

Université Mohamed Khider – Biskra

Faculté des Sciences et de la technologie

Département : Génie Mécanique

Réf. :



جامعة محمد خيضر بسكرة

كلية العلوم و التكنولوجيا

قسم: الهندسة الميكانيكية

المرجع:.....

Thèse présentée en vue de l'obtention
Du diplôme de
Doctorat en sciences en : Génie Mécanique

Option : Génie Mécanique

Intitulé

**Contribution à l'analyse de l'efficacité des techniques
d'optimisation appliquées en production industrielle**

Présentée par :

BELLOUFI Abderrahim

Soutenue publiquement le 29/01/2015

Devant le jury composé de :

| | | | |
|----------------------|-----------------------------|---------------|----------------------|
| MOUMMI Nouredine | Professeur | Président | Université de Biskra |
| ASSAS Mekki | Professeur | Rapporteur | Université de Batna |
| HECINI Mebrouk | Professeur | Co-Rapporteur | Université de Biskra |
| MAZOUZ Hammoudi | Professeur | Examineur | Université de Batna |
| BRIOUA Mourad | Professeur | Examineur | Université de Batna |
| CHEBBAH Mohamed Saïd | Maitre de conférences « A » | Examineur | Université de Biskra |

A ma famille, à tous mes amis et à ...

Remerciement

Je tiens à exprimer toute ma gratitude et ma reconnaissance à mon encadreur Pr. M. ASSAS d'avoir accepté de diriger ce travail et d'avoir mis à ma disposition tous les moyens qui m'ont permis de mener à terme cette étude et aussi pour ses précieux conseils et ses encouragements. Je remercie également Pr. M. HECINI, d'avoir accepté d'être mon co-encadreur. Je le remercie pour ses conseils judicieux,

Je tiens aussi à remercier Pr. N. MOUMMI pour l'intérêt qu'il a bien voulu porter à ce travail ainsi que pour l'honneur qu'il m'a fait en présidant le jury.

Mes remerciements vont également au : Dr. M.S. CHEBBAH, Pr. M. BRIOUA et au Pr. H. MAZOUZ pour s'être intéressé à ce travail et d'avoir bien voulu m'honorer de leur présence dans ce jury.

Je tien à remercie tous les membres du Laboratoire de Recherche en Productique de l'université de Batna, en particulier Pr. H. MAZOUZ.

Mes remerciements s'adressent également à tous ce qui m'ont aidé et m'ont permis de faire aboutir ce travail, en particulier : Pr. B. BENMOHAMMED et Mr. M. BELLOUFI.

Mes remerciements vont également à tous les enseignants qui ont contribué à ma formation en particulier les enseignants du Département de Génie Mécanique de l'université de Biskra et les enseignants du Département de Mécanique de l'université de Batna.

Table des Matières

| | |
|---------------------------------|---|
| I.1 Introduction Générale | 4 |
|---------------------------------|---|

Chapitre I

Les Techniques d'Optimisation

| | |
|--|----|
| I.1 Introduction | 7 |
| I.2 Généralités..... | 7 |
| I.2.1 Variables du problème | 7 |
| I.2.2 Espace de recherche | 7 |
| I.2.3 Fonction objectif | 8 |
| I.2.4 Les contraintes..... | 8 |
| I.2.5 Problème d'optimisation..... | 8 |
| I.2.6 Types de problèmes d'optimisation rencontrés en production industrielle..... | 8 |
| I.3 Conditions d'optimalité..... | 9 |
| I.3.1 Conditions de Lagrange | 10 |
| I.3.2 Conditions de Karush-Kuhn-Tucker (KKT) | 11 |
| I.4 Méthodes de transformation..... | 12 |
| I.4.1 Les méthodes de pénalité intérieure | 13 |
| I.4.2 Les méthodes de pénalité extérieure | 14 |
| I.5 Classement des méthodes numériques d'optimisation..... | 15 |
| I.6 Méthodes d'optimisation déterministes | 15 |
| I.6.1 Méthode des gradients conjugués (GC) | 16 |
| I.6.2 La programmation quadratique séquentielle (SQP) | 18 |
| I.7 Méthodes stochastiques..... | 19 |
| I.7.1 L'algorithme génétique (GA)..... | 20 |
| I.7.2 La méthode d'essaim particulaire (PSO) | 24 |

| | |
|---|----|
| I.8 Conclusion | 26 |
| Chapitre II | |
| Etude Comparative des Techniques d'Optimisation | |
| II.2 Enjeux de la comparaison | 27 |
| II.3 Fonctions tests | 28 |
| II.3.1 Fonction de Griewangk's | 29 |
| II.3.2 Fonction de Ackley's | 29 |
| II.3.3 Fonction de Rosenbrock's | 30 |
| II.3.4 Fonction de Schwefel's | 30 |
| II.3.5 Fonction de Rastrigin's | 31 |
| II.4 Critères d'arrêt | 31 |
| II.5 Paramétrage des algorithmes stochastique..... | 31 |
| II.5.1 Balayage pour chaque couple algorithme/benchmark | 32 |
| II.6 Organigrammes des méthodes d'optimisation utilisées..... | 32 |
| II.6.1 La méthode du gradient conjuguée | 32 |
| II.6.2 La programmation quadratique séquentielle | 34 |
| II.6.3 L'algorithme génétique | 35 |
| II.6.4 La méthode d'essaim particulière | 36 |
| II.7 Résultats | 37 |
| II.7.1 Méthodes déterministes..... | 37 |
| II.7.2 Méthodes stochastique | 38 |
| II.8 Discussion | 43 |
| II.9 Proposition d'une méthode hybride | 44 |
| II.10 La Méthode hybride GA – SQP..... | 44 |
| II.10.1 Résultats de la méthode hybride GA – SQP | 46 |
| II.11 Comparaison | 46 |
| II.11.1 Critères de comparaison | 46 |
| II.12 Conclusion | 49 |

Chapitre III

Application Industrielle

| | |
|--|----|
| III.1 Introduction | 50 |
| III.2 Formulation d'un modèle d'optimisation des conditions de coupe en tournage | 50 |
| III.2.1 Généralité sur le procédés de tournage | 50 |
| III.2.2 Modèle 1 | 52 |
| III.2.3 Modèle 2 | 61 |
| III.3 Formulation d'un modèle d'optimisation des conditions de coupe en fraisage | 65 |
| III.3.1 Généralité sur le procédés de fraisage | 65 |
| III.3.2 Les variables d'optimisation | 67 |
| III.3.3 Formulation de la fonction objectif en fraisage | 67 |
| III.3.4 Formulation des contraintes | 69 |
| III.3.5 Modélisation multi-passe des conditions de coupe en fraisage | 70 |
| III.4 Applications | 77 |
| III.4.1 Exemple d'application 1 | 77 |
| III.4.2 Exemple d'application 2 | 85 |
| III.4.3 Résultats et discussion | 86 |
| III.4.4 Exemple d'application 3 | 87 |
| III.5 Conclusion | 93 |
| Conclusion Générale et Perspectives | 94 |
| Références bibliographiques | 96 |

Liste des Figures

Chapitre I

Les Techniques d'Optimisation

| | | |
|-------------------|--|----|
| Figure I.1 | Interprétation géométrique des conditions d'optimalité de Lagrange | 11 |
| Figure I.2 | Interprétation géométrique des conditions d'optimalité de KKT | 12 |
| Figure I.3 | Le croisement | 22 |
| Figure I.4 | La mutation | 22 |
| Figure I.5 | Sélection par roue de la fortune | 23 |

Chapitre II

Etude Comparative des Techniques d'Optimisation

| | | |
|---------------------|---|----|
| Figure II.1 | Taux d'utilisation de six algorithmes stochastiques | 28 |
| Figure II.2 | Fonction de Griewangk's | 29 |
| Figure II.3 | Fonction de Ackley's | 29 |
| Figure II.4 | Fonction de Rosenbrock's | 30 |
| Figure II.5 | Fonction de Schwefel's | 30 |
| Figure II.6 | Fonction de Rastrigin's | 31 |
| Figure II.7 | Organigramme de la méthode du gradient conjuguée | 33 |
| Figure II.8 | Organigramme de la programmation quadratique séquentielle | 34 |
| Figure II.9 | Organigramme de l'algorithme génétique | 35 |
| Figure II.10 | Organigramme de la méthode d'essai particulière | 36 |
| Figure II.11 | Résultats de minimisation de la fonction Schwefel's avec l'algorithme génétique | 38 |
| Figure II.12 | Résultats de minimisation de la fonction Schwefel's avec l'algorithme PSO | 39 |
| Figure II.13 | Résultats de minimisation de la fonction Ackley's avec l'algorithme génétique | 39 |
| Figure II.14 | Résultats de minimisation de la fonction Ackley's avec l'algorithme PSO | 40 |
| Figure II.15 | Résultats de minimisation de la fonction Rosenbrock's avec l'algorithme génétique | 40 |
| Figure II.16 | Résultats de minimisation de la fonction Rosenbrock's avec l'algorithme PSO | 41 |
| Figure II.17 | Résultats de minimisation de la fonction Schwefel's avec l'algorithme génétique | 41 |
| Figure II.18 | Résultats de minimisation de la fonction Schwefel's avec l'algorithme PSO | 42 |
| Figure II.19 | Résultats de minimisation de la fonction Rastrigin's avec l'algorithme génétique | 42 |
| Figure II.20 | Résultats de minimisation de la fonction Rastrigin's avec l'algorithme PSO | 43 |

| | | |
|---------------------|-----------------------------------|----|
| Figure II.21 | Organigramme de la méthode GA-SQP | 45 |
| Figure II.22 | Nombre d'évaluations moyen | 47 |
| Figure II.23 | Taux de convergence | 47 |

Chapitre III

Application Industrielle

| | | |
|----------------------|---|----|
| Figure III.1 | Mouvements de coupe et d'avance en tournage | 51 |
| Figure III.2 | Conditions de coupe en tournage | 51 |
| Figure III.3 | Dimension de la pièce à usinée | 53 |
| Figure III.4 | Schéma représentant la programmation dynamique lorsque la profondeur de coupe totale est divisée en 7 sections. | 58 |
| Figure III.5 | Organigramme de la méthodologie pour l'optimisation multi-passe en tournage | 60 |
| Figure III.6 | Organigramme de la méthode de résolution | 61 |
| Figure III.7 | Mouvements de coupe et d'avance en fraisage | 65 |
| Figure III.8 | Conditions de coupe en fraisage | 65 |
| Figure III.9 | Schéma représentant la programmation dynamique lorsque la profondeur de coupe totale est divisée en 5 sections. | 71 |
| Figure III.10 | Organigramme de la méthodologie pour l'optimisation multi-passe | 75 |
| Figure III.11 | Comparaison entre les résultats obtenus par Agapiou [35] et les résultats obtenus par la méthode Hybrid GA-SQP | 84 |
| Figure III.12 | Comparaison entre les résultats obtenus par la méthode Hybrid GA-SQP et les résultats obtenus avec d'autres techniques. | 93 |

Liste des Tableaux

Chapitre II

Etude Comparative des Techniques d'Optimisation

| | | |
|---------------------|---|----|
| Tableau II.1 | Résultats de minimisation de la fonction Griewangk's | 37 |
| Tableau II.2 | Résultats de minimisation de la fonction Ackley's | 37 |
| Tableau II.3 | Résultats de minimisation de la fonction Rosenbrock's | 37 |
| Tableau II.4 | Résultats de minimisation de la fonction Schwefel's | 38 |
| Tableau II.5 | Résultats de minimisation de la fonction Rastrigin's | 38 |
| Tableau II.6 | Jeu de paramètres optimums | 48 |

Chapitre III

Application Industrielle

| | | |
|-----------------------|---|----|
| Tableau III.1 | Tableau des résultats d'optimisation | 76 |
| Tableau III.2 | Les paramètres d'usinage | 77 |
| Tableau III.3 | Les valeurs optimales de V_c et f (Critère du temps de production) | 79 |
| Tableau III.4 | Les valeurs optimales de V_c et f (critère de coût de production) | 79 |
| Tableau III.5 | Paramètres optimums d'usinage pour N=4(Critère du coût de production) | 80 |
| Tableau III.6 | Paramètres optimums d'usinage pour N=4(Critère du temps de Production) | 80 |
| Tableau III.7 | Paramètres optimums d'usinage pour N=8(Critère du coût de production) | 81 |
| Tableau III.8 | Paramètres optimums d'usinage pour N=8(Critère du temps de Production) | 81 |
| Tableau III.9 | Paramètres optimums d'usinage pour N=10(Critère du coût de production) | 82 |
| Tableau III.10 | Paramètres optimums d'usinage pour N=10(Critère du temps de production) | 82 |
| Tableau III.11 | Paramètres optimums d'usinage pour N=12(Critère du coût de production) | 83 |
| Tableau III.12 | Paramètres optimums d'usinage pour N=12(Critère du temps de production) | 83 |
| Tableau III.13 | Les paramètres d'usinage | 85 |
| Tableau III.14 | Paramètres optimums d'usinage | 86 |
| Tableau III.15 | Paramètres optimums d'usinage pour les différentes méthodes | 86 |
| Tableau III.16 | Les paramètres d'usinage | 87 |
| Tableau III.17 | Les paramètres d'usinage (suite) | 88 |
| Tableau III.18 | Les valeurs optimales de V_c , et f_z | 89 |

| | | |
|-----------------------|--|----|
| Tableau III.19 | Résultats de la programmation dynamique pour (N=3) | 89 |
| Tableau III.20 | Paramètres optimales de coupe pour N= 3 | 89 |
| Tableau III.21 | Résultats de la programmation dynamique pour (N=5) | 90 |
| Tableau III.22 | Paramètres optimales de coupe pour N= 5 | 90 |
| Tableau III.23 | Résultats de la programmation dynamique pour (N=8) | 91 |
| Tableau III.24 | Paramètres optimales de coupe pour N= 8 | 91 |

Nomenclature

| <u>Symbole</u> | <u>Unité</u> | <u>Désignation</u> |
|----------------|--------------|---|
| V_c | m/min | Vitesse de coupe |
| N | tr/min | Vitesse de la broche |
| D | mm | Diamètre de la pièce en tournage et diamètre de la fraise en fraisage |
| V_f | mm/min | La vitesse d'avance |
| f | mm/tour | Avance par tour |
| a | mm | Profondeur de passe |
| t_c | min | temps de coupe |
| t_{cs} | min | Temps de changement d'outil |
| t_r | min | Temps de retour |
| t_h | min | Temps de montage et démontage de la pièce |
| T_u | min | Le temps d'usinage unitaire |
| lu | mm | Longueur usinée |
| L | mm | Longueur totale de déplacement de l'outil |
| C_0 | \$/min | Coût de machine |
| C_t | \$/arrêt | Coût d'une arrête de coupe |
| C_u | \$ | Coût d'usinage unitaire |
| K | mm | Constant de durée de vie |
| P_m | kW | Puissance du moteur |
| F_{max} | N | La force de coupe maximale |
| d | mm | La profondeur d'une section |
| D_c | mm | La profondeur de coupe totale |
| N | | Nombre de sections |
| T_t | min | le temps de production total |
| C_t | \$ | Le coût de production total |
| $DMOP$ | mm | La profondeur de coupe optimale pour l'usinage à passe simple |
| $DMAXP$ | mm | La profondeur de coupe maximale admissible |
| a_{min} | mm | Profondeur de passe minimal |

| | | |
|------------|----------------------|--|
| a_{\max} | mm | Profondeur de passe maximal |
| d_a | mm | Diamètre de l'arbre |
| E | kg f/mm ² | Module d'élasticité de la matière d'arbre |
| f_z | mm/dent | L'avance par dent |
| m | | Exposant déterminée empiriquement |
| N_p | | Nombre de passes |
| P_C | Kw | Puissance de coupe |
| T | min | Durée de vie de l'outil |
| T_p | min | Temps de préparation de la machine par composante |
| T_s | min | Le temps de mettre en place de la machine pour une nouvelle partie |
| T_m | min | Le temps de changement par composant |
| T_{pr} | min | Temps de production totale |
| η | | Rendement de la machine |
| C_I | (\$/piece) | Coût dû au chargement et de déchargement |
| C_M | (\$/piece) | Coût de coupe en temps réel dans la coupe |
| C_R | (\$/piece) | Coût de remplacement d'outil de coupe |
| C_T | (\$/piece) | Coût d'un outil de coupe |

Symboles spécifiques pour les modèles de tournage

| | | |
|-----------------|----------|--|
| h_1, h_2 | (min) | Constantes relatives aux déplacements de l'outil de coupe |
| k_0 | (\$/min) | Coût de main-d'œuvre directe et des frais généraux |
| k_1, μ, ν | | Constantes de l'équation de la force de coupe |
| k_3, k_4, k_5 | | Constantes de la relation entre les paramètres d'ébauche et de la finition |
| λ, ν | | Constantes liées à l'expression de la région de coupe stable |
| q | | Poids pour T_p [0,1] |
| R | (mm) | Rayon de bec de l'outil de coupe |
| SC | | Limite de la région stable de la contrainte de coupe |
| T_p | (min) | Durée de vie de la combinaison de T_r et T_s |
| C_f, x, y, z | | Coefficients spécifiques et exposantes de la force de coupe |

| | | |
|-----------------|--------------------|---|
| C_s, e, g, i | | Coefficients spécifiques et exposantes de la rugosité de surface résultante |
| R_{\max} | μm | La rugosité maximale autorisée de surface |
| C_m, m, p, q | | Coefficients spécifiques et exposantes de la température de coupe |
| θ_{\max} | $^{\circ}\text{C}$ | Le maximum autorisé de température de coupe |

Symboles spécifiques pour le modèle de fraisage

| | | |
|-----------------|--------------------------|--|
| b_v, b_z | | Exposants déterminée empiriquement |
| B | mm | La largeur de fraisage |
| B_m | | Coefficient de correction de l'équation de durée de vie de l'outil |
| B_h | | Coefficient de correction de l'équation de durée de vie de l'outil |
| B_t | | Coefficient de correction de l'équation de durée de vie de l'outil |
| B_p | | Coefficient de correction de l'équation de durée de vie de l'outil |
| C_v | | Constante de l'influence de tous les facteurs |
| $C_{z,p}$ | | Constante de l'équation force de coupe |
| e | mm | La valeur admissible de déviation de l'arbre |
| e_v, e_z | | Exposants déterminée empiriquement |
| F_C | kgf | Force de coupe périphérique moyenne |
| F_d | kgf | Force admissible à l'égard de déviation arbre |
| F_s | kgf | Force admissible à l'égard de la force tonnelle |
| k_b | kgf/mm^2 | La contrainte admissible de flexion du matériau d'arbre |
| k_t | kgf/mm^2 | La contrainte admissible de torsion de la matière d'arbre |
| q, q_v | | Exposants déterminée empiriquement |
| λ_s | degrés | Angle d'inclinaison de couper |
| r_v, r_z | | Exposants déterminée empiriquement |
| u_v, u_z | | Exposant déterminée empiriquement |
| Z | | Nombre de dent |
| C_v | | Coefficient de loi de Taylor |
| n | | Exposant de loi de Taylor |
| a_1, a_2, a_3 | | Exposants de la loi de Gilbert dépendants de la nuance de l'outil |

Introduction Générale

Introduction Générale

La production industrielle est un domaine soumis à de fortes contraintes économiques, technologiques, et autres. Les industriels ont besoin d'optimiser leurs processus de production afin d'augmenter la productivité et de réduire les différents coûts. Pour atteindre ces objectifs il faut chercher une solution satisfaisante ce qui nécessite le choix et l'implémentation d'une méthode d'optimisation. Ces méthodes sont très nombreuses et variées, certaines s'appuient sur une analyse mathématique du modèle, d'autres utilisent la génération de nombres aléatoires (méthodes stochastiques) et reproduisent des comportements issus de la nature pour identifier une solution optimale : sélection naturelle, vols d'insectes, systèmes immunitaires, etc. Il est donc nécessaire d'étudier le choix d'une méthode d'optimisation appropriée.

Ce choix repose sur [1]:

- Sa capacité à traiter des fonctions « objectifs » non-linéaires, non-dérivables et multi-modales,
- Son habilité à utiliser le principe de parallélisations,
- Sa facilité d'utilisation (compréhension du principe, faible nombre de paramètres),
- Son bon taux de convergence

L'optimisation, qui concerne toute amélioration économique, technologique et de qualité, est devenue un objectif majeur pour la production industrielle. Le champ d'opportunité de l'optimisation est très étendu. Néanmoins, l'avancement concernant l'optimisation dans la production mécanique est en retard par rapport aux autres progrès obtenus dans l'industrie [2].

Le champ d'utilité de l'optimisation en production mécanique étant très étendu, nous nous sommes alors intéressés à une seule partie de cette problématique : l'optimisation des conditions de coupe.

En pratique, l'optimisation des conditions de coupe est fréquemment confrontée à deux problèmes [2]:

D'une part, la performance de l'optimisation est très dépendante du modèle utilisé, ainsi de meilleurs résultats d'optimisation résultent de l'utilisation des modèles. En conséquence, la plus grande partie des travaux sur l'optimisation des conditions de coupe est menée soit en absence de modèles mathématiques, soit avec un modèle mathématique simple. Dans ces cas, les résultats obtenus restent modestes.

D'autre part, les résultats de l'optimisation dépendent de la bonne construction du problème d'optimisation et du choix de la méthode de résolution de ce problème.

Une bonne stratégie d'optimisation consiste alors à utiliser le modèle mathématique le plus adéquat au problème d'optimisation et, enfin, à choisir la méthode de résolution la plus adaptée aux besoins. Dans cette optique, l'objectif de la thèse est de proposer des stratégies d'optimisation adaptées aux procédés d'usinage (tournage et fraisage), pour aboutir aux conditions optimales de coupe.

L'intérêt de l'optimisation des conditions de coupe réside dans l'amélioration de certains paramètres technologiques ou techno-économique d'usinage, dites critères d'optimisation (prix de revient des pièces usinées, temps de production, productivité, état de surface,...etc.), en tenant compte des limites de performance de la machine, de l'outil de coupe et de la matière usinée. Ces limites sont tirées des lois de coupe et exprimées en fonction des conditions de coupe (vitesse de coupe, avance par dent et profondeur de passe). Elles sont appelées contraintes d'optimisation et elles limitent le domaine de recherche des conditions optimales.

En vue d'atteindre cet objectif le travail présenté dans cette thèse s'articule de la façon suivante :

Le premier chapitre porte sur une étude bibliographique permet de rappeler les principales techniques d'optimisation ainsi que leurs avantages et inconvénients, après avoir clarifiés quelques généralités avec une présentation des conditions d'optimalité pour les problèmes d'optimisation avec contraintes, puis une classification des différentes méthodes d'optimisation, avant de présenter quelques méthodes, les plus adaptées pour le problème des conditions de coupe.

Dans le deuxième chapitre les techniques d'optimisation présentées dans le chapitre précédent ainsi qu'une méthode hybride proposée dans ce chapitre ont été implémentées et sont comparés à travers de différentes fonctions-tests. Plusieurs aspects de ces algorithmes sont étudiés et l'un d'entre eux est sélectionné sur la base de son efficacité afin d'être appliqué au cas industriel présenté au troisième chapitre.

Le troisième chapitre permet de faire le point sur la formulation des modèles d'optimisation pour le choix des conditions de coupe en tournage et en fraisage, on se basant sur les fonctions objectifs et les contraintes liés aux problèmes, nous expliquons enfin, les méthodes utilisées pour la modélisation et nous traitons trois exemples réels :

- L'exemple 1 concerne l'étude d'une opération de chariotage multi-passe avec un modèle d'optimisation possédant trois variables de décision. Ce modèle est basé sur la programmation dynamique.
- L'exemple 2 captive toujours l'étude d'une opération de chariotage multi-passe mais avec un modèle d'optimisation possède six variables de décision.
- L'exemple 3 l'étude d'une opération de fraisage plan (surfaçage multi-passe) avec un modèle d'optimisation possédant trois variables de décision. Ce modèle est basé sur la programmation dynamique.

Enfin, nous concluons cette thèse en analysant les principaux résultats de cette étude, celles-ci portent notamment sur la représentativité des cas étudiés, l'efficacité de la stratégie d'optimisation utilisée lors des exemples d'applications et la qualité des résultats vis-à-vis à d'autres résultats issus de la littérature pour les mêmes problèmes. Nous finalisons cette conclusion par quelques perspectives qui sont proposées tant au niveau de l'application de ce travail.

Chapitre I
Les Techniques d'Optimisation

I.1 Introduction

L'optimisation peut être définie comme la science qui détermine la meilleure solution à certains problèmes mathématiquement définie, qui sont souvent des modèles de physique réelle. C'est une technique qui permet de « quantifier » les compromis entre des critères parfois non commensurables.

En fait, l'optimisation cherche à améliorer une performance en se rapprochant d'un point optimum.

Dans ce chapitre nous dresserons dans une première section une rapide recherche bibliographique des principales techniques de programmation non linéaire. Une seconde section abordera les Méta-heuristiques d'optimisation.

I.2 Généralités

La résolution d'un problème d'optimisation s'appuie généralement sur un algorithme d'optimisation. L'algorithme d'optimisation cherche le jeu de paramètres de l'objet à optimiser donnant à une fonction relative au problème, la valeur maximale ou minimale.

L'algorithme d'optimisation doit permettre de converger vers l'objet optimal en minimisant ou maximisant cette fonction par rapport aux paramètres variables. Généralement, pour toute méthode d'optimisation, l'utilisateur définit les variables du problème, l'espace de recherche et la fonction d'adaptation relative au problème.

I.2.1 Variables du problème

La définition des variables passe par un choix empirique. C'est à l'utilisateur de le définir. Il peut avoir à faire varier un grand nombre de paramètres afin d'augmenter les degrés de liberté de l'algorithme.

Dans notre étude nous désignerons par x_1, x_2, \dots, x_n les n variables du problème. Celles-ci peuvent être réelles, complexes, entières, booléennes, etc. Ici elles seront supposées réelles.

I.2.2 Espace de recherche

Dans certains algorithmes d'optimisation tels que les Stratégies d'Evolution, la norme de l'espace de recherche est infini, en effet seule la population initiale est confinée dans un espace fini.

Mais dans le cas des algorithmes génétique, il est généralement nécessaire de définir un espace de recherche fini. Cette limitation de l'espace de recherche n'est généralement

pas problématique. De ce fait, pour des raisons technologiques ou informatiques (taille de la fenêtre de modélisation), les intervalles de définition des variables sont en général limités [3].

I.2.3 Fonction objectif

Un algorithme d'optimisation nécessite généralement la définition d'une fonction rendant compte de la pertinence des solutions potentielles, à partir des grandeurs à optimiser.

La convergence de l'algorithme vers l'optimum de la fonction objectif f , quelle que soit sa définition en est la principale conséquence. Ainsi, la pertinence de la solution est un enjeu de la question posée à l'ordinateur. La fonction f est la traduction en langage mathématique du désir de l'utilisateur.

I.2.4 Les contraintes

On appelle contraintes toutes les relations limitant le choix des valeurs possibles des variables.

Ces relations peuvent être de simples bornes sur les variables.

I.2.5 Problème d'optimisation

Un programme mathématique est un problème d'optimisation sous contraintes dans \mathbb{R}^n de la forme :

$$\left\{ \begin{array}{l} \text{minimiser } f(x); \quad f \in \mathbb{R} \\ \text{sous les contraintes} \\ g_i(x) = 0 \quad , i = 1, \dots, m. \\ h_j(x) \leq 0 \quad , j = 1, \dots, n. \end{array} \right. \quad (\text{I-1})$$

Où le vecteur $x \in \mathbb{R}^n$ a pour composantes (x_1, x_2, \dots, x_n) qui sont les inconnues (les variables) du problème (I-1).

$f(x)$: La fonction objectif ou critère économique.

$g_i(x) = 0 \quad (i = 1, \dots, m)$: L'ensemble des contraintes d'égalité du problème.

$h_j(x) \leq 0 \quad (j = 1, \dots, n)$: L'ensemble des contraintes d'inégalité du problème.

I.2.6 Types de problèmes d'optimisation rencontrés en production industrielle

Dans l'univers des méthodes disponibles pour la résolution de problèmes d'optimisation, certaines méthodes sont plus adaptées aux caractéristiques d'un problème

en particulier. Il est indispensable d'identifier, dans un premier temps, les différents types de problèmes d'optimisation et le groupe de méthodes adéquat pour leur résolution. Evidemment, un système peut être optimisé grâce à l'action d'experts, ou bien par essai-erreur, mais nous ne traiterons que les cas d'optimisation à l'aide d'outils mathématiques.

Selon les caractéristiques du critère d'optimisation, des contraintes et des variables du système, on peut trouver différents types de problèmes d'optimisation : optimisation linéaire sous contraintes (programmation linéaire), optimisation non-linéaire sous contraintes (programmation non linéaire), optimisation multicritères (programmation multi-objectif), optimisation non-linéaire à variables entières ou mixtes « mixed integer nonlinear programming »,...

Si les fonctions de critère et de contraintes sont linéaires et les variables du système sont continues, c'est à dire si celles-ci peuvent prendre des valeurs réelles, il s'agit d'un problème du type programmation linéaire (LP). Les problèmes de programmation quadratique (QP) sont tout à fait différents de ceux linéaires car le critère est une fonction quadratique. Pour résoudre ces deux types de problèmes il existe des méthodes spécifiques rapides et fiables, avec un nombre fini de pas [4].

D'autre part, si les critères et/ou les contraintes sont non linéaires, on est en face à un problème de type programmation non linéaire (NLP). Dans ce cas, il est très difficile voire impossible de calculer analytiquement la solution, ce qui exige l'intervention d'un calcul défini par des méthodes numériques.

Le problème posé par l'optimisation de la production industrielle (optimisation des conditions de coupe en usinage) est du type non linéaire sous contraintes.

I.3 Conditions d'optimalité

Afin d'analyser ou de résoudre de manière efficace un problème d'optimisation, il est fondamental de pouvoir disposer de conditions d'optimalité. En effet, celles-ci nous servent non seulement à vérifier la validité des solutions obtenues, mais souvent l'étude de ces conditions aboutit au développement des algorithmes de résolution eux-mêmes.

On distingue deux types de conditions d'optimalité :

- Conditions de Lagrange,
- Conditions de Karush-Kuhn-Tucker.

I.3.1 Conditions de Lagrange

Théorème I.1 [5] : Soit $x^* \in S = \{x \in \mathbb{R}^n, g(x) = 0\}$ un point régulier solution du problème d'optimisation, vérifiant donc : $f(x^*) \leq f(x), \forall x \in S$.

Et Soit la matrice Jacobienne $J(x)$ de dimension $m \times n$:

$$J(x) = \begin{bmatrix} \nabla g_1(x)^T \\ \vdots \\ \nabla g_m(x)^T \end{bmatrix} \Leftrightarrow J^T(x) = \nabla g^T(x) \quad (\text{I-2})$$

Un point x^* tel que $g(x^*) = 0$ est appelé un point régulier des contraintes si $\text{rang } J(x^*) = m$.

Les composantes du vecteur λ sont appelées multiplicateurs de Lagrange.

a. Condition nécessaire du premier ordre [6]

Soit $f(x)$ minimum en x^* sous la contrainte $h(x) = 0$, Si x^* est un point régulier, alors :

$$\exists \lambda^* \in \mathbb{R}^m / \nabla f(x^*) + \nabla g^T(x^*) \lambda^* = 0 \quad (\text{I-3})$$

Condition nécessaire mais pas suffisante.

Soit : $l(x, \lambda) = f(x) + \lambda^T g(x)$ fonction de lagrange.

Soit : $L(x, \lambda)$ le hessien de $l(x, \lambda)$.

