tromperie, Deb et Goldberg recommandent l'utilisation de techniques de répartition des individus en niches [GOL92]. Ils établissent également que le choix d'une taille appropriée de la population est primordial pour éviter la tromperie.

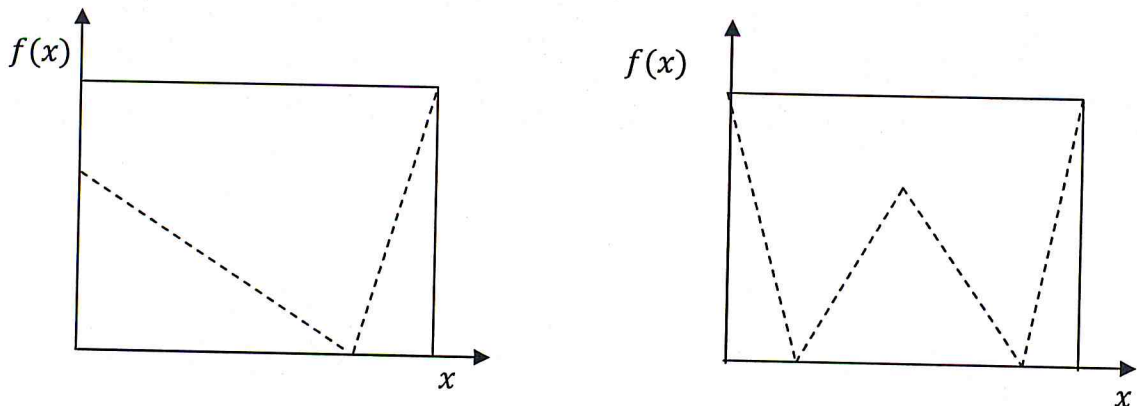


Fig.1-10 Problèmes trempés [BER01]

6.1.2 Maintenir la diversité sur la frontière de Pareto : [BER01]

La difficulté à maintenir une bonne répartition des solutions sur la frontière de Pareto résulte principalement des caractéristiques suivantes : convexité ou non convexité de la frontière de Pareto, discontinuité de cette frontière et non uniformité de la distribution.

c) Convexité et non convexité

Certains problèmes ont une frontière de Pareto non convexe. Les méthodes dont le calcul de la fitness est basé sur le nombre d'individus dominés (MOGA, SPEA) vont être moins efficaces.

d) Discontinuité

Si une frontière de Pareto est discontinue, nous retrouvons le même principe que pour une fonction multimodale. Les différentes parties de cette frontière vont exercer proportionnellement à leur taille, une attraction plus ou moins importante sur les individus d'une population. Certaines d'entre elles pourront donc ne pas être découvertes.

Les méthodes basées sur une population génétique sont plus sensibles à ce phénomène que les méthodes utilisant des stratégies d'évolution.

e) La non uniformité de répartition sur la frontière

Les solutions sur la frontière de Pareto peuvent ne pas être réparties uniformément. La raison principale vient du choix des fonctions objectifs. Par exemple, si une des fonctions objectifs est multimodale, elle va influencer de manière très différente la répartition des solutions sur la frontière de Pareto.

6.2 La mise en œuvre [BER01] :

6.2.1 Le paramétrage :

La mise en œuvre de la plupart des méthodes d'optimisation multiobjectifs présentées ci-dessus exige en priorité de maîtriser le fonctionnement d'un algorithme génétique (codage, fonction de sélection, opérateurs de croisement et de mutation, heuristique de partage (éventuellement), critère d'arrêt...ect) et de comprendre l'interaction de ses différents paramètres.

Si la compréhension des interactions entre paramètres peut être aisément appréhendée sur des exemples simples, elle reste une question délicate pour les nouveaux venus dans ce domaine. Car déterminer la taille de la population, le taux de croisement et le taux de mutation n'est pas une chose aisée. Si en plus, on ajoute une heuristique de partage, il faut également prendre en compte les paramètres de définition du voisinage.

Les méthodes basées sur une stratégie d'évolution sont algorithmiquement plus simples. Mais l'usage systématique d'une population externe pour maintenir les meilleurs individus exige la détermination de stratégies de mise à jour et de réutilisation basées sur une heuristique de remplacement géographique qui suppose la définition d'un maillage de l'espace de recherche.

6.2.2 La définition d'un critère d'arrêt

La définition d'un critère d'arrêt est une difficulté supplémentaire. Le critère le plus usité pour stopper la recherche d'un algorithme génétique est la perte de diversité génétique. Or celui-ci n'a plus de sens dans des méthodes qui maintiennent la diversité génétique de la population pour permettre une recherche plus efficace. Peu d'auteurs expriment clairement leur critère d'arrêt de l'algorithme. Mais en pratique, cela n'est pas critiquable car l'utilisation de ces méthodes par un décideur pourra se faire de manière interactive.

Dans la méthode Memetic PAES [BER01], basée sur une stratégie d'évolution, les auteurs définissent un critère d'arrêt basé sur le nombre d'échecs consécutifs de la recherche locale. Dans ce cas il est difficile pour un décideur d'établir une relation de cause à effet entre le nombre d'échecs et la qualité du résultat.



Les algorithmes génétiques sont très largement utilisés dans plusieurs domaines grâce à leur robustesse et leur grande flexibilité.

Le domaine de l'optimisation multiobjectif est l'un des domaines d'application des AGs les plus récents. Les AGMs sont l'adaptation des AGs classiques aux problèmes multiobjectifs. Ces derniers doivent assurer deux propriétés importantes : La convergence vers la frontière de Pareto et la diversité des solutions pour assurer une bonne répartition sur la frontière de Pareto. Pour cela, ces algorithmes exploitent de nouvelles notions et techniques avancées tels que la dominance de Pareto, le niching...etc.

Plusieurs variantes des AGMs sont proposées selon le type de problèmes traités. Elles sont très utiles en présence de critères nombreux, ou d'un grand nombre de solutions possibles, et/ou de critères contradictoires ou dépendants. Cependant les AGMs présentent des difficultés supplémentaires tels que : le paramétrage et la mise en œuvre.

On présentera dans le chapitre suivant un problème d'optimisation multiobjectif très important en économie : le problème de sélection des entreprises pour l'octroi de marchés, ce problème fera l'objet de notre application des AGMs.

CHAPITRE 2

**Sélection des fournisseurs dans le contexte
d'appels d'offres :**

Critères, méthodes et risques opérationnels

CHAPITRE 2

Sélection des fournisseurs dans le contexte d'appels d'offres :

Critères, méthodes et risques opérationnels

Introduction :

Face à des marchés fortement compétitifs, caractérisés par une demande de produits personnalisés, de bonne qualité, livrés dans des délais minimaux et le tout au moindre coût, les entreprises d'aujourd'hui réalisent qu'une gestion efficace de leurs achats locaux et/ou internationaux peut constituer un avantage concurrentiel substantiel.

La sélection des fournisseurs devient ainsi une décision stratégique qui a un impact crucial sur la performance globale de toute entreprise [AGU06].

Ce chapitre est un bref survol sur le champ du problème de sélection de fournisseurs dans le contexte d'appel d'offre. Ce problème à grande importance économique, est au carrefour de plusieurs disciplines telles que les sciences économiques, le management, la recherche opérationnelle ou encore l'aide à la décision.

Vus sa nature multidisciplinaire, ce problème bénéficie de plusieurs définitions et diverses stratégies de traitement. Nous s'intéressons bien sûr dans cette étude aux aspects communs à notre domaine tel que l'optimisation multiobjective et l'aide à la décision.

Ce chapitre constitue, d'un autre coté, une très brève synthèse de recherche bibliographique concernant le problème de sélection de fournisseurs dans le contexte d'appel d'offre. Nous définissons ce problème, ensuite nous passons aux différentes étapes qui le constituent. On présentera par la suite les risques opérationnels liés aux différentes phases de ce processus. Ceci nous permettra de cerner enfin notre problématique et de la restreindre à une phase bien déterminée.

1. Le problème de sélection de fournisseurs :

L'octroi d'un marché de réalisation est conditionné par la déposition d'une offre de soumission de la part des entreprises de réalisation intéressées par le projet.

Chaque offre de soumission est composée d'une offre technique qui représente la capacité technique de l'entreprise à réaliser le projet, alors que l'offre financière représente le prix que propose l'entreprise pour la réalisation du projet. Le problème qui se pose est de sélectionner le fournisseur ayant le meilleur compromis entre l'offre technique et l'offre financière.

Le problème de sélection de fournisseurs a été largement étudié dans la littérature scientifique. Gary Dickson [DIC66] est un des pionniers à s'intéresser à ce problème. Il a défini le problème de sélection de fournisseurs comme suit :

Définition 1 : Problème de sélection de fournisseurs

"The vendor selection problem is associated with deciding how one vendor should be selected from a number of potential alternatives." [DIC66]

Cette définition peut être réécrite en français de la manière suivante :

Définition 2 : Problème de sélection de fournisseurs

« Le problème de sélection de fournisseurs consiste à décider de la manière avec laquelle un ou plusieurs fournisseurs sont sélectionnés à partir d'un certain nombre d'alternatives et ce pour la réalisation d'une activité ou la fourniture d'un produit. » [HAM03]

Le problème de sélection de fournisseurs est complexe en raison du grand nombre de combinaisons à considérer, du caractère multicritère et multiobjectifs du problème à résoudre et du caractère évolutif des caractéristiques de chaque fournisseur. Ces caractéristiques, et essentiellement les deux premières, nous ont incités à explorer la possibilité de résoudre ce problème très célèbre par les AGs multiobjectifs et d'en déduire éventuellement les degrés d'adaptabilité de ces derniers (en cas où ils sont applicables bien sûr).

La recherche bibliographique que nous avons faite à un stade précoce, nous donne déjà une idée très globale : ce problème est très rarement étudié d'une manière intégrale en une

seule étude, en parle dans ce cas d'une « méthodologie » de résolution de ce problème. A l'heure actuel, la recherche de ces méthodologies reste un domaine de recherche très fertile, néanmoins aucune « méthodologie » n'a été standardisée ou connue comme étant la plus efficace.

Si ce problème n'est pas facilement franchi de bout en bout c'est parce qu'il n'est pas si simple comme la définition de Dickson nous fait accroire. En effet, il s'avère que ce problème est en faite un cumul de sous-problèmes. Chaque sous-problème correspond à une phase ou une étape du processus globale de sélection.

Notre objectif dans cette étude n'est pas de résoudre le problème de sélection de fournisseur mais plutôt de chercher un des sous-problèmes de ce dernier où on peut appliquer les AGMs. Autrement dit : On tente de trouver au sein du processus de sélection un sous-problème qui nous sert d'application des AGMs.

Dans la section suivante, on présente une vue globale du processus sélection de fournisseur dans le contexte d'un appel d'offre. On verra que chaque étape présente des risques qui ont des impacts très sérieux sur le résultat de la sélection. C'est l'un de ces risques qui jouera le rôle de base pratique pour le problème d'application de nos AGMs.

2. Processus de sélection de fournisseurs dans le contexte d'un appel d'offre :

L'appel d'offre (AO) est un moyen efficace pour l'acheteur vers la réduction des charges de l'entreprise. L'atteinte de cet objectif passe nécessairement par la mise en œuvre de certaines actions telles que la comparaison objective et méthodique des offres, la connaissance du marché en vue d'atteindre les résultats escomptés (telles que la sélection de fournisseurs performants, l'obtention de prix bas) [CAV07].

2.1 Étapes du processus d'appel d'offres [NIG08]:

Les auteurs ci après ont proposé les différentes étapes du processus d'appel d'offres illustrées comme suit:

Tab.2-1 : synthèse des étapes du processus d'AO [NIG08]

Auteurs	Caverivière [CAV07]	Bouvier [BOU90]	Loubere [LOU97]	Blanchard &al [BLA99]
Etapes				
1. Présélection des fournisseurs	X			X
2. Réalisation du dossier d'AO	X	X	X	X
3. Envoi des dossiers d'AO/ Publication d'un avis d'appel d'offres	X			X
4. Réception et conservation des offres		X		X
5. Dépouillement des offres et attribution du marché à un fournisseur	X	X	X	X

Ainsi, la synthèse de ce tableau donne cinq étapes principales du processus d'AO qui sont: la présélection des fournisseurs ou prestataires, la réalisation du dossier d'AO, l'envoi des dossiers d'AO/ publication d'un avis d'appel d'offres, la réception et la conservation des offres, le dépouillement des offres et l'attribution du marché à un fournisseur.

a) Présélection des fournisseurs ou prestataires

Cette étape englobe les opérations de consultation des fournisseurs potentiels et homologués, l'analyse (ou revue) de la liste des fournisseurs anciens et nouveaux, le repérage des fournisseurs susceptibles de couvrir l'achat à réaliser, la définition des critères de présélection. Ces opérations sont effectuées par l'acheteur. Selon Bouvier [BOU90], l'acheteur peut recevoir aussi la proposition des demandeurs quand aux meilleurs fournisseurs possibles du marché sans pour autant se laisser influencer par la décision de ceux-ci.

Pour Blanchard & al [BLA99], la présélection des fournisseurs est une pré-consultation ayant pour avantage d'une part de consulter l'ensemble des fournisseurs potentiels du marché et d'autre part, de limiter le nombre de fournisseurs auxquels le dossier d'AO (DAO) sera envoyé. Elle est rarement utilisée par les acheteurs, affirment Blanchard & al [BLA99]. C'est la première classification des fournisseurs possibles du marché dont le DAO sera destiné [BRU98]. On dit qu'elle est utilisée lorsque l'appel d'offres est restreint au mieux disant ou au meilleur prix.

b) Réalisation du dossier ou document d'appel d'offres

Le dossier d'AO (DAO) peut être la pièce maîtresse du processus d'AO. Sa réalisation englobe toutes les opérations de recherche des informations et d'élaboration du dossier d'appel d'offres et sont effectuées par l'acheteur. La plupart des informations recherchées proviennent des services demandeurs à travers une demande d'achat faite par ces derniers [CAV07], [BLA99]

Selon Blanchard & al [BLA99], le DAO comporte: l'avis d'AO, la présentation de l'organisation qui émet le DAO, la description sommaire de l'objet de l'appel d'offres, le rappel des grands objectifs de l'organisation dans le cadre de l'achat à effectuer, les critères de choix afin que le fournisseur puisse bien orienter sa réponse, la définition précise et claire de l'adresse du destinataire, la date limite de réception des offres, la validité de l'offre, l'existence de l'organigramme de l'organisation, la description des compétences de l'acheteur, le planning de déroulement du marché, le cahier des charges, la confidentialité de l'avis d'AO pour interdire qu'il y ait sous-traitance de la part d'un fournisseur.

c) Envoi du dossier d'appel d'offres

Cette étape regroupe toutes les opérations d'envoi du DAO aux fournisseurs présélectionnés dans le panel ou liste. Pour Caverivière [CAV07], c'est l'acheteur qui s'en occupe. Mais pour Bouvier [BOU90], une personne différente de l'acheteur et donc impartiale doit être chargée des opérations d'envoi du DAO. Selon Cova & al [SIM97], les opérations d'envoi du DAO aux fournisseurs présélectionnés sont effectuées pour un appel d'offres restreint au mieux disant ou au meilleur prix. En ce qui concerne l'appel d'offres ouvert au mieux disant ou au meilleur prix, l'acheteur fait le plus de publicité possible du DAO ou sa publication dans un journal ou un site internet; les fournisseurs intéressés par l'avis d'AO passeront faire le retrait du DAO dans l'entreprise [SIM97].

d) Réception et conservation des offres

Selon Bouvier [BOU90], elle concerne les opérations de nomination d'un responsable, de préparation d'un bureau fermé non accessible à l'acheteur ainsi qu'à une autre personne, de réception et d'enregistrement des offres. Elle couvre aussi la conservation des offres jusqu'au jour de dépouillement. Il affirme que cette tâche est souvent confiée à la secrétaire du service achats et qu'elle peut aussi être confiée à une autre personne. Blanchard & al [BLA99] disent que, la réception des offres des fournisseurs doit se faire au jour et à l'heure indiquées dans le

DAO. Ils affirment que des restrictions peuvent être accordées aux fournisseurs qui effectueront les offres en retard si l'acheteur les considère pertinentes [BLA99].

e) Comparaison des offres ou dépouillement et attribution du marché

Elle englobe les opérations d'élaboration d'une grille de dépouillement des offres, de détermination du coefficient de pondération ou de cotation des critères définis dans le DAO. Elle couvre aussi les opérations d'ouverture des offres, d'évaluation des propositions techniques et commerciales et de mise au point technique des offres. Cette étape est souvent effectuée par une commission composée de l'acheteur, du gestionnaire des offres, du responsable du service achats. Contrairement à Bruel [BRU98], qui dit qu'elle doit être effectuée par un ou plusieurs employés nouveaux en vue de faire un choix objectif et non la commission ci-dessus citée. En général, les offres sont comparées à une offre dite « de référence » établie par le maître d'ouvrage lui-même.

Cette section nous a permis d'avoir des pré-requis sur le processus d'AO pour pouvoir identifier les risques opérationnels qui y sont liés dans le but, de repérer une problématique qui nous sert de validation pour l'AGM. La section qui suit récapitule les différentes problématiques liées au processus d'achat par appel d'offres.

