ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

MÉMOIRE PRÉSENTÉ À
L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

COMME EXIGENCE PARTIELLE
À L'OBTENTION DE LA
MAÎTRISE EN GÉNIE DE LA PRODUCTION AUTOMATISÉE
M. Ing

PAR
ETSINDA MPIGA, Félicia

NOUVELLE APPROCHE HYBRIDE D'OPTIMISATION MULTIOBJECTIVE BASÉE SUR LA MÉTHODE DES SURFACES DE RÉPONSE ET LE SYSTÈME DE COLONIES DE FOURMIS

MONTRÉAL, LE 12 FÉVRIER 2008

© Etsinda Mpiga, 2008

CE MÉMOIRE A ÉTÉ ÉVALUÉ

PAR UN JURY COMPOSÉ DE :

M. Songmene Victor, directeur de mémoire
Département de Génie Mécanique à l'École de technologie supérieure

M. Dao Thien-My, codirecteur de mémoire
Département de Génie mécanique à l'École de technologie supérieure

M. Wong Tony, président du jury
Directeur du département de Génie de la production automatisée à l'École de technologie supérieure

M. Ateme Nguema Barthélémy, examinateur externe
Département des sciences de la gestion à l'Université du Québec en Abitibi Témiscamingue

IL A FAIT L'OBJET D'UNE SOUTENANCE DEVANT JURY ET PUBLIC

LE 10 JANVIER 2008

À L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

REMERCIEMENTS

Je tiens tout d'abord à exprimer ma reconnaissance à monsieur Thien-My Dao, mon codirecteur de recherche, pour son implication dans la réalisation de ce travail de recherche et pour le support qu'il m'a apporté. Sa patience, sa disponibilité, la pertinence de ses conseils, ainsi que ses encouragements lors de moments plus difficiles, m'ont été d'une aide précieuse tout au long de ce travail.

Je remercie également monsieur Victor Songmene, mon directeur de recherche, de m'avoir fait bénéficier de ses connaissances, ainsi que pour ses conseils et ses encouragements qui m'ont permis d'améliorer la qualité de mon travail.

Mes remerciements vont aussi à monsieur Barthélémy Ateme Nguema pour son aide précieuse tout au long de ce projet. Sa disponibilité et ses conseils à chaque étape de ma recherche, m'ont permis d'aller au bout de mon travail.

Je remercie aussi mes amis et collègues Neila et Christophe pour leur soutien et leurs encouragements.

Je tiens à remercier tout particulièrement mes parents et ma sœur pour leur support moral constant pendant la réalisation de cette recherche.

Pour finir, un chaleureux merci a Franck, qui a toujours été là pour moi tout au long de mes études. Merci de m'avoir encouragée et d'avoir toujours cru en moi.

Trouvez tous dans ces quelques mots l'expression de toute ma reconnaissance.

NOUVELLE APPROCHE HYBRIDE D'OPTIMISATION MULTIOBJECTIVE BASÉE SUR LA MÉTHODE DES SURFACES DE RÉPONSE ET LE SYSTÈME DE COLONIES DE FOURMIS

ETSINDA MPIGA, Félicia

RÉSUMÉ

L'environnement industriel est devenu très compétitif et exige des délais de fabrication de plus en plus courts, des coûts réduits, ainsi que des produits de bonne qualité. Ces besoins conduisent à des problèmes d'ingénierie complexes, caractérisés par de nombreux objectifs ainsi que des contraintes plus complexes. De par le grand nombre de variables mises en jeu et la nécessité d'utiliser des logiciels pour les calculs des contraintes, ce processus d'optimisation est coûteux en temps de calcul et en expérimentation. Une des possibilités de réduction des coûts vient de l'introduction de la méthode des surfaces de réponse dans le processus d'optimisation.

L'objectif principal de notre recherche est le développement d'un nouvel outil efficace d'optimisation et d'analyse. Nous avons développé une méthodologie souple et robuste, capable de résoudre des problèmes complexes. Le terme « optimisation » est très répandu, mais beaucoup de ceux qui l'emploient ne disposent pas d'outils spécifiques à cette fin. Ainsi, l'ingénieur cherche-t-il toujours la performance maximale, sans renoncer aux contraintes de coût minimum du projet.

Pour ce faire, nous proposons une nouvelle approche multiobjective combinant un outil de simulation à la modélisation avec la méthode des surfaces de réponse et aux algorithmes des colonies de fourmis (ACO). Le modèle d'optimisation développée est appliqué à l'optimisation d'un procédé de dessalement de l'eau de mer et à l'optimisation d'un procédé d'usinage cinq axes. Ces applications ont conduit à de grandes améliorations des résultats, de l'ordre de 30% pour le problème d'usinage, comparativement à l'usuelle fonction de désirabilité. L'approche hybride développée

constitue une technique puissante et flexible pour la recherche de solution optimale pour différents problèmes.

NOUVELLE APPROCHE HYBRIDE D'OPTIMISATION MULTIOBJECTIVE BASÉE SUR LA MÉTHODE DES SURFACES DE RÉPONSE ET LE SYSTÈME DE COLONIES DE FOURMIS

ETSINDA MPIGA, Félicia

ABSTRACT

The industrial environment became very competitive and requires shorter manufacturing lead times, reduced costs, and higher quality products. These requirements produce complex engineering problems characterized by multiple objectives as well as more complex constraints. From the high number of variables brought into play and the need for using software for calculations of the constraints, this optimization process is expensive when we must evaluate the computational time. One of the possibilities of reduction of the costs is an introduction of the response surface methodology into the optimization process.

The main objective of our research is the development of a new effective tool of optimization and analysis. We want to obtain a flexible and robust methodology, able to solve complex problems. The word "optimization" is already used in a current way, but much of those which employ it do not have tools specific to this end. Thus, the engineer always seeks the maximum performance, without giving up the constraints of minimum cost of the project.

With this intention, we propose a new approach of multiobjective optimization combining a tool for simulation with the response surfaces methodology (RSM) modeling and the ant colonies optimization (ACO). The model of developed optimization is applied to the optimization of a desalination process of sea water and to the optimization of five axes machining process. These applications led to great improvements of the results, approximatively 30% for machining problem,

comparatively with the usual desirability function. The developed hybrid approach thus constitutes a powerful technique for the search of optimal solution to various problems.

TABLE DES MATIÈRES

Page

INTRODUCTION 1

CHAPITRE 1 REVUE DE LITTÉRATURE 3
1.1 Introduction 3
1.2 Méthodes usuelles d'optimisation multiobjective 3
1.2.1 Méthodes d'optimisation 3
1.2.2 Problématique de l'optimisation multiobjective (MOO) 4
1.3 Les plans d'expérience pour l'optimisation multiobjectif 10
1.4 Revue de la littérature. 11
1.4.1 Méthode des surfaces de réponse et fonction de désirabilité 11
1.4.2 Méthode des surfaces de réponse modifié pour s'adapter à l'optimisation multiobjective 12
1.4.3 Méthode des surfaces de réponse associée aux méthodes usuelles de résolution des MOO 13
1.5 Conclusion 15

CHAPITRE 2 LA MÉTHODE DES SURFACES DE RÉPONSE 16
2.1 Introduction à la planification expérimentale 16
2.1.1 Objectifs de la planification expérimentale 16
2.1.2 Choix d'une stratégie expérimentale 20
2.1.3 Démarche expérimentale 20
2.1.4 Les étapes de l'expérimentation 22
2.2 La méthodologie des surfaces de réponse 25
2.2.1 Les principes généraux 25
2.2.2 Étapes de la méthodologie de surface de réponse 26
2.3 La méthode des surfaces de réponse (RSM) pour l'optimisation en ingénierie.. 32
2.3.1 Méthode graphique 32
2.3.2 Méthodes numériques 32

CHAPITRE 3 L'APPROCHE DE L'ALGORITHME DES COLONIES ..... DE FOURMIS (ACO) 38
3.1 Introduction 38
3.2 Définition et origines de l'ACO 39
3.3 Méthode usuelle d'optimisation avec l'ACO 41
3.3.1 Description de l'algorithme de base « Ant system » (AS) 41
3.3.2 Les variantes de l'ACO 44
3.3.3 Autres problèmes combinatoires 47
3.4 Extension de l'ACS dans le domaine continu 48
3.4.1 Problèmes d'adaptation 49
3.4.2 L'algorithme CACO 50

3.4.3 Une méthode hybride 52
3.4.4 L'algorithme API 54
3.4.5 ACO pour l'optimisation continue 55
3.4.6 L'algorithme CACS 57
3.5 Conclusion 58

CHAPITRE 4 MÉTHODE HYBRIDE D'OPTIMISATION PROPOSÉE 59
4.1 Introduction 59
4.2 Approche d'optimisation proposée 60
4.3 Conclusion 76

CHAPITRE 5 APPLICATION NUMÉRIQUE À DES PROBLÈMES VARIÉS 77
5.1 Introduction 77
5.2 Application à un problème industriel à deux objectifs : .....
Dessalement instantanée 77
5.2.1 Description du procédé étudié 77
5.2.1.1 Le procédé de distillation à détentes étagées .....
(Multi-Stage Flash distillation MSF). 77
5.2.1.2 Facteurs qui influent sur les coûts de production 79
5.2.2 Définition du problème 80
5.2.3 Modélisation avec RSM 81
5.2.3.1 Réponses du système 81
5.2.3.2 Les facteurs indépendants 81
5.2.3.3 Le plan d'expérience 82
5.2.3.4 Formulation des modèles mathématiques 86
5.2.4 Optimisation multiobjective 90
5.3 Application à un problème industriel multiobjectifs (à 5 objectifs) : .....
Usinage à 5 axes des aciers à moule 97
5.3.1 Description du problème 97
5.3.2 Modélisation avec RSM 98
5.3.3 Optimisation multiobjectif 109
5.4 Conclusion 114

CONCLUSION 116

ANNEXE I ARTICLE DE CONFERENCE APIEMS & CIIE - .....
DÉCEMBRE 2007- CHINE 118

BIBLIOGRAPHIE 135

LISTE DES TABLEAUX

Page

Tableau 2.1 Étapes de l'expérimentation 22

Tableau 5.1 Paramètres de conception 80

Tableau 5.2.A Matrice des expériences et résultats expérimentaux des réponses. ..... [Abdul-Wahab et Abdo, 2007] 84

Tableau 5.2.B Matrice des expériences et résultats expérimentaux des réponses. ..... [Abdul-Wahab et Abdo, 2007] 85

Tableau 5.3 Légende 88

Tableau 5.4 Les contraintes sur les objectifs de l'étude 90

Tableau 5.5 Les meilleures solutions 93

Tableau 5.6 Résultats de CACO multiobjectifs versus la fonction de ..... désirabilité 95

Tableau 5.7 Domaine des variables [Amtout Y., 2003] 99

Tableau 5.8.A Matrice des expériences et résultats expérimentaux des réponses ..... [Amtout Y., 2003] 100

Tableau 5.9.B Matrice des expériences et résultats expérimentaux des réponses ..... [Amtout Y., 2003] 101

Tableau 5.10 Facteur de corrélation de la modélisation de chaque réponse 103

Tableau 5.11 Les contraintes sur les objectifs de l'étude 110

Tableau 5.12 Les meilleures solutions (optimales) 112

Tableau 5.13 Résultats de méthode proposée versus la fonction de désirabilité .. 114

LISTE DES FIGURES

Page

Figure 1.1 Les différentes méthodes d'optimisation 4

Figure 1.2 Difficultés additionnelles de l'optimisation multicritère ..... associée à l'espace des objectifs [Avila S. L., 2006] 7

Figure 2.1 Schématisation d'un processus expérimental..... 18

Figure 2.2 Démarche générale de la méthodologie expérimentale..... 21

Figure 2.3 Étapes de la méthodologie de surface de réponse. 26

Figure 3.1 Expérience de sélection du chemin le plus court par une colonie..... de fourmis. [Dorigo M., Maniezzo V., Colorni A., 1996] 40

Figure 3.2 Algorithme de colonies de fourmis de base : le « Ant System ». 42

Figure 3.3 Fonctionnement de l'algorithme CACO : Les fourmis globales (a) ..... participent au déplacement des régions que les fourmis locales (b) ..... évaluent. 51

Figure 3.4 Algorithme de colonies de fourmis hybride en domaine continu 53

Figure 3.5 Fonctionnement de l'algorithme API : Les fourmis (cercles pleins) ..... explorent des sites de recherche (petits carrés) ..... dans un périmètre (grand cercle) autour du nid..... 55

Figure 3.6 ACO pour l'optimisation continue..... 56

Figure 4.1 Vecteur idéal Z^* versus vecteur Nadir Z^{nad}. 65

Figure 4.3 Organigramme de l'approche hybride d'optimisation 76

Figure 5.1 Principe de fonctionnement d'un système par détentes successives ..... (MSF) à 3 étages 79

Figure 5.3 Modélisation avec RSM, représentation graphique des réponses..... du plan..... 86

Figure 5.4 Diagramme Pareto du taux de production d'eau distillé (DF) 87

Figure 5.5: Diagramme Pareto du taux de perte d'écoulement (BDF)..... 88

Figure 5.6 Graphiques des effets des principaux facteurs sur les objectifs de l'étude 90

Figure 5.7 Valeurs de la fonction de fitness selon le nombre d'itération 92

Figure 5.8 Marge de valeurs des variables pour les solutions optimales 94

Figure 5.9 Représentation graphique des valeurs de BDF et DF des solutions optimales 96

Figure 5.10 Modélisation avec RSM Représentation graphique des réponses du plan 102

Figure 5.11 Diagramme Pareto de la rugosité moyenne dans la direction perpendiculaire à l'avance (Ra) 104

Figure 5.12 Diagramme Pareto de la rugosité maximale dans la direction perpendiculaire à l'avance, (Rt) 104

Figure 5.13 Diagramme Pareto de la rugosité moyenne dans la direction de l'avance, (Ra //) 105

Figure 5.14 Diagramme Pareto du temps de cycle réel (Tc) 105

Figure 5.15 Diagramme Pareto du temps d'usinage réel (Tu) 106

Figure 5.16 Graphiques des effets des principaux facteurs sur les objectifs de l'étude 109

Figure 5.17 Valeurs de la fonction de fitness selon le nombre d'itération 112

Figure 5.18 Emplacement de la solution optimale sur la marge de chaque variable 113

LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

RSM	Méthode des surfaces de réponse
ACO	Optimisation de colonies de fourmis
ACS	Système de colonie de fourmis
MOO	Optimisation multiobjective
TSP	Problème de chemin optimal
QSA	Problème d'allocation quadratique
CSP	Problème de résolution de contraintes
MRO	Optimisation de surfaces multi réponses
AG	Algorithme Génétique
ANN	Réseaux de neurones artificiels
AS	Système de fourmis
MSF	Multi Stage Flash distillation
DF	Production d'eau distillée
BDF	Taux de perte d'écoulement
SWIT	Température d'entrée de l'eau de mer
TD	Différence de température
LSBL	Dernier niveau de salinité
FSBL	Premier niveau de salinité
BRPF	Taux d'écoulement de la pompe de recyclage d'eau salée
PQRSM	Méthode quadratique progressive de surface de réponse
SAO	Optimisation par approximation séquentielle

INTRODUCTION

L'optimisation numérique est une approche de recherche de la meilleure solution s'appliquant tant en ingénierie qu'en fabrication et en assemblage. Un grand nombre d'algorithmes relatifs à cette problématique a été développé pour des modèles à objectif unique. Toutefois, due à la complexité des problèmes auxquels font face les ingénieurs, les chercheurs ont axé leurs travaux sur des algorithmes d'optimisation pour des problèmes multiobjectifs. Il s'agit alors de problème d'optimisation ayant plusieurs objectifs et dans plusieurs cas une ou des contraintes multiples. Dans les procédés d'optimisation en ingénierie, nous nous heurtons souvent à des évaluations coûteuses des fonctions objectives. C'est particulièrement le cas lorsque ces fonctions de coût et leurs contraintes sont obtenues par l'intermédiaire de simulations par éléments finis impliquant des mailles fines, un grand nombre de degrés de liberté, des comportements géométriques divers et non linéaires. Pour pallier à cela, la méthode des surfaces de réponse (RSM) est employée. Elle constitue une alternative pour remplacer le modèle complexe par une approximation basée sur des résultats calculés en quelques points de l'espace de recherche.

Une fois qu'une modélisation adéquate du problème à l'étude est obtenue à l'aide de l'approche RSM, il faut se pencher sur l'étape d'optimisation. La méthode de recherche de la meilleure solution évalue plusieurs objectifs simultanément. Certains de ces objectifs étant contradictoire, il s'agit donc d'établir un compromis. La littérature courante montre qu'il est habituel d'utiliser des fonctions de désirabilité ou des métaheuristiques dont la plus usuelle est l'algorithme génétique.

L'algorithme des colonies de fourmis (ACO) est une métaheuristique qui a été utilisée avec succès pour résoudre plusieurs problèmes d'optimisation combinatoire. Mais nous remarquons que pour le type de problèmes multi-objectifs qui nous intéresse, l'optimisation à l'aide d'ACO est très peu documentée. Aussi certains travaux nous

laissent penser que les colonies de fourmis nous conduisent plus rapidement à un optimum que les algorithmes génétiques. Les colonies de fourmis sont presque exclusivement utilisées dans des problèmes de chemin optimal (TSP), d'allocation quadratique (QSA), d'ordonnancement et de résolution des contraintes (CSP), de conception de systèmes manufacturiers, et pour des problèmes à variables discrètes et d'optimisation combinatoire.

Dans ce mémoire, une approche hybride combinant la RSM et une extension de l'ACS dans le domaine continu est proposée pour la résolution de problème de d'optimisation multiobjective (MOO) continu. La modélisation par RSM est utilisée pour déterminer les fonctions objectives du problème et l'ACS est alors employé pour la recherche de la solution « optimale ».

Ce mémoire s'amorce par une présentation des problèmes multiobjectifs (chapitre 1) suivie par la description de la RSM et de son processus de modélisation (chapitre 2), puis la description de l'ACO (chapitre 3) et vient alors la présentation de notre approche d'optimisation hybride (chapitre 4). Pour terminer, des applications sont présentées (chapitre 5) pour prouver l'efficacité de l'approche développée dans ce mémoire.

CHAPITRE 1

REVUE DE LITTÉRATURE

1.1 Introduction

L'optimisation de procédé a fait l'objet de nombreux travaux. Grâce à ces études, la performance et l'efficacité des algorithmes d'optimisation ont beaucoup évolué et sont largement appliquées dans différentes disciplines. Ce type d'optimisation porte très souvent sur de nombreux critères, il s'agit alors d'une optimisation multiobjective. Pour mener à bien un tel processus d'optimisation il faut, dans un grand nombre de cas, passer par une phase d'expérimentation qui est coûteuse en temps et en argent. Il est donc important de mettre en place un processus d'expérimentation efficace, qui permet de retirer le maximum d'informations. Plusieurs chercheurs ont appliqué les plans d'expériences, plus précisément, la méthode des surfaces de réponse, à l'optimisation dans le domaine de l'ingénierie. Nous orienterons nos travaux de recherche dans cet axe.

Dans ce chapitre, une analyse de la littérature dans ce domaine permet d'en apprendre plus et de faire ressortir notre contribution.

1.2 Méthodes usuelles d'optimisation multiobjective

1.2.1 Méthodes d'optimisation

Actuellement, il existe plusieurs méthodes d'optimisation. Chacune d'elles permet d'obtenir de meilleurs résultats selon le type de problème traité. Le choix d'une méthode dépend d'une série de caractéristiques du problème à optimiser, et surtout du comportement de la fonction qui le représente, comportement habituellement difficile à

déterminer. Pour faire le choix de la méthode d'optimisation, il est aussi nécessaire d'avoir une bonne connaissance des outils d'optimisation existants. La figure 1.1 présente ces différents outils d'optimisation et le type de problème pour lequel chacun peut être utilisé.

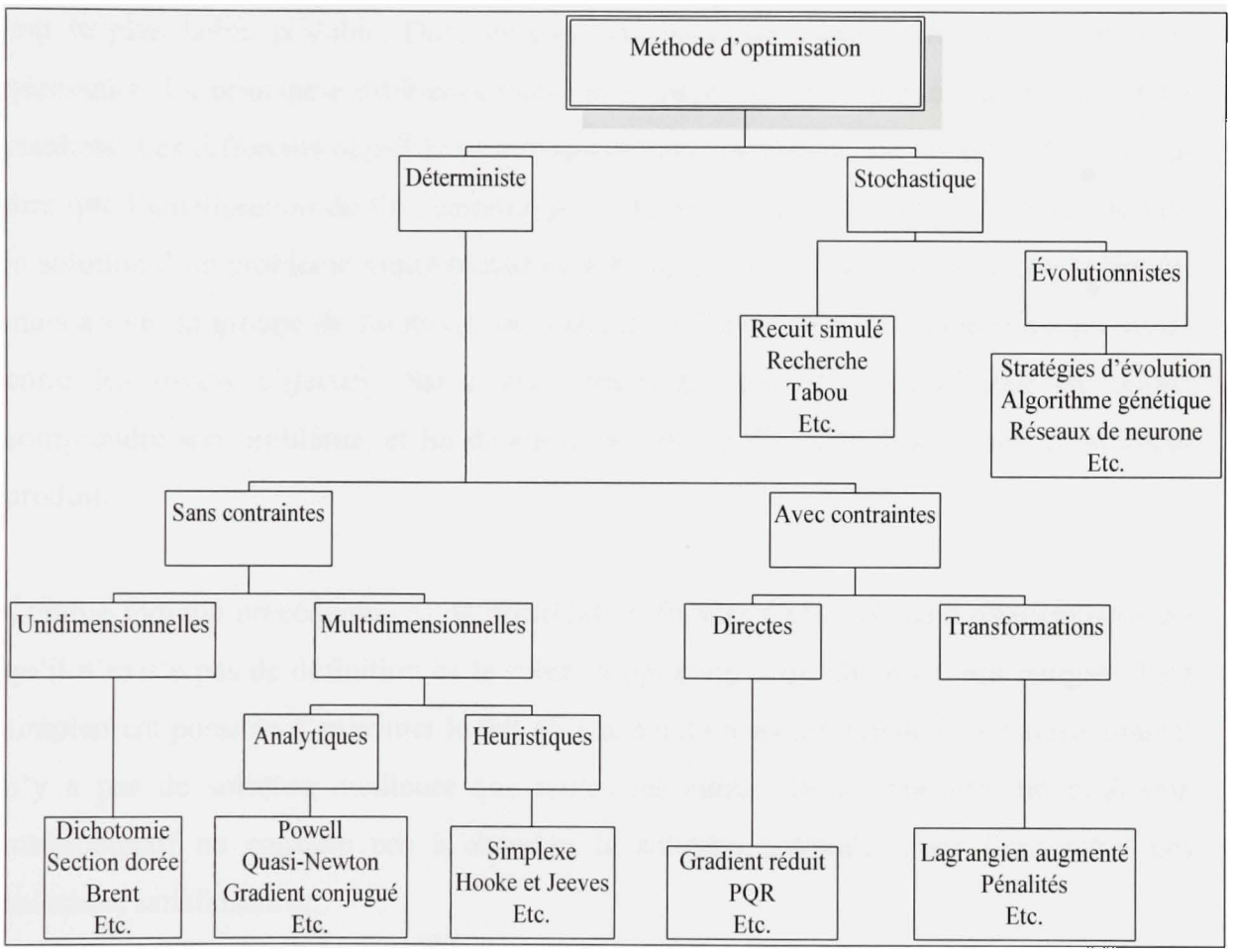


Figure 1.1 ***Les différentes méthodes d'optimisation.***

1.2.2 Problématique de l'optimisation multiobjective

Un point important doit être souligné, concernant l'optimisation en général. Dans la plupart des problèmes, la solution optimale ne prend pas en compte une seule caractéristique à minimiser ou maximiser (approche monocritère), mais plusieurs.

Normalement, ces caractéristiques devraient être considérées simultanément pendant la recherche de la meilleure solution. À titre d'exemple, un ingénieur ne doit pas concevoir un dispositif en pensant exclusivement à en obtenir la meilleure performance. Il est aussi nécessaire que le produit final respecte divers critères comme un niveau de bruit admissible, une consommation d'énergie minimale ou encore, que le coût de fabrication soit le plus faible possible. Dans ce cas, une approche multicritère du problème est nécessaire. La principale différence dans cette approche est la manière de présenter les résultats. Les différents objectifs sont fréquemment concurrents ou antagonistes, c'est-à-dire que l'amélioration de l'un entraîne la détérioration de l'autre et vice-versa. De fait, la solution d'un problème multiobjectif ne correspond pas à une seule solution optimale, mais à tout un groupe de solutions, qui caractérise l'ensemble des compromis possibles entre les divers objectifs. Saisir ces compromis permet à l'ingénieur de mieux comprendre son problème, et lui donne la possibilité d'obtenir finalement un meilleur produit.

Comme indiqué précédemment, la principale difficulté d'un problème multiobjectif est qu'il n'existe pas de définition de la solution optimale et qu'elle n'est pas unique. Il est simplement possible d'exprimer le fait qu'une solution est préférable à une autre, mais il n'y a pas de solution meilleure que toutes les autres. Donc, résoudre un problème multiobjectif ne consiste pas à chercher la solution optimale, mais l'ensemble des solutions satisfaisantes.

Pour répondre à ce problème, deux approches ressortent dans la littérature :

- Ramener un problème multiobjectif à un problème à un seul objectif au risque d'enlever toute signification au problème;
- Tenter d'apporter des réponses au problème en prenant en compte l'ensemble des objectifs.

La différence entre ces deux approches est que dans le premier cas, l'utilisateur intervient dès le début de la définition du problème, en octroyant des poids à chaque objectif, afin de transformer un problème multiobjectif en un problème mono-objectif. Dans le deuxième cas, l'utilisateur effectue son choix dans l'ensemble des solutions proposées par la résolution multiobjective.

La principale qualité d'une optimisation multiobjective est donc de rendre les décisions plus faciles et moins subjectives.

Habituellement dans ce type de problème, il faut prendre en compte n fonctions objectives et un ensemble de m contraintes sur les paramètres et les objectifs. La forme générale du problème est la suivante :

$$\begin{array}{rl} \text{Trouver} & x = [x_1, x_2, x_3, \ldots, x_n]^T \\ \text{qui minimise} & f(x) = \{f_1(x), f_2(x), f_3(x), \ldots, f_n(x)\} \\ \text{sujet à} & g_j(x) \leq 0 \qquad \text{pour } j = 1, m \\ & x_i^L \leq x_i \leq x_i^U \qquad \text{pour } i = 1, n \end{array} \tag{1-1}$$

Difficultés des problèmes multiobjectifs

Comme dans l'optimisation à un seul objectif, les difficultés de la résolution des problèmes multiobjectifs découlent de la présence des contraintes et du comportement des fonctions objectives. Les principales difficultés de ce type de problème sont présentées à la Figure 1.2 soit problèmes de convexité, discontinuités et multimodalité (optimaux locaux et/ou globaux multiples) des fonctions objectives.

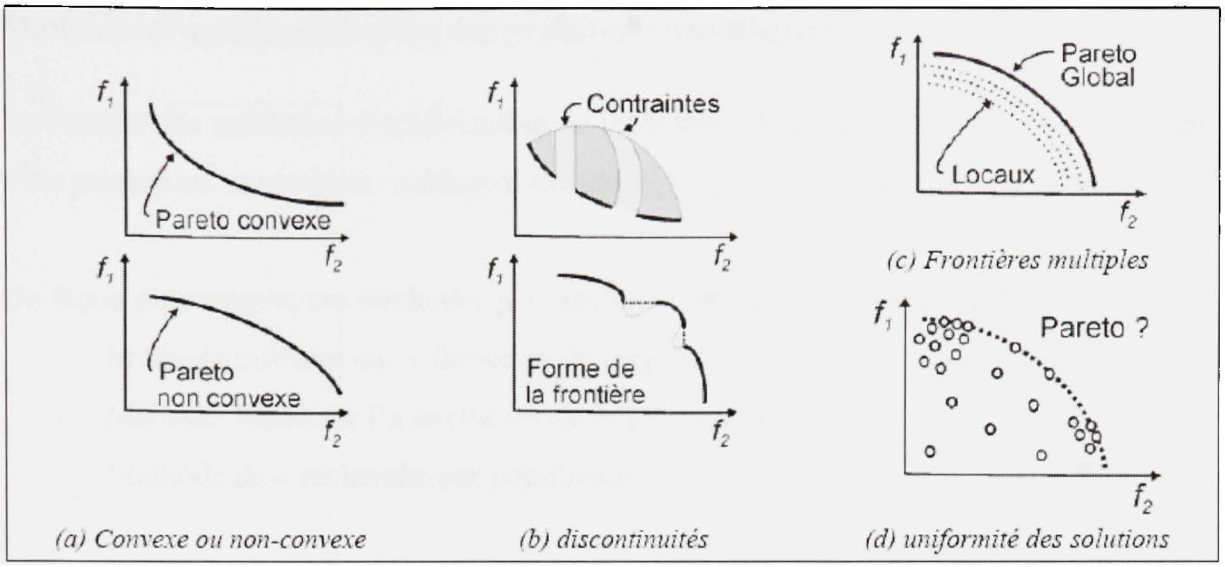


Figure 1.2 ***Difficultés additionnelles de l'optimisation multicritère associée à l'espace des objectifs [Avila S. L., 2006].***

En plus de ces difficultés, on peut aussi citer la « non-uniformité » des solutions dans l'espace des objectifs. En effet, certains problèmes peuvent présenter des caractéristiques qui concentrent les solutions dans des régions données (cette « concentration des solutions » ne concerne que les méthodes à population). Si ces régions ne sont pas proches des solutions de Pareto, ou si ces régions sont juste une fraction de la frontière optimale, la caractérisation globale de l'ensemble Pareto-optimal peut être compromise, comme le montre la Figure 1.2(d). La détermination correcte du groupe de solutions efficaces est fondamentale pour comprendre l'arrangement entre les objectifs.

En ingénierie, les problèmes sont habituellement complexes, non-linéaires, et nécessitent souvent l'utilisation de méthodes numériques pour leur résolution. Ainsi, les outils de programmation non-linéaire sont les plus indiqués pour leur optimisation.

Méthodes d'optimisation pour des problèmes non-linéaires

La famille des méthodes d'optimisation de problèmes non linéaires peut être divisée en trois principaux ensembles : méthodes déterministes, stochastiques et énumératives.

De façon plus simple, ces méthodes peuvent être classées en trois ensembles :

- Méthode utilisant une « direction de recherche » ;
- Méthode basée sur l'« exclusion de semi-espaces » ;
- Méthode de « recherche par populations ».

La logique de construction de ces méthodologies conduit à des limites d'utilisation, qui font que l'efficacité de chaque méthode dépend du type de problème à résoudre.

- **Méthodes de « direction de recherche »**

Ces méthodes sont basées sur la recherche d'une suite de points dans l'espace d'optimisation ; cette recherche nécessite la connaissance d'un vecteur indiquant la direction de décroissance de la fonction, autrement dit du gradient de la fonction à minimiser (dans le cas des problèmes de minimisation). La recherche du point optimal utilise le point courant comme point d'origine pour l'itération suivante (k+1). Parmi ces méthodes nous pouvons citer, la méthode de la plus grande pente, la méthode de Newton Raphson, celle de Nelder et mead et la méthode quasi-Newton. Pour ces méthodes de résolution, la continuité et la différentiabilité des fonctions doivent être assurées pour permettre une résolution du problème et garantir la localisation du point optimal. Cependant, il n'est pas possible d'avoir la certitude qu'il s'agit d'un optimal global. Il est nécessaire de répéter le processus d'optimisation plusieurs fois, toujours avec de points initiaux différents pour améliorer la probabilité d'avoir trouvé l'optimum global.

- **Méthodes d'« exclusion de semi-espaces »**

Les méthodes d'« exclusion de semi-espaces » sont celles qui utilisent des approximations du gradient des fonctions objectives pour définir un plan qui divise l'espace des objectifs en deux demi-espaces, sachant que le gradient doit nécessairement décroître dans l'un des demi-espaces. Ce groupe compte les diverses méthodes de « plans de coupe » par des contraintes, les méthodes de points intérieurs, les méthodes ellipsoïdales, etc. Ces méthodes peuvent être appliquées à des problèmes ayant des fonctions objectives discontinues et non différentiables. Par contre, la convexité des fonctions traitées est nécessaire au fonctionnement de ces méthodes.

En ce qui a trait à la vitesse de convergence de problèmes où les méthodes d'« exclusion de semi-espaces » et les méthodes de « direction de recherche » peuvent être appliquées, il y a le plus souvent une tendance pour que ces dernières soient plus rapides. Il est alors recommandé de n'utiliser les méthodes d'« exclusion de semi-espaces » que lorsqu'il existe des non différentiabilités qui empêchent l'utilisation des méthodes de « direction de recherche ».

- **Méthodes de « recherche par populations »**

Les deux familles de méthodes présentées jusqu'ici travaillent avec une seule solution courante. La solution suivante est calculée d'après la solution courante, plus une information sur la région où elle est incluse (gradient). Au contraire, les méthodes de « recherche par populations » travaillent simultanément avec un ensemble de solutions courantes, appelées population courante. La construction de la population suivante est alors basée sur les informations obtenues en plus d'un point de l'espace des paramètres. Ces informations font référence aux valeurs des fonctions objectives, qui permettent de déterminer quelles solutions sont les meilleures ; aucune information portant sur le gradient de ces fonctions n'est utilisée. Les méthodes de « recherche par populations » les plus connues sont les Algorithmes génétiques (AG), les algorithmes de colonies de fourmis (ACO), les réseaux de neurones artificiels (ANN) etc. Le grand avantage de ces

méthodes par rapport aux autres familles est la possibilité de détecter des optimums locaux et l'optimum global. Les méthodes de recherche par populations sont celles qui approchent le mieux le concept « d'algorithme d'optimisation pour des problèmes génériques » (multimodaux, avec contraintes, convexes ou non convexes, discontinues, etc.).

Il faut noter que les méthodes de recherche par populations présentent également des limites :

- La première est que le résultat donné par ces méthodes dépend de la distribution de la population initiale. La méthode convergera difficilement si l'optimum global est éloigné de la région où se concentre la population initiale. L'efficacité de ces méthodes est donc très liée au choix d'une population initiale qui explore bien tout l'espace de recherche.

- Le second point négatif est la vitesse de convergence. Par comparaison avec d'autres méthodes, les méthodes à populations sont souvent plus « lentes », dans le sens qu'elles peuvent nécessiter un plus grand nombre d'évaluations de la fonction objective. Le coût de calcul peut de ce fait être plus élevé pour les mêmes résultats (lorsque les autres méthodes permettent d'obtenir un résultat).

Malgré ces limites, les méthodes de recherche par population sont très utilisées pour la résolution des problèmes d'optimisation multiobjectifs.

1.3 Les plans d'expérience pour l'optimisation multiobjectif

Pour mener à bien un processus d'optimisation, en plus des procédés usuels présentés ci-dessus, les plans d'expériences constituent un outil également très utilisé pour divers problèmes. En effet, pour un grand nombre de problèmes en ingénierie, la complexité

des modèles et de la modélisation associée au produit étudié (modèle par éléments finis de grandes tailles, modèles non linéaires, etc.) fait que le processus d'optimisation classique devient long et parfois non envisageable pour des raisons de coût de calcul. La méthode expérimentale la plus utilisée pour la résolution de ce type de problème est la méthode des surfaces de réponses (RSM).

Plusieurs chercheurs ont appliqué les plans d'expériences, plus précisément la RSM, à l'optimisation de procédé. Une analyse de ces travaux qui ont une incidence sur le projet que nous traitons, est nécessaire.