On a donc

$$L(x, \lambda) = H(x) + \sum_{i=1}^m \lambda_i G_i(x) \quad (\text{I-4})$$

Avec $H(x)$: le Hessien de $f(x)$

$G_i(x)$: le Hessien de $g_i(x)$

b. Condition nécessaire du second ordre

Soit $f(x)$ minimum en x^* sous la contrainte $g(x) = 0$, Si x^* est un point régulier, alors :

$$\exists \lambda^* / \quad 1^\circ \quad \nabla l(x^*, \lambda^*) = \nabla f(x^*) + \nabla g^T(x^*) \lambda^* = 0 \quad \text{et} \quad g(x^*) = 0 \quad (\text{I-5})$$

$$2^\circ \quad y^T L(x^*, \lambda^*) y > 0 \quad \forall y \in T(x^*) \Leftrightarrow \forall y / \quad y^T \nabla g^T(x^*) = 0 \quad (\text{I-6})$$

c. Condition suffisante

$$\text{Si } \exists x^* \text{ et } \exists \lambda^* / 1^\circ \nabla f(x^*) + \nabla g^T(x^*) \lambda^* = 0 \text{ et } g(x^*) = 0. \tag{I-7}$$

$$2^\circ y^T L(x^*, \lambda^*) y > 0 \text{ ou } < 0 \quad \forall y \neq 0 / y^T \nabla g^T(x^*) = 0.$$

(I-8)

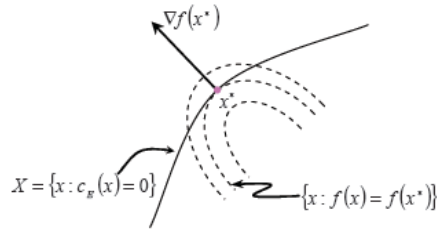


Figure I.1 Interprétation géométrique des conditions d'optimalité de Lagrange

I.3.2 Conditions de Karush-Kuhn-Tucker (KKT)

Théorème I.2 [5] : Soit le problème d'optimisation: $\min f(x)$ avec $g(x) = 0$ et $h(x) \leq 0$. tel que $f : \mathbb{R}^n \rightarrow \mathbb{R}$ et $x \in \mathbb{R}^n$

Une contrainte d'inégalité $h(x) \leq 0$ est active en x si $h(x) = 0$. Elle est inactive si $h(x) < 0$

Un point x^* satisfaisant les contraintes $g(x^*) = 0$ et $h_j(x^*) = 0 \quad \forall j \in I(x^*)$. l'ensemble des indices des contraintes actives est appelé un point régulier si les vecteurs $\nabla g_i(x^*), \nabla h_j(x^*) \quad j \in I(x^*)$ sont linéairement indépendants.

a. Condition nécessaire du premier ordre [6]

Soit x^* qui minimise $f(x)$ sous les contraintes $g(x) = 0$ et $h(x) \leq 0 (\geq 0)$.

Si x^* est un point régulier, alors :

$$\exists \lambda^*, \mu^* / 1^\circ \mu^* \geq 0 (\mu^* \leq 0). \tag{I-9}$$

$$2^\circ \nabla f(x^*) + \nabla g^T(x^*) \lambda^* + \nabla h^T(x^*) \mu^* = 0. \tag{I-10}$$

$$3^\circ \mu^{*T} h(x^*) = 0. \tag{I-11}$$

$\Rightarrow \mu^* = 0$ si $h(x^*) < 0$ Contrainte inactive.

b. Condition nécessaire du second ordre

Soit

$$L(x, \lambda, \mu) = H(x) + \sum_{i=1}^m \lambda_i G_i(x) + \sum_{j=1}^p \mu_j H_j(x). \quad (I-12)$$

Avec $H(x)$: le Hessien de $f(x)$

$G_i(x)$: le Hessien de $g_i(x)$

$H_j(x)$: le Hessien de $h_j(x)$

Soit x^* qui minimise $f(x)$ sous les contraintes $g(x) = 0$ et $h(x) \leq 0 (\geq 0)$.

Si x^* est un point régulier, alors :

$$\exists \lambda^*, \mu^* / y^T L(x^*, \lambda^*, \mu^*) y \geq 0 \quad \forall y \neq 0 / \begin{cases} y^T \nabla g^T(x^*) = 0 \\ y^T \nabla h_j(x^*) = 0 \quad \forall j \in I(x^*) \end{cases} \quad (I-13)$$

c. Condition suffisante

Si $\exists \lambda^*, \mu^* / 1^\circ \quad \mu^* \geq 0 (\mu^* \leq 0)$.

$$2^\circ \quad \nabla f(x^*) + \nabla g^T(x^*) \lambda^* + \nabla h^T(x^*) \mu^* = 0. \quad \text{et} \quad g(x^*) = 0 \quad \begin{cases} h(x^*) \leq 0 \\ h(x^*) \geq 0 \end{cases} \quad (I-14)$$

$$3^\circ \quad \mu^{*T} h(x^*) = 0.$$

$$4^\circ \quad y^T L(x^*, \lambda^*, \mu^*) y > 0 (\text{où } < 0) \quad \forall y \neq 0 / \begin{cases} y^T \nabla g^T(x^*) = 0 \\ y^T \nabla h_j(x^*) = 0 \quad \forall j \in I(x^*) \end{cases} \quad (I-15)$$

$\Rightarrow x^*$ est un minimum local au sens strict de $f(x)$ avec $g(x) = 0$ et $h(x) \leq 0$
maximum

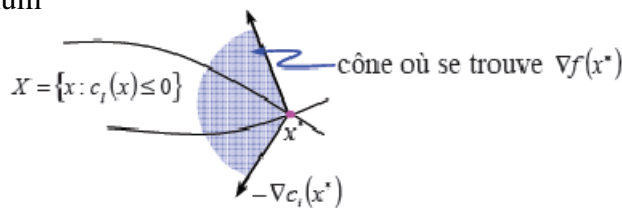


Figure I.2 Interprétation géométrique des conditions d'optimalité de KKT

I.4 Méthodes de transformation

Dans toutes les méthodes de transformation, la fonction objectif et les contraintes sont rassemblées dans une fonction de transformation ϕ sous la forme suivante [7, 8, 9, 10,11]:

$$\phi(x, r) = f(x) + P(g(x), r) \quad (I-16)$$

P est la fonction de pénalité, dont l'action et l'amplitude de pénalisation sont contrôlées par le paramètre de pénalités r [9, 10, 11,12]. L'intérêt dans ce type de transformation est clair: si ϕ et r sont bien choisis, le problème pourrait être résolu en utilisant un algorithme d'optimisation sans contrainte.

La pénalisation est un concept simple qui permet de transformer un problème d'optimisation avec contraintes en un problème ou en une suite de problèmes d'optimisation sans contrainte. C'est un outil à la fois théorique et numérique.

L'approche par pénalisation est parfois utilisée pour étudier un problème d'optimisation dont les contraintes sont difficiles à prendre en compte, alors que le problème pénalisé a des propriétés mieux comprises ou plus simples à mettre en évidence. Si on a de la chance, ou si la pénalisation est bien choisie, des passages à la limite parfois délicats permettent d'obtenir des propriétés du problème original (l'existence de solutions par exemple). D'autre part, la pénalisation est un outil permettant d'étudier les problèmes d'optimisation avec et sans contraintes identiquement. D'un point de vue numérique, cette transformation en problèmes sans contraintes permet d'utiliser des algorithmes d'optimisation sans contraintes pour obtenir la solution de problèmes dont l'ensemble admissible peut avoir une structure complexe. Cette approche est très souvent utilisée et permet d'obtenir rapidement des solutions de qualité suffisante [13].

I.4.1 Les méthodes de pénalité intérieure

Le principe de ces méthodes réside dans la transformation d'un problème contraint en une séquence de problèmes sans contraintes, en ajoutant au coût une pénalité en cas de violation de celles-ci. Un tel sous problème est résolu à chaque itération d'une méthode de pénalité. L'appellation "pénalité intérieure" est employée car le minimum est approché depuis l'intérieur de X . Les méthodes de barrière s'appliquent aux problèmes dont l'ensemble admissible X est défini uniquement par une collection d'inégalités :

$$\begin{cases} \text{minimiser } f(x) \\ \text{sous les contraintes } h_i(x) \leq 0, \quad i = 1, \dots, m \end{cases}$$

La fonction de barrière, notée $B(x)$, est ajoutée à $f(x)$; elle est continue sur X_I (où $X_I = \{x \mid h_i(x) < 0, \forall i \in \{1, \dots, m\}\}$) et sa valeur tend vers l'infini lorsque la frontière de X est approchée par l'intérieur, c'est-à-dire lorsque l'un des $h_i(x)$ approche le zéro par les valeurs négatives. Une itération de la méthode consiste ensuite à minimiser la fonction

$f(x) + \varepsilon B(x)$ (où ε est un paramètre réel strictement positif) à l'aide d'algorithmes de minimisation directe [14].

Les fonctions de barrière les plus répandues sont les suivantes :

La logarithmique:

$$B(x) = -\sum_{i=1}^m \ln(-h_i(x)). \quad (\text{I-17})$$

L'inverse :

$$B(x) = -\sum_{i=1}^m \frac{1}{h_i(x)}. \quad (\text{I-18})$$

Il est important de noter que, si tous les $h_i(x)$ sont convexes, ces deux fonctions de barrière le sont également. La méthode de barrière est définie en introduisant la séquence de paramètres $\{\varepsilon^k\}, k = 0, 1, \dots$ avec $0 < \varepsilon^{k+1} < \varepsilon^k$ et $\varepsilon^k \rightarrow 0$ lorsque $k \rightarrow \infty$. Une itération de celle-ci consiste à déterminer :

$$x^k = \arg \min_{x \in X_I} \{f(x) + \varepsilon^k B(x)\}. \quad (\text{I-19})$$

Théorème I.3 [14] Tout point limite d'une séquence $\{x^k\}$ générée par une méthode de barrière est un minimum global du problème contraint original.

Les inconvénients :

Quelques inconvénients de cette méthode sont à noter tel que :

- Il n'existe pas de règle universelle permettant d'obtenir un bon choix de ε^0 .
- L'utilisateur d'une méthode de barrière sera souvent condamné à exécuter la méthode plusieurs fois avec différentes valeurs de ε^0 et x^0 jusqu'à obtenir une convergence satisfaisante.

I.4.2 Les méthodes de pénalité extérieure

Les méthodes de pénalité extérieures cherchent à approcher le minimum depuis l'extérieur de X . A la fonction $f(x)$ est ajoutée une fonction de pénalité extérieure $P(x)$ dont la valeur est égale à zéro si x est admissible et supérieure à zéro s'il ne l'est pas. L'ajout de cette fonction a pour seul but de pénaliser la fonction objectif en cas de violation d'une ou de plusieurs contraintes [14]. La fonction de pénalité quadratique $P(x)$ est définie comme suit:

$$P(x) = \sum_{i=1}^m (h_i(x))^2 u_i(h_i(x)) \quad (\text{I-20})$$

$$\text{où } u_i(h_i(x)) = \begin{cases} 0 & \text{si } h_i(x) \leq 0, \\ 1 & \text{si } h_i(x) > 0, \end{cases}$$

De façon analogue à la méthode de barrière, nous introduisons un paramètre μ qui permet d'amplifier ou de diminuer sa valeur et, à chaque itération, le sous-problème à résoudre à l'aide d'une méthode de minimisation directe sera de la forme suivante :

minimiser $f(x) + \mu P(x)$ avec $x \in \mathbb{R}^n$. A chaque itération, la méthode obtient x^k de la manière suivante :

$$x^k = \arg \min_{x \in \mathbb{R}^n} \{f(x) + \mu P(x)\} \quad (\text{I-21})$$

L'inconvénient, c'est qu'il n'existe pas de règle universelle permettant d'obtenir un bon choix de μ^0 .

Le principal avantage c'est que le comportement de la méthode de pénalité extérieure est très similaire à celui de la méthode de barrière.

I.5 Classement des méthodes numériques d'optimisation

Sachant que ces méthodes, en général, exécutent une recherche itérative de la solution optimale, nous nous contentons de proposer un classement basé sur le principe de cette recherche, ainsi nous distinguons deux groupes de méthodes:

- Méthodes déterministes
- Méthodes stochastiques

La recherche menée par les méthodes déterministes est guidée par des critères mathématiques et analytiques, tandis que pour les méthodes stochastiques, une partie de la recherche est conduite de façon aléatoire.

I.6 Méthodes d'optimisation déterministes

Une méthode d'optimisation est dite déterministe lorsque son évolution vers la solution du problème est toujours la même pour un même contexte initial donné, ne laissant aucune place au hasard. Ce sont en général des méthodes efficaces, peu coûteuses, mais qui nécessitent une configuration initiale (point de départ) pour résoudre le problème. Ce sont souvent des méthodes locales, c'est-à-dire qu'elles convergent vers l'optimum le plus proche du point de départ, qu'il soit local ou global.

Nous pouvons diviser ces méthodes, qu'elles soient directes ou indirectes, en deux catégories: les méthodes de descente et les méthodes géométriques [15].

Le principe des méthodes de descente se base sur la connaissance d'une direction de recherche généralement donnée par le gradient de la fonction. Les exemples les plus significatifs des méthodes de descente sont la méthode du gradient conjugué [16, 17], et les méthodes de Quasi-Newton [18].

Les méthodes géométriques explorent l'espace par essais successifs en recherchant les directions les plus favorables. La méthode géométrique la plus connue est la méthode du simplexe [19].

Dans cette partie on va présenter uniquement les deux méthodes utilisées pour la résolution de notre cas: la méthode des gradients conjugués et la programmation quadratique séquentielle.

I.6.1 Méthode des gradients conjugués (GC)

a. Cas d'une fonction quadratique

Soit $q(x) = \frac{1}{2}x^tAx + b^tx + c$ une fonction quadratique, où A est définie positive.

La méthode consiste, à partir d'un point x^0 , à minimiser q suivant n directions d^0, d^1, \dots, d^{n-1} mutuellement conjuguées par rapport à A, c'est-à-dire vérifiant :

$$\text{Pour } 1 \leq i \leq j \leq n, \quad (d^i)^t Ad^j = 0 \quad (\text{I-22})$$

Soient n telles directions : d^0, d^1, \dots, d^{n-1}

Ayant déterminé x^k , le point x^{k+1} est le point :

$$x^{k+1} = x^k + s_k d^k \quad (\text{I-23})$$

Où s^k est choisi de façon à minimiser $q(x^k + s_k d^k)$

$$\text{On a donc : } (d^k)^t \nabla q(x^k + s_k d^k) = 0 \text{ ou encore : } (d^k)^t [A(x^k + s_k d^k) + b] = 0$$

D'où l'on déduit :

$$s_k = - \frac{(d^k)^t (Ax^k + b)}{(d^k)^t Ad^k} \quad (\text{I-24})$$

Si d^0, d^1, \dots, d^{k-1} sont mutuellement conjuguées, alors on a pour tout $i < k$ la relation :

$$(d^i)^t \nabla q(x^k) = 0.$$

Le point x^n est l'optimum de $q(x)$ sur \mathbb{R}^n [20]

La méthode de Fletcher et Reeves [21] engendre au fur et à mesure les directions d^i : on l'explique ci-dessous en posant :

$$g(k) = \nabla q(x^k) = Ax^k + b \quad (\text{I-25})$$

Choisir un point de départ x^0 , poser $d^0 = -g^0$

Pour k variant de 0 à n faire :

$$s_k \leftarrow -\frac{(d^k)^t g^k}{(d^k)^t A d^k}$$

$$x^{k+1} \leftarrow x^k + s_k d^k$$

$$b_k \leftarrow -\frac{(g^{k+1})^t A d^k}{(d^k)^t A d^k}$$

$$d^{k+1} \leftarrow -g^{k+1} + b_k d^k$$

b. Cas d'une fonction quelconque

L'algorithme de Fletcher et Reeves pour une fonction quelconque est le suivant :

Partir d'un point x^0 ,

Faire $d^0 \leftarrow -\nabla f(x^0)$ et $k \leftarrow 0$

Répéter

Choisir s_k minimisant $f(x^k + s d^k)$, par rapport à s

$$x^{k+1} \leftarrow x^k + s_k d^k$$

$$b_k \leftarrow -\frac{\|\nabla f(x^{k+1})\|^2}{\|\nabla f(x^k)\|^2}$$

$$d^{k+1} \leftarrow -\nabla f(x^{k+1}) + b_k d^k$$

$$k \leftarrow k + 1$$

Jusqu'à ce qu'un test d'arrêt soit vérifié.

Cette méthode a deux avantages : elle nécessite le stockage de très peu d'informations et sa vitesse de convergence est très supérieure à celle des algorithmes de gradient classiques.

I.6.2 La programmation quadratique séquentielle (SQP)

La programmation quadratique séquentielle (SQP) est utilisée pour la résolution des problèmes généraux d'optimisation non linéaire avec contraintes.

Les problèmes à résoudre avec cette méthode sont de la forme :

$$\left\{ \begin{array}{l} \min_{x \in \mathbb{R}^n} f(x) \\ \text{sous } g_i(x) = 0, \text{ pour } i = 1, \dots, n \\ h_j(x) \geq 0, \text{ pour } j = n+1, \dots, m \\ x^{\text{inf}} \leq x \leq x^{\text{sup}} \end{array} \right. \quad (\text{I-26})$$

Où toutes les fonctions du problème sont continues différentiables.

La programmation quadratique séquentielle basée sur la formulation itérative et la solution de sous problème de la programmation quadratique, on obtient le sous problème en utilisant une approximation quadratique de Lagrangien et par linéarisation des contraintes [22].

Alors on peut écrire le sous problème sous la forme :

$$\left\{ \begin{array}{l} \min_{d \in \mathbb{R}^n} \frac{1}{2} d^T B_k d + \nabla f(x^k)^T d \\ \text{sous } \nabla g_i(x^k)^T d + g_i(x^k) = 0, \quad i = 1, \dots, n \\ \nabla h_j(x^k)^T d + h_j(x^k) \geq 0, \quad j = n+1, \dots, m \\ x^{\text{inf}} - x^k \leq d \leq x^{\text{sup}} - x^k \end{array} \right. \quad (\text{I-27})$$

Où B_k est une approximation de l'hessien définie positive. On utilise la recherche linéaire pour trouver le nouveau point x^{k+1} .

$$x^{k+1} = x^k + \alpha d_k, \quad \alpha \in]0,1] \quad (\text{I-28})$$

Tel qu'une "fonction modale" aura une valeur de la fonction inférieure au nouveau point. Ici, la fonction de Lagrange est utilisée comme une fonction modale.

Lorsque l'optimalité n'est pas accompli, B_k est mis à jour d'après la formule Broyden, Fletcher, Goldfarb et Shanno (BFGS).

La formule de mise à jour de BFGS est une formule de correction de rang 2. Cette formule permet de mettre à jour une approximation H_k du hessien possédant les mêmes propriétés, à savoir $H_{k+1} > 0$ si $H_k > 0$ et vérifiant la relation de quasi-Newton :

$$y_k = H_k s_k.$$

La formule est donc la suivante :

$$H_{k+1} = H_k + \frac{y_k y_k^T}{y_k^T s_k} - \frac{H_k s_k s_k^T H_k}{s_k^T H_k s_k}.$$

$$\text{Avec } \begin{cases} s_k = B_{k+1} y_k, \\ y_k = \nabla f(x^{k+1}) - \nabla f(x^k). \end{cases}$$

Les avantages :

- avec la programmation quadratique on peut résoudre tous les types d'optimisation non linéaire avec contraintes (quadratique ou non quadratique), et quelque soit le type de contraintes (contraintes d'égalité ou contraintes d'inégalité).
- Cette méthode utilise la formule BFGS pour calculer B_k (la remise à jour de type BFGS ne nécessite que la manipulation et la mémorisation de vecteurs de dimension n).
- L'utilisation de remise à jour de type BFGS peut accélérer le processus de convergence.
- Cette méthode est bénéficiée d'une convergence quadratique.
- Cette méthode utilise une recherche linéaire pour calculer un pas d'optimisation.

I.7 Méthodes stochastiques

Les méthodes d'optimisation stochastiques s'appuient sur des mécanismes de transition probabilistes et aléatoires. Cette caractéristique indique que plusieurs exécutions successives de ces méthodes peuvent conduire à des résultats différents pour une même configuration initiale d'un problème d'optimisation.

Ces méthodes ont généralement une capacité plus forte à trouver l'optimum global du problème. Contrairement à la plupart des méthodes déterministes, elles ne nécessitent ni la définition d'un point de départ, ni la connaissance du gradient de la fonction objectif pour atteindre la solution optimale. Cependant, elles demandent un nombre important d'évaluations avant d'arriver à la solution du problème.

Du même que dans la partie des méthodes déterministes, dans cette partie on va présenter uniquement les deux méthodes utilisées pour la résolution du notre problème : l'algorithme génétique (GA) et la méthode d'essaim particulaire (PSO)

I.7.1 L'algorithme génétique (GA)

Le principe de calcul des algorithmes génétique est inspiré des principes évolutifs darwiniens : les espèces s'adaptent – de génération en génération – à leur environnement par l'acquisition, la sélection et la conservation de certaines caractéristiques [23].

Avant tout, nous allons définir la terminologie utilisée dans la description de la méthode numérique :

- Un individu représente une solution potentielle (candidate) du problème d'optimisation.
- Un gène est la valeur codée d'une seule variable de contrôle
- Un chromosome (ensemble de gènes) représente numériquement le vecteur de contrôle.
- Une population est formée par l'ensemble des solutions candidates
- La génération est le passage d'une population à une autre
- Les parents sont les individus sélectionnés pour former une nouvelle génération.

a. Principe de l'algorithme génétique

Le but des algorithmes génétiques est de déterminer les extrêmes d'une fonction $f : X \rightarrow \mathbb{R}$, où X est un ensemble quelconque appelé espace de recherche et f est appelée fonction d'adaptation ou fonction d'évaluation ou encore fonction fitness [24].

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

1. Un principe de codage de l'élément de population.
2. Un mécanisme de génération de la population initiale.
3. Une fonction à optimiser. Celle-ci retourne une valeur appelée fitness ou fonction d'évaluation de l'individu.
4. Des opérateurs permettant de diversifier la population au cours des générations et d'explorer l'espace d'état.
5. Des paramètres de dimensionnement : taille de la population, nombre total de générations ou critère d'arrêt, probabilités d'application des opérateurs de croisement et de mutation.

b. Description de l'algorithme génétique

Codages

Premièrement, il faut représenter les différents états possibles de la variable dont on cherche la valeur optimale sous une forme utilisable par un GA : c'est le codage. Cela

permet d'établir une connexion entre les valeurs de la variable et les individus de la population de manière à imiter la connexion qui existe en biologie entre le génotype et le phénotype. Il existe principalement deux types de codage : le codage binaire et le codage réel [25].

Codage binaire

C'est la méthode choisie par le créateur des algorithmes génétiques (Holland) et développée ensuite par son élève (Goldberg) sous le nom de "Théorie des Schémas". Cette théorie suggère de prendre les alphabets les plus réduits possibles pour coder le problème, donc par exemple un codage binaire pour coder des entiers ou des réels. Ce codage permet de faciliter la conception du croisement entre les individus.

L'inconvénient majeur de ce codage est qu'une petite variation consistant à changer le bit de poids fort correspond à une modification importante de la valeur entière qui est codée (quand le codage est de type binaire naturel).

Codage réel

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

Génération de la population initiale

Ce mécanisme doit être capable de produire une population d'individus non homogène qui servira de base pour les générations futures. Le choix de la population initiale est important car il peut rendre plus ou moins rapide la convergence vers l'optimum global. Dans le cas où l'on ne connaît rien du problème à résoudre, il est essentiel que la population initiale soit répartie sur tout le domaine de recherche [24].

Opérateur de croisement

L'opérateur de croisement permet la création de nouveaux individus selon un processus fort simple. Il permet donc l'échange d'information entre les chromosomes (individus). Tout d'abord, deux individus, qui forment alors un couple, sont tirés au sein de la nouvelle population issue de la reproduction. Enfin, selon une probabilité P_c que le croisement s'effectue, les segments finaux (dans le cas d'un seul site de croisement) des deux parents sont alors échangés autour de ce site [26].

- Le croisement double : est plus généralement à plusieurs sites, il permet d'échanger des morceaux de chromosomes compris entre deux sites de croisement.

- Le croisement moyen : il propose de calculer la moyenne de certains gènes choisis au hasard chez les deux parents.
- Le croisement multi parental : il consiste à combiner les gènes de plus de deux parents en utilisant par exemple la méthode du simplexe pour orienter la recombinaison.

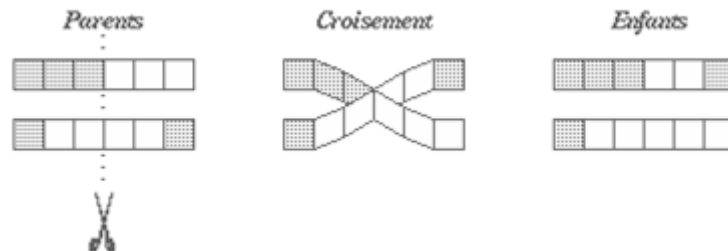


Figure I.3 Le croisement [26]

Opérateur de mutation

Il permet de transformer au hasard le codage d'un individu afin d'apporter une certaine diversité dans la population et empêcher que celle-ci converge trop vite vers un seul type d'individu parfait, incapable de sortir d'un minimum local. La mutation est réalisée en modifiant un gène d'un individu pris au hasard. L'amplitude de la modification peut elle même être déterminée aléatoirement où correspondre à une valeur fixée [26].

Très couramment, la mutation consiste à rajouter + 1 ou - 1 à la valeur d'un gène, suivant une probabilité de mutation qui est en général assez faible : comprise entre 0,001 et 0,005.

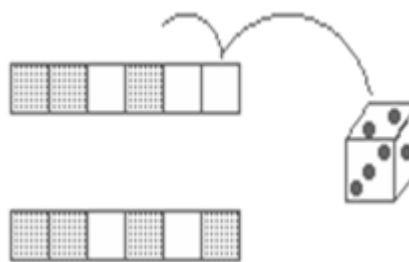


Figure I.4 La mutation [26]

Principes de sélection

La sélection s'effectue de différentes façons : Généralement elle est réalisée à partir d'un tirage probabiliste basé sur le principe de la "Roue de la fortune" découpée en secteurs. Chaque secteur correspond à un individu de la population, la superficie de chaque secteur étant proportionnelle à la réponse de la fonction objective : plus un individu est

adapté, plus le secteur qui lui correspond est grand. C'est la méthode préconisée par (Holland) et (Goldberg) et c'est aussi la plus employée.

On obtient donc une roue de la fortune découpée en 4 secteurs (pourcentages arrondis à la valeur entière) : Pour réaliser la sélection, il suffit de faire "tourner" la roue de la fortune autant de fois qu'il y a d'individus dans une population. Comme on le voit sur l'exemple (figure I.5), l'individu 3 qui a la meilleure fonction objective, se voit attribuer une portion plus importante de la roue. Si la loi de distribution est répartie uniformément de 0 à 1, chaque tirage à une probabilité de 0,303 de tomber sur le secteur qui correspond à l'individu 3. De cette façon, les individus bien adaptés ont un plus grand nombre de chances d'être utilisés pour la reproduction de la génération suivante.

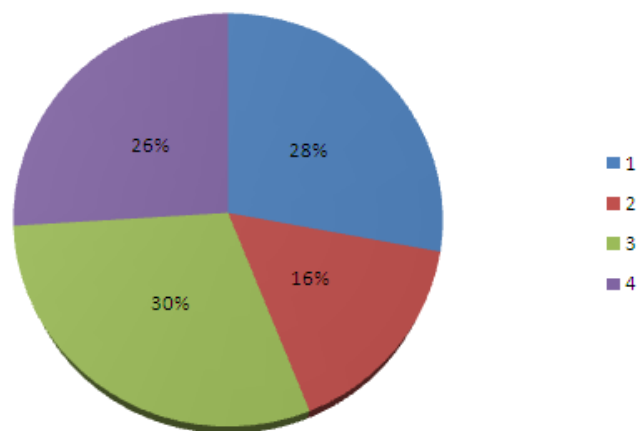


Figure I.5 Sélection par roue de la fortune

c. Autres paramètres de l'algorithme génétique

Les opérateurs de l'algorithme génétique sont guidés par un certain nombre de paramètres fixés à l'avance. La valeur de ces paramètres influence la réussite ou non d'un algorithme génétique. Ces paramètres sont les suivants :

- La taille de la population N , et la longueur du codage de chaque individu l . Si N est trop grand le temps de calcul de l'algorithme peut s'avérer très important, et si N est trop petit, il peut converger trop rapidement vers un mauvais chromosome.
- La probabilité de croisement P_c : elle dépend de la forme de la fonction de fitness. Son choix est en général heuristique. Plus elle est élevée, plus la population subit de changements importants. Les valeurs généralement admises sont comprises entre 0.5 et 0.9.

- La probabilité de mutation P_m : Ce taux est généralement faible puisqu'un taux élevé risque de conduire à une solution sous-optimale. Plutôt que de réduire P_m , une autre façon d'éviter que les meilleurs individus soient altérés est d'utiliser la reconduite explicite de l'élite dans une certaine proportion [27].

I.7.2 La méthode d'essaim particulaire (PSO)

L'optimisation par essaim de particule (OEP) ou bien (PSO : Particle swarm optimization), a été inventée par Russel Eberhart et James Kennedy [28, 29] en 1995. Au départ J. Kennedy et R. Eberhart cherchaient à simuler la capacité des oiseaux à voler de façon synchrone et leur aptitude à changer brusquement de direction tout en restant en une formation optimale. Le modèle qu'ils ont proposé à ensuite été étendu en un algorithme simple et efficace d'optimisation.

L'optimisation par essaim de particule est une technique utilisée pour explorer l'espace de recherche d'un problème quelconque pour trouver l'ensemble des paramètres qui maximise/minimise un objectif particulier. Cet objectif est atteint en suivant un algorithme dédié que l'on verra par la suite.

a. Algorithme d'essaim particulaire

Chaque particule représente une solution potentielle dans l'espace de recherche. La nouvelle position d'une particule est déterminée en fonction de sa propre valeur et celle de ses voisines. Soit $\vec{x}_i(t)$ la position de la particule P_i au temps t , sa position est modifiée en ajoutant une vitesse $\vec{v}_i(t)$ à sa position courante :

$$\vec{x}_i(t) = \vec{x}_i(t-1) + \vec{v}_i(t) \quad (\text{I-29})$$

La vitesse de chaque particule est mise à jour suivant l'équation suivante:

$$v_i(t+1) = \omega v_i(t) + c_1 r_1 [xp_i(t) - x_i(t)] + c_2 r_2 [g(t) - x_i(t)] \quad (\text{I-30})$$

$v_i(t)$ est la vitesse de particule i à l'instant t et $x_i(t)$ est la position de particule i à l'instant t , les paramètres ω , c_1 , et c_2 ($0 \leq \omega \leq 1.2$, $0 \leq c_1 \leq 2$, et $0 \leq c_2 \leq 2$) sont des coefficients constants fixés par l'utilisateur, r_1 et r_2 sont des nombres aléatoires tirés à chaque itération, $g(t)$ est la meilleure solution trouvée jusqu'à l'instant t et $xp_i(t)$ est la meilleure solution trouvée par la particule i . C'est le vecteur vitesse qui dirige le processus de recherche et reflète la "sociabilité" des particules.

b. Les variables et paramètres de l'algorithme :

N : Nombre de particules

$\vec{x}_i(t)$: Position de la particule P_i , $\vec{v}_i(t)$: vitesse de la particule P_i

$pbest_i$: Meilleure fitness obtenue pour la particule P_i

\vec{x}_{pbest_i} : Position de la particule P_i pour la meilleure fitness

\vec{x}_{gbest_i} : Position de la particule ayant la meilleure fitness de toutes

ρ_1 et ρ_2 : Valeurs aléatoires positives

c. Initialisations :

Initialiser aléatoirement la population.

d. Traitement :**Répéter**

Pour i de 1 à N : faire

Si $(F(\vec{x}_i) > pbest_i)$ Alors

$pbest_i \leftarrow F(\vec{x}_i)$

$\vec{x}_{pbest_i} \leftarrow \vec{x}_i$

Fin Si

Si $(F(\vec{x}_i) > gbest_i)$ Alors

$gbest_i \leftarrow F(\vec{x}_i)$

$\vec{x}_{gbest_i} \leftarrow \vec{x}_i$

Fin Si

Fin Pour

Pour i de 1 à N : faire

$\vec{v}_i \leftarrow \vec{v}_i + c_1\rho_1(\vec{x}_{pbest_i} - \vec{x}_i) + c_2\rho_2(\vec{x}_{gbest_i} - \vec{x}_i)$

$\vec{x}_i \leftarrow \vec{x}_i + \vec{v}_i$

Fin Pour

Jusqu'à ce que (le processus converge)

La variation de la vitesse est proportionnelle à l'éloignement d'une solution par rapport à la solution globale.

Les variables aléatoires ρ_1 et ρ_2 peuvent être définies de la façon suivante :

$$\begin{cases} \rho_1 = r_1 c_1 \\ \rho_2 = r_2 c_2 \end{cases}$$

r_1 et r_2 suivent une loi uniforme sur $[0..1]$ et c_1 et c_2 sont des constantes et représentent une accélération positive, avec $c_1 + c_2 < 4$.