2.2 Identification des problématiques spécifiques au processus d'appel d'offres : [NIG08]

Différents travaux dans la littérature ont étudié les risques opérationnels qui y sont liés au processus d'appel d'offres. Le tableau suivant récapitule les problématiques les plus importantes suivant les étapes du processus complet :

Tab.2-2 : synthèse des étapes du processus d'AO [NIG08]

Risques Etapes	Risques opérationnels encourus
<i>Présélection des fournisseurs</i>	<ul style="list-style-type: none"> · mauvaise supervision des différentes étapes de l'AO, absence et réexamen périodique de fichier fournisseurs, · répartition inadéquate des tâches dans le processus d'AO, · fixation du prix par rapport à celui antérieurement établi

	<p>sans référence au prix du marché</p> <ul style="list-style-type: none"> · absence de challenge entre les fournisseurs existants et nouveaux pouvant apporter des nouvelles solutions et conditions tarifaires, · absence ou mauvaise définition des critères de présélection des fournisseurs, · choix de fournisseurs incomparables, collusion avec des fournisseurs récurrents, · présélection de fournisseurs inappropriés, absence de connaissance des fournisseurs clés,
<i>Réalisation du dossier d'appel d'offres</i>	<ul style="list-style-type: none"> · participation de l'un des fournisseurs présélectionnés à l'élaboration du DAO, · rédaction orientée du cahier des charges techniques pour qu'un seul fournisseur soit sélectionné, · inexistence de certains éléments importants dans le DAO (délai de réponse du fournisseur par rapport à la demande, adresse du lieu de remise de l'offre), · incompétence de l'acheteur à réaliser le DAO, · collusion entre l'acheteur et le fournisseur au détriment de l'intérêt de l'entreprise,
<i>envoi du DAO ou publication du DAO</i>	<ul style="list-style-type: none"> · réduction du délai de publicité et de dépôt des offres afin de favoriser un fournisseur informé du contenu du DAO à être le seul en mesure de répondre à l'AO, · diffusion d'informations incomplètes ou imprécises du DAO à un fournisseur de façon à le privilégier, · absence de publicité, publication tardive du DAO et frustration du fournisseur, · modification des dates limites de remise des offres, · absence de liste émarginée des fournisseurs ayant procédé au retrait du DAO,
<i>Réception et conservation des offres</i>	<ul style="list-style-type: none"> · inexistence de bureau de réception des offres, · inexistence de responsable chargé de la réception, · confusion des offres avec les autres courriers de l'entreprise, · absence de liste émarginée par les soumissionnaires au dépôt

<i>Dépouillement des offres et attribution du marché au fournisseur</i>	des offres, · non respect de la procédure de comparaison des offres, · absence de critères de comparaison des offres (comme le prix, les conditions de paiement, le transport), · mauvaise pondération des éléments techniques du dossier d'appel d'offres (DAO), · risques liés au dépouillement des offres pour éliminer arbitrairement un ou des fournisseurs, · absence de comité de dépouillement, · non sélection des fournisseurs offrant le meilleur ratio qualité/prix, · non prise en compte de tous les paramètres lors de la sélection des fournisseurs, · absence de listes de fournisseurs interdits, absence de publication de l'attribution du marché, · non maîtrise du DAO par les membres de la commission de dépouillement,
---	---

Discussion :

L'étude récapitulée dans le tableau ci-dessus, met en évidence un tas de problématiques très importantes. Parmi ces dernières, nous avons choisi le problème de **pondération des critères** pour faire l'objet de problème d'application pour l'AGMs (que nous proposons dans le chapitre suivant). Plus de détails sur ce problème seront présentés dans la section suivante.

2.3 La problématique de pondération des critères :

Dans le domaine de la réalisation de projet, la concurrence est sévère en terme de coût, souvent la logique du moins disant l'emporte ; de ce fait l'octroi du marché de réalisation est souvent attribuée à l'entreprise qui propose le prix le plus bas.

L'offre financière d'une entreprise peut être déterminante dans l'octroi du marché de réalisation si le projet ne présente pas de difficultés techniques majeures.

Par contre, si le projet présente des difficultés techniques, alors la sélection de la meilleure offre composée de l'offre technique et de l'offre financière de l'entreprise devient très complexe du fait de la multitude de critères à considérer, ainsi que le caractère concurrentiel de l'offre financière.

La problématique de cette sélection qui va permettre la passation de contrat de réalisation peut être résumée en trois points qui sont :

- L'évaluation des capacités techniques.
- L'évaluation du coût proposé par chaque entreprise.
- La combinaison des deux évaluations afin d'éviter à une seule évaluation globale qui est l'aptitude d'une entreprise à réaliser le projet.

2.3.1 Définition de la pondération des critères :

La pondération des critères signifie l'attribution à chacun des critères de choix d'un coefficient traduisant l'importance de ce critère. Ceci se traduit pratiquement par l'établissement d'une formule pondérée utilisée lors de la sélection.

Définition : formule de sélection

Soient $C_1, C_2 \dots, C_n$ les critères de sélection considérés; et soient $NC_1, NC_2 \dots NC_n$ les notes attribuées respectivement à ces critères.

Une formule pondérée de sélection est de la forme : $\sum_{i=1}^n p_i NC_i$.

p_i est appelé le poids du critère i . tel que :

$$\left\{ \begin{array}{l} \sum_{i=1}^n p_i = 1 \\ p_i > 0, \forall i = 0 \dots n \end{array} \right.$$

Exemple de formule de sélection :

Si les critères de sélection sont C_1, C_2 et C_3 ; et NC_1, NC_2 et NC_3 les notes attribuées respectivement à ces critères.

Une formule de sélection possible serait :

$$0.2*NC_1 + 0.3* NC_2 + 0.5*NC_3$$

2.3.2 L'importance du risque de la mauvaise pondération :

La mauvaise pondération des éléments techniques du dossier d'appel d'offres est considérée comme étant un risque très implorant pour deux raisons primordiales :

1. *L'exigence de pondération* : La pondération des critères de choix dans un contexte d'appel d'offre a été rendue obligatoire par le nouveau code des marchés publics 2006.
2. *L'inexistence de méthode pour la pondération* : la pondération de critères de sélection est un élément majeur au cœur du processus de décision qui conduit à l'attribution du marché. Paradoxalement, aucune indication n'est fournie par le code des marchés publics concernant la mise en œuvre de la pondération, et il revient à chaque maître d'ouvrage de définir la méthode. En générale, cette pondération se fait soit par une réunion des experts du domaine, soit par similitude à des situations préalables. Et dans les deux cas, l'expertise (donc risque de subjectivité) est un facteur très élémentaire et décisif.

3. Vue décisionnelle du processus de sélection de fournisseurs :

Dans la section précédente, on a repéré la problématique de pondération de critère. On envisage à ce stade d'appliquer les AGMs sur le problème mathématique correspondant.

Si nous décidons maintenant de voir le problème de sélection de fournisseur d'un point de vue d'aide à la décision, le processus de sélection suivra les étapes communes à tous les problèmes d'aide à la décision proposées par Guitouni [GUI98]. Ce derniers stipule que le processus d'aide multicritère à la décision peut généralement être vu comme un processus récursif (itératif), non linéaire, composé de cinq étapes principales :

- (1) La structuration de la situation (problème) de décision.
- (2) L'articulation et la modélisation des préférences au niveau de chaque point de vue (modélisation des préférences locales).
- (3) L'agrégation de ces préférences locales en vue d'établir un ou plusieurs systèmes relationnels de préférences globaux.
- (4) L'exploitation de cette agrégation.
- (5) La recommandation.

Le schéma suivant illustre ces étapes :

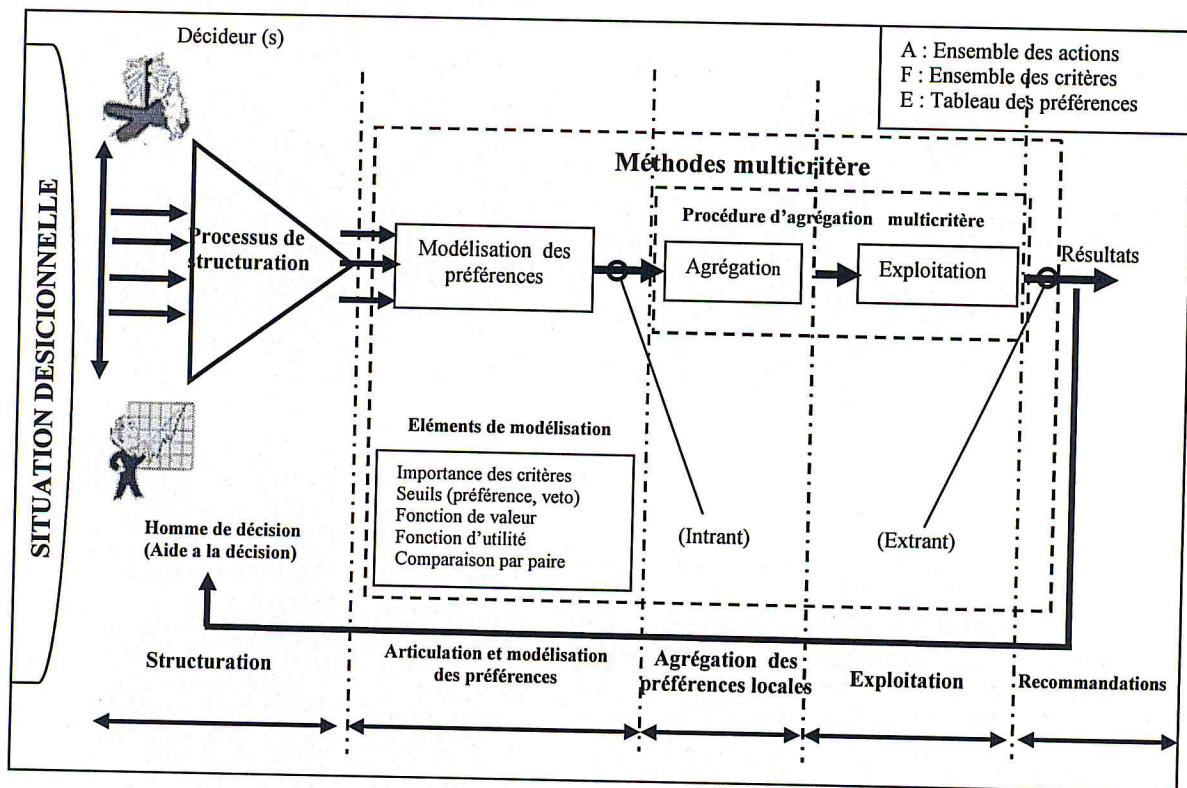


Fig. 3-1 Représentation du processus d'aide à la décision [GUI99]

Chacune de ces étapes, est elle seule, un axe de recherche très captivant qui fait l'objet de plusieurs études. Evidemment, le champ d'étude de chaque étape varie selon la nature de cette dernière. La phase qui nous intéresse dans notre étude, la pondération des critères, coïncide avec la troisième étape du processus décisionnel: l'évaluation des fournisseurs par rapport aux critères retenus.

Du point de vu décisionnel, cette phase est très fortement liée à celle qui la précède. En effet, on ne peut étudier l'évaluation des fournisseurs sans passer par la phase de détermination des critères de sélection. Pour cela, nous présenterons par la suite une brève étude de la littérature sur le problème du choix des fournisseurs et, plus précisément, sur les critères et les méthodes de sélection des fournisseurs.

La section suivante présente une brève étude de la littérature sur le problème du choix des fournisseurs et, plus précisément, sur les critères et les méthodes de sélection des fournisseurs.

4. Critères de sélection des fournisseurs :

Plusieurs études de type empirique et exploratoire sur le problème de la sélection d'évaluation des fournisseurs sont apparues dans la littérature pour montrer l'importance stratégique de ce problème. Les premiers écrits dans ce domaine sont ceux de Dickson [DIC66] qui, à partir d'une enquête réalisée auprès de 273 industriels, a pu identifié plus de 50 critères utilisés par les entreprises pour sélectionner leurs fournisseurs. Ces critères ont été traités par Dickson et regroupés en 23 critères seulement. Selon Dickson, les 7 critères les plus usités sont respectivement : la qualité, le délai, l'historique de la performance, la politique de réclamation, la capacité, le prix, et la capabilité technique.

L'étude a montré que le choix des fournisseurs est une décision multicritères qui implique souvent la considération simultanée des critères su-cités, et qu'il est extrêmement difficile de trouver un fournisseur qui excelle partout.

Une synthèse de 74 articles de la littérature scientifique présentée dans [WEB91] a montré que les 23 critères recensés par Dickson en 1966, sont quasiment les mêmes que ceux utilisés aujourd'hui pour traiter le problème de sélection de fournisseurs. Bien que l'importance relative à chacun des critères ait changé suite au changement du contexte industriel. Selon [WEB91], les 7 critères les plus cités sont respectivement: le prix, le délai, la qualité, la capacité, la localisation géographique, la capabilité technique et l'organisation.

Le tableau 2-3 ci-dessous montre les deux types de classification du degré d'importance des critères relatifs au choix des fournisseurs selon Dickson et Weber :

- Une classification pratique basée sur l'étude empirique réalisée auprès des entreprises de NAPM [DIC66] ;
- Une classification théorique basée sur l'analyse de s travaux de recherche publiés entre 1966 et 1990 [WEB91].

Tab.2-3 : Critères de sélection des fournisseurs et leurs poids selon Dickson et Weber [AGU06]

Critères	Rang selon Dickson	Rang selon Weber
Prix	6	1
Livraison	2	2
Qualité	1	3
Capacité de production	5	4

Localisation géographique	20	5
Capacité technique	7	6
Gestion et organisation	13	7
Réputation et position dans l'industrie	11	8
Situation financière	8	9
Performance Passée	3	9
Service de réparation	15	9
Attitude	16	10
Habilité d'emballage	18	11
Contrôle des opérations	14	11
Formation et support	22	12
Conformité des processus	9	12
Relations sociales	19	12
Système de communication	10	12
réciprocité des relations	23	12
Impression	17	12
Désir de faire des affaires	12	13
Volumes des achats dans le passé	21	13
Politique de garantie	4	14

Nous pouvons constater de ce tableau que la classification des critères n'est pas générique. L'ordre de l'importance du critère dépend du niveau d'exigence du maître d'ouvrage, de son secteur d'activité, de la criticité du produit et du type de relation qu'il projette de créer avec le fournisseur : le changement de contexte industriel a modifié les degrés d'importance relative de ces critères.

Autres les méthodes empiriques, citées ci-dessus, ils existent des méthodes d'évaluation très célèbres. Ces dernières sont présentées dans ce qui suit.

5. Méthodes d'évaluation des fournisseurs [AGU06] :

Aicha Aguezzoul et Pierre Ladet ont présenté dans leur article [AGU06] une classification des méthodes de sélection et d'évaluation des fournisseurs faite à base de leur

analyse de 36 articles publiés dans les revues spécialisées dans les domaines des achats et de la gestion de la chaîne logistique ainsi que de l'étude de Weber et al. [WEB91]. Cette classification met en évidence *six* catégories de méthodes :

1. Modèles linéaires de pondération :

En attribuant un poids à chaque critère défini de manière *subjective*, un score est calculé pour chaque fournisseur en effectuant la somme des performances des fournisseurs relatives à chaque critère multipliée par le facteur de pondération associé. Cette méthode repose également sur le jugement et l'expérience de l'acheteur pour affecter les pondérations. Timmerman [TIM86] est parmi les premiers auteurs ayant proposé cette méthode dans ce domaine. Nous classons dans cette catégorie, les méthodes suivantes :

- **AHP (Analytic Hierarchy Process)** [NAR83] [NYD92] [MAS00]: C'est un processus qui se distingue par sa façon de déterminer les poids des critères par combinaisons binaires de chaque niveau de la hiérarchie par rapport aux éléments du niveau supérieur. Le premier niveau de cette hiérarchie contient les critères principaux, suivi au deuxième niveau par les critères secondaires associés à chacun des critères principaux et ainsi de suite. Le dernier niveau donne l'arrangement relatif des fournisseurs potentiels ;
- **FST (Fuzzy Sets Theory)** [KUM04] : La théorie des ensembles permet de modéliser l'incertitude et l'imprécision relatives aux valeurs des poids attribués aux critères.

2. Modèles de programmation mathématique (MP : Mathematical Programming)

Ils représentent une fonction objectif à optimiser (minimiser ou maximiser) et peuvent inclure des contraintes sur les fournisseurs, sur le donneur d'offre, etc. Les plus utilisés sont : La programmation entière linéaire/non linéaire [HON92], La programmation linéaire/non linéaire à variables entières mixtes [CHA93] [MUR04], MOP (Multi-Objectifs Programming) ou programmation multiobjectifs [WEB93] [WEB00] [DAH03] [KUM04], DEA (Data Envelopment Analysis) [WEB96] [WEB00] [LIU00]...