1.4 Revue de littérature.

1.4.1 RSM et fonction de désirabilité

La principale méthode présente dans la littérature pour l'optimisation de la conception de système complexe avec la RSM est l'optimisation par la fonction de désirabilité. Cette méthode est choisie par de nombreux auteurs, car elle est simple, rapide et surtout elle est intégrée aux logiciels de plans d'expériences, tels que JMP ou Statgraphics, pour servir à la phase d'optimisation. Parmi les travaux utilisant cette méthode, il est possible de citer Derringer et Suich [Derringer G. and Suich R., 1980] qui ont défini une fonction de désirabilité pour transformer plusieurs variables de réponse en réponse simple permettant ainsi de transformer un problème multiréponses en un problème à réponse unique qui peut être aisément optimisé. Cette méthode de résolution se retrouve également dans d'autres études telles que celle de Abdul-Wahad et Abdo [Abdul-Wahab S.A.and. Abdo J., 2007] qui ont effectué l'optimisation bi-objective d'un procédé de dessalement instantané à plusieurs étapes de l'eau de mer.

Dans le même ordre d'idée, Khuri et Conlon [Khuri, A.I. and Conlon, M., 1981] ont simultanément optimisé plusieurs réponses en utilisant des modèles de régression

polynomiale. Ils ont, premièrement, défini une fonction de distance en considérant la solution idéale et ont, par la suite, déterminé la solution optimale en minimisant cette fonction. Dans ce cas, le problème multiréponses à également été réduit avec succès à un problème a réponse unique.

1.4.2 RSM modifiée pour s'adapter à l'optimisation multiobjective

Dans un autre ordre d'idée, certains auteurs ont effectué la résolution de problèmes d'optimisation complexes avec des variantes de la RSM. En effet, ces auteurs n'ont pas utilisé la RSM dans sa forme usuelle, décrite dans la suite de notre étude, mais ils l'ont modifiée pour l'adapter à leurs problèmes d'optimisation. Il s'agit alors de « méthodes adaptatives de surfaces de réponses ». En partant d'une surface de réponse initiale, ces méthodes consistent à déterminer itérativement de nouvelles expériences afin d'enrichir le dispositif expérimental et ainsi améliorer l'approximation de la réponse. Parmi les travaux utilisant ce type de méthode, il y a ceux de Wang et Dong [Wang G.G. and Dong Z., 2000] qui présentent un nouveau schéma d'optimisation globale. La méthode des surfaces de réponse adaptative (ARSM) qu'ils appliquent avec succès sur un système mécanique complexe ; la pile à combustible PEM est basée sur le concept de radiateur. Le résultat de cette optimisation est une importante amélioration de la performance et une diminution des coûts du système. Les auteurs affirment que la méthode présentée peut être appliquée pour la résolution d'autres problèmes complexes de « conception-optimisation ». Ils le confirment en effectuant avec leur méthode, l'optimisation multidisciplinaire du godet d'adduction d'air d'un hélicoptère [Wang G.G. et Dong, Z., 2003].

Park et Moon [Park K. et Moon. S., 2005] présentent également une nouvelle méthode d'optimisation pour des problèmes d'optimisation non linéaire : la méthode quadratique progressive de surface de réponse (PQRSM), qui est un des algorithmes d'optimisation

par approximation séquentiel (SAO). Ils appliquent avec succès cette méthode à un problème d'optimisation multidisciplinaire de la conception.

Les méthodes adaptatives de surfaces de réponse permettent d'obtenir des résultats appréciables. Toutefois, le fait qu'il faille rajouter progressivement des expériences au premier modèle de surfaces de réponses obtenu constitue une lacune. En effet, selon le type de problème traité, chaque expérience additionnelle peut entraîner des coûts supplémentaires plus ou moins importants.

1.4.3 RSM associée aux méthodes usuelles de résolution des MOO

Pour ce type de résolution alliant la modélisation à l'aide de la RSM et l'optimisation simultanée des réponses, certains chercheurs ont eu recours à des méthodes de recherche par populations, dont principalement, les algorithmes génétiques, pour l'optimisation multiréponses.

Il s'agit alors d'appliquer les méthodes d'optimisation multiobjectives à l'optimisation multiréponses. Cette idée est présentée dans un article de Park et Kim [Park K.S. et Kim K.J., 2005]. Où ils affirment que l'optimisation de surface multiréponses (MRO) peut être considérée comme un problème d'optimisation multiobjectifs (MOO) et que diverses techniques développées dans le cadre de l'optimisation multiobjectives peuvent être utilisées avec succès pour traiter des problèmes d'optimisation de surface multiréponses. Ils citent plusieurs auteurs qui ont développé et appliqué des théories et des méthodologies utiles pour l'optimisation multi objectifs (MOO) à des problèmes d'optimisation multiréponses (MRO).

D'autres auteurs démontrent que les approches existantes de MRO peuvent être considérées comme des cas spéciaux des approches développées pour MOO. Ainsi Sun et Lee [Sun H. et Lee S., 2005], ont présenté dans leurs travaux une approche associant,

fructueusement, la RSM et les algorithmes génétiques (AG) pour l'optimisation de la conception aérodynamique d'une aile d'avion. Dans le même ordre d'idée, en 2004, Lanzi et al. [Lanzi L., Castelletti L.M.L, Anghileri M., 2004] ont associé la RSM et les AG pour optimiser avec succès la conception d'amortisseurs coniques à sections transversales elliptiques. De nombreuses autres applications utilisant la RSM et les AG sont présentes dans la littérature et démontrent que cette méthode d'optimisation multi objectifs permet d'optimiser avec succès des problèmes complexes. Les AG ne sont pas les seules métaheuristiques qui ont été associées à la RSM pour l'optimisation multiobjective. Il apparaît aussi, en moindre quantité, des travaux associant la RSM et les réseaux de neurones artificiels (ANN). Ces travaux consistent pour la plupart, à optimiser la conception du réseau de neurones artificiels en utilisant la RSM [Chiu C.C.; Pignatiello J.J. Jr., Cook D.F., 1994]. Mais des applications en optimisation multiréponses existent également tels que les travaux de Hsieh [Hsieh K.L., 2003] qui présentent une nouvelle approche appliquant les réseaux de neurones artificiels à des problèmes d'optimisation multiréponses.

Le système de colonies de fourmis (ACS), est une métaheuristique inspirée de l'étude du comportement des fourmis et constitue une nouvelle approche multiagents efficace pour la résolution de problème d'optimisation combinatoire [Dorigo M., 1992]. Cependant, nous avons constaté que très peu de travaux traitent de l'optimisation multiobjective avec l'ACS. Or pour des problèmes d'optimisation combinatoire, les algorithmes de colonies de fourmis ont été quelquefois employés et ont démontré qu'elles conduisent souvent plus rapidement à de meilleures solutions que les AG [Nagesh K.D., 2006; Liang Y.C., 2004] qui sont les plus utilisées. Ces travaux, démontrant la qualité des résultats fournis par ACO pour diverses applications comparativement aux autres métaheuristiques, ont suscité notre intérêt pour cette méthode comme outils d'optimisation de problème complexe en ingénierie.

1.5 Conclusion

Cette revue de littérature nous a permis d'explorer différents travaux portant sur l'optimisation de problèmes complexes d'ingénierie. Notre intérêt est axé sur l'optimisation des systèmes en recourant à l'expérimentation. Il s'agit donc de problème d'optimisation multiréponses.

L'étude des travaux sur le sujet laisse apparaître trois approches de résolution :

- La première consistante à employer la méthode des surfaces de réponses usuelle, associée à l'optimisation de la fonction de désirabilité. Cette méthode est simple et rapide, mais présente des résultats de moindre qualité.

- La deuxième approche de résolution utilise uniquement la méthode des surfaces de réponses pour la résolution complète du problème. Mais de nombreuses modifications sont apportées à la méthode usuelle pour l'adapter aux problèmes à l'étude. Il s'agit alors de méthodes adaptatives de surfaces de réponse.

- La dernière approche est celle qui associe la méthode des surfaces de réponses à des méthodes d'optimisation multiobjectives continues. Cette approche fournit des résultats appréciables pour des problèmes complexes dans différents domaines d'étude et se prête bien aux problèmes rencontrés en ingénierie.

Parmi ces méthodes d'optimisation, l'ACO nous ouvre des horizons intéressants qui nous conduisent à envisager son hybridation avec la RSM dans le but de développer une méthode d'optimisation multiobjective efficace et flexible.

CHAPITRE 2

LA MÉTHODE DES SURFACES DE RÉPONSE

2.1 Introduction à la planification expérimentale

2.1.1 Objectifs de la planification expérimentale

La planification expérimentale regroupe un ensemble de techniques statistiques destinées à analyser le comportement d'un système expérimental dans le but de comprendre et d'améliorer son fonctionnement. Dans le domaine industriel en particulier, l'utilisation des plans expérimentaux est en développement constant et peut servir de support pour l'optimisation des procédés de fabrication et de contrôle, comme dans le cas qui nous intéresse, ainsi que pour la formulation des produits.[1]

La planification expérimentale consiste à imposer aux entrées du problème des variations particulières et mesurer les variations induites aux sorties afin de déduire les relations de causes à effets. Elle peut donc être considérée comme un système donnant l'expression d'une ou plusieurs réponses dépendamment d'un certain nombre de facteurs. Les réponses de ce système sont généralement constituées de variables qui serviront à l'optimisation. Ces variables d'entrées du système qui doivent varier indépendamment les unes des autres lors de l'expérimentation, sont appelées facteurs.

---

[1] J.-N. Baléo, B. Bourges, Ph. Courcoux, C. faur-Brasquet, P. Le Cloirec. (2003). *Méthodologie expérimentale : Méthodes et outils pour les expérimentations scientifiques*. Paris : Éditions TEC et DOC.

Facteurs

Les facteurs caractérisant les conditions expérimentales peuvent être très variés. Il existe :

- Des facteurs quantitatifs continus tels qu'une température, un débit, une pression, une concentration, une vitesse etc;

- Des facteurs quantitatifs discontinus comme le nombre de vis d'un montage etc;

- Des facteurs qualitatifs comme la nature d'un procédé (discontinu ou continu), le type de climat etc.

Les facteurs peuvent également être classés selon d'autres critères dont:

- Des facteurs contrôlés, auxquels il est possible d'imposer un état déterminé à l'avance. C'est ce type de facteurs que l'expérimentateur fait varier lors du processus expérimental;

- Des facteurs mesurables, auxquels il n'est pas possible d'imposer un état donné, mais dont le niveau peut être connu précisément (exemple : température extérieure);

- Des facteurs constants, qui sont maintenus dans un état fixe tout au long des différentes expériences;

- Des facteurs bruits, dont l'effet ne s'exerce que sur la dispersion ou la qualité des résultats (Exemple : vieillissement du matériel, fatigue de l'opérateur).

Les réponses

Les réponses caractérisent ce que l'expérimentateur considère comme conséquence du phénomène et constituent des objectifs à optimiser. Elles peuvent, comme les facteurs, être de natures très variées comme:

- Des réponses quantitatives continues tels qu'un rendement, une viscosité, une résistance à la traction etc;

- Des réponses quantitatives discontinues. Il s'agit de réponses ne pouvant prendre que certaines valeurs discrètes;

- Des réponses qualitatives quantifiables ou au moins ordonnables parmi lesquelles on retrouve les appréciations, telles que "très bon", "bon", etc.);

- Des réponses binaires (0 ou 1, avec ou sans, etc.);

- Des réponses sous forme de courbe. (Exemple : cinétique, profil de température, etc.).

Le processus expérimental est résumé dans la figure 2.1 suivante :

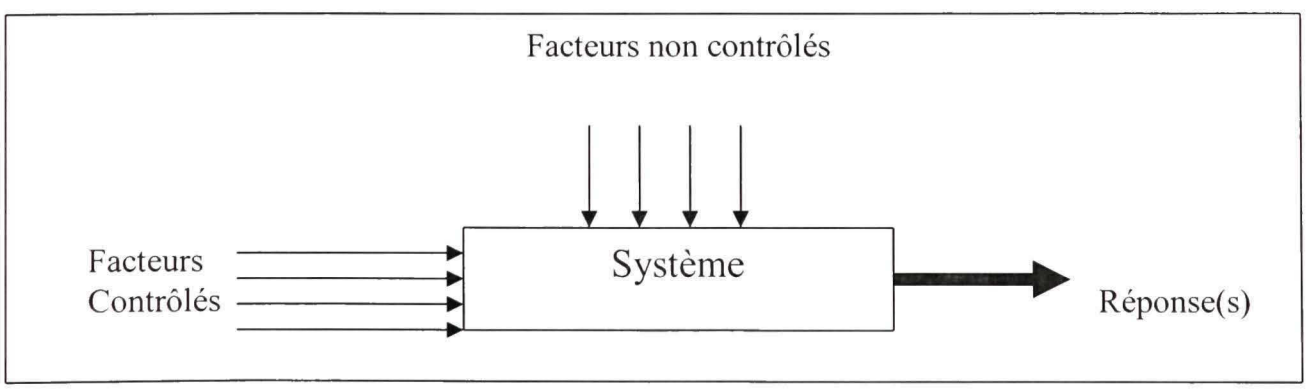


Figure 2.1 ***Schématisation d'un processus expérimental.***

Lors de l'étude d'un tel système, deux approches se distinguent :

- Une approche « déterministe » consistant à utiliser des lois théoriques pour modéliser le phénomène ;

- Une approche « empirique » considérant le système comme une « boîte noire » dont on essaie de comprendre le comportement par une ou plusieurs séries d'expériences.

L'objectif de la méthodologie des plans d'expériences est d'aider l'expérimentateur à résoudre les problèmes suivants :

- *La détermination des facteurs influant sur la réponse* : dans cette étape, appelée screening (ou criblage) de facteurs, on cherche à déterminer, parmi un nombre assez grand de facteurs, ceux qui ont une influence significative sur la réponse. Cette phase intervient donc souvent en début d'étude expérimentale.

- *La description des variations d'une réponse en fonction des fluctuations des facteurs et la validation expérimentale de ce modèle.*
  Cette phase de modélisation est une suite naturelle de la phase de screening lorsque l'objectif de l'étude est la compréhension du lien entre réponses et facteurs. La technique de choix dans ce cadre est la méthodologie de surfaces de réponse que nous allons approfondir dans la suite de cette étude. Mais il faut noter que pour un plan de surfaces de réponse, la phase de screening est éliminée, car elle est induite dans le plan de surface de réponse.

- *La détermination de la (ou des) combinaison de facteurs conduisant à une optimisation de la réponse.*
  Faisant suite à l'étape de modélisation, l'optimisation du système expérimental étudié en est le prolongement normal et utilise souvent les mêmes techniques statistiques.

2.1.2 Choix d'une stratégie expérimentale

Dans le cadre d'une expérimentation, deux stratégies expérimentales sont envisageables :

Le maillage du domaine expérimental :

Cette méthode permet d'étudier une réponse dépendant de plusieurs facteurs en réalisant un maillage du domaine expérimental et en effectuant une expérience à chaque nœud de ce réseau. L'interprétation d'une telle série d'expériences est alors aisée dans le cas d'un problème d'optimisation. Mais le nombre d'expériences à réaliser augmente de façon considérable avec le nombre de facteurs et le nombre de modalités de chacun des facteurs.

La méthode du « un facteur à la fois » :

Cette méthode est beaucoup moins coûteuse que le maillage expérimental. Elle consiste à faire varier successivement chacun des facteurs en laissant les autres constants. Le risque d'une interprétation erronée des résultats est alors élevé.

Pour pallier aux faiblesses de ces deux stratégies, la planification expérimentale permet de trouver un nombre minimum d'expérience à réaliser de façon à obtenir une modélisation fiable avec un minimum d'expérience.

2.1.3 Démarche expérimentale

Si un système dont la réponse y est une variable dépendant de p facteurs $X_1, \ldots, X_p$, le but de l'expérimentateur est en général de valider un modèle de la réponse en fonction des valeurs des facteurs influents. On utilise alors un modèle du type :

$$y = f_{\beta}(X_1, X_2, ..., X_p) + \varepsilon \qquad (2\text{-}1)$$

Où :

β: est un ensemble de paramètres caractérisant la forme de la fonction f_{β}

ε : terme d'erreur

L'objectif est d'estimer les paramètres du modèle en minimisant l'erreur ϵ. La méthodologie expérimentale doit être conçue comme un processus itératif entre le modèle et le plan expérimental. Elle permet de planifier les expériences pertinentes pour ajuster un modèle aux données et le valider[2].

La Figure 2.2 illustre cette démarche expérimentale.

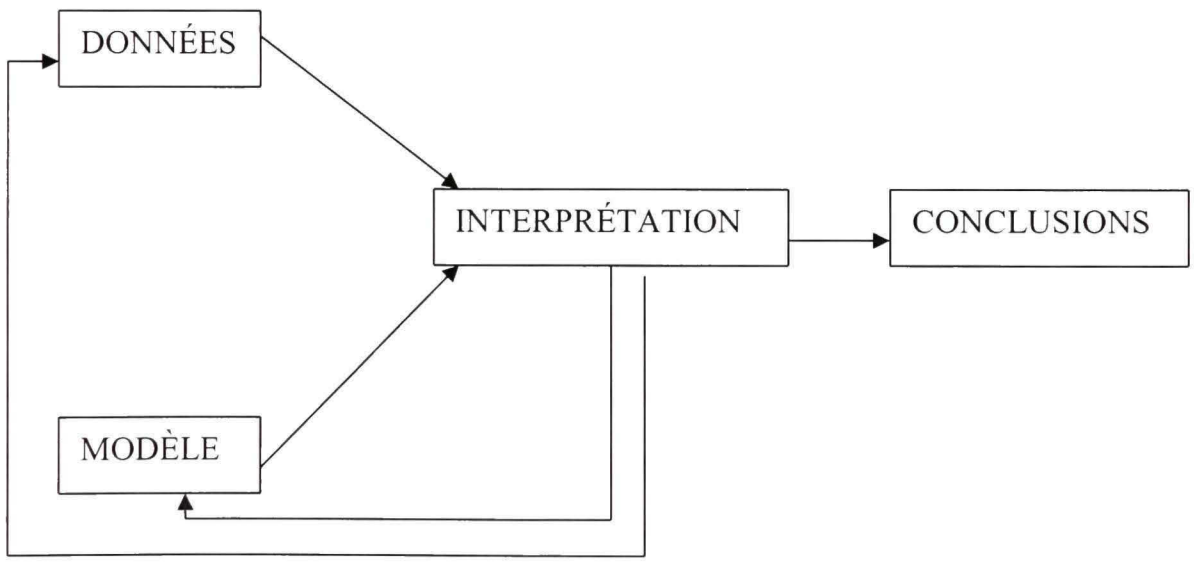


Figure 2.2 ***Démarche générale de la méthodologie expérimentale.***

[2] Box G.E.P., Hunter W.G., Hunter J.S. (1978). Statistics for experimenters. Wiley and sons, New York.

L'expérimentation consiste à réaliser séquentiellement des plans expérimentaux permettant d'estimer au mieux les paramètres du modèle recherché. À chaque étape, on procède à un test d'ajustement de ce modèle et on complète ce dernier si l'analyse montre qu'il est insuffisant pour décrire efficacement les données expérimentales. La modification du modèle envisagé doit bien entendu être accompagnée de la réalisation d'un nouveau plan d'expériences optimal pour ce modèle.

2.1.4 Les étapes de l'expérimentation

Le Tableau 2.1 présente les différentes étapes à suivre pour mener à bien une expérimentation.

Tableau 2.1

Étapes de l'expérimentation[3]

PHASES	ÉTAPES
Planification	1. Définir PROBLÈME / PROCESSUS - objectifs
	2. Choisir les variables de RÉPONSE (S) Y à mesurer
	3. Choisir les VARIABLES facteurs X et l'espace de variation
	4. Définir & comparer des DESIGNS EXPÉRIMENTAUX
Exécution	5. PRÉPARER l'expérience
	6. CONDUIRE l'expérimentation
Analyse	7. ANALYSE statistique des résultats
	8. AGIR en fonction des conclusions de l'analyse

[3] Adapté de : Professeur Bernard CLÉMENT, PhD. Cours 1. *MTH 6301 Planification et analyse statistique d'expériences.* Département de mathématiques et de génie industriel. École polytechnique de Montréal. Copyright © Génistat Conseils Inc., 2004, Montréal, Canada

Phase de planification

Cette phase consiste à définir le problème à résoudre avec ses réponses et les facteurs qui influent sur ces dernières. L'élaboration d'un design expérimental approprié conduit à l'obtention d'une matrice d'expérience. Cette matrice d'expérience est un ensemble d'expériences supposées homogènes, effectuées consécutivement, dont l'exploitation ne peut être faite que lorsque la totalité des résultats expérimentaux est obtenue. Elle doit apporter des réponses aux questions ou aux hypothèses du chercheur et est définie par :

- Le nombre d'expérience à effectuer
- La nature des expériences (niveaux des variables codées)
- L'organisation des expériences dans le temps (par lots ou séquentielles)

Phase d'exécution

Cette phase consiste à mener à bien les différentes expériences indiquées par la matrice d'expérience.

Phase d'analyse

Il s'agit d'une phase d'analyse statistique des résultats consistant à modéliser les réponses et à vérifier la validité des modèles obtenus.

- **Modélisation**

Toute stratégie expérimentale est élaborée pour établir une relation de cause à effet entre les facteurs et les réponses. Cette relation a généralement la forme d'un modèle mathématique. Ce modèle mathématique peut être théorique (c'est-à-dire qu'il est obtenu à partir d'une loi physique), ou empirique. Il peut être linéaire ou non linéaire par rapport aux coefficients qui sont les inconnues. Deux principaux types de modèle se distinguent :

- **Modèles de connaissance**

Ils dérivent des lois physiques. Afin de mieux comprendre certains phénomènes, les chercheurs ont été amenés à étudier différents mécanismes théoriques et à les traduire sous forme de modèle mathématique. En général, ces formules contiennent un grand nombre de variables et sont sous forme non linéaire. Pour ce type de modèle, l'objectif de la planification expérimentale est seulement de chercher à obtenir une estimation des coefficients de ce modèle sans chercher à le remettre en cause.

- **Modèles empiriques**

Dans ce cas, il s'agit de trouver un modèle mathématique qui permet de relier les facteurs aux réponses. Ce modèle mathématique doit avoir deux atouts majeurs. D'abord, il doit bien représenter le phénomène dans le domaine d'expérience étudié. Ensuite, il doit permettre une prévision de bonne qualité en n'importe quel point du domaine expérimental.

La méthodologie des surfaces de réponses est efficace pour effectuer cette modélisation. Les modèles les plus utilisés pour la représentation des réponses sont des polynômes, car ils permettent une démarche séquentielle. Quant aux matrices d'expériences, les plus utilisées sont les matrices composites, les matrices de Doehlert ou les matrices hybrides

2.2 La méthodologie des surfaces de réponse

2.2.1 Les principes généraux

La méthode des surfaces de réponse (RSM) est un ensemble de techniques statistiques et mathématiques utilisées pour développer, améliorer et optimiser des procédés. Elle a aussi d'importantes applications dans la conception, le développement et la formulation de nouveaux produits, ainsi que dans l'amélioration de la conception de produits existants.[4]

L'objectif de cette méthode est, plus que de hiérarchiser les effets des différents facteurs, de décrire le plus précisément possible le comportement de la réponse en fonction des variations des facteurs.

---

[4] Raymond H. Myers, Douglas C. Montgomery (1995). "Response surface methodology. Process and product optimization using designed experiments". Page 1.

2.2.2 Étapes de la méthodologie de surface de réponse

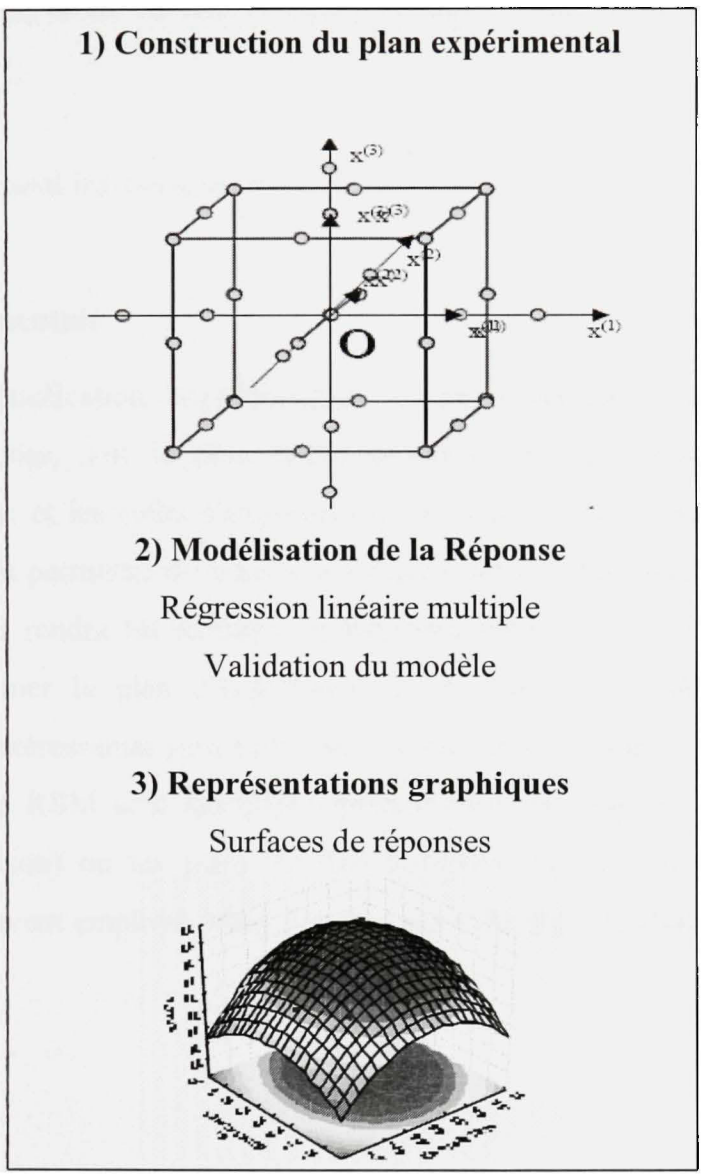


Figure 2.3 ***Étapes de la méthodologie de surface de réponse.***

Figure 2.3 schématise l'enchaînement des étapes de la méthodologie de surface de réponses. Les phases de construction du plan et de modélisation sont étroitement liées et peuvent être menées de manière itérative.

Dans une première étape, on construit un plan d'expérience optimal pour le modèle le plus simple possible. Ensuite, ce plan pourrait, éventuellement, être complété si les phases de modélisation et de validation montrent que le modèle présupposé s'avère inadéquat.

Les étapes de la RSM sont les suivantes :

Planification expérimentale

Pour une bonne planification expérimentale, il est souhaitable que le nombre d'expériences effectuées, soit le plus faible possible afin de réduire les besoins informatiques, le délai et les coûts d'expérimentation. Une bonne sélection des points pour l'expérimentation permettra de réduire la variance des coefficients de notre modèle mathématique, ce qui rendra les surfaces de réponses obtenues plus fiables. Pour ce faire, il faut déterminer le plan d'expérience le plus approprié afin d'obtenir les expériences les plus intéressantes pour notre type de problème. Les plans utilisés dans le cadre d'une étude de RSM sont des plans quadratiques tels que les plans centraux composites (Box-Wilson) ou les plans de Box-Behnken. Le plan central composite (CCD) est le plus souvent employé. Mais d'autres types de plan quadratique s'offrent à l'expérimentateur.

Modélisation

Lorsque l'on utilise la RSM, on cherche à relier une réponse continue y à k facteurs continus et contrôlés X1, X2,...Xk à l'aide d'un modèle de régression linéaire qui peut s'écrire :

$$\boxed{y = f_{\beta}(X_1, X_2, ..., X_p) + \varepsilon} \qquad (2\text{-}2)$$

Bien que la forme précise de la fonction de réponse f_β soit inconnue, l'expérience montre que l'on peut en avoir une bonne approximation par un polynôme.

Dans le cas de deux facteurs, le modèle de régression linéaire est de la forme :

$$y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_{12} X_1 X_2 + \varepsilon \tag{2-3}$$

Il s'agit d'un des modèles les plus simples qui correspond à un modèle du premier degré avec interaction. Lorsque ce modèle est insuffisant pour décrire avec efficacité la réalité expérimentale, il est courant d'utiliser un modèle du second degré incluant les effets quadratiques des facteurs qui, dans la plupart des cas, est suffisant pour modéliser correctement les phénomènes étudiés.

Un modèle de surface de réponses du second degré avec d variables peut s'écrire de la manière suivante :

$$y = \beta_0 + \sum_{i=1}^{d} \beta_i x_i + \sum_{i=1}^{d} \beta_{ii} x_i^2 + \sum_{j=2}^{d} \sum_{i=1}^{j-1} \beta_{ij} x_i x_j + \varepsilon \tag{2-4}$$

Où :

x_i : variables de prédiction codées appelées facteurs

β : coefficient de régression

ϵ : erreur totale qui est la différence entre les valeurs observées et les valeurs estimées de la réponse

Dans la phase d'expérimentation de RSM, il convient de transformer les variables de conception réelles en variables codées, qui sont définies comme étant sans dimension avec une moyenne de zéro et le même écart-type. Le but de l'opération est de traiter tous les facteurs de la même manière et de façon plus aisée car, simplifiée. La règle de passage des variables courantes aux variables codées est :

$$x_i = \frac{2R_i - [R_{max} + R_{min}]}{[R_{max} - R_{min}]} \tag{2-5}$$

Où

- x_i : variables de prédiction centrée réduite (codée)
- R_i : Variable courante (naturelle, réelle)

Les coefficients de régression, paramètres inconnus du modèle (équation 2-4), sont estimés par la technique des moindres carrées et la qualité d'ajustement du modèle est évaluée à l'aide d'outils classiques de régression linéaire multiple.

Le modèle d'approximation peut s'écrire sous forme matricielle :

$$\hat{y} = Xb \tag{2-6}$$

Où :

- $\hat{y} = \left[\hat{y}_1, \hat{y}_2, ..., \hat{y}_n \right]^T$: estimation du vecteur réponse des n réponses observées
- X : la matrice de terme de modèle évalué aux points de conception
- b est l'estimateur des moindres carrés du vecteur de régression β, dont la valeur est estimée en minimisant la somme des carrés de l'erreur SS_E définie par :

$$SS_E = \sum_{i=1}^{n} \left(y_i - \hat{y}_i \right)^2 \tag{2-7}$$

Une fois les modèles mathématiques obtenus, il est nécessaire de vérifier que ces modèles donnent une approximation adéquate du système réel à l'étude.

Validation des modèles

Une fois la modélisation effectuée, il faut valider les modèles obtenus. Pour cela, des mesures statistiques importantes sont à considérer :

- le coefficient de détermination statistique : $R^{2;}$
- le coefficient de détermination ajusté : R^2_{adj} ;
- l'erreur rms : RMSE.

Leurs définitions impliquent la partition de l'erreur totale :

$$\boxed{SS_T = \sum_{i=1}^{n} (y - \bar{y})^2} \tag{2-8}$$

en une somme de l'erreur accidentelle et de l'erreur systématique,

$$\boxed{SS_T = SS_R + SS_E} \tag{2-9}$$

Où

- $\bar{y} = \sum_{i=1}^{n} \frac{y_i}{n}$ est la moyenne des réponses.

Quant au coefficient de détermination R^2 il permet d'estimer la variation de la réponse autour de la moyenne qui peut être représentée par le modèle. Il est compris entre 0 et 1 et un R^2 de 1 exprime une correspondance parfaite entre l'évaluation et la valeur réelle $(\hat{y}_i = yi)$. Par contre, un R^2 de 0 signifie que l'évaluation est incapable de prédire la valeur réelle.

$$\boxed{R^2 = 1 - \frac{SS_E}{SS_T}} \tag{2-10}$$

Le coefficient de détermination ajusté R^2_{adj} est une mesure améliorée de R^2. Contrairement à R^2, R^2_{adj} n'augmente pas lorsque des paramètres redondants sont ajoutés à l'approximation de surface de réponse.

$$R^2_{adj} = 1 - \left[\frac{n-1}{n-p}\right]\left(1 - R^2\right) \tag{2-11}$$

Quant à la RMSE, il s'agit d'une estimation de la déviation standard de l'erreur aléatoire et peut s'écrire :

$$RMSE = \sqrt{\frac{SS_E}{(n-p)}} \tag{2-12}$$

Représentation graphique

Après l'étape d'ajustement du modèle, nous représentons graphiquement celui-ci à l'aide de graphique de type « surfaces de réponse » permettant de visualiser le comportement de la réponse en fonction des facteurs en jeu. L'utilisation des surfaces de réponse et de courbes d'isoréponses procure une aide appréciable dans l'exploitation du modèle en vue d'une optimisation du système expérimental.

Une fois toutes ces étapes terminées, nous avons toutes les équations qui constituent notre problème d'optimisation multiobjectif. La forme générale d'un tel problème est :

$$\begin{aligned} \text{Trouver} \quad & x = [x_1, x_2, x_3, \ldots, x_n]^T \\ \text{qui minimise} \quad & f(x) = \{f_1(x), f_2(x), f_3(x), \ldots, f_n(x)\} \\ \text{sujet à} \quad & g_j(x) \leq 0 \qquad \text{pour } j = 1, m \\ & x_i^L \leq x_i \leq x_i^U \qquad \text{pour } i = 1, n \end{aligned} \tag{2-13}$$

2.3 La RSM pour l'optimisation en ingénierie

Une fois la modélisation terminée, le problème à l'étude doit être optimisé. Plusieurs méthodes d'optimisation peuvent permettre d'optimiser simultanément les différentes réponses découlant de l'emploi de la méthode des surfaces de réponses.

2.3.1 Méthode graphique

La première méthode de résolution est l'approche graphique. Pour cela, il suffit de tracer des courbes d'isoréponses pour chaque critère en fonction des variables, puis d'hachurer les zones intéressantes suivant les objectifs retenus. Enfin, par superposition des graphes, on fait apparaître les zones de compromis. Malheureusement, si cette méthode semble très attractive pour des cas simples avec 2 ou 3 variables, elle devient très lourde pour des problèmes plus complexes. Parce que, l'utilisateur doit jongler avec un grand nombre de graphiques pour découvrir le meilleur compromis.

2.3.2 Méthodes numériques

Le quadrillage

Cette méthode numérique présente le double avantage d'une conception simple et d'une recherche systématique. Pour réaliser cette recherche, il suffit de quadriller la zone expérimentale, et de calculer en chacun des nœuds les valeurs obtenues pour les différentes réponses. On obtient ainsi les différents points présentant des valeurs obtenues pour tous les critères. L'inconvénient majeur vient essentiellement de la difficulté d'obtenir à la fois une grande précision et un temps de calcul raisonnable :

- Un maillage trop fin augmente le temps calcul,
- Un maillage trop large risque de ne pas signaler des zones solutions qui seraient intéressantes.

Méthode GDF

Il s'agit d'une méthode d'optimisation dite 'line search', fondée sur un algorithme dérivé de celui de la plus grande pente et qui a été présentée par Geoffrion, Dyer et Feinberg [Geoffrion, A.M.D., Feinberg J.S., 1972]. Bien que rapide, cette méthode sous-entend trop de contraintes d'utilisation sur les fonctions étudiées (différentiables, concaves). Cette méthode demande une certaine connaissance mathématique des fonctions utilisées. Toutefois, sa généralisation ne semble pas assez pratique. Nous ne la détaillerons pas plus.