Le critère de convergence peut être un nombre fixe d'itérations, suivant la fitness ou bien la variation lorsqu'elle tend vers 0.

On remarque qu'il y a six paramètres qui rentrent en ligne de compte :

- La dimension du problème.
- Le nombre de particules.
- Les valeurs des coefficients ρ .
- La taille du voisinage.
- La vitesse maximale.
- L'inertie.

La vitesse peut être limitée par une valeur maximale V_{\max} et une autre minimale V_{\min} pour éviter le déplacement trop rapide ou trop long des particules d'une région à une autre dans l'espace de recherche.

Un facteur d'inertie ϕ peut être appliqué à la vitesse pour contrôler l'influence de celle-ci.

I.8 Conclusion

Dans ce chapitre, nous avons présenté tout d'abord quelques définitions introductives de l'optimisation. Par la suite nous avons classé les approches d'optimisation suivant leur nature. Les méthodes déterministes sont, en général, peu coûteuses, mais elles nécessitent souvent un point de départ et le calcul du gradient de la fonction objectif pour résoudre le problème. Par ailleurs les méthodes stochastiques ne nécessitent ni connaissance point de départ, ni calcul du gradient pour atteindre la solution optimale, cependant elles demandent un nombre d'évaluations plus important avant d'arriver à la solution du problème.

Chapitre II
Etude Comparative des Techniques
d'Optimisation

II.1 Introduction

Dans l'objectif de compléter la démarche de modélisation d'un problème de production industrielle, nous proposons d'adjoindre un processus d'optimisation. Ils se posent alors deux problèmes, le choix de la méthode d'optimisation la mieux adaptée aux problématiques de production industrielle et le choix des critères de comparaison de telles méthodes.

Une méthode d'optimisation est un processus automatique permettant d'identifier l'optimum d'une fonction objectif. Lorsque cet optimum est atteint, on parle de « convergence » de la méthode. Contrairement à une méthode de balayage, qui consiste à évaluer toutes les possibilités afin d'établir la plus pertinente, une méthode d'optimisation identifie l'optimum en procédant via des opérations mathématiques ou logiques [1]. Storn prescrit les qualités d'une méthode d'optimisation efficiente [1]:

- Sa capacité à traiter des fonctions « objectifs » non-linéaires, non-dérivables et multi-modale (Une fonction multi-modale présente plusieurs optimaux locaux.),
- Son habilité à utiliser le principe de parallélisations,
- Sa facilité d'utilisation (compréhension du principe, nombre de paramètres réduit) ;
- Son bon taux de convergence.

Dans le chapitre précédent nous avons présenté quatre algorithmes d'optimisations qui sont :

- La méthode des gradients conjugués (GC)
- La programmation quadratique séquentielle (SQP)
- L'algorithme génétique (GA)
- La méthode d'essaim particulaire (PSO)

Dans ce chapitre ceux-ci ont été implémentés et sont comparés au travers de différentes fonctions-tests, aussi appelées "benchmarks". Plusieurs aspects de ces algorithmes sont étudiés pour choisir une méthode d'optimisation la plus fiable afin d'être appliqué au cas industriel présenté au troisième chapitre.

II.2 Enjeux de la comparaison

Si le choix entre méthodes déterministes et stochastiques est souvent évoqué, la plupart des travaux présentés dans la littérature considèrent que les méthodes stochastiques sont

semblables entre elles. Le choix de la méthode se trouve alors guidé par les connaissances propres de l'auteur, les habitudes de l'équipe dans laquelle sont menées les recherches ou, plus souvent, les méthodes les plus usitées dans la littérature scientifique.

La figure II.1 illustre bien cette constatation. Sur le nombre total de résultats, cette figure illustre la proportion de résultats pour chaque type d'algorithme. On constate que les algorithmes génétiques sont largement représentés dans l'ensemble des résultats obtenus [1].

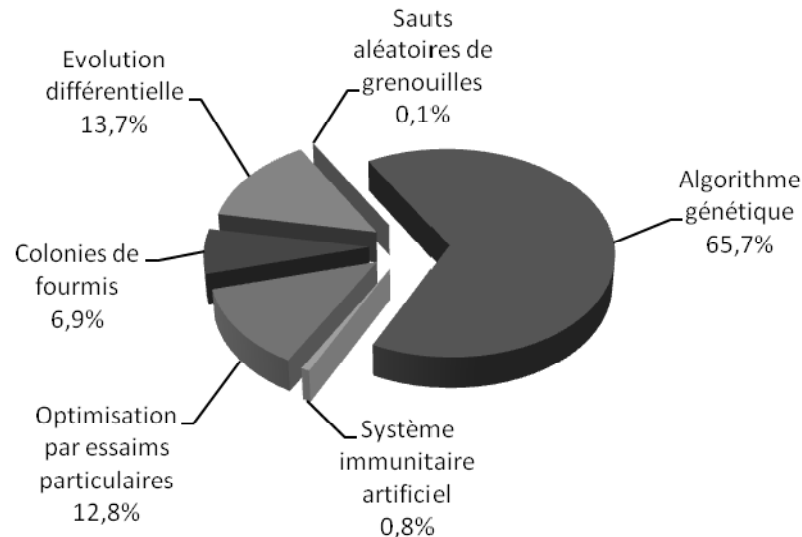


Figure II.1 Taux d'utilisation de six algorithmes stochastiques [1]

Il est cependant légitime d'étudier les différentes caractéristiques de tels algorithmes afin de les discriminer, et d'identifier celui qui sera le mieux adapté à la démarche du problème que nous proposons.

II.3 Fonctions tests

Afin de comparer les algorithmes présentés dans le chapitre I, il est nécessaire de les confronter à des problèmes identiques, pour ensuite estimer leurs performances suivant différents critères. Des "fonctions-tests", aussi appelées "benchmarks", doivent être définies afin de constituer un socle de comparaison commun aux différents algorithmes d'optimisation.

Nous proposons de détailler ci-dessous les cinq fonctions-tests [30] que nous employons pour comparer les algorithmes. L'objectif est de trouver le minimum global de chacune d'entre elles.

II.3.1 Fonction de Griewangk's

$$f(x) = \frac{1}{4000} \sum_{i=1}^d x_i^2 - \prod_{i=1}^d \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1, \quad x_i \in [-600, 600] \quad (\text{II-1})$$

Dont le minimum global est $f^* = 0$ pour $x^* = (0, 0, \dots, 0)$

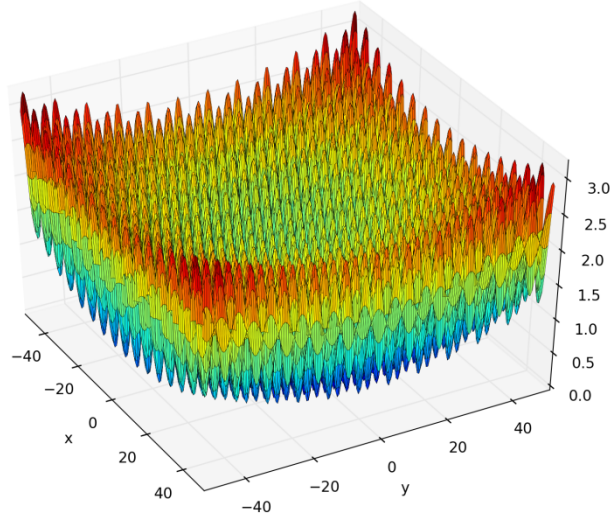


Figure II.2 Fonction de Griewangk's

II.3.2 Fonction de Ackley's

$$f(x) = -20 \exp\left[-0.2 \sqrt{\frac{1}{d} \sum_{i=1}^d x_i^2}\right] - \exp\left[\frac{1}{d} \sum_{i=1}^d \cos(2\pi x_i) + (20 + e)\right] \quad (\text{II-2})$$

$$x_i \in [-32.768, 32.768]$$

Dont le minimum global est $f^* = 0$ pour $x^* = (0, 0, \dots, 0)$

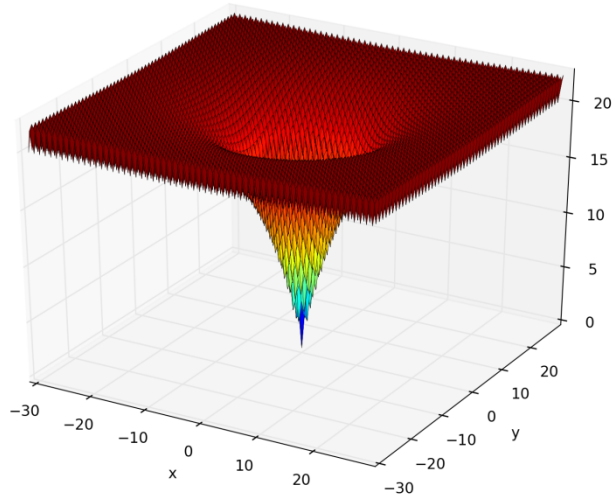


Figure II.3 Fonction de Ackley's

II.3.3 Fonction de Rosenbrock's

$$f(x) = \sum_{i=1}^{d-1} \left[(1 - x_i)^2 + 100(x_{i+1} - x_i^2)^2 \right] \quad (\text{II-3})$$

Dont le minimum global est $f^* = 0$ pour $x^* = (1, 1, \dots, 1)$

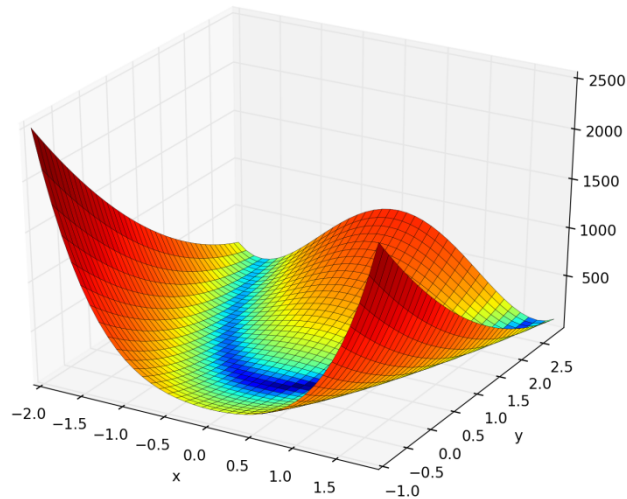


Figure II.4 Fonction de Rosenbrock's

II.3.4 Fonction de Schwefel's

$$f(x) = \sum_{i=1}^d \left[-x_i \sin(\sqrt{|x_i|}) \right], \quad x_i \in [-500, 500] \quad (\text{II-4})$$

Dont le minimum global est $f^* = -418.9829d$ à $x^* = 420.9687$

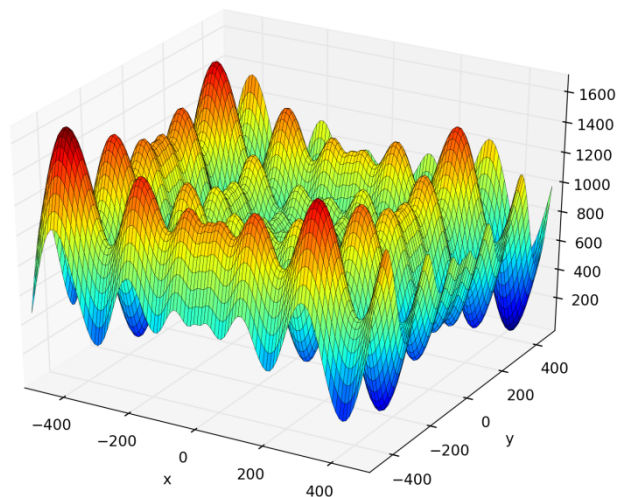


Figure II.5 Fonction de Schwefel's

II.3.5 Fonction de Rastrigin's

$$f(x) = 10d + \sum_{i=1}^d [x_i^2 - 10 \cos(2\pi x_i)], \quad x_i = [-5.12, 5.12] \quad (\text{II-5})$$

Dont le minimum global est $f^* = 0$ à $x^* = (0, 0, \dots, 0)$

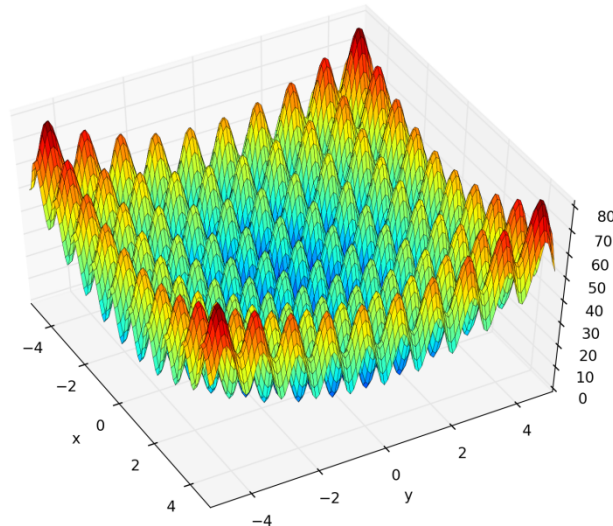


Figure II.6 Fonction de Rastrigin's

II.4 Critères d'arrêt

Le critère d'arrêt est la condition ou l'ensemble de conditions qui mène à un arrêt programmé de l'algorithme.

On trouve généralement trois types de critères d'arrêt, utilisés conjointement ou non [1] :

- L'atteinte d'une valeur cible pour la fonction objectif,
- Un nombre d'évaluations ou d'itérations limité,
- Une absence d'amélioration de la valeur de la fonction objectif pour la meilleure solution sur plusieurs itérations consécutives.

Dans ce chapitre et pour les méthodes déterministes nous avons choisi comme critère d'arrêt le nombre d'évaluations maximal égal à 3000 et le nombre d'itérations maximal égal à 1000 (le critère d'arrêt est atteint si l'évaluation de la meilleure solution d'un ensemble donné est inférieure à 10^{-6}).

Pour les méthodes stochastiques nous avons choisi comme critère d'arrêt le nombre de générations maximal égal à 1000 et le nombre d'évaluations maximal égal à 100 000.

II.5 Paramétrage des algorithmes stochastique

Pour réaliser la comparaison entre algorithmes via les benchmarks précédemment décrits, il n'est pas judicieux de fixer arbitrairement les paramètres de chaque algorithme et

de les conserver pour chaque benchmark. Vue la difficulté de détermination du meilleur jeu de paramètres, et il n'est pas souhaitable de s'aider d'un benchmark unique pour déterminer ces paramètres, tant que les fonctions-tests sont différentes [1]. C'est pourquoi nous proposons de déterminer le meilleur jeu de paramètres pour chaque couple algorithme/benchmark.

Pour cela, nous avons réalisé un balayage de toutes les combinaisons possibles de paramètres pour un couple donné.

II.5.1 Balayage pour chaque couple algorithme/benchmark

Pour paramètre 1 = [valeur 11, valeur 12, valeur 13]

Pour paramètre 2 = [valeur 21, valeur 22, valeur 23]

Pour paramètre 3 = [valeur 31, valeur 32, valeur 33]

Pour Exécuter = [1, 2, 3]

Résultat (Exécution) \leftarrow algorithme (paramètre 1, paramètre 2, paramètre 3)

End Pour

$r \leftarrow r + 1$

Bilan(r) \leftarrow moyenne (résultat)

End Pour

End Pour

End Pour

II.6 Organigrammes des méthodes d'optimisation utilisées

Nous présentons ci-dessous les organigrammes des quatre méthodes d'optimisation utilisées.

II.6.1 La méthode du gradient conjuguée

L'organigramme suivant (page 33) présente la méthode du gradient conjuguée

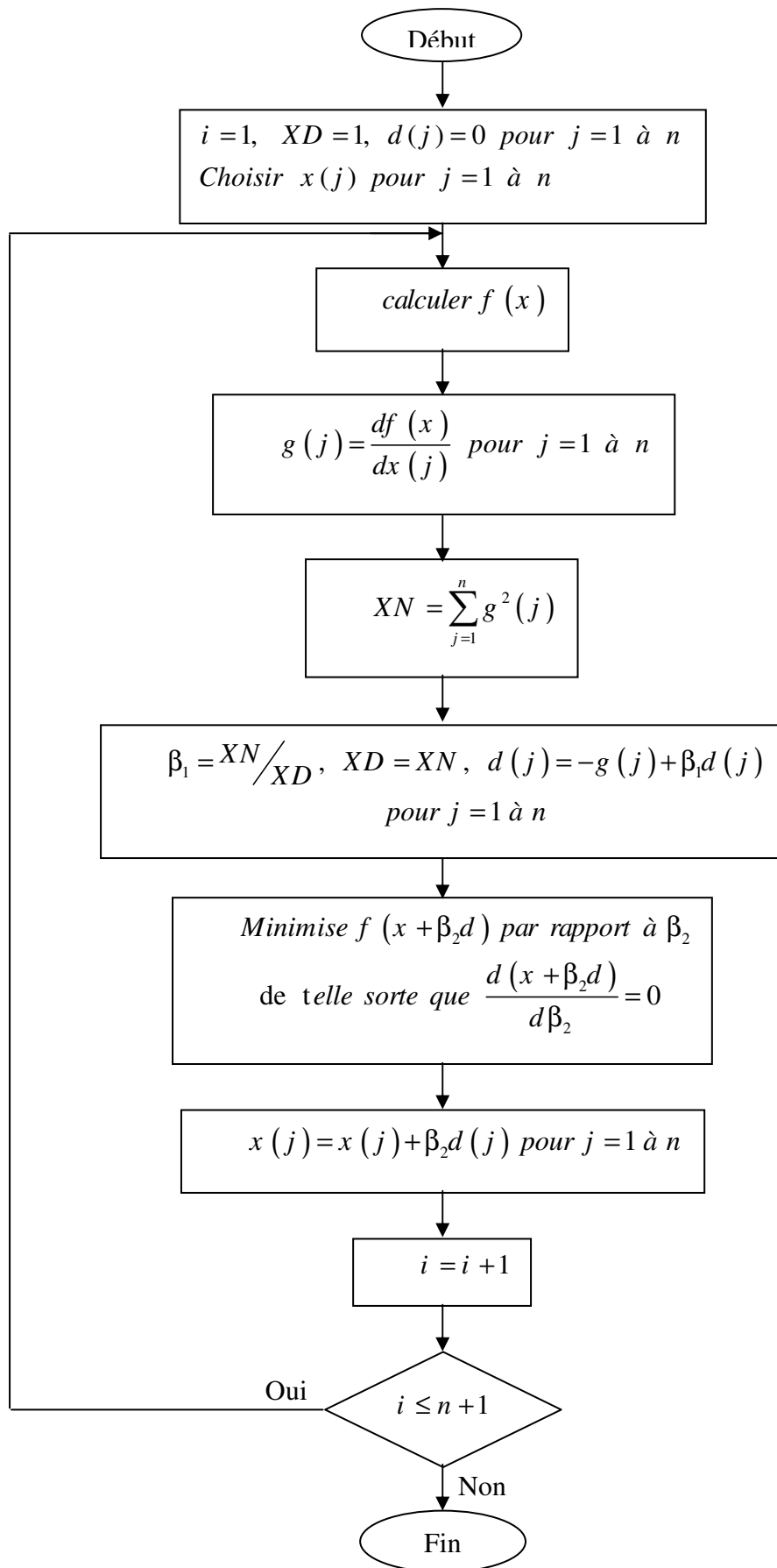


Figure II.7 Organigramme de la méthode du gradient conjuguée

II.6.2 La programmation quadratique séquentielle

L'organigramme suivant présente la programmation quadratique séquentielle

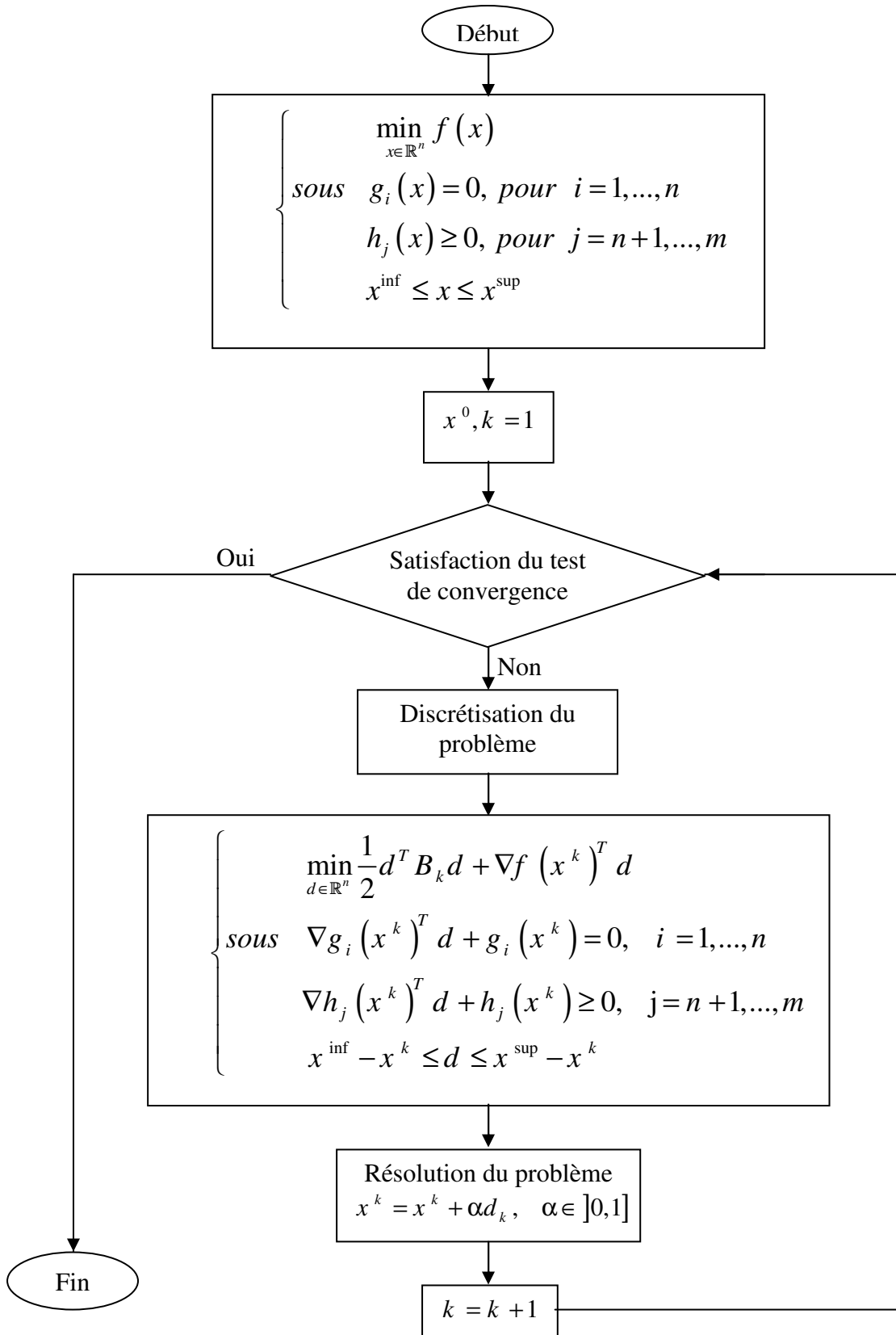


Figure II.8 Organigramme de la programmation quadratique séquentielle

II.6.3 L'algorithme génétique

L'organigramme suivant présente l'algorithme génétique

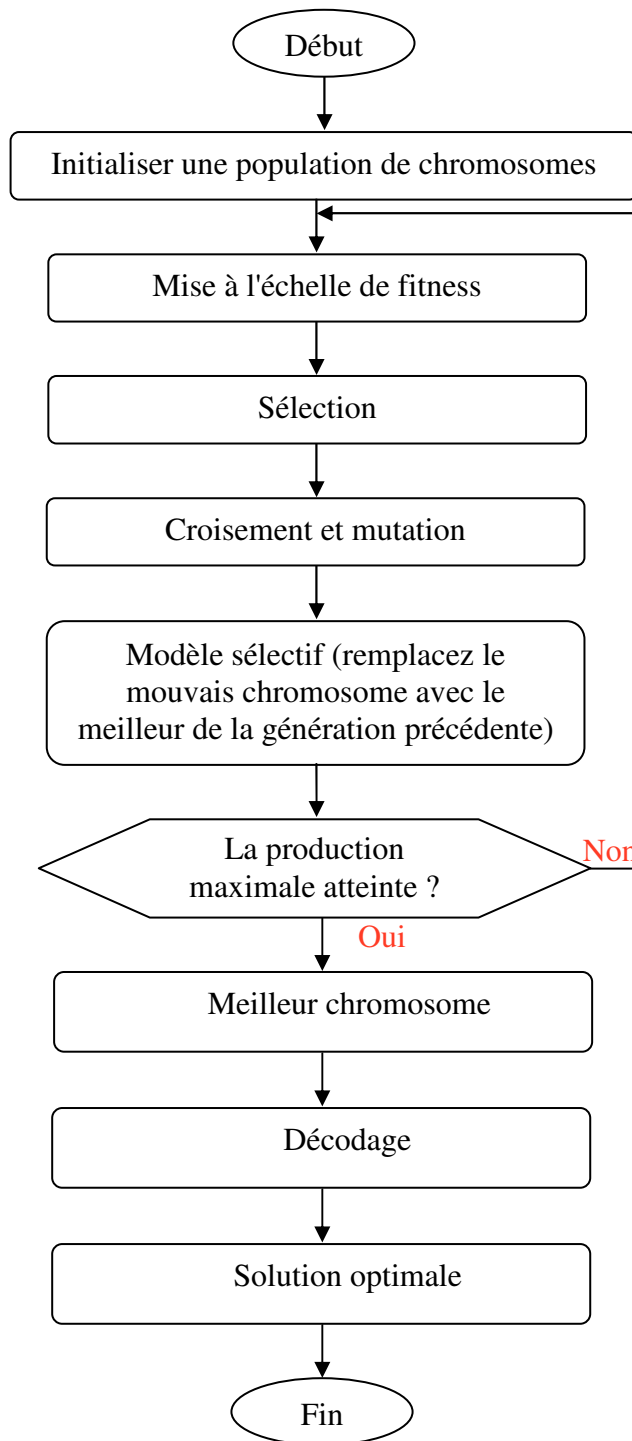


Figure II.9 Organigramme de l'algorithme génétique

II.6.4 La méthode d'essai particulaire

L'organigramme suivant présente la méthode d'essai particulaire

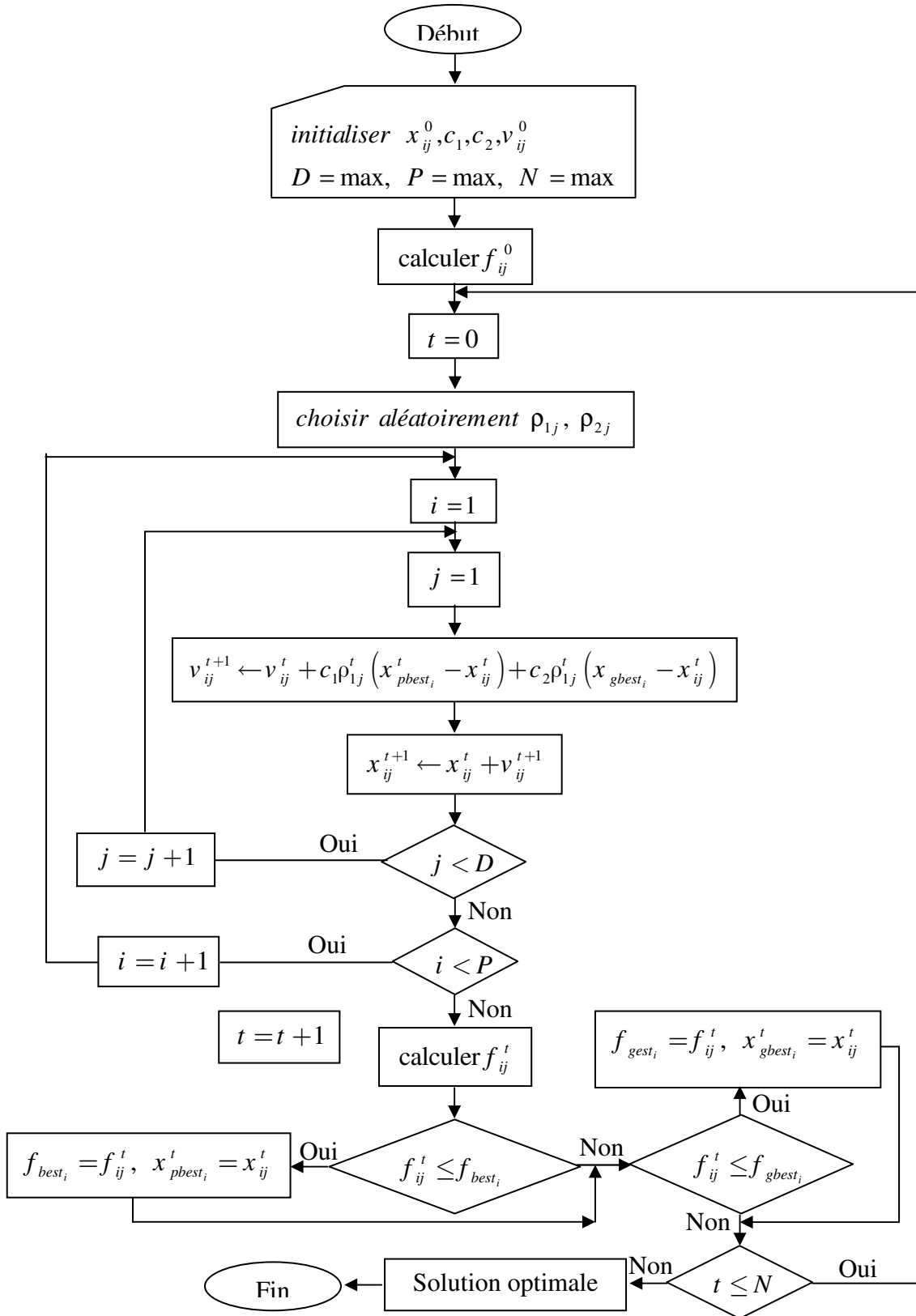


Figure II.10 Organigramme de la méthode d'essai particulaire [31]

II.7 Résultats

II.7.1 Méthodes déterministes

a. Fonction de Griewangk's

Tableau II.1 Résultats de minimisation de la fonction Griewangk's

| pt. initial | pt. final GC | pt. final SQP |
|-------------|--------------|---------------|
| (0.0,0.0) | (0.0, 0.0) | (0.0, 0.0) |
| (3.0, 3.0) | (3.14, 6.27) | (3.14, 6.27) |
| (5.0, 2.0) | (6.28, 0) | (6.28, 0) |
| (8.0, 5.0) | (9.42, 6.27) | (9.42, 6.27) |

b. Fonction de Ackley's

Tableau II.2 Résultats de minimisation de la fonction Ackley's

| pt. initial | pt. final GC | pt. final SQP |
|-------------|---------------|---------------|
| (0.0,0.0) | (0.0, 0.0) | (0.0, 0.0) |
| (3.0, 3.0) | (1.99, 1.99) | (1.99, 1.99) |
| (5.0, 2.0) | (1.98, -0.99) | (1.98, -0.99) |
| (8.0, 5.0) | (7.99, 4.95) | (7.99, 4.95) |

c. Fonction de Rosenbrock's

Tableau II.3 Résultats de minimisation de la fonction Rosenbrock's

| pt. initial | pt. final GC | pt. final SQP |
|-------------|--------------|---------------|
| (0.0,0.0) | (1, 1) | (1.0, 1.0) |
| (3.0, 3.0) | (1, 1) | (1.0, 1.0) |
| (5.0, 2.0) | (1, 1) | (0.27, 0.06) |
| (8.0, 5.0) | (1, 1) | (0.07, 0) |

d. Fonction de Schwefel's

Tableau II.4 Résultats de minimisation de la fonction Schwefel's

| pt. initial | pt. final GC | pt. final SQP |
|---------------|-----------------|-----------------|
| (500.0,500.0) | (420.97, 42.97) | (420.97, 42.97) |
| (3.0, 3.0) | (5.24, 5.24) | (5.24, 5.24) |
| (5.0, 2.0) | (5.24, 5.24) | (5.24, 5.24) |
| (80.0, 50.0) | (65.55, 65.55) | (65.55, 65.55) |

e. Fonction de Rastrigin's

Tableau II.5 Résultats de minimisation de la fonction Rastrigin's

| pt. initial | pt. final GC | pt. final SQP |
|-------------|--------------|---------------|
| (0.0,0.0) | (0.0, 0.0) | (0.0, 0.0) |
| (3.0, 3.0) | (0.0, 0.0) | (0.0, 0.0) |
| (5.0, 2.0) | (0.0, 0.0) | (-2, -1) |
| (8.0, 5.0) | (0.0, 0.0) | (0.0,2.0) |