3. Méthodes basées sur le coût total :

Ce sont des techniques assez complexes qui nécessitent l'identification et le calcul des coûts générés par les différentes activités intervenant dans l'opération d'achat telles que le contrôle de la qualité des produits, le transport, les frais administratifs, etc. Nous classons dans cette catégorie les modèles suivants :

- **ABC (Activity Based Costing) ou loi de Pareto ou méthode 80/20 [ROO97]** : Dans le cas de la sélection des fournisseurs, il s'agit de classer et par ordre décroissant, les achats réalisés auprès des fournisseurs en trois catégories : les 20% des fournisseurs (classe A) représentent 80% de la valeur des achats, les 30 % suivants (classe B) correspondent à 15% de la valeur des achats et les 50% (classe C) se partagent les derniers 5 % .
- **TCO (Total Cost of Ownership) [SMY93] [ELL95]** : C'est la méthode avancée de la méthode du ratio de coûts (Cost-Ratio) et qui nécessite le calcul du coût total d'acquisition d'un produit et qui inclut le prix d'acquisition et tous les coûts opérationnels sous jacents tels que la qualité, l'inspection, la livraison, etc.

4. Modèles statistiques/probabilistes :

Diverses approches sont suggérées dans la littérature, on distingue :Payoff Matrix [SOU87], VPA (Vendor Profile Analysis) [ELL00], FA (Factor Analysis) ou analyse factorielle [TRA01], ISM (Interpretive Structural Modeling) [MAN94], CA (Cluster Analysis) [HIN69].

5. Méthodes de catégorisation :

Elles permettent de regrouper les fournisseurs en fonction de leur positionnement stratégique et de la criticité du produit en des catégories homogènes afin de mieux gérer le réseau des fournisseurs. Des travaux de nature exploratoire de D'Amours et al. [DAM01] et qui sont basés sur un benchmark des pratiques des entreprises canadiennes ont permis d'identifier cinq catégories de fournisseurs : partenaires, stratégiques, niches, préférés ou généraux.

6. Intelligence Artificielle :

Contrairement aux approches quantitatives présentées auparavant, les outils de l'intelligence artificielle visent à intégrer les facteurs qualitatifs et l'expertise humaine dans le processus de sélection des fournisseurs. On distingue dans cette catégorie les systèmes suivants :

- **ES (Expert System) [VOK96]**: Les systèmes experts sont pour représenter les connaissances et l'expertise que détiennent les professionnels des achats sur les fournisseurs ainsi que les informations recueillies de la littérature sur les différentes étapes du choix et de l'évaluation de s fournisseurs tels que la formulation des critères de sélection, etc. ;
- **CBR (Case-Based-Reasoning system)** : C'est une approche qui utilise les connaissances déduites des expériences similaires ou antérieures sur les fournisseurs afin de prendre les décisions sur leur pré-qualification. La préqualification consiste à réduire le nombre des fournisseurs en un ensemble de fournisseurs acceptables ou approuvés [SKI95].

Dans leur étude sur les différentes méthodes de sélection des fournisseurs publiées dans la littérature, [WEB91] [DEB01] ont montré que ces méthodes varient selon les quatre étapes du processus de sélection des fournisseurs et du type de l'achat à réaliser. Un positionnement approximatif de ces méthodes ainsi que leur répartition selon le type d'achat sont respectivement donnés dans la figure 2. 2 et le tableau 2-4 suivants :

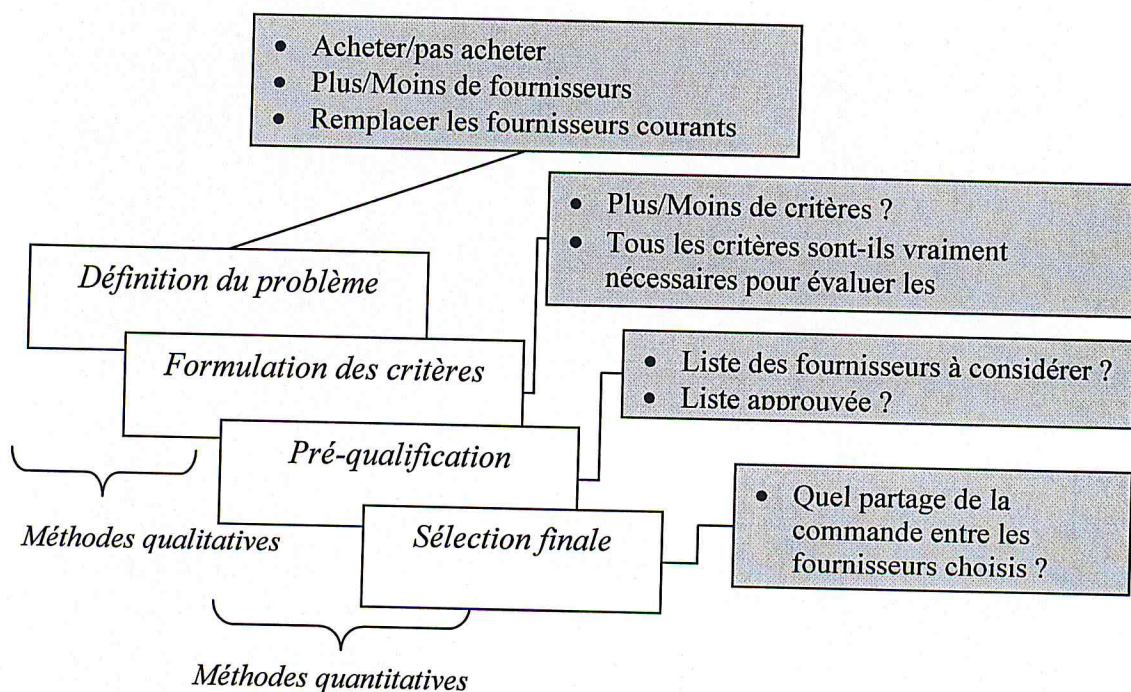
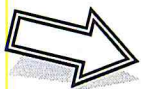


Fig. 2-2 Positionnement approximatif des méthodes de sélection de fournisseurs [AGU09]

Tab.2-4 : Critères de sélection des fournisseurs et leurs poids selon Dickson et Weber [AGU09]

Type d'achats Etapes de sélection	Achat de nouveau produit	Achat modifié (produit influent)	Achat urgent	Achat urgent (stratégique/goulot d'étranglement)
Définition du problème				
Formulation des critères			ISM	ISM, ES
Pré-qualification des fournisseurs	Catégorisation	AHP, CA, DEA, ES, CRB, UT, Catégorisation	AHP, CA, DEA, Catégorisation	Catégorisation
Sélection finale	AHP, ES, CA, FST, UT	AHP, CA, DEA, GP, TCO, MP	AHP, CA, MP, UT	ABC, AHP, CA, FST, UT

Notons que dans cette étude, aucune publication traitant l'étape de définition du problème de sélection des fournisseurs n'a été identifiée.



Nous avons présenté dans ce chapitre une vue globale du processus de sélection des entreprises dans le contexte d'appel d'offre. Nous avons également soulevé les problématiques concernant les différentes étapes de ce processus complexes. Parmi ces problématiques, nous avons retenu celle de pondération de critères. Cette dernière a fait l'objet de plusieurs études vu son importance. Les AGMs nous semblent très intéressants vis-à-vis cette problématique. Pour cette fin, on modélisera cette dernière d'une manière formelle dans le chapitre qui. La définition formelle multiobjectif de ce problème permettra l'élaboration d'un AGM correspondant.

CHAPITRE 3

Conception :

**Algorithme Génétique Multiobjectif pour le
problème de Sélection des Entreprises**

CHAPITRE 3

Conception : Algorithme Génétique Multiobjectif pour le problème de Sélection des Entreprises

Introduction :

Le but du travail actuel est d'appliquer les algorithmes génétiques multiobjectifs pour le problème de sélection des entreprises pour l'octroi de marchés.

Notre étude bibliographique, récapitulée dans les deux premiers chapitres, nous a permis de situer la phase d'application des AGs multiobjectif à l'intérieur des différentes étapes du processus d'aide à la décision du problème de sélection de fournisseurs : les AGs multiobjectifs sont à priori supposés être applicables à l'étape de modélisation des préférences, et plus précisément pour définir l'importance relative de chaque critère.

Ce chapitre est la présentation de notre approche d'application des AGs multiobjectif afin de déterminer les poids relatifs de chaque critère considéré dans l'appel d'offre. Le résultat envisagé est donc l'automatisation de la définition de la formule de sélection des entreprises qui est, comme déjà annoncée dans les chapitres précédents, une phase très délicate et décisive à la fois.

Nous présentons dans ce qui suit notre adaptation de la métaheuristique des AGs multiobjectifs. Cette adaptation sera précédée par une formulation multiobjectifs détaillée.

1 Formulation multiobjectif du problème de sélection d'entreprise:

Dans ce chapitre nous présentons notre formulation du problème de sélection des entreprises pour l'octroi de marchés. Nous avons opté pour une approche multiobjectif, c à d

que lors de la résolution de ce problème, nous prenons en considération plusieurs objectifs tels que : un prix minimal, une qualité maximale, une capacité de production maximale...ect (voir chapitre 2).

Il est important à noter aussi que dans le travail actuel, nous traitons le problème de sélections des entreprises de manière multiobjectif *sans aucune agrégation*. En effet, les méthodes d'agrégation, i.e. qui transforment le problème multiobjectif en un problème mono-objectif, ont un inconvénient majeur: chaque exécution de ces méthodes donne une seule solution Pareto¹ optimale ; pour avoir toutes les solutions Pareto optimales, il faudra exécuter l'algorithme autant de fois que le nombre d'objectifs du problème.

Les AGs multiobjectifs présentent l'avantage de traiter les problèmes multiobjectif sans aucune agrégation grâce à leur aspect de parallélisme intrinsèque.

1.1 Eléments de la formulation proposée :

La formulation que nous proposons dans ce travail est très rigoureuse du faite qu'elle s'appuie sur des notions mathématiques tels que : l'optimisation multiobjectif (cette notion détaillée dans la section qui suit intervient dans le cœur de la modélisation du problème lui même), la notion de dominance ainsi que le front de Pareto (section 2.2.3). Ces deux dernières sont à la base de la fonction de fitness utilisée dans le corps de l'algorithme proposé.

Nous définissons, selon le besoin, chacun de ces trois concepts d'une manière très simple et nous les illustrons par des exemples pratiques du problème considéré.

1.1.1 La sélection des entreprises et l'optimisation multiobjectif :

L'optimisation multiobjectif répond au besoin de satisfaire des besoins contradictoires. Ainsi pour attribuer le marché à une entreprise, on cherchera à minimiser le coût, à accélérer le temps de réalisation, et maximiser la qualité de réalisation.

Une grande entreprise perfectionniste sera très rapide en réalisation et offre des résultats très satisfaisant. Pour cela elle nécessite un effectif très élevé de travailleurs compétents, ce qui revient à un coût de réalisation assez élevé.

Une autre entreprise pourra réaliser le projet à très faible coût, mais elle ne pourra pas assurer une qualité et un délai de réalisation très satisfaisants.

¹ Une solution est optimale au sens de Pareto, si à partir de cette situation, on ne peut améliorer le sort d'un individu sans détériorer celui d'au moins un autre. (Pour plus de détails voir chapitre 1 section 6.2).

Comme il n'existe pas de solution meilleure qu'une autre en tout point, un compromis différent selon les experts du domaine doit être choisi. Le choix est donc subjectif, et il est indispensable de proposer l'ensemble des choix possibles afin ne pas exclure une possibilité.

L'optimisation multiobjectif est donc avant tout un outil d'aide à la décision. La décision finale revient bien sûr à l'utilisateur du système.

Par la suite, nous donnons la formulation multiobjectif retenue pour notre problème. Nous passerons avant cette formulation par les notations utilisées.

1.1.2 Définition et notations : *Problème d'optimisation multiobjectif*

Rappel : Un *problème d'optimisation multiobjectif* est défini par un triplet (X, F, g) qui consiste à minimiser $F(x)$ pour $x \in X$ sachant que $g(x) \leq 0$. [GRA08]

a) *Les variables de décision :*

Dans un problème d'optimisation, les m variables de décision sont les valeurs à choisir [GRA08]. Dans notre cas ces variables de décision sont les entreprises candidates pour l'octroi de marché.

On notera ces variable par x_i tel que $i \in \{1, \dots, m\}$.

Le vecteur X de m variables de décision est représenté par $X = (x_1, x_2, \dots, x_m)$.

b) *Les fonctions objectifs*

Les n *fonctions objectifs* à optimiser sont notées f_i , $i \in \{1, \dots, n\}$ et le vecteur f de n objectifs pour $x \in X$ est représenté par : $f(x) = (f_1(x), f_2(x), \dots, f_n(x))$ [GRA08].

Les fonctions objectifs considérées dans cette étude seront détaillées dans la section suivante.

Remarque :

Dans un *problème d'optimisation multiobjectif*, il est très fréquent de limiter les valeurs que peuvent prendre les variables de décision. Ces restrictions se traduisent par les g *contraintes*. Elles sont notées $g_i(x)$, $i \in \{1, \dots, p\}$.

Dans notre cas pratique, on a affaire à différentes entreprises sans restrictions explicites. Notre modélisation du problème d'optimisation fera abstraction de toute contrainte concernant les entreprises elles mêmes (nos variables de décision).

1.2 Formulation du problème d'optimisation multiobjectif:

L'objectif de notre algorithme est d'estimer les poids relatifs de chaque critère. Cette notion d'importance relative des critères est centrale en aide multicritère à la décision. Elle est utilisée dans la majorité des méthodes proposées et vise à distinguer le rôle de chaque critère dans l'élaboration des préférences globales. Dans une situation décisionnelle réelle, connaître les préférences du décideur et déterminer les poids des critères sont bien souvent des tâches très délicates. [SEF07]

Estimer les poids relatifs à chaque critère revient à vérifier la qualité de fournisseurs obtenue en appliquant ces poids sur les valeurs de critères proposée par les entreprises : Le principal objectif de l'algorithme proposé est d'avoir une pondération des critères d'évaluation, tel que si on utilise cette pondération dans la formule de sélection, on est plus au moins sûr de choisir l'entreprise la plus appropriée.

Il est clair que le maître d'ouvrage tente de sélectionner l'offre qui s'approche le plus de celle qu'il a établit lui-même (l'offre de référence). Ce qui se traduit mathématiquement par la minimisation de la différence entre les valeurs des critères dans l'offre de l'entreprise i et les valeurs des critères dans l'offre de référence.

Ceci revient à résoudre le problème d'optimisation multiobjectif suivant :

$$\text{Soient } F \text{ la fonction objectif globale, tq } F : \mathbb{R}_+^n \rightarrow \mathbb{R}_+^n$$

$$\text{Min } F = [|f_1(X) - f_1(X^*)|, |f_2(X) - f_2(X^*)|, \dots, |f_i(X) - f_i(X^*)|, |f_{i+1}(X) - f_{i+1}(X^*)|, \dots, |f_n(X) - f_n(X^*)|]^T$$

Avec :

$2 < i \leq n$; tel que n est le nombre de critères considérés.

X est la variable de décision, i.e. l'entreprise.

$f_i(X)$: la valeur du critère i dans l'offre de l'entreprise X

X^* : l'offre de référence établie par le maître d'ouvrage.

Ce problème d'optimisation multicritère minimise le coût total en considérant *simultanément* tous les critères d'évaluation.

La suite de ce chapitre comportera l'algorithme AGMSE (Algorithme Génétique Multiobjectif pour le problème de Sélection des Entreprises). Comme son nom l'indique, cet algorithme, basé sur les algorithmes génétiques multiobjectifs, tente de résoudre le problème de sélection des entreprises.

Il est important de souligner à ce stade que l'AGMSE traite le problème cité *tel qu'il a été défini ci-dessus*. Cette modélisation concorde très bien avec l'objectif ciblé après la recherche bibliographique effectuée (c à d la définition de la pondération optimale des critères dans la formule de sélection). Cette modélisation ne répond pas à d'autres objectifs, tels que le traitement du processus de sélection de fournisseurs du début jusqu'à la fin, qui nécessite, fort probablement, d'autres modélisations ou d'autres techniques qu'on n'a pas pu toucher.

2 Application des algorithmes génétiques multiobjectifs pour le problème de sélection d'entreprises pour l'octroi de marchés :

Nous avons vu, dans les deux premiers chapitres, que les AGs multiobjectif sont utilisés dans la phase de définition des poids relatifs de chaque critère au sein du processus d'aide à la décision du problème de sélection des entreprises.

Le paragraphe suivant est la présentation de l'algorithme AGMSE (Algorithme Génétique Multiobjectif pour le problème de Sélection des Entreprises) qui est notre adaptation de la métaheuristique des AGs au problème de sélection des entreprises pour l'octroi du marché. L'objectif de cet algorithme est de générer une formule d'évaluation des offres qui sera la plus adéquate vis-à-vis les préférences des décideurs.

2.1 Principe de l'algorithme AGMSE :

L'AGMSE peut être schématisé par une boîte noire à trois entrées et une seule sortie.

Les entrées :

1. *Les critères d'évaluation* : ce sont les critères de sélection fixés par le maître d'ouvrage pour le projet courant.
2. *L'offre de référence* : c'est l'offre établie par le maître d'ouvrage lui-même afin qu'elle lui sert de référence pour l'évaluation des entreprises candidates. Plus l'offre d'une entreprise se rapproche de cette offre, plus elle aura de probabilité d'être sélectionnée pour la réalisation du projet.
3. *Les offres des entreprises* : c'est les réponses des différentes entreprises à l'appel d'offre lancé par le maître d'ouvrage.