Méthode de Khuri et Conlon

Pour cette méthode, Khuri et Conlon [Khuri, A.I., Conlon, M. (1981)] définissent un optimum idéal comme étant le point tel que toutes les réponses y atteignent leur optimum individuel. La déviation par rapport à l'optimum idéal est exprimée au moyen d'une fonction "distance". En minimisant cette 'distance', on peut alors atteindre un optimum dit "de compromis". Cependant, l'utilisateur n'a aucun moyen de hiérarchiser les différentes réponses. La distance à un optimum est rarement proportionnelle à une mesure de qualité. De plus, si deux réponses sont corrélées, on doit en supprimer une. C'est ce manque de souplesse qui handicape cette méthode.

Fonction de désirabilité

L'approche de la fonction de désirabilité est l'une des méthodes le plus largement répandues dans l'industrie pour traiter l'optimisation des problèmes multiréponses. L'idée de base est que la qualité d'un produit qui a des caractéristiques multiples est complètement inacceptable si une des caractéristiques se trouve à l'extérieur des limites désirées.

C'est un procédé simple pour combiner les valeurs de plusieurs propriétés en une mesure quantitative, représentative de la qualité globale du compromis. Harrington [Harrington E.C. Jr ,1965], a été le premier à présenter un schéma d'optimisation en terme de fonction de désirabilité développé comme suit, et présenté à la figure 2.5 :

- Stratégie expérimentale ;
- Estimation d'un modèle mathématique pour chaque réponse ;
- Transformation des réponses pour exprimer les objectifs ;
- Désirabilité globale ;
- Désirabilité pondérée.

Algorithme d'optimisation employant la fonction de désirabilité

1- Stratégie expérimentale

2- Modélisation mathématique de chaque réponse

$$Y_j = f_j(x_1, x_2, ..., x_k)$$

3- Transformation de chaque modèle en fonction des objectifs de désirabilité

Pour N réponses Yj à optimiser

Définir à chaque point du plan d'expérimentation pour chaque réponse

Les fonctions élémentaires de désirabilité (di). di $\in$[0, 1]

$$d_i = \frac{Y_i - Y_{\text{intolérable}}}{Y_{\text{idéale}} - Y_{\text{intolérables}}}$$

Les fonctions de désirabilité finale D fonction

$$D = \left\{ \prod_{i=1}^{m} d_i \right\}^{\frac{1}{m}}$$

4- Définition d'une fonction de désirabilité caractérisant le compromis global :

$$D = g(d_1, d_2, ..., d_m)$$

Figure 2.4 ***Algorithme d'optimisation multiobjectif employant la fonction de désirabilité.***

Dans cet algorithme, chaque transformation d_i donne l'équivalent d'un pourcentage de satisfaction en fonction de la réponse calculée par le modèle, par rapport aux objectifs fixés sur cette réponse. Sur le même principe, la fonction de désirabilité évalue un pourcentage de satisfaction du compromis. Si une valeur pour un critère n'est pas

acceptable, la fonction de désirabilité globale est nulle, le compromis est rejeté. Grâce à ces transformations, nous sommes capables de calculer en tout point du domaine la fonction de désirabilité globale. Il s'agit d'une méthode de recherche performante sur des fonctions quelconques pour déterminer la solution optimale.

L'utilisation de la fonction de désirabilité, représentative du comportement du phénomène étudié face aux objectifs semble être suffisamment générale pour répondre à un grand nombre de problèmes déjà rencontrés, mais certains auteurs tels que Derringer et Suich, émettent des critiques à l'encontre de cette méthode :
« *La méthode de la fonction de désirabilité pour les problèmes d'optimisation est simple et facile à mettre en œuvre. Elle oblige l'utilisateur à effectuer des jugements subjectifs sur l'importance de chaque réponse. Cependant, à cause de la nature subjective de cette approche, l'inexpérience de l'utilisateur dans l'estimation de la fonction de désirabilité peut conduire à des résultats erronés. De plus, cette technique ne prend pas en considération les variances et les corrélations des réponses. Le fait d'ignorer ces corrélations peut altérer la structure de la fonction de désirabilité et ainsi conduire à de médiocres solutions.*» [Derringer G. et Suich R. 1980].

Différentes solutions ont été proposées à ces problèmes par les mêmes auteurs pour améliorer la qualité des solutions. Ainsi, connaissant les modèles des réponses ainsi que les variances associées, il est possible d'étudier l'influence de l'incertitude des modèles sur la solution proposée. De plus, les corrélations des réponses, ou plus précisément les corrélations des réponses après transformation, sont des conséquences du choix des objectifs. Lors de la phase de calcul, l'utilisateur peut les détecter facilement. Pour finir, la subjectivité de l'expérimentateur doit toujours être prise en compte, elle permet à l'expérimentateur de rester maître de son problème et des décisions à prendre.

Il est donc évident qu'une majorité des défauts attribués à la fonction de désirabilité possède des solutions envisageables permettant de justifier son utilisation dans de

nombreuses applications industrielles où elle a donné des informations précieuses et s'est révélée très souple d'emploi. Mais le fait que cette méthode présente parfois des résultats de moindre qualité, en terme de précision, comparativement à d'autres approches d'optimisation est une limite qui ne peut être ignorée.

CHAPITRE 3

L'APPROCHE DE L'ALGORITHME DES COLONIES DE FOURMIS

3.1 Introduction

Les métaheuristiques forment une famille d'algorithmes d'optimisation visant à résoudre des problèmes d'optimisation difficiles, pour lesquels des méthodes classiques plus efficaces ne sont pas connues. Elles sont généralement utilisées comme des méthodes génériques pouvant optimiser une large gamme de problèmes, sans nécessiter de changements profonds dans l'algorithme employé. Elles offrent un compromis intéressant entre la qualité des résultats et le temps de résolution. De nombreuses métaheuristiques ont été développées depuis quelques années. Parmi ces dernières, nous comptons les algorithmes des colonies de fourmis récemment proposées pour les problèmes d'optimisation combinatoire difficile.

Bien que les problèmes d'optimisation difficile en variables continues soient très courants en ingénierie, ils sont peu étudiés dans le développement de métaheuristiques d'optimisation. La plupart des algorithmes d'optimisation sont, en effet, proposés dans le domaine combinatoire. Dès lors, l'adaptation de ces métaheuristiques combinatoires aux problèmes continus est profitable à un grand nombre de problèmes réels.

Les algorithmes des colonies de fourmis (ACO) s'inspirent des comportements collectifs de dépôt de phéromone et de suivi de traînée observés dans les colonies de fourmis. Les fourmis d'une colonie communiquent indirectement via des modifications dynamiques de leur environnement, les traînées de phéromone, et construisent ainsi une solution à un problème, en s'appuyant sur leur expérience collective.

3.2 Définition et origines de l'ACO

Dès son introduction au début des années 90, l'algorithme de colonies de fourmis s'est montré efficace pour la résolution de nombreux problèmes.

L'optimisation par les colonies de fourmis (ACO) [Dorigo, 1992], qui a été inspirée par des études du comportement des fourmis [Deneubourg et al., 1983 ; Deneubourg et Goss, 1989 ; Goss et al., 1990], est une approche multiagent pour la résolution de problème d'optimisation combinatoire comme le problème du voyageur de commerce (TSP).

Les fourmis communiquent entre elles par le biais d'une substance appelée « phéromone », que chacune dépose sur le sol en quantité variable durant son déplacement et marque ainsi le chemin qu'elle a emprunté. Il a été observé que plus les fourmis utilisent un chemin, plus il y a de phéromone déposée sur ce chemin et plus ce dernier devient attrayant à d'autres fourmis cherchant de la nourriture. Comme le montre la figure 3.1, si un obstacle est soudainement placé sur un chemin établi menant à une source de nourriture, les fourmis se dirigeront tout d'abord à gauche ou à droite de l'obstacle de façon aléatoire dans le but d'atteindre la nourriture. Mais les fourmis choisissant le côté du chemin le plus court, atteindront plus rapidement leur objectif et feront donc le voyage de retour plus souvent. La quantité de phéromone sur le chemin le plus court sera alors plus importante ce qui ferra de ce chemin le favori. Mais le phéromone déposé sur le chemin a également un taux d'évaporation qu'il faut considérer. [Dorigo M., Maniezzo V., Colorni A., 1996]

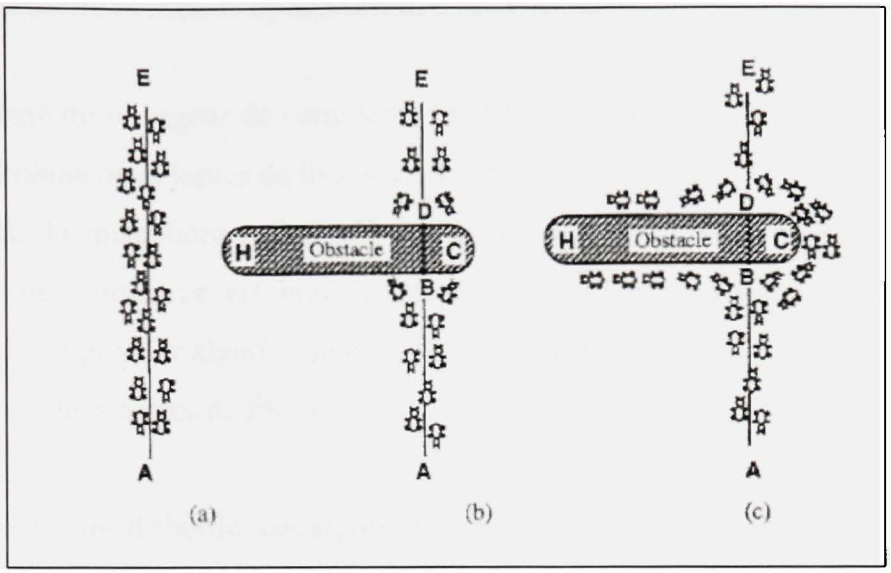


Figure 3.1 ***Expérience de sélection du chemin le plus court par une colonie de fourmis. [Dorigo M., Maniezzo V., Colorni A., 1996].***

La gestion de la trace de phéromone représente un élément central dans les différents algorithmes d'ACO développés et fait l'objet de la majorité des modifications proposées par différents auteurs tels que Dorigo et Stützle [Dorigo et Stützle, 2004].

Dans notre domaine de recherche d'optimisation de problèmes continus, l'ACO a été très peu utilisé et s'est essentiellement focalisé sur des problèmes à objectif unique [Viana Felipe A.C et al, 2006]. Pour des problèmes multiobjectifs, l'ACO a été peu utilisé [Gagné C. et al, 2004; Nagesh K.D. et Janga R.M., 2006; Y.C. Liang et A.E. Smith, 2004] et principalement dans des problèmes d'optimisation combinatoire. D'où la pertinence de notre travail qui vise à adapter des colonies de fourmis continues à des problèmes d'expérimentation à objectifs multiples.

3.3 Méthode usuelle d'optimisation avec l'ACO

Le problème du voyageur de commerce (TSP) a fait l'objet de la première implantation d'un algorithme de colonies de fourmis : le «Ant System» (AS) [Colorni et al., 1992]. Le passage de la métaphore à l'algorithme est ici relativement facile et le problème du voyageur de commerce est bien connu et étudié. Il est intéressant d'approfondir le principe de ce premier algorithme pour bien comprendre le mode de fonctionnement des algorithmes de colonies de fourmis.

Il y a deux façons d'aborder ces algorithmes :

- La première est celle qui a, historiquement, mené au « Ant System » original ; et que nous allons décrire dans la cette section;

- La seconde est une description plus formelle des mécanismes communs aux algorithmes de colonies de fourmis.

3.3.1 Description de l'algorithme de base « Ant system » (AS)

Le problème du voyageur de commerce, à l'origine de l'AS, consiste à trouver le trajet le plus court reliant « n » villes données, chaque ville ne devant être visitée qu'une seule fois. Le problème est plus généralement défini comme un graphe complètement connecté (N; A), où les villes sont les noeuds « N » et les trajets entre ces villes, les arêtes A.

Dans cet algorithme de colonies de fourmis de base, à chaque itération t ($1 \leq t \leq t_{max}$), chaque fourmi k (k = 1, …, m) parcourt le graphe et construit un trajet complet de n = |N| étapes (on note |N| le cardinal de l'ensemble N). Pour chaque fourmi, le trajet entre une ville i et une ville j dépend de :

1. la liste des villes déjà visitées, liste tabou, qui définit les mouvements possibles à chaque étape, quand la fourmi k est sur la ville i : J_{ki} ;

2. l'inverse de la distance entre les villes : $\eta_{ij} = 1/d_{ij}$, appelée visibilité. Cette information "statique" est utilisée pour diriger les fourmis vers des villes proches, et éviter les villes trop lointaines ;

3. la quantité de phéromone déposée sur l'arête reliant les deux villes, appelée l'intensité de la traînée. Ce paramètre définit l'attractivité d'une partie du trajet global et change à chaque passage d'une fourmi. C'est, en quelque sorte, une mémoire globale du système, qui évolue par apprentissage.

Algorithme Ant system (AS)

Pour t = 1, …, t_{max}
  Pour chaque fourmi k = 1, …, m
    Choisir une ville au hasard
    Pour chaque ville non visitée i
      Choisir une ville j, dans la liste J_{ki} des villes restantes, selon *l'équation 3-1*
    Fin Pour
    Déposer une piste $\Delta\tau_{ij}^{k}(t)$ sur le trajet $T^k(t)$ conformément à *l'équation 3-2*
  Fin Pour
  Évaporer les pistes selon *l'équation 3-3*
Fin Pour

Figure 3.2 ***Algorithme de colonies de fourmis de base : le « Ant System ».***

La règle de déplacement appelée « règle aléatoire de transition proportionnelle » par les auteurs Bonabeau et al. [Bonabeau et al., 1999] est la suivante :

$$P_{ij}^{k}(t) = \begin{cases} \dfrac{(\tau_{ij}(t))^{\alpha} \cdot (\eta_{ij})^{\beta}}{\sum_{l \in J_i^k} (\tau_{il}(t))^{\alpha} \cdot (\eta_{ij})^{\beta}} & si \in J_i^k \\ 0 & si \notin J_i^k \end{cases} \qquad (3\text{-}1)$$

Où

- α et β sont deux paramètres qui contrôlent l'importance relative de l'intensité de la traînée, $\tau_{ij}(t)$, et de la visibilité, η_{ij} .

Si $\alpha = 0$, seule la visibilité de la ville est prise en compte ; la ville la plus proche est donc choisie à chaque pas. Par contre, si $\beta = 0$, seule la quantité de phéromone détermine le choix de la ville à explorer.

Pour éviter une sélection trop rapide d'un trajet, un compromis entre ces deux paramètres, jouant sur les comportements de diversification et d'intensification, est nécessaire.

Après un tour complet, chaque fourmi laisse une certaine quantité de phéromone $\Delta\tau_{ij}^{k}(t)$ sur l'ensemble de son parcours, quantité qui dépend de la qualité de la solution trouvée :

$$\Delta\tau_{ij}^{k}(t) = \begin{cases} \dfrac{Q}{L^{k}(t)} & si\ (i,j) \in T^{k}(t) \\ 0 & si\ (i,j) \notin T^{k}(t) \end{cases} \qquad (3\text{-}2)$$

Où

- $T^{k}(t)$ est le trajet effectué par la fourmi k à l'itération t,
- $L^{k}(t)$ la longueur de la tournée,
- Q paramètre fixé.

L'algorithme ne serait pas complet sans le processus d'évaporation des traînées de phéromone. En effet, pour éviter d'être piégé dans des solutions « sous-optimales », minimum local, il est nécessaire de permettre au système « d'oublier » les mauvaises solutions. Pour cela, il faut contrebalancer l'additivité des traînées par une diminution constante de la quantité de phéromone des arêtes à chaque itération. La règle de mise à jour des traînées de phéromone est donc :

$$\tau_{ij}(t+1) = (1-\rho).\tau_{ij}(t) + \sum_{k=1}^{m} \Delta\tau_{ij}^{k}(t) \qquad (3\text{-}3)$$

Le processus itératif décrit ci-dessus permet de trouver rapidement une solution de qualité pour des problèmes d'optimisation combinatoire.

3.3.2 Les variantes de l'ACO

Ant System (AS) et élitisme

Une première variante de « Ant system» a été proposée par Dorigo [Dorigo et al., 1996]: elle est caractérisée par l'introduction de fourmis « élitistes ». Dans cette version, la fourmi qui a effectué le trajet le plus court (la meilleure) dépose une quantité plus importante de phéromone, dans l'optique d'accroître la probabilité des autres fourmis d'explorer la solution la plus prometteuse.

Ant-Q

Gambardella et Dorigo [Gambardella and Dorigo, 1995] ont présenté une variante de AS, dans laquelle la règle de mise à jour locale est inspirée d'un algorithme d'apprentissage par renforcement le « Qlearning ».Cependant, aucune amélioration par rapport à l'algorithme AS n'a pu être démontrée. Cet algorithme n'est d'ailleurs, de l'aveu même des auteurs, qu'une version préalable du « Ant Colony System ».

Ant Colony System

L'algorithme « Ant Colony System » (ACS) a été introduit par Dorigo et Gambardella [Dorigo and Gambardella, 1997b, Dorigo and Gambardella, 1997a] pour améliorer les performances du premier algorithme de fourmis sur des problèmes de grandes tailles.
ACS est obtenu en apportant les modifications suivantes à AS :

1. ACS introduit une règle de transition dépendant d'un paramètre q_0 ($0 \leq q_0 \leq 1$), qui définit une balance diversification/intensification. Une fourmi k sur une ville i choisira une ville j par la règle :

$$j = \begin{cases} \arg\max_{u \in J_i^k} \left[(\tau_{iu}(t)).(\eta_i J)^{\beta} \right] & si\ q \leq q_o \\ \mathrm{J} & si\ q > q_o \end{cases} \qquad (3\text{-}4)$$

Où

- q est une variable aléatoire comprise entre 0 et 1,
- $\mathrm{J} \in J_i^k$ une ville sélectionnée aléatoirement selon la probabilité définie par la règle de déplacement qui est dans le cas présent :

$$P_{iJ}^k(t) = \frac{(\tau_{iJ}(t)).(\eta_{iJ})^{\beta}}{\sum_{l \in J_i^k} (\tau_{il}(t)).(\eta_{il})^{\beta}} \qquad (3\text{-}5)$$

En fonction du paramètre q_0, il y a donc deux comportements possibles :

- si $q > q_0$ le choix se fait de la même manière que pour l'algorithme AS, et le système tend à effectuer une diversification ;

- si $q \leq q_0$, le système tend au contraire vers une intensification. En effet, pour $q \leq q_0$, l'algorithme exploite davantage l'information récoltée par le système, il ne peut pas choisir un trajet non exploré.

2. La gestion des pistes de phéromone est séparée en deux niveaux :
   - une mise à jour locale,
   - une mise à jour globale.

Chaque fourmi dépose une traînée de phéromone lors de la mise à jour locale :

$$\tau_{ij}(t+1) = (1-\rho).\tau_{ij}(t) + \rho.\tau_o \qquad (3\text{-}6)$$

Où

- τ_o est la valeur initiale de la traînée de phéromone.

À chaque passage, les arêtes visitées voient leur quantité de phéromone diminuer, ce qui favorise la diversification par la prise en compte des trajets non explorés. À chaque itération, la mise à jour globale s'effectue comme ceci :

$$\tau_{ij}(t+1) = (1-\rho).\tau_{ij}(t) + \rho.\Delta\tau_{ij}(t) \qquad (3\text{-}7)$$

Où les arêtes (i;j) appartiennent au meilleur tour T^+ de longueur L^+ et où $\Delta\tau_{ij}(t) = \frac{1}{L^+}$.

Ici, seule la meilleure piste est mise à jour, ce qui participe à une intensification par sélection de la meilleure solution.

3. Le système utilise une liste de candidats. Cette liste mémorise pour chaque ville les v plus proches voisines, classées par distances croissantes. Une fourmi ne prendra en compte une arête vers une ville en dehors de la liste que si celle-ci a déjà été explorée. Concrètement, si toutes les arêtes ont déjà été visitées dans la liste de candidats, c'est la plus proche des villes non visitées qui sera choisie.

ACS et 3-opt

L'algorithme « ACS et 3-opt » est une variante de ACS, développé par Dorigo et Gambardella [Dorigo and Gambardella, 1997b], qui est une hybridation entre le ACS et une recherche locale de type 3-opt. Ici, une recherche locale est lancée pour améliorer les solutions obtenues avec les fourmis en les ramenant ainsi à l'optimum local le plus proche conduisant à de meilleurs résultats.

Max-Min Ant System

La variante « Max-Min Ant System », notée MMAS, présentée par Stützle et Hoos [Stützle and Hoos, 1997, Stützle and Hoos, 2000] est fondée sur l'algorithme AS et présente quelques différences notables énoncées ici :

1. Seule la meilleure fourmi met à jour une piste de phéromone.
2. La quantité de phéromone des pistes est bornée par τ_{min} et τ_{max}.
3. Les pistes sont initialisées avec la quantité de phéromone maximale τ_{max}.
4. La mise à jour des pistes se fait de façon proportionnelle, les pistes les plus fortes étant moins renforcées que les plus faibles.
5. Une réinitialisation des pistes peut être effectuée.

De meilleurs résultats sont obtenus en mettant à jour la meilleure solution avec une fréquence de plus en plus forte au cours de l'exécution de l'algorithme.

3.3.3 Autres problèmes combinatoires

Les algorithmes de colonies de fourmis développés pour la résolution du problème du voyageur de commerce ont évolué et ont depuis quelques années été appliqués à de nombreux autres problèmes combinatoires dans des domaines variés. La présentation de ces travaux s'avérerait longue même en ne considérant que le domaine de l'optimisation.

Mais si nous nous basons sur l'optimisation comme champ principal d'application, les problèmes NP-difficile, certains algorithmes ont fourni de très bons résultats.

Nous pouvons notamment retenir des performances particulièrement intéressantes dans le cas de :

- l'affectation quadratique [Stützle and Hoos, 2000];
- les problèmes de planification [Merkle et al., 2000];
- l'ordonnancement séquentiel [Gambardella and Dorigo, 2000; Gagné C et al., 2004];
- le routage de véhicule [Gambardella et al., 1999, Gagné et al., 2006];
- le routage sur réseau [Di Caro and Dorigo, 1998b];
- la fiabilité [Zhao J.H, Zhaoheng L., Dao T.M., 2007].

Il existe une littérature importante sur différents types de problèmes : voyageur de commerce, coloriage de graphes, affectation de fréquence, affectation généralisée, sac à dos multidimensionnel, satisfaction de contraintes, etc. Une présentation plus approfondie de ces applications n'est pas nécessaire dans le cadre de cette étude.

3.4 Extension de l'ACS dans le domaine continu

Devant le succès rencontré par les algorithmes des colonies de fourmis, des recherches sur son utilisation dans un cadre autre que l'optimisation combinatoire ont été effectuées. Parmi lesquelles, l'utilisation de ces algorithmes dans des problèmes continus ou dynamiques, ou encore la mise en relation de ce type d'algorithmes dans un cadre d'intelligence en essaim et hybridée avec d'autres métaheuristiques.

Dans le cas qui nous intéresse il s'agit de problèmes d'ingénierie dans le domaine continu. Nous présenterons donc uniquement l'extension de l'ACS dans le domaine continu.

3.4.1 Problèmes d'adaptation

L'extension des algorithmes des colonies de fourmis au domaine continu pose quelques problèmes d'adaptation. En effet, les métaheuristiques sont bien souvent élaborées pour des problèmes combinatoires. Mais il existe une classe de problèmes souvent rencontrée en ingénierie, où la fonction objective est à variables continues et pour laquelle les métaheuristiques peuvent être d'un grand secours (fonction non dérivable, multiples minimums locaux, grand nombre de variables, non-convexité, etc.). Pour cela, plusieurs tentatives pour adapter les métaheuristiques des colonies de fourmis au domaine continu sont apparues.

Outre les problèmes classiques d'adaptation d'une métaheuristique, les algorithmes de colonies de fourmis posent un certain nombre de problèmes spécifiques, dont le principal vient du fait que dans la formalisation d'ACO la solution est construite composant par composant. En effet, un problème continu peut, selon la perspective choisie, présenter une infinité de composants, solutions faisables, le problème de la construction est difficilement solvable dans ce cas. La plupart des algorithmes s'inspirent donc des caractéristiques d'auto-organisation et de mémoire externe des colonies de fourmis, laissant de côté la construction itérative de la solution. Cependant, depuis peu, le caractère probabiliste du formalisme ACO commence à être employé.

En dehors de nos propres travaux, qui seront exposés en détail dans les chapitres 4 et 5, nous avons recensé dans la littérature plusieurs algorithmes de colonies de fourmis pour l'optimisation continue : CACO, un algorithme hybride non baptisé, API, ACO pour l'optimisation continue et CACS.

3.4.2 L'algorithme CACO

Le premier de ces algorithmes nommé CACO ("Continuous Ant Colony Algorithm") a été développé par Belchev et Parmee [Belchev G. et Parmee I.C, 1995]. Cet algorithme combine deux approches de résolution.

Tout d'abord, un algorithme de type évolutionnaire sélectionne et croise des régions d'intérêt, que des fourmis explorent et évaluent. Ensuite, une fourmi sélectionne une région avec une probabilité proportionnelle à la concentration en phéromone de cette région, de la même manière que dans l'« Ant System », une fourmi sélectionnerait une piste allant d'une ville à une autre. La probabilité de sélection associée à chaque région candidate est :

$$P_i(t) = \frac{\tau_i^{\alpha}(t).\eta_i^{\beta}(t)}{\sum_{j=1}^{N} \tau_j^{\alpha}(t).\eta_j^{\beta}(t)} \qquad (3\text{-}8)$$

où

- N est le nombre de régions,
- $\eta_j^{\beta}(t)$ est utilisé pour inclure une heuristique spécifique au problème.

Les fourmis partent alors du centre de la région où elles sont situées et se déplacent selon une direction choisie aléatoirement, tant qu'une amélioration de la fonction objective n'est pas trouvée. Le pas de déplacement utilisé par la fourmi entre chaque évaluation est donné par :

$$\delta_r(t, R) = R.\left(1 - u^{\left(1 - \frac{t}{t}\right)^{\cdot}}\right) \qquad (3\text{-}9)$$

Où

- R est le diamètre de la région explorée,
- u un nombre aléatoire compris entre 0 et 1,
- T le nombre total d'itérations de l'algorithme
- c un paramètre de refroidissement (permettant de réduire le pas à chaque itération).

Si la fourmi trouve une meilleure solution, la région est déplacée de façon à ce que son centre coïncide avec cette solution, et la fourmi augmente la quantité de phéromone de la région proportionnellement à l'amélioration trouvée. Le phénomène d'évaporation des traînées de phéromone se fait normalement en fonction d'un coefficient d'évaporation ρ.

Des modifications ont été apportées par Wodrich et Bilchev [Wodrich and Bilchev, 1997] pour améliorer les performances de l'algorithme CACO original. Ainsi, en plus des fourmis « locales » de CACO, des fourmis « globales » vont explorer l'espace de recherche (Figure 3.3) pour éventuellement remplacer les régions peu intéressantes par de nouvelles régions non explorées. Les régions sont également affectées d'un âge, qui augmente si aucune amélioration n'est obtenue. De plus, le paramètre « t » dans le pas de recherche des fourmis $\delta_r(t,R)$ est défini par l'âge de la région explorée.

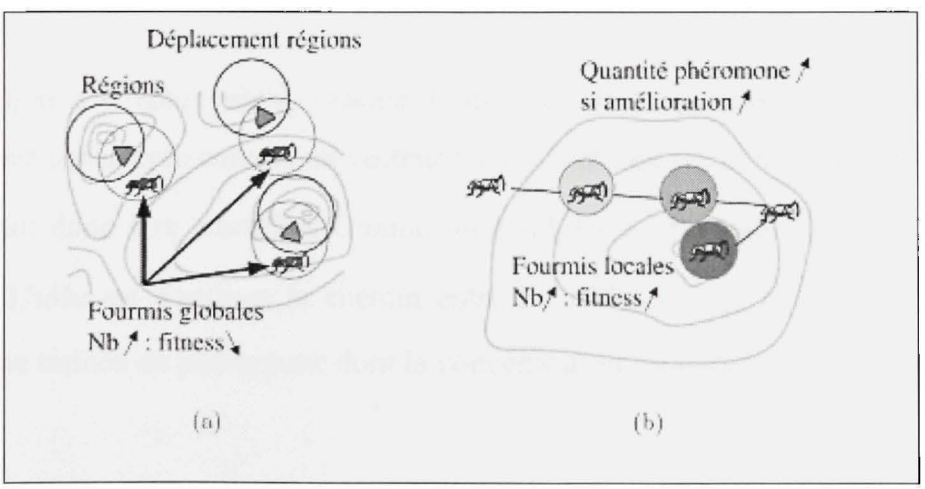


Figure 3.3 ***Fonctionnement de l'algorithme CACO : Les fourmis globales (a) participent au déplacement des régions que les fourmis locales (b) évaluent.***

Une nouvelle modification de l'algorithme a été apportée par Mathur et al. [Mathur et al., 2000] en vue de relier plus fortement CACO avec le paradigme des colonies de fourmis et d'abandonner la liaison avec les algorithmes évolutionnaires. Nous parlons par exemple de diffusion pour définir la création de nouvelles régions d'intérêt.

3.4.3 Une méthode hybride

Ling et al. [Ling et al., 2002] ont proposé une approche semblable à CACO, pour l'optimisation de problèmes continus, utilisant à la fois une approche de colonies de fourmis et d'algorithme évolutionnaire, mais peu de résultats sont actuellement disponibles.

L'idée principale de cette méthode est de considérer les écarts entre deux individus sur chaque dimension comme autant des parties d'un chemin où la traînée de phéromone est déposée. L'évolution des individus est assurée par des opérateurs de mutation et de croisement. Cette méthode tente de reproduire le mécanisme usuel de construction de la solution par les fourmis, soit composant par composant.

La résolution s'effectue ainsi, chaque fourmi x_i de la population contenant « m » individus est considérée comme un vecteur à « n » dimensions. Chaque élément $x_{i,e}$ de ce vecteur peut donc être considéré comme un candidat à l'élément $x^*_{i,e}$ de la solution optimale. L'idée est d'utiliser le chemin entre les éléments $x_{i,e}$ et $x_{j,e}$ noté (i; j) pour déposer une traînée de phéromone dont la concentration est notée $\tau_{ij}(t)$ au pas de temps t.

Les auteurs ont également proposé une version « adaptative », où les probabilités de mutation et de croisement sont variables. Mais cette version est encore à l'étude.

La figure 3.4 résume le fonctionnement de cet algorithme de fourmis hybride pour le cas continu.

Algorithme de colonies de fourmis hybride pour le cas continu.

1. À chaque itération, sélectionner pour chaque fourmi une valeur initiale dans le groupe de valeurs candidates avec la probabilité :

$$P_{ij}^{k}(t) = \frac{\tau_{ij}(t)}{\sum \tau_{ir}(t)}$$

2. Utiliser des opérateurs de mutation et de croisement sur les m valeurs, afin d'obtenir m nouvelles valeurs ;
3. Ajouter ces nouvelles valeurs au groupe de valeurs candidates pour le composant $x_{i,e}$;
4. Les utiliser pour former m solutions de la nouvelle génération ;
5. Calculer la valeur de la « fitness » de ces solutions ;
6. Quand m fourmis ont parcouru toutes les arêtes, mettre à jour les pistes de phéromone des valeurs candidates de chaque composant par :

$$\tau_{ij}(t+1) = (1-\rho).\tau_{ir}(t) + \sum \tau_{ij}^{k}(t)$$

7. Si la $k^{ème}$ fourmi choisit la $j^{ème}$ valeur candidate du groupe de composants, alors :

$$\Delta\tau_{ij}^{k}(t+1) = Wf_{k}$$

sinon $\Delta\tau_{ij}^{k}(t+1) = 0$, en désignant par W une constante et par f_k la valeur de la fonction de fitness pour la solution trouvée par la $k^{ème}$ fourmi ;
8. Effacer les m valeurs ayant les plus basses intensités de phéromone dans chaque groupe de candidats.

Figure 3.4 ***Algorithme de colonies de fourmis hybride en domaine continu.***

3.4.4 L'algorithme API

Dans tous les algorithmes évoqués jusqu'ici, le terme « colonies de fourmis » implique l'utilisation de la stigmergie comme processus d'échange d'information.

Cependant, il existe un algorithme adapté au cas continu développé par Monmarché et al. [Monmarché et al.,2000b] qui s'inspire du comportement de fourmis primitives de l'espèce « *Pachycondyla Apicalis* », et qui ne fait pas usage de la communication indirecte par traînées de phéromone : l'algorithme API.

Cet algorithme peut être résumé ainsi :

- un nid est positionné aléatoirement sur l'espace de recherche.

- ensuite des fourmis sont distribuées aléatoirement dans l'espace. Ces fourmis vont alors explorer localement leur « site de recherche » en évaluant plusieurs points dans un périmètre donné (figure 3.5).

- Chaque fourmi mémorise le meilleur point trouvé. Si, lors de l'exploration de son site de recherche, elle trouve un meilleur point, alors elle reviendra sur ce site, sinon, après un certain nombre d'explorations, elle choisira un autre site.

- Une fois les explorations des sites de recherche terminées, des fourmis tirées au hasard comparent deux à deux leurs meilleurs résultats, puis mémorisent le meilleur des deux sites de chasse.

- Le nid est finalement réinitialisé sur le meilleur point trouvé après un temps donné, la mémoire des sites des fourmis est remise à zéro, et l'algorithme effectue une nouvelle itération.

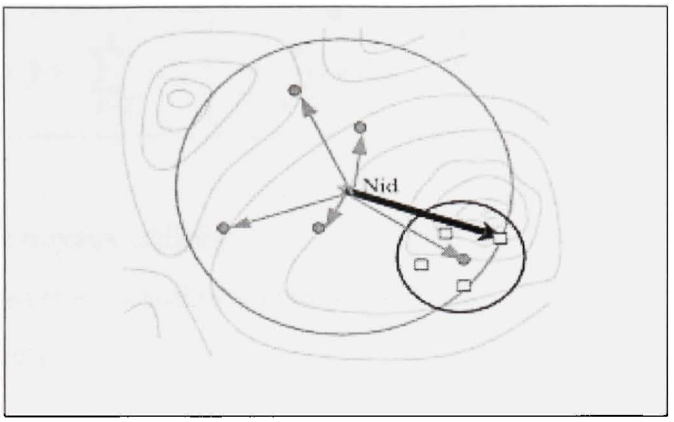


Figure 3.5 ***Fonctionnement de l'algorithme API : Les fourmis (cercles pleins) explorent des sites de recherche (petits carrés) dans un périmètre (grand cercle) autour du nid.***

3.4.5 ACO pour l'optimisation continue

L'algorithme « ACO pour l'optimisation continue » développé par Socha [Socha, 2004] tente de maintenir la construction itérative des solutions inhérentes dans le cas de variables continues, en adoptant un point de vue différent des précédents. En effet, l'algorithme considère que les composants de toutes les solutions sont formés par les différentes variables optimisées. De plus, plutôt que de considérer l'algorithme du point de vue de la fourmi, l'auteur se place au niveau de la colonie, les fourmis n'étant plus que des points à évaluer.

Dans cette méthode, à chaque itération, une population de fourmis est choisie aléatoirement selon une distribution de probabilité. De cet ensemble de points ne sont conservés que les meilleurs points, qui servent alors à construire une « meilleure » distribution de probabilité.