II.7.2 Méthodes stochastique

a. Fonction de Griewangk's

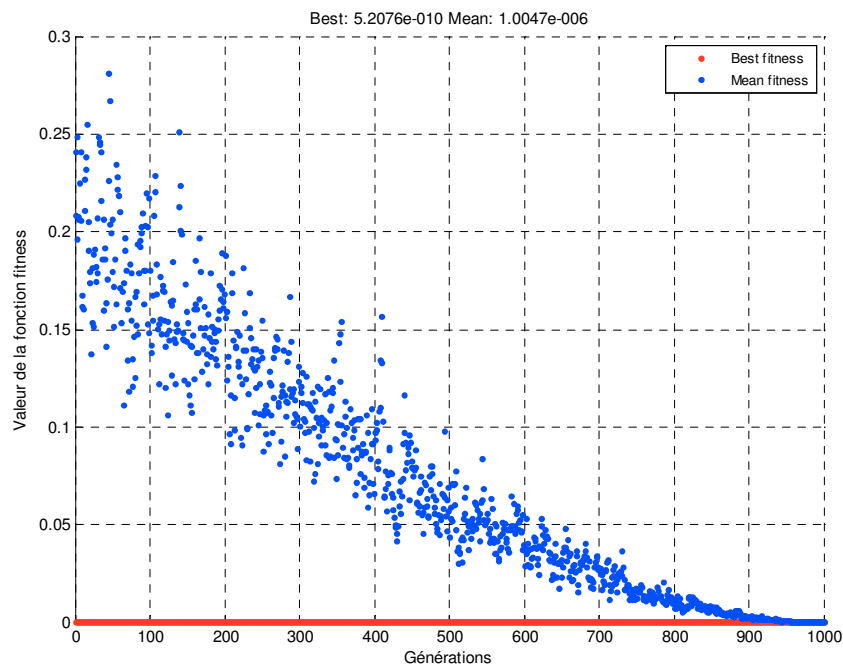


Figure II.11 Résultats de minimisation de la fonction Griewangk's avec l'algorithme génétique

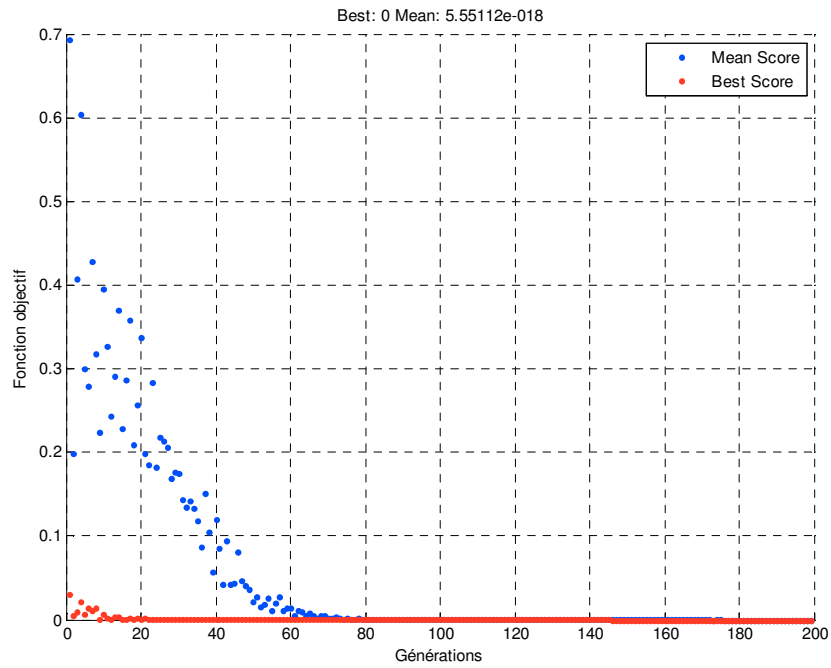


Figure II.12 Résultats de minimisation de la fonction Griewangk's avec l'algorithme PSO

b. Fonction de Ackley's

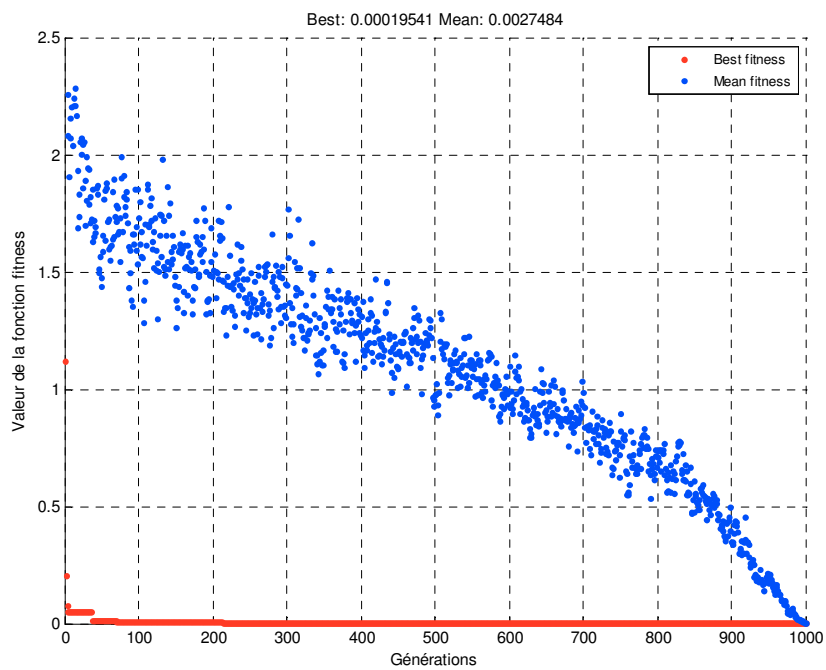


Figure II.13 Résultats de minimisation de la fonction Ackley's avec l'algorithme génétique

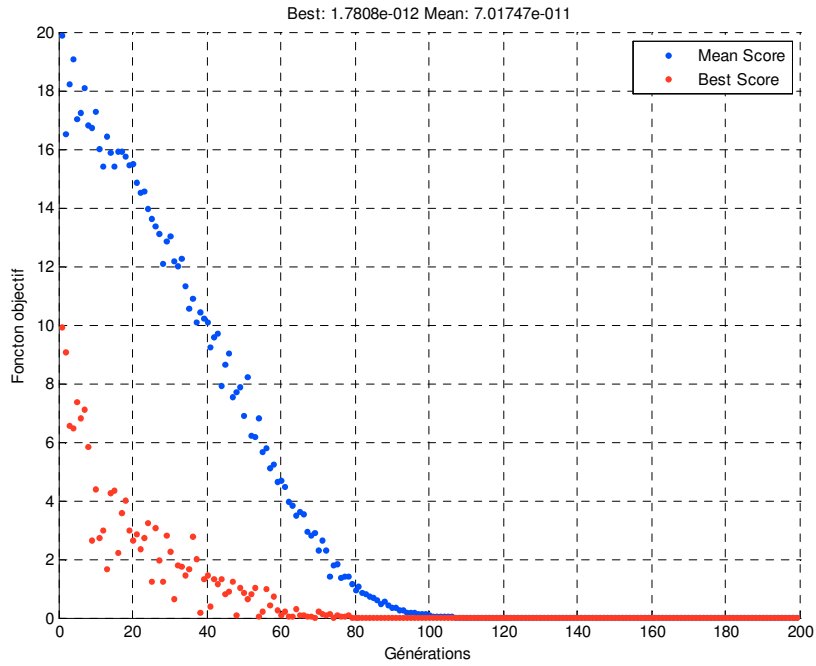


Figure II.14 Résultats de minimisation de la fonction Ackley's avec l'algorithme PSO

c. Fonction de Rosenbrock's

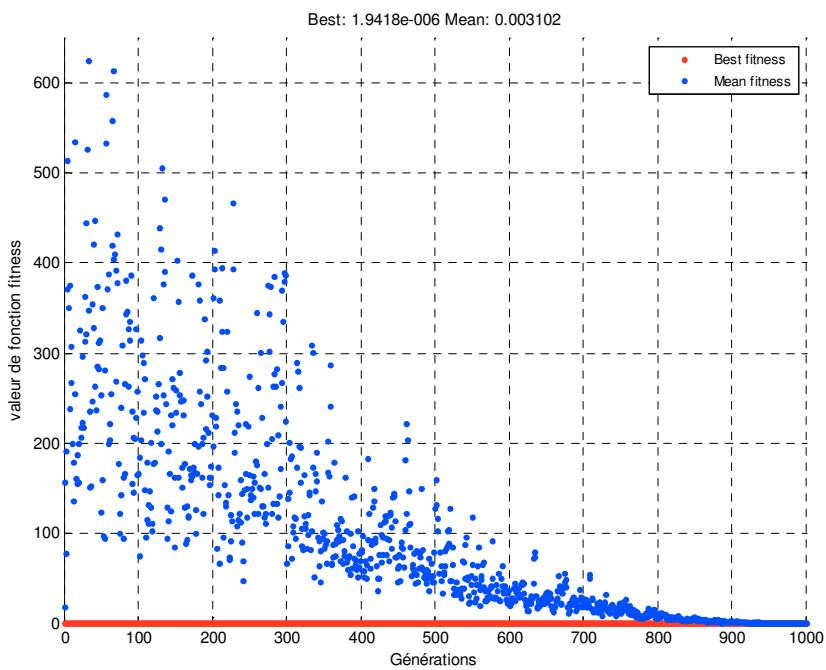


Figure II.15 Résultats de minimisation de la fonction Rosenbrock's avec l'algorithme génétique

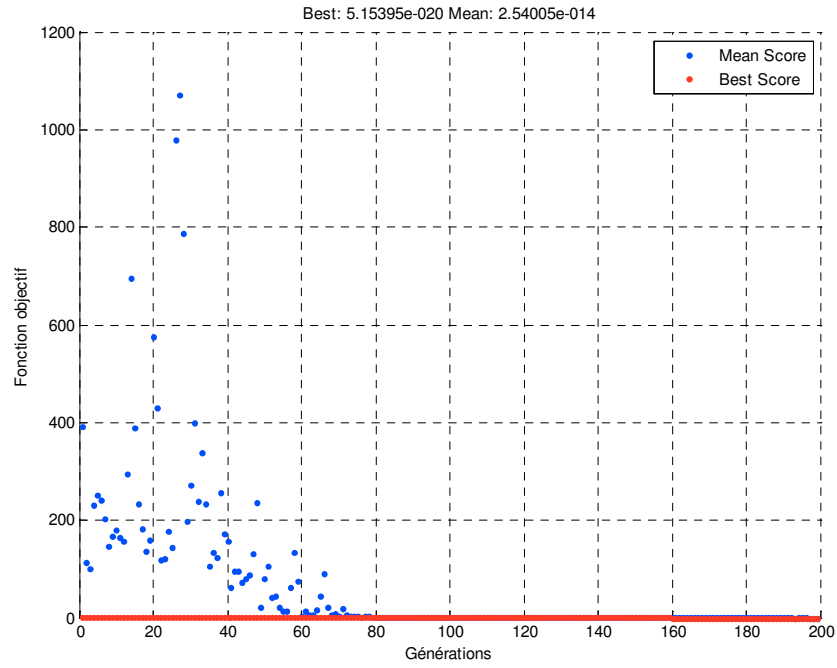


Figure II.16 Résultats de minimisation de la fonction Rosenbrock's avec l'algorithme PSO

d. Fonction de Schwefel's

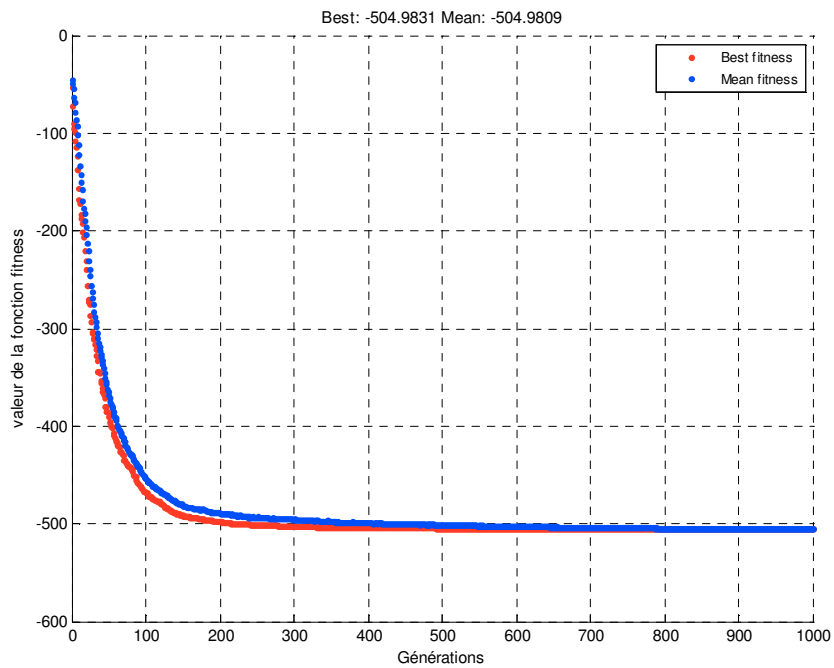


Figure II.17 Résultats de minimisation de la fonction Schwefel's avec l'algorithme génétique

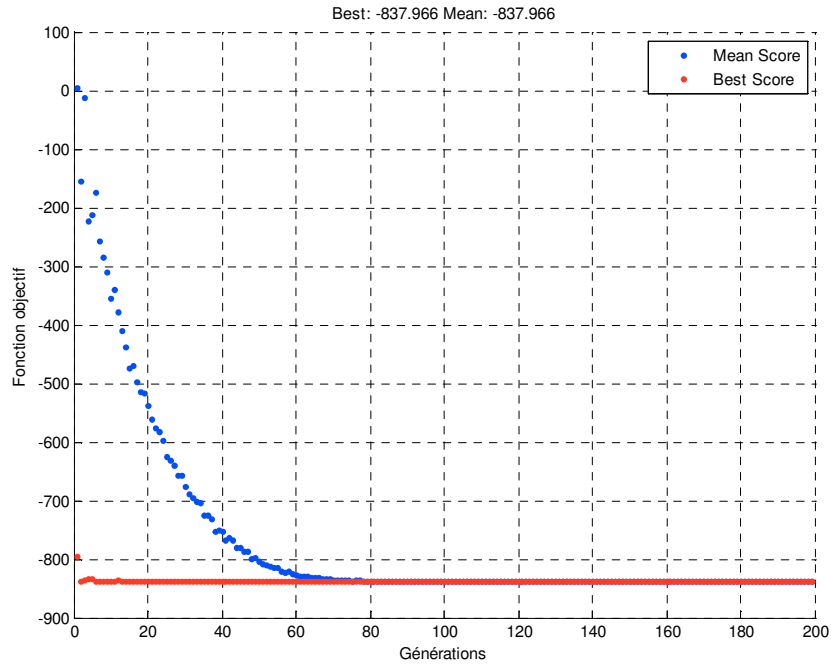


Figure II.18 Résultats de minimisation de la fonction Schwefel's avec l'algorithme PSO

e. Fonction de Rastrigin's

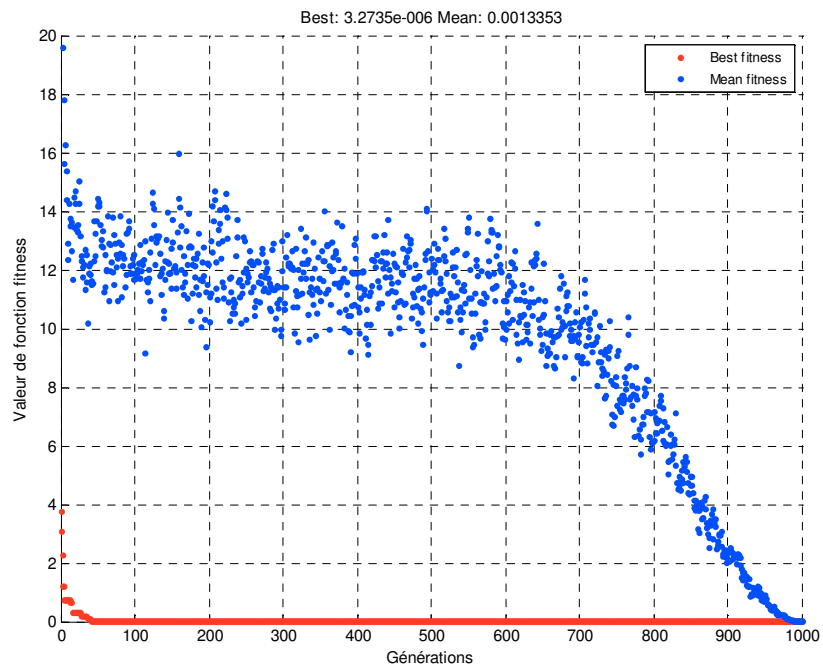


Figure II.19 Résultats de minimisation de la fonction Rastrigin's avec l'algorithme génétique

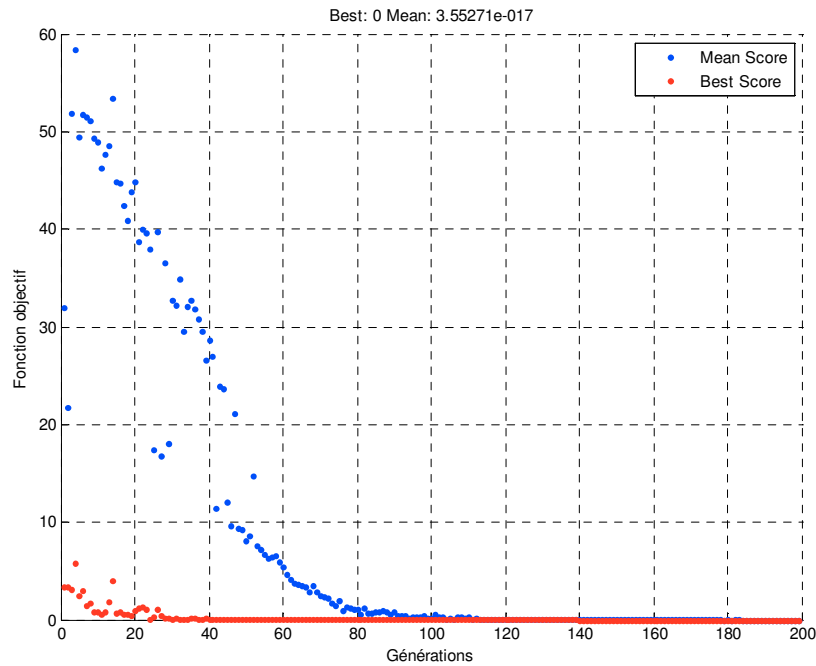


Figure II.20 Résultats de minimisation de la fonction Rastrigin's avec l'algorithme PSO

II.8 Discussion

La méthode du gradient conjugué (GC) et la programmation quadratique séquentielle (SQP) commencent le calcul avec un point spécifié par l'utilisateur (point d'initialisation). La solution optimale est ensuite recherchée, de façon itérative, en suivant la direction estimée à l'aide du gradient.

En examinant les résultats obtenus dans les tableaux II.1, II.2, II.3, II.4 et II.5 on remarque que les solutions obtenus par les méthodes déterministes varient proportionnellement avec le point initial (malgré que la méthode du gradient conjugué converge vers le même point pour les fonctions Rosenbrock's et Rastrigin's mais pour les autres fonctions elle converge vers d'autres points), ce qui exprime la difficulté de trouver l'optimum global.

On peut conclure que ces méthodes de recherche sont de nature locale, c'est à dire qu'elles convergent vers une solution proche du point initial du calcul.

L'efficacité des méthodes déterministes s'articule autour de la rapidité de l'évaluation de la fonction. En général, l'utilisation de ces méthodes nécessite comme étape préliminaire la localisation des extrema.

L'avantage que possède la programmation quadratique séquentielle par rapport à la méthode du gradient conjugué c'est qu'elle peut résoudre n'importe quel problème d'optimisation quelle que soit la nature de la fonction objectif et les contraintes.

Les méthodes déterministes sont de nature locale. Leur temps de calcul croît rapidement avec le nombre de variables optimisées. De plus, leur robustesse dépend de la disponibilité des solutions réalisables (qui satisfont toutes les contraintes) dès le début du calcul, ainsi que de l'absence de discontinuités dans le critère et les contraintes.

L'algorithme génétique (GA) et la méthode d'essaim particulaire (PSO) s'appuient sur des mécanismes de transition probabilistes et aléatoires. Cette caractéristique indique que plusieurs exécutions successives de ces méthodes peuvent conduire à des résultats différents pour une même configuration initiale d'un problème d'optimisation.

Ces deux méthodes ont de large capacité pour localiser l'optimum global du problème (figures de II.11 à II.20 : l'optimum global est toujours obtenu avec ces deux méthodes). Contrairement à la plupart des méthodes déterministes, elles ne nécessitent ni la définition d'un point de départ, ni la connaissance du gradient de la fonction objectif pour atteindre la solution optimale. Cependant, elles demandent un nombre important d'évaluations avant d'arriver à la solution du problème.

Ces deux méthodes, assez simple à utiliser et à implémenter, sont plus adaptées aux problèmes à plusieurs optimaux ou à la présence des variables discrètes. De plus, elles n'ont pas besoin d'information concernant le problème original, ce qui est intéressant lorsque la méthode d'optimisation et le simulateur ont été programmés séparément.

II.9 Proposition d'une méthode hybride

Afin de bénéficier de la capacité de recherche globale des méthodes stochastiques et la recherche locale précise des méthodes déterministes, nous proposons de les utiliser en complément l'un à l'autre.

Alors nous proposons la méthode hybride Algorithme génétique (GA)–programmation quadratique séquentielle (SQP). Pour ce faire, les solutions obtenues par l'algorithme génétique sont considérés comme des initiales pour la programmation quadratique séquentielle.

II.10 La Méthode hybride GA – SQP

Cette méthode consiste à exécuter séquentiellement les deux méthodes (GA et SQP) de telle façon que les résultats de l'algorithme génétique servent de solutions initiales à la programmation quadratique séquentielle. Cette technique d'hybridation est simple, elle ne nécessite pas de modification des deux méthodes de résolution utilisées. Il suffit

d'initialiser la programmation quadratique séquentielle à partir de la solution pré-calculée par l'algorithme génétique.

Les critères d'arrêt de l'algorithme génétique sont définis de sorte qu'elle serait arrêtée prématurément. Par exemple : avec une faible génération et une faible population. De façon à assurer que le résultat trouvé par l'algorithme génétique est proche du minimum global.

La figure II.21 ci-dessous présente le principe de cette méthode.

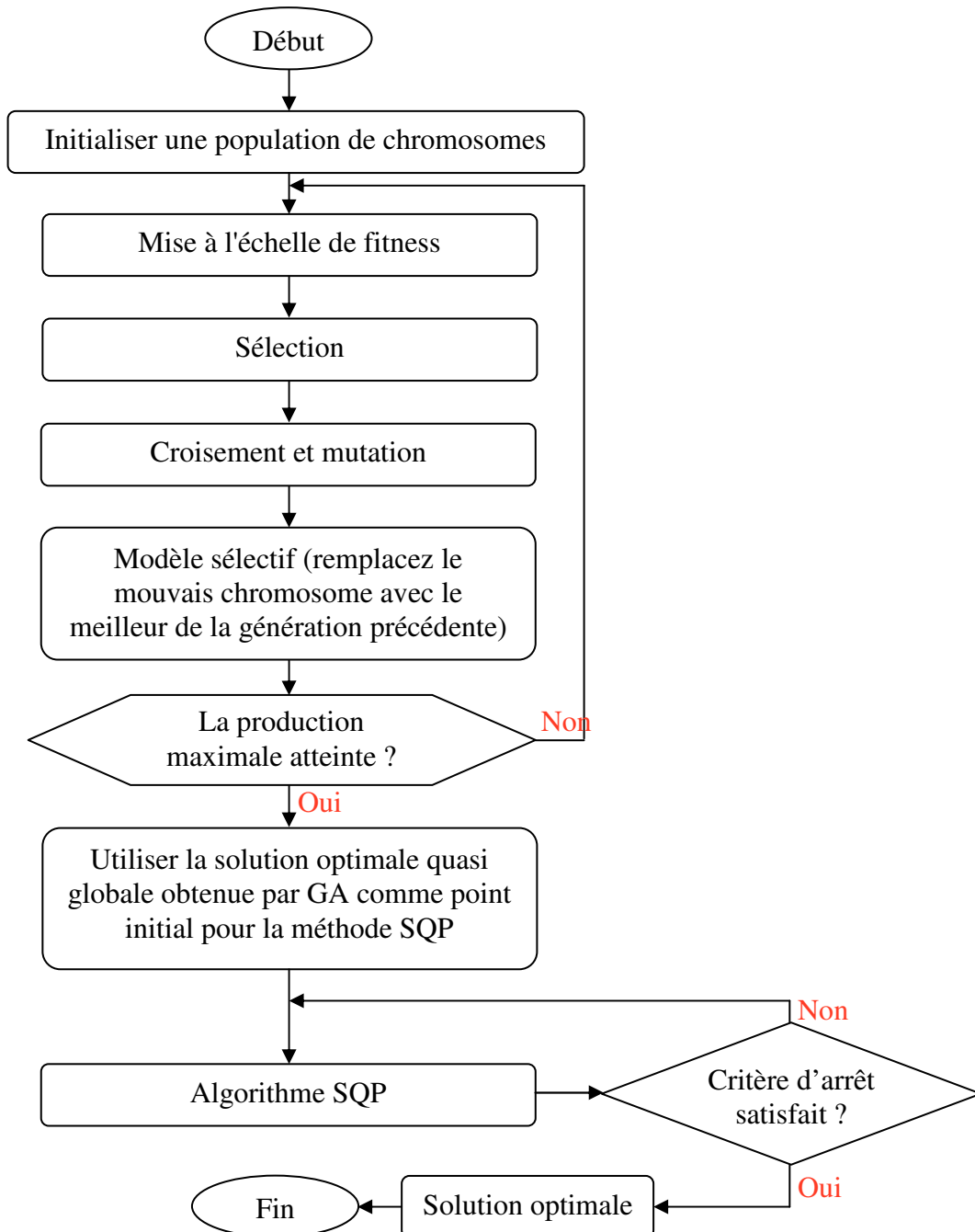


Figure II.21 Organigramme de la méthode GA-SQP

II.10.1 Résultats de la méthode hybride GA – SQP

Après exécution de l'algorithme de la méthode hybride GA – SQP, nous avons remarqué qu'il a convergé vers la solution exacte pour toutes les fonctions tests.

II.11 Comparaison

Les algorithmes d'optimisation peuvent être discriminés au moyen de différents critères [1] :

- Son faible temps de calcul,
- Sa robustesse (taux de convergence : sa sensibilité à la variabilité),
- Sa précision (proximité entre l'optimum identifié et l'optimum réel).

Nous considérons trop fluctuant le temps utilisé par l'algorithme pour converger : il dépend de l'ordinateur sur lequel il est exécuté, ainsi que des éventuels autres programmes en cours. En outre, il ne fait pas la distinction entre le temps de calcul de la fonction objectif (phase évaluative) et le temps nécessaire à l'exécution des opérateurs (phase opérative). La valeur de la fonction objectif nous semble aussi trop fluctuante, elle dépend du caractère stochastique de l'algorithme une fois le critère de convergence atteint, lui-même étant une valeur seuil pour la fonction objectif.

II.11.1 Critères de comparaison

Pour chaque couple d'algorithme/benchmark vingt exécutions ont été réalisées. Les résultats sont présentés au travers des trois critères : le nombre d'évaluations, le taux de convergence et la sensibilité paramétrique.

a. Nombre d'évaluations

Afin d'estimer l'efficacité de l'algorithme sans dépendre de la puissance de l'ordinateur via a vis du temps de convergence, nous proposons de compter le nombre d'évaluations nécessaires à l'atteinte du critère d'arrêt. Ce critère de comparaison permet de s'affranchir du nombre d'itérations (celles-ci étant composées d'un nombre d'évaluations, dépendant du nombre de solutions par population [1]) en étant indépendant du paramétrage de l'algorithme.

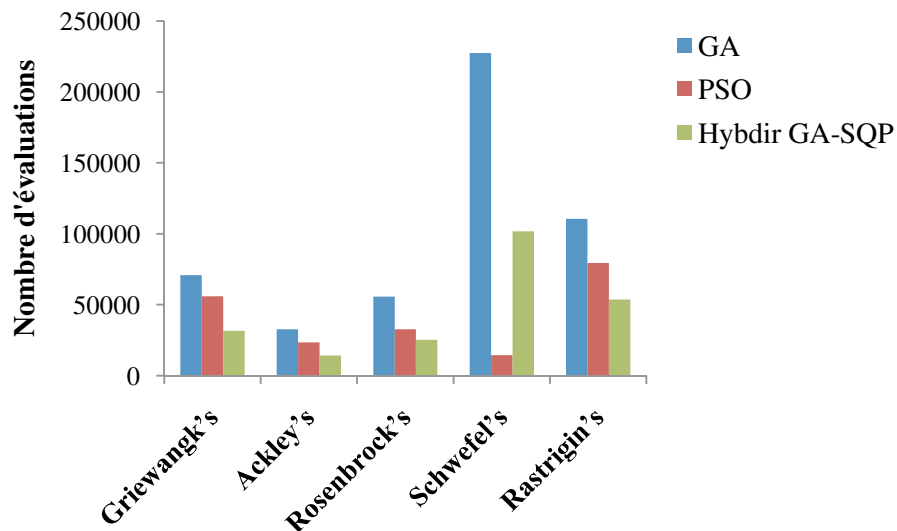


Figure II.22 Nombre d'évaluations moyen

Les exécutions ayant mené à un non convergence ne sont pas considérées ici.

Nous observons de plus que :

- La méthode hybrid GA-SQP est plus rapide que la méthode à essaim de particule et l'algorithme génétique
- La méthode à essaim de particule est plus rapide que l'algorithme génétique.

b. Taux de convergence

Le taux de convergence, exprimé en pourcentage, indique le nombre d'exécutions ayant convergé sur les dix tests pour chaque couple algorithme/benchmark. Il permet d'estimer la régularité (ou la robustesse, au sens de Chamekh [32]) d'un algorithme et, ainsi, de s'affranchir partiellement de son caractère stochastique, à l'origine de variabilité et d'imprévisibilité.

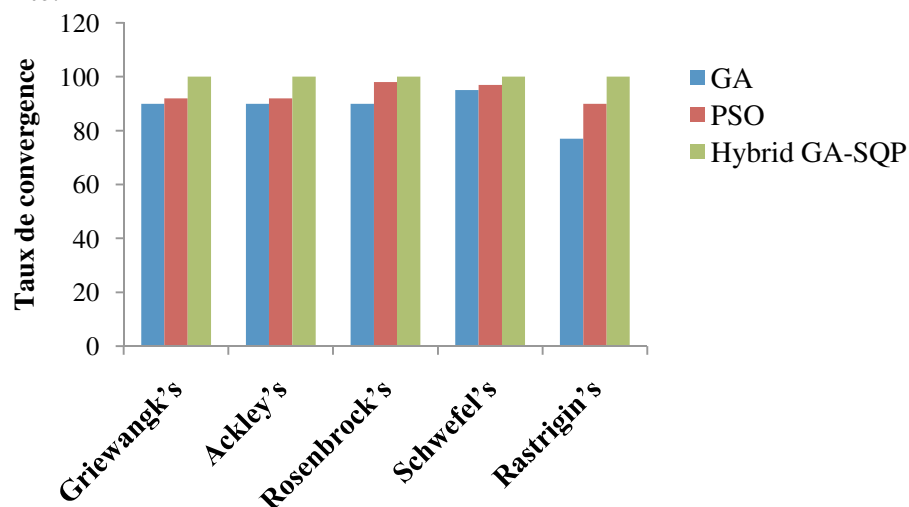


Figure II.23 Taux de convergence

D'après la figure II.23, la méthode s'avère être extrêmement régulière puisqu'elle dispose d'un taux de convergence de 100%, quelque soit la fonction test considérée. En revanche, la méthode PSO affiche un assez mauvais taux de convergence, s'élevant à 94% sur l'ensemble des fonctions test. De même l'algorithme génétique affiche un taux de convergence un peu plus faible (88 %) par rapport à la méthode hybride GA-SQP et la méthode PSO.

c. Sensibilité paramétrique

Ce critère original est proposé dans un contexte d'utilisation industrielle en partant du principe que l'utilisateur d'un algorithme n'en connaît pas forcément le fonctionnement. A ce titre, il lui est difficile de le paramétrer efficacement. Ce critère est un indicateur de la nécessité ou non de modifier l'un des paramètres de l'algorithme pour différents benchmarks à traiter [1].