La sortie :

La sortie va correspondre à la pondération la plus adéquate des critères. Cette pondération correspond en fait à la formule de sélection des entreprises utilisée lors de la phase de sélection.

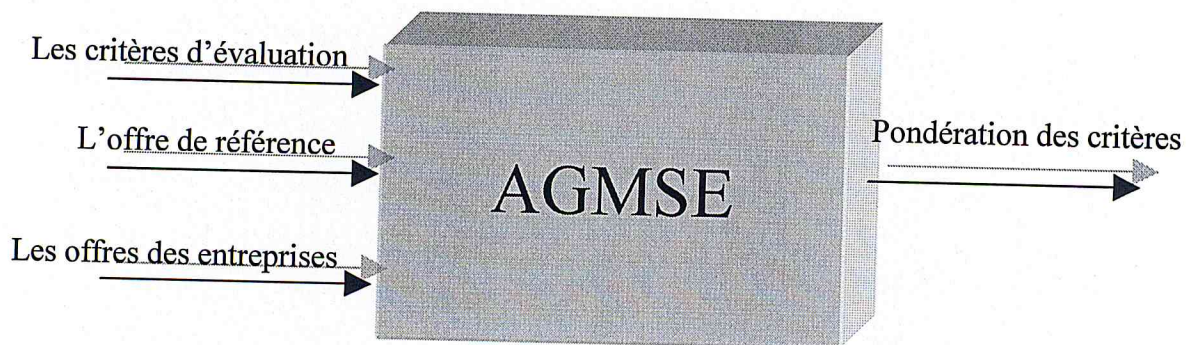


Fig. 3-1: Principe générale de l'AGMSE

2.1.1 Pseudo algorithme :

L'algorithme AGMSE peut être résumé dans les six étapes qui suivent. Les détails de cet algorithme y compris ses éléments (le codage, la génération de population initiale, les mécanismes de reproduction et la fonction d'évaluation...) seront repris dans la section 2.2.

- Etape 0 :** Définir le codage du problème et les paramètres de l'algorithme.
- Etape 1 :** $t = 0$, créer une population initiale de poids $P(0) = p_1, p_2, p_3, \dots, p_p$;
- Etape 2 :** Pour chaque individu $p_i, i = 1..p$: Appliquer la valeur de p_i sur la matrice Entreprise/ Critère;
- Etape 3 :** Pour chaque individu $p_i, i = 1..p$: Appliquer la fonction d'évaluation EV.
- Etape 4 :** Appliquer la stratégie de sélection : Sélectionner 'p' individus de $P(t)$; les ranger dans $S(t)$;
- Etape 5 :** Pour chaque paire d'individus de $S(t)$:
 Appliquer l'opérateur de croisement avec une probabilité p_c ;
 Appliquer l'opérateur de mutation avec une probabilité p_m ;
- Etape 6 :** $t = t+1, P(t) = S(t)$,
 Si [(NbreIt=MaxIt) Ou (Min F ≤ Eps)] alors Stop,
 Sinon : aller à l'étape 2 ;

Fig. 3-2: Pseudo algorithme de l'AGMSE

2.1.2 L'organigramme AGMSE

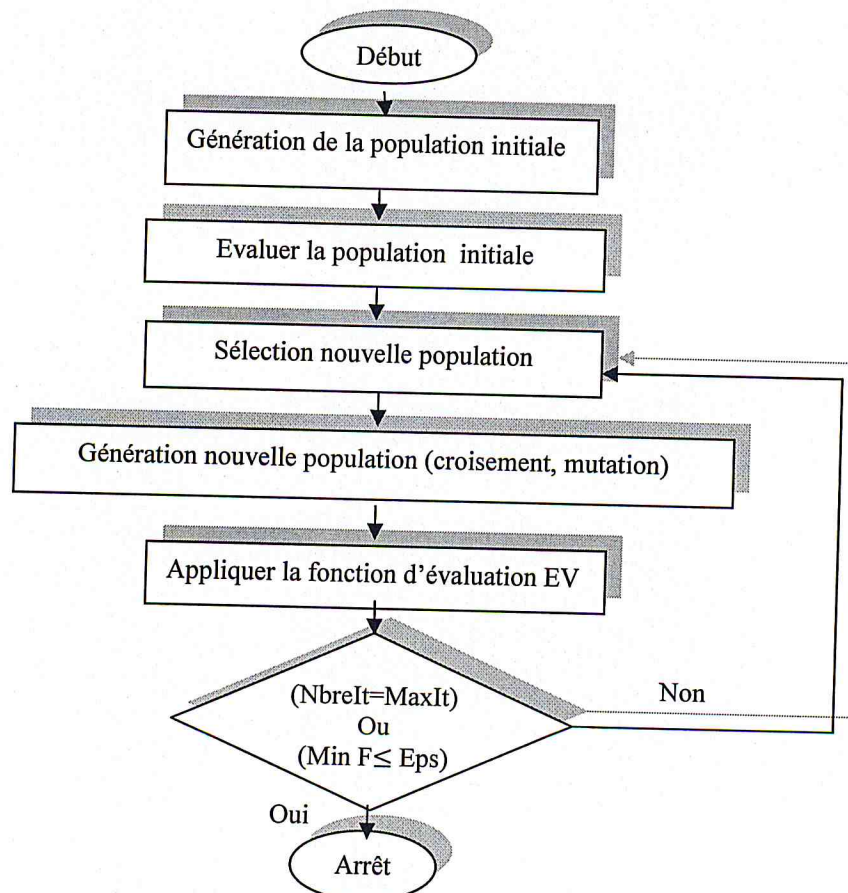


Fig. 3-3: Organigramme général de l'AGMSE

2.2 Éléments de l'algorithme AGMSE:

2.2.1 Codage :

Dans les AGs, un des facteurs les plus importants est la façon dont sont codées les solutions. Comme le dit Goldberg lui-même : « le codage des données est un art, et de cet art dépend le succès ou l'échec de la tentative » [ALL94].

Parmi les codages proposés dans la littérature (voir chapitre 1), le codage retenu dans notre algorithme est le codage *réel*.

Les autres types de codage, et plus précisément le codage binaire, sont éventuellement applicables mais ne présentent, jusqu'à preuve du contraire, aucun intérêt pratiques. Nous conjecturons que dans notre cas, un individu codé en binaire nécessite un traitement supplémentaire point de vu opérateurs (croisement, mutation...) et ne présente absolument aucun avantage.

D'un autre coté, la tendance actuelle va de plus en plus vers la représentation la plus naturelle du problème traité qui n'est autre que l'utilisation du codage *réel*.

a) Le codage réel des individus :

Dans notre codage réel, chaque individu $P = (p_1, p_2, \dots, p_n)$ représente les valeurs réelles des poids p_i assignés aux critères C_i ($i=1..n$). Un individu P aura donc la longueur n qui sera égale au nombre de critères considérés dans le problème. Autrement dit, un individu regroupera, dans l'ordre, les coefficients des critères de sélection apparaissant dans la formule de sélection utilisée dans le bureau d'étude.

On pourra schématiser l'individu de la manière suivante :

p_1	p_2	p_3	...	p_i	p_{i+1}	p_{n-1}	p_n
-------	-------	-------	-----	-------	-----------	-----	-----	-----------	-------

Un individu $P = (p_1, p_2, \dots, p_n)$ de chaque population doit respecter au moins ² les deux conditions suivantes :

$$\left\{ \begin{array}{l} \sum_{i=1}^n p_i = 1 \quad (1) \\ p_i > 0, \forall i = 0 \dots n \quad (2) \end{array} \right.$$

² L'expression « au moins » suppose que le bureau d'étude peut exiger d'autres conditions telles que les valeurs extrêmes que le coefficient d'un critère ne doit pas dépasser.

Exemples :

Exemple 1: Considérons que l'appel d'offre comporte les six critères suivants :

- C_1 : Prix ;
- C_2 : Délai de livraison ;
- C_3 : Qualité ;
- C_4 : Capacité de production ;
- C_5 : Localisation géographique ;
- C_6 : Capacité technique.

L'individu représenté par le vecteur $P_1 = (0.4; 0.1; 0.3; 0.075; 0.06; 0.065)$, représente les poids attribués à chaque critère par les experts en fin de leur réunion.

Ces poids sont utilisés dans la formule de sélection des entreprises suivante :

$$0.4 * NC_1 + 0.1 * NC_2 + 0.3 * NC_3 + 0.075 * NC_4 + 0.06 * NC_5 + 0.065 * NC_6$$

Tels que : les NC_i sont les notes attribuées pour le critère i , à l'entreprise en cours de traitement.

Exemple 2: Considérons le même appel d'offre détaillé dans l'exemple précédent :

Soit P_2 un individu de la population initiale généré par notre système, tel que P_2 est représenté par le vecteur $(0.4; 0.15; 0.2; 0.15; 0.09; 0.075)$,

Ces poids sont utilisés dans la formule de sélection des entreprises suivante :

$$0.4 * NC_1 + 0.15 * NC_2 + 0.2 * NC_3 + 0.15 * NC_4 + 0.09 * NC_5 + 0.075 * NC_6$$

Nous pouvons facilement vérifier que cette formule n'est pas valide puisque :

$$\sum_{i=1}^6 p_i > 1$$

Exemple 3 : Le problème inverse se pose si on prend l'exemple d'un autre individu de la population initiale généré par notre système : l'individu $P_3 = (0.3; 0.05; 0.3; 0.075; 0.06; 0.065)$,

La formule de sélection d'entreprises basée sur cet individu, elle aussi, ne sera pas valide.

$$0.3 * NC_1 + 0.05 * NC_2 + 0.3 * NC_3 + 0.075 * NC_4 + 0.06 * NC_5 + 0.065 * NC_6$$

En effet, dans ce cas en aura :

$$\sum_{i=1}^6 p_i < 1$$

Remarques :

1. D'autres cas de figures peuvent se présenter si le bureau d'étude exige un intervalle de poids pour chaque critère, qui est une condition très courante. Dans ce cas, l'individu d'une population devra non seulement vérifier la condition générale (la somme des poids égale à un) mais aussi, les valeurs de poids attribuées à chaque critères devront appartenir à l'intervalle considéré.

Ainsi, pour la formule d'évaluation générale:

$$p_1 * NC_1 + p_2 * NC_2 + \dots + p_i * NC_i + \dots + p_n * NC_n$$

On devra vérifier que pour chaque critère i , on a $p_i \in [min_i, max_i]$, tels que min_i, max_i sont respectivement les valeurs extrêmes minimales et maximales que peut prendre le critère i , d'après notre bureau d'étude.

2. La condition (2), les poids doivent être strictement positifs, ne peut en aucun cas être violée vu l'espace de recherche considéré (c à d : \mathbb{R}_+^n).

Nous verrons par la suite la solution adoptée pour assurer la conformité des individus par rapport aux exigences du bureau d'études.

b) La normalisation des individus :

Les cas illustrés ci-dessus, nous ont poussé à introduire une certaine *normalisation* des individus. On doit *forcer* les individus à vérifier la condition de la formule de sélection (la somme des poids égale à un, voir formule (1) section 2.2.2) et de vérifier, éventuellement, les restrictions des valeurs définies par le bureau d'étude.

i. Principe de la normalisation:

Le principe de cette normalisation est en fait très simple : si la somme des poids dans un individu est supérieur à un : on retranche cet excès. Dans le cas contraire, c à d la somme des poids dans un individu est inférieur à un : on ajoute la quantité manquante.

Cette quantité ajoutée (respectivement retranchée) devra être suffisamment petite pour ne pas trop s'éloigner de l'individu d'origine. Pour cela, on introduit deux paramètres Δ_i et Δ_i' tel que :

Δ_i : représente la portée de l'intervalle $[min_i, max_i]$ se rapportant au critère i .

$$\Delta_i = max_i - min_i$$

Δ_i' : c'est la quantité (*suffisamment petite*) qu'on ajoute (resp. retranche) lors de la phase de normalisation.

$$\Delta_i' = \frac{\Delta_i}{Np}$$

Avec Np : la taille de la population.

Par la suite, on donnera le pseudo algorithme de la procédure de normalisation.

ii. L'algorithme de la normalisation:

Soit $P = (p_1, p_2, \dots, p_n)$ un vecteur représentant un individu de la population, N sa taille, Δ_i et Δ_i' les deux paramètres définies dans la section précédente ; la procédure de normalisation peut être mise sous forme algorithmique comme suit :

```

Normalisation (P) :
Début
somme =0;
  Pour  $i=1 \dots N$  :
    Début :
      somme =somme+  $p_i$ ;
      Calculer  $\Delta_i$  et  $\Delta_i'$  ;
    fin ;
   $\Delta_j =$  somme-1;
   $i=0$ ;
  Si la somme < 1 alors :
    Tant que ( $\Delta_j < 0$ ) faire :
      minDelta= Min ( $\Delta_j, \Delta_i'$  ) ;
      si ( $p_i + \text{minDelta} \leq \text{max}_i$ )
        Début
           $p_i = p_i + \text{minDelta}_i$ ;
           $\Delta_i = \Delta_i - \text{minDelta}_i$ ;
        Finsi ;
       $i=(i) \bmod (N)+1$ ;
    FinTantque
  Sinon
    Tant que ( $\Delta_j > 0$ ) faire :
      si ( $p_i - \text{minDelta} \geq \text{min}_i$ )
        Début
           $p_i = p_i - \text{minDelta}_i$ ;
           $\Delta_i = \Delta_i - \text{minDelta}_i$ ;
        Finsi;
       $i=(i) \bmod (N)+1$ ;
    FinTantque
  FinSi
Fin

```

Fig. 3-4: Algorithme de Normalisation

2.2.2 Population initiale :

Nous avons présenté dans le premier chapitre quelques stratégies pour la génération de la population initiale. Dans l'AGMSE, on tente de générer la population initiale suivant diverses stratégies, exactement trois, afin d'arriver à de meilleurs résultats.

La première stratégie est la *génération aléatoire* : Cette génération permet au AG une meilleure exploration de l'espace de recherche. Théoriquement, l'exécution de l'algorithme risque d'être un peu coûteuse en temps.

La deuxième stratégie est une *génération heuristique* à partir d'une seule solution retenue dans l'évaluation des fournisseurs au sein du bureau d'étude ATRIUM³. Cette solution unique subira des mutations et des croisements afin d'obtenir une population initiale composée d'individus assez proches de la solution envisagée. L'utilisation de cette approche permet de réduire le temps d'exécution de l'algorithme. En revanche, on risque de stagner dans des optima locaux puisqu'on a borné au préalable notre espace de recherche (et donc limiter la diversification).

La troisième stratégie est un *jumelage* entre les deux stratégies précédentes : on considère l'intervalle $[min_i, max_i]$ avec $min_i, max_i \in \mathbb{R}$, les valeurs de min_i et max_i seront déterminées par observation des valeurs extrêmes de chaque poids de critères. Les individus de la population initiale seront *générés aléatoirement* dans l'intervalle $[min_i, max_i]$.

Ainsi, un individu $P=(p_1, p_2, \dots, p_n)$ généré via cette stratégie doit respecter, en plus de la condition (1) (section 2.1.1), la condition suivante :

$$p_i \in [min_i, max_i], \forall i = 1 \dots n \quad (2)$$

Pratiquement, on propose, pour assurer les conditions de cette génération, d'implémenter la formule suivante :

$$P_i = min_i + \Delta_i \cdot S \quad (3)$$

Tel que :

P_{ij} : valeur du poids du critère i dans l'individu courant.

Δ_i : la quantité définie dans la section 2.2.1.

S : un nombre aléatoire généré entre 0 et Np (la taille de la population.)

³ Nous avons opté pour ce choix de population initiale car on n'a pas pu avoir les différentes propositions de poids établies par les experts lors de leurs réunions. En effet, on a eu accès uniquement aux résultats finaux des réunions d'évaluations.

Cette stratégie offre une bonne exploration de l'espace de recherche évitant ainsi de converger rapidement vers des optima locaux ou d'explorer des points non significatifs de l'espace de recherche.

Notons que quelque soit la stratégie de génération de population initiale, il est indispensable d'effectuer la normalisation proposée précédemment.

Chacune de ces trois stratégies sera implémentée dans le logiciel dédié, et des tests numériques seront établis afin de comparer et d'évaluer ces trois approches (voir chapitre 4).

2.2.3 Evaluation :

La fonction d'évaluation (fitness) EV que nous définissons dans L'AGMSE est basée sur les deux notions mathématiques les plus célèbres dans le domaine d'aide à la décision: la notion de dominance, et le concept du front de Pareto (les concepts théoriques de ces deux piliers de la fonction d'évaluation ont été définis au niveau du premier chapitre.)

L'AGMSE met en évidence une fonction d'évaluation (fitness) EV dont le rôle principal est de calculer le nombre de fois qu'un individu est dominé.

On calcule ainsi les valeurs des critères f_i ($i=1 \dots n$) pour chaque individu de la génération et on les compare à celles des autres individus. Le classement des individus d'une génération (N individus) se fait enfin grâce à leurs valeurs globales de leurs fonctions d'évaluation.