La distribution de probabilité utilisée est un « amalgame pondéré de noyaux normaux », soit un ensemble de distributions normales combinées exprimé de la manière suivante :

$$P(x) = \sum_{j=1}^{k} w_j \cdot \left(\frac{1}{\sigma \sqrt{2\Pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \right) \qquad (3\text{-}10)$$

Où

- k le nombre de noyaux utilisés,
- μ et σ^2 la moyenne et la variance d'un noyau,
- w_i la pondération.

Chaque distribution n'est utilisée que sur une variable, sans dépendance avec les autres. La modification des distributions est nommée « mise à jour de phéromone » et consiste à renforcer ou à diminuer l'influence des noyaux correspondant aux solutions. Le principe de la méthode est présenté à la figure 3.6.

Algorithmes du type colonies de fourmis, utilisant l'estimation de distribution pour l'optimisation en variables continues.

Construire la distribution de probabilité initiale : $\tau_i^0 = P_i^0(x_i),\ i \in \{1,...,n\}$
Tant que le critère d'arrêt n'est pas atteint :
  Pour chaque fourmi, de a = 1 à m :
    Pour chaque variable, de i = 1 à n :
      Choisir aléatoirement une valeur xi selon la distribution $P_i(x_i)$
Ajouter à la solution en construction :

$$S^a = \{s_1^a,...,s_{i-1}^a\} \cup \{x_i\}$$

    Fin
  Fin
  Mémoriser les k meilleures solutions trouvées : $S^* = \{s_1^*,...,s_k^*\}$
  Reconstruire la distribution de probabilité selon les meilleures solutions : $\tau = P(S^*)$

Figure 3.6 ***ACO pour l'optimisation continue.***

3.4.6 L'algorithme CACS

Pourtakdoust et Nabahari [Pourtakdoust and Nobahari, 2004] ont développé, quant à eux, un algorithme nommé « Continuous Ant Colony System » qui est très proche du précédent, bien qu'ils aient été présentés simultanément. En effet, dans CACS comme dans ACO pour l'optimisation continue, le coeur de l'algorithme consiste à faire évoluer une distribution de probabilité.

Le principe de la méthode est le même que celui présenté dans la figure 3.6. Dans CACS, la distribution utilisée est dite « normale », mais la formule de la distribution diffère légèrement de la formule classique (équation 3-11) et la variance utilisée est en fait un nouvel indice de dispersion (équation 3-12).

$$P(x) = e^{-\frac{(x - x_{min})^2}{2\sigma^2}} \tag{3-11}$$

$$\sigma^2 = \frac{\sum_{j=1}^{m} \frac{1}{f_j - f_{min}} (x_j - x_{min})^2}{\sum_{j=1}^{m} \frac{1}{f_j - f_{min}}} \tag{3-12}$$

Où

- x_{min} le mode de distribution;
- σ^2 l'indice de dispersion;
- m le nombre de fourmis;
- f_j la valeur de la fonction associée à la fourmi j;
- f_{min} la meilleure valeur trouvée.

Dans l'algorithme CACS, la seule distribution utilisée est centrée sur le mode de la distribution de l'itération précédente, et non sur la moyenne.

3.5 Conclusion

En définitive, les algorithmes de colonies de fourmis introduits au début des années 90 constituent un outil efficace dans le domaine de l'optimisation difficile. Ces algorithmes ont été développés pour être utilisés sur des problèmes d'optimisation combinatoire et se sont révélés d'une grande efficacité en fournissant des résultats de qualité en un temps plus court qu'une grande partie des méthodes existantes. Cette efficacité a conduit les chercheurs à envisager l'utilisation de cet algorithme pour des problèmes difficiles autres que les problèmes combinatoires. C'est ainsi que le champ de recherche sur les algorithmes de fourmis continus a vu le jour.

À ce jour cinq (5) types d'algorithmes de fourmis continus sont connus. Parmi eux, deux types d'algorithmes sont en fait plus ou moins hybridés avec un algorithme de type évolutionnaire, et un troisième n'utilise pas la métaphore « classique » des colonies de fourmis. Les deux derniers algorithmes utilisent une approche différente des précédents, plus centrée sur l'aspect probabiliste que comportemental des algorithmes à colonies de fourmis. D'une manière générale, la recherche en est encore à ses débuts et les algorithmes produits n'ont pas atteint leur pleine maturité et ne sont donc pas encore, pour la plupart, vraiment compétitifs par rapport à d'autres classes de métaheuristiques plus élaborées sur les problèmes continus. CACO est une exception car il a fait l'objet d'un plus grand nombre de travaux que les autres algorithmes de colonies de fourmis continus et que sa robustesse à pu être démontrée pour divers problèmes [Baskar N., 2004]. C'est pour cette raison que lors de notre processus d'optimisation multiobjectif, nous nous intéresserons à l'algorithme CACO plutôt qu'aux autres algorithmes de colonies de fourmis continus.

CHAPITRE 4

MÉTHODE HYBRIDE D'OPTIMISATION PROPOSÉE

4.1 Introduction

L'objectif de cette recherche est de développer un outil d'optimisation qui permet de trouver la meilleure conception d'un processus tel que l'usinage d'une pièce, le soudage, le moulage, l'assemblage, en tenant compte simultanément de plusieurs objectifs parfois antagonistes. Cette optimisation doit se faire en utilisant la méthode des surfaces de réponses (RSM), très utile pour pallier aux problèmes d'évaluation coûteuse des fonctions objectives auxquelles se heurtent souvent les chercheurs. Particulièrement lorsque ces fonctions objectives et les contraintes associées sont obtenues par l'intermédiaire de simulations par éléments finis impliquant des mailles fines, un grand nombre de degrés de liberté et des comportements divers non linéaires. La RSM permet donc de remplacer des modèles complexes par des approximations facilitant ainsi le calcul des fonctions objectives telles que le coût. De plus, associée à la fonction de désirabilité, comme vu précédemment, elle permet d'effectuer l'optimisation multicritère désirée. Mais notre étude a révélé que l'emploi de la fonction de désirabilité conduit à des résultats de qualité moindre comparativement à ceux obtenus avec l'emploi d'une métaheuristique d'optimisation telle que les réseaux de neurones, les algorithmes génétiques, etc.

Dans la littérature, outre l'utilisation de la fonction de désirabilité pour ce type d'optimisation, des exemples associant les algorithmes génétiques et la RSM sont les plus présents et présentent des résultats appréciables. Une étude comparative plus approfondie des algorithmes génétiques et des algorithmes des colonies de fourmis a permis d'observer que l'algorithme des colonies de fourmis conduit plus rapidement à l'optimum que les AG pour plusieurs problèmes combinatoires [Nagesh Kumar D. et

Janga Raddy M., 2006], [Liang Y.C. et Smith A.E., 2004]. Ces observations ont éveillé l'intérêt pour cet algorithme d'optimisation et ont conduit au développement d'un nouvel algorithme hybride associant la RSM et les colonies de fourmis continues dans le but de fournir de meilleurs résultats que ceux obtenus présentement et cela avec une vitesse de convergence acceptable.

Le nouvel algorithme hybride qui est notre contribution est une hybridation entre une méthode d'approximation et un algorithme évolutionnaire où la méthode d'approximation est utilisée pour accélérer la convergence de l'algorithme évolutionnaire. Dans notre algorithme toutes les évaluations des fourmis sont faites par la fonction approchée obtenue avec la RSM. La modélisation à l'aide de la RSM est donc une phase importante dans le processus d'optimisation. Mais il faut s'assurer que l'erreur d'approximation est faible pour garantir la qualité du résultat final.

4.2 Approche d'optimisation proposée

Notre algorithme de résolution est constitué de deux phases qui sont représentées à la figure 4.3. Les différentes étapes de cet algorithme sont les suivantes :

Étape 1 : Configuration du système à l'étude

Déterminer les différents objectifs quantifiables de notre conception et les contraintes de notre système. Ensuite déterminer toutes les variables de notre système et leurs domaines de valeurs.

Selon le nombre d'objectifs et de variables à l'étude, il est possible de mettre en place le meilleur plan d'expériences permettant de réduire au maximum le nombre d'expériences nécessaire à la modélisation de nos objectifs.

Étape 2 : RSM

Mettre en place un plan d'expérience, effectuer les essais ou la simulation et modéliser les différents objectifs en fonction des paramètres influents.

Le résultat obtenu est un modèle mathématique pour chacun des objectifs à optimiser. Le type de modèle associé aux plans quadratiques tels que les plans composites, sont des modèles additifs. Le coefficient de corrélation de chaque équation permet de valider le modèle mathématique. Plus sa valeur est proche de 1, plus notre modèle mathématique est représentatif des données expérimentales. Un modèle mathématique est considéré représentatif si le coefficient de corrélation correspondant est supérieur ou égal à 0.9.

Une fois la phase de modélisation terminée, une équation mathématique codée pour chaque objectif à optimiser est disponible.

$$\boxed{y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_{12} X_1 X_2 + \varepsilon} \tag{4-1}$$

Chaque objectif f_i est exprimé en fonction des variables de conception codées x_i qui influent sur sa valeur. Le modèle d'optimisation multi objectif obtenu grâce à la RSM est le suivant:

$$\boxed{\begin{array}{c} \text{minimiser} \quad f(x) = \{f_1(x), f_2(x), f_3(x), ..., f_n(x)\} \\ \text{sujet à} \quad x_i^L \le x_i \le x_i^U \quad \text{pour } i = 1, n \end{array}} \tag{4-2}$$

Maintenant que notre problème est modélisé, nous passons à la phase d'optimisation des différents objectifs simultanément soit la recherche d'une solution de compromis

Étape 3 : Recherche des points idéaux

L'optimisation de chaque objectif pris séparément, à l'aide de la RSM, permet de trouver les différentes solutions optimales de chacun des objectifs de l'étude. Le vecteur de solutions correspond alors aux ***points idéaux*** et peut être exprimé comme suit : $F^* = \{F_1^*, F_2^*, ..., F_Z^*\}$ où $F_i^* = \min_{x \in S} f_i(x)$. Il s'agit dans le cas d'objectifs conflictuels, du but à atteindre, qui généralement est une solution non réalisable. Cette recherche peut être effectuée rapidement en utilisant la RSM et la fonction de désirabilité. Dans certains cas, cette recherche n'est pas nécessaire car la valeur optimale de chaque objectif est disponible dans les donnés du problème. Les valeurs du vecteur ***points idéaux*** F* sont utilisées dans la fonction de fitness pour le processus d'optimisation

Étape 4 : Formulation de la fonction de fitness (fonction d'optimisation)

a) Énoncé des préférences de l'usager (pondération des objectifs) :

b) Différentes méthodes sont présentées dans la littérature pour formuler la fonction de fitness d'un problème multiobjectif. Parmi elles, trois types de méthode se distinguent:

- Les méthodes agrégées qui transforment un problème multiobjectifs en un problème à objectif unique.

- Les méthodes basées sur l'approche de Pareto qui sont fondées sur la notion de dominance au sens de Pareto privilégient une recherche satisfaisant au mieux tous les objectifs.

- Les méthodes non agrégées et non Pareto qui possèdent un processus de recherche qui traite séparément les objectifs.

Notre intérêt se porte sur les méthodes agrégées et les méthodes fondées sur Pareto qui, assemblées, nous permettront de mieux représenter la fonction de fitness.

C'est l'économiste Vilfredo Pareto [Pareto V., 1896] qui, à la fin du 19ème siècle, a formulé le concept d'optimum de Pareto qui constitue les origines de la recherche sur l'optimisation multiobjectifs. Ce concept se résume ainsi : «*dans un problème multiobjectifs, il existe un équilibre tel que l'on ne peut pas améliorer un critère sans détériorer au moins un des autres critères*». Cet équilibre a été appelé «*optimum de Pareto*».

Un point x est dit **Pareto-optimal** s'il n'est dominé par aucun autre point appartenant à l'espace de recherche. Ces points sont également appelés solutions *non inférieures* ou *non dominées*. La notion de dominance est défini par :

$$\boxed{\begin{array}{l}\text{Un point } x \in S \text{ domine } x' \in S \text{ si:} \\ \quad \forall i, f_i(x) \le f_i(x') \\ \quad \text{avec au moins un } i \text{ tel que } f_i(x) < f_i(x')\end{array}} \qquad (4\text{-}3)$$

Concernant les méthodes agrégées, les deux techniques d'agrégation les plus fréquentes permette de ramener un problème multiobjectifs à un problème mono-objectif sont la méthodologie de la somme pondérée et la méthode ϵ-contraintes. En modifiant leurs paramètres, ces deux méthodes permettent de retrouver l'ensemble de solutions optimales au sens de Pareto. Elles sont très souvent utilisées car elles permettent l'utilisation de techniques d'optimisation monocritère pour aborder des problèmes multiobjectifs.

La formation de la fonction de fitness pour l'optimisation multiobjectifs s'effectue de la manière suivante :

- À l'aide de la méthode du "*Compromise Programming*" (CP) proposée par Zeleny (1982), il est possible d'agréger les différents objectifs en une seule mesure permettant de refléter la qualité d'une solution réalisable quelconque en calculant sa distance au *point idéal* F*. Ce qui permet d'effectuer la recherche de solutions s'approchant du *point idéal* en représentant les différents objectifs à atteindre au sein d'une fonction unique.

- L'importance relative accordée à chaque objectif par l'utilisateur est exprimée par un ensemble de pondération $0 \leq p_i \leq 1$ et $\sum_{i=1}^{z} p_i = 1$ qui doit être considéré dans le calcul d'une distance pondérée au point idéal. La pondération de chaque objectif est donnée par l'utilisateur lors de la définition du problème.

- De plus, une normalisation est souvent nécessaire pour réduire les effets indésirables provenant des unités de mesure des différents objectifs ainsi que de l'étendue du domaine des variables de façon à ne pas biaiser la distance calculée.

- La distance normalisée et pondérée (dist) au point idéal F* d'une solution réalisable quelconque k dont les résultats sur les différents objectifs sont donnés par $F^k = \{F_1^k, F_2^k, ..., F_Z^k\}$. Cette fonction permet de normaliser les objectifs pour réduire les effets indésirables provenant des différentes unités de mesure ainsi que de l'étendue du domaine des variables de façon à ne pas biaiser la fonction de fitness. [Gagné C. et al, 2004].

$$fitness = \left(\sum_{i=1}^{Z} p_i \left(\frac{F_i^k - F_i^*}{F_i^{nad} - F_i^*} \right)^2 \right)^{\frac{1}{2}} \qquad (4\text{-}4)$$

Où :

- F* : Vecteur de solution correspondant au point idéal de chaque objectif

- F^{nad} : Point Nadir qui représente les valeurs maximales pour chacun des objectifs dans l'ensemble de solutions Pareto optimales.

$$\boxed{F^{nad} = \{F_1^{nad}, F_2^{nad}, ..., F_z^{nad}\}} \quad (4\text{-}5)$$

Où $F_i^{nad} = \max_{x \in S} f_i(x)$.

À la différence du vecteur idéal qui représente les bornes inférieures de chaque objectif dans l'espace de recherche, le vecteur Nadir F^{nad} correspond à leurs bornes supérieures sur la surface de Pareto et non dans tout l'espace de recherche (figure 4.1). Ce vecteur est bien plus difficile à trouver que le vecteur idéal. Le vecteur Nadir peut correspondre à une ou à aucune solution existante, en fonction du problème (notamment, de la convexité du domaine de recherche).

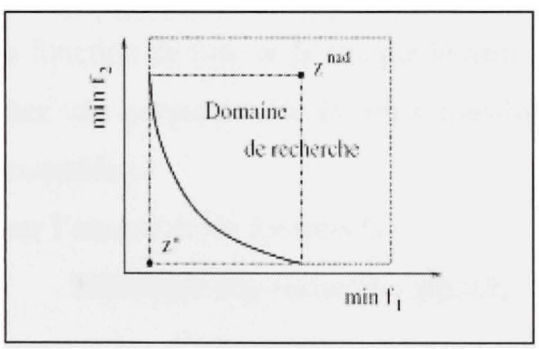


Figure 4.1 Vecteur idéal Z^* versus vecteur Nadir Z^{nad}.

L'utilisation de ce point F^{Nad} est l'un des éléments qui différencie cette méthode de résolution des problèmes multiobjectifs des autres méthodes couramment utilisées. En effet, ce point associé à la méthode agrégée dans la recherche de la solution de compromis, permet d'ajouter les avantages de la méthode de Pareto à la méthode agrégée.

Étape 5 : Détermination de la solution de compromis

Une fois le problème modélisé et la fonction de fitness définie, il faut passer à l'étape d'optimisation simultanée de nos différents objectifs.
Pour cette phase, notre intérêt s'est porté sur l'algorithme des colonies de fournis encore très peu utilisé dans les problèmes de conception mécanique. Un algorithme de fourmis continu et multiobjectif est donc développé dans la suite de ce chapitre. Il fait appel à l'algorithme CACO (Continues Ant Colonies Optimization) développé par Belchev et al. [Belchev G. et Parmee I.C., 1995] pour l'optimisation monocritère.

Les étapes de cette phase d'optimisation multiobjectif se résument ainsi :

Génération aléatoire de R fourmis solutions faisables
Tant que le critère d'arrêt n'est pas atteint
    Pour l'ensemble de fourmis de R
        Évaluer la fonction de fitness de chaque fourmi
        Sélectionner une proportion de fourmis constituée des pires solutions pour former l'ensemble G
            Pour l'ensemble de fourmis G
                Effectuer une recherche globale
                    Croisement
                    Mutation
                    Diffusion de traînée
            Fin
            Pour le nouvel ensemble de fourmis R
                Effectuer une recherche locale
            Fin
    Fin
Fin

Figure 4.2 ***Algorithme de détermination de la solution de compromis***

Les différentes étapes d'optimisation permettant de déterminer la solution de compromis sont présentées à la figure 4.3 et se décrivent de la manière suivante :

Génération aléatoire de R fourmis solutions faisables

Pour appliquer la méthodologie ACO aux problèmes d'optimisation de fonctions continues, le domaine de recherche doit être subdivisé en nombre spécifique de régions distribuées aléatoirement, R. Pour cela, il faut générer des solutions réalisables représentant nos fourmis initiales, chacune de ces fourmis faisant partie d'une région de recherche à explorer. Comme tous les algorithmes évolutionnaires, la population initiale a une grande importance sur la qualité du résultat obtenu et sur la vitesse de convergence de notre algorithme. Pour un meilleur résultat il est important que nos fourmis initiales soient distribuées sur tout l'espace de recherche. Pour ce faire, nous avons opté pour une génération aléatoire des fourmis initiales afin de ne pas influer sur leur emplacement.

Chaque fourmi est codée par un ensemble de variables. Chaque variable correspond à un facteur qui influe sur les objectifs. La population de fourmis initiales est évaluée en calculant la valeur de la fonction de fitness de chaque fourmi. Les valeurs obtenues sont rangées par ordre décroissant pour un objectif de minimisation. Un pourcentage de fourmis ayant les plus grandes valeurs de la fonction de fitness (les pires résultats) est sélectionné pour constituer les fourmis globales G. Ce pourcentage de fourmis global est un paramètre important dans l'algorithme CACO qui varie en fonction du problème étudié.

Recherche globale

À cette étape il s'agit d'appliquer une recherche globale sur les fourmis globales obtenues précédemment (constituant les pires solutions disponibles).

La recherche globale crée de nouvelles solutions de *G* par le remplacement des parties (solutions) plus faibles du domaine existant. Ce procédé se compose essentiellement de deux opérateurs génétiques. Dans la terminologie de CACO, ces opérateurs s'appellent la recherche aléatoire et la diffusion de traînée.

- **La recherche aléatoire** :

Dans ce procédé, les fourmis se déplacent dans de nouvelles directions à la recherche de plus récentes et plus riches sources de nourriture. Dans la simulation de CACO, une recherche aléatoire dans tout le domaine est menée par un processus semblable au croisement et à la mutation d'algorithmes évolutionnaires.

a) Croisement *(Crossover ou random walk)* :

L'opération de croisement est réalisée afin de remplacer les solutions inférieures du domaine par les solutions supérieures, en terme de qualité de la fonction de fitness, avec la probabilité de croisement (CP). Ainsi une proportion de fourmis globales sera remplacée par des fourmis locales qui constituent les solutions supérieures. Il s'agit donc d'une étape de remplacement qui sert à déterminer quelles fourmis locales remplaceront la proportion de fourmis globales sujette au croisement lors de l'itération en cours. Pour chaque itération, les fourmis globales sujettes au croisement ne sont modifiées qu'une fois. Il s'agit de remplacement général. L'avantage de cette théorie est d'éliminer les fourmis les plus éloignées de la source de nourriture et de rajouter des fourmis dans des zones qui sont plus proches de la source de nourriture. Pour ce faire, une fourmi locale est choisie aléatoirement et la fourmi globale placée le plus loin d'une source de nourriture se déplace pour aller au même emplacement que la fourmi locale choisie. Ce processus est effectué sur toutes les fourmis globales.

b) Mutation :

Les fourmis déplacées lors du « croisement » sont encore rapprochées de la source de nourriture par mutation. Lors de cette opération, une variable x_i d'une fourmi est réduite

ou augmentée d'une valeur stochastique, Δ, donnée par l'équation 4-6, avec une probabilité proportionnelle à la probabilité de mutation déterminée au préalable. Dans notre cas, la probabilité de mutation a été fixée à 0.5 pour réduire son influence sur le résultat de la mutation. Ainsi le programme est plus autonome.

$$\Delta = MS\left(1 - r^{(1-T)^b}\right) \tag{4-6}$$

Où :

- r : Nombre aléatoire compris entre 0 et 1;
- MS : Intervalle de chaque paramètre de la région. MS = (x_{max} - x_i);
- T : Rapport du nombre courant d'itération sur le nombre total d'itérations;
- b: paramètre positif qui commande le degré de non linéarité.

Si le nombre aléatoire r est supérieur à la probabilité de mutation, la valeur de la variable muté devient :

$$x_i = x_i - \Delta \tag{4-7}$$

Par contre, si r est inférieur à la probabilité de mutation, la valeur de la variable mutée devient :

$$x_i = x_i + \Delta \tag{4-8}$$

Cette mutation s'effectue sur toutes les variables x_i de chaque fourmi globale choisie. Lorsque la modification rapproche la fourmi de l'objectif, cette dernière est déplacée sinon elle conserve son emplacement.

À cette étape le domaine de recherche globale est progressivement diminué au fur et à mesure que la recherche progresse. Cette réduction permet d'augmenter la probabilité de localiser l'optimum par des procédures de recherche plus concentriques.

c) Diffusion de traînée :

La diffusion de traînée est semblable au croisement arithmétique des AG. Elle est appliquée sur la proportion de fourmis globale n'ayant pas subi de croisement et de mutation. À cette étape deux parents sont choisis au hasard à partir de l'espace de population de fourmis parents correspondant aux solutions supérieures de R. Les éléments du vecteur de la fourmi enfant peuvent être les suivant :

1. L'enfant correspond au premier parent
2. L'enfant correspond au deuxième parent
3. L'enfant est une combinaison des parents (Équation 34)

$$X(enfant) = (\alpha) X_{i(parent1)} + (1-\alpha) X_{i(parent2)} \tag{4-9}$$

Où:

- α est le nombre aléatoire uniforme compris entre 0 et 1.

La probabilité de choisir l'une des trois options dépend de la probabilité de mutation. Ainsi avec une probabilité de mutation de 0.5, l'option 3 peut être choisie avec une probabilité de 50%, alors que la probabilité de choisir l'option 1 ou la 2 est de 25%. Ces probabilités nous permettent de donner plus de chance à l'option qui engendre une fourmi enfant à partir d'un mélange d'informations venant de deux fourmis parents plutôt que l'option qui fournit une fourmi enfant étant une réplique d'une des fourmis parents. Nous conservons quand même la probabilité que la fourmi enfant soit à la même position que la fourmi parent pour permettre à la fourmi enfant de se déplacer vers l'emplacement d'un parent; emplacement qui peut être plus proche du nid que l'emplacement obtenu en combinant les deux parents.

Au sortir de cette recherche globale, la matrice R composée des régions de recherche a été modifiée en vue d'optimiser la recherche locale. Pour ce faire, les fourmis disposées aux régions les plus éloignées de la source de nourriture ont été déplacées dans des zones plus proches.

Une fois que les nouvelles régions de la matrice R sont générées, il faut initialiser la quantité de phéromone de ces nouvelles régions. La quantité de phéromone de chaque région correspond à la valeur de la fonction de fitness de la fourmi qui définit cette région.

Recherche locale

Une fois la recherche globale terminée, les zones où envoyer les fourmis locales sont définies et la recherche locale peut commencer.

Dans la recherche locale, les fourmis locales choisissent la région à explorer parmi les régions de la matrice R en fonction de la quantité courante de phéromone de chacune de ces régions. La probabilité de choisir une région i est donnée par :

$$P_i(t) = \frac{\tau_i(i)}{\sum_k \tau_k(t)} \qquad (4\text{-}10)$$

Où :

- i est l'index de solution (région);
- τi (k) est la quantité de phéromone de la solution i au temps T.

Après avoir choisi sa région destination, la fourmi locale va se placer sur la région choisie. Ensuite cette dernière se déplace sur une courte distance dans une direction choisie aléatoirement. Dans chaque région de R, la direction de recherche reste la même d'une solution locale à une autre tant qu'il y a amélioration de la valeur de la fonction de

fitness. S'il n'y a aucune amélioration, la fourmi se réoriente aléatoirement dans une autre direction. Par contre, si une amélioration de la fonction de fitness est obtenue dans la procédure précédente, le vecteur de position de la région est mis à jour. Précisons que la quantité de phéromone déposée par la fourmi est proportionnelle à l'amélioration de la fonction de fitness.

Un autre paramètre à considérer lors du processus de recherche, est la notion d'âge de la région explorée. Si, lors du processus de recherche, une valeur de la fonction de fitness plus élevée (moindre qualité) est obtenue, l'âge de la région explorée augmente. L'âge de la région est un autre paramètre important dans l'algorithme de CACO. La taille du déplacement (LS) de la fourmi dans la recherche locale dépend de l'âge courant. Le rayon de recherche est maximal pour l'âge zéro et minimal pour l'âge maximal, avec une variation linéaire.

$$\boxed{LS = k_1 - (\hat{a}ge * k_2)} \tag{4-11}$$

La valeur de X_{new}, nouvelle coordonnée d'un des facteurs de la fourmi locale, est :

$$\boxed{X_{new} = X_i \pm LS} \tag{4-12}$$

On augmente ou diminue de LS aléatoirement.

Les valeurs de k_1 et k_2 ($k_1 >> k_2$) dépendent du type de problème. Ces constantes permettent de déterminer la taille du déplacement dans un domaine continu. Plus LS est petit, plus petit est le déplacement de la fourmi. De cette façon, il est possible d'augmenter la précision du résultat. Mais il faut noter qu'avec des déplacements plus petits le nombre d'itérations nécessaires pour atteindre la meilleure solution se trouve augmenté. Par contre un LS plus grand permet, avec de plus grands déplacements,

d'atteindre plus rapidement l'optimum. Mais une fois proche de la meilleure solution un grand déplacement peut nous empêcher de l'atteindre.
Pour chaque déplacement de fourmi obtenu avec la recherche locale, il faut évaluer la fonction de fitness et continuer le processus itératif débutant par la recherche globale jusqu'à ce que la condition d'arrêt du programme, soit le nombre d'itération, soit respectée.

L'évaporation de phéromone

Dans un système de fourmi conventionnel, il y a un processus d'évaporation de phéromone laissé par les fourmis. Ce processus fait perdre de l'intérêt aux fourmis pour les chemins empruntés par de précédentes fourmis.

Ce dispositif est incorporé dans l'algorithme de CACO usuel tel que vu précédemment dans le chapitre 3. Dans le cadre de notre étude, nous avons tout d'abord entré le processus d'évaporation de phéromone dans le programme d'optimisation. Mais une étude plus poussée nous à permis de constater que sans la phase d'évaporation de phéromone nous obtenons les mêmes résultats qu'avec l'évaporation de phéromone. La mise en place du processus d'évaporation de phéromone augmente énormément le temps de résolution.

Suite à ces observations, nous avons conclu qu'il était plus pertinent de ne pas inclure une phase d'évaporation de phéromone dans notre processus d'optimisation car il n'y a aucun gain avec cette opération aussi bien sur la qualité des résultats que sur la vitesse de convergence de notre algorithme.

Étape 6 : Évaluation des meilleures solutions (quasi optimale) par la simulation ou l'expérimentation utilisée au départ dans notre plan d'expérience (Exemple FEM)

Étape 7 : Vérification du critère d'arrêt

Le meilleur résultat obtenu à la sortie de l'optimisation doit être testé et s'il répond au critère d'arrêt, le processus d'optimisation est terminé. Sinon il faut revenir à l'étape de modélisation avec RSM.

Le critère d'arrêt du processus est que, si la différence pondérée entre les valeurs de nos objectifs aux points optimaux trouvés, évalués avec les fonctions approximées, et les valeurs de nos objectifs obtenues par expérimentation avec les facteurs au niveau de la solution optimale est inférieure à une erreur e, la solution obtenue est validée.

Il s'agit donc de vérifier que :

$$\boxed{\frac{RSM - SIMUL}{SIMUL} < e} \tag{4-13}$$

Où :

- RSM : Valeur d'un objectif à l'optimum évalué avec la fonction issue du processus de modélisation par RSM;
- SIMUL : Valeur d'un objectif à l'optimum évalué par simulation ou par expérimentation physique;
- e : pourcentage d'erreur acceptable entre la modélisation et la valeur réelle.

Cette vérification permet de savoir si notre modélisation est suffisamment représentative du comportement de nos objectifs en fonction des facteurs influents afin de donner une solution valable. Si le critère n'est pas respecté, il faut revenir à la phase de modélisation de l'algorithme. En effet dans un problème d'optimisation de la conception d'un système mécanique, le nombre de variables de conception est très souvent supérieur ou égal à trois. De plus, chacune d'entre elles a un large domaine de variation. Par conséquent, dans notre processus de résolution, il est possible que le domaine de recherche de chaque variable de conception soit progressivement rétréci tant que le critère d'arrêt n'est pas respecté. Et donc que les modèles obtenus avec RSM soient progressivement améliorés (affinés).

Les nouveaux modèles mathématiques obtenus avec RSM sont utilisés dans toutes les étapes du processus d'optimisation pour obtenir de meilleures solutions pour notre optimisation.

Le processus de recherche se termine quand $\frac{RSM - SIMUL}{SIMUL} < e$.

e étant une marge d'erreur préalablement définie.

4.3 Conclusion

La figure 4.3 résume toutes les étapes de notre algorithme d'optimisation.

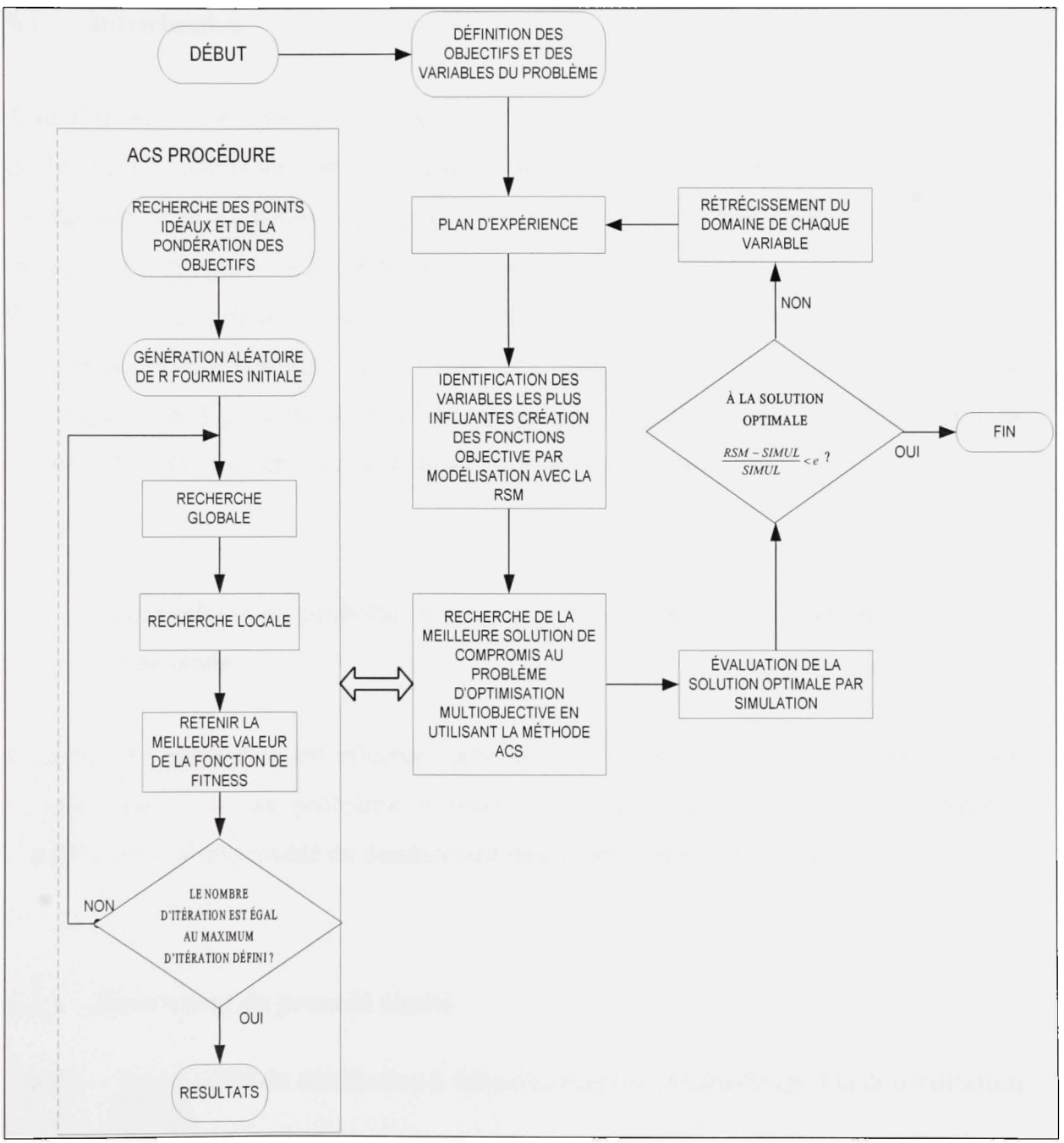


Figure 4.3 ***Organigramme de l'approche hybride d'optimisation***

CHAPITRE 5

APPLICATION NUMÉRIQUE À DES PROBLÈMES VARIÉS

5.1 Introduction

Afin d'illustrer les performances de l'approche de résolution proposée, nous l'avons appliquée à l'optimisation d'un procédé de dessalement instantanée à plusieurs niveaux (multistage flash desalination process) et à un problème de fabrication de moules par usinage cinq (5) axes. Le problème de dessalement traité est tiré de l'article de Abdul-Wahab et Abdo [Abdul-Wahab et Abdo, 2007] et a été résolu en utilisant les plans d'expériences et l'optimisation à l'aide des fonctions de désirabilité. Le problème d'usinage quant à lui, a été étudié à l'École de technologie supérieure (ETS) par Amtout (Amtout Y., 2003) en utilisant la RSM et les fonctions de désirabilité

5.2 Application à un problème industriel à deux objectifs : Dessalement instantanée

La première application est effectuée pour vérifier les performances de notre méthode d'optimisation sur un problème à deux objectifs. Pour ce faire, nous étudions l'optimisation d'un procédé de dessalement instantané à plusieurs niveaux.