Pour un couple algorithme/fonction test donné, le meilleur jeu de paramètres retenu (par balayage, voir paramétrage des algorithmes) est celui qui a mené à une convergence avec un minimum d'évaluations. Le Tableau II.6 présente les résultats pour chaque couple.

Tableau II.6 Jeu de paramètres optimums

| | Griewangk's | Ackley's | Rosenbrock's | Schwefel's | Rastrigin's | |
|--------|---|---|---|--|---|-----------------------------------|
| GA | $N_{pop} = 150$ $c_f = 0.3$ $m_f = 0.3$ | $N_{pop} = 150$ $c_f = 0.5$ $m_f = 0.3$ | $N_{pop} = 100$ $c_f = 0.5$ $m_f = 0.1$ | $N_{pop} = 50$ $c_f = 0.3$ $m_f = 0.1$ | $N_{pop} = 150$ $c_f = 0.8$ $m_f = 0.1$ | |
| PSO | $N = 100$ $c_1 = 0.5$ $c_2 = 1.25$ | $N = 200$ $c_1 = 0.5$ $c_2 = 1.25$ | $N = 100$ $c_1 = 0.5$ $c_2 = 1$ | $N = 40$ $c_1 = 0.5$ $c_2 = 1.25$ | $N = 150$ $c_1 = 0.5$ $c_2 = 1$ | |
| GA-SQP | $N_{pop} = 50$ $c_f = 0.3$ $m_f = 0.1$ | $N_{pop} = 50$ $c_f = 0.3$ $m_f = 0.1$ | $N_{pop} = 50$ $c_f = 0.3$ $m_f = 0.1$ | $N_{pop} = 50$ $c_f = 0.3$ $m_f = 0.1$ | $N_{pop} = 50$ $c_f = 0.3$ $m_f = 0.1$ | |
| | N_{pop} : | c_f : | m_f : | N : | c_1 : | c_2 : |
| | la taille de la population | fraction de croisement | fraction de migration | nombre de particules | constante positive d'accélération | constante positive d'accélération |

L'algorithme génétique et la méthode d'essaim de particules affichent une grande sensibilité paramétrique, indiquant qu'un paramétrage adéquat est à réaliser pour optimiser chaque fonction-test individuellement. Un tel comportement nécessite donc une connaissance préalable de l'algorithme, sans évoquer une connaissance du problème à traiter. La méthode hybride GA-SQP, au contraire, dispose d'une sensibilité paramétrique nulle : la taille de population, la fraction de croisement et fraction de migration sont globalement constantes quelque soit la fonction traitée.

II.12 Conclusion

Dans ce chapitre, nous avons testé deux méthodes déterministes et deux méthodes stochastiques ainsi qu'une méthode hybride en vue de choisir une méthode efficace et fiable pour résoudre un problème de production industrielle.

Nous avons constaté que :

- L'efficacité des méthodes déterministes s'articule sur de la rapidité de la convergence mais elles convergent vers une solution proche du point initial.
- Les méthodes stochastiques sont des méthodes d'optimisation globale simples à utiliser et à implémenter, mais elles demandent un nombre important d'évaluations et d'exécutions avant d'arriver à la solution du problème.

A travers la comparaison que nous avons fait sur la base de trois critères nous concluons que la méthode Hybride GA-SQP est plus efficace par rapport aux autres méthodes. Donc cette méthode est souhaitable pour la résolution des problèmes de production industrielle.

Chapitre III
Application Industrielle

III.1 Introduction

La résolution du problème d'optimisation des conditions de coupe a un intérêt considérable dans le domaine de la production industrielle puisqu'elle permet l'utilisation rationnelle des moyens de fabrication (Machine-outil, outillage...)

L'intérêt de l'optimisation des conditions de coupe réside dans l'amélioration de certains paramètres d'usinage, dit critères d'optimisation (temps de production, productivité, état de surface,...etc.), en tenant compte des limites de performance de la machine, de l'outil de coupe et de la matière usinée. Ces limites sont tirées des lois de coupe et exprimées en fonction des conditions de coupe (vitesse de coupe, avance par dent et profondeur de passe). Elles sont appelées contraintes d'optimisation et elles limitent le domaine de recherche des conditions optimales.

Dans ce chapitre, nous proposons d'utiliser la méthode Hybride GA–SQP pour rechercher les conditions de coupe optimales (la vitesse de coupe, l'avance et la profondeur de passes) pour les processus de tournage et de fraisage multi-passe.

Pour atteindre cet objectif nous présentons la formulation et la modélisation mathématique d'un problème d'optimisation multi-passe des conditions de coupe en tournage et en fraisage.

En vue de la validation de nos modèles et notre méthode d'optimisation, nous proposons la résolution de trois exemples d'application et la comparaison des résultats obtenus avec ceux de la littérature.

III.2 Formulation d'un modèle d'optimisation des conditions de coupe en tournage

III.2.1 Généralité sur le procédés de tournage

a. Définition

Le tournage est un procédé de fabrication mécanique par coupe (enlèvement de matière) mettant en jeu des outils à arête unique.

La pièce est animée d'un mouvement de rotation (mouvement de coupe), qui est le mouvement principal du procédé (Figure III.1).

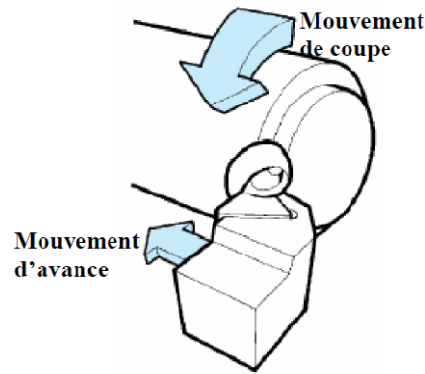


Figure III.1 Mouvements de coupe et d'avance en tournage [33].

L'outil est animé d'un mouvement complémentaire de translation (rectiligne ou non) appelé mouvement d'avance, permettant de définir le profil de la pièce.

La combinaison de ces deux mouvements, ainsi que la forme de la partie active de l'outil, permettent d'obtenir des usinages de formes de révolution (cylindres, plans, cônes ou formes de révolution complexes).

b. Les conditions de coupe

Le mouvement de coupe anime la pièce (pièce tournante). On en déduit la vitesse de coupe v_c . Le mouvement d'avance est un mouvement de translation de l'outil par rapport à la pièce, On en déduit v_f .

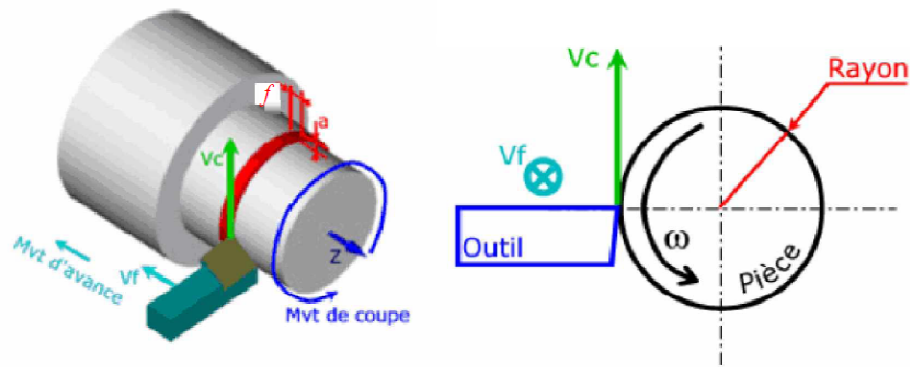


Figure III.2 Conditions de coupe en tournage [34].

▪ La vitesse de coupe

La pièce est entraînée sur le tour à une certaine vitesse ω rad / s (soit N tr / min), cette vitesse angulaire étant communiquée par la broche de la machine via la porte pièce (Figure III. 2).

Compte tenu du diamètre de la pièce au point d'usinage situé sur un diamètre D , la vitesse relative de la pièce en ce point par rapport à l'outil (supposé fixe par rapport à la machine) vaut :

$$v_c (m / s) = \frac{D}{2} (m) \times \omega (rad / s) \quad (III-1)$$

Cette vitesse est appelée vitesse de coupe, soit, avec les unités traditionnelles de la fabrication mécanique :

$$N (tr / min) = \frac{1000 \times v_c (m / min)}{\pi D (mm)} \quad (III- 2)$$

Il convient d'observer que la vitesse de coupe n'est constante que si la vitesse de la broche et le diamètre de la pièce demeurent inchangeables.

- **La vitesse d'avance - Avance par tour**

La vitesse d'avance $v_f (mm / min)$ (Figure III. 2) est la vitesse à laquelle la machine déplace l'outil par rapport au bâti. L'avance par tour $f (mm / tr)$ est la valeur du déplacement de l'outil, lorsque la pièce a effectué une révolution. C'est une donnée clé pour la qualité de la surface usinée.

$$v_f = f (mm / tr) \times N (tr / min) \quad (III- 3)$$

- **La profondeur de passe**

En chariotage, la profondeur de passe $a(mm)$ (Figure III. 2) est la différence de rayon entre la surface non usinée et la surface usinée (c'est-à-dire la moitié de la différence entre le diamètre non usiné et le diamètre usiné). La profondeur de coupe est toujours mesurée perpendiculairement à la direction de l'avance et non pas suivant l'arête de l'outil.

Dans la partie suivante, nous allons présenter deux modèles d'optimisation en vue de réaliser des comparaisons avec diverses méthodes trouvées dans la littérature.

III.2.2 Modèle 1

Pour construire ce modèle nous utilisons la méthode élaborée par J.S Agapiou [35] basée sur la programmation dynamique pour laquelle nous avons introduit des modifications.

a. Les variables de décision

Dans la construction du modèle d'optimisation, trois variables de décision sont envisagées: les vitesses de coupe (v_c), l'avance par tour (f), et la profondeur de chaque passe pour l'ébauche et la finition (d)

b. Formulation de la fonction objectif

Critère du coût de production C_u

Le coût de production par pièce pour une opération d'usinage contient la somme des coûts pour l'outillage, l'usinage, le temps de changement d'outil, le temps de montage et démontage de la pièce et le temps du retour rapide.

$$C_u = c_0 t_c + \left(\frac{t_c}{T} \right) \times (c_0 t_{cs} + c_t) + c_0 (t_r + t_h) \quad (\text{III-4})$$

Le temps de coupe par passe t_c est donné par :

$$t_c = \frac{\pi \cdot D \cdot L}{1000 \times V_c \times f} \quad (\text{III-5})$$

Où, V_c et f sont respectivement, la vitesse de coupe et la vitesse d'avance « variables d'usinage » tandis que, D et L sont respectivement le diamètre extérieur et la longueur de la partie de la pièce à usinée.

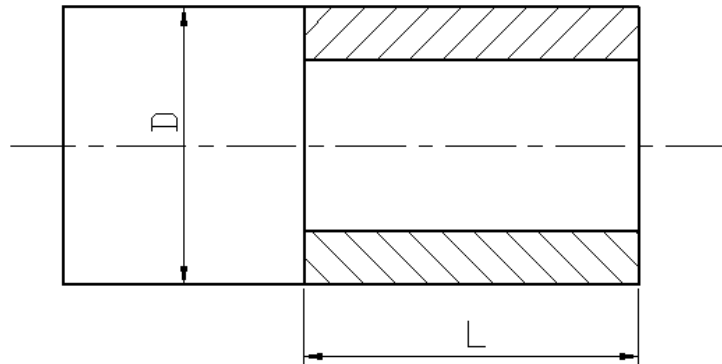


Figure III.3 Dimensions de la pièce à usinée

L'équation de la durée de vie de l'outil « modèle de Taylor » est de la forme :

$$V f^{a_1} d^{a_2} T^{a_3} = K \quad (\text{III-6})$$

Où, a_1, a_2, a_3 et K sont des constantes empiriques, et le pas de la profondeur de coupe pour une passe donnée [36].

Ainsi, le coût de production est donné par :

$$C_u = C_0 A V_c^{-1} f^{-1} + A V_c^{(1-a_3)/a_3} f^{(a_1-a_3)/a_3} d^{(a_2/a_3)} K^{(-1/a_3)} \cdot (C_0 t_{cs} + C_t) + C_0 (t_h + t_r) \quad (\text{III-7})$$

Avec :

$$A = \frac{\pi \cdot D \cdot L}{1000}$$

Critère du temps de production T_{pr}

Le temps total exigé pour produire une pièce est la somme des temps nécessaires pour l'usinage, le changement d'outil, le retour rapide d'outil et le temps de montage et démontage de la pièce.

$$T_{pr} = t_c + t_{cs} \cdot \left(\frac{t_c}{T} \right) + t_r + t_h \quad (\text{III-8})$$

En remplaçant t_c et T par leurs expressions respectives dans l'équation du temps d'usinage précédent, On aura le temps de production sous la forme suivant :

$$T_{pr} = A V_c^{-1} f^{-1} + A V_c^{(1-a3)/a3} f^{(a1-a3)/a3} d^{(a2/a3)} . K^{(-1/a3)} t_{cs} + t_h + t_r \quad (\text{III-9})$$

c. Les contraintes

L'optimisation de la fonction objectif nécessite souvent des limitations pratiques des conditions de coupe (puissance de coupe, force de coupe, température, l'état de surface, gammes des vitesses et des avances, etc..).

Les limitations additionnelles sont nécessaires pour la stabilité du processus de coupe et le respect de la qualité de la pièce à usiner.

Limitation sur l'intervalle des vitesses de coupe

La vitesse de coupe V_c doit être comprise dans les limites de validité du modèle de la loi de la durée de vie établie pour la couple outil-matière:

$$V_{c\min} \leq V_c \leq V_{c\max} \quad (\text{III-10})$$

Limitation sur l'intervalle des avances

L'avance par tour f doit être comprise entre deux valeurs extrêmes programmables sur la machine-outil, soit:

$$f_{\min} \leq f \leq f_{\max} \quad (\text{III-11})$$

Limitation sur la puissance de coupe

La puissance nécessaire à la coupe doit être inférieure à une puissance limite, soit :

$$C_p v_c^b f^c d^{d'} \leq P_m \quad (\text{III-12})$$

C_p, b, c, d' : Coefficients spécifiques et exposantes de la puissance de la machine.

P_m : Le maximum autorisé de puissance pour toutes les opérations.

Limitation sur l'effort de coupe

L'effort de coupe doit être inférieur à une force de coupe limite, soit:

$$C_f v_c^x f^y d^z \leq F_{\max} \quad (\text{III-13})$$

C_f, x, y, z : Coefficients spécifiques et exposantes de la force de coupe.

F_{\max} : Le maximum autorisé de la force de coupe pour toutes les opérations.

Limitation sur l'état de surface

L'état de surface résultant d'un usinage doit être inférieur à une rugosité maximale imposée, soit:

$$C_s v_c^e f^g d^i \leq R_{\max} \quad (\text{III-14})$$

C_s, e, g, i : Coefficients spécifiques et exposantes de la rugosité de surface résultante.

R_{\max} : Le maximum autorisé de rugosité de surface.

Limitation sur la température de coupe

Le facteur de contrôle est la température d'interface outil-copeau doit être inférieure à une température θ_{\max} , soit :

$$C_m v_c^m f^p d^q \leq \theta_{\max} \quad (\text{III-15})$$

C_m, m, p, q : Coefficients spécifiques et exposantes de la température de coupe.

θ_{\max} : Le maximum autorisé de température de coupe

Toutes les équations mentionnées ci dessus déterminant les contraintes qui ont été développées par des données expérimentales collectées pour les études de l'usinage des aciers ordinaires [37].

Après remplacement des exposants et les constants dans les équations des contraintes nous obtenons le problème suivant:

$$\left\{ \begin{array}{l} U_1(x) = C_u = C_0 A V_c^{-1} f^{-1} + A V_c^{(1-a_3)/a_3} f^{(a_1-a_3)/a_3} d^{(a_2/a_3)} K^{(-1/a_3)} \cdot (C_0 t_{cs} + C_t) \\ \quad + C_0 (t_h + t_r) \\ U_2(x) = T_{pr} = A V_c^{-1} f^{-1} + A V_c^{(1-a_3)/a_3} f^{(a_1-a_3)/a_3} d^{(a_2/a_3)} K^{(-1/a_3)} t_{cs} + t_h + t_r \\ 0,0373 V^{0.91} f^{0.78} d^{0.75} \leq P_{\max} \\ 844 V^{-0.1013} f^{0.725} d^{0.75} \leq F_{\max} \\ 74,96 V^{0.4} f^{0.2} d^{0.105} \leq \theta_{\max} \\ V_{\min} \leq V_i \leq V_{\max} \\ f_{\min} \leq f_i \leq f_{\max} \end{array} \right. \quad (\text{III-16})$$

d. Modélisation multi-passe par la programmation dynamique

Dans les opérations de tournage, un processus de coupe peut éventuellement être complété par une passe unique ou par plusieurs passes. Une opération de coupe multi-passe implique plusieurs passes d'ébauche et des passes de finition. Les paramètres d'usinage peuvent être déterminés en fonction de l'expérience de l'opérateur de la machine ou en suivant le guide de coupe fournit par le fabricant de l'équipement.

Toutefois, ces données ne sont pas garanties pour être optimales. Par conséquent, le développement de modèles mathématiques pour les opérations multi-passe de tournage est devenu un outil utile pour déterminer les conditions de coupe optimales.

La valeur optimale du nombre de passes, la vitesse, l'avance et la profondeur de coupe pour chaque passe sont obtenues par le processus multi-passe. C'est un problème à quatre variables où le nombre de passe et la profondeur de coupe pour chaque passe sont déterminés à l'aide de la procédure de programmation dynamique et où la vitesse de coupe optimale et l'avance pour chaque passe sont déterminées en utilisant la méthode hybride GA-SQP.

Le concept de la programmation dynamique est très utile pour le traitement du problème multi-passe dans lequel chaque passe est indépendante des passes antérieures, comme dans le cas des opérations de tournage.

La variable de décision qui est la profondeur de coupe a à enlever par passe i est représentée par $d(i, j)$ et la variable d'état qui est le diamètre de la pièce à la passe i est représentée par D_i . Le $d(i, j)$ signifie que la profondeur de coupe commence au diamètre D_i et contient j sections de dimension d .

Par conséquent la profondeur de coupe totale D_c est divisée en N sections égales qui sont les N états de décision discrets pour la programmation dynamique.

L'incrément minimal (pas) de la profondeur de coupe est :

$$d = D_c / N \quad \text{(III-17)}$$

Optimisation du nombre de passes

En générale lorsque la passe débute à la section i et se termine à la section q la fonction optimale est donnée par :

$$U(i, j) = C_u(i, j) \quad \text{(III-18)}$$

Avec $j = i - q$

j : est le nombre de sections de taille d .

Cependant, l'optimisation à simple passe est généralement considérée pour les quelques premières étapes de la procédure de la programmation dynamique ou « a » peut être enlevée en une seule passe.

Le coût de production par passe est donné par :

$$C_u(i, j) = C_0 \cdot AV_c^{-1}(i, j) \cdot f^{-1}(i, j) + AV_c^{((1/a_3)-1)}(i, j) \cdot f^{((a_1/a_3)-1)}(i, j) \cdot d^{(a_2/a_3)}(i, j) \cdot K^{(-1/a_3)} \cdot (C_0 \cdot t_{cs} + C_t) + C_0 \cdot t_r \quad (\text{III-19})$$

Le temps de production est donné par :

$$T_{pr}(i, j) = C_0 \cdot V_c^{-1}(i, j) \cdot f^{-1}(i, j) + AV_c^{((1/a_3)-1)}(i, j) \cdot f^{((a_1/a_3)-1)}(i, j) \cdot d^{(a_2/a_3)}(i, j) \cdot K^{(-1/a_3)} \cdot t_{cs} + t_r \quad (\text{III-20})$$

$$\text{Avec} \quad A = \frac{\pi \cdot D_i \cdot L(i, j)}{1000} \quad \text{et} \quad d(i, j) = a = j \cdot d$$

La longueur de coupe reste toujours constante pour toutes les passes d'usinage
 $L(i, j) = L$

Le diamètre pour chaque passe est donné par :

$$D_i = D - 2(N + j - i) \cdot d = D_0 + 2q \cdot d \quad (\text{III-21})$$

Et

$$D_{i-1} = D_i - 2j \cdot d$$

Pour l'usinage multi-passes où N_p passes sont utilisées pour effectuer le processus de tournage, le coût et le temps de production total sont donnés par les relations suivantes :

$$C_u = \sum_{k=1}^{N_p} \left[C_0 \cdot (t_c(i, j)_k + t_r) + \frac{t_c(i, j)_k}{T(i, j)_k} (C_0 \cdot t_{cs} + C_t) \right] + C_0 \cdot t_h \quad (\text{III-22})$$

$$T_{pr} = \sum_{k=1}^{N_p} \left[t_c(i, j)_k + t_r + \frac{t_c(i, j)_k}{T(i, j)_k} \cdot t_{cs} \right] + t_h \quad (\text{III-23})$$

k : est l'indice indiquant le numéro d'une passe quelconque.

Le temps machine pour chaque passe est :

$$t_c(i, j)_k = \frac{\pi \cdot (D_i)_k \cdot L(i, j)_k}{1000 V(i, j)_k \cdot f(i, j)_k} \quad (\text{III-24})$$

Les étapes des itérations dynamiques sont représentées par une matrice triangulaire inférieure de dimension $[N \times N]$. Les éléments (i, j) de la matrice représentent

respectivement la section du diamètre de démarrage et le nombre des sections usinées pour une passe quelconque.

La représentation schématique de la procédure de programmation dynamique est donnée par le diagramme de la (figure III.4) où D_c est divisé en sept sections égales ($N = 7$) .

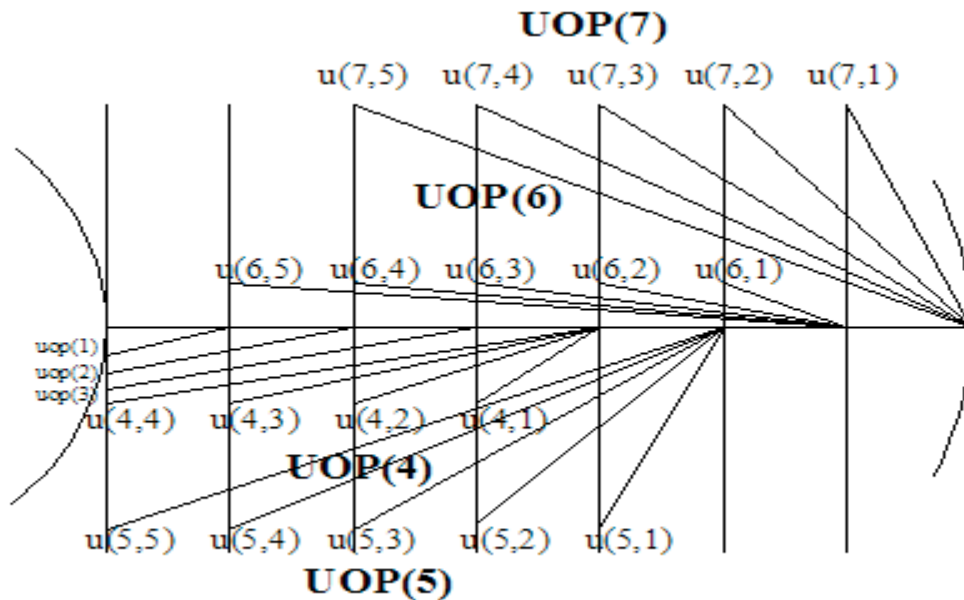


Figure III.4 Schéma représentant la programmation dynamique lorsque la profondeur de coupe totale est divisée en 7 sections. [36]

Il est connu qu'au-dessous d'une certaine valeur de profondeur de coupe $DMOP$, pour un couple spécifique outil-matière, l'usinage à passe simple est optimal. La valeur maximale de la profondeur de coupe $DMAXP$ est aussi connue.

La matrice triangulaire inférieure pour l'exemple décrit dans la figure (III.4) est donnée par :

$$U(i, j) = \begin{pmatrix} X & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & X & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & X & 0 & 0 & 0 & 0 \\ X & X & X & X & 0 & 0 & 0 \\ X & X & X & X & X & 0 & 0 \\ X & X & X & X & X & 0 & 0 \\ X & X & X & X & X & 0 & 0 \end{pmatrix}$$

Où X représente les valeurs de la fonction objective, les éléments de la diagonale représentent les passes simples possibles. Les éléments $(2,1), (3,1), (3,2)$ sont nuls puisque

la profondeur de coupe pour une seule passe donnée est inférieure à $DMOP$ et les éléments $(6,6)$, $(6,7)$ et $(7,7)$ sont nuls puisque la profondeur de coupe pour une seule passe donnée est supérieure à $DMAXP$.

De la même manière, les vitesses de coupe et les avances sont aussi rangées dans deux matrices triangulaires inférieures séparées.

Les étapes de la programmation dynamique

En utilisant la stratégie de la programmation dynamique, en supposant que l'optimum est obtenue à partir de la $i^{ème}$ section, $1 < i < N$ jusqu'au diamètre final (interne) de la section, les étapes ci-dessous, caractérisant la programmation dynamique sont nécessaires.

- **Etape 1 :** Pour la profondeur de coupe est inférieure à $DMOP$ tel que :
 $d(i,i) = i.d \leq DMOP$, on calcule les fonctions objectif correspondantes $UOP(i)$ pour des opérations de finition d'une seule passe est on continue avec les pas suivants.
- **Etape 2 :** Pour des profondeurs de passe $DMOP < i.d < DMAXP$, on calcule les fonctions objectives correspondantes $U(i,i)$ pour des opérations d'une seule passe de finition.
- **Etape 3 :** Par la suite on évalue les calculs des fonctions objectives pour des opérations de plusieurs passes telle que :

$$UOP(i) = UOP(i-r) + u(i,r) \quad (III-25)$$

Pour $r = 1, 2, \dots, p$

$$\text{Avec } p = \min\{i-1, DMAXP/d\} \quad (III-26)$$

$UOP(i-r)$: est la fonction objectif minimale allant de la section $(i-r)$ jusqu'à la première section. C'est elle-même la fonction objectif d'une seule passe allant de la section (i) à la section $(i-r)$ avec une profondeur de coupe $a = r.d$.

- **Etape 4 :** On continue ce processus (première étape à la troisième étape) Pour $i = 1, 2, \dots, N$

e. Méthodologie de l'optimisation multi-passe

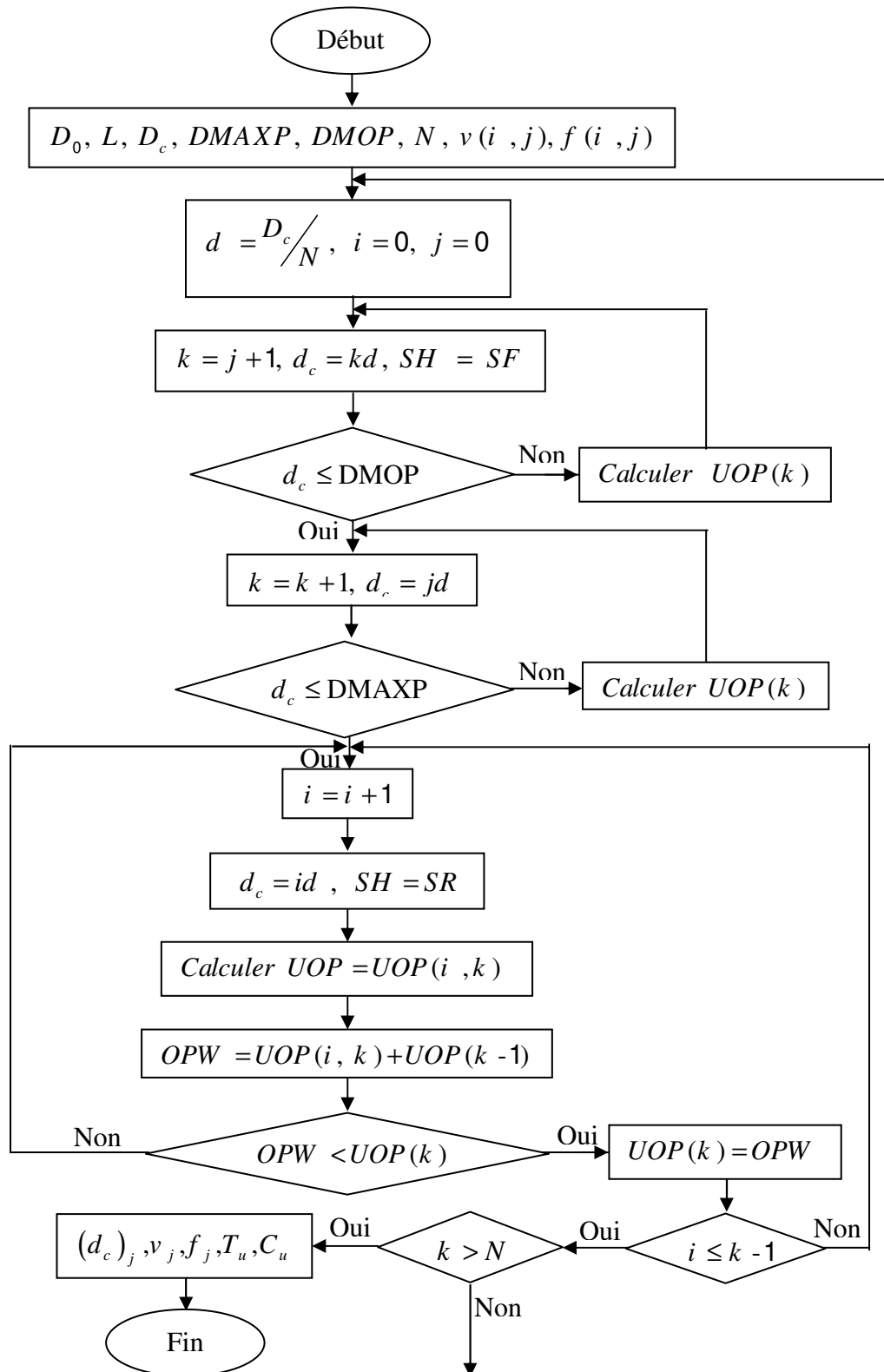


Figure III.5 Organigramme de la méthodologie pour l'optimisation multi-passe en tournage

III.2.3 Modèle 2

a. Les variables de décision

Dans la construction du modèle d'optimisation, six variables de décision sont envisagées: les vitesses de coupe pour l'ébauche et la finition (V_{ce}, V_{cf}), l'avance par tour pour l'ébauche et la finition (f_e, f_f), et la profondeur de chaque passe pour l'ébauche et la finition (d_e, d_f)

b. Formulation de la fonction objectif

Sur la base du critère coût de production unitaire minimum, C_u , la fonction objectif pour le tournage multi-passe est donnée comme suit [38],

$$C_u = C_M + C_I + C_R + C_T \quad (\text{III-27})$$

Avec:

$$C_M = k_0 \left[\frac{\pi DL}{1000V_{ce}f_e} \left(\frac{d_t - d_f}{d_e} \right) + \frac{\pi DL}{1000V_{cf}f_f} \right] \quad (\text{III-28})$$

$$C_I = k_0 \left[t_c + (h_1L + h_2) \left(\frac{d_t - d_f}{d_e} + 1 \right) \right] \quad (\text{III-29})$$

$$C_R = k_0 \frac{t_c}{T_p} \left[\frac{\pi DL}{1000V_{ce}f_e} \left(\frac{d_t - d_f}{d_e} \right) + \frac{\pi DL}{1000V_{cf}f_f} \right] \quad (\text{III-30})$$

$$C_T = \frac{C_t}{T_p} \left[\frac{\pi DL}{1000V_{ce}f_e} \left(\frac{d_t - d_f}{d_e} \right) + \frac{\pi DL}{1000V_{cf}f_f} \right] \quad (\text{III-31})$$

c. Les contraintes

Les contraintes qui influent sur le choix optimal des conditions de coupe seront prises en compte, alors les contraintes pour l'ébauche et pour la finition sont décrites ci-dessous:

Contraintes pour l'ébauche

▪ Limites des paramètres

Les paramètres de coupe doivent être compris entre deux valeurs extrêmes :

$$\text{La vitesse de coupe : } V_{ce_{\min}} \leq V_{ce} \leq V_{ce_{\max}} \quad (\text{III-32})$$

$$\text{L'avance par tour: } f_{e_{\min}} \leq f_e \leq f_{e_{\max}} \quad (\text{III-33})$$

$$\text{La profondeur de passe : } d_{e_{\min}} \leq d_e \leq d_{e_{\max}} \quad (\text{III-34})$$

- **Limitation sur la durée de vie de l'outil**

$$T_{\min} \leq T_e \leq T_{\max} \quad (\text{III-35})$$

- **Limitation sur la force de coupe**

L'effort de coupe doit être inférieur à une force de coupe limite, soit:

$$F_e = k_1 (f_e)^u (d_e)^v \leq F_{\max} \quad (\text{III-36})$$

- **Limitation sur la puissance de coupe**

La puissance nécessaire à la coupe doit être inférieure à une puissance limite, soit :

$$P_{ce} = \frac{F_e V_{ce}}{6120 \eta} \leq P_m \quad (\text{III-37})$$

Avec $\eta = 0.85$

- **Limitation sur la région stable de coupe**

La limitation sur la région stable de coupe a été proposée par Philipson et Ravindran [39] dans le but de prise en considération des vibrations, et d'adhésion.

$$(V_{ce})^\lambda f_e (d_e)^v \geq SC \quad (\text{III-38})$$

Les valeurs de λ et SC sont basés sur les valeurs utilisés par Philipson et Ravindran [39] tandis que la valeur de v est supposée être celle proposée par Narang et Fischer [40].

