



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



## **Université Amar Telidji- Laghouat**

**FACULTÉ : DE GENIE CIVIL ET D'ARCHITECTURE**

**DÉPARTEMENT : DE GENIE CIVIL**

### **MÉMOIRE DE MASTER**

**Présenté par : OUARNOUGHI Fatima**

**DOMAINE : Sciences et Techniques**

**FILIERE : Hydraulique**

**OPTION : Ressources Hydrauliques**

### **Thème**

**Algorithmes des Buses de Harris et du Recuit Simulé pour  
l'optimisation des paramètres hydrodynamiques**

#### **Jury de soutenance :**

<b>Nom et Prénom</b>	<b>Grade</b>	<b>Qualité</b>
DJEHICHE AbdelKader	Pr.	Président
CHETTIH Mohamed	Pr.	Rapporteur
TADJ Walid	MCA	Examineur

Juin 2024

# Remerciement

*Louange à Dieu, qui supprime toute louange, car celui qui ne remercie pas les gens ne remercie pas Dieu.*

*Je voudrais adresser mes sincères remerciements et ma reconnaissance à mon encadreur, Monsieur Chettih Mohamed, pour ses conseils et sa direction, qu'il n'a jamais négligés.*

*Je voudrais également remercier les membres du comité de jugement qui m'ont fait l'honneur d'accepter de juger mon travail et tous les professeurs qui m'ont aidé et auprès desquels j'ai beaucoup appris.*

*Je remercie également tous ceux qui m'ont accompagné dans ce travail, de près ou de loin, ainsi que mes parents. Ils ont eu à cœur de me fournir toutes les conditions favorables à l'accomplissement de ce travail.*

OUARNOUGHI Fatima

# Dédicace

*Je remercie Dieu tout-puissant, qui m'a permis d'apprécier cette étape de ma carrière académique avec ce mémoire.*

*Je dédie mon humble travail à celle qui a été mon soutien et mon compagnon de route, à celle qui m'a apporté un amour et un soutien sans limites, ma mère bien-aimée, que Dieu prolonge sa vie.*

*À mon espoir dans cette vie, au propriétaire d'un cœur pur et tendre et d'une âme innocente, mon cher père, que Dieu le protège et prolonge sa vie.*

*À la prunelle de mes yeux, ma sœur Oum El-Kheir et mon frère Allal.*

*À mes amies qui ne sont pas nées de ma mère, compagnes du chemin de la vie et qui font partie de moi, chacune en son nom propre, à tous ceux qui ont contribué à me soutenir et à me motiver pour atteindre cet accomplissement et qui ont prié pour moi avec succès et réussite.*

*Fatima*

# Résumé

Le développement scientifique