5.2.1 Description du procédé étudié

5.2.1.1 Le procédé de distillation à détentes étagées (Multi-Stage Flash distillation MSF).

Les procédés de distillation, consistent à chauffer l'eau de mer pour en évaporer une partie. La vapeur ainsi produite ne contient plus de sels, il suffit alors de condenser cette

vapeur pour obtenir de l'eau douce liquide. Il s'agit en fait d'accélérer le cycle naturel de l'eau. En effet l'eau s'évapore naturellement des océans, la vapeur s'accumule dans les nuages puis l'eau douce retombe sur terre par les précipitations. Ce principe de dessalement très simple a été utilisé dès l'antiquité pour produire de très faibles quantités d'eau douce sur les bateaux.

Le procédé de distillation dit Flash consiste à maintenir l'eau sous pression pendant toute la durée du chauffage ; lorsqu'elle atteint une température de l'ordre de 120°C, elle est introduite dans une enceinte (ou étage) où règne une pression réduite. Il en résulte une vaporisation instantanée par détente, appelée Flash. Une fraction de l'eau s'évapore (voir figure 3) puis va se condenser sur les tubes condenseurs placés en haut de l'enceinte, et l'eau liquide est recueillie dans des réceptacles en dessous des tubes. C'est l'eau de mer chaude qui se refroidit pour fournir la chaleur de vaporisation, l'ébullition s'arrête quand l'eau de mer atteint la température d'ébullition, correspondant à la pression régnant dans l'étage considéré. Le phénomène de Flash est reproduit ensuite dans un deuxième étage où règne une pression encore plus faible. La vaporisation de l'eau est ainsi réalisée par détentes successives dans une série d'étages où règnent des pressions de plus en plus réduites. On peut trouver jusqu'à 40 étages successifs dans une unité MSF industrielle.
Pour chauffer l'eau de mer jusqu'à 120°C, celle-ci circule d'abord dans les tubes des condenseurs des différents étages en commençant d'abord par le dernier étage où la température est la plus faible, elle est alors préchauffée en récupérant la chaleur de condensation de la vapeur d'eau. Elle est finalement portée à 120 °C grâce à de la vapeur à une température supérieure à 120°C produite par une chaudière ou provenant d'une centrale de production d'électricité.

Nous remarquons lors du phénomène de flash que des gouttelettes d'eau salée peuvent être entraînées avec la vapeur, elles sont séparées grâce à un dévésiculeur constitué par une sorte de grillage qui limite le passage des gouttelettes qui retombent alors au fond de l'enceinte.

L'avantage principal du procédé MSF est que l'évaporation de l'eau de mer ne se produit pas autour des tubes de chauffage, puisque le liquide « flash », ce qui limite les risques d'entartrage.

L'énergie requise est principalement l'énergie thermique à fournir à la chaudière. Cette énergie peut être peu coûteuse si on récupère de la vapeur basse pression à la sortie d'une turbine de centrale électrique. Il faut également fournir de l'énergie électrique pour les pompes de circulation de l'eau de mer.

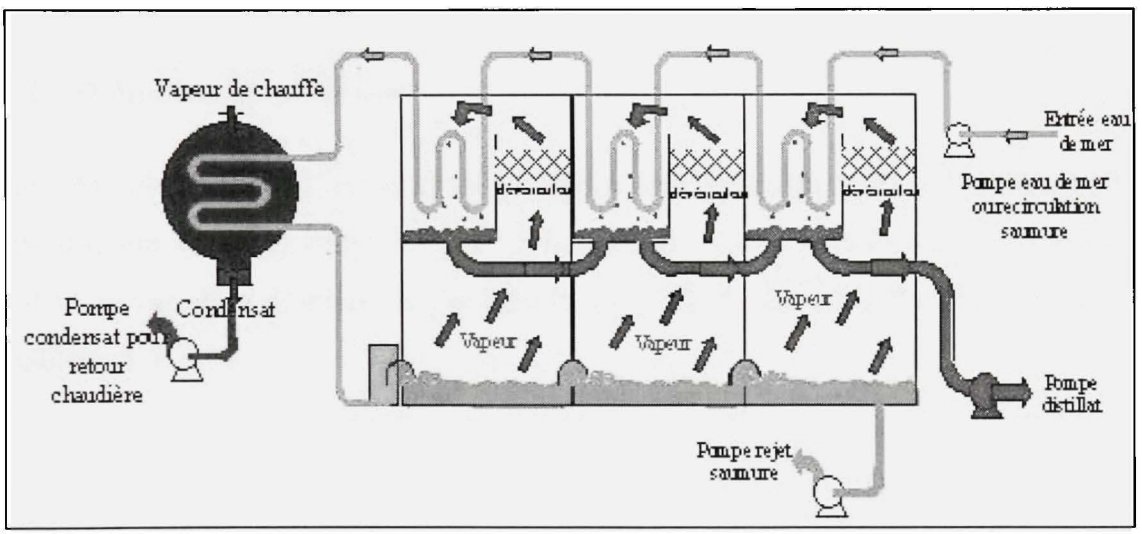


Figure 5.1 ***Principe de fonctionnement d'un système par détentes successives (MSF) à 3 étages***

5.2.1.2 Facteurs qui influent sur les coûts de production

Il y a trois facteurs essentiels qui influent sur les coûts de production d'eau douce à partir d'eau salée :

- Le niveau de salinité et les caractéristiques physico-chimiques plus ou moins complexes;

- Le coût de l'énergie nécessaire aux processus de transformation qui varie entre 45% et 75 % du coût total selon les procédés;

- La taille des unités de production (m^3/j produits).

D'une manière générale, la réduction du coût de production d'eau potable à partir d'eau salée passe par l'amélioration technologique des procédés et par le potentiel à réduire le coût énergétique.

5.2.2 Définition du problème

Dans cette étude deux objectifs de performance sont considérés soit la maximisation de la production d'eau distillée (DF) et la minimisation du taux de perte d'écoulement (BDF).Les variables d'entrée de l'opération qui influent sur ces objectifs sont données au tableau 5.1.

Tableau 5.1

Paramètres de conception

Paramètres	**Nomenclature**	**Bas niveau**	**Haut Niveau**
Température d'entrée de l'eau de mer (°C)	SWIT (X_A)	24	35
Différence de température (°C)	TD (X_B)	5,2	8,0
Dernier niveau de salinité (mm)	LSBL (X_C)	50	850
Premier niveau de salinité (mm)	FSBL (X_D)	40	320
Taux d'écoulement de la pompe de recyclage d'eau salée (m3/h)	BRPF (X_E)	8200	11 500

5.2.3 Modélisation avec RSM

Pour exprimer nos objectifs en fonction des variables de décision, il faut utiliser la modélisation à l'aide des surfaces de réponse.

5.2.3.1 Réponses du système

Dans cette étude, les réponses sont de type quantitatif :

- Taux de production d'eau distillée

  Il s'agit de maximiser le taux de production d'eau distillé de l'installation. Le taux minimal et le taux maximal de production de l'installation sont donnés dans la description du problème. Selon les réglages du processus de MSF nous pouvons déterminer le taux de production d'eau distillée et les valeurs des facteurs qui maximisent ce taux.

- Taux de perte d'écoulement

  Il s'agit de minimiser le taux de perte d'énergie du système due à l'écoulement.

5.2.3.2 Les facteurs indépendants

Les variables d'entrée de l'opération qui influent sur ces objectifs et qui constituent les facteurs de l'expérimentation sont :

- Température d'entrée de l'eau de mer (°C)

  Il s'agit de la température que doit avoir l'eau de mer au moment où elle rentre dans le système de dessalement. En effet, l'eau de mer doit être portée à une certaine température avant de pénétrer dans le système de MSF pour en optimiser l'utilisation.

- Différence de température (°C)

Il s'agit de la différence de température entre l'eau de mer qui est entrée dans le système et l'eau distillée qui en sort. Cette différence dépend du réglage de température des étages du système.

- Dernier niveau de salinité (mm)
  Il s'agit de la quantité de sel récupérée au sortir du dernière étage du procédé de dessalement.

- Premier niveau de salinité (mm)
  Il s'agit de la quantité de sel récupérée au sortir du premier étage du procédé de dessalement.

- Taux d'écoulement de la pompe de recyclage d'eau salée (m^3/h)
  Il s'agit du taux d'écoulement de l'eau de mer qui pénètre dans la pompe de recyclage au début du procédé MSF.

5.2.3.3 Le plan d'expérience

Le plan d'expérience utilisé pour la modélisation est le même que celui utilisé par les auteurs du problème [Abdul-Wahab et Abdo, 2007]. N'étant pas en possession du matériel permettant d'effectuer les expériences physiques décrites par les auteurs nous avons considéré les résultats de leurs expérimentations pour notre modélisation. Les auteurs ont eu recours à un plan factoriel complet à deux niveaux qui totalise 64 essais plus cinq essais aux points centraux. Les variables de conception sont codées sur deux niveaux soit (-1) pour la plus faible valeur et (+1) pour la plus élevée.

Le plan d'expérience obtenu nécessite d'effectuer 64 essais plus cinq essais aux points centraux, nombre d'essai proposé par des tables et des logiciels statistiques, comme JMP 5.1 ou Statgraphics. Ces logiciels permettent de déterminer les différents essais à

effectuer pour obtenir une modélisation des plus fiables pour nos différentes réponses. Leur utilisation a permis d'obtenir la matrice d'expériences suivante qui a été complétée par les résultats des expérimentations menées par les auteurs du problème [Abdul-Wahab et Abdo, 2007]

Tableau 5.2.A

Matrice des expériences et résultats expérimentaux des réponses.[Abdul-Wahab et Abdo, 2007]

No. Test	Facteurs					Réponses	
	X_A SWIT C	X_B TD en C	X_C LSBL en mm	X_D FSBL en mm	X_E BRPF en m^3/h	BDF	DF
1	1	-1	1	1	-1	1047,10	1910,42
2	1	-1	-1	1	-1	817,69	901,22
3	-1	-1	1	1	-1	821,78	1877,87
4	1	1	1	1	1	1274,95	1920,71
5	-1	1	1	-1	1	1274,11	1812,11
6	0	0	0	0	0	1246,97	1439,69
7	-1	1	-1	1	-1	876,80	1036,54
8	0	0	0	0	0	1241,99	1456,81
9	1	1	-1	-1	-1	973,47	1101,85
10	-1	1	-1	1	1	1195,37	1103,19
11	1	-1	-1	1	1	1091,78	923,14
12	1	1	-1	-1	1	1146,93	1042,18
13	-1	-1	1	-1	-1	863,78	1740,69
14	-1	1	1	-1	-1	789,99	1719,52
15	-1	1	1	1	-1	964,82	1901,01
16	1	-1	-1	-1	-1	891,32	890,11
17	1	1	-1	1	-1	904,29	1000,91
18	1	1	-1	1	1	1127,91	946,87
19	-1	-1	-1	-1	-1	940,01	1056,31
20	1	1	1	-1	-1	970,12	1831,42
21	1	-1	1	1	1	1272,63	1936,70
22	-1	1	1	-1	-1	851,36	1612,32
23	-1	-1	-1	1	1	1107,94	1123,80
24	1	-1	-1	-1	1	1112,24	1112,40
25	-1	-1	1	-1	1	1017,82	1811,47
26	-1	-1	-1	-1	1	1149,54	911,09
27	1	-1	1	-1	-1	895,40	1778,97
28	1	1	-1	-1	1	1214,47	951,59
29	1	-1	1	-1	1	1210,34	1638,33
30	-1	-1	1	1	1	1081,29	1621,12
31	1	1	1	-1	-1	952,75	1830,99
32	1	1	1	-1	1	1243,85	1961,66
33	-1	-1	-1	-1	-1	930,12	1010,24
34	1	1	1	1	-1	1095,30	2001,87
35	-1	-1	-1	1	1	995,67	1066,08
36	1	-1	-1	1	-1	845,77	1102,50
37	0	0	0	0	0	1273,14	1430,13
38	-1	-1	1	-1	-1	825,51	1890,25
39	1	1	1	1	-1	984,77	1950,14
40	-1	1	1	-1	1	1089,34	1631,40
41	-1	1	1	1	1	1251,79	1781,09

Tableau 5.3.B

Matrice des expériences et résultats expérimentaux des réponses. [Abdul-Wahab et Abdo, 2007]

No. Test	Facteurs					Réponses	
	X_A SWIT C	X_B TD en C	X_C LSBL en mm	X_D FSBL en mm	X_E BRPF en m^3/h	BDF	DF
42	-1	1	-1	-1	-1	1002,14	1000,97
43	-1	1	-1	-1	1	1120,99	909,71
44	-1	-1	-1	1	-1	1101,20	1003,24
45	-1	1	-1	1	-1	861,31	1203,47
46	1	-1	-1	1	1	1232,45	1005,07
47	-1	-1	1	1	1	1247,96	1836,99
48	1	1	-1	1	1	1017,88	1071,22
49	1	1	-1	-1	-1	957,44	1012,44
50	1	-1	1	1	1	1228,97	1930,01
51	0	0	0	0	0	1266,74	1410,60
52	-1	1	-1	-1	1	1240,73	875,33
53	-1	-1	1	1	-1	1074,99	1920,58
54	-1	-1	-1	1	-1	947,31	832,65
55	1	-1	1	-1	-1	921,54	1810,04
56	-1	1	1	1	1	1284,34	1878,11
57	-1	1	1	1	-1	987,54	1883,20
58	1	-1	-1	-1	1	1175,41	830,53
59	1	1	1	1	1	1242,31	1957,23
60	-1	1	-1	-1	-1	950,33	1090,01
61	1	-1	1	1	-1	945,82	1893,21
62	1	1	1	-1	1	1270,91	1885,74
63	-1	-1	-1	-1	1	1107,84	1017,85
64	1	-1	1	-1	1	1142,33	1795,25
65	-1	-1	1	-1	1	1058,72	1730,03
66	-1	1	-1	1	1	1207,93	1060,78
67	0	0	0	0	0	1264,92	1581,47
68	1	-1	-1	-1	-1	830,74	1009,87
69	1	1	-1	1	-1	915,73	946,77

Code	Niveaux				
	X_A	X_B	X_C	X_D	X_E
-1	24	5,2	50	40	8200
1	35	8,0	850	320	11500

5.2.3.4 Formulation des modèles mathématiques

- Validation du modèle

  La modélisation à l'aide du plan d'expérience nous donne un modèle de régression linéaire codé pour chacune des réponses de l'étude.

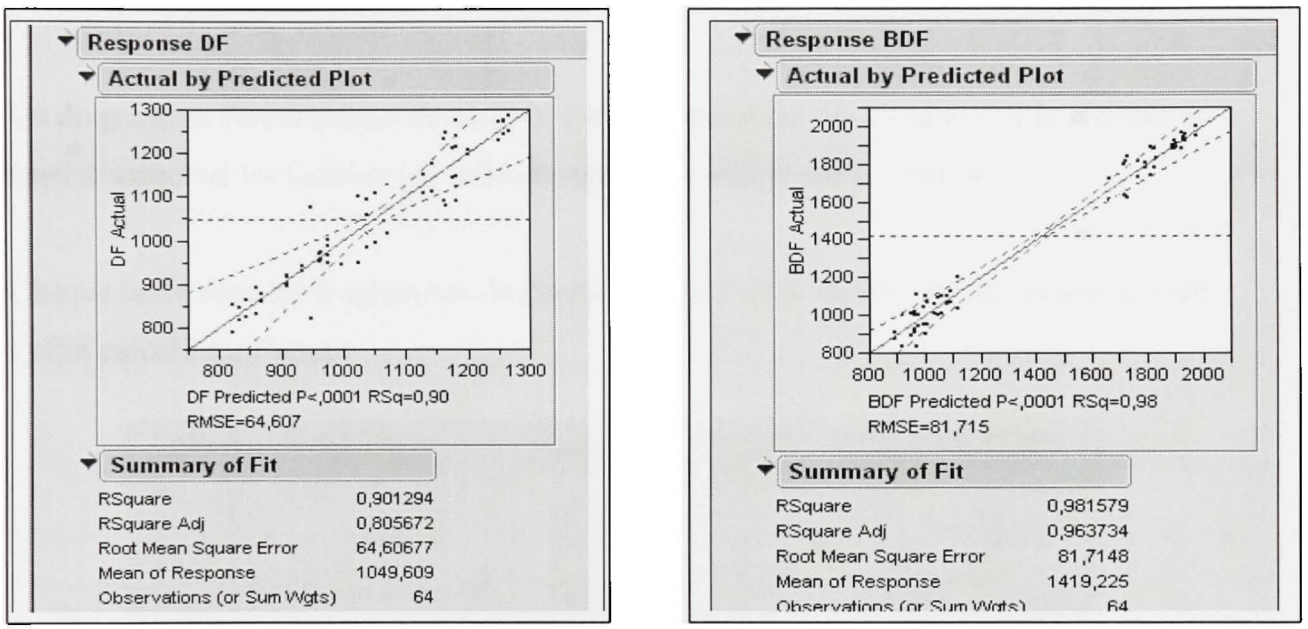


Figure 5.2 ***Modélisation avec RSM, représentation graphique des réponses du plan***

Les modèles obtenus ont des facteurs de corrélation de 0,91 pour le taux de production d'eau distillée (DF) et 0,98 pour le taux de perte d'écoulement (BDF) ce qui permet de conclure qu'ils représentent adéquatement les relations entre les facteurs et les réponses. Puisqu'ils ont des facteurs de corrélation supérieurs à 0,9.

- Analyse des résultats et discussions

  Dans la modélisation de chaque réponse du système, tous les facteurs ou combinaisons de facteurs ne sont pas considérés. Ne sont considérés que ceux dont l'impact sur la réponse est jugé important par l'expérimentateur. Pour déterminer cela, il faut utiliser le diagramme de Pareto avec un seuil de confiance fixé.

Un diagramme Pareto permet de montrer l'effet standardisé des facteurs sur la réponse et ainsi d'identifier les facteurs qui influent significativement sur la réponse.

Chaque barre dans les diagrammes de Pareto illustre l'effet standardisé du facteur qui est l'effet calculé sur l'erreur.

Term	Scaled Estimate	Std Error	t Ratio	Prob>|t|
Intercept	1049,6091	8,075847	129,97	<,0001
A	11,4725	8,075847	1,42	0,1651
B	20,4525	8,075847	2,53	0,0164
C	18,648125	8,075847	2,31	0,0275
D	14,559375	8,075847	1,80	0,0808
E	120,28906	8,075847	14,89	<,0001
A*B	-0,716563	8,075847	-0,09	0,9299
A*C	26,463437	8,075847	3,28	0,0025
B*C	6,8059375	8,075847	0,84	0,4056
A*D	-10,30656	8,075847	-1,28	0,2111
B*D	-10,05594	8,075847	-1,25	0,2221
C*D	30,080937	8,075847	3,72	0,0008
A*E	6,464375	8,075847	0,80	0,4293
B*E	9,8875	8,075847	1,22	0,2298
C*E	10,9325	8,075847	1,35	0,1853
D*E	-5,634375	8,075847	-0,70	0,4904
A*B*C	-3,365	8,075847	-0,42	0,6797
A*B*D	-4,621875	8,075847	-0,57	0,5711
A*B*E	-25,05719	8,075847	-3,10	0,0040
A*C*D	-4,045625	8,075847	-0,50	0,6198
A*C*E	-8,092813	8,075847	-1,00	0,3238
A*D*E	-0,343438	8,075847	-0,04	0,9663
B*C*D	5,6275	8,075847	0,70	0,4909
B*C*E	4,8253125	8,075847	0,60	0,5544
B*D*E	1,2028125	8,075847	0,15	0,8825
C*D*E	-2,954687	8,075847	-0,37	0,7169
A*B*C*D	-1,275313	8,075847	-0,16	0,8755
A*B*C*E	9,38625	8,075847	1,16	0,2537
A*B*D*E	-11,43875	8,075847	-1,42	0,1663
A*C*D*E	-2,426875	8,075847	-0,30	0,7657
B*C*D*E	-10,92813	8,075847	-1,35	0,1855
A*B*C*D*E	13,185937	8,075847	1,63	0,1123

Figure 5.3 ***Diagramme Pareto du taux de production d'eau distillé (DF)***

Term	Scaled Estimate	Std Error	t Ratio	Prob>|t|
Intercept	1419,2247	10,21435	138,94	<,0001
A	14,567812	10,21435	1,43	0,1635
B	15,53625	10,21435	1,52	0,1381
C	414,53938	10,21435	40,58	<,0001
D	34,76625	10,21435	3,40	0,0018
E	-10,51281	10,21435	-1,03	0,3111
A*B	14,020625	10,21435	1,37	0,1794
A*C	28,71125	10,21435	2,81	0,0084
B*C	-1,892813	10,21435	-0,19	0,8542
A*D	-6,184375	10,21435	-0,61	0,5491
B*D	8,1671875	10,21435	0,80	0,4298
C*D	18,985937	10,21435	1,86	0,0723
A*E	8,5096875	10,21435	0,83	0,4110
B*E	0,059375	10,21435	0,01	0,9954
C*E	-2,754375	10,21435	-0,27	0,7892
D*E	4,15375	10,21435	0,41	0,6870
A*B*C	12,762813	10,21435	1,25	0,2205
A*B*D	-25,63344	10,21435	-2,51	0,0173
A*B*E	5,744375	10,21435	0,56	0,5778
A*C*D	12,925313	10,21435	1,27	0,2149
A*C*E	5,918125	10,21435	0,58	0,5664
A*D*E	-3,15625	10,21435	-0,31	0,7593
B*C*D	-0,156875	10,21435	-0,02	0,9878
B*C*E	19,306563	10,21435	1,89	0,0678
B*D*E	-6,494687	10,21435	-0,64	0,5294
C*D*E	-20,65781	10,21435	-2,02	0,0516
A*B*C*D	-2,8525	10,21435	-0,28	0,7818
A*B*C*E	-12,40594	10,21435	-1,21	0,2334
A*B*D*E	1,2390625	10,21435	0,12	0,9042
A*C*D*E	17,125937	10,21435	1,68	0,1034
B*C*D*E	-7,985	10,21435	-0,78	0,4401
A*B*C*D*E	-16,6075	10,21435	-1,63	0,1138

Figure 5.4 ***Diagramme Pareto du taux de perte d'écoulement (BDF)***

Tableau 5.4

Légende

Termes du graphique	Signification
«Term»	Facteurs de l'étude
«Scaled estimated»	Effets des facteurs sur la réponse
«Standard Error»	Erreur standard
«T-ratio»	Effet standardisé des facteurs, soit l'effet calculé sur l'erreur, qui est représenté par chaque barre dans le diagramme de pareto.

Les graphiques de Pareto (figure 5.3 et 5.4) classent les facteurs et leurs interactions en fonction de leur influence sur la réponse. Tous les facteurs et interactions ayant un effet standardisé (T-ratio) inférieur à la limite des facteurs significatifs (intervalle de confiance de 95%) sont négligés et ne sont pas représentés dans la modélisation.

À partir de cela nous pouvons déterminer les modèles mathématiques des objectifs de l'étude sous formes d'équation codées.

$$DF = 1041,61 + 20.45X_B + 18.65X_C + 120.29X_E + 26.46X_AX_C + 30.08X_CX_D - 25.06X_AX_BX_E \quad (5\text{-}1)$$

$$BDF = 1419.22 + 414.54X_C + 34.77X_D + 28.71X_AX_C - 25.63X_AX_BX_D \quad (5\text{-}2)$$

Où :

- DF et BDF : Réponse codée du système;

- X_A, X_B, X_C, X_D et X_E : les variables indépendantes codées.

La Figure 5.5 présente les effets des principaux facteurs sur les réponses. De cette figure, il ressort que le facteur le plus influant sur le taux de production d'eau distillé (DF) est le taux d'écoulement de la pompe de recyclage d'eau salée (Facteur E). À cela il faut ajouter les interactions de facteurs (AB, CD, ABE, etc.). Quant au taux de perte d'écoulement (BDF), le facteur qui influe le plus dessus est le dernier niveau de salinité (Facteur C)

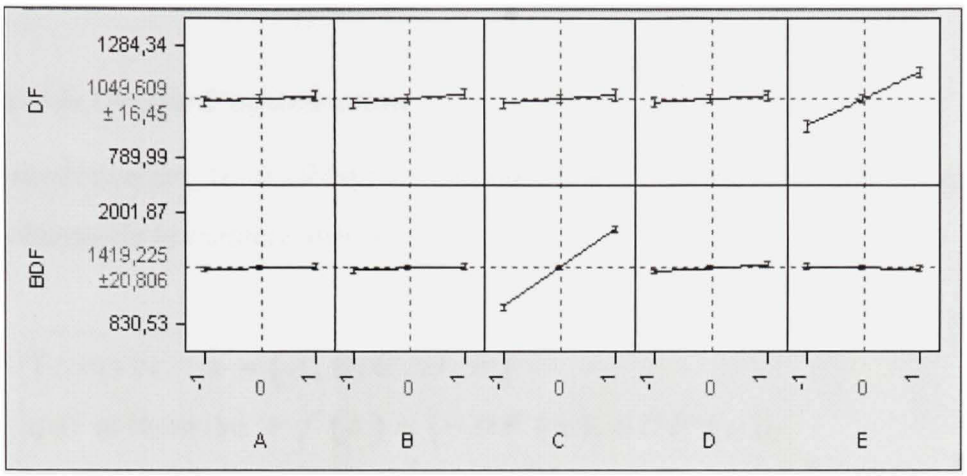


Figure 5.5 ***Graphiques des effets des principaux facteurs sur les objectifs de l'étude.***

5.2.4 Optimisation multiobjective

Une fois les différents objectifs exprimés par un modèle mathématique, la procédure d'optimisation peut être abordée.

Des informations complémentaires sur les objectifs de l'étude sont données dans le tableau 5.5 :

Tableau 5.5

Les contraintes sur les objectifs de l'étude

Objectifs	But	Limite inférieure	Limite supérieure	**pondération**
BDF	Minimiser	830.53	2001.87	**3**
DF	**Maximiser**	**789.99**	**1284.34**	**5**

Les entrées de l'étape d'optimisation

Après la modélisation, le problème à optimiser grâce à notre approche multiobjective peut être résumé de la manière suivante :

$$\begin{aligned} &\text{Trouver} \quad x = [A, B, C, D, E]^T \\ &\text{qui minimise} \quad f(x) = \{-DF(x), BDF(x)\} \\ &\text{sujet à} \quad DF(x) \leq 1284.34 \\ &\qquad DF(x) \geq 789.99 \\ &\qquad BDF(x) \leq 2001.87 \\ &\qquad BDF(x) \geq 830.53 \\ &\qquad -1 \leq A, B, C, D, E \leq 1 \end{aligned} \tag{5-3}$$

La fonction de fitness qui exprime les objectifs de l'étude est:

$$fitness = \left(\frac{5}{8} * \left(\frac{DF_{\max} - DF_i}{DF_{\max} - DF_i^{Pareto}} \right)^2 + \frac{3}{8} * \left(\frac{BDF_i - BDF_{\min}}{BDF_i^{Pareto} - BDF_{\min}} \right)^2 \right)^{\frac{1}{2}} \tag{5-4}$$

La minimisation de cette fonction de fitness nous permet d'atteindre notre objectif de minimisation de BDF et de maximisation de DF.

En utilisant le programme que nous avons élaboré sur Matlab pour l'optimisation multiobjective dans le domaine continu avec les colonies de fourmis, nous obtenons comme résultat un ensemble de solutions Pareto optimales.

Le réglage des paramètres de l'algorithme est effectué par essai-erreur.
Les résultats du problème posé sont obtenus avec les paramètres suivants :

- Nombre de fourmis : NA= 50;

- Nombre maximal d'itération : $G_{max} = 300$;
- $K_1 = 0.1$;
- $K_2 = 0.002$.

Les résultats

La figure 5.6 nous permet de constater que la meilleure solution est obtenue après environs 260 itérations. La forme en escalier de notre courbe s'explique par le fait qu'un effet mémoire a été utilisé dans notre programme. Ainsi, lorsqu'une itération donne une moins bonne solution que la précédente, la solution de l'itération précédente est conservée.

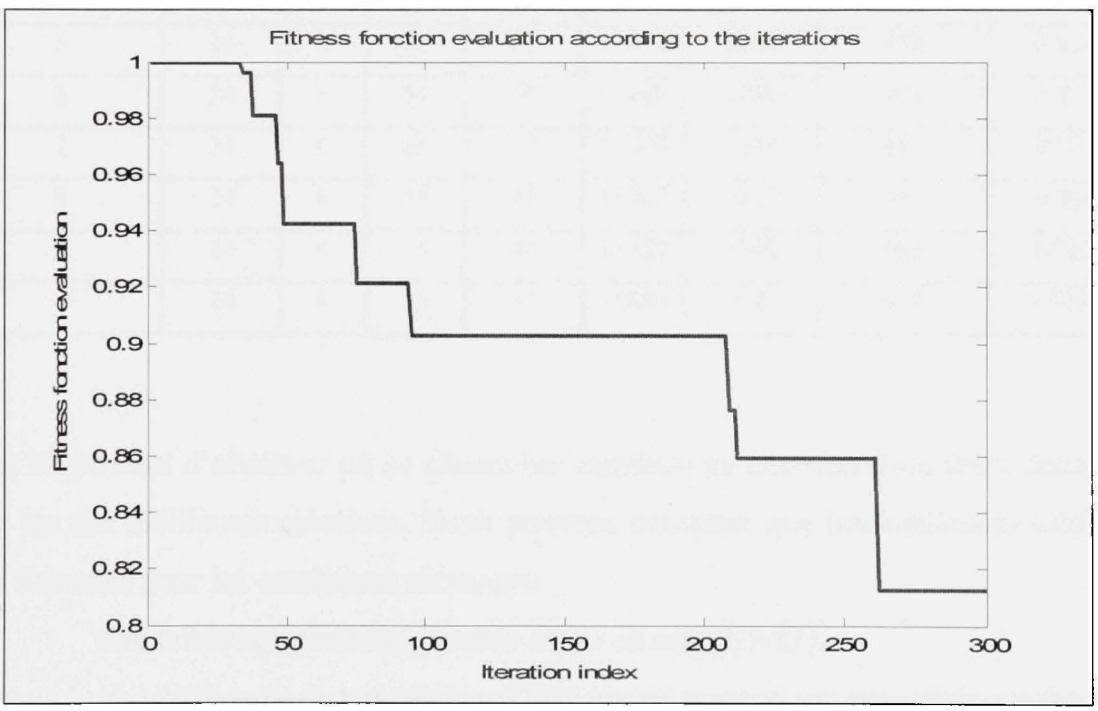


Figure 5.6 ***Valeurs de la fonction de fitness selon le nombre d'itération***

Nous devons présenter à l'usager plus d'une solution pour lui donner une marge de manœuvre.

Le tableau 5.5 présente les résultats de l'étude. Ces solutions respectent les contraintes du problème et donnent des résultats qui minimisent BDF et maximisent DF tout en restant dans le domaine de chaque variable de décision.

Tableau 5.6

Les meilleures solutions

No. solutions	SWIT	TD	LSBL	FSBL	BRPF	DF	BDF	FITNESS
1	24	8	52	40	11495	1252	975	0.8125
2	24.3	8	56	41	11485	1247	980	0.8497
3	24.6	8	50	54	11477	1240	980	0.8584
4	24	8	52	46	11476	1248	978	0.859
5	24	8	52	41	11433	1244	976	0.876
6	24	8	54	50	11498	1249	981	0.877
7	24	8	55	47	11479	1246	981	0.887
8	24	8	58	42	11482	1247	983	0.902
9	24	8	73	40	11499	1248	996	0.921
10	24	8	50	41	11480	1249	974	0.9238

Il est intéressant d'observer où se situent les variables de décision dans leurs domaines pour les dix meilleures solutions. Nous pouvons constater que les meilleures solutions sont obtenues pour les conditions suivantes:

- une faible température d'entrée d'eau de mer (SWIT),
- une différence de température (TD) élevée et identique pour chacune des solutions,
- un dernier niveau de salinité faible,
- un premier niveau de salinité faible,
- un taux d'écoulement élevé de la pompe de recyclage du sel.

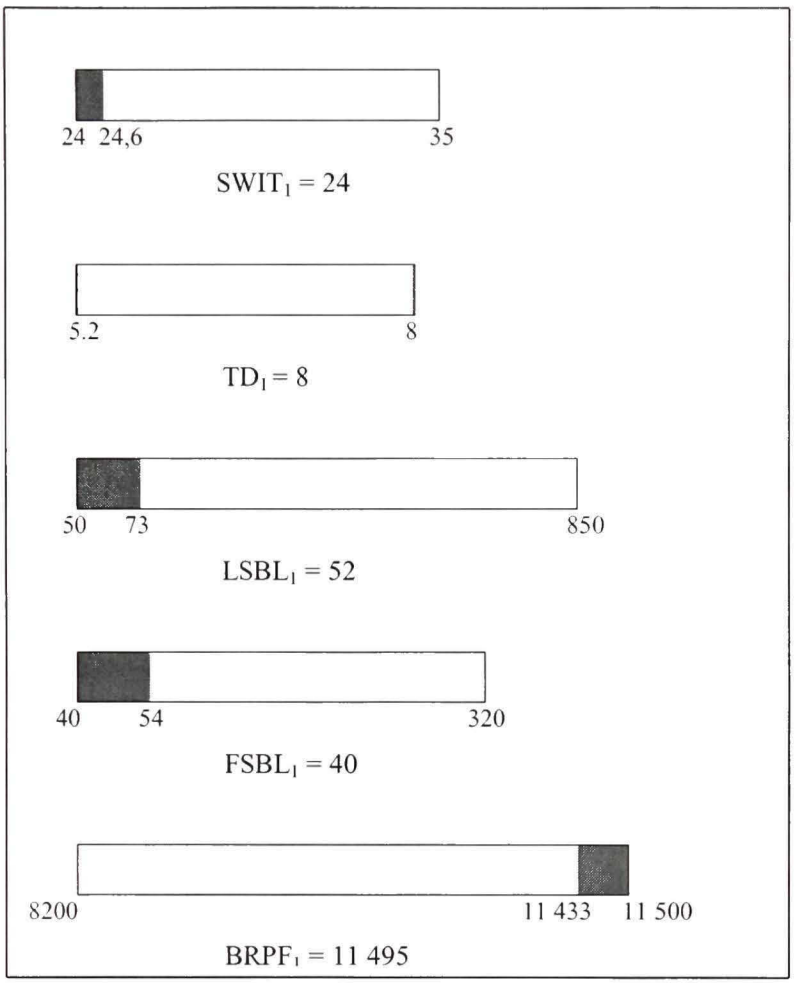


Figure 5.7 ***Marge de valeurs des variables pour les solutions optimales***

Comparaison avec les résultats des auteurs du problème.

Tableau 5.7

Résultats de CACO multiobjectifs versus la fonction de désirabilité

No. solutions	BDF désirabilité	BDF CACO multi objectifs	% d'amélioration de BDF	DF désirabilité	DF CACO multiobjectif	% d'amélioration de DF
1	991,725	975	**1,69%**	1224,52	1252	**2,24%**
2	1038,83	980	**5,66%**	1222,91	1247	**1,97%**
3	1033,34	980	**5,16%**	1213,11	1240	**2,22%**
4	1035,78	978	**5,58%**	1210,98	1248	**3,06%**
5	975,718	976	**-0,03%**	1181,75	1244	**5,27%**
6	965,137	981	**-1,64%**	1173,29	1249	**6,45%**
7	996,446	981	**1,55%**	1176,02	1246	**5,95%**
8	1035,8	983	**5,10%**	1166,87	1247	**6,87%**
9	973,657	996	**-2,29%**	1151,36	1248	**8,39%**
10	1005,05	974	**3,09%**	1141,94	1249	**9,38%**

Une comparaison entre les résultats obtenus avec la fonction de désirabilité et ceux de l'approche multiobjective développée montre que la seconde donne des résultats de meilleure qualité avec un pourcentage d'amélioration de l'ordre 2,24% pour le principal objectif et de 1.69% pour le secondaire. Si nous considérons les autres solutions présentées à l'usager, les améliorations peuvent aller jusqu'à 9% sur l'objectif premier et 5% sur l'objectif secondaire.