- **Limitation sur la température de coupe**

Le facteur de control est la température d'interface outil-copeau qui doit être inférieure à une température maximale :

$$\theta_e = C_m (V_{ce})^m (f_e)^p (d_e)^q \leq \theta_{\max} \quad (\text{III-39})$$

Contraintes pour la finition

Toutes les contraintes sauf la contrainte sur l'état de surface sont similaires pour l'ébauche et la finition [41].

- **Contrainte sur l'état de surface en finition**

Dans les opérations de finition la rugosité de la surface obtenue doit être inférieur à R_{\max} ,

$$\frac{f_f^2}{8R} \leq R_{\max} \quad (\text{III-40})$$

▪ **Relation entre les paramètres de coupe en ébauche et en finition**

$$V_f \geq k_3 V_e \quad (\text{III-41})$$

$$f_e \geq k_4 f_f \quad (\text{III-42})$$

$$d_e \geq k_5 d_f \quad (\text{III-43})$$

Nombre de passe en ébauche

Le nombre de passes possible en ébauche est donnée par :

$$N_p = \frac{d_t - d_f}{d_e} \quad (\text{III-44})$$

$$\text{Où } N_{p \min} \leq N_p \leq N_{p \max}$$

$$N_{p \min} = (d_t - d_{f \max}) / d_{e \max} \quad (\text{III-45})$$

$$N_{p \max} = (d_t - d_{f \min}) / d_{r \min} \quad (\text{III-46})$$

Le problème d'optimisation dans le tournage multi-passe est divisé en $m_p = (N_{p \max} - N_{p \min} + 1)$ sous-problèmes. Dans chaque sous-problème, le nombre de passes en ébauche N_p est fixé. Par conséquent, la recherche de la solution du problème d'optimisation consiste à trouver les solutions de m_p sous-problèmes, le minimum de ces résultats sera la solution du problème d'optimisation global.

Donc, le problème à résoudre doit prendre la forme suivante :

$$\left\{ \begin{array}{l} C_u = C_M + C_I + C_R + C_T \\ \text{SOUS} \\ V_{ce \min} \leq V_{ce} \leq V_{ce \max} \\ f_{e \min} \leq f_e \leq f_{e \max} \\ d_{e \min} \leq d_e \leq d_{e \max} \\ T_{\min} \leq T_e \leq T_{\max} \\ k_1 (f_e)^\mu (d_e)^v \leq F_{\max} \\ \frac{F_e V_{ce}}{6120 \eta} \leq P_m \\ (V_{ce})^\lambda f_e (d_e)^v \geq SC \\ C_m (V_{ce})^m (f_e)^p (d_e)^q \leq \theta_{\max} \\ \frac{f_f^2}{8R} \leq R_{\max} \end{array} \right.$$

d. Méthode de résolution

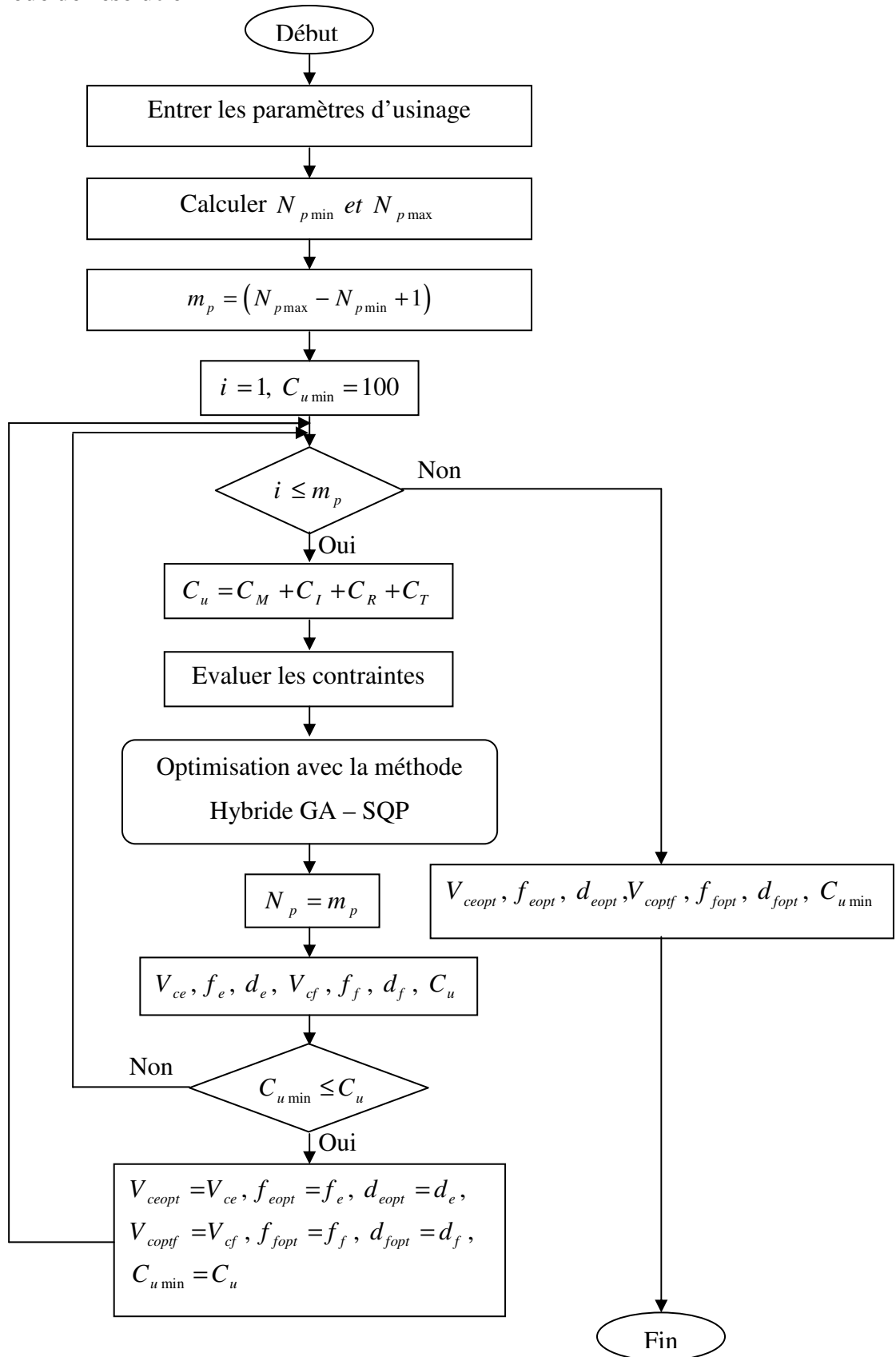


Figure III.6 Organigramme de la méthode de résolution

III.3 Formulation d'un modèle d'optimisation des conditions de coupe en fraisage

III.3.1 Généralité sur le procédé de fraisage

a. Définition

Le fraisage est un procédé d'usinage réalisé au moyen d'un outil multiple (à plusieurs arêtes de coupe) qui est animé d'un mouvement de rotation (Figure. III-7):

- Mouvement de rotation de la fraise qui est entraîné par la broche de la machine, M_c (mouvement rapide circulaire de coupe);
- Mouvement de translation de la pièce qui est fixée sur la table de la machine, M_a (mouvement lent rectiligne uniforme d'avance).

Le mouvement de translation est orienté de façon à pousser la pièce contre la fraise suivant une direction généralement perpendiculaire par rapport à son axe.

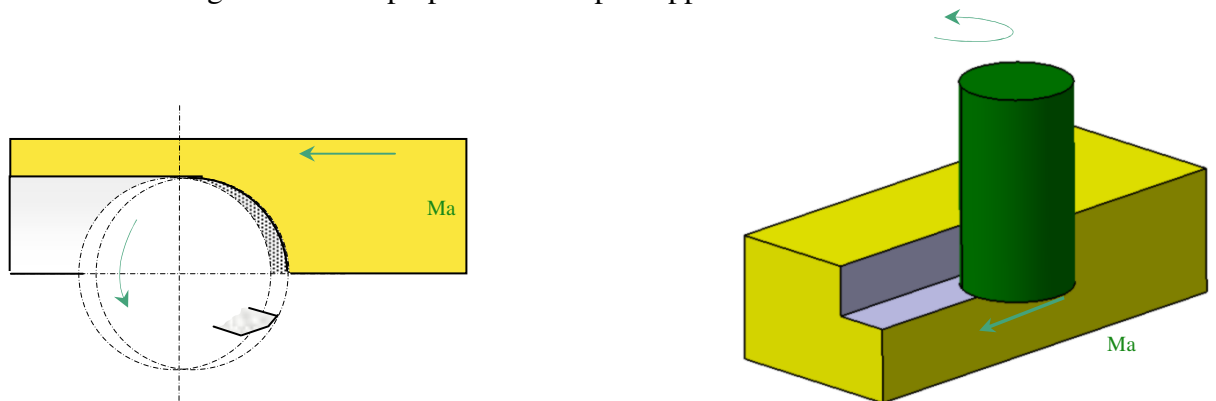


Figure III.7 Mouvements de coupe et d'avance en fraisage [42].

b. Les conditions de coupe en fraisage

Le mouvement de coupe anime l'outil (fraise tournante). Le mouvement d'avance est un mouvement de translation de l'outil.

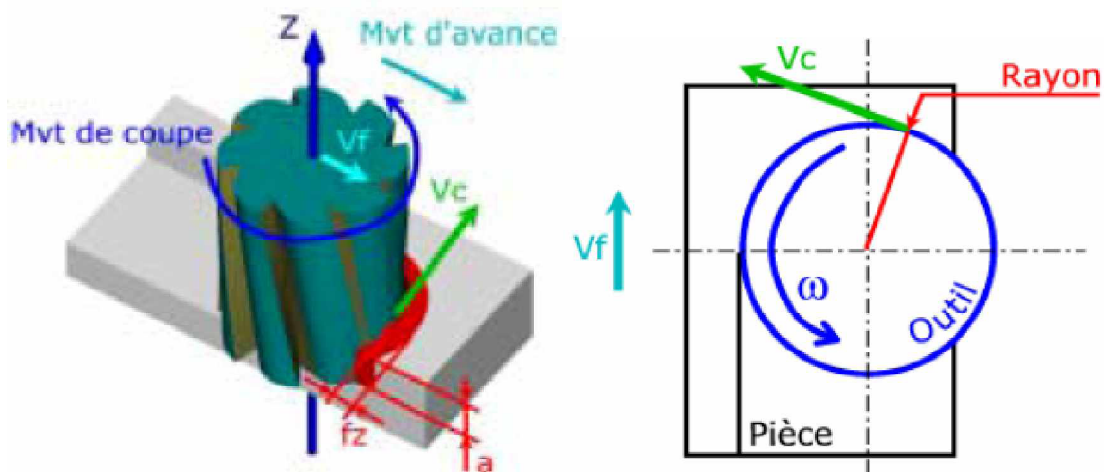


Figure III.8 Conditions de coupe en fraisage [34].

- **La vitesse de coupe**

Elle indique la vitesse à laquelle l'arête de coupe se déplace en travaillant dans la pièce. C'est un important paramètre de l'outil, qui fait partie intégrante des conditions de coupe avec, pour fonction, de garantir que l'opération est effectuée dans les meilleures conditions d'efficacité par l'outil concerné (figure III.8).

La vitesse de broche (N en tr/min) est le nombre de tours que l'outil de fraisage monté sur la broche de la machine-outil effectue par minute.

La vitesse de broche, le diamètre de l'outil et la vitesse de coupe sont naturellement liés par la formule (III-1). avec D : le diamètre de la fraise.

- **La vitesse d'avance - Avance par tour - Avance par dent**

La vitesse d'avance (v_f en mm/min) (figure III.8) est l'avance de l'outil en direction de la pièce, exprimée en unités de distance par unité de temps. On parle également ici d'avance de la table.

L'avance par tour (f en mm/tr) est une valeur spécialement utilisée pour calculer l'avance et déterminer l'aptitude d'une fraise à surfacer à travailler en finition. Elle indique de combien l'outil avance au cours d'une rotation.

L'avance par dent (f_z en $mm/dent$) (figure III.8) est un important paramètre en fraisage.

L'avance par dent indique la distance linéaire parcourue par l'outil alors qu'une certaine dent est engagée. L'avance par dent représente aussi la distance couverte entre la pénétration de deux dents successives dans la pièce. Elle peut donc être exprimée en fonction du nombre d'arête de l'outil (z) et de l'avance par minute, ou sous forme d'avance par tour.

- **La profondeur de passe**

La profondeur de passe a (mm) (figure III.8) correspond à l'épaisseur de matière enlevée par l'outil. C'est la distance à laquelle l'outil est réglé au-dessous de la surface initiale de la pièce.

L'approche utilisée pour la formulation du modèle d'optimisation des conditions de coupe en fraisage est basée sur la programmation dynamique.

III.3.2 Les variables d'optimisation

Les variables définissant les conditions de coupe en fraisage sont des grandeurs géométriques et cinématique qui vont intervenir dans la dynamique de coupe et la nature du milieu dans lequel est effectuée la coupe (lubrification). Les variables retenues pour une opération de fraisage sont :

- L'avance par dent f_z en (mm/dent).
- La profondeur de passe a en (mm).
- La vitesse de coupe V_c en (m/min).

III.3.3 Formulation de la fonction objectif en fraisage

La production maximale sera atteinte lorsque le temps de fabrication unitaire sera minimal. Le temps d'usinage unitaire est le temps nécessaire pour la réalisation d'une passe sur une pièce. C'est la somme des temps suivants [43]:

- Le temps de préparation de machine T_p (min) :

$$T_p = \frac{T_s}{N_b} \quad (\text{III-47})$$

Où T_s est le temps de mise en place de la machine pour un nouveau lot, et N_b est le nombre total de composants dans le lot.

- Le temps de Chargement et déchargement t_h (min).
- Le temps de réglage du processus et de retour rapide t_r (min).
- le temps de changement d'outil par composant T_m (min) :

$$T_m = \frac{t_{cs} \cdot t_c}{T} \quad (\text{III-48})$$

Où T_{cs} (min) est le temps de changement d'une arête de coupe ou d'un outil de coupe,

t_c (min) est le temps de coupe et T la durée de vie de l'outil de coupe.

- Le temps de coupe T_c (min) :

Donc le temps de production total par composante T_{pr} pour une opération à une seule passe est donné par l'expression suivant:

$$T_{pr} = T_p + t_h + t_r + t_c + T_m \quad (\text{III-49})$$

$$\text{Où } T_{pr} = \frac{T_s}{N_b} + t_h + t_r + T_c + \frac{T_{cs} \cdot t_c}{T} \quad (\text{III-50})$$

Pour une opération de fraisage multi-passe, le temps de production total T_{pr} est donné par l'expression suivante :

$$T_{pr} = \frac{T_s}{N_b} + t_h + \sum_{i=1}^{N_p} \left(t_{r_i} + t_{c_i} + \frac{t_{cs} \times t_{c_i}}{T_i} \right) \quad (\text{III-51})$$

Où, N_p représente le nombre total de passes.

Dans ce cas le temps de coupe T_c (min) et la durée de vie de l'outil dépendent des conditions de coupe qui doivent être calculés.

a. Calcul du temps de coupe en fraisage

Le temps de coupe en fraisage est donné par l'expression suivante:

$$T_c = \frac{l_u}{v_f} \quad (\text{III-52})$$

Où l_u (mm) : la longueur de coupe, et v_f (mm/min) : la vitesse d'avance.

b. Durée de vie de l'outil de coupe

Pour une opération de fraisage, la formule de la durée de vie de l'outil est donnée par Kaczmarek [43] comme suit:

$$T = \frac{c_v^{1/m} \times D^{b_v/m}}{V_c^{1/m} \times f_z^{u_v/m} \times v_f^{e_v/m} \times a^{r_v/m} \times B^{n_v/m} \times Z^{q_v/m} \times \lambda_s^{q_v/m}} \left(B_m \times B_h \times B_p \times B_t \right)^{1/m} \quad (\text{III-53})$$

Où B : largeur du copeau (mm), Z : nombre de dents de la fraise, λ_s angle de coupe (en degrés), $u_v, b_v, e_v, r_v, n_v, q_v$, et m : exposants et B_m, B_h, B_p, B_t : coefficients de correction.

La fonction objectif pour les opérations de fraisage multi-passe peut être exprimée comme suit:

$$T_{pr} = \frac{T_s}{N_b} + t_h + N_p \cdot t_r + \sum_{i=1}^{N_p} \left(\frac{\pi D l_u}{1000 f_{z_i} Z V_{c_i}} + \frac{T_{cs} \pi l_u \cdot V_{c_i}^{(1/m)} \cdot f_{z_i}^{(u_v/m)} \cdot a_i^{(e_v/m)} \cdot B_i^{(n_v/m)} \cdot Z_i^{(q_v/m)} \cdot \lambda_s^{(q_v/m)}}{1000 \cdot c_v^{(1/m)} \cdot D^{(b_v/m)} \cdot (B_m \cdot B_h \cdot B_p \cdot B_t)^{(1/m)}} \right) \quad (\text{III-54})$$

III.3.4 Formulation des contraintes

a. Limitation sur l'intervalle des vitesses de coupe

La vitesse de coupe doit être comprise entre une valeur minimale et une valeur maximale qui doivent être assurées par la cinématique de la machine :

$$V_{c_{\min}} \leq V_c \leq V_{c_{\max}} \quad (\text{III-55})$$

b. Limitation sur l'intervalle des avances

L'avance par dent doit être aussi comprise entre une valeur minimale et une valeur maximale :

$$f_{z_{\min}} \leq f_z \leq f_{z_{\max}} \quad (\text{III-56})$$

c. Limitation sur l'intervalle de profondeur de passe

La profondeur de passe doit être aussi comprise entre une valeur minimale et une valeur maximale :

$$a_{\min} \leq a \leq a_{\max} \quad (\text{III-57})$$

d. Puissance de coupe

La puissance requise pour l'opération de coupe ne doit pas dépasser la puissance de la machine-outil [44]:

$$P_m \geq \frac{P_c}{\eta} \quad (\text{III-58})$$

Où P_m : la puissance nominale du moteur (kW), P_c : la puissance de coupe (kW), et η : le rendement global de Machine-outil. En fraisage, la valeur moyenne de la puissance de coupe est donnée comme suit:

$$P_c = \frac{V_c \cdot F_c}{6120} \quad (\text{III-59})$$

Où F_c est l'effort de coupe périphérique moyen (kgf). Pour le cas de fraisage ordinaire, la force de coupe est donnée par [44]:

$$F_c = C_{zp} \cdot B \cdot Z \cdot D^{b_z} \cdot a^{e_z} \cdot f_z^{u_z} \quad (\text{III-60})$$

Pour le cas de fraisage en bout, la force de coup F_c est donnée par:

$$F_c = C_{zp} \cdot B^{r_z} \cdot Z^{n_z} \cdot D^{b_z} \cdot a^{e_z} \cdot f_z^{u_z} \quad (\text{III-61})$$

Où r_z, n_z, b_z, e_z, u_z et C_{zp} : des exposants déterminées empiriquement.

e. Rigidité de l'arbre (force tonnelle)

L'arbre est soumis à une torsion de l'action de la résistance à la coupe. Par conséquent, les valeurs des paramètres de coupe devraient assurer que l'arbre est sécurisé du point de vue force :

$$F_c \leq F_s \quad (\text{III-62})$$

où F_c : est la force de coupe périphérique moyenne (kgf), F_s : la force admissible (kgf) [43]:

$$F_s = \frac{0.1k_b d_a^3}{0.08L + 0.65\sqrt{(0.25L_a)^2 + (0.5\alpha D)^2}} \quad (\text{III-63})$$

Où k_b la contrainte admissible de flexion du matériau de l'arbre (kgf/mm²), d_a : diamètre de l'arbre (mm), L longueur de l'arbre entre les appuis (mm), $\alpha = k_b / (1.3k_t)$ un coefficient dans laquelle k_t est la contrainte admissible de torsion de l'arbre (kgf/mm²).

f. Déviation de l'arbre

Le taux d'entrée choisi devrait être examiné pour assurer la déviation de l'arbre comme suit [16]:

$$F_c \leq F_d \quad (\text{III-64})$$

Où F_d : la force admissible de déviation d'arbre (kgf):

$$F_d = \frac{4Eed_a^4}{L^3} \quad (\text{III-65})$$

Où E est le module d'élasticité de la matière de l'arbre (kgf/mm²), et e : la valeur admissible de la déviation de la broche (dans des opérations d'ébauche, $e = 0,2$ mm, dans des opérations de finition, $e = 0,05$ mm).

III.3.5 Modélisation multi-passe des conditions de coupe en fraisage

Dans cette partie nous allons aborder l'optimisation des conditions d'usinage multi-passe. Ainsi, nous allons utiliser une méthode basée sur la programmation dynamique.

La valeur optimale du nombre de passes, la vitesse, l'avance et la profondeur de coupe pour chaque passe sont obtenues par le processus multi-passe. C'est un problème à quatre variables où le nombre de passe et la profondeur de coupe pour chaque passe sont déterminés à l'aide de la procédure de programmation dynamique. Tandis que la vitesse de coupe et l'avance par dent pour chaque passe sont déterminées en utilisant la méthode hybrid GA-SQP.

Le concept de la programmation dynamique est très utile pour le traitement du problème multi-passe dans lequel chaque passe est indépendante des passes antérieures, comme dans le cas des opérations de fraisage.

La variable de décision dans la programmation dynamique (dans le problème d'optimisation) est la profondeur de coupe a_i à prendre dans la $i^{\text{ème}}$ passe, qui est représentée par $a(i, j)$. La profondeur de coupe totale D_c est divisée en N sections égales. La profondeur de coupe de chaque sections est égale à:

$$a = \frac{D_c}{N} \quad (\text{III-66})$$

Elle est définie de façon arbitraire et doit être toujours plus petite que la profondeur de coupe maximale (a_{\max}) autorisée pour une machine-outil. Le nombre de passes optimal N_p est déterminé par la méthode de programmation dynamique telle que chacune des passes est constituée d'un certain nombre de section a .

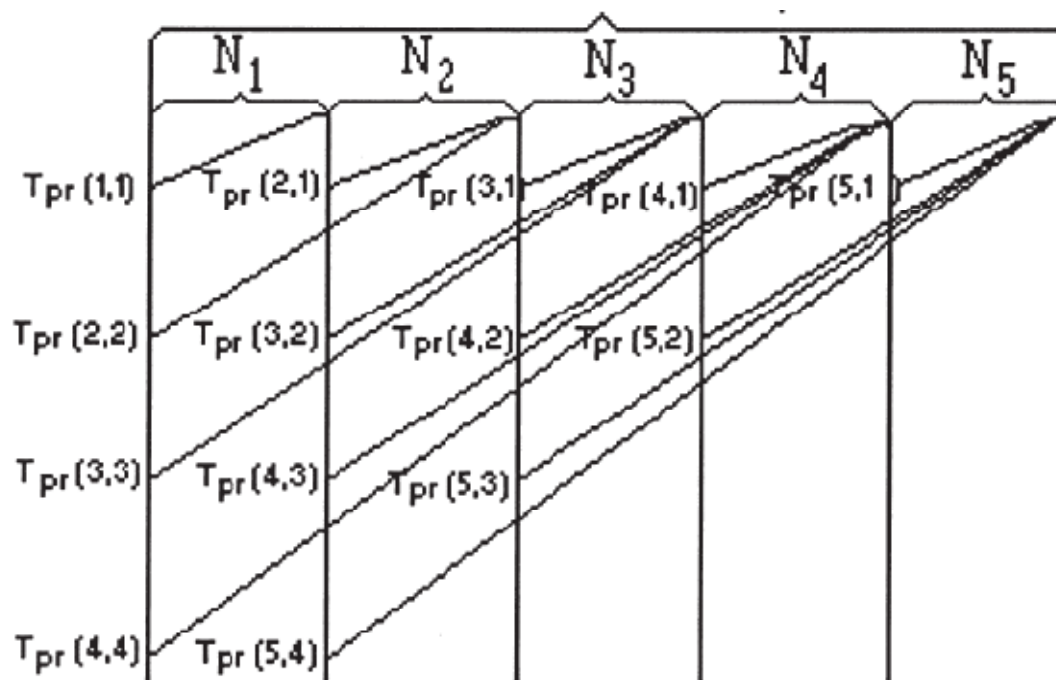


Figure III.9 Schéma représentant la programmation dynamique lorsque la profondeur de coupe totale est divisée en 5 sections. [43]

La variable de décision qui est la profondeur de passe a à enlever par passe i est représentée par $a(i, j)$ et la variable d'état qui est l'épaisseur de la pièce à la passe i est représenté par a_i . Le $a(i, j)$ signifie que la profondeur de coupe commence à la section a_i et contient j sections de profondeur égale à a .

Par conséquent la profondeur de coupe totale est divisée en N sections égales qui sont les N états de décision discrets pour la programmation dynamique.

Selon la figure (III.9), pour minimiser le temps de production de la première section à la 5^{ème} section, nous avons plusieurs solutions :

- 1/ $T_{pr}(5,4) + T_{pr}(1,1)$
- 2/ $T_{pr}(5,3) + T_{pr}(2,2)$
- 3/ $T_{pr}(5,3) + T_{pr}(2,1) + T_{pr}(1,1)$
- 4/ $T_{pr}(5,2) + T_{pr}(3,3)$,
- 5/ $T_{pr}(5,2) + T_{pr}(3,2) + T_{pr}(1,1)$
- 6/ $T_{pr}(5,2) + T_{pr}(3,1) + T_{pr}(2,2)$
- 7/ $T_{pr}(5,2) + T_{pr}(3,1) + T_{pr}(2,1) + T_{pr}(1,1)$
- 8/ $T_{pr}(5,1) + T_{pr}(4,3) + T_{pr}(1,1)$
- 9/ $T_{pr}(5,1) + T_{pr}(4,3) + T_{pr}(1,1)$
- 10/ $T_{pr}(5,1) + T_{pr}(4,4)$
- 11/ $T_{pr}(5,1) + T_{pr}(4,2) + T_{pr}(2,1) + T_{pr}(1,1)$
- 12/ $T_{pr}(5,1) + T_{pr}(4,1) + T_{pr}(3,3)$
- 13/ $T_{pr}(5,1) + T_{pr}(4,1) + T_{pr}(3,2) + T_{pr}(1,1)$
- 14/ $T_{pr}(5,1) + T_{pr}(4,1) + T_{pr}(3,1) + T_{pr}(2,2)$
- 15/ $T_{pr}(5,1) + T_{pr}(4,1) + T_{pr}(3,1) + T_{pr}(2,1) + T_{pr}(1,1)$

A partir de l'exemple précédent, on remarque que le temps nécessaire pour l'usinage de cette pièce prend de nombreuses valeurs selon le nombre de passe et la valeur de la profondeur de passe. Mais nous voulons rechercher un temps minimal dans un nombre de passes optimal.

a. Optimisation du nombre de passes

Lorsque (V_c, f_z) sont les conditions de coupe optimales pour une profondeur de passe $(x_i = i \cdot a)$, la solution optimale est atteinte, tels que des profondeurs de coupe optimales pour un minimum du temps de production.

Dans ce cas le temps de production totale est :

$$T_{pr} = T_t + \left(\frac{T_s}{N_b} + t_h + N_p \cdot t_r \right) \quad (\text{III-67})$$

Où T_t le temps de réalisation de N_p passes:

$$T_t = \sum_{i=1}^{N_p} \left(t_{c_i} + \frac{t_{cs} \times t_{c_i}}{T_i} \right) \quad (\text{III-68})$$

Le problème est de rechercher le nombre de passes, pour avoir un temps $t(x_i)$ minimal. La valeur optimale de la vitesse de coupe et de l'avance pour chaque passe ($x_i = i \cdot a$) est déterminée par la méthode hybride GA-SQP.

x_i : Profondeur de passe, ($i=1, 2, 3, \dots, N$)

N : nombre des sections égales.

a : La plus petite profondeur de passe. Où $a = \frac{D_c}{N}$ (III-69)

b. Méthodologie d'optimisation des conditions de coupe en fraisage multi-passe

Donc pour résoudre le problème, on utilise la méthodologie suivante :

1. Le problème à résoudre devient sous cette forme :

$$\sum_{i=1}^N x_i = x_1 + x_2 + x_3 + \dots + x_N = D_c \quad (\text{III-70})$$

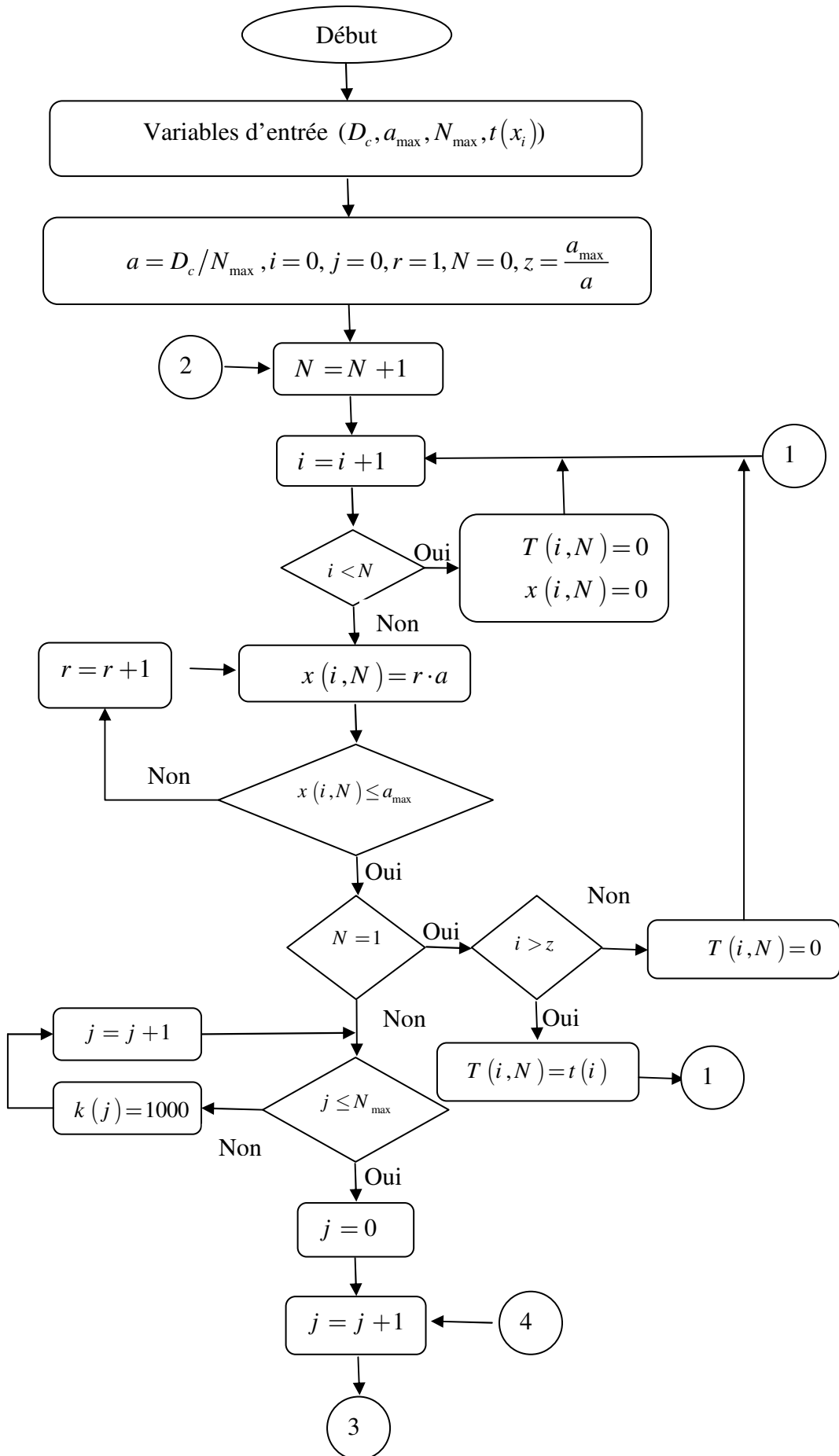
$$\text{Minimiser } T_t = \sum_{i=1}^N t(x_i)$$

Le problème est simplifié comme suit :

$$\left\{ \begin{array}{l} x_1 + x_2 + x_3 + \dots + x_N = \alpha \\ \text{Min} \sum_{i=1}^N t(x_i) = T_N(\alpha) = \text{Min}(t(x_N) + T_{N-1}(\alpha - x_N)) \\ \text{Où} \\ N \cdot a \leq \alpha \leq D_c \\ a \leq x_N \leq (\alpha - (N-1)a) \text{ et } x_N \leq a_{\max} \end{array} \right. \quad (\text{III-71})$$

Où $T_N(\alpha)$ est le temps minimal nécessaire pour usiner une section d'épaisseur égale à α (mm) avec un nombre de passe égale à N , x_N est la profondeur à usiner à chaque passe. $x_N = j \cdot a$, pour que la condition $x_N \leq a_{\max}$ soit satisfaite alors que j doit être varié de 1 à $\frac{a_{\max}}{a}$ au temps que N doit être varié de 1 à N_{\max} .