La fonction EV d'évaluation des individus est définie de la manière suivante :

$$EV: \mathbb{R}^n \rightarrow \mathbb{N}$$

$$F \rightarrow nb$$

$$\left\{ \begin{array}{l} \text{Si } f_i(x_1) \leq f_i(x_2), \forall i = 1, \dots, n \text{ et } \exists i^* : f_{i^*}(x_2) < f_{i^*}(x_1) \text{ alors } EV(x_2) = EV(x_1) + 1 \\ \quad \quad \quad // x_2 \text{ est dominé par } x_1 \\ \text{Si } f_i(x_1) \geq f_i(x_2), \forall i = 1, \dots, n \text{ et } \exists i^* : f_{i^*}(x_1) > f_{i^*}(x_2) \text{ alors } EV(x_1) = EV(x_2) + 1 \\ \quad \quad \quad // x_1 \text{ est dominé par } x_2 \end{array} \right.$$

2.2.4 Sélection :

Parmi les nombreuses stratégies de sélection utilisées par les AGs (voir chapitre 1), l'AGMSE fait appel à deux stratégies de sélection : La sélection élitiste et la sélection par loterie biaisée. Le choix de ces deux stratégies est influencé par la réputation de ces dernières ainsi que par leur facilité d'implémentation vis-à-vis des autres méthodes existantes.

1. La sélection élitiste :

Dans notre cas, moins l'individu est dominé par d'autres individus de sa génération plus il est adapté pour le problème posé. Puisque la valeur de la fonction d'évaluation EV augmente à chaque fois qu'un individu est dominé par d'autres, elle est inversement proportionnelle avec la notion d'adaptabilité de l'individu au problème.

Pour cela notre population est classée en ordre *croissant* de valeurs de fonction d'évaluation. Les NP /2 premiers individus seront sélectionnés pour générer la population suivante.

2. La sélection par loterie biaisée: [MER07]

Cette sélection permet d'octroyer aux individus ayant une bonne évaluation, plus de chance d'être sélectionné que ceux ayant une moins bonne évaluation. Une probabilité de sélection, notée $Prob(P_i)$, est attribuée à chaque individu P_i et peut être calculée par l'équation suivante :

$$Prob(P_i) = \frac{EV_{max} - EV(P_i)}{EV_{max} - EV_{min}} .100\%$$

Avec :

EV_{min} : L'évaluation du meilleur individu de la population.

EV_{max} : l'évaluation du plus mauvais individu de la population.

Grace à cette expression, on la certitude que le meilleur individu de la population, donc celui dont l'évaluation est égale à EV_{max} , est toujours sélectionné puisque sa probabilité de sélection est de 100%. Par contre, le plus mauvais individu de la population est automatiquement éjecté de la génération suivante.

Une fois les probabilités de sélection des individus de la population sont calculées, on sélectionne un parent comme suit :

- On calcule S_1 , la somme des probabilités de sélection pour l'ensemble de la population.
- On tire un nombre aléatoire, notée r , selon une distribution uniforme sur l'intervalle $[0, S_1]$.
 - On calcule une somme S_2 des probabilités de sélection en s'arrêtant dès que cette somme dépasse r .
 - Le dernier individu dont la probabilité de sélection vient d'être ajoutée est sélectionné pour jouer le rôle du parent.

Une fois le premier parent sélectionné, on réitère l'opération pour sélectionner un deuxième parent tout en assurant qu'il est différent du premier. L'application des opérateurs de croisement permet alors de générer des nouvelles solutions à partir des deux parents.

2.2.5 Croisement :

L'opérateur de croisement est l'opérateur le plus important dans les AGs.

L'AGMSE met en évidence deux types de croisement : un croisement à un point et un croisement à k points ($k > 1$). Chacun de ces deux types de croisement peut éventuellement générer des anomalies exactement similaires à celles énoncées lors de la présentation du codage des individus (section 2.2.1).

Exemple :

Prenons l'exemple des deux chromosomes suivants :

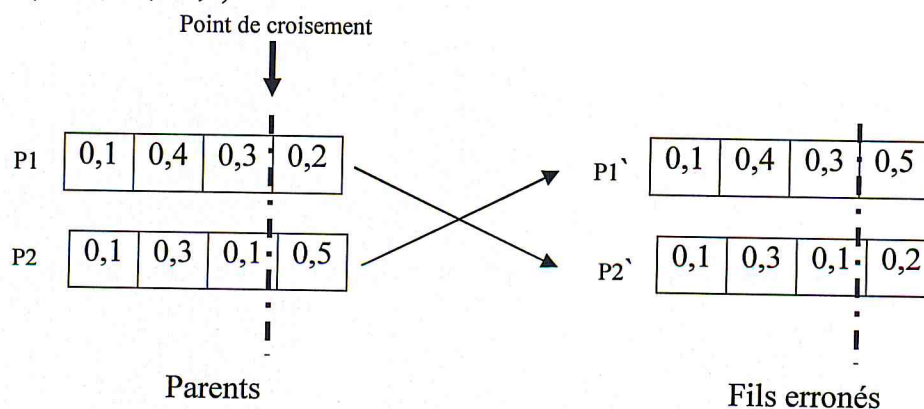
$$P_1 = (0,1 \ 0,4 \ 0,3 \ | \ 0,2)$$

$$P_2 = (0,1 \ 0,3 \ 0,1 \ | \ 0,5)$$

Le croisement à un point de ces deux individus donnera le résultat suivant :

$$P_1' = (0,1 \ 0,4 \ 0,3 \ 0,5)$$

$$P_2' = (0,1 \ 0,3 \ 0,1 \ 0,2)$$



Or que :

$$\sum_{i=1}^4 p_1' > 1$$

$$\sum_{i=1}^4 p_2' < 1$$

De ce faite, il est clair que des croisements sans normalisation vont générer des individus erronés. On remédiera à cette situation par la formule de normalisation proposée postérieurement dans la section 2.2.1.

2.2.6 Mutation :

La mutation a pour but de réintroduire des valeurs significatives perdues au cours des générations, i.e. de guider la recherche d'une façon heuristique.

Nous définissons dans notre algorithme un opérateur de mutation qui consiste au *remplacement* des valeurs du bit P_i du vecteur P, tel que i est un nombre aléatoire entre 1 et n (n est la longueur de l'individu, c à d le nombre de critères).

Par analogie à l'étape de génération des individus et à l'étape de croisement, la mutation passe par deux étapes *successives et atomiques* :

1. La mutation : Le remplacement de la valeur de P_i avec une valeur P_i' . Cette dernière doit appartenir également à l'intervalle $[min_i, max_i]$ du bit P_i .
2. La normalisation de l'individu muté.

Dans ce cas aussi, la normalisation précédente est nécessaire pour que l'individu P_j' vérifie la condition (1).

Selon la valeur de P_i' , l'AGSME offre deux stratégies de mutation définies comme suit :

a) *Stratégie 1 : la mutation aléatoire :*

Dans cette stratégie la valeur de P_i est remplacée avec une valeur aléatoire P_i' tel que :

$$P_i' = min_i + \Delta_i' * S$$

Tel que :

P_{ij} : valeur du poids du critère i dans l'individu courant.

Δ_i' : la quantité définie dans la section 2.2.1.

S : un nombre aléatoire généré entre 0 et Np (*la taille de la population.*)

b) Stratégie 2 : mutation guidée :

Dans cette stratégie on tente de ne pas trop s'éloigner d'ancienne valeur de P_i . Cette dernière est remplacée avec une valeur P_i' , tel que :

$$\begin{cases} \text{Si } P_i + \Delta_i' < \max_i \text{ alors } P_i' = P_i + \Delta_i' \\ \text{Sinon } P_i' = P_i - \Delta_i' \end{cases}$$

Avec :

Δ_i' : la quantité définie dans la section 2.2.1.

Dans notre algorithme, la mutation se fait avec une probabilité P_m . On définira la valeur de ce paramètre dans la partie des tests numériques.

2.2.7 Condition d'arrêt :

Nous définissons la condition d'arrêt de l'AGMSE avec l'expression : [(NbreIt=MaxIt) Ou (Min F ≤ Eps)].

Cette expression est basée sur les deux paramètres : Le nombre d'itérations maximal et le vecteur de paramètres *grand epsilon* Eps. Ces deux paramètres seront décrits par la suite.

a) Le nombre d'itérations maximal:

Ce paramètre permet de limiter l'exécution de l'algorithme à un nombre d'itérations défini par l'utilisateur. La phase de teste numérique, dans le chapitre suivant, pourra nous donner une idée sur le nombre d'itération par défaut, c à d qui donne des résultats satisfaisant dans la plupart des cas.

b) Le vecteur de paramètres grand epsilon Eps :

Nous considérons le vecteur de paramètres Eps = $(\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n)$ tel que ε_i est la valeur minimale souhaitée pour $|f_i(X) - f_i(X^*)|$, dans notre fonction objectif.

D'une manière plus simple, ce critère d'arrêt peut être lu ainsi : *Lorsque pour chaque critère considéré la différence entre le résultat obtenu et le résultat optimum est suffisamment petite, on peut décider de mettre fin à l'exécution de l'algorithme.*

Nous avons pris l'initiative d'utiliser ce critère d'arrêt qui n'est, en fait, rien qu'une simple *généralisation* du critère d'arrêt *epsilon standard* dans le cas de fonction à un seul objectif (voir chapitre 1).

On notera finalement que cette condition d'arrêt peut être omise dans le cas d'absence d'information sur les valeurs des ε_i (utilisateur non expérimenté ou hors du domaine de construction).



Nous avons présenté dans ce chapitre la modélisation proposée pour le problème de sélection des entreprises pour l'octroi de marchés. Cette modélisation est rigoureuse du fait qu'elle est basée sur des notions mathématiques d'aide à la décision. Le grand intérêt de cette approche est de permettre à l'ingénieur du bureau d'étude de ne pas restreindre à priori son champ de recherche son recherche par une pondération de ses objectifs. Cette modélisation a permis l'élaboration de l'algorithme AGMSE qui constitue notre adaptation des AGs multiobjectifs au problème de sélection des entreprises. Cet algorithme a la particularité capitale d'être multiobjectif, d'un côté. Et d'un autre côté, il offre un tas de stratégies pour chaque phase ce qui nous permettra d'effectuer une validation et des tests assez consistants. Ces derniers seront concrétisés dans le chapitre qui suit.

CHAPITRE 4

Implémentation

&

Tests expérimentaux

CHAPITRE 4

Implémentation et tests expérimentaux

Introduction

Ce dernier chapitre est la concrétisation du travail élaboré dans ce mémoire. On y présente le logiciel dédié à l'implémentation de l'AGMSE. Cette dernière a été développée sous l'environnement MATLAB. Le logiciel ainsi conçu servira d'outil d'aide à la décision au sein des bureaux d'études. Il peut être utilisé seul, pour accélérer le processus de sélection d'entreprise en automatisant la phase de pondération des critères.

Nous commençons par la présentation de l'interface utilisateur et les différentes fonctionnalités offertes par le logiciel.

Nous procédons par la suite aux réglages de paramètres et de jeux de test effectués. Les résultats de ces tests sont comparés avec des données réelles du bureau d'étude AUTIRIUM.

Tous les tests ont été réalisés sur Intel Core 2 Duo processor T5670, 1.8GHz avec 3Go de RAM.

1. Présentation du logiciel :

Le logiciel que nous avons conçu est développé sous l'environnement MATLAB. Nous passerons avant la présentation du logiciel par un bref aperçu de cet environnement.

1.1 L'environnement MATLAB [MAH09]

1.1.1 Présentation de MATLAB

MATLAB était destiné à faciliter l'accès au logiciel matriciel développé dans les projets LINPACK et EISPACK. La version actuelle, écrite en C par the MathWorks Inc., existe en version professionnelle et en version étudiant. Sa disponibilité est assurée sur plusieurs plates

formes : Sun, Bull, HP, IBM, compatibles PC (DOS, Unix ou Windows), Macintosh, iMac et plusieurs machines parallèles.

MATLAB est un environnement puissant, complet et facile à utiliser destiné au calcul scientifique. Il apporte aux ingénieurs, chercheurs et à tout scientifique un système interactif intégrant calcul numérique et visualisation. C'est un environnement performant, ouvert et programmable qui permet de remarquables gains de productivité et de créativité.

MATLAB possède son propre langage, intuitif et naturel qui permet des gains de temps de CPU spectaculaires par rapport à des langages comme le C, le Turbo Pascal et le Fortran.

MATLAB comprend aussi un ensemble d'outils spécifiques à des domaines, appelés *Toolboxes* (ou Boîtes à Outils). Ces dernières sont des collections de fonctions qui étendent l'environnement MATLAB pour résoudre des catégories spécifiques de problèmes. Les domaines couverts sont très variés et comprennent notamment le traitement du signal, l'automatique, l'identification de systèmes, les réseaux de neurones, la logique floue, le calcul de structure, les statistiques, etc.

1.1.2 particularités de MATLAB

MATLAB permet le travail interactif soit en mode commande, soit en mode programmation ; tout en ayant toujours la possibilité de faire des visualisations graphiques. Considéré comme un des meilleurs langages de programmations (C ou Fortran), MATLAB possède les particularités suivantes par rapport à ces langages :

- la programmation facile,
- la continuité parmi les valeurs entières, réelles et complexes,
- la gamme étendue des nombres et leurs précisions,
- la bibliothèque mathématique très compréhensive,
- l'outil graphique qui inclut les fonctions d'interface graphique et les utilitaires,
- la possibilité de liaison avec les autres langages classiques de programmations (C ou Fortran).

Dans MATLAB, aucune déclaration n'est à effectuer sur les nombres. En effet, il n'existe pas de distinction entre les nombres entiers, les nombres réels, les nombres complexes et la simple ou double précision. Cette caractéristique rend le mode de programmation très facile et très rapide.

La bibliothèque des fonctions mathématiques dans MATLAB donne des analyses mathématiques très simples.

Pour l'interface graphique, des représentations scientifiques et même artistiques des objets peuvent être créées sur l'écran en utilisant les expressions mathématiques. Les graphiques sur MATLAB sont simples et attirent l'attention des utilisateurs, vu les possibilités importantes offertes par ce logiciel.

1.2 Description du logiciel

1.2.1 Les interfaces utilisateur

a) L'interface choix d'utilisateur

La première interface utilisateur de notre logiciel est un menu de choix d'utilisateur. Elle propose de sélectionner entre deux utilisateurs : l'ingénieur du bureau d'étude et l'informaticien.

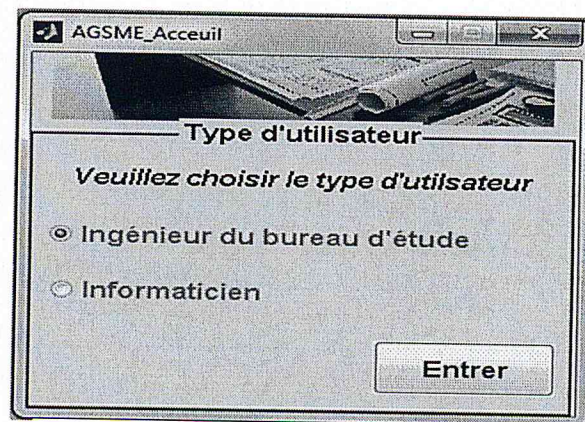


Fig.4.1 : Le menu de choix principal

b) Interface: Ingénieur du bureau d'étude :

Cette interface permet à l'ingénieur du bureau d'étude de procéder directement aux données de son problème de sélection en commençant par le nombre de critères et le nombre d'offres (Fig.4.2). Un formulaire d'insertion de données (Fig.4.3) apparaît en cliquant sur le bouton « ok » de cette interface.