En observant la variation des valeurs de nos réponses pour les différentes solutions présentées, il est utile d'observer que les solutions obtenues avec notre approche varient beaucoup moins que celles obtenues avec la fonction de désirabilité. Il semble donc que nos solutions se trouvent dans un voisinage proche. Cela s'explique par le fait que nous avons fixé un faible déplacement à nos fourmis locales. Donc, lorsque la fonction de

fitness diminue, nos fourmis n'avancent que sur une courte distance avant de re-tester la valeur de la fonction de fitness et obtenir une nouvelle solution, si elle est meilleure que la précédente, ou se réorienter en cas de baisse de performance.

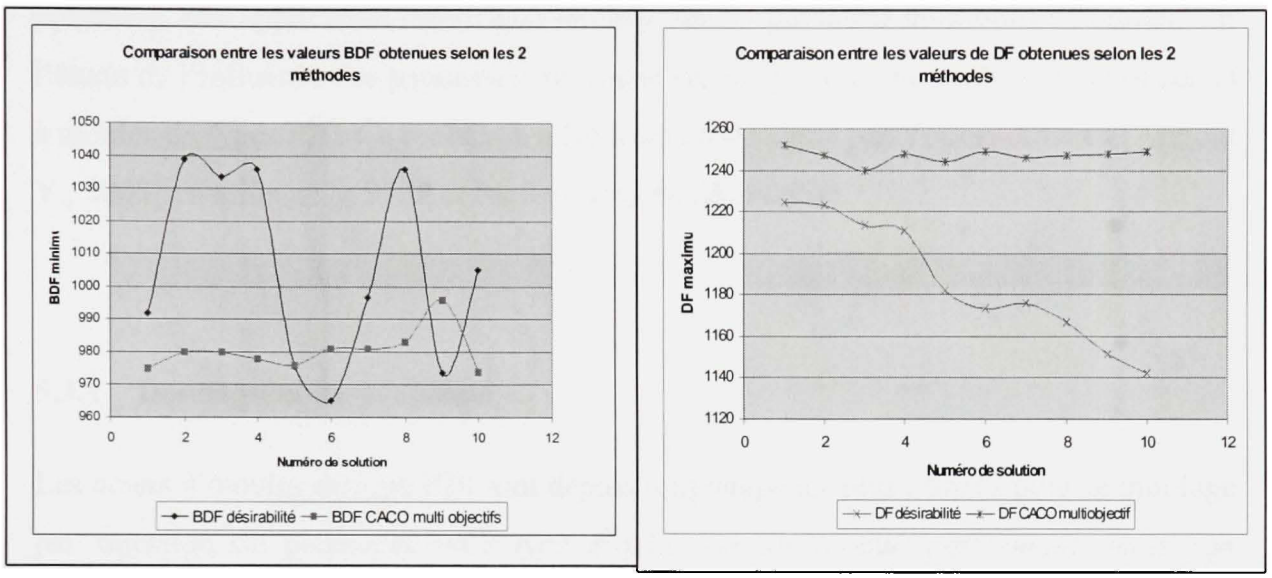


Figure 5.8 ***Représentation graphique des valeurs de BDF et DF des solutions optimales***

Suite à cette application, et à l'obtention de résultats appréciables, nous pouvons conclure que notre algorithme a prouvé son bon fonctionnement pour un problème d'optimisation à deux objectifs, en conduisant à des solutions cohérentes. De plus, notre algorithme a prouvé son efficacité en obtenant de meilleures solutions que celles des auteurs.

5.3 Application à un problème industriel multiobjectifs (à 5 objectifs) : Usinage à 5 axes des aciers à moule

Dans le but de compléter l'illustration des performances et la capacité de l'approche présentée, une application numérique est faite sur un problème multiobjectif portant sur l'étude de l'influence des paramètres de coupe sur les performances d'usinage des aciers à moules de types P20. Ce problème a été résolu à l'origine par Youcef Amtout [Amtout Y., 2003] en utilisant la RSM et les fonctions de désirabilité.

5.3.1 Description du problème

Les aciers à moules de type P20 sont depuis longtemps les plus utilisés pour le moulage par injection de plastiques. «*Ce type d'acier est un produit performant, mais son usinabilité varie et peut affecter la vie des outils, le temps de cycle, le temps de polissage pour la finition, la productivité et la structure même du métal suite à d'éventuels écrouissages dépendamment des conditions et des stratégies de coupe*» [Amtout Y., 2003].

De nouvelles stratégies d'usinage multiaxes à commande numérique allant du parcours et de l'orientation des outils à leur géométrie intrinsèque ne cessent d'évoluer. Une bonne combinaison des stratégies et des paramètres de coupe est essentielle pour générer les conditions optimales d'usinage à commande numérique de formes complexes incluant les moules. Plusieurs paramètres, en plus du matériau, influent sur les performances d'usinage des moules, tels que la vitesse de coupe, l'avance de l'outil, la profondeur de coupe, la largeur de coupe et l'orientation de l'outil par rapport aux pièces à usiner.

Dans son mémoire, Youcef Amtout [Amtout Y., 2003] étudie et analyse les comportements et l'influence de ces différents paramètres de coupe sur les performances de l'usinage en cinq axes de l'acier à moule de type P20, ainsi que leurs interactions. Pour ce faire, il propose une approche utilisant la théorie des surfaces de réponses pour modéliser et développer des équations de prédiction du temps d'usinage et du fini de surface en fonction des paramètres les plus significatifs. Par la suite, l'auteur utilise la fonction de désirabilité comme technique d'optimisation simultanée de plusieurs réponses du système afin de dégager des stratégies permettant le meilleur compromis des conditions de coupe et de générer des réglages combinatoires et optimaux de ces paramètres pour atteindre tous les objectifs souhaités.

Dans le cadre de cette application nous allons démontrer que dans la résolution du même problème d'optimisation multiobjectif, notre approche fournit de meilleurs résultats.

5.3.2 Modélisation avec RSM

Le plan de surface de réponse que nous utiliserons pour la modélisation est le même que celui utilisé par l'auteur [Amtout Y., 2003] dans son étude, un central composite à faces centré de Box Wilson. Ce plan fait partie de la famille des plans composites qui sont les plus appropriés pour la RSM à plus de deux objectifs. De plus, il nous est impossible de réaliser des expériences supplémentaires dans le cadre de cette étude n'étant pas en possession du matériel pour ces expérimentations, tout comme dans l'application précédente.

Les réponses du système

Les objectifs de performance à optimiser lors de cette étude et qui constituent les réponses de l'expérimentation sont les suivants :

- Rugosité moyenne dans la direction perpendiculaire à l'avance, (Ra);
- Rugosité maximale dans la direction perpendiculaire à l'avance, (Rt);
- Rugosité moyenne dans la direction de l'avance, (Ra //);
- Temps de cycle réel (Temps total des opérations du début à la fin), (Tc);
- Temps d'usinage réel (ou temps de coupe), (Tu);

Les facteurs indépendants

Les facteurs qui influent sur les réponses du système sont les variables indépendantes suivantes [Amtout Y., 2003] :

Tableau 5.8

Domaine des variables [Amtout Y., 2003]

Facteurs	**Nomenclature**	**Niveau bas (-1)**	**Niveau central (0)**	**Niveau Haut (+1)**
Vitesse d'avance de l'outil (m/min)	V	90	164.317	300
Avance par dent de l'outil (mm)	f	0.1	0.173	0.3
Profondeur de coupe (mm)	d	0.35	0.477	0.65
Largeur de coupe ou émersion (mm)	W	0.5	0.866	1.5
Angle d'engagement de l'outil (°)	α	0	4.56	30
Angle d'inclinaison (°)	β	0	4.56	30

Le plan d'expérience

Le plan d'expérience utilisé pour la modélisation est le plan de box-Wilson utilisant les surfaces de réponse aussi dénommé central composite (Central Composite Design). Celui-ci utilise 46 essais, nombre d'essai proposé par des tables et des logiciels de statistiques, comme JMP 5.1. Leur utilisation a permis d'obtenir la matrice d'expériences suivantes qui a été complétée par les résultats expérimentaux [Amtout Y., 2003]

Tableau 5.9.A

Matrice des expériences et résultats expérimentaux des réponses [Amtout Y., 2003]

Variables							Réponses				
Test no.	X_v Vitesse (m min)	X_f Avance (mm/tooth)	X_d Profondeur (mm)	X_W Emersion (mm)	X_α Angle d'engagement (degrees)	X_β Angle d'inclination (degrees)	Ra (μm)	Rt (μm)	Ra// (μm)	Tc (sec)	Tu (sec)
1	0	0	0	0	0	-1	4,72	17	1,65	263	170
2	-1	-1	-1	-1	1	1	2,06	12	0,46	1036	871
3	1	1	-1	-1	-1	-1	2,07	8,93	1,3	304	139
4	-1	1	1	-1	-1	-1	1,46	7,84	1	469	306
5	0	0	0	-1	0	0	1,7	9	1,04	432	293
6	-1	1	1	1	-1	1	8,78	32,96	1,87	172	104
7	-1	-1	-1	1	1	-1	8	36	1,81	365	296
8	1	0	0	0	0	0	4,47	24,87	1,9	198	106
9	0	0	0	0	0	0	3,92	19,98	1,46	261	170
10	-1	-1	1	1	1	1	7,17	32,56	0,63	369	297
11	0	-1	0	0	0	0	3,44	16	0,95	374	282
12	1	-1	-1	1	1	1	8,42	34,4	0,96	168	95
13	1	-1	-1	-1	1	-1	1,97	9,6	1,68	440	279
14	0	0	0	0	0	1	4	17,4	1,1	272	170
15	0	0	0	1	0	0	8,42	33,85	1,4	164	99
16	-1	-1	-1	1	-1	1	8,53	37,82	0,57	365	297
17	-1	1	1	1	1	-1	8,58	32,24	2,7	173	104
18	1	1	1	-1	-1	1	3,4	21,2	1,57	296	140
19	-1	1	-1	1	-1	-1	7,34	36,56	1,89	173	104
20	1	-1	-1	1	-1	-1	9,02	37	1,7	165	95
21	1	1	-1	-1	1	1	2,21	11,22	2,31	307	140
22	0	0	-1	0	0	0	3,71	17	1,31	262	170
23	-1	-1	1	-1	-1	1	2,25	11,14	0,45	1026	873
24	1	1	1	1	-1	-1	8,24	36,5	2,37	118	47
25	0	1	0	0	0	0	4,49	21,74	2,4	200	109
26	-1	-1	-1	-1	-1	-1	5	24	1,02	1035	873
27	-1	1	-1	1	1	1	8,25	34,12	2,32	176	104
28	1	-1	1	1	-1	1	8,68	34,32	0,62	165	95
29	0	0	0	0	-1	0	3,68	17,6	1,04	262	170
30	1	1	1	1	1	1	13,63	52,06	4	121	47
31	1	-1	-1	-1	-1	1	1,36	6,88	0,55	434	279
32	-1	1	-1	-1	-1	1	1,94	11	1,2	460	306
33	1	-1	1	-1	-1	-1	2,38	15,8	1,32	443	279
34	1	1	-1	1	1	-1	8,55	37,67	3,29	118	48
35	1	-1	1	-1	1	1	1,86	9,55	0,64	446	279

Tableau 5.10.B

Matrice des expériences et résultats expérimentaux des réponses [Amtout Y., 2003]

Variables							Réponses				
Test no.	X_v Vitesse (m min)	X_f Avance (mm/tooth)	X_d Profondeur (mm)	X_W Emersion (mm)	X_α Angle d'engagement (degrees)	X_β Angle d'inclination (degrees)	Ra (μm)	Rt (μm)	Ra// (μm)	Tc (sec)	Tu (sec)
36	-1	-1	1	-1	1	-1	1,99	10	1,05	1031	873
37	-1	-1	1	1	-1	-1	8,4	42,19	1,18	366	297
38	-1	0	0	0	0	0	4	16	1,07	387	297
39	0	0	0	0	0	0	3,83	19,44	1,34	262	170
40	-1	1	1	-1	1	1	2,3	11,87	1,7	470	305
41	-1	1	-1	-1	1	-1	1,3	8,8	1,8	466	307
42	1	1	-1	1	-1	1	6,82	35,01	3	117	47
43	1	-1	1	1	1	-1	9,24	48,39	2,4	165	95
44	0	0	0	0	1	0	5,16	17,22	1,54	275	170
45	1	1	1	-1	1	-1	2,4	18	2,76	300	141

Code	Niveaux					
	X_v	X_f	X_d	X_W	X_α	X_β
-1	90	0.1	0.35	0.5	0	0
0	164.317	0.173	0.477	0.866	4.56	4.56
1	300	0.3	0.65	1.5	30	30

Formulation des modèles mathématiques

- Validation du modèle

  La modélisation à l'aide de la RSM nous donne un modèle de régression linéaire codé pour chacune des réponses de l'étude.

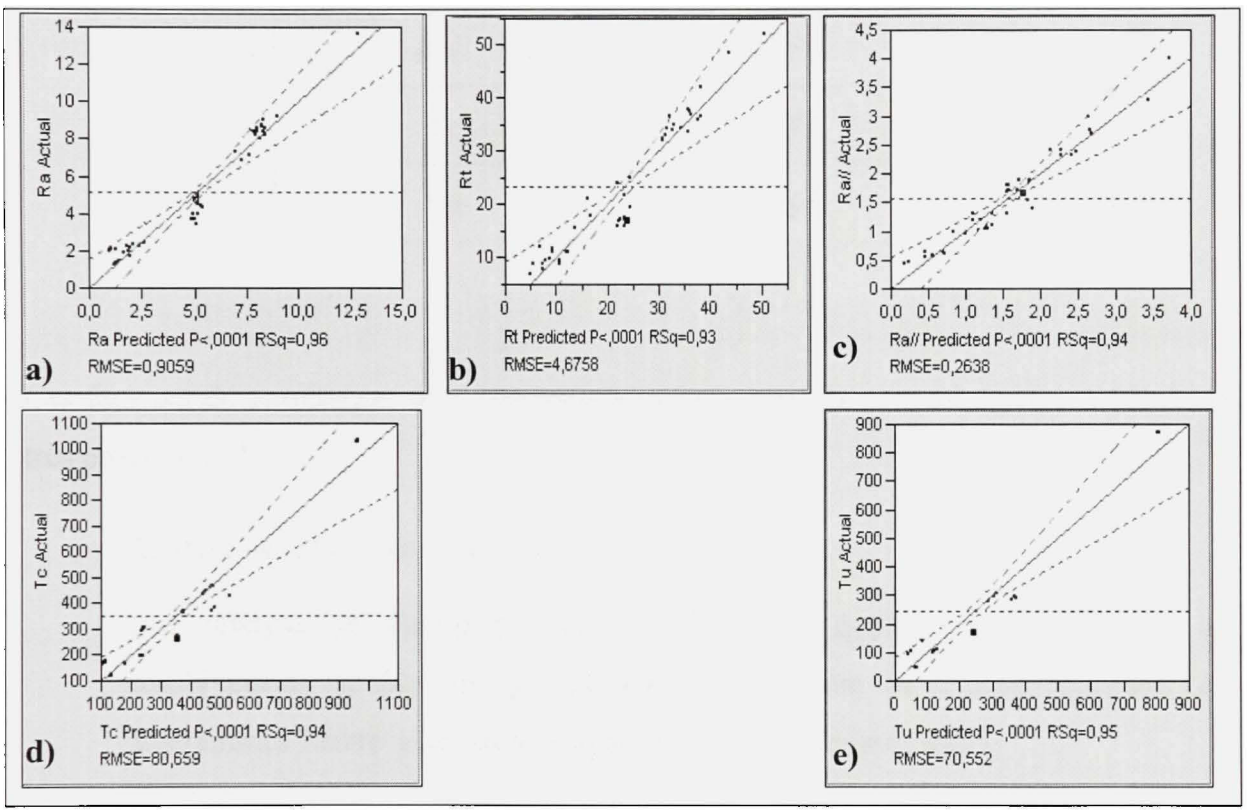


Figure 5.9 ***Modélisation avec RSM Représentation graphique des réponses du plan***

Les modèles obtenus à l'aide de la RSM ont des facteurs de corrélation présentés dans le tableau 5.11 :

Tableau 5.11

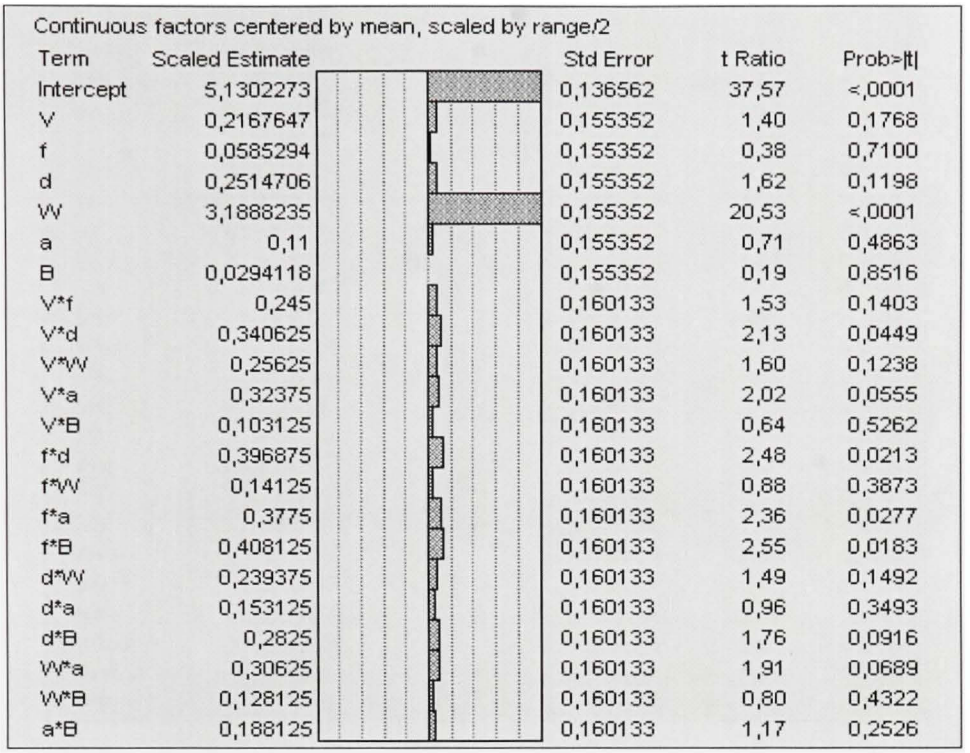
Facteur de corrélation de la modélisation de chaque réponse

Réponse du système	**Facteur de corrélation**
Ra	0,96
Rt	0,93
Ra//	0,94
Tc	0,94
Tu	0,95

Ces facteurs de corrélation sont tous satisfaisants puisqu'ils sont supérieurs à 0,9 donc très proches de 1.

- Analyse des résultats et discussion

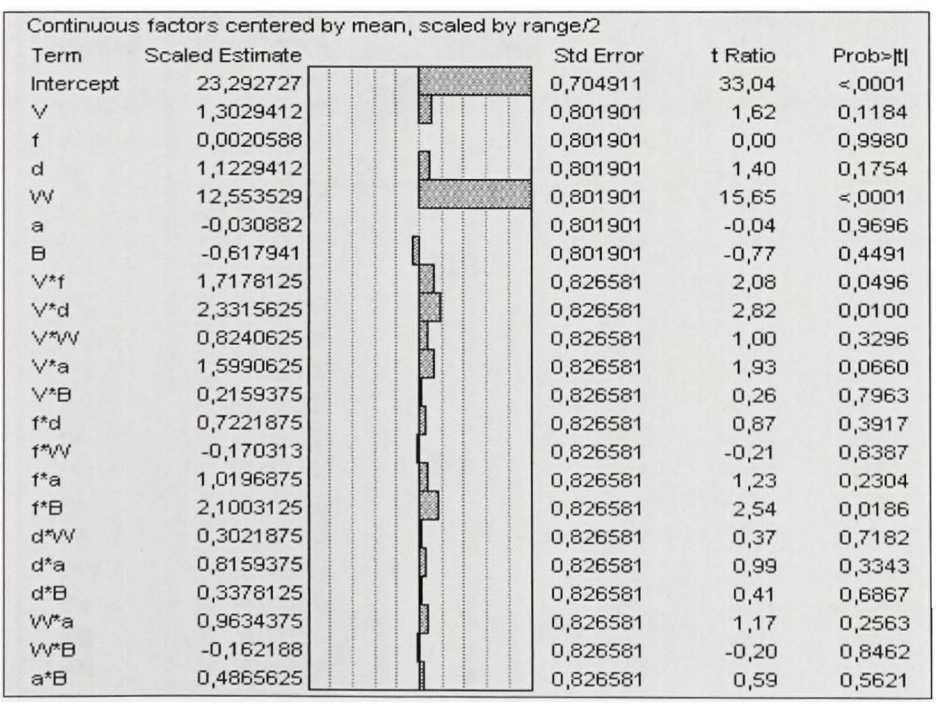
  Une analyse de Pareto permet de déterminer quels sont les facteurs et combinaisons de facteurs qui influent sur la valeur de chaque réponse. Les diagrammes Pareto associés à chaque réponse sont les suivants :



Continuous factors centered by mean, scaled by range/2

Term	Scaled Estimate	Std Error	t Ratio	Prob>|t|
Intercept	5,1302273	0,136562	37,57	<,0001
V	0,2167647	0,155352	1,40	0,1768
f	0,0585294	0,155352	0,38	0,7100
d	0,2514706	0,155352	1,62	0,1198
W	3,1888235	0,155352	20,53	<,0001
a	0,11	0,155352	0,71	0,4863
B	0,0294118	0,155352	0,19	0,8516
V*f	0,245	0,160133	1,53	0,1403
V*d	0,340625	0,160133	2,13	0,0449
V*W	0,25625	0,160133	1,60	0,1238
V*a	0,32375	0,160133	2,02	0,0555
V*B	0,103125	0,160133	0,64	0,5262
f*d	0,396875	0,160133	2,48	0,0213
f*W	0,14125	0,160133	0,88	0,3873
f*a	0,3775	0,160133	2,36	0,0277
f*B	0,408125	0,160133	2,55	0,0183
d*W	0,239375	0,160133	1,49	0,1492
d*a	0,153125	0,160133	0,96	0,3493
d*B	0,2825	0,160133	1,76	0,0916
W*a	0,30625	0,160133	1,91	0,0689
W*B	0,128125	0,160133	0,80	0,4322
a*B	0,188125	0,160133	1,17	0,2526

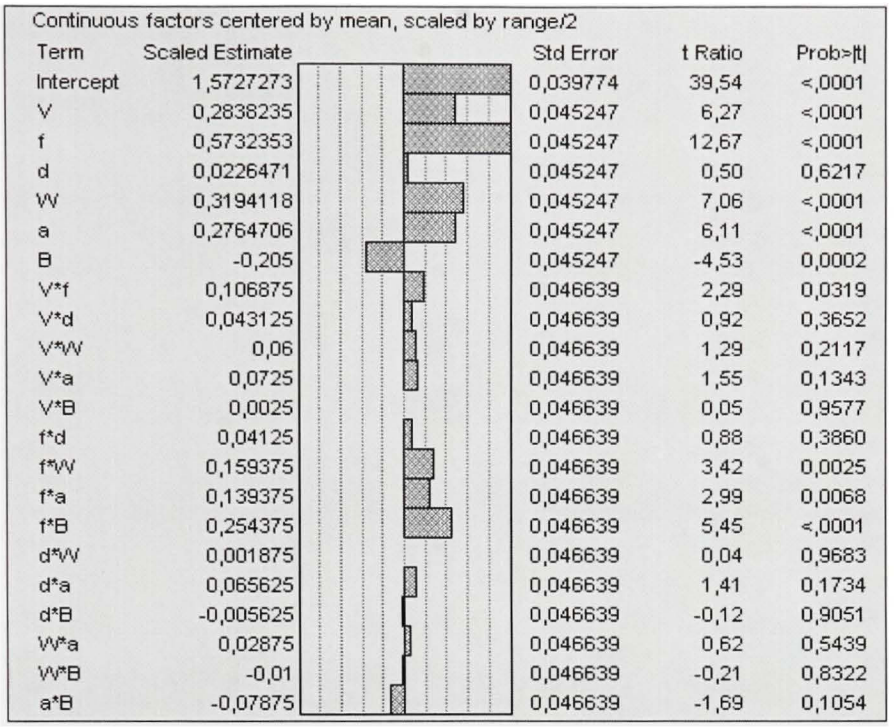
Figure 5.10 ***Diagramme Pareto de la rugosité moyenne dans la direction perpendiculaire à l'avance (Ra).***



Continuous factors centered by mean, scaled by range/2

Term	Scaled Estimate	Std Error	t Ratio	Prob>|t|
Intercept	23,292727	0,704911	33,04	<,0001
V	1,3029412	0,801901	1,62	0,1184
f	0,0020588	0,801901	0,00	0,9980
d	1,1229412	0,801901	1,40	0,1754
W	12,553529	0,801901	15,65	<,0001
a	-0,030882	0,801901	-0,04	0,9696
B	-0,617941	0,801901	-0,77	0,4491
V*f	1,7178125	0,826581	2,08	0,0496
V*d	2,3315625	0,826581	2,82	0,0100
V*W	0,8240625	0,826581	1,00	0,3296
V*a	1,5990625	0,826581	1,93	0,0660
V*B	0,2159375	0,826581	0,26	0,7963
f*d	0,7221875	0,826581	0,87	0,3917
f*W	-0,170313	0,826581	-0,21	0,8387
f*a	1,0196875	0,826581	1,23	0,2304
f*B	2,1003125	0,826581	2,54	0,0186
d*W	0,3021875	0,826581	0,37	0,7182
d*a	0,8159375	0,826581	0,99	0,3343
d*B	0,3378125	0,826581	0,41	0,6867
W*a	0,9634375	0,826581	1,17	0,2563
W*B	-0,162188	0,826581	-0,20	0,8462
a*B	0,4865625	0,826581	0,59	0,5621

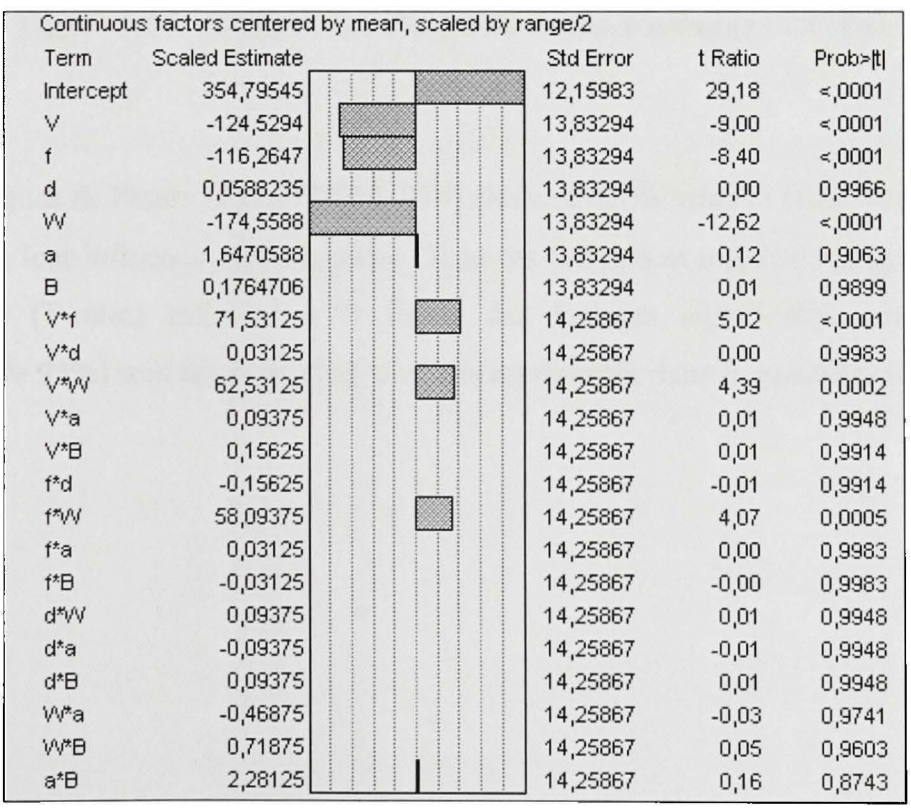
Figure 5.11 ***Diagramme Pareto de la rugosité maximale dans la direction perpendiculaire à l'avance, (Rt).***



Continuous factors centered by mean, scaled by range/2

Term	Scaled Estimate	Std Error	t Ratio	Prob>|t|
Intercept	1,5727273	0,039774	39,54	<,0001
V	0,2838235	0,045247	6,27	<,0001
f	0,5732353	0,045247	12,67	<,0001
d	0,0226471	0,045247	0,50	0,6217
W	0,3194118	0,045247	7,06	<,0001
a	0,2764706	0,045247	6,11	<,0001
B	-0,205	0,045247	-4,53	0,0002
V*f	0,106875	0,046639	2,29	0,0319
V*d	0,043125	0,046639	0,92	0,3652
V*W	0,06	0,046639	1,29	0,2117
V*a	0,0725	0,046639	1,55	0,1343
V*B	0,0025	0,046639	0,05	0,9577
f*d	0,04125	0,046639	0,88	0,3860
f*W	0,159375	0,046639	3,42	0,0025
f*a	0,139375	0,046639	2,99	0,0068
f*B	0,254375	0,046639	5,45	<,0001
d*W	0,001875	0,046639	0,04	0,9683
d*a	0,065625	0,046639	1,41	0,1734
d*B	-0,005625	0,046639	-0,12	0,9051
W*a	0,02875	0,046639	0,62	0,5439
W*B	-0,01	0,046639	-0,21	0,8322
a*B	-0,07875	0,046639	-1,69	0,1054

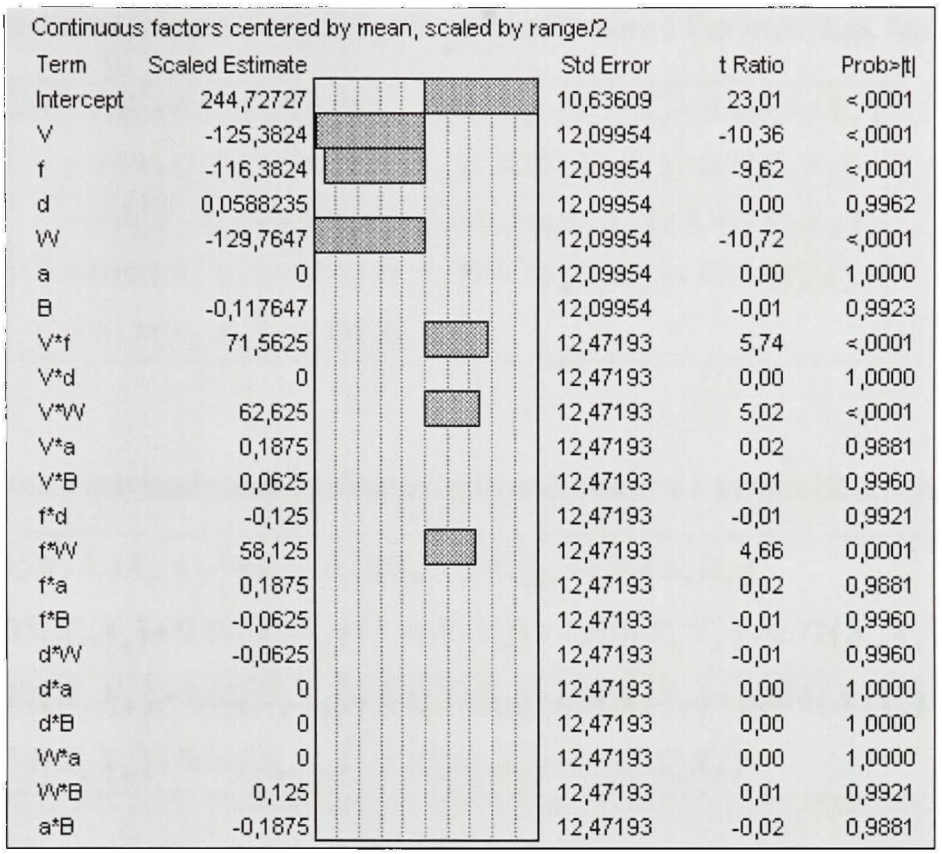
Figure 5.12 ***Diagramme Pareto de la rugosité moyenne dans la direction de l'avance, (Ra //).***



Continuous factors centered by mean, scaled by range/2

Term	Scaled Estimate	Std Error	t Ratio	Prob>|t|
Intercept	354,79545	12,15983	29,18	<,0001
V	-124,5294	13,83294	-9,00	<,0001
f	-116,2647	13,83294	-8,40	<,0001
d	0,0588235	13,83294	0,00	0,9966
W	-174,5588	13,83294	-12,62	<,0001
a	1,6470588	13,83294	0,12	0,9063
B	0,1764706	13,83294	0,01	0,9899
V*f	71,53125	14,25867	5,02	<,0001
V*d	0,03125	14,25867	0,00	0,9983
V*W	62,53125	14,25867	4,39	0,0002
V*a	0,09375	14,25867	0,01	0,9948
V*B	0,15625	14,25867	0,01	0,9914
f*d	-0,15625	14,25867	-0,01	0,9914
f*W	58,09375	14,25867	4,07	0,0005
f*a	0,03125	14,25867	0,00	0,9983
f*B	-0,03125	14,25867	-0,00	0,9983
d*W	0,09375	14,25867	0,01	0,9948
d*a	-0,09375	14,25867	-0,01	0,9948
d*B	0,09375	14,25867	0,01	0,9948
W*a	-0,46875	14,25867	-0,03	0,9741
W*B	0,71875	14,25867	0,05	0,9603
a*B	2,28125	14,25867	0,16	0,8743

Figure 5.13 ***Diagramme Pareto du temps de cycle réel (Tc).***



Continuous factors centered by mean, scaled by range/2

Term	Scaled Estimate	Std Error	t Ratio	Prob>|t|
Intercept	244,72727	10,63609	23,01	<,0001
V	-125,3824	12,09954	-10,36	<,0001
f	-116,3824	12,09954	-9,62	<,0001
d	0,0588235	12,09954	0,00	0,9962
W	-129,7647	12,09954	-10,72	<,0001
a	0	12,09954	0,00	1,0000
B	-0,117647	12,09954	-0,01	0,9923
V*f	71,5625	12,47193	5,74	<,0001
V*d	0	12,47193	0,00	1,0000
V*W	62,625	12,47193	5,02	<,0001
V*a	0,1875	12,47193	0,02	0,9881
V*B	0,0625	12,47193	0,01	0,9960
f*d	-0,125	12,47193	-0,01	0,9921
f*W	58,125	12,47193	4,66	0,0001
f*a	0,1875	12,47193	0,02	0,9881
f*B	-0,0625	12,47193	-0,01	0,9960
d*W	-0,0625	12,47193	-0,01	0,9960
d*a	0	12,47193	0,00	1,0000
d*B	0	12,47193	0,00	1,0000
W*a	0	12,47193	0,00	1,0000
W*B	0,125	12,47193	0,01	0,9921
a*B	-0,1875	12,47193	-0,02	0,9881

Figure 5.14 ***Diagramme Pareto du temps d'usinage réel (Tu).***

Les graphiques de Pareto (figure 5.10 à 5.14) classent les facteurs et leurs interactions en fonction de leur influence sur la réponse. Tous les facteurs et interactions ayant un effet standardisé (T-ratio) inférieur à la limite des facteurs significatifs (intervalle de confiance de 95%) sont négligés et ne sont pas représentés dans la modélisation.