L'approche utiliser est basée sur la programmation dynamique est décrite par le diagramme de la figure (III.10).



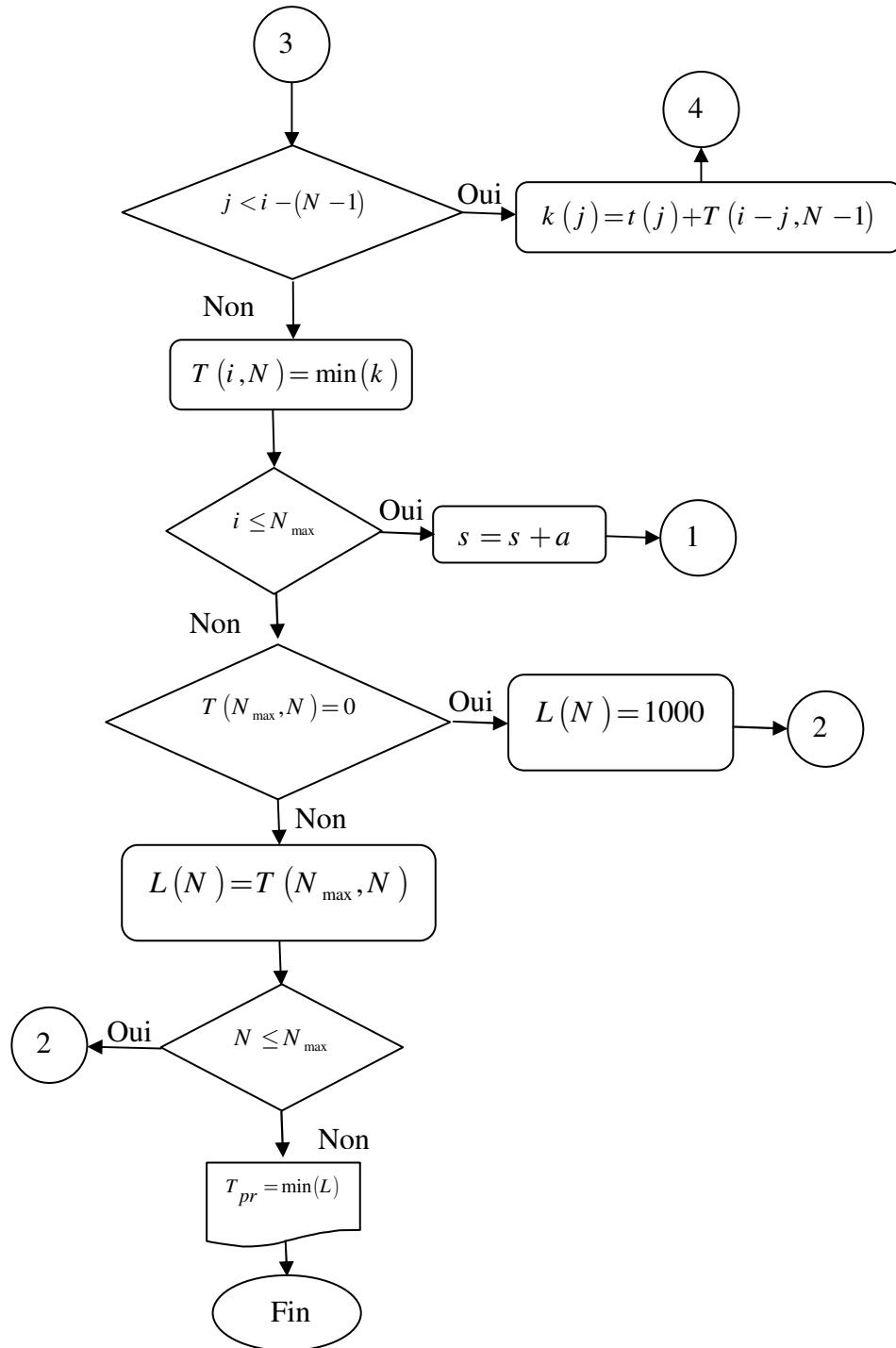


Figure III.10 : Organigramme de la méthodologie pour l’optimisation multi-passe

Pour expliquer la méthode de calcul, on prend l’exemple de la figure (III.10) tel que :

$$D_c = 5 \text{ (mm)}$$

$$a = \frac{a_T}{N_{\max}} = 1$$

$$N_{\max} = 5$$

$$a_{\max} = 4 \text{ (mm)}$$

$$j = \frac{a_{\max}}{a} = 4$$

Pour N=1 on a :

$$T_1(\alpha) = \text{Min}(t(x_1) + T_0(\alpha - x_1)) = t(x_1) \quad \text{Où } T_0(\alpha - x_1) = 0$$

$$\text{Où } 1 \leq \alpha \leq a_T, \quad 1 \leq x_1 \leq \alpha$$

$$\alpha = 1 \rightarrow T_1(1) = t(1) \rightarrow x_1 = 1$$

$$\alpha = 2 \rightarrow T_1(2) = t(2) \rightarrow x_1 = 2$$

$$\alpha = 3 \rightarrow T_1(3) = t(3) \rightarrow x_1 = 3$$

$$\alpha = 4 \rightarrow T_1(4) = t(4) \rightarrow x_1 = 4$$

Pour N=2 on a :

$$T_2(\alpha) = \text{Min}(t(x_2) + T_1(\alpha - x_2))$$

$$\text{Où } 2 \leq \alpha \leq a_T, \quad 1 \leq x_2 \leq \alpha - 1$$

$$\alpha = 2 \rightarrow T_2(2) = \text{Min}(t(x_2) + T_1(2 - x_2)) \rightarrow 1 \leq x_2 \leq 1 \Rightarrow T_2(2) = t(1) + T_1(1)$$

$$\alpha = 3 \rightarrow T_2(3) = \text{Min}(t(x_2) + T_1(3 - x_2)) \rightarrow 1 \leq x_2 \leq 2$$

$$\Rightarrow T_2(3) = \text{Min}[(t(1) + T_1(2)), (t(2) + T_1(1))]$$

$$\alpha = 4 \rightarrow T_2(4) = \text{Min}(t(x_2) + T_1(4 - x_2)) \rightarrow 1 \leq x_2 \leq 3$$

$$\Rightarrow T_2(4) = \text{Min}[(t(1) + T_1(3)), (t(2) + T_1(2)), (t(3) + T_1(1))]$$

$$\alpha = 5 \rightarrow T_2(5) = \text{Min}(t(x_2) + T_1(5 - x_2)) \rightarrow 1 \leq x_2 \leq 4$$

$$\Rightarrow T_2(5) = \text{Min}[(t(1) + T_1(4)), (t(2) + T_1(3)), (t(3) + T_1(2)), (t(4) + T_1(1))]$$

Et ainsi de suit pour tout les valeurs de N jusqu'à $N = N_{\max}$

Les résultats sont résumés dans un tableau. La forme générale de ce tableau est illustrée ci dessous :

Tableau III.1 : Tableau des résultats d'optimisation.

| | N=1 | | N=2 | | N=3 | | N=4 | | N=5 | |
|----------|-------|---------------|-------|---------------|-------|---------------|-------|---------------|-------|---------------|
| α | X_1 | $T_1(\alpha)$ | X_2 | $T_2(\alpha)$ | X_3 | $T_3(\alpha)$ | X_4 | $T_4(\alpha)$ | X_5 | $T_5(\alpha)$ |
| 1 | | | - | - | - | - | - | - | - | - |
| 2 | | | | | - | - | - | - | - | - |
| 3 | | | | | | | - | - | - | - |
| 4 | | | | | | | | | - | - |
| 5 | - | - | | | | | | | | |

Le temps minimal recherché est le minimum de $T_N(\alpha = D_c)$

Dans le cas où N_{\max} est supérieure à (a_{\max}/a) notre problème est divisé en plusieurs sous problèmes, à chaque sous problème la profondeur de coupe est inférieure ou égale à a_{\max} .

III.4 Applications

Dans cette partie, nous utilisons les méthodologies proposées précédemment, pour résoudre le problème d'optimisation des conditions de coupe en usinage multi-passe, avec la méthode hybride GA-SQP.

Afin de prouver l'efficacité de la méthode GA-SQP et de la programmation dynamique pour les processus de tournage et fraisage, on va aborder trois exemples d'application.

III.4.1 Exemple d'application 1

Nous proposons d'étudier une opération de chariotage sur une pièce brute (matériau) Les dimensions et les paramètres d'usinage sont illustrés dans le tableau ci dessous :

Tableaux III. 2 Les paramètres d'usinage [35]

| Paramètre | Valeur | Paramètre | Valeur |
|-------------|---------------|-----------------|-----------------|
| L | 203 mm | t_R | 0.13 min/ passe |
| D | 152 mm | t_h | 1.5 min/ passe |
| V_{\min} | 30 m / min | θ_{\max} | 500 °C |
| V_{\max} | 200 m / min | a_1 | 0.29 |
| f_{\min} | 0.254 mm / tr | a_2 | 0.35 |
| f_{\max} | 0.762 mm / tr | a_3 | 0.25 |
| SF_{\max} | 2 μm | K | 193.3 |
| SR_{\max} | 8 μm | t_{cs} | 0.5 min/ arrête |
| P_{\max} | 5 KW | C_0 | 0.1 u / min |
| F_{\max} | 1100 N | C_t | 0.5 u / arrête |

a. Formulation du modèle d'optimisation multi-passe

Après le remplacement des paramètres d'usinage illustrés dans le tableau III.2 nous obtenons les formules du temps et du coût de production pour une passe.

Le temps de production

Le temps de production par passe est donnée par :

$$T_u(i, j) = 96,887.V_c^{-1}(i, j).f^{-1}(i, j) + 3,465 \cdot 10^{-8} V_c^3(i, j).f^{0,16}(i, j).a^{1,4}(i, j) + 0.13 \quad (\text{III-72})$$

Le coût de production

Le coût de production par passe est donnée par :

$$C_u(i, j) = 9,688 V_c^{-1}(i, j).f^{-1} + 3,81 \cdot 10^{-8} V_c^3(i, j).f^{1,6}(i, j).a^{1,4}(i, j) + 0.013 \quad (\text{III-73})$$

Où les expressions du temps de production total et du coût de production total pour M passe sont donnés par :

$$T_u = \sum_{k=1}^M [T_u(i, j)_k] + 1.5 \quad (\text{III-74})$$

$$C_u = \sum_{k=1}^M [C_u(i, j)_k] + 0.15 \quad (\text{III-75})$$

Les contraintes

$$\left\{ \begin{array}{l} 0,0373.V_c^{(0,91)}.f^{(0,78)}.a^{(0,75)} \leq P_{\max} \\ 844.V_c^{(-0,1013)}.f^{(0,725)}.a^{(0,75)} \leq F_{\max} \\ 75.V_c^{(0,4)}.f^{(0,2)}.a^{(0,105)} \leq T_{\max} \\ 14,785.V_c^{(-1,52)}.f^{(1)}.a^{(0,25)} \leq S_{\max} \\ 0,254 \leq f \leq 0,762 \\ 30 \leq V_c \leq 200 \end{array} \right.$$

b. Résolution du problème

Nous divisons la profondeur de coupe totale $D_c = 10.16 \text{ mm}$ en 4,8 ,10 et 12 sections égales. Après la résolution du problème d'optimisation (Optimisation avec la méthode hybride GA-SQP) et avec les deux critères (temps et coût de production) pour les différentes profondeurs de coupe possibles pour des opérations d'ébauche $SR_{\max} = 8 \mu\text{m}$ et de finition $SF_{\max} = 2.5 \mu\text{m}$ les valeurs des vitesses d'avances et de coupe sont regroupées dans les tableaux III.3 et III.4

Tableau III.3 Les valeurs optimales de V_c et f (Critère du temps de production)

| a (mm) | Opération ébauche $SR_{\max} = 8\mu\text{m}$ | | Opération finition $SF_{\max} = 2.5\mu\text{m}$ | |
|----------|--|-------------|---|-------------|
| | f mm/tour | V_c m/min | f mm/tour | V_c m/min |
| 0.85 | 0.7620 | 149.9275 | 0.4956 | 185.9109 |
| 1.02 | 0.7620 | 142.9210 | 0.4633 | 183.2816 |
| 1.27 | 0.7620 | 134.9290 | 0.4274 | 180.1694 |
| 1.69 | 0.7448 | 126.6131 | 0.3846 | 176.1924 |
| 2.03 | 0.6962 | 124.8128 | 0.3595 | 173.6871 |
| 2.54 | 0.6409 | 122.6463 | 0.3310 | 170.6724 |
| 3.05 | 0.5991 | 120.9056 | 0.3094 | 168.2499 |
| 3.38 | 0.5768 | 119.9391 | 0.2979 | 166.9049 |
| 3.81 | 0.5519 | 118.8222 | 0.2850 | 165.3507 |
| 4.07 | 0.5329 | 117.3729 | 0.2782 | 164.5001 |
| 4.24 | 0.5049 | 114.8363 | 0.2740 | 163.9751 |
| 5.08 | 0.4611 | 110.6807 | 0.2563 | 161.6759 |

Tableau III.4 Les valeurs optimales de V_c et f (critère de coût de production)

| a (mm) | Opération ébauche $SR_{\max} = 8\mu\text{m}$ | | Opération finition $SF_{\max} = 2.5\mu\text{m}$ | |
|----------|--|-------------|---|-------------|
| | f mm/tour | V_c m/min | f mm/tour | V_c m/min |
| 0.85 | 0.7620 | 114.7877 | 0.4056 | 162.9633 |
| 1.02 | 0.7620 | 118.2820 | 0.3674 | 157.3365 |
| 1.27 | 0.7313 | 119.3514 | 0.3262 | 150.8359 |
| 1.69 | 0.6262 | 112.9564 | 0.2793 | 142.7598 |
| 2.03 | 0.5669 | 109.0414 | 0.2540 | 138.2035 |
| 2.54 | 0.5020 | 104.4356 | 0.2540 | 143.3932 |
| 3.05 | 0.4545 | 100.8105 | 0.2540 | 147.7742 |
| 3.38 | 0.4299 | 98.8398 | 0.2540 | 150.2924 |
| 3.81 | 0.4028 | 96.5887 | 0.2540 | 153.2819 |
| 4.07 | 0.3887 | 95.3692 | 0.2540 | 154.9553 |
| 4.24 | 0.3716 | 93.8663 | 0.2540 | 156.0017 |
| 5.08 | 0.3446 | 91.3841 | 0.2540 | 160.7090 |

Nous résolvons le problème par la programmation dynamique pour les deux critères (temps et coût de production) ou on considère que :

$$u(i, j) = C_u(i, j) \quad \text{où} \quad u(i, j) = T_u(i, j)$$

La valeur de $DMOP$ correspond à la valeur optimale maximale inférieure à $DMAXP$

Etape1 : $N = 4$

En premier lieu on divise la profondeur de coupe totale en 4 sections égales avec :

La profondeur de passe optimal $DMOP = 3.81 \text{ mm}$

La profondeur de passe maximal $DMAXP = 5.08 \text{ mm}$

Après exécution du programme on trouve les résultats suivants :

$$uop(1) = u(1,1)$$

$$uop(2) = uop(1) + u(2, 1)$$

$$uop(3) = uop(1) + u(3,2)$$

$$uop(4) = uop(2) + u(4,2)$$

$$\text{Donc } uop(4) = u(1,1) + u(2,1) + u(4,2)$$

Et le nombre de passe optimal est $M = 3$, deux passes pour l'ébauche et une passe pour la finition avec les profondeurs de passes : 5.08mm, 2.54mm et 2.54mm.

Tableau III.5 Paramètres optimaux d'usinage pour $N=4$ (Critère du coût de production)

| N° de passe | a (mm) | f mm / tour | V_c m / min | C_u \$ | T_u (min) |
|------------------------------------|----------|---------------|---------------|----------|-------------|
| 1 | 5.08 | 0.3446 | 91.3841 | 0.5232 | 3.2038 |
| 2 | 2.54 | 0.5020 | 104.4356 | 0.3304 | 2.0424 |
| 3 | 2.54 | 0.2540 | 143.3932 | 0.5923 | 2.9941 |
| Cout et temps de production totale | | | | 1.5959 | 9.7403 |

Tableau III.6 Paramètres optimaux d'usinage pour $N=4$ (Critère du temps de Production)

| N° de passe | a (mm) | f mm / tour | V_c m / min | C_u \$ | T_u (min) |
|------------------------------------|----------|---------------|---------------|----------|-------------|
| 1 | 5.08 | 0.4611 | 110.6807 | 0.6054 | 2.2791 |
| 2 | 2.54 | 0.6409 | 122.6463 | 0.3659 | 1.5338 |
| 3 | 2.54 | 0.3310 | 170.6724 | 0.7455 | 2.3030 |
| Cout et temps de production totale | | | | 1.8668 | 7.6159 |

Etape 2 : $N = 8$

La profondeur de passe optimale $DMOP = 3.81 \text{ mm}$

La profondeur de passe maximale $DMAXP = 5.08 \text{ mm}$

En exécutant le programme on obtient:

$$uop(1) = u(1, 1)$$

$$uop(2) = u(2, 2)$$

$$uop(3) = u(3, 3)$$

$$uop(4) = uop(1) + u(4, 3)$$

$$uop(5) = uop(1) + u(5, 4)$$

$$uop(6) = uop(5) + u(6, 1)$$

$$uop(7) = uop(5) + u(7, 2)$$

$$uop(8) = uop(4) + u(8, 4)$$

Donc $uop(8) = u(1,1) + u(4,3) + u(8, 4)$.

Et le nombre de passes optimal est $M = 3$, deux passes pour l'ébauche et une passe pour la finition avec les profondeurs de passes : 5.08 mm, 3.81 mm et 1.27 mm.

Tableau III.7 Paramètres optimaux d'usinage pour $N=8$ (Critère du coût de production)

| N° de passe | a (mm) | f mm / tour | V_c m / min | C_u \$ | T_u (min) |
|------------------------------------|----------|---------------|---------------|----------|-------------|
| 1 | 5.08 | 0.3446 | 91.3841 | 0.5232 | 3.2038 |
| 2 | 3.81 | 0.4028 | 96.5887 | 0.4333 | 2.6623 |
| 3 | 1.27 | 0.3262 | 150.8359 | 0.3571 | 2.2030 |
| Cout et temps de production totale | | | | 1.4636 | 9.5691 |

Tableau III.8 Paramètres optimaux d'usinage pour $N=8$ (Critère du temps de Production)

| N° de passe | a (mm) | f mm / tour | V_c m / min | C_u \$ | T_u (min) |
|------------------------------------|----------|---------------|---------------|----------|-------------|
| 1 | 5.08 | 0.4611 | 110.6807 | 0.6054 | 2.2791 |
| 2 | 3.81 | 0.5519 | 118.8222 | 0.5131 | 1.8604 |
| 3 | 1.27 | 0.4274 | 180.1694 | 0.4044 | 1.6107 |
| Cout et temps de production totale | | | | 1.6729 | 7.2502 |

Etape 3 : $N = 10$

La profondeur de passe optimale $DMOP = 3.048 \text{ mm}$

La profondeur de passe maximale $DMAXP = 5.08 \text{ mm}$

En exécutant le programme on aura :

$$uop(1) = u(1, 1)$$

$$uop(2) = u(2, 2)$$

$$uop(3) = u(3, 3)$$

$$uop(4) = uop(1) + u(4, 3)$$

$$uop(5) = uop(1) + u(5, 4)$$

$$uop(6) = uop(1) + u(6, 5)$$

$$uop(7) = uop(3) + u(7, 4)$$

$$uop(8) = uop(3) + u(8, 5)$$

$$uop(9) = uop(5) + u(9, 4)$$

$$uop(10) = uop(5) + u(10, 5)$$

$$\text{Donc } uop(10) = u(1,1) + u(5,4) + u(10, 5)$$

Le nombre de passes optimal est $M = 3$, deux passes pour l'ébauche et une passe pour la finition avec les profondeurs de passes : 5.08 mm, 4.07 mm et 1.02 mm.

Tableau III.9 Paramètres optimaux d'usinage pour $N=10$ (Critère du coût de production)

| N° de passe | a (mm) | f mm / tour | V_c m / min | C_u \$ | T_u (min) |
|------------------------------------|----------|---------------|---------------|----------|-------------|
| 1 | 5.08 | 0.3446 | 91.3841 | 0.5232 | 3.2038 |
| 2 | 4.07 | 0.3887 | 95.3692 | 0.4526 | 2.7787 |
| 3 | 1.02 | 0.3674 | 157.3365 | 0.3068 | 1.9004 |
| Cout et temps de production totale | | | | 1.4344 | 9.3829 |

Tableau III.10 Paramètres optimaux d'usinage pour $N=10$ (Critère du temps de production)

| N° de passe | a (mm) | f mm / tour | V_c m / min | C_u \$ | T_u (min) |
|------------------------------------|----------|---------------|---------------|----------|-------------|
| 1 | 5.08 | 0.4611 | 110.6807 | 0.6054 | 2.2791 |
| 2 | 4.07 | 0.5329 | 117.3729 | 0.5365 | 1.9388 |
| 3 | 1.02 | 0.4633 | 183.2816 | 0.3363 | 1.4472 |
| Cout et temps de production totale | | | | 1.6282 | 7.1651 |

Etape 4: $N = 12$

La profondeur de passe optimale $DMOP = 3.38$ mm

La profondeur de passe maximale $DMAXP = 5.08$ mm

En exécutant le programme on obtient :

$$uop(1) = u(1, 1)$$

$$uop(2) = u(2, 2)$$

$$uop(3) = u(3, 3)$$

$$uop(4) = u(4, 4)$$

$$uop(5) = uop(1) + u(5, 4)$$

$$uop(6) = uop(1) + u(6, 5)$$

$$uop(7) = uop(1) + u(7, 6)$$

$$uop(8) = uop(2) + u(8, 6)$$

$$uop(9) = uop(3) + u(9, 6)$$

$$uop(10) = uop(5) + u(10, 5)$$

$$uop(11) = uop(6) + u(11, 5)$$

$$uop(12) = uop(6) + u(12, 6)$$

$$\text{Donc : } uop(12) = u(1,1) + u(6,5) + u(12, 6)$$

Et le nombre de passes optimal est $M = 3$, deux passes pour l'ébauche et une passe pour la finition avec les profondeurs de passes : 5.08 mm, 4.24 mm et 0.85 mm.

Tableau III.11 Paramètres optimaux d'usinage pour $N=12$ (Critère du coût de production)

| N° de passes | a (mm) | f mm / tour | V_c m / min | C_u \$ | T_u (min) |
|------------------------------------|----------|---------------|---------------|----------|-------------|
| 1 | 5.08 | 0.3446 | 91.3841 | 0.5232 | 3.2038 |
| 2 | 4.24 | 0.3801 | 94.6208 | 0.4650 | 2.8639 |
| 3 | 0.85 | 0.4056 | 162.9633 | 0.2705 | 1.6817 |
| Cout et temps de production totale | | | | 1.4087 | 9.2494 |

Tableau III.12 Paramètres optimaux d'usinage pour $N=12$ (Critère du temps de production)

| N° de passes | a (mm) | f mm / tour | V_c m / min | C_u \$ | T_u (min) |
|------------------------------------|----------|---------------|---------------|----------|-------------|
| 1 | 5.08 | 0.4611 | 110.6807 | 0.6054 | 2.2791 |
| 2 | 4.24 | 0.5188 | 116.1078 | 0.5486 | 1.9975 |
| 3 | 0.85 | 0.4956 | 185.9109 | 0.2896 | 1.3268 |
| Cout et temps de production totale | | | | 1.5936 | 7.1034 |

Alors, le coût et le temps de production sont égaux respectivement à : 1.4087 \$, 9.2494 min (pour le critère de coût) et 1.5936 \$, 7.1034 min (pour le critère de temps).

c. Interprétation des résultats

L'influence du nombre de sections N utilisé par la technique de la programmation dynamique est bien lisible en comparant les résultats des tableaux précédents pour les deux critères et pour $N= 4, 8, 10, 12$

En augmentant le nombre de sections N de 4 à 8 il en résulte une diminution du coût et du temps de production total pour les deux critères respectivement de 9.03 % , 1.94 % et 10.3% , 4.8% de même l'augmentation du nombre de sections de 8 à 10 provoque une diminution du coût et du temps de production totale d'environ de 2 % et 1.94 % (pour le critère du coût de production), 2.67% et 1.17% (pour le critère du temps de production)

En fin si on augmente le nombre de sections de 10 à 12 on obtient une légère diminution du coût et du temps total.

Finalement on constate que le nombre de passes optimal et les conditions d'usinages correspondantes en respectant toutes les limitations des contraintes pour une profondeur de coupe totale égale à 10.16 mm est obtenu avec le nombre de sections N égale à 12.

d. Comparaison des résultats

L'efficacité de la stratégie utilisée pour l'optimisation des conditions de coupe en tournage multi-passe est illustrée par une comparaison avec d'autres méthodes dans la figure III.4.

La stratégie d'optimisation est appliquée avec les mêmes données utilisées par J.S. Agapiou [35], qui a utilisé une stratégie basée sur la technique de la programmation dynamique et les conditions de coupe sont optimisées grâce à la méthode du Simplex de Nelder-Mead.

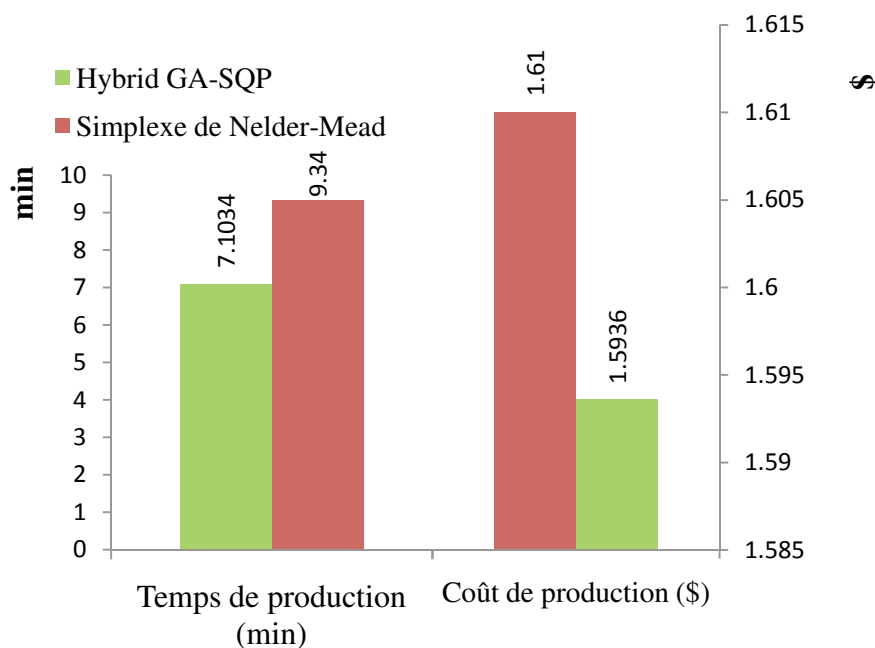


Figure III.11 : Comparaison entre les résultats obtenus par Agapiou [35] et les résultats obtenus par la méthode Hybrid GA-SQP

III.4.2 Exemple d'application 2

Nous résolvons cet exemple pour valider la méthode hybride GA-SQP utilisé pour l'optimisation d'une opération de tournage multi-passe. Les paramètres utilisés sont regroupés dans le tableau III.13.

Tableaux III. 13 Les paramètres d'usinage [46]

| Caracteristiques de la machine outils | | | | | |
|--|--------|------------------------|--------------------|--------------------------------|--------------------|
| Paramètre | Valeur | Paramètre | Valeur | Paramètre | Valeur |
| $V_{ce_{max}}$ (m/min) | 500 | $V_{ce_{min}}$ (m/min) | 50 | $f_{e_{max}}$ (mm/rev) | 0.9 |
| $f_{e_{min}}$ (mm/rev) | 0.1 | $d_{e_{max}}$ (mm) | 3.0 | $d_{e_{min}}$ (mm) | 1.0 |
| $V_{cf_{max}}$ (m/min) | 500 | $V_{cf_{min}}$ (m/min) | 50 | $f_{f_{max}}$ (mm/rev) | 0.9 |
| $f_{f_{min}}$ (mm/rev) | 0.1 | $d_{f_{max}}$ (mm) | 3.0 | $d_{f_{min}}$ (mm) | 1.0 |
| η | 0.85 | k_0 (\$/min) | 0.5 | t_c (min/ piece) | 0.75 |
| t_e (min/edge) | 1.5 | P_m (kW) | 5 | F_u (Kgf) | 200 |
| Caracteristiques de l'outil et de la pièce | | | | | |
| Matériau de l'outil: carbure métallique (P40) / Matériau de la pièce: acier (C 35) | | | | | |
| D (mm) | 50 | L (mm) | 300 | d_t (mm) | 6 |
| P | 5 | q | 1.75 | r | 0.75 |
| k_1 | 108 | μ | 0.75 | ν | -1 |
| λ | 2 | ν | 0.95 | C_m | 132 |
| m | 0.4 | p | 0.2 | q | 0.105 |
| R (mm) | 1.2 | C_0 | 6×10^{11} | h_1 | 7×10^{-4} |
| h_2 | 0.3 | T_{min} (min) | 25 | T_{max} (min) | 45 |
| SC | 140 | R_{max} (μm) | 10 | θ_{max} ($^{\circ}C$) | 1000 |
| k_3 | 1.0 | k_4 | 2.5 | k_5 | 1.0 |
| C_t (\$/edge) | 2.5 | | | | |

III.4.3 Résultats et discussion

Plusieurs générations de l'algorithme génétique sont effectuées afin d'identifier une bonne estimation initiale pour la programmation quadratique séquentielle.

Les résultats obtenus sont regroupés dans le tableau III.14.

Tableaux III. 14 Paramètres optimales d'usinage

| N_p | Ebauche | | | Finition | | | C_u (\$) |
|-------|--------------------|------------------|------------|--------------------|------------------|------------|------------|
| | V_{ce} (m / min) | f_e (mm / rev) | d_e (mm) | V_{cf} (m / min) | f_f (mm / rev) | d_f (mm) | |
| 1 | 94.4640 | 0.8660 | 3.0000 | 162.2890 | 0.2580 | 3.0000 | 1.9308 |
| 2 | 182.9710 | 0.4520 | 2.4996 | 217.3229 | 0.1794 | 1.0009 | 2.5840 |
| 3 | 145.6160 | 0.9000 | 1.6670 | 191.3630 | 0.2580 | 1.0000 | 2.6450 |
| 4 | 157.2560 | 0.9000 | 1.2430 | 171.6070 | 0.2580 | 1.0260 | 3.1230 |
| 5 | 166.5110 | 0.9000 | 1.0000 | 191.3630 | 0.2580 | 1.0000 | 3.4585 |

Nous constatons que la valeur minimale du coût est 1,9308 \$ pour $n=1$. Les performances de la méthode hybride GA-SQP sont illustrées par une comparaison avec d'autres méthodes dans le tableau III.15.