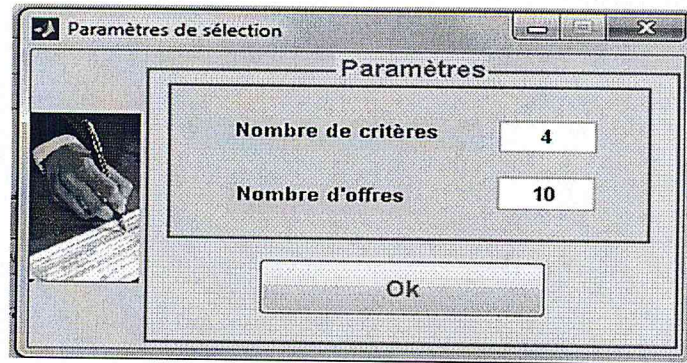


Fig.4.2 : interface ingénieur du bureau d'étude

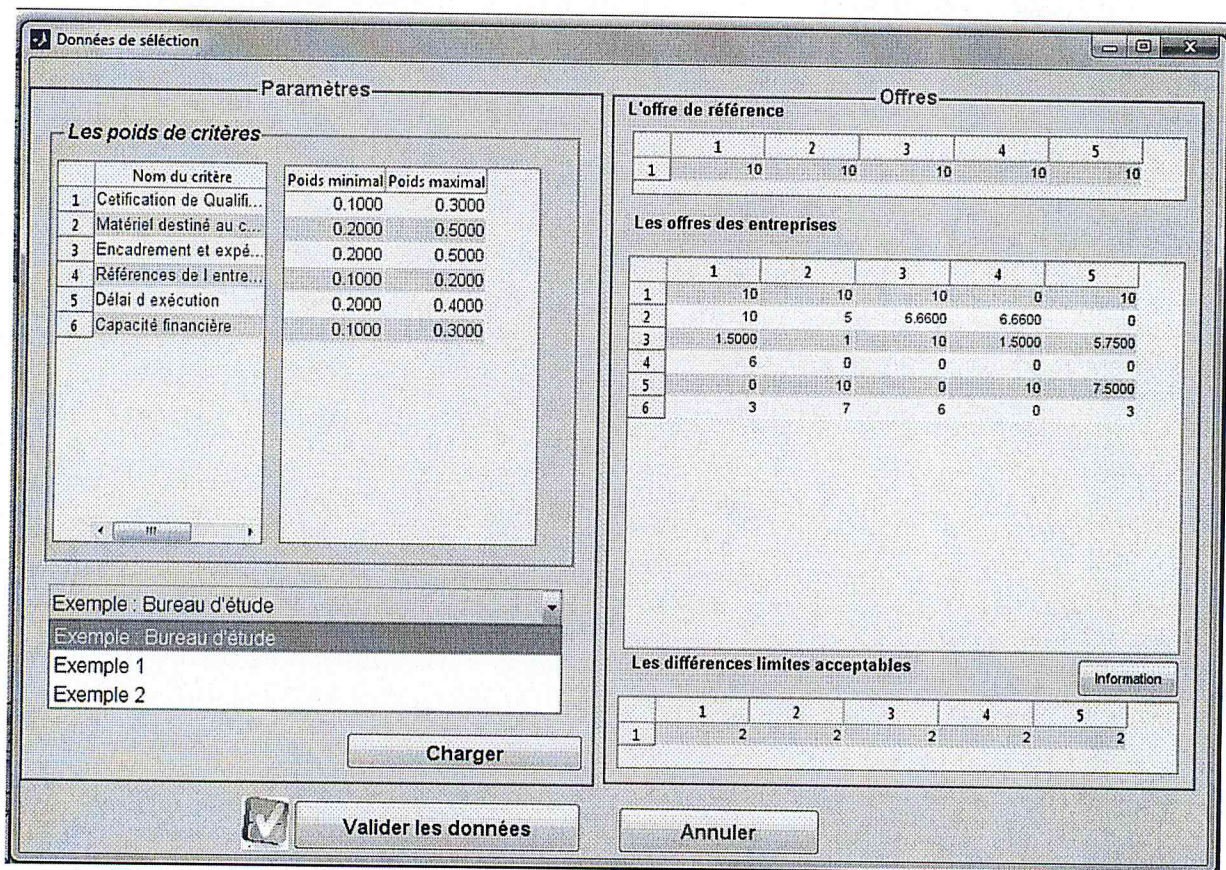


Fig.4.3 : Formulaire de données de sélection

c) Interface: Informaticien

Cette interface (Fig.4.4) permet à l'informaticien de procéder directement aux réglages de paramètres de l'AGMSE (Fig.4.5). Il doit bien sûr introduire les données du problème de sélection en commençant par le nombre de critères et le nombre d'offres (Fig.4.2) et les autres données de sélection (Fig.4.3). Une fois ces paramètres réglés, on lance l'exécution en cliquant sur le bouton « exécution ».

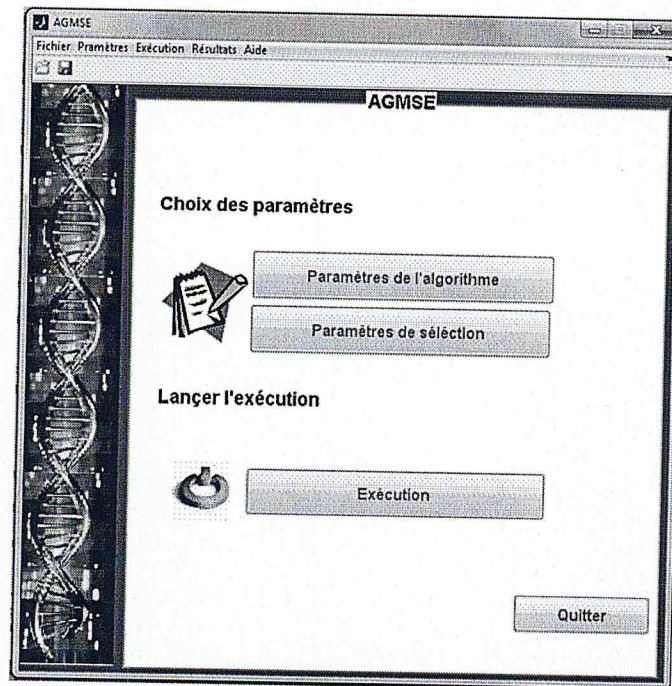


Fig.4.4 : interface informaticien

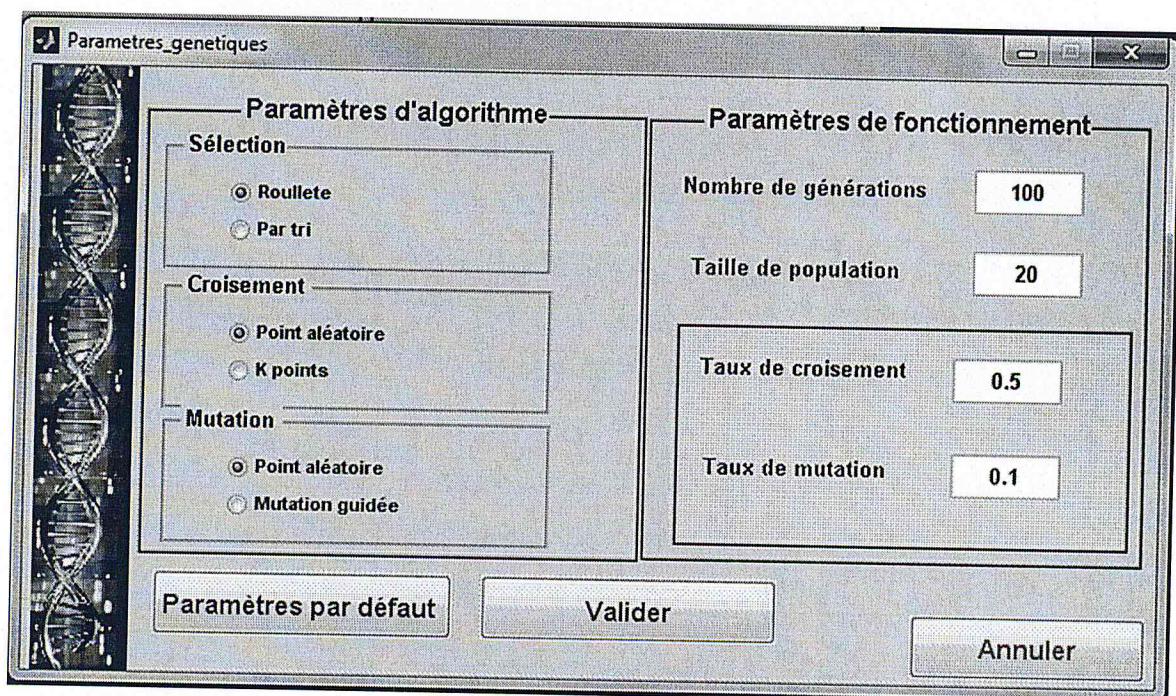


Fig.4.5 : Menu réglages de l'algorithme génétique

d) Exécution et détails de résultats :

L'exécution à partir du menu précédent fait apparaître le menu suivant (fig.4.6). il présente les évaluations maximales, minimales et la moyenne des évaluations pour chaque offres par rapport à tout les poids de critères. Plus de détails sur l'interet de ces évaluation sera présenté dans la partie des tests numériques.

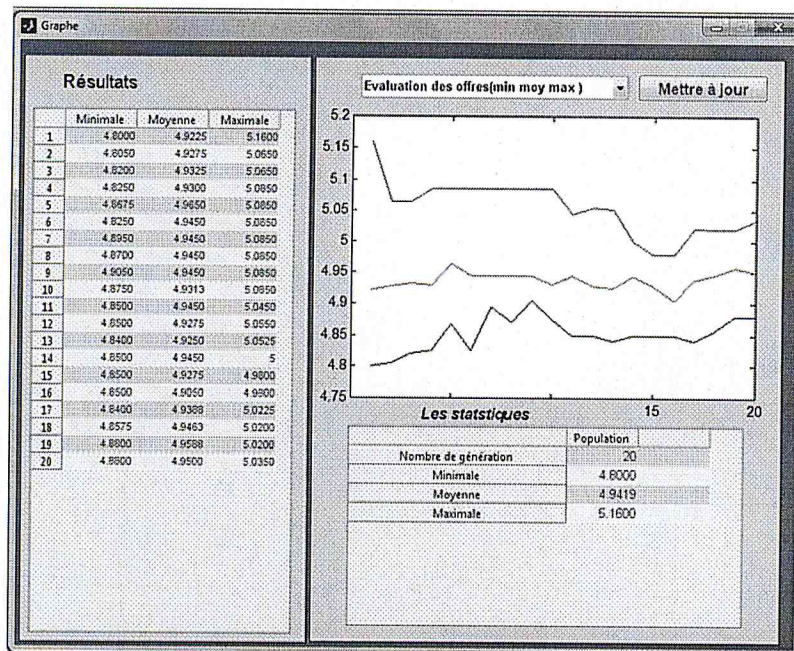


Fig.4.6 : Menu détails des résultats

2. Résultats expérimentaux

Pour analyser les performances de l'algorithme développé sous l'environnement MATLAB, une série de tests a été effectuée sur un problème (qui sera définie ultérieurement dans cette section). Ces tests ont été réalisés sur un Intel Core 2 Duo processor T5670, 1.8GHz avec 3Go de RAM.

Pour mettre en évidence l'influence des différents paramètres de l'algorithme génétique, nous avons appliqué ce dernier sur un même problème en changeant à chaque fois, seulement la valeur d'un des paramètres.

Nous nous sommes intéressées en particulier à la taille de la population, puis au nombre de génération et enfin au taux de mutation. Nous avons également comparé

différentes stratégies de sélection (tri ou roulette), de croisement (croisement d'un point aléatoire ou croisement K points) et enfin de mutation (mutation guidé ou aléatoire).

Tous les tests ont été faits sur un problème que nous définissons ainsi dans la section suivante.

2.1 Les données du problème de tests :

Le problème que nous proposons par la suite est un problème qui permet de mettre en relief le maximum d'aspects de l'algorithme.

Il est défini par les quatre points suivants :

1. Le nombre de critères de choix est égal à trois (3).
2. Les intervalles des poids de critères sont donnés dans le tableau suivant :

Tab.4-1 : Les intervalles de poids pour chaque critère

Poids Critères	Valeur minimale	Valeur maximale
Critère 1	0.2	0.6
Critère 2	0.2	0.6
Critère 3	0.2	0.3

3. Le nombre d'offres est égal à douze (12). Les offres sont décrites ainsi :

Tab.4.2 : les détails des offres du problème de teste

Notes N° Offre	Note du critère 1 /10	Note du critère 2 /10	Note du critère 3 /10
<i>Offre de référence</i>	8	10	8
1	5	4	3
2	3	4	5
3	5	4	5
4	4	4	5
5	4	5	6
6	5	4	6
7	4	4	6
8	2	3	4
9	4	4	7
10	4	5	4
11	5	3	5
12	3	4	2

4. Les différences limites acceptables pour chaque critère par rapport à l'offre de référence sont données dans le tableau suivant :

Tab.4.3 : les valeurs des Epsilon pour chaque critère

Critères	Epsilon
Critère 1	1
Critère 2	2
Critère 3	1

2.2 Les tests :

Sauf indications contraires, tous les tests qui suivent utilisent les stratégies suivantes :

5. Sélection par tri,
6. Croisement d'un seul point,
7. mutation aléatoire.

2.2.1 L'influence du nombre d'individus :

Les premiers tests effectués ont été menés afin de constater l'influence de nombre d'individus sur la qualité de la solution retenue.

Afin de montrer l'influence du nombre d'individus sur la qualité de la solution, nous avons fixé le nombre de génération à 20, le taux de croisement à 0.5 et le taux de mutation à 0.08. On fait varier le nombre d'individus entre 10 et 100.

Pour chaque test on retient la meilleure évaluation et la plus mauvaise. Nous calculons aussi la moyenne de toutes les évaluations de la génération courante.

Les résultats de ces tests sont récapitulés dans le tableau **Tab.4.4**.

Tab.4.4 : l'influence du nombre d'individus sur la qualité de solution

Nombre d'individus	Evaluation		
	minimale	moyenne	maximale
10	4.8400	4.9700	5.0700
20	4.8000	4.9437	5.0700
30	4.8063	4.9203	5.1500
40	4.8025	4.9200	5.1600
50	4.8000	4.9192	5.3077
100	4.8000	4.9243	5.2240

Les résultats présentés ci-dessus, correspondent à la forme graphique suivante (**Fig.4.7**) :

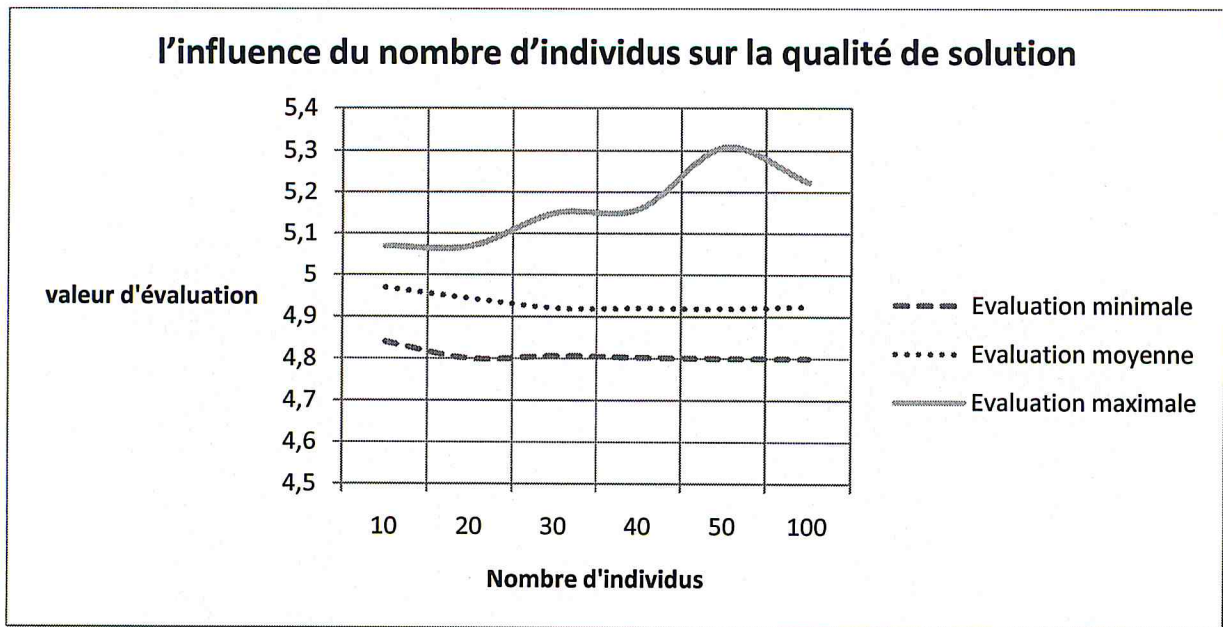


Fig.4.7 : l'influence du nombre d'individus sur la qualité de solution

Discussion du graphe :

Nous évaluons le nombre d'individus en se basant sur l'évaluation moyenne. Plus ce dernier est proche de l'évaluation maximale, plus le nombre d'individus est intéressant. Au contraire, si l'évaluation moyenne est proche de l'évaluation minimale, le nombre d'individus est moins intéressant.

Une moyenne d'évaluation qui s'approche de l'évaluation maximale se traduit pratiquement par la bonne qualité de tout les individus, c à d les poids.

Dans le graphe de la figure **Fig.4.7**, nous pouvons constater que la taille de population idéale est entre 10 et 20.

A partir de 20 individus, plus la taille de population augmente plus les résultats sont moins bons : Ceci est du au paramètre Δ_i' (voir chapitre 3) qui est proportionnel avec la diversification et inversement proportionnel avec la taille de population. Dans le cas ou la taille de population sera très grande, Δ_i' aura une valeur très petite et donc on aura une diversification très faible. Ceci se traduit pratiquement par une population homogène et une convergence rapide.

Cependant, nous avons constaté que le temps de calcul est plus important lors de tests avec une taille de population relativement grande.

De ce qui suit, on déduit que la bonne définition de paramètre est très importante. En effet, cette dernière permet d'établir un compromis entre la qualité de solution et la rapidité de l'algorithme.

Le tableau suivant (Tab.4.5) illustre les détails des résultats des évaluations suivant le nombre d'individus.