1. La rugosité moyenne dans la direction perpendiculaire à l'avance (Ra), figure 5.9a.

$$\begin{aligned} Ra = {} & 5.13 + 0.27X_V + 0.25X_d + 3.19X_W + 0.11X_\alpha + 0.24(X_V X_f) \\ & + 0.34(X_V X_d) + 0.25(X_V X_W) + 0.32(X_V X_\alpha) + 0.1(X_V X_\beta) \\ & + 0.4(X_f X_d) + 0.14(X_f X_W) + 0.38(X_f X_\alpha) + 0.41(X_f X_\beta) \\ & + 0.24(X_d X_W) + 0.15(X_d X_\alpha) + 0.28(X_d X_\beta) + 0.31(X_W X_\alpha) \\ & + 0.13(X_W X_\beta) + 0.19(X_\alpha X_\beta) \end{aligned} \tag{5-5}$$

2. La rugosité maximale dans la direction perpendiculaire à l'avance (Rt), figure 5.9b.

$$\begin{aligned} Rt = {} & 23.29 + 1.3X_V + 1.12X_d + 12.55X_W - 0.61X_\beta + 1.72(X_V X_f) \\ & + 2.33(X_V X_d) + 0.82(X_V X_W) + 1.6(X_V X_\alpha) + 0.216(X_V X_\beta) + 0.72(X_f X_d) \\ & - 0.17(X_f X_W) + 1.02(X_f X_\alpha) + 2.1(X_f X_\beta) + 0.3(X_d X_W) + 0.816(X_d X_\alpha) \\ & + 0.34(X_d X_\beta) + 0.96(X_W X_\alpha) - 0.16(X_V X_d) + 0.48(X_\alpha X_\beta) \end{aligned} \tag{5-6}$$

3. la rugosité moyenne dans la direction de l'avance, (Ra //), figure 5.9c.

$$\begin{aligned} Ra // = {} & 1.5727 + 0.284X_V + 0.32X_W + 0.276X_\alpha - 0.2X_\beta \\ & + 0.1(X_V X_f) + 0.16(X_V X_W) + 0.14(X_f X_\alpha) + 0.25(X_f X_\beta) \end{aligned} \tag{5-7}$$

4. Le temps de cycle (Tc), figure 5.9d.

$$\begin{aligned} Tc = {} & 354.79 - 124.53X_V - 116.26X_f - 174.56X_W + 71.53(X_V X_f) \\ & + 62.53(X_V X_W) + 58.093(X_f X_W) + 2.28(X_\alpha X_\beta) \end{aligned} \tag{5-8}$$

5. Temps d'usinage réel (Tu), figure 5.9e.

$$\boxed{\begin{aligned} Tu = {} & 244.73 - 125.38X_v - 116.38X_f - 129.76X_W \\ & + 71.56\left(X_v X_f\right) + 62.62\left(X_v X_W\right) + 58.12\left(X_f X_W\right) \end{aligned}} \quad (5\text{-}9)$$

Où

- Ra, Rt, Ra//, Tc, Tu : Réponses codées du système.

- X_v, X_f, X_d, X_w, X_α, X_β : Variables indépendantes codées, représentant respectivement la vitesse d'avance de l'outil, l'avance par dent de l'outil, la profondeur de coupe, l'émersion, l'angle d'engagement de l'outil et l'angle d'inclinaison de l'outil.

Dès la phase de modélisation, nos résultats diffèrent de ceux de l'auteur [Amtout Y., 2003]. En effet dans son étude, Amtout a opté pour une modélisation logarithmique des réponses du système. De notre côté notre modèle de surface de réponses nous a conduit à des modèles arithmétiques bien corrélés donc aptes à modéliser correctement les réponses de l'étude. Nous n'avons donc pas jugé nécessaire d'avoir recours à une modélisation logarithmique.

La figure 5.15 présente les effets des principaux facteurs sur les réponses. De cette figure, il ressort que :

- le facteur le plus influent sur la rugosité moyenne dans la direction perpendiculaire à l'avance (Ra) est l'émersion (W).

- le facteur le plus influent sur la rugosité maximale dans la direction perpendiculaire à l'avance (Rt) est l'émersion (W).

- le facteur le plus influent sur la rugosité moyenne dans la direction de l'avance (Ra//) est l'avance par dent de l'outil (f).

- les facteurs les plus influents sur le temps de cycle (Tc) sont l'émersion (W) suivie de l'avance par dent de l'outil et la vitesse d'avance de l'outil.

- les facteurs les plus influents sur le temps d'usinage réel (Tu) sont l'émersion (W) suivie de l'avance par dent de l'outil et la vitesse d'avance de l'outil.

Il ressort de ce graphique que l'émersion est le facteur le plus influent de l'étude. De plus le temps de cycle total et le temps d'usinage réel semblent réagir de la même façon aux variations de facteur.

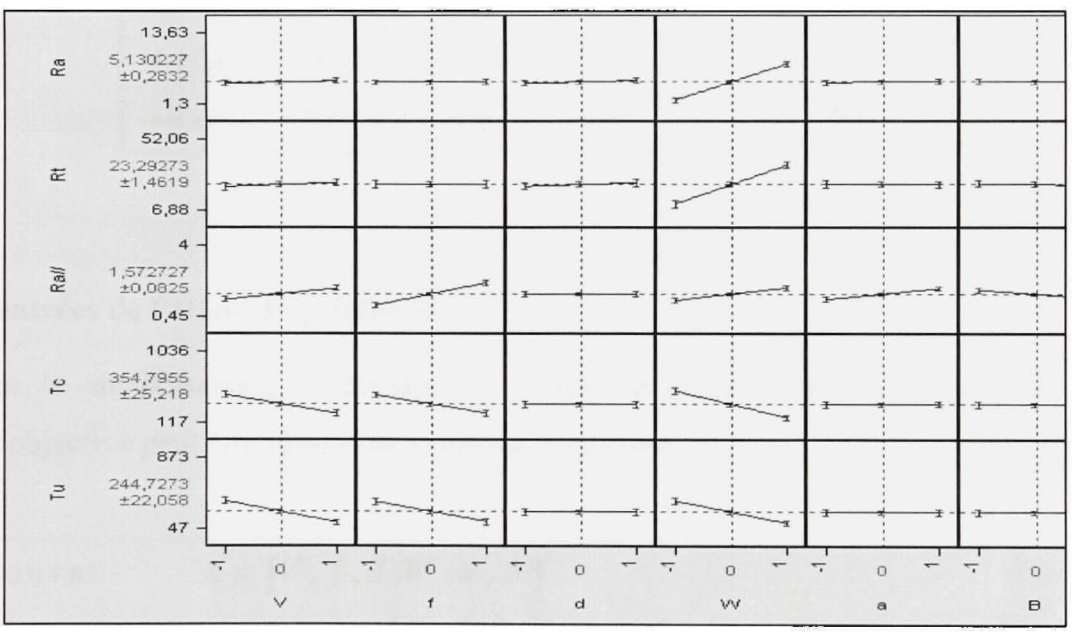


Figure 5.15 ***Graphiques des effets des principaux facteurs sur les objectifs de l'étude.***

5.3.3 Optimisation multiobjective

De l'information complémentaire sur les objectifs de l'étude est nécessaire à la poursuite de l'étude, il s'agit de la limite inférieure des réponses et de leurs pondérations.

Les limites inférieures ont été obtenues avec la RSM en optimisant chaque objectif séparément. Quant aux poids attribués à chacun des objectifs, ils correspondent à ceux utilisés par Y. Amtout [Amtout Y., 2003] dans son étude.

Tableau 5.12

Les contraintes sur les objectifs de l'étude

Objectifs	But	Limite inférieure	**Pondération**
Ra (μm)	To minimize	1.3	**0.6**
Rt (μm)	To minimize	8.8	**0.1**
Ra// (μm)	To minimize	0.45	**0.1**
Tc (sec)	To minimize	117	**0.1**
Tu (sec)	To minimize	47	**0.1**

Les entrées de l'étape d'optimisation

Après la modélisation, le problème à optimiser grâce à notre approche CACO multiobjective peut être résumé de la manière suivante :

$$
\begin{aligned}
&\text{Trouver} && x = [V, f, d, W, \alpha, \beta]^T \\
&\text{qui minimise} && f(x) = \{Ra(x), Rt(x), Ra//(x), Tc(x), Tu(x)\} \\
&\text{sujet à} && Ra(x) \geq 1.3 \\
& && Rt(x) \geq 8.8 \\
& && Ra//(x) \geq 0.45 \\
& && Tc(x) \geq 117 \\
& && Tu(x) \geq 47
\end{aligned}
\qquad (5\text{-}10)
$$

La fonction de fitness qui exprime les objectifs de l'étude est:

$$fitness = \left(\begin{array}{l} 0.6*\left(\frac{Ra_i - Ra_{\min}}{Ra_i^{Nad} - Ra_{\min}} \right)^2 + 0.1*\left(\frac{Rt_i - Rt_{\min}}{Rt_i^{Nad} - Rt_{\min}} \right)^2 + 0.1*\left(\frac{Ra//_i - Ra//_{\min}}{Ra//_i^{Nad} - Ra//_{\min}} \right)^2 \\ +0.1*\left(\frac{Tc_i - Tc_{\min}}{Tc_i^{Nad} - Tc_{\min}} \right)^2 + 0.1*\left(\frac{Tu_i - Tu_{\min}}{Tu_i^{Nad} - Tu_{\min}} \right)^2 \end{array} \right)^{\frac{1}{2}} \quad (5\text{-}11)$$

La minimisation de cette fonction de fitness nous permet d'atteindre notre objectif de minimisation de Ra, Rt, Ra//, Tc, et Tu.

En utilisant le programme que nous avons élaboré sur Matlab pour l'optimisation multi-objectif dans le domaine continu avec les colonies de fourmis nous obtenons notre résultat en 252 itérations.

Les résultats

La figure 5.17 présente la valeur de la fonction de fitness en fonction du nombre d'itérations lors du processus d'optimisation. La meilleure solution est obtenue en 260 itérations.

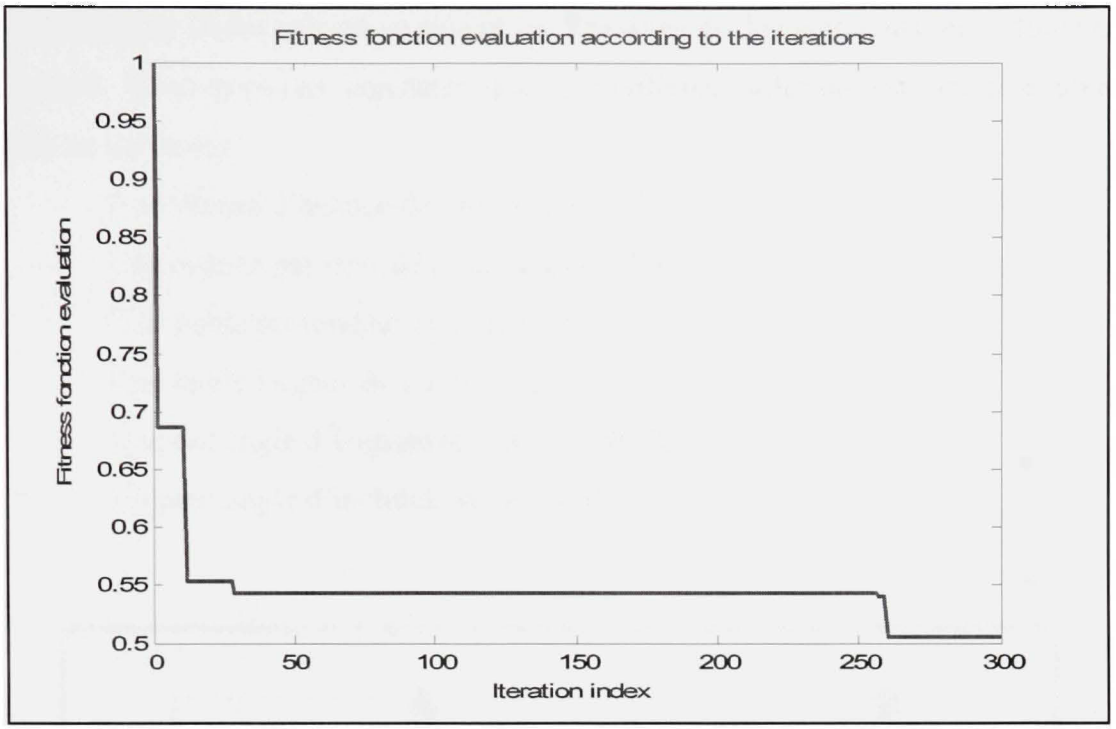


Figure 5.16 ***Valeurs de la fonction de fitness selon le nombre d'itération.***

La meilleure solution obtenue au sortir de la phase d'optimisation est :

Tableau 5.13

Les meilleures solutions (optimales)

X_V	X_f	X_d	X_W	X_α	X_β
296.2	0.298	0.35	0.549	1.98	0.7
Ra	**Rt**	**Ra//**	**Tc**	**Tu**	**fitness**
1.3	8.828	1.86	248.49	92.806	0.505

La solution respecte les contraintes du problème et présente des résultats qui minimisent les différents objectifs simultanément.

Il est intéressant d'observer où se situent les variables de décision dans leurs domaines à l'optimum. Nous pouvons constater que la meilleure solution est obtenue pour les conditions suivantes:

- Une vitesse d'avance de l'outil élevée (X_V);
- Une avance par dent de l'outil élevée (X_f);
- Une faible profondeur de coupe (X_d);
- Une faible largeur de coupe (X_W);
- Un petit angle d'engagement de l'outil (X_α);
- Un petit angle d'inclinaison de l'outil (X_β);

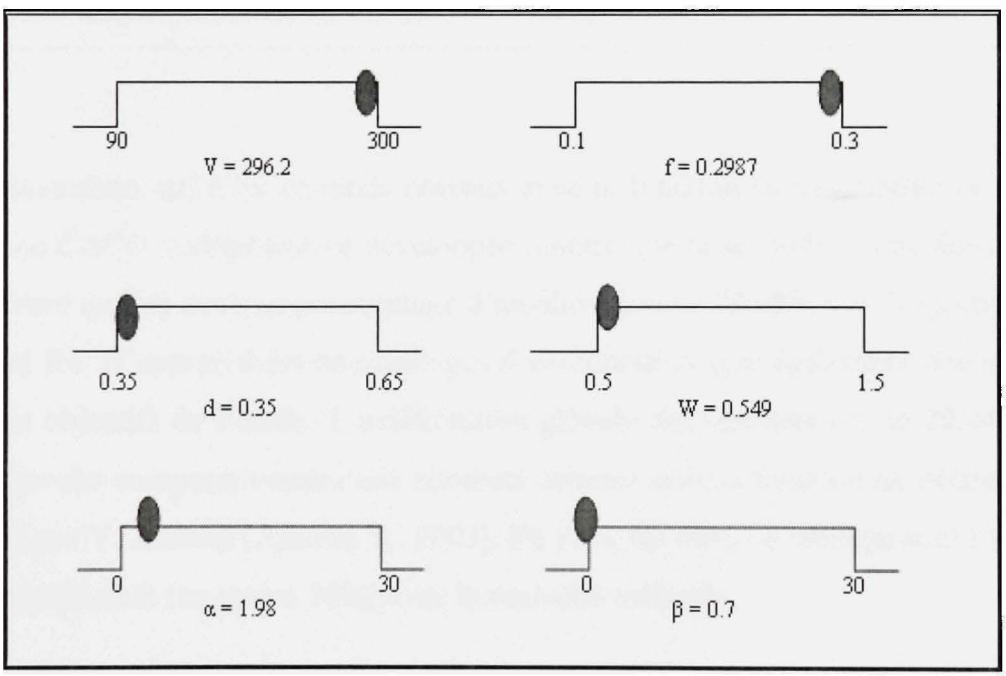


Figure 5.17 ***Emplacement de la solution optimale sur la marge de chaque variable.***

Comparaison avec les résultats des auteurs du problème

Tableau 5.14

Résultats de méthode proposée versus la fonction de désirabilité

Objectifs	Fonction de Désirabilité [19]	Méthode proposée	% d'amélioration
Ra (μm)	1,84	1,3013	29,28%
Rt (μm)	11,18	8,8279	21,04%
Ra// (μm)	1,6	1,86	-16,25%
Tc (sec)	482	248,49	48,44%
Tu (sec)	315	92,8	70,54%

Une comparaison entre les résultats obtenus avec la fonction de désirabilité et ceux de l'approche CACO multiobjective développée montre que la seconde donne des résultats de meilleure qualité avec un pourcentage d'amélioration de 29,28% sur l'objectif le plus important Ra. D'appréciables pourcentages d'amélioration sont également obtenus pour les autres objectifs de l'étude. L'amélioration globale des résultats est de 29,94% avec notre approche comparativement aux résultats obtenus avec la fonction de désirabilité et présentés par Y. Amtout [Amtout Y., 2003]. De plus, les temps d'usinage sont réduits de façon considérable (au moins 70%) avec la nouvelle méthode.

5.4 Conclusion

Ce chapitre a présenté la résolution de deux problèmes d'optimisations multiobjectives de procédé industriel en utilisant la méthode développée dans cette étude.

La première application s'effectue sur un problème bi-objectif d'optimisation d'un procédé de dessalement multi niveaux de l'eau de mer [Abdul-Wahab et Abdo, 2007]. L'utilisation de notre méthode d'optimisation a permis d'obtenir de meilleurs résultats que ceux présentés par les auteurs du problème dans leurs travaux montrant ainsi la supériorité de notre méthode sur l'utilisation de la fonction de désirabilité pour des problèmes bi-objectifs.

Il se pose alors la question de savoir si pour un plus grand nombre d'objectifs, la méthode proposée fournit également des résultats valables. Pour y répondre nous avons étudié un problème d'optimisation d'un procédé d'usinage à cinq axes résolu au préalable par Y. Amtout [Amtout Y., 2003]. Ce problème porte sur l'optimisation simultanée de cinq (5) objectifs de performance du procédé d'usinage cinq axes. Notre méthode a permis d'obtenir pour ce problème également des résultats nettement supérieurs à ceux obtenus par l'auteur dans ses travaux.

Ces études nous permettent de conclure que la méthode hybride d'optimisation que nous proposons est applicable en optimisation multiobjective et fournit des solutions de haute qualité même pour un grand nombre d'objectifs concurrents.

CONCLUSION

La contribution principale de cette étude est le développement d'une nouvelle approche d'optimisation multi-objective utilisant la méthode des surfaces de réponse (RSM) et un système de colonies de fourmis continu (CACS). En effet dans plusieurs domaines, la grande majorité des problèmes comporte plusieurs aspects à améliorer. Il est donc nécessaire de prendre en compte le caractère multicritère dans l'approche de ce type de problème. Le travail de l'ingénieur est le plus souvent de trouver des compromis, par exemple : arbitrage entre performance maximale et coût minimal de différents dispositifs. Percevoir les relations entre les objectifs aiderait l'ingénieur à mieux comprendre le problème, et ainsi, à obtenir la meilleure solution.

Dans la résolution de ce type de problème, diverses techniques traditionnelles ont été employées. Notre intérêt s'est porté sur l'algorithme des colonies de fourmis, car nous avons pu observer dans divers travaux qu'il a surpassé d'autres métaheuristiques, dans le cadre de l'optimisation multiobjective. L'ACO se révèle donc une solution innovatrice et encourageante pour l'optimisation de la conception. Car elle est complètement généralisée et indépendante du type de problème de sorte qu'elle peut être modifiée pour optimiser des problèmes complexes sujets à divers critères économiques, mécaniques, physiques, etc.., tout en respectant de nombreuses contraintes. Par contre, l'ACO a été développé pour la résolution de problèmes discrets et comme son utilisation sur des problèmes continus est toujours en cours de développement, notre étude présente donc une contribution majeure dans le développement de l'ACO continu pour les problèmes multiobjectifs.

Cette étude a démontré que l'approche que nous présentons, ici, permet d'optimiser efficacement des problèmes dont les fonctions sont complexes et ont des coûts d'évaluation élevés ou des problèmes dont la modélisation est inconnue. Cette approche hybride réduit les coûts du procédé de MOO et a prouvé sur des applications

numériques, qu'elle est puissante et peut fournir des solutions de qualité pour divers problèmes de MOO.

Dans beaucoup de cas, le problème à optimiser est un système complexe, où interagissent de multiples dispositifs ou sous-systèmes. Ils doivent être pris en compte dans le processus d'optimisation, si le but est bien de tester toutes les possibilités. On peut donner comme exemple la fabrication d'une aube de turbine Francis. La conception de cette aube peut faire appel à différentes méthodes, numériques et/ou analytiques. Ce qui peut éventuellement rendre la simulation trop coûteuse en temps de calcul et en coût d'expérimentation. La croissance des besoins de l'ingénieur alliée à l'évolution des possibilités d'analyse qui lui sont offertes (amélioration constante des outils comme les méthodes d'éléments finis, par exemple) explique en grande partie le développement de notre approche d'optimisation.

Une étude de sensibilité sur les résultats obtenus serait très utile pour réduire l'impact de la pondération des objectifs effectuée par le décideur au moment de la modélisation du problème. En effet de cette manière, il est possible de déterminer les pondérations adéquates des objectifs pour obtenir les meilleures solutions en éliminant l'erreur que pourrait entraîner un mauvais jugement du décideur.

Ce mémoire se termine ainsi avec la présentation d'une voie pour des suites possibles de cette recherche. Nous avons la conviction que les objectifs fixés au départ ont été traités. Nous espérons que nos travaux seront utiles aux concepteurs en entreprise et à tous les chercheurs qui travaillent dans le domaine de l'optimisation.

ANNEXE I

ARTICLE DE CONFERENCE APIEMS & CIIE - DÉCEMBRE 2007 - CHINE

The Hybrid Simulation Approach Based on RSM and Ants Colony System for Multiobjective Optimization Problem

Félicia Etsinda Mpiga, Thien-My Dao*, and Victor Songmene

Mechanical/Manufacturing Engineering Department
École de technologie supérieure (ÉTS)
1100, Notre-Dame Street West
Montreal (Québec) H3C 1K3, Canada
Corresponding author's e-mail: thien-my.dao@etsmtl.ca*

Abstract: Today's industrial environment is very competitive, and requires shorter manufacturing lead times, lower costs, and higher quality products. These requirements produce complex problems characterized by multiple objectives as well as complex design objective and constraint relations. From high number of variables, multiobjective optimization (MOO) becomes complex and very expansive to obtain the solution, particularly when the computational time is taken into account. The computational time could be reduced by using response surface methodology (RSM) into the optimization process. However, RSM needs a lot of independent variables to construct a proper response surface.

This paper proposes a new approach for a MOO for continuous problems. The approach uses a combination of RSM modeling, and continuous Ant colony algorithm as the search method for the optimization phase. The proposed hybrid model is applied for optimizing the surface finish of mold steels machining using different tool orientation and cutting parameters. Is shows a great improvement over the classical desirability function. The hybrid approach is demonstrated that it is a powerful technique for solution research of the optimization problem.

Keywords: RSM, Continuous ant colony algorithm (CACO), MOO, Simulation.

1. INTRODUCTION

Numerical optimization is one of the searching methods often used to determine the best solution of engineering problem. Several algorithms applications have been developed for single-objective. However, given the complexity and the multiple objectives of the engineering problem, the researchers are now more on optimization algorithms for multiple objectives and multiple constraints. MOO problems are however difficult to solve. To overcome these difficulties, the response surface method (RSM) is often employed ([1], [2], [3]) to replace a complex model by an approximation based on calculated results on certain points in the search space.

When an adequate model is obtained, it then becomes necessary to consider the optimization step. MOO problems are know difficult to solve, because of conflicts among the objectives, and appropriate tradeoffs have to be made in finding the optimal solutions. Existing literature shows that desirability or metaheuristic functions are usually used, the most common being the genetic algorithm (GA). Sun and Lee [4], present an approach which associates the RSM and GA with the optimal aerodynamic design of a helicopter rotor blade. The Ants Colony System (ACS), a metaheuristic inspired from the studies of Ants behaviour is considered as a multi-agent approach for resolving combinative optimization problem. By

application, the ACS metaheuristic was proved to be a powerful technique for optimization problem ([5, 6]).

In this paper, a hybrid approach combined the RSM and an extension of the ACS in continuous fields is proposed for solution of the continuous MOO problem. The RSM modelling is used to determine the objective functions and the ACS is then applied to search the (optimum) solution.

2. MODELLING WITH RSM

RSM is a collection of statistical and mathematical techniques used to develop, improve and optimize processes [1]. Furthermore, it is very used for optimization in engineering and manufacturing application. The objective of RSM is to evaluate a response, i.e., the objective physical quantities, which are influenced by several problem variables. In RSM, a continuous answer Y is connected with continuous and controlled factors X_1, X_2... X_k, using a linear regression model which can be written as [1]:

$$y = f_\beta(X_1, X_2, \ldots, X_p) + \varepsilon \tag{1}$$

Since the response surface is described by a polynomial representation, it is possible to reduce the optimization resolution process time by assessing the objectives with their models rather than using more complex empirical models. Although the specific form of response factor f_β is unknown, experience shows that it can be significantly approximated using a polynomial.

In the case of two factors, the linear regression model is one of the simplest available, and corresponds to a first-degree model with interaction, and which has the following form:

$$y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_{12} X_1 X_2 + \varepsilon \tag{2}$$

Whenever this model is unable to describe the experimental reality effectively, it is common practice to use a second-degree model, which includes the quadratic effects of the factors involved:

$$y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_{12} X_1 X_2 + \beta_{11} X_1^2 + \beta_{22} X_2^2 + \varepsilon \tag{3}$$

where y is the response (study objective, for example, the total manufacturing cost); ϵ is the estimate of the error; X_1 and X_2 are influential factors of the coded response (e.g., design variables).

The unknown parameters of these mathematical models (Eq. 2,3), βi values, are estimated through the least-squares technique, and the adjustment quality of the model is assessed using traditional multiple linear regression tools.

Ideally, the number of experiments carried out during the application of RSM, should be as small as possible, in order to reduce data-processing requirements. Properly selecting the points to be used for the simulation will allow a reduction of the variance of the coefficients of the mathematical model, which will in turn ensure that the response surfaces obtained are more reliable.

Once the mathematical models are obtained, their adequateness need to be evaluated. This can be done using the statistical coefficient of determination R^2, which must be as close as possible to 1 ($0<R^2<1$).

Once this stage is completed, the continuous MOO problem can be formulated. Generally, such problems are of the following form:

$$
\begin{aligned}
&\text{Find} && x=[x_1,x_2,x_3,...,x_n]^T \\
&\text{Which minimize} && f(x)=\{f_1(x),f_2(x),f_3(x),...,f_n(x)\} \\
&\text{Subject to} && g_j(x)\le 0 \quad \text{for } j=1,m \\
& && x_i^L \le x_i \le x_i^U \quad \text{for } i=1,n
\end{aligned}
\tag{4}
$$

Some resolution methods exist for this kind of problem, such as the desirability function and the genetic GA used by some authors, such as Sun H.et al. [4] and Abdul-Wahad S.A. et al. [7]. Literature shows that for most of problems, the Ant colony approach produces better results in terms of quality solutions and resolution speed. This allowed us to begin this research with the resolution of the continuous MOO problem [6], [7]. To optimize this problem, we explored the Ant colony system (ACS).

3. ANT COLONY ALGORITHM APPROACH

The Ant colony optimization (ACO) metaheuristic, called the ant system [8], was inspired by studies of the behaviour of ants [11], as a multi-agent approach for resolving combinative optimization problems such as the Travelling Salesman Problems (TSP).

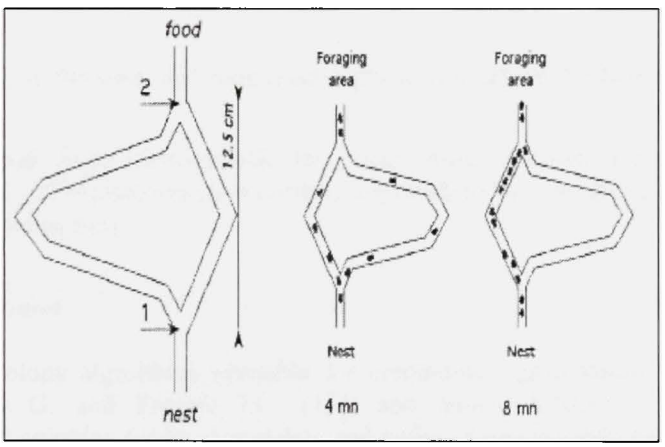


Figure 1. Experimental setup and drawings of the selection of short branches by a colony of Linephitema humile, 4 and 8 min after the bridge was placed (By Dorigo, M. [17])

Ants communicate among themselves through the "pheromone", a substance they deposit on the ground in variable amounts as they move about. It has been observed that the more ants use a particular path, the more pheromone is deposited on that path, and the more attractive it becomes to other ants seeking food. If an obstacle is suddenly placed on an established path leading to a food source, ants will initially go randomly right or left, but those choosing the side that is in fact shorter will reach the food more quickly, and will make the return journey more often. The pheromone concentration on the shorter path will therefore be more strongly reinforced, and it will eventually become the new preferred route for the stream of ants; however, it must also be borne in mind that the pheromone deposited along the way does evaporate. Works by Dorigo M. et al. [9] provide detailed information on the operation of the algorithm and on the determination of the values of the various parameters.

In optimization process, the ACO has been used very sparingly, and has been focused primarily on single-objective problems. For multiobjective problems, the ACO has hardly been used at all [10], and when

used, it has been mainly on combinatorial optimization problems. The importance of this work therefore lies in its attempt to adapt continuous ant colonies to multiobjective problems.

4. PROPOSED HYBRID OPTIMIZATION APPROACH

The objective of this paper is to propose a new MOO algorithm. A Multiobjective optimization problem (MOOP) involves a number of objective functions to be optimized simultaneously. The ACO requires an objective function which can be quickly assessed. The objective functions are modelled by RSM, and the ACO as the optimization research method. Reducing the resolution time in the optimization process requires a reduction of the precision of the assessment of objective functions, then an approximate modeling of the objectives will be used.

Each objective f_i is expressed according to the input variables x_i which influence its value. The MOO model obtained with RSM is:

$$\begin{aligned} &\text{minimize} \quad f(x) = \{f_1(x), f_2(x), f_3(x), ..., f_n(x)\} \\ &\text{subject to} \quad x_i^L \le x_i \le x_i^U \quad \text{for } i = 1, n \end{aligned} \tag{5}$$

After the problem model is obtained, the optimization phase determines the best compromise solution for the various objectives.

The steps for the general ACO metaheuristic for compromise solutions for combinatorial problems presented by Gagné et al. [11] constitute an interesting approach to be considered in our resolution process for developing the fitness function.

4.1 Continuous ant colonies

There are several ant colony algorithms available for continuous optimization. The first of which was developed by Belchev G. and Parmee I.C. [12], and named CACO (Continuous Ant Colony Optimization), using ant colonies for local searches and calling upon revolutionary algorithms for global searches. Ling C. et al. [13] present an unspecified hybrid algorithm whose main premise is to consider the differences between two individuals on each dimension as many parts of a path on which the pheromones are deposited. The evolution of the individuals dealt with mutation and crossing-over operators. This method thus tries to reproduce the construction mechanism of the solution component by component.

Monmarché N. [14] developed the API algorithm which takes the primitive ant behaviour of the species *Pachycondyla Apicalis*, and which does not use indirect communication by tracks of pheromone. In this method, it is necessary to start by positioning a nest randomly on the research space, after which ants are distributed randomly over it. These ants explore their "hunting site" locally by evaluating several points within a given perimeter.

Socha K. [15] presents the ACO algorithm for continuous optimization which tries to maintain the iterative construction solutions for continuous variables. He to consider that the components of all solutions are formed by the various optimized variables. Moreover, before considering the algorithm from the ant's point of view, he chooses to operate at the colony level, with the ants being simply points to be evaluated.

Pourtakdoust S.H. and Nobahari H. [16] developed the CACS (Continuous Ant Colony System) algorithm, which is very similar to that of Socha K. [15]. Indeed, in CACS, as is the case with ACO, for continuous optimization, the core of the algorithm consists in evolving a probability distribution which for CACS, is normal.

4.2. Proposed hybrid algorithm

Our resolution algorithm, consisting of two levels, can be summarized as follows (See also fig. 2):

Step 1: System configuration

Determine the objectives of this study, the constraints and the variables which can influence these objectives.

Evaluate the field of application of these variables.

Step 2: RSM

Set up an experimental design, carry out tests, and model the various objectives according to influential parameters.

Step 3: Seek ideal point

Determine distinct optimum for each study objective, using RSM or problem information's.

Step 4: Optimization function formulation

a) State user preferences (weighting of the objectives).

b) The various objectives are expressed in a single function: the fitness function. Gagné et al. [11] describe well. It acts as an equation which for each objective, expresses the standard and balanced distance at the ideal point F* of an unspecified solution k, whose various objectives are given by F^k. This function makes it possible to standardize objectives in order to reduce the adverse effects obtained from the various measuring units, as well as the extent of the field of the variables, in order to not skew the fitness function [11]:

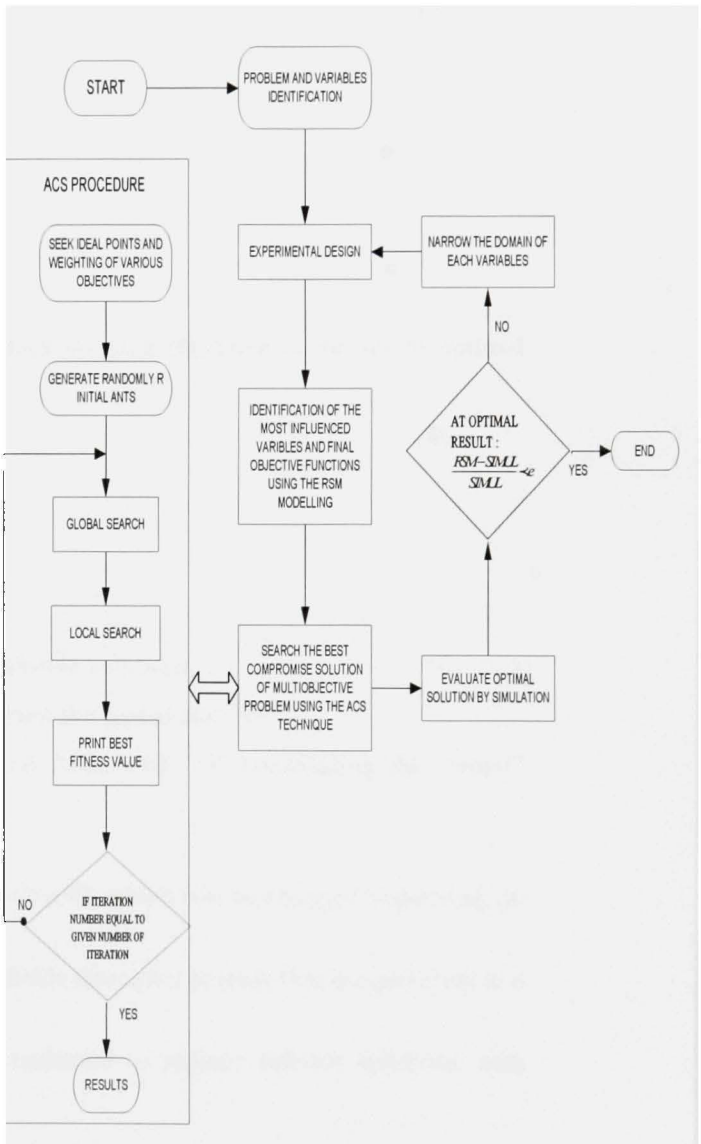


Figure 2. Flow chart for hybrid optimization approach

$$fitness = \left(\sum_{i=1}^{Z} p_i \left(\frac{F_i^k - F_i^*}{F_i^{nad} - F_i^*} \right)^2 \right)^{\frac{1}{2}} \quad (6)$$

with:

F* is a solution vector corresponding to the ideal point of each separate objective, and expressed by:

$$F^* = \{F_1^*, F_2^*, \ldots, F_Z^*\} \quad (7)$$

with $F_i^* = \min_{x \in S} f_i(x)$.