La méthode hybride GA-SQP est appliquée avec le même modèle et les mêmes données utilisés dans les références [38, 45, 46, 47, 48 et 49], mais les auteurs de ces références ont utilisé d'autres méthodes.

Tableaux III. 15 Paramètres optimums d'usinage pour les différentes méthodes

| Les algorithms | coût (\$) |
|------------------------|-----------|
| FEGA [46] | 2.3084 |
| SA/SP [38] | 2.2795 |
| PSO [47] | 2.2721 |
| GA [48] | 2.2538 |
| SS [49] | 2.0754 |
| GA-based approach [45] | 2.0298 |
| GA-SQP | 1.9308 |

Selon le tableau III.15, on peut dire que la méthode hybride GA-SQP donne de meilleurs résultats que les autres méthodes. Cette méthode est plus efficace pour résoudre le problème d'optimisation des conditions de coupe en tournage multi-passe.

III.4.4 Exemple d'application 3

L'application concerne l'opération de fraisage (plan) d'une pièce dont les caractéristiques sont regroupées dans les tableaux III.16 et III.17.

Tableaux III. 16 Les paramètres d'usinage [43]

| Caractéristiques de la machine | | | |
|--------------------------------|----------|------------------------------|--------|
| Paramètre | Valeur | Paramètre | Valeur |
| P_m (kw) | 5.5 | E (kgf / mm ²) | 20.387 |
| η | 0.7 | L_a (mm) | 210 |
| k_t (kgf / mm ²) | 12.23 | d_a (mm) | 27 |
| k_b (kgf / mm ²) | 14.27 | N_b | 100 |
| T_s (min) | 10 | t_r (min) | 0.1 |
| t_h (min) | 1.5 | t_{cs} (min) | 5 |
| Caractéristiques de la pièce | | | |
| Paramètre | Valeur | Paramètre | Valeur |
| l_u (mm) | 160 | D_c (mm) | 5 |
| B (mm) | 50 | | |
| Caractéristiques de l'outil | | | |
| Paramètre | Valeur | Paramètre | Valeur |
| D (mm) | 63 | Z | 8 |
| Paramètres d'usinage | | | |
| Paramètre | Valeur | Paramètre | Valeur |
| V_c min (mm / min) | 6.234 | V_c max (mm / min) | 395.84 |
| f_z min (mm / min) | 0.000857 | f_z max (mm / dent) | 3.571 |
| a min (mm) | 0.5 | a max (mm) | 4 |

Tableaux III. 17 Les paramètres d'usinage (suite) [43]

| Les constants de la durée de vie | | | |
|----------------------------------|--------|-------------|--------|
| Paramètre | Valeur | Paramètre | Valeur |
| B_m | 1 | B_h | 1 |
| B_p | 0.8 | B_t | 0.8 |
| m | 0.33 | u_v | 0.4 |
| b_v | 0.45 | e_v | 0.3 |
| r_v | 0.1 | n_v | 0.1 |
| q_v | 0 | λ_s | 30° |
| Les constants de la force | | | |
| Paramètre | Valeur | Paramètre | Valeur |
| b_z | -0.86 | e_z | 0.86 |
| u_z | 0.72 | C_{zp} | 68.2 |

a. Formulation du modèle d'optimisation multi-passe**La fonction objectif**

$$T_{pr} = \sum_{i=1}^{N_p} \left(3,958 f_{z_i}^{-1} \times V_{c_i}^{-1} + 3,365 \times 10^{-5} V_{c_i}^{2,03} \times f_{z_i}^{0,212} \times a_i^{0,909} \right) + (1,6 + 0.1N_p) \quad (\text{III-76})$$

Les contraintes

$$6,234 \leq v_c \leq 395,84 \quad (\text{III-77})$$

$$0.000875 \leq f_z \leq 3.571 \quad (\text{III-78})$$

$$0.5 \leq a \leq 4 \quad (\text{III-79})$$

$$0.126 V_c \cdot f_z^{0.72} \cdot a^{0.86} \leq 3.85 \quad (\text{III-80})$$

$$773.413 f_z^{0.72} \cdot a^{0.86} \leq 505.596 \quad (\text{III-81})$$

$$773.413 f_z^{0.72} \cdot a^{0.86} \leq 0.935 \quad (\text{III-82})$$

b. Résolution du problème

Nous allons diviser la profondeur de coupe totale $D_c = 5\text{mm}$ en 3, 5 et 8 sections égales. Après la résolution du problème d'optimisation avec la méthode hybride GA-SQP nous avons opté à des résultats qui sont présentés dans le tableau III.18:

Tableau III. 18 Les valeurs optimales de V_c , et f_z

| N | $a(\text{mm})$ | $V_c (\text{m} / \text{min})$ | $f_z (\text{mm} / \text{dent})$ |
|---|----------------|-------------------------------|---------------------------------|
| 3 | 1.667 | 46.942 | 0.3 |
| | 3.333 | 46.798 | 0.131 |
| 5 | 1 | 46.74 | 0.554 |
| | 2 | 46.883 | 0.242 |
| | 3 | 46.756 | 0.149 |
| | 4 | 46.751 | 0.105 |
| 8 | 0.625 | 43.469 | 0.971 |
| | 1.25 | 46.942 | 0.424 |
| | 1.875 | 46.816 | 0.26 |
| | 2.5 | 46.753 | 0.185 |
| | 3.125 | 45.92 | 0.146 |
| | 3.75 | 46.821 | 0.114 |

Premier cas : $N = 3$

En premier lieu nous avons divisé la profondeur de coupe totale en 3 sections égales. Les résultats obtenus après exécution du programme sont présentés dans le tableau III.19 :

Tableau III.19 Résultats de la programmation dynamique pour (N=3)

| α | N=1 | | N=2 | | N=3 | |
|----------|-------|---------------|--------------|---------------|-------|---------------|
| | X_1 | $T_1(\alpha)$ | X_2 | $T_2(\alpha)$ | X_3 | $T_3(\alpha)$ |
| 1.67 | 1.67 | 0.383 | – | – | – | – |
| 3.33 | 3.33 | 0.802 | 1.67 | 0.766 | – | – |
| 5 | – | – | 3.33 et 1.67 | 1.185 | 1.67 | 1.149 |

Le temps minimale est le Min de $T_N (\alpha = D_c)$

Dans ce cas T_{pr}

Tableau III.20 Paramètres optimales de coupe pour N= 3

| Nombre de passes | a (mm) | V_c (m / min) | f_z (mm / dent) | T_{pr} (min) |
|------------------|----------|-----------------|-------------------|----------------|
| 3 | 1.67 | 46.942 | 0.3 | 0.383 |

Donc le temps de production total est :

$$T_{pr} = (0.383 + 0.383 + 0.383) + 1.6 + 3 \times 0.1 = 3.049 \text{ min}$$

Deuxième cas: $N = 5$

Dans ce cas nous avons divisé la profondeur de coupe totale en 5 sections égales. Les résultats obtenus après exécution du programme sont présentés dans le tableau III.21:

Tableau III.21 Résultats de la programmation dynamique pour (N=5)

| α | N=1 | | N=2 | | N=3 | | N=4 | | N=5 | |
|----------|-------|---------------|-------|---------------|-------|---------------|-------|---------------|-------|---------------|
| | X_1 | $T_1(\alpha)$ | X_2 | $T_2(\alpha)$ | X_3 | $T_3(\alpha)$ | X_4 | $T_4(\alpha)$ | X_5 | $T_5(\alpha)$ |
| 1 | 1 | 0.225 | – | – | – | – | – | – | – | – |
| 2 | 2 | 0.464 | 1, 1 | 0.45 | – | – | – | – | – | – |
| 3 | 3 | 0.717 | 2, 1, | 0.689 | 1 | 0.675 | – | – | – | – |
| 4 | 4 | 0.981 | 2, 2 | 0.928 | 2, 1 | 0.914 | 1 | 0.9 | – | – |
| 5 | – | – | 3, 2 | 1.181 | 2, 1 | 1.153 | 2, 1 | 1.139 | 1 | 1.25 |

Dans ce cas T_{pr}

Tableau III.22 Paramètres optimales de coupe pour N= 5

| Nombre de passes | a (mm) | V_c (m / min) | f_z (mm / dent) | T_{pr} (min) |
|------------------|----------|-----------------|-------------------|----------------|
| 5 | 1 | 46.741 | 0.554 | 0.225 |

Donc le temps de production total est :

$$T_{pr} = (0.225 + 0.225 + 0.225 + 0.225 + 0.225) + 1.6 + 5 \times 0.1 = 3.35 \text{ min}$$

Troisième cas : $N = 8$

Dans ce cas nous avons divisé la profondeur de coupe totale en 8 sections égales. Les résultats obtenus après exécution du programme sont présentés dans le tableau III.23:

Tableau III.23 Résultats de la programmation dynamique pour (N=8)

| | α | 0.625 | 1.25 | 1.875 | 2.5 | 3.125 | 3.75 | 0.625 | 1.25 |
|-----|---------------|-------|-------|---------------|---------------|-------|-------|-------|-------|
| N=1 | $T_1(\alpha)$ | 0.139 | 0.283 | 0.433 | 0.589 | 0.739 | 0.913 | 0.139 | 0.283 |
| | X_1 | 0.625 | 1.25 | 1.875 | 2.5 | 3.125 | 3.75 | 0.625 | 1.25 |
| N=2 | $T_2(\alpha)$ | – | 0.278 | 0.422 | 0.566 | 0.716 | 0.866 | – | 0.278 |
| | X_2 | – | 0.625 | 1.25 0.625 | 1.25 | 2.5 | 0.625 | – | 0.625 |
| N=3 | $T_3(\alpha)$ | – | – | 0.417 | 0.561 | 0.705 | 0.844 | – | – |
| | X_3 | – | – | 0.625 | 1.25 0.625 | | | – | – |
| N=4 | $T_4(\alpha)$ | – | – | – | 0.556 | 0.700 | 0.844 | – | – |
| | X_4 | – | – | – | 0.625 | | | – | – |
| N=5 | $T_5(\alpha)$ | – | – | – | – | 0.695 | 0.839 | – | – |
| | X_5 | – | – | – | – | 0.625 | | – | – |
| N=6 | $T_6(\alpha)$ | – | – | – | – | – | 0.834 | – | – |
| | X_6 | – | – | – | – | – | 0.625 | – | – |
| N=7 | $T_7(\alpha)$ | – | – | – | – | – | – | – | – |
| | X_7 | – | – | – | – | – | – | – | – |
| N=8 | $T_8(\alpha)$ | – | – | – | – | – | – | – | – |
| | X_8 | – | – | – | – | – | – | – | – |

Dans ce cas, le problème est subdivisé en deux sous problème, où le premier ayant une profondeur de passe égale à 3.75 mm et le deuxième égale à 1.25 mm.

Tableau III.24 Paramètres optimales de coupe pour N= 8

| Nombre .de passes | a (mm) | V_c (m / min) | f_z (mm / dent) | T_{pr} (min) |
|----------------------|----------|-----------------|-------------------|----------------|
| 8 | 0.625 | 43.469 | 0.971 | 0.139 |

Le temps de production pour le premier sous problème est :

$$T_{pr1} = (0.139 + 0.139 + 0.139 + 0.139 + 0.139 + 0.139) + 1.6 + 6 \times 0.1 = 3.034 \text{ min}$$

Le temps de production pour le deuxième sous problème est :

$$T_{pr2} = (0.139 + 0.139) + 1.6 + 2 \times 0.1 = 2.078 \text{ min}$$

Donc le temps de production total est :

$$T_{pr} = T_{pr1} + T_{pr2} = 3.034 + 2.078 = 5.112 \text{ min}$$

c. Interprétation des résultats

- L'influence du nombre de sections N utilisé par la technique de la programmation dynamique est observé en comparant les résultats des tableaux précédents pour N= 3, 5 et 8.
- En augmentant le nombre de sections N de 3 à 8 il en résulte une augmentation du temps de production.
- Finalement on constate que le nombre de passes optimal et les conditions d'usinages correspondantes en respectant toutes les limitations des contraintes pour une profondeur de coupe totale égale à 5 mm sont obtenues avec le nombre de sections N égale à 3.

d. Comparaison des résultats

L'efficacité de la stratégie utilisée pour l'optimisation des conditions de coupe en fraisage multi-passe est illustrée par une comparaison avec d'autre méthode dans la figure III.12.

La stratégie d'optimisation à été appliquée avec les mêmes données utilisées dans les références [43, 44], mais les auteurs de ces références ont utilisés d'autres méthodes.

Les auteurs de la référence [43] ont utilisés une combinaison de la programmation non linéaire et de la programmation géométrique pour déterminer les valeurs optimums des conditions de coupe.

Les auteurs de la référence [44] ont utilisés trois méthodes pour déterminer les conditions de coupe optimums : colonies de Fourmies artificielles (ABC), essaim de particules (PSO), et le recuit simulé (SA).

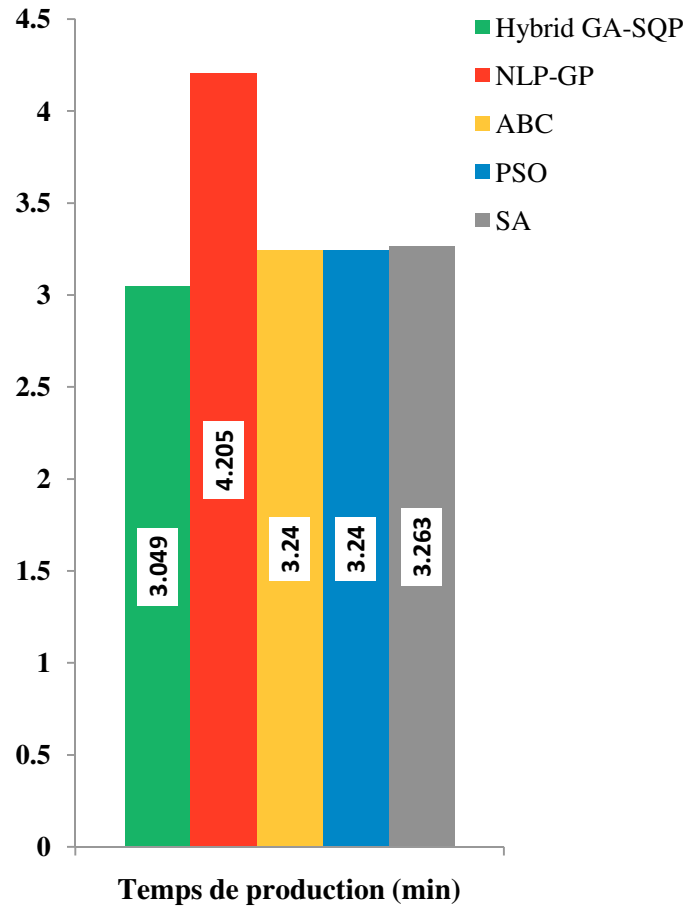


Figure III.12 : Comparaison entre les résultats obtenus par la méthode Hybrid GA-SQP et les résultats obtenus avec d'autres techniques.

III.5 Conclusion

Dans ce chapitre, nous avons traité le problème de minimisation du temps et du coût de production en cherchant à trouver les conditions de coupe optimales qui admettent un usinage parfait. Pour atteindre cet objectif nous avons utilisé la programmation dynamique et la méthode hybride GA-SQP pour l'optimisation multi-passe des conditions de coupe en tournage et en fraisage.

Conclusion Générale

Conclusion Générale et Perspectives

L'objectif de ce travail concerne la mise en place d'une méthodologie permettant de résoudre des problèmes d'optimisation couramment rencontrés dans la production industrielle. Cette méthodologie prend en compte le développement d'outils de modélisation et l'utilisation efficace des méthodes d'optimisation.

La contribution principale de notre travail se situe dans la proposition d'une stratégie d'optimisation, adaptée aux caractéristiques des procédés d'usinage, dans le but d'obtenir les conditions de coupe optimales en tournage et en fraisage. Cette stratégie inclut trois étapes :

- le choix de la méthode de résolution la plus adéquate. Ce choix a été effectué sur la base d'une comparaison entre quatre méthodes d'optimisation : la méthode du gradient conjuguée (GC), la programmation quadratique séquentielle (SQP), l'algorithme génétique (AG) et la méthode d'essaim particulaire (PSO).
- la construction du problème d'optimisation des conditions de coupe,
- la résolution des problèmes d'optimisation avec la méthode choisie

Dans une première étape, nous avons passé en revue cinq algorithmes d'optimisation, et l'ensemble des algorithmes ont été implémentés sous MATLAB. Leurs principes sont détaillés, puis ils sont confrontés à cinq fonctions test. Une méthode a été sélectionnée comme la plus adaptée aux problèmes de production industrielle.

En deuxième étape, nous avons utilisé les lois d'optimisation afin de trouver les modèles mathématiques des problèmes d'usinage multi-passe pour le processus de tournage et de fraisage.

Pour la modélisation du problème d'optimisation multi-passe nous avons proposé deux méthodes en tournage et une méthode en fraisage, la première méthode utilisée pour la modélisation en tournage et celle utilisée pour la modélisation en fraisage ont été effectuées selon la programmation dynamique.

Dans la troisième étape, la résolution des modèles mathématiques issues de la première étape a été effectuée à l'aide des programmes implémentés en langage MATLAB. Ces programmes ont été appliqués sur trois exemples d'application.

L'une des étapes importantes dans l'exécution des programmes consiste à sélectionner un certain nombre de sections appropriée pour le problème, puis prendre un nombre plus élevé, alors le temps d'exécution augmente proportionnellement avec le nombre de sections.

La valeur de la fonction objective doit être sélectionnée toujours en fonction de la profondeur totale.

Les valeurs obtenues ont été comparés avec ceux de la littérature pour approuver l'efficacité et la fiabilité de la méthodologie d'optimisation proposée. Les résultats ont montré que l'approche multi-passes pour l'optimisation des conditions de coupe est efficace. Il a été également reconnu que dans les opérations multi-passes, il est toujours préférable d'utiliser des profondeurs de sous passe inégales dans chaque passe au lieu d'utiliser des profondeurs de sous passe égales pour toutes les passes.

Or, les limitations de notre démarche sont :

- Nous avons réalisé l'optimisation d'une opération unitaire sans compter que, dans le processus de fabrication mécanique, cette dernière ne forme qu'une partie d'un ensemble de procédés enchaînés.
- Nous notant aussi que nous n'avons pas validé nos résultats expérimentalement.

En synthèse, le meilleur moyen pour évaluer la pertinence et la conformité des résultats est de réaliser une validation expérimentale. Malheureusement, cette validation n'a pas été possible dans notre cas.

Ce travail, qui n'est pas une fin en soi, s'inscrit dans un des axes de recherche du laboratoire de recherche en productique (LRP) de l'université de Batna. Pour son développement, nos modèles et nos résultats pourront être utilisés lors des études ultérieures (de l'optimisation d'usinage complexe, par exemple). De plus, les stratégies d'optimisations développées dans ce travail, peuvent être appliquées à d'autres procédés d'usinage.

Dans un travail futur, il est souhaitable de mener une optimisation jusqu'à sa validation expérimentale. Ceci permettra d'analyser l'impact des incertitudes sur la justesse des résultats, ainsi de tester si notre étude donne une information réaliste.

Les perspectives de ce travail au niveau de l'élaboration du problème sont différentes, il reste des questions à éclairer concernant la meilleure façon de définir le critère d'optimisation, et la manière de traiter les problèmes multicritères.

De plus, une réflexion sur la façon de réduire les incertitudes, s'impose, pour donner des résultats plus réalistes.

Références Bibliographiques

Références Bibliographiques

- [1] **A. Collignan**, Méthode D'optimisation et D'aide a la Décision en Conception Mécanique : Application a une Structure Aéronautique, Thèse de Doctorat, Université Bordeaux 1, 2011
- [2] **A. Olmos Perez**, Contribution à l'optimisation de la conduite des procédés alimentaires, Thèse de Doctorat, Ecole Nationale Supérieure Des Industries Agricoles Et Alimentaires, 2003
- [3] **M. Z. Es-sadek**, Contribution a l'Optimisation Globale. Approche Déterministe et Stochastique et Application, Université Mohammed V - Agdal Institut National des Sciences, École Mohammadia d'Ingénieurs, 2009
- [4] **P.E. Gill, W. Murray, M.H. Wright**, Practical Optimization, Academic Press, 401 pp ,London, (1981)
- [5] **S. Mottelet**, Optimisation Non-linéaire -RO04/TI07, Université de Technologie de Compiègne, Printemps 2003
- [6] **I. Charon, O. Hurdy**, Optimisation Non Linéaire, Ecole Nationale Supérieure des Télécommunications, France 2007
- [7] **J. S. Arora, A. 1. Chahande, J. K. Paeng**, Multiplier Methods For Engineering Optimization, International Journal for Numerical Methods Engineering, 32, 1485-1525 (1991).
- [8] **A. D. Belegundu, J. S. Arora**, Potential of Transformation Methods in Optimal Design, AIAA Journal, 19, 10, 1372-1374 (1981).
- [9] **A. D. Belegundu, J. S. Arora**, A Computational Study of Transformation Methods for Optimal Design, AIAA Journal, 22, 4, 535-542 (1984).
- [10] **G. N. Vanderplaats**, Numerical Optimization Techniques for Engineering Design: With Applications, McGraw-Hill (1984).
- [11] **J.A. Vasconcelos**, Optimisation de Forme des Structures Electromagnétique, Thèse de Doctorat, L'Ecole Doctorale de Lyon des Sciences pour l'Ingénieur, 1994

-
- [12] **D. M. Ryan, P. E. Gill and W. Murray**, Penalty and Barrier Functions, Academic Press, (1974).
- [13] **B. Lassami**, Optimisation Non Différentiable pour la Prise en Compte de Cahier des Charges Générique en Automatique, Thèse de Doctorat, Université Paris Sud 11, 2008
- [14] **T. Diallo**, Etude et Illustration de Méthodes Itératives D'optimisation non Linéaire, Projet de master 2005-2006, Ecole Polytechnique Fédérale de Lausanne, Faculté Sciences de Base, Chaire de Recherche Opérationnelle Sud Est, Lausanne, avril 2006
- [15] **H. Khalil**, Développement des Techniques D'optimisation de Forme Pour la Conception de Composants Hyperfréquences, Université de Limoges Ecole Doctorale Sciences Technologie Santé Faculté des Sciences et Techniques, Thèse No 26-2009
- [16] **J.C. Gilbert, J. Nocedal**, Global Convergence Properties of Conjugate Gradient Method For Optimization, SIAM Journal on Optimization, Vol. 2, No. 1, pp. 21-24, (1992).
- [17] **S.A. Vandar, M.C. Pinar**, The Rate of Convergence of Conjugate Gradients, Numerische Mathematik., Vol. 48, No. 5, pp. 543-560, (1986).
- [18] **P. Segalat**, Méthodes de Points Intérieurs et de Quasi-Newton, Thèse de Doctorat, Université de Limoges, 20 décembre 2002.
- [19] **P. Y. Gupta**, A Modified Simplex Method for Solving 1-Norm Minimization Problem in Model Predictive Control, The Canadian Journal of Chemical Engineering, Vol. 85, pp. 199-207, (2008).
- [20] **M. Bergounioux**, Optimisation sans contraintes, Département de Mathématiques, Orléans, 2004
- [21] **F.X. Vialard**, Optimisation Numérique, 2012.
- [22] **IMSL**, Fortran Subroutines for Mathematical Applications, IMSL STAT/LIBRARY User's Manual, Version 2.0, IMSL, Houston, 1991.
- [23] **D.E. Goldberg**, Genetic Algorithms in Search, Optimization and Machine Learning, Ed. Addison Wesley, N.Y, 412 pp, (1989).

- [24] **S. Bouallagui**, Techniques D'optimisation Déterministe et Stochastique Pour la Résolution de Problèmes Difficiles en Cryptologie, L'institut National des Sciences Appliquées de Rouen, 2010.
- [25] **T. Chaari**, Un Algorithme Génétique Pour l'Ordonnancement Robuste : Application au Problème du Flow Shop Hybride, Valenciennes, 2010.
- [26] **T. Vallée, M. Yıldızoglu**, Présentation des Algorithmes Génétiques et de Leurs Applications en Economie, Université de Nantes, 2001.
- [27] **B. Sid**, Optimisation Topologique de Structures par les Algorithmes Génétique, Franche –Comité, 2006.
- [28] **R. Poli, J. Kennedy, T .Blackwell**, Particle Swarm Optimization. Swarm Intelligence, 1(1), 33–57, 2007.
- [29] **C .Russell. Eberhart, Y. Shi, J .Kennedy**. Swarm Intelligence. The Morgan Kaufmann Series in Artificial Intelligence. Morgan Kaufmann, San Francisco, CA, USA, 2001.
- [30] **X.-S. Yang, S. Deb**, Cuckoo search via Levy flights, in: Proc. Of World Congress on Nature & Biologically Inspired Computing (NaBIC 2009), December 2009, India. IEEE Publications, USA, pp. 210-214 (2009).
- [31] **S. Talukder**, Mathematical Modelling and Applications of Particle Swarm Optimization, School of Engineering at Blekinge Institute of Technology, 2011
- [32] **A. Chamekh**, Optimisation des Procédés de Mise en Forme par les Réseaux de Neurones Artificiels, Thèse de Doctorat, E.N.I. Monastir (2008).
- [33] **P. Alain**, Tournage, Technique de l'ingénieur, Traité Génie Mécanique, BM 7086, pp.3 22, 1997
- [34] **A. Toumine**, Usinage, INSA de Lyon, Groupe conception, productique, 2007
- [35] **J. S. Agapiou**, The Optimization of Machining Operations Based on a Combined Criterion, Part 2: Multi-pass Operations, Computers Ind. Trans, ASME 114, 508-513, 1992.
- [36] **M. Djennane**, Optimisation des Conditions de Coupe en Tournage, Mémoire de Magistère, Université de Batna, 2001.

- [37] **K. Iwata, et al**, A Probabilistic Approach to the Determination of the Optimum Cutting Conditions, *ASME Journal of Engineering for Industry* 94, 1099–1107, 1972.
- [38] **M.C. Chen, D.M. Tsai**, A Simulated Annealing Approach for Optimization of Multi-Pass Turning Operations, *International Journal of Production Research* 34 (10) (1996) 2803–2825
- [39] **R.H. Philipson, A. Ravindran**, Application of Mathematical Programming to Metal Cutting. *Mathematical Programming Study* 11, 116–134, 1979.
- [40] **R.V. Narang, G.W. Fischer**, Development of a Frame Work to Automate Process Planning Functions and to Determine Machining Parameters. *International Journal of Production Research* 31, 1921–194, 1993.
- [41] **K. Vijayakumar, G. Prabhakaran, P. Asokan , R. Saravanan**, Optimization of Multi-pass Turning Operations Using ant Colony System, *International Journal of Machine Tools and Manufacture* 43 (15) (2003).1633–1639
- [42] **A. Benyoucef**, Modélisation de l'Effort de Coupe dans le Cas du Fraisage Hémisphérique par une Approche Mécanistique, *Mémoire de Magistère, Université de Batna*, 2006
- [43] **I. Sonmez, A. Baykasoglu, T. Dereli, I. Huseyin**, Dynamic Optimization of Multi pass Milling Operations via Geometric Programming, *International Journal of Machine Tools & Manufacture* 39 (1999) 297–320.
- [44] **R. Venkata Rao, P.J. Pawar**, Parameter Optimization of a Multi-pass Milling Process Using Non-Traditional Optimization Algorithms, *Applied Soft Computing* 10 (2010) 445–456.
- [45] **S. XIE, Y. GUO**, Intelligent Selection of Machining Parameters in Multi-pass Turnings Using a GA-based Approach, *Journal of Computational Information Systems*, 7:5 (2011) 1714-1721.
- [46] **M. C. Chen, K. Y. Chen**, Optimization of Multi-pass Turning Operations with Genetic Algorithms: a Note, *International Journal of Production Research*, 41 (14):3385-3388, 2003.

- [47] **J. Srinivas, R. Giri, S.H. Yang**, Optimization of Multi-pass Turning Using Particle Swarm Intelligence, *International Journal of Advanced Manufacturing Technology*, 40 (1-2):56-66, 2009.
- [48] **R. S. Sankar, P. Asokan, R. Saravanan, S. Kumanan and G. Prabharan**, Selection of Machining Parameters for Constrained Machining Problem Using Evolutionary Computation, *International Journal of Advanced Manufacturing Technology*, 32 (9-10):892-901, 2007.
- [49] **M. C. Chen**, Optimizing Machining Economics Models of Turning Operations Using the Scatter Search Approach, *International Journal of Production Research*, 42 (13):2611-2625, 2004.

Résumé :

L'objectif de cette thèse est la mise au point d'une méthodologie de travail pour l'optimisation des conditions de coupe pour les procédés de tournage et de fraisage.

La première partie du travail est consacrée au choix de la méthode d'optimisation la plus fiable. Ce choix a été effectué selon une comparaison entre les méthodes d'optimisation déterministes (la méthode du gradient conjugué, GC et la programmation quadratique séquentielle, SQP), et les méthodes stochastiques (l'algorithme génétique, GA et la méthode d'essaim particulaire, PSO) ou bien, une méthode hybride (GA-SQP).

La deuxième partie du travail intéresse à l'élaboration d'une stratégie d'optimisation. Cette stratégie est adaptée aux caractéristiques des procédés d'usinage, c'est-à-dire à des systèmes non linéaires, multi-variables et soumis à des contraintes diverses. L'optimisation a été réalisée numériquement à l'aide de trois modèles.

Dans la troisième partie du travail, la méthodologie est appliquée à trois exemples d'application afin d'obtenir les conditions de coupe pertinentes optimisant les critères du temps et du coût de production.

Cette méthodologie peut être appliquée dans un grand nombre de procédés d'usinage. La performance des résultats dépend néanmoins de l'utilisation des modèles d'optimisation adéquats.

Abstract :

The objective of this thesis is the development of a working methodology for optimizing cutting conditions for the processes of turning and milling. . The first part of the work is devoted to the choice of the optimization method. This choice is made by means of a comparison between the deterministic optimization methods (the conjugate gradient method , and the GC sequential quadratic programming , SQP) and stochastic methods (genetic algorithm , GA and method of particle swarm PSO) or a hybrid method (GA- SQP).

The second part of the work focuses on the development of an optimization strategy. This strategy is adapted to the characteristics of the machining processes, that is to say non-linear, multi- variable and subject to the various system constraints. Optimization is performed digitally using three models.

In the third part of the work, the methodology is applied to three examples of application to obtain optimum cutting conditions optimization criteria of the time and cost of production.

This methodology can be applied in a large number of machining processes. However, the performance results depend on the use of appropriate optimization models.

ملخص

الهدف من هذه الرسالة هو تطوير منهجية العمل لتحسين شروط القطع لعمليات الخراطة و التفريز. الجزء الأول من هذا العمل مخصص لاختيار طريقة الأمثلة وذلك عن طريق المقارنة بين الطرق الجبرية (طريقة التدرج المترافق GC ، والبرمجة التربيعية المتعاقبة SQP) والطرق العشوائية (الخوارزمية الجينية GA ، وطريقة سرب الجسيمات PSO) أو الطريقة الهجينة (GA- SQP).

الجزء الثاني من العمل يركز على وضع إستراتيجية للأمثلة. هذه الإستراتيجية تم تكييفها لتلائم مع خصائص عمليات القطع، و نخص بالذكر المعادلات غير الخطية و ذات متغيرات عديدة و المميزة لنماذج القطع. تم تنفيذ الأمثلة رقمياً باستخدام ثلاثة نماذج.

في الجزء الثالث من العمل ، تم تطبيق إستراتيجية الأمثلة على ثلاثة أمثلة تطبيقية بغية إيجاد شروط القطع المثلى لحساب وقت و تكلفة الإنتاج.

هذه الإستراتيجية يمكن تطبيقها في عدد كبير من عمليات القطع. ومع ذلك فإن النتائج تعتمد على استخدام نماذج أمثلة مناسبة.