Tab.4.5 : Détails des résultats des évaluations suivant le nombre d'individus

Nombre d'individus	Offres finales sélectionnées	Détails des évaluations	
		Minimale	Maximale
10	5 → (5/10) 6 → (5/10)	Offre 6 Individu ₇ (0.4400 0.3600 0.2000) Génération 4	Offre 6 Individu ₃ (0.5500 0.2400 0.2100) Génération 5
20	5 → (13/20) 6 → (7/20)	Offre 5,6 Individu ₅ (0.5400 0.2600 0.2000) Génération 8	Offre 6 Individu ₁₃ (0.5200 0.2400 0.2400) Génération 10
32	5 → (10/30) 6 → (18/30) 10 → (4/30)	Offre 6 Individu ₁₇ (0.4063 0.3937 0.2000) Génération 14	Offre 10 Individu ₁₉ (0.2562 0.5375 0.2063) Génération 4
40	5 → (16/40) 6 → (17/40) 10 → (7/40)	Offre 6 Individu ₂₆ (0.3100 0.4700 0.2200) Génération 8	Offre 10 Individu ₂₅ (0.2575 0.5400 0.2025) Génération 11
52	5 → (25/52) 6 → (26/52) 10 → (1/52)	Offre 5,6 Individu ₁₈ (0.4000 0.4000 0.2000) Génération initial	Offre 10 Individu ₄₀ (0.3635 0.3846 0.2519) Génération initial

2.2.2 l'influence du nombre de générations

Afin de montrer l'influence du nombre de générations sur la qualité de la solution, nous avons fixé le nombre d'individus à 20, le taux de croisement à 0.5 et le taux de mutation à 0.08. On fait varier le nombre de génération entre 10 et 250.

Pour chaque test on retient la meilleure évaluation et la plus mauvaise. Nous calculons aussi la moyenne de toutes les évaluations de la génération courante. Nous retenons cette fois les offres finales avec la fréquence de leurs apparitions.

Tab.4.6: l'influence du nombre de générations sur la qualité de solution

Nombre de générations	Evaluation			N° Offres sélectionnées/ Fréquence d'apparition
	Minimale	Moyenne	Maximale	
10	4.8100	4.9912	5.1400	{5 (7/20), 6 (12/20), 10 (1/20)}
20	4.8000	4.9238	5.0200	{5 (5/20), 6 (13/20), 10 (2/20)}
30	4.8000	4.9000	5.0600	{5 (11/20), 6 (9/20)}
40	4.8000	4.9237	5.2400	{5 (8/20), 6 (12/20)}
50	4.8000	4.9213	5.0800	{5 (7/20), 6 (13/20)}
100	4.8000	4.9500	5.2800	{5 (16/20), 6 (3/20), 10 (1/20)}
150	4.8000	4.9300	5.2400	{5 (3/20), 6 (16/20), 10 (1/20)}
200	4.8000	4.9175	5.0600	{5 (9/20), 6 (11/20)}
250	4.8000	4.9300	5.1400	{5 (10/20), 6 (8/20), 10 (2/20)}

Les résultats présentés ci-dessus, correspondent à la forme graphique suivante (Fig.4.8) :

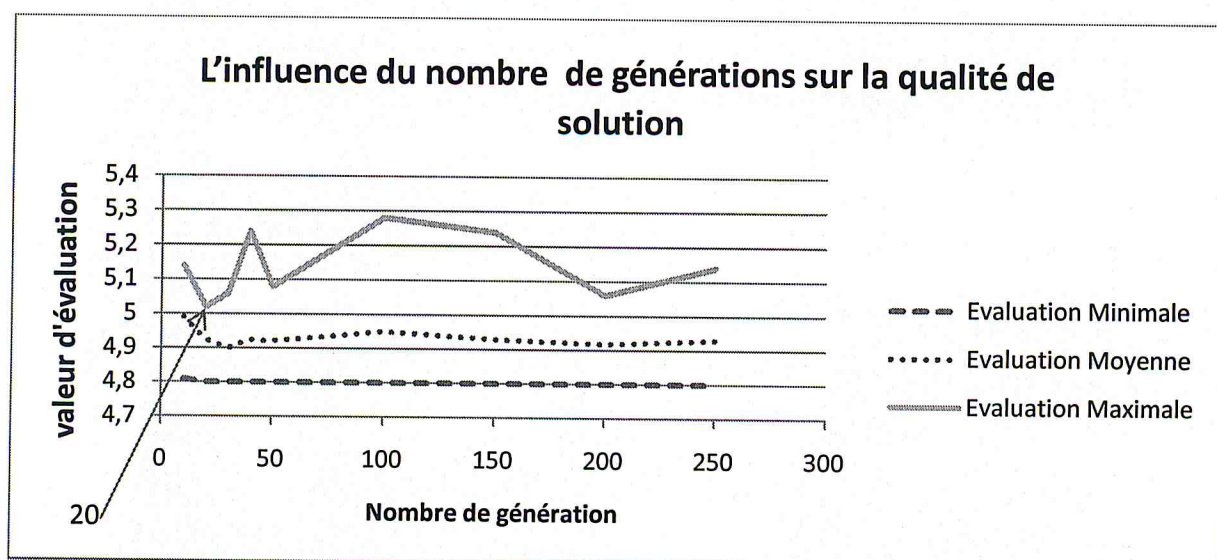


Fig.4.8: L'influence du nombre de générations sur la qualité de solution

Discussion du graphe :

Le nombre de génération idéal correspond au minimum de la courbe d'évaluation maximale, d'après le graphe (**Fig.4.8**) le nombre de génération idéal est de 20.

Point de vu temps de calcul, ce nombre de génération est très avantageux (car le temps de calcul et le nombre de génération sont proportionnels).

2.2.3 L'influence du taux de mutation

Afin de montrer l'influence du taux de mutation sur la qualité de la solution, nous avons fixé le nombre d'individus à 20, le nombre génération à 20 et le taux de croisement à 0.5. On fait varier le taux de mutation entre 0.01 et 0.1.

Tab.4.7: l'influence du taux de mutation sur la qualité de solution

Taux de mutation	Evaluation			N° Offres sélectionnées/ Effectif d'apparition
	Minimale	Moyenne	Maximale	
0.01	4.8000	4.9612	5.0950	{5 (10/20), 6(8/20), 10 (2/20)}
0.02	4.8100	4.9100	5.0400	{5 (10/20), 6 (8/20), 10 (2/20)}
0.03	4.8200	4.9750	5.1600	{5 (14/20), 6 (3/20), 10 (3/20)}
0.04	4.8000	4.9150	5.1600	{5 (10/20), 6 (9/20), 10 (1/20)}
0.05	4.8250	4.9375	5.0900	{5 (3/20), 6 (17/20)}
0.06	4.8000	4.9600	5.3000	{5 (9/20), 6 (6/20), 10 (5/20)}
0.07	4.8000	4.9350	5.1800	{5 (14/20), 6 (6/20), 10 (2/20)}
0.08	4.8050	4.9350	5.200	{5 (6/20), 6 (14/20)}
0.09	4.8000	4.9200	5.0900	{5 (4/20), 6 (15/20), 10 (1/20)}
0.1	4.8000	4.8850	5.0800	{5 (11/20), 6 (8/20), 10 (1/20)}
0.2	4.8150	4.9200	5.2000	{5 (11/20), 6 (9/20)}

Les résultats présentés ci-dessus, correspondent à la forme graphique suivante (Fig.4.8) :

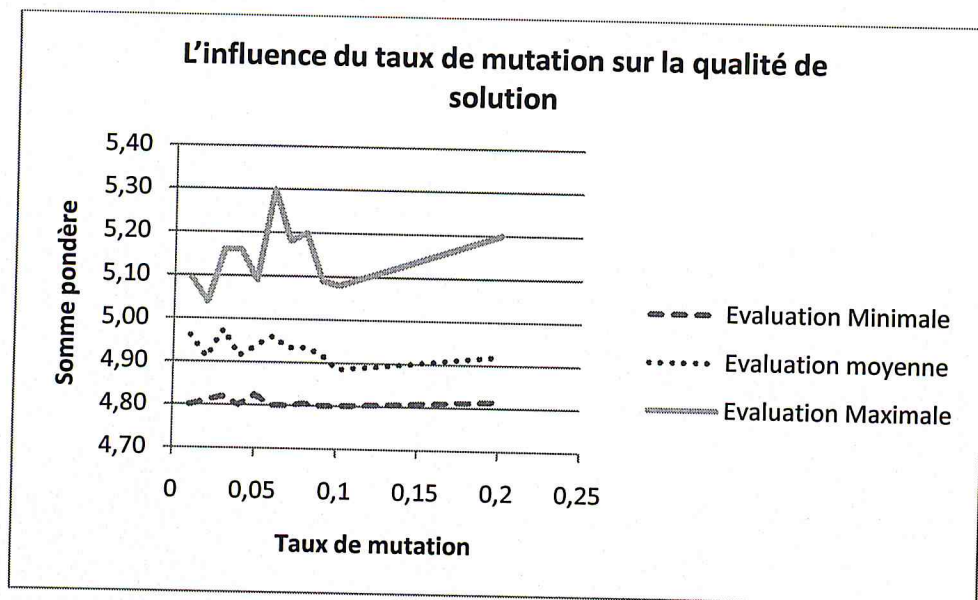


Fig.4.9: L'influence du taux de mutation sur la qualité de solution

Discussion du graphe :

Le taux de mutation idéal correspond au minimum à la différence minimale entre la courbe d'évaluation maximale et la courbe de moyenne des évaluations. D'après le graphe (Fig.4.9) taux de mutation est de 0.08 ou 0.09

2.2.4 L'influence de taux de croisement :

Afin de montrer l'influence du taux de mutation sur la qualité de la solution, nous avons fixé le nombre d'individus à 20, nombre génération à 20 et le taux de mutation à 0.08. On fait varier le taux de croisement entre 0.1 et 0.9.

Les résultats obtenus sont récapitulés dans le tableau (Tab.4.8) :

Tab.4.8: l'influence du taux de croisement sur la qualité de solution

Taux de croisement	Evaluation			N° Offres sélectionnées/ Effectif d'apparition
	Minimale	Moyenne	Maximale	
0.01	4.8250	4.9250	5.2400	{5 (8/20), 6 (11/20), 10 (1/20)}
0.1	4.8300	4.9725	5.2600	{5 (8/20), 6 (11/20), 10 (1/20)}
0.2	4.8000	4.9600	5.1800	{5 (11/20), 6 (5/20), 10 (4/20)}
0.3	4.8000	4.9425	5.1400	{5 (11/20), 6(6/20), 10 (3/20)}
0.4	4.8150	4.9500	5.2400	{5 (8/20), 6 (12/20)}
0.5	4.8150	4.9387	5.2400	{5 (8/20), 5 (8/20), 10 (4/20)}
0.6	4.8000	4.9275	5.0800	{5 (8/20), 6 (10/20), 10 (2/20)}
0.7	4.8000	4.9225	5.0800	{5 (8/20), 6 (10/20), 10 (2/20)}
0.8	4.8200	4.9388	5.1600	{5 (7/20), 6 (11/20), 10 (2/20)}
0.9	4.8000	4.9225	5.3200	{4 (5/20), 5 (14/20), 10 (1/20)}

Les résultats présentés ci-dessus, correspondent à la forme graphique suivante (Fig.4.10) :

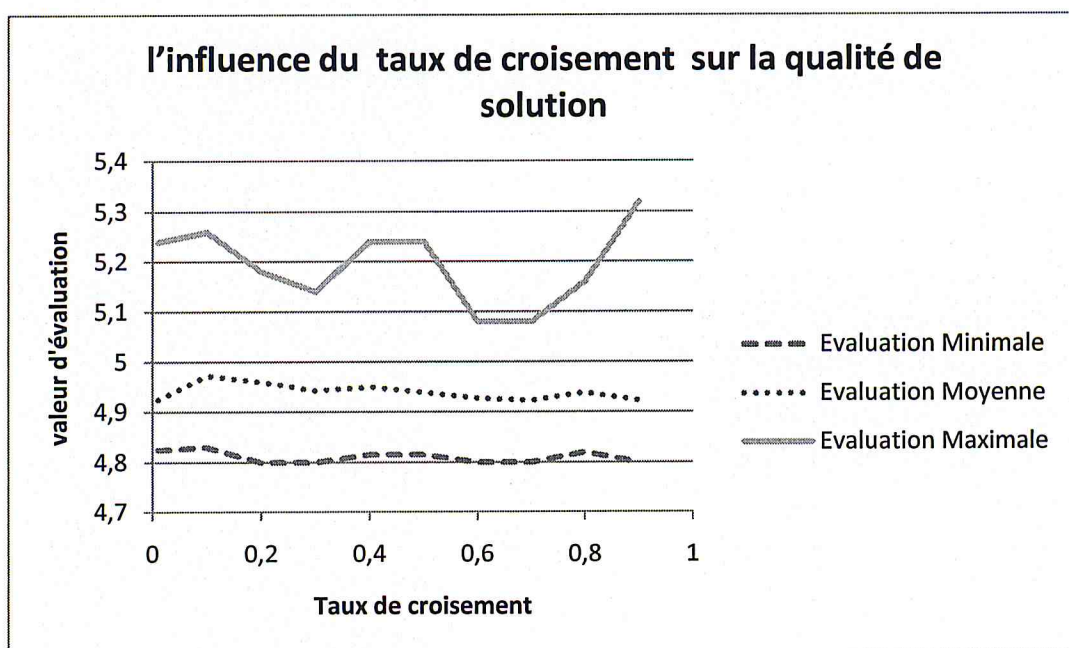
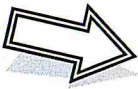


Fig.4.10: L'influence du taux de croisement sur la qualité de solution

Discussion du graphe :

Le taux de croisement idéal correspond au minimum à la différence minimale entre la courbe d'évaluation maximale et la courbe de moyenne des évaluations. D'après le graphe (Fig.4.10) le taux de croisement idéal est de 0.5.



On a présenté dans ce chapitre le logiciel correspondant à l'implémentation de l'AGMSE élaboré dans ce mémoire. Ce logiciel est en fait un outil d'aide à la décision qui peut être exploité, seul ou accompagné d'un autre outil, dans les bureaux d'études.

Il est important à noter à ce stade que la réalisation de ce logiciel n'était pas si évidente que l'on peut imaginer. Bien que l'environnement MATLAB ait une réputation d'être facile, il présente des inconvénients tels que le manque de documentation compréhensible. D'un autre côté, on a constaté que la réalisation sous l'environnement MATLAB a une philosophie tout à fait différente des environnements que nous avons utilisé durant notre cursus.

La deuxième partie de ce chapitre illustre quelques tests effectués. Évidemment, plusieurs autres peuvent être réalisés. Notons que, pour des raisons de temps et d'espace, il nous est impossible de les présenter tous. Seuls les plus significatifs ont été introduits dans ce chapitre. Notre objectif n'est pas de les réaliser tous, mais d'étudier le comportement de l'algorithme selon différentes configurations.

Les tests effectués nous ont permis de définir les valeurs par défauts de l'AGMSE. Ils nous ont permis également de toucher les aspects les plus implorants des AGs. Cependant, nous pensons que les résultats manquent de précisions, vu que le nombre de tests est relativement petit pour pouvoir généraliser les constatations. En effet, l'aspect aléatoire des AG donne un certain indéterminisme. Par conséquence, les résultats ne peuvent être validés qu'à partir d'un grand nombre d'exécutions de l'AG.

Ces travaux nous ont permis aussi de toucher l'un des inconvénients majeurs des algorithmes des AGs : le paramétrage relativement difficile de ces méthodes.

Conclusion & perspectives

Le domaine de l'optimisation multiobjectifs a suscité un grand intérêt vu son importance dans la pratique. Les méthodes proposées à la résolution des problèmes de ce domaine sont très variées, y compris les métaheuristiques du type évolutionnaire dites « Algorithmes Génétiques ».

Dans notre mémoire nous avons étudié les Algorithmes Génétiques Multiobjectifs (AGMs). Ces derniers sont la version multiobjectif des AGs classiques. Ils se sont montrés très efficaces face à de nombreux problèmes difficiles. Depuis leurs introductions, plusieurs variantes et améliorations de ces AGMs ont été proposées.

Ces algorithmes utilisent, en plus des notions standards des AGs, d'autres notions pour prendre en considération l'aspect multiobjectif. Parmi ces nouvelles notions il convient de mentionner le « front de Pareto » et la « dominance ». Ces derniers interviennent délicatement dans la définition de plusieurs éléments essentiels des AGMs, on cite parmi les quelles : les opérateurs de reproduction (croisement et mutation) et la fonction d'évaluation.

La consécration de notre travail est l'élaboration d'un AGM assez efficace, en termes de qualité de solution, pour la détermination de poids relatifs de critères au sein du processus complet du problème de sélection des entreprises. La sélection se réfère pour l'octroi de marchés. Un problème multiobjectif complexe à grande importance dans le domaine économique.

Notre logiciel AGMSE est un outil simple à utiliser. Il peut servir d'outil d'aide à la décision dans le bureau d'étude. Son rôle primaire est d'éviter à l'ingénieur de pondérer les critères et donc de ne pas restreindre au préalable son espace de recherche.