F* generally corresponds to an unrealizable solution.

S is the space of acceptable search

F^{nad} is the Nadir point, which represents the maximum values for each objective in the set of optimal Pareto solutions [11]:

$$F^{nad} = \{F_1^{nad}, F_2^{nad}, \ldots, F_z^{nad}\} \quad (8)$$

where $F_i^{nad} = \max_{x \in S} f_i(x)$.

Step 5: Determine compromise solution

a) Randomly generate R initial ants corresponding to feasible solutions.

b) Use an proportion of the higher values of R to constitute the global ants "G".

c) Apply a global search to a percentage of the initial Ants, with "G" constituting the "worst" solutions available.

The percentage of global Ants is an important parameter of CACO, which can be changed depending on the problem studied.

In the CACO simulation, a global search is conducted in all fields through a process that is equivalent to a GA crossover and mutation.

Crossover or random walk: The crossover operation is conducted to replace inferior solutions with superior ones, with the crossover probability (CP).

Mutation: Replace solutions are further improvement by mutation. The mutation step size is reduced or increased those solutions as per Eq. 9. [18]

$$\Delta = R\left(1 - r^{(1-T)^b}\right) \quad (9)$$

Where r is a random number from 0 to 1; R is the maximum step size; T is the ratio of the current iteration number and that of the total number of iterations; b is a positive parameter controlling the degree of nonlinearity.

Trail diffusion: Two parents are randomly selected from the parent population space. The elements of the child's vector can be any one of the following: [13]

1. An element from the first parent.

2. An element from the second parent.

3.The child is a combination of the parents (Eq.10)

$$X(child) = (\alpha) X_{i(parent1)} + (1-\alpha) X_{i(parent2)} \tag{10}$$

where α is a uniform random number ranging from [0-1].

The probability of selecting one of the three options depends on the mutation probability.

Pheromone evaporation: In real there is a process of food exhaustion, which causes the ant to loose interest. This feature is incorporated in the ant algorithm by reducing the trail pheromone value of all regions by a certain value after each iteration step. This evaporation of the pheromone strength is incorporated as per:

$$\tau_i (t+1) = \rho . \tau_i (t) \tag{11}$$

where ρ is the evaporation rate and τ_i (t) is the pheromone trail of region at time *t*.

d) Send local Ants *L* in the various *R* areas. Once the global search is completed, the zones to which you send the local Ants are defined and a local search can begin. The probability of choosing a global solution *i* is given by [13]:

$$P_i (t) = \frac{\tau_i (i)}{\sum_k \tau_k (t)} \tag{12}$$

where *i* is the solution index and τ_i (t) is the pheromone trail of region at time *t*.

After choosing its destination, the ant proceeds across a short distance. The search direction remains the same from one local solution to the next as long as there is improvement in the fitness function. If there is no improvement, the Ant reorients itself randomly to another direction. If an improvement in the fitness function is obtained in the preceding procedure, the position vector of the area is updated. The quantity of pheromone deposited is proportional to the improvement of the fitness function. The size of the ant movement in a local search depends on the current area age. This age increase when a higher fitness value is obtained. The search ray is maximum for age zero, and minimal for the maximum age, with the variation in radius set to follow a linear relationship. [13]

Evaluate the fitness function for each Ant obtained, and continue the iterative process, beginning with a global search until stop conditions are observed.

Step 6: Evaluate the best solutions (quasi optimal) by simulation.

Step 7: Evaluate the stop criterion $\frac{RSM - SIMUL}{SIMUL} \prec e$ with SIMUL being the simulation result.

In the optimization problem the number of variables is very often equal to or higher than 3, and each one of them has a broad field of variation. Consequently, in our resolution process, it is necessaries that search field for each design variable be gradually narrowed for as long as the stop criterion has not been encountered.

5. APPLICATION: OPTIMIZATION OF MACHINING PROCESS

In order to illustrate the performances and the capability of the proposed approach, a numerical application concerning the behaviours and the influence of cutting parameters on machining performances when milling P20 mould steel was studied. The problem was originally studied by [19] using RSM and desirability functions.

5.1 Problem definition

Industries of numerical control machining of the moulds and the aeronautical parts are continuously seeking for the improvement of the productivity and the part quality. The machining process, including the finishing operations represents most of the manufacturing costs of the moulds. Significant savings in machining time and quality of the finished parts can obtain the optimization of the machining parameters. A more extensive description of the machining problem considered in this work can be found in [19].

In this study, the work consists to study and analyse the behaviours and influence of the parameters of cut, that is to say, the advance by tooth, the cutting speed, the cutting width, the depth of axial cutting, the orientations of tool and the tool path strategies on the performances of the machining in the 5 axes of the steel mould of the P20 type. Five performance objectives are considered: the minimization of the average roughness in the direction perpendicular to feed (Ra), the maximum roughness in the direction perpendicular to the feed (Rt), the average roughness in the direction of the feed (Ra//), the cycle time (Tc), the metal cutting time of machining (Tu).

The fields of variables which influence these objectives are presented in Table 1 (**Step 1**). They include:

Table 1. Fields of variables [19]

Parameters name	**Nomenclature**	**Low level (-1)**	**High level (+1)**
Cutting speed (m/min)	V	90	300
Feed per tooth (mm)	f	0.1	0.3
Axial depth of cut (mm)	d	0.35	0.65
Cutting width (mm)	W	0.5	1.5
Angle of engagement(°)	α	0	30
Angle of inclination (°)	β	0	30

5.2 Modeling with RSM

To express our objectives according to decision variables, the RSM is used to obtain the models of different performance objective functions (**steps 2 and 3**). To that end, the experiments carried out by Amtout Y. [19], had been used. It resorted that a Box Wilson plan using RSM, Face-Centered Central Composite Design (FCCD), is a best pan for this problem. A total of 46 tests were necessary in order to isolate the effects of the six (6) independent variables of the study. The design variables are coded at three levels: low (-1), middle (0) and high (+1). Matrix of experiment is shown in the table 2.

The experimental design provides us with a linear regression model coded for each response in this study.

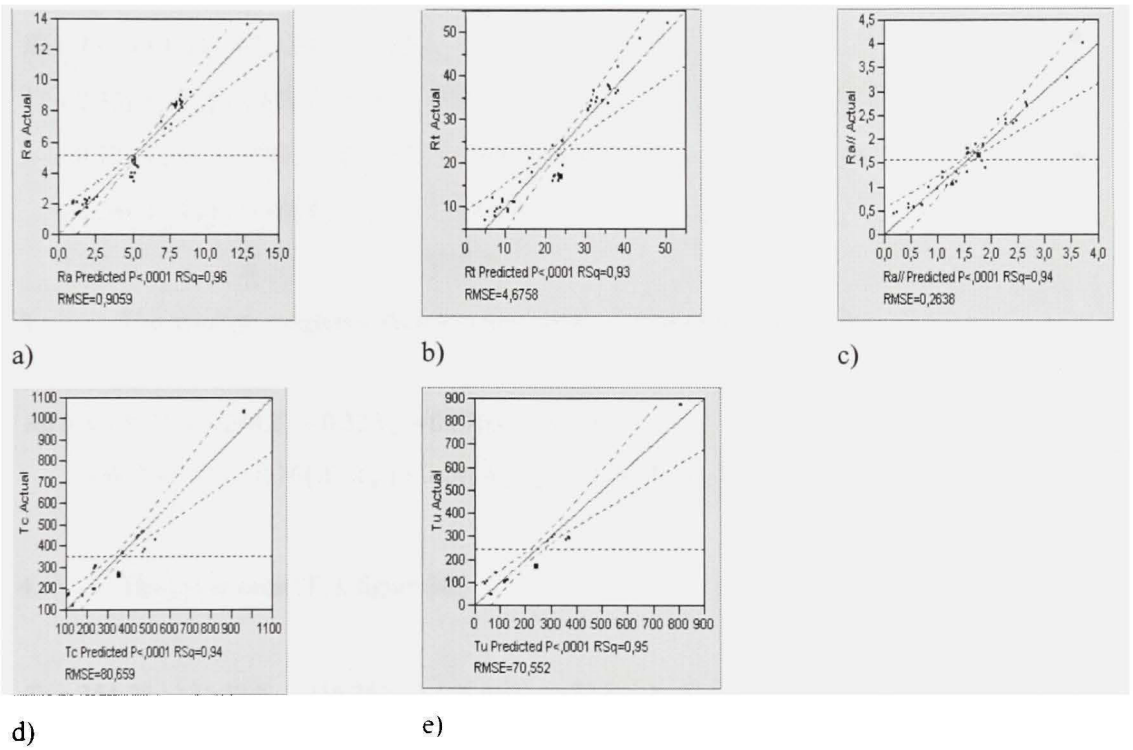


Figure 3. Modeling with RSM

It is possible also to predict the outcome with a simple equation that uses the overall average modified up or down depending on the level for each variable. This equation is called the coded equation. Final equations in terms of coded factors for our objectives are as follow:

1. The average roughness (Ra) in the direction perpendicular to feed, figure 3a.

$$
\begin{aligned}
Ra = {} & 5.13 + 0.27X_{V} + 0.25X_{d} + 3.19X_{W} + 0.11X_{\alpha} + 0.24(X_{V}X_{f}) \\
& + 0.34(X_{V}X_{d}) + 0.25(X_{V}X_{W}) + 0.32(X_{V}X_{\alpha}) + 0.1(X_{V}X_{\beta}) \\
& + 0.4(X_{f}X_{d}) + 0.14(X_{f}X_{W}) + 0.38(X_{f}X_{\alpha}) + 0.41(X_{f}X_{\beta}) \\
& + 0.24(X_{d}X_{W}) + 0.15(X_{d}X_{\alpha}) + 0.28(X_{d}X_{\beta}) + 0.31(X_{W}X_{\alpha}) \\
& + 0.13(X_{W}X_{\beta}) + 0.19(X_{\alpha}X_{\beta})
\end{aligned}
\tag{13}
$$

2. The maximum roughness (Rt) in the direction perpendicular to the feed, figure 3b.

$$
\begin{aligned}
Rt = {} & 23.29 + 1.3X_{V} + 1.12X_{d} + 12.55X_{W} - 0.61X_{\beta} + 1.72(X_{V}X_{f}) \\
& + 2.33(X_{V}X_{d}) + 0.82(X_{V}X_{W}) + 1.6(X_{V}X_{\alpha}) + 0.216(X_{V}X_{\beta}) + 0.72(X_{f}X_{d}) \\
& - 0.17(X_{f}X_{W}) + 1.02(X_{f}X_{\alpha}) + 2.1(X_{f}X_{\beta}) + 0.3(X_{d}X_{W}) + 0.816(X_{d}X_{\alpha}) \\
& + 0.34(X_{d}X_{\beta}) + 0.96(X_{W}X_{\alpha}) - 0.16(X_{V}X_{d}) + 0.48(X_{\alpha}X_{\beta})
\end{aligned}
\tag{14}
$$

3. The average roughness (Ra//) in the direction of the feed, figure 3c.

$$
\begin{aligned}
Ra// = {} & 1.5727 + 0.284X_{V} + 0.32X_{W} + 0.276X_{\alpha} - 0.2X_{\beta} \\
& + 0.1(X_{V}X_{f}) + 0.16(X_{V}X_{W}) + 0.14(X_{f}X_{\alpha}) + 0.25(X_{f}X_{\beta})
\end{aligned}
\tag{15}
$$

4. The cycle time (Tc), figure 3d.

$$
\begin{aligned}
Tc = {} & 354.79 - 124.53X_{V} - 116.26X_{f} - 174.56X_{W} + 71.53(X_{V}X_{f}) \\
& + 62.53(X_{V}X_{W}) + 58.093(X_{f}X_{W}) + 2.28(X_{\alpha}X_{\beta})
\end{aligned}
\tag{16}
$$

5. The metal cutting time of machining (Tu), figure 3e.

$$
\begin{aligned}
Tu = {} & 244.73 - 125.38X_{V} - 116.38X_{f} - 129.76X_{W} \\
& + 71.56(X_{V}X_{f}) + 62.62(X_{V}X_{W}) + 58.12(X_{f}X_{W})
\end{aligned}
\tag{17}
$$

We also note that in our experimental design, the variable «W» is the most influential on our objectives. Additional information on optimization, as well as the goals of the study, are summarized in table 3. The weight used corresponding to those Y. Amtout [19].

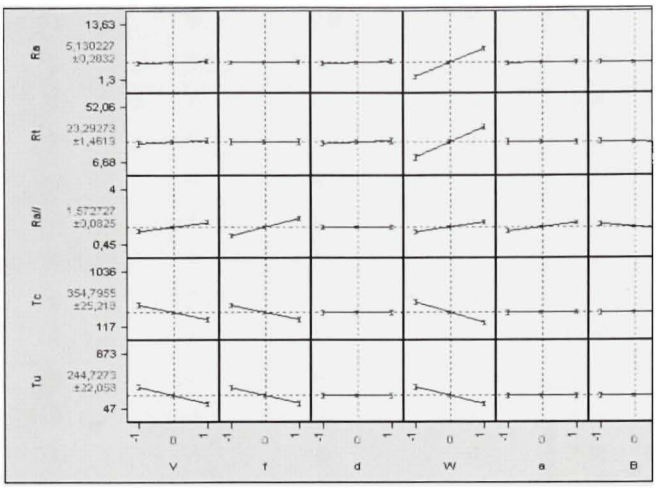


Figure 4. Impact of the decision variables on study objectives

Table 2. Matrix of experiments and experimental results of the responses [19]

	Variables						Responses				
	X_v	X_f	X_d	X_w	X_α	X_β					
Test no.	Speed (m min)	Advance (mm/tooth)	Depth of axial cutting (mm)	Cutting width (mm)	Angle of engagement (degrees)	Angle of inclination (degrees)	Ra (μm)	Rt (μm)	Ra// (μm)	Tc (sec)	Tu (sec)
1	0	0	0	0	0	-1	4,72	17	1,65	263	170
2	-1	-1	-1	-1	1	1	2,06	12	0,46	1036	871
3	1	1	-1	-1	-1	-1	2,07	8,93	1,3	304	139
4	-1	1	1	-1	-1	-1	1,46	7,84	1	469	306
5	0	0	0	-1	0	0	1,7	9	1,04	432	293
6	-1	1	1	1	-1	1	8,78	32,96	1,87	172	104
7	-1	-1	-1	1	1	-1	8	36	1,81	365	296
8	1	0	0	0	0	0	4,47	24,87	1,9	198	106
9	0	0	0	0	0	0	3,92	19,98	1,46	261	170
10	-1	-1	1	1	1	1	7,17	32,56	0,63	369	297
11	0	-1	0	0	0	0	3,44	16	0,95	374	282
12	1	-1	-1	1	1	1	8,42	34,4	0,96	168	95
13	1	-1	-1	-1	1	-1	1,97	9,6	1,68	440	279
14	0	0	0	0	0	1	4	17,4	1,1	272	170
15	0	0	0	1	0	0	8,42	33,85	1,4	164	99
16	-1	-1	-1	1	-1	1	8,53	37,82	0,57	365	297
17	-1	1	1	1	1	-1	8,58	32,24	2,7	173	104
18	1	1	1	-1	-1	1	3,4	21,2	1,57	296	140
19	-1	1	-1	1	-1	-1	7,34	36,56	1,89	173	104
20	1	-1	-1	1	-1	-1	9,02	37	1,7	165	95
21	1	1	-1	-1	1	1	2,21	11,22	2,31	307	140

22	0	0	-1	0	0	0	3,71	17	1,31	262	170
23	-1	-1	1	-1	-1	1	2,25	11,14	0,45	1026	873
24	1	1	1	1	-1	-1	8,24	36,5	2,37	118	47
25	0	1	0	0	0	0	4,49	21,74	2,4	200	109
26	-1	-1	-1	-1	-1	-1	5	24	1,02	1035	873
27	-1	1	-1	1	1	1	8,25	34,12	2,32	176	104
28	1	-1	1	1	-1	1	8,68	34,32	0,62	165	95
29	0	0	0	0	-1	0	3,68	17,6	1,04	262	170
30	1	1	1	1	1	1	13,63	52,06	4	121	47
31	1	-1	-1	-1	-1	1	1,36	6,88	0,55	434	279
32	-1	1	-1	-1	-1	1	1,94	11	1,2	460	306
33	1	-1	1	-1	-1	-1	2,38	15,8	1,32	443	279
34	1	1	-1	1	1	-1	8,55	37,67	3,29	118	48
35	1	-1	1	-1	1	1	1,86	9,55	0,64	446	279
36	-1	-1	1	-1	1	-1	1,99	10	1,05	1031	873
37	-1	-1	1	1	-1	-1	8,4	42,19	1,18	366	297
38	-1	0	0	0	0	0	4	16	1,07	387	297
39	0	0	0	0	0	0	3,83	19,44	1,34	262	170
40	-1	1	1	-1	1	1	2,3	11,87	1,7	470	305
41	-1	1	-1	-1	1	-1	1,3	8,8	1,8	466	307
42	1	1	-1	1	-1	1	6,82	35,01	3	117	47
43	1	-1	1	1	1	-1	9,24	48,39	2,4	165	95
44	0	0	0	0	1	0	5,16	17,22	1,54	275	170
45	1	1	1	-1	1	-1	2,4	18	2,76	300	141

Table 3. Constraints on objectives of study

Objectives	Goal	Lower limit	**Weight**
Ra (μm)	To minimize	1.3	**0.6**
Rt (μm)	To minimize	8.8	**0.1**
Ra// (μm)	To minimize	0.45	**0.1**
Tc (sec)	To minimize	117	**0.1**
Tu (sec)	**To minimize**	**47**	**0.1**

5.3 Multiobjective optimization

Let us optimize the following problem with our CACO multiobjective approach:

$$\text{Find} \quad x = [V, f, d, W, \alpha, \beta]^T$$

$$\text{which minimize} \quad f(x) = \{Ra(x), Rt(x), Ra//(x), Tc(x), Tu(x)\}$$

$$\text{subject to} \quad Ra(x) \geq 1.3$$

$$Rt(x) \geq 8.8$$

$$Ra//(x) \geq 0.45$$

$$Tc(x) \geq 117$$

$$Tu(x) \geq 47$$

The fitness function used, obtained by the Compromise Programming (CP) [11] method, allows the search for solutions approaching the ideal point for each objective (**Step 4**):

$$fitness = \left(\begin{array}{l} 0.6*\left(\frac{Ra_i - Ra_{\min}}{Ra_i^{Nad} - Ra_{\min}} \right)^2 + 0.1*\left(\frac{Rt_i - Rt_{\min}}{Rt_i^{Nad} - Rt_{\min}} \right)^2 + 0.1*\left(\frac{Ra//_i - Ra//_{\min}}{Ra//_i^{Nad} - Ra//_{\min}} \right)^2 \\ +0.1*\left(\frac{Tc_i - Tc_{\min}}{Tc_i^{Nad} - Tc_{\min}} \right)^2 + 0.1*\left(\frac{Tu_i - Tu_{\min}}{Tu_i^{Nad} - Tu_{\min}} \right)^2 \end{array} \right)^{\frac{1}{2}}$$

The minimization of the fitness function enables us to reach our Ra minimization, Rt minimization, Ra// minimization, Tc minimization and Tu minimization goals.

Using MatLab software (**step 5**), Figure 5 enables us to state that the best solution is obtained after about 255 iterations. The staircase shape of our curve is explained by the fact that a memory effect was used in our program. Thus when iteration produces a worse solution than the preceding one, the solution of the preceding iteration is retained.

Table 4 presents the results obtained by the optimization process. The solutions respect the constraints of the problem, and produce results which minimize objectives of study while remaining in the field of each decisional variable.

It is interesting to observe the values of the decision variables in their respective fields. We can see that the best solutions are obtained under the following conditions:

- High cutting speed (V)
- High advance by tooth (f)
- Low depth of axial cutting (d)
- Low cutting width (W)
- Low angle of engagement (α)

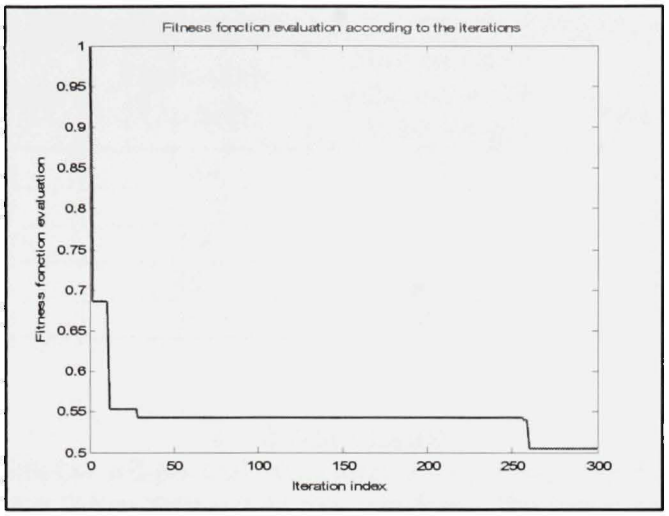


Figure 5. Values of the function of fitness according to the iteration count

Table 4. Optimal (best) solutions

V	f	d	W	α	β	Ra	Rt	Ra//	Tc	Tu	fitness
296.2	0.298	0.35	0.549	1.98	0.7	1.3	8.828	1.86	248.49	92.806	0.505

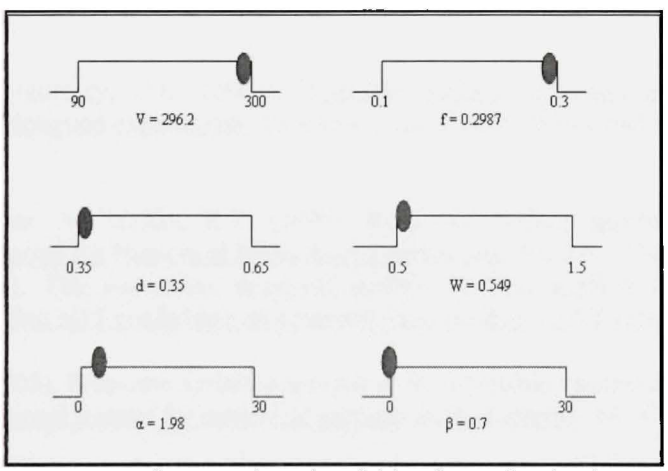


Figure 6. Value margins of variables for optimal solution

5.4 Comparison with authors' results

As shown in the table 5, the most important objective Ra is improved of 29,28% and for others objectives, excepted Ra//, there are also a great improvement. The global improvement is 28,25% compared with the results obtained with the desirability function as discussed in [19]

Table 5. Results of multiobjective CACO versus the desirability function

Objective	Desirability (Author)	Multiobjective optimization (Our méthodology)	% of amelioration
Ra	1.84	1.3013	29.28%
Rt	11.18	8.8279	21.04%
Ra//	1.6	1.86	-16.25%
Tc	482	248.49	48.44%
Tu	315	92.8	70.54%

6. CONCLUSION

In this study, a novel efficient and practical MOO methodology using Response Surface Methodology (RSM) and Continuous Ant Colony System (CACS) is introduced. This hybrid approach allows to solving challenging optimization problems that are defined in terms of complex functions that are expensive to evaluate, or when the model functions are unknown. Such problems arise frequently in engineering applications and process control problems.

We have described a methodology based on using numerical simulation, RSM and ACO process for resolution of this type of problem. This hybrid approach reduces cost of MOO process and as demonstrated in the numerical application, it is powerful and potentially able to furnish a high quality solution for the multiple objectives optimization problem.

REFERENCES

1. Myers, R.H., Montgomery, D.C. (2002). Response surface methodology process and product optimization using designed experiments, Chapters 2 and 3, John Wiley and Sons, Inc. (2nd ed.), New York, USA.

2. Roux, W.J., Stander, N., Haftka, R.T. (1998). Response surface approximations for structural optimization, Int.Journal for Numerical Methods in Engineering, 42: 517–534.
3. Stander, N. (2001). The successive response surface method applied to sheet-metal forming, Proceedings of the first MIT conference on computational fluid and solid mechanics, pp. 481– 485.

4. Sun, H., Lee, S. (2005). Response surface approach to aerodynamic optimization design of helicopter rotor blade, International journal for numerical methods in engineering, 64: 125–142.

5. Nagesh Kumar, D., Janga Raddy, M. (2006). Ant Colony Optimization for Multi-Purpose Reservoir Operation, Water Resources Management, 20: 879–898.

6. Liang, Y.-C., Smith, A.E. (2004). An Ant Colony Optimization Algorithm for the Redundancy Allocation Problem (RAP), IEEE transaction on reliability, 53(3).

7. Abdul-Wahab, S.A., Abdo, J. (2007). Optimization of multistage flash desalination process by using a two-level factorial design, Applied Thermal Eng, 27: 413-421.

8. Dorigo, M. (1992). Optimization, learning and natural algorithms, PhD thesis, Dipartimento di Elettronica, Politecnico di Milano, Italia.

9. Dorigo, M., Maniezzo, V., Colorni, A. (1996). Ant System: Optimization by a Colony of Cooperating Agents, IEEE Trans. on Syst., Man., & Cybernetics-part B, 26(1): 29-41.

10. Zhao, J.H., Liu, Z., Dao, T.M. (2007). Reliability optimization using multiobjective ant colony system approaches, Reliability Engineering and System Safety, 92(1): 109-120.

11. Gagné, C., Gravel, M., Price, W.L. (2004). Optimisation multiobjectif à l'aide d'un algorithme de colonie de fourmis, INFOR, 42(1): 23-42.

12. Bilchey, G., Parmee, I.C. (1995). Ant colony metaphor for searching continuous design spaces, Lecture Notes in Computer Science, 993: 25-39.

13. Ling, C., Jie, S., Ling, Q. (2002). A method for solving optimization problem in continuous space by using ant colony algorithm, Journal of Software, 13(12): 2317-2323.

14. Monmarché, N., Venturini, G., Slimane, M. (2000). On how Pachycondyla apicalis ants suggest a new search algorithm, Future Generation Computer Systems, 16(8): 937-946.

15. Socha, K. (2004). ACO for continuous and mixed-variable optimization, Proceeding in Ant Colony Optimization and Swarm Intelligence (ANTS) 4th International Workshop, Lecture Notes in Computer Science, 3172, 2004, 25-36.

16. Pourtakdoust, S.H., Nobahari, H. (2004). An extension of ant colony system to continuous optimization problems, Proceedings in Ant Colony Optimization and Swarm Intelligence (ANTS) 4th International Workshop.

17. Dorigo, M., Bonabeau, E., Theraulaz, G. (2000). Ant algorithms and stigmergy, Proceeding in Future Generation Computer Systems, 16(8): 851-871.

18. Mathur, M., Karale Sachin, B., Prive, S., Jayaraman, V.K., Kulkarni, B.D. (2000). Ant colony approach to continuous optimization, Industrial and Engineering Chemistry Research, 39(10): 3814-3822.

19. Amtout, Y. (2003). Étude de l'influence des stratégies de coupe sur la performance de l'usinage en 5 axes par la méthode des surfaces de réponses. Mémoire de maîtrise en génie mécanique, Montréal, ÉTS, iv, 35-114.

BIBLIOGRAPHIE

Abdul-Wahab S.A., Abdo J., (2007), Optimization of multistage flash desalination process by using a two-level factorial design, *Applied Thermal Eng*, pp. 413-421.

Avila Sergio Luciano. (2006) *Optimisation multiobjectif et analyse de sensibilité appliquées à la conception de dispositifs. Application : Synthèse d'antennes à réflecteur embarquées dans un satellite.* Thèse présentée devant l'école centrale de Lyon pour l'obtention du grade de docteur. Spécialité : Génie électrique.

Baléo J-N, Bourges B., Courcoux Ph., faur-Brasquet C., Le Cloirec P. (2003). *Méthodologie expérimentale : Méthodes et outils pour les expérimentations scientifiques*. Paris : Éditions TEC et DOC.

Baskar N., Saravanan R., P. Asokan P., Prabhaharan G. (2004). Ants colony algorithm approach for multi-objective optimisation of surface grinding operations *The International Journal of Advanced Manufacturing Technology. Volume 23, Numbers 5-6.*

Battaglia G.J., Maynard J.M, (1996). Mean Square Error: A useful tool for statistical process management. *AMP J. Technol.*, Vol. 34, p. 1256-1260.

Bilchey G., Parmee I.C. (1995) Ant colony metaphor for searching continuous design spaces, *Lecture Notes in Computer Science, 993*, pp 25-39.

Chiu C. C., Pignatiello J. T. Jr., Cook D.F.(1994). Response surface methodology for optimal neural network selection. *Proceedings of the 6th International Conference on Tools with Artificial Intelligence*, New Orleans, LA, USA, Nov 6-9, pp 161-167.

Clément B., PhD. (2004). Cours 1. *MTH 6301 Planification et analyse statistique d'expériences*. Département de mathématiques et de génie industriel. École polytechnique de Montréal. Copyright © Génistat Conseils Inc., Montréal, Canada

Coppens Ph. (1990). Optimisation simultanée de plusieurs variables de réponse représentées par des modèles non linéaires. *Université de Bruxelles, Faculté des sciences appliquées.*

Derringer, G., and Suich, R., (1980), Simultaneous Optimization of Several Response Variables, *Journal of Quality Technology*, 12, 4, pp 214-219.

Dorigo M. (1992). *Optimization, learning and natural algorithms*. Thèse présentée à l'école polytechnique de Milan, Dipartimento di Elettronica, Politecnico di Milano, IT.

Dorigo M., Bonabeau E., Theraulaz G. (2000). Ant algorithms and stigmergy. *Proceeding in Future Generation Computer Systems, 16(8)*, pp. 851-871.

Dorigo M.; Maniezzo V.; Colorni A. (1996) Ant system: optimization by a colony of cooperating agents. *Systems, Man and Cybernetics, Part B, IEEE Transactions on*
Volume 26, Issue 1, pp 29-41

Gatza P.E., Mc Millan R.C. (October 3-6 1972). The use of experimental design and computerized data analysis in elatomer development studies. *Division of Rubber Chemistry, American Chemical society fall meeting, Paper N°6.* Cincinnati, Ohio.

Geoffrion A.M., Dyer J.S., Feinberg A. (1972) An Interactive Approach for Multi-Criterion Optimization with an Application to the Operation of an Academic Department. *Management Science vol. 19 no. 4*, pp. 357-363

Ghanmi S., Bouazizi M.L., Bouhaddi N. Optimisation multi-objectifs en mécanique des structures : Approche basée sur les surfaces de réponse adaptative. *Institut FEMTO ST UMR 6174, Laboratoire de Mécanique Appliquée R.Chaléat, Besançon France*

Groupy J. (2005). *Pratiquer les plans d'expériences*. Paris : L'usine nouvelle Éditions DUNOD.

Harrington E.C. Jr (1965). The Desirability Function. *Industrial Quality Control.*

Hsieh Kun-Lin. (2005). Parameter optimization of a multi-response process for lead frame manufacturing by employing artificial neural networks, *Springer-Verlag London Limited.*

Khuri, A.I., Conlon, M. (1981), Simultaneous optimization of multiple response represented by polynomial regression functions, *Technometrics*, Vol. 23 No.4, pp.363-75

Khuri A.I., Cornell J.A. (1987). Response surfaces :Designs and analyses. *Marcel Decker*, New York :.

Lanzi L., Castelletti L.M.L., Anghileri M. (2004). Multi-objective optimisation of composite absorber shape under crashworthiness requirements. *Composite Structures, Volume 65, Issues 3-4*, pp 433-441

Liang Y.C., Smith A.E. (2004). An Ant Colony Optimization Algorithm for the Redundancy Allocation Problem (RAP), *IEEE transaction on reliability, Vol. 53, No. 3.*

Ling C., Jie S., Ling Q. (2002). A method for solving optimization problem in continuous space by using ant colony algorithm. *Journal of Software, 13(12)*, pp 2317-2323.

Mathur M., Karale Sachin B., Prive S., Jayaraman V.K., Kulkarni B.D. (2000). Ant colony approach to continuous optimization. *Industrial and Engineering Chemistry Research, 39(10)*, pp 3814-3822.

Miettinen K., Makela M.M.(1993). Interactive method for nonsmooth multiobjective optimization with an application to optimal control. *Optimization methods and software, v2,* pp31-441

Monmarché N., Venturini G., Slimane M. (2000). On how Pachycondyla apicalis ants suggest a new search algorithm. *Future Generation Computer Systems, 16(8),* pp 937-946.

Myers. R.H, Montgomery. D.C (2002), *Response Surface Methodology*, Second Edition, New York, Wiley.

Nagesh Kumar D., Janga Raddy M. (2006). Ant Colony Optimization for Multi-Purpose Reservoir Operation, *Water Resources Management, Vol. 20*, pp.879–898.

Pareto V. (1896). Cours d'Economie Politique. Lausanne.

Park K.S., Kim K.J. (2005). Optimizing multi-response surface problems: How to use multi-objective optimization techniques

Park K., Moon S. (2005). Optimal design of heat exchangers using the progressive quadratic response surface model. *International Journal of Heat and Mass Transfer 48*. pp 2126–2139

Pourtakdoust S.H., Nobahari H. (2004). An extension of ant colony system to continuous optimization problems. *Proceedings in Ant Colony Optimization and Swarm Intelligence (ANTS) 4th International Workshop.*

Socha K. (2004). ACO for continuous and mixed-variable optimization, *Proceeding in Ant Colony Optimization and Swarm Intelligence (ANTS) 4th International Workshop, Lecture Notes in Computer Science, 3172*, pp. 25-36.

Sun H., Lee S. (2005). Response surface approach to aerodynamic optimization design of helicopter rotor blade. *International journal for numerical methods in engineering.*

Viana F.A.C., Kotinda G.I., Rade D.A., Valder S. Jr. (2006). Can ants design mechanical engineering systems? *IEEE Congress on Evolutionary Computation, CEC 2006*, Vancouver, BC, Canada, pp 831-837

Wang G.G, Dong Z. (2000). Design optimization of a complex mechanical system using adaptive response surface method. *Transactions of the Canadian Society for Mechanical Engineering*, v 24, n 1 B, p 295-306.

Wang, G. Gary, Dong, Zuomin. (Septembre 2-6 2003).Optimization of helicopter air intake scoop design . *ASME 2003 Design Engineering Technical Conferences and Computers and Information in Engineering Conference.* Chicago, Illinois, USA

Zhao J.H.,, Zhaoheng L., Dao T.M. (2007). Reliability optimization using multiobjective ant colony system approaches, *Reliability Engineering and System Safety*, 92(1), pp 109-120