Le plus grand obstacle que nous avons rencontré est bien celui de l'ajustement des paramètres de la méthode. En effet, malgré leurs nombreux avantages, les AGMs (tout comme les AGs) nécessitent un temps très important de tests afin d'en déterminer la configuration paramétrique notamment la taille de la population, le taux de mutation et de croisement. Il faut mentionner également le caractère indéterministe des AGs, qui est un

inconvenient majeur : comme les opérateurs d'un AGs se basent essentiellement sur l'aléatoire, un AG peut se comporter différemment pour des paramètres et population identiques. Afin d'évaluer correctement l'algorithme il faut l'exécuter plusieurs fois et analyser statistiquement les résultats. Les tests effectués sur l'algorithme nous ont permis de définir l'influence des différents paramètres de l'AG.

Nous pouvons annoncer à ce stade que les objectifs fixé au début de cette étude sont atteint : on a pu vérifier la possibilité d'appliquer les AGs au sein du processus de sélection des entreprises ensuite nous avons élaboré l'AGMSE qui est l'AGM dédié à la résolution du sous-problème de ce processus complexe (la pondération des critères).

Cependant, nous pensons qu'un certain nombre de points restent à explorer, tel que la mise en œuvre des techniques spécifiques aux AGMs tel que le sharing, le niching...

Les résultats obtenus à partir de cette étude nous permettent de dégager plusieurs perspectives intéressantes qui se résument par : une étude plus complète des aspects des AGMs.

Du point de vue pratique, une perspective très évidente serait d'automatiser le processus de sélection de fournisseurs en entier pour avoir un outil d'aide à la décision complet. Ceci revient à avoir une « méthodologie » de sélection qui nécessitera l'utilisation de plusieurs autres méthodes appropriées à chaque étape de ce processus de sélection.

BIBLIOGRAPHIE

- AGU06 Aicha Aguezzoul et Pierre Ladet, SELECTION ET EVALUATION DES FOURNISSEURS : CRITERES ET METHODES, Manuscrit auteur, publié dans "Revue Française de Gestion Industrielle 2, 2006
- ALL 94 Alliot & Schiex, " Intelligence Artificielle & Informatique Théorique", Cépadués- éditions 1994.
- BAE07 M. Oscar Baez Senties, METHODOLOGIE D'AIDE A LA DECISION MULTICRITERE POUR L'ORDONNANCEMENT D'ATELIERS DISCONTINUS, THESE de doctorat, L'INSTITUT NATIONAL POLYTECHNIQUE DE TOULOUSE, 2007
- BEN05 Ahmed Chamseddine BEN ABDALLAH, Optimisation multiobjectif évolutionnaire, Mémoire de Mastère d'Ingénierie Mathématique, école polytechnique de Tunisie, 2005.
- BER01 Alain Berro, Optimisation multiobjectif et stratégies d'évolution en environnement dynamique, l'Université des Sciences Sociales Toulouse I, thèse de Doctorat en informatique, Décembre 2001.
- BLA99 Blanchard, Gilles & al (1999), *La fonction achat en informatique et télécoms : matériel et maintenance*, Editions Hermès, Paris, P.231
- BOU06 BOUMEDIENE MEROUANE Hocine, Les problèmes d'ordonnancement à machine parallèles de tâches dépendantes, Mémoire de Magistère, Département de mathématiques, Université de Blida, Juillet 2006.

- BOU09 Bouvier, Christian (1990), *Audit des achats*, Editions d'organisation, Paris, P.158
- BRU98 Bruel, Olivier (1998), *Politique d'achat et gestion des approvisionnements*, 2^{ème} éd, Editions Bordas, Paris, P.298
- CAV07 Carverivière, Patrick, *Le guide de l'acheteur : la boîte à outils des bonnes pratiques d'achat*, Editions démos, Paris, 2007.
- CHA93 Chaudhry, S. S., Forst, F. G. & J. L. Zydiak (1993). Vendor selection with price breaks. *European Journal of Operational Research*, 70, 52-66.
- COR00 Corne DW, Knowles JD, Oates MJ. The Pareto envelope-based Selection algorithm for multiobjective optimization. In: *Proceedings of sixth international conference on parallel problem solving from Nature*, 18–20 September, 2000. Paris, France: Springer; 2000.
- COR99 Knowles J, Corne D. The Pareto archived evolution strategy: a new baseline algorithm for Pareto multiobjective optimisation. In: *Proceedings of the 1999 congress on evolutionary computation- CEC99*, 6–9 July 1999. Washington, DC, USA: IEEE; 1999.
- DAH03 Dahel, N. E. (2003). Vendor selection and order quantity allocation in volume discount environments. *Supply Chain Management: An International Journal*, 8 (4), 335-342.
- DAM01 D'amours, S., Jabiri, A., & M. Levasseur (2001). Les processus de catégorisation des fournisseurs au sein des entreprises de classe mondiale. 4^{ème} congrès de génie industriel, France.
- DAV91 Davis, "Handbook of Genetic Algorithms", Van Nostrand, Reinhold, New York 1991.

- DEB00 K. Deb, S. Agrawal, A. Patrap, and T.Meyarivan. A fast and elitist non-dominated sorting genetic algorithm for multi-objective optimization: Nsga-II. In Proceedings of the 6th Conference on Parallel Problem Solving from Nature, pages 849–858. Springer Verlag, 2000.
- DEB01 De Boer, L., Labro, E., & P. Morlacchi (2001). A review of methods supporting supplier selection. *European Journal of Purchasing & Supply Management*, 7, 75-89.
- DEJ75 De Jong, ” An Analysis of the Behavior of Class of Genetic Adaptive System”, *Dissertation Abstracts International* 36(10), 5140BA, 1975.
- DIC66 Dickson, G. W. (1966). An analysis of vendor selection systems and decisions. *Journal of Purchasing*, 2 (1), 28-41.
- ELL00 Ellram, L. M. (1990). The supplier selection decision in strategic partnerships. *Journal of Purchasing & Material Management*, 26 (4), 8-14.
- ELL95 Ellram, L. M. (1995). Total cost of ownership: an analysis approach for purchasing. *International Journal of Physical Distribution & Logistics Management*, 25 (8), 4-23.
- FON93 Fonseca C., Fleming P., Genetic algorithms for multiobjective optimization: Formulation, discussion and generalisation. *Proc. Of the 5th Int. Conf. on genetics algorithms*, 416-423.
- FON93 Fonseca CM, Fleming PJ. Multiobjective genetic algorithms. In: *IEE colloquium on ‘Genetic Algorithms for Control Systems Engineering’* (Digest No. 1993/130), 28 May 1993. London, UK: IEE; 1993.
- GOL89 Goldberg. D. E., *Genetic Algorithms in Search, Optimisation and Machine*

Learning, Addison-Wesley Publishing Company, 1989

- GOL92 David E. Goldberg, Kalyanmoy Deb and Jeffrey Horn, *Massive Multimodality, Deception and Genetic Algorithms*, Ed. R. Manner, B. Manderick, *Parallel Problem Solving from Nature 2*, Brussels, p. 37-46, 1992.
- GOL94 Goldberg, "Algorithmes génétiques : exploration, optimisation et apprentissage automatique", Addison-Wesley, 1994.
- GRA08 Tristram Gräbener, Alain Berro, Optimisation multiobjectif discrète par propagation de contraintes, Manuscrit auteur, publié dans "JFPC 2008-Quatrième Journées Francophones de Programmation par Contraintes, Nantes: France (2008).
- GUI98 Guitouni A. L'Ingénierie du Choix d'une Procédure d'Agrégation Multicritère. Thèse de Doctorat, Université Laval, 1998
- GUI99 Guitouni A, Martel J-M, Vincke Ph. *Un Cadre de Référence pour le Choix d'une Procédure d'Agrégation Multicritère*. Document de travail, FSA-1999-013, Université Laval, Québec, Canada. 1999.
- HAJ92 Hajela P, Lin C-y. Genetic search strategies in multicriteria on optimal design. *Struct Optimization* 1992;4(2):99-107.
- HAM03 Abdelkader HAMMAMI , MODELISATION TECHNICO-ECONOMIQUE D'UNE CHAINE LOGISTIQUE DANS UNE ENTREPRISE RESEAU, THESE DE DOCTORAT, L'Ecole Nationale Supérieure des Mines de Saint-Etienne, France), septembre 2003

- HIN69 Hinkle, C. L., Robinson, P. J., & P. E. Green (1969). Vendor evaluation using cluster analysis. *Journal of Purchasing*, 5 (3), 49-58.
- HOL75 Holland J.H, *Adaptation in natural and artificial system*, Cambridge, Mass : MIT press, 1975.
- HON92 Hong, J. D. & J. C. Hayya (1992). Just-In-Time purchasing: Single or multiple sourcing? *International Journal of Production Economics*, 27, 175-181.
- HOR94 Horn J., Nafpliotis N., Goldberg D., 1994, A niched pareto genetic algorithm for multiobjective optimisation. *Proc. of the 1st IEEE Conf. on evolutionary computation*, 1, 82-87.
- KON06 Abdullah Konak, David W. Coit, Alice E. Smith, Multi-objective optimization using genetic algorithms: A tutorial, *Reliability Engineering and System Safety* 91 (2006) 992–1007
- KUM04 Kumar, M. Vrat, P., & R. Shankar (2004). A fuzzy goal programming approach for selection problem in a supply chain. *Computers & industrial Engineering*, 24, 69-85.
- LER00 Learman et Ngouenet, "Algorithmes génétiques séquentiels et parallèles pour une représentation affine des proximités". *Rapport de recherche N° 2570 Année 2000*.
- LIU00 Liu, J., Ding, F. Y., & V. Lall (2000). Using data envelopment analysis to compare suppliers for supplier selection and performance improvement. *Supply Chain Management: An International Journal*, 5 (3), 143-150.
- LOU97 Loubere, Jean-Michel & al (1997), *Nouvelles stratégies d'achat : sous-traitance, coopération, partenariat ?*, Editions d'organisation, Paris, P.246

- LUY03a Lu H, Yen GG. Rank-density-based multiobjective genetic algorithm and benchmark test function study. *IEEE Trans Evol Comput* 2003; 7(4):325–43.
- LUY03b Yen GG, Lu H. Dynamic multiobjective evolutionary algorithm: adaptive cell-based rank and density estimation. *IEEE Trans Evol Comput* 2003;
- MAH09 Maher HAMDI ,Connaitre tous en MATLAB, du site : <http://high-tech-matlab.blogspot.com/>, dernière mise à jours dimanche 12 juillet 2009
- MAH95 Samir W. Mahfoud, Niching Methods for Genetic Algorithms, IlliGAL TR. n° 95001, University of Illinois at Urbana- Champaign, Urbana, May 1995.
- MAN94 Mandal, A., & S. G. Deshmukh (1994). Vendor selection using Interpretive Structural Modeling (ISM). *International Journal of Operations & Production Management*, 14 (6), 52-59.
- MAS00 Masella, C., & A. Rangone (2000). A contingent approach to the design of vendor selection systems for different types of co-operative customer/supplier relationships. *International journal of Operations & Production Management*.
- MER07 MERZOUK Salah Eddine, Problème de dimensionnement de lots et de livraison : Application au cas d'une chaine logistique, Université de Technologie de Belfort-Montbéliard et Université de Franche Comté, thèse de Doctorat 2007.
- MIC92 Michalewicz, Z., *Genetic Algorithms + Data Structures = Evolution Programs*, Springer- Verlag, (1992).
- MIC94 Michalewicz, Z. (1994), "Evolutionary computation techniques for nonlinear programming problems", *International Transactions in Operational Research*

1(2), 223–240.

- MIC95 Michalewicz, Z. (1995), A survey of constraint handling techniques in evolutionary computation methods. *Proceedings of the 4th Annual Conference on Evolutionary Programming*, MIT Press.
- MUR04 Murthy, N. N., Soni, S., & S. Ghosh (2004). A framework for facilitating sourcing and allocation decisions for make-to-order items. *Decisions Sciences*, 35 (4), 609-636.
- MUR95 Murata T, Ishibuchi H. MOGA: multi-objective genetic algorithms. In: *Proceedings of the 1995 IEEE international conference on evolutionary computation*, 29 November–1 December, 1995. Perth, WA, Australia: IEEE; 1995.
- NAR83 Narasimhan, R. (1983). An analytical approach to supplier selection. *Journal of Purchasing & Materials Management*, 19 (1), 27-32.
- NIG08 Ramata NIMAGA, EVALUATION DE LA MAÎTRISE DES RISQUES LIES AU PROCESSUS D'ACHAT PAR APPEL D'OFFRES : CAS DE L'UNFPA AU SENEGAL ,*DIPLOME D'ETUDES SUPERIEURES SPECIALISEES EN AUDIT ET CONTROLE DE GESTION (DAC)*, CENTRE AFRICAIN D'ETUDES SUPERIEURES EN GESTION, Novembre 2008
- NYD92 Nydick, R. L. & R. P.Hill (1992). Using the analytic hierarchy process to structure the supplier selection procedure. *International Journal of Purchasing & Materials Management*, 28 (2), 31-36.
- OUM00 H. OUMSALEM, K. CHAUCHE, la résolution des problèmes Max-Sat & Partitionnement A l'aide de la méthode de Recherche Tabou Hybride implémentée sur une machine parallèle MIMD, mémoire d'ingénieur présenté à l'INI, 2000.
- PRE94 Preux , « Les algorithmes évolutives », Lif USTL/LIL/UL, 1994.

- RIC06 Rodolphe Le Riche, Marc Schoenauer, Michèle Sebag, Un état des lieux de l'optimisation évolutionnaire et de ses implications en sciences pour l'ingénieur, CNRS UMR 5146 et Ecole des Mines de St-Etienne, 2006.
- ROB05 Denis Robilliard, Cours sur les Algorithmes Génétiques et Evolutionnaires, université du Littoral Côte d'Opale, Décembre 2005.
- ROO97 Roodhooft, J., & J. Konings (1997). Vendor selection and evaluation: an activity based costing approach. *European Journal of Operational Research*, 96, 97-102.
- SCH85 Schaffer J., Multi-objective optimisation with vector evaluated genetic algorithms. *Proc. Int. Conf. on Genetic Algorithms and their applications*, 93-100. 1985
- SEF07 Hakim SEFIANE, Olivier SENECHAL, Application du multicritère pour l'aide à la décision en maintenance, *UNIVERSITE DE VALENCIENNES ET DU HAINAUT CAMBRESIS Laboratoire d'Automatique, de Mécanique et d'Informatique Industrielles, CPI 2007*.
- SHA85 Schaffer JD. Multiple objective optimization with vector evaluated genetic algorithms. In: *Proceedings of the international conference on genetic algorithm and their applications*, 1985.
- SIM97 Simon, Yves & al (1997), *Encyclopédie de Gestion*, 2^{ème} éd, Editions Ecominica, Paris, P.1205
- SKI95 Ng, S. T., & R. M. Skitmore (1995). CP-DSS: decision support system for contractor prequalification. *Civil Engineering System s: Decision Making Problem Solving*, 12 (2), 133-160.

- SMY93 Smytka, D. L. & M. W. Clemens (1993). Total cost supplier selection model: a case study. *International Journal of Purchasing & Materials Management*, 29 (1), 42-49.
- SOU04 Souquet Amédéen Radet Francois-Gérard, ALGORITHMES GENETIQUES, TE de fin d'année Tutorat de Mr Philippe Audebaud, 2004.
- SOU87 Soucoup, W. R. (1987). Supplier selection strategies. *Journal of Purchasing & Materials Management*, 23 (3), 7-12.
- SRI94 Srinivas N, Deb K. Multiobjective optimization using nondominated sorting in genetic algorithms. *J Evol Comput* 1994; 2(3):221-48.
- TAL95 Talbi, "*Algorithmes génétiques parallèles: Techniques et applications*".rap technique LIL, 1995.
- TIM86 Timmerman, E. (1986). An approach to vendor performance evaluation. *Journal of Purchasing & Supply Management*, 1, 27-32.
- TRA01 Tracey, M., & C. L. Tan (2001). Empirical analysis of supplier selection and involvement, customer satisfaction and firm performance. *Supply Chain Management*, 6 (4), 178-188.
- VAL04 Thomas Vallée, Murat Yıldızoğlu, représentation des algorithmes génétiques et de leurs applications en économie, Mai 2004.
- VOK96 Vokurka, R. J., Choobineh, J., & L. Vadi (1996). A prototype expert system for evaluation and selection of potential suppliers, *International Journal of Operation & Production Management*, 16 (12), 106-127.
- WEB00 Weber, C. A., Current, J., & A. Desai (2000). An optimization approach to determining the number of vendors to employ. *Supply Chain Management: An International Journal*, 5 (2), 90-98.

- WEB91 Charles A. Weber, John R. Current and W.C. Benton , Vendor selection criteria and methods, European Journal of Operational Research 50 (1991) 2-18, North-Holland.
- WEB96 Weber, C. A. (1996). A data envelopment analysis approach to measuring vendor performance. Supply Chain Management, 1 (1), 28-39
- ZIT01 Zitzler E, Laumanns M, Thiele L. SPEA2: improving the strength Pareto evolutionary algorithm. Swiss Federal Institute Technology: Zurich, Switzerland; 2001.
- ZIT99 Zitzler E., Thiele L., Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. IEEE Transactions on evolutionary computation, 3, 4, 257-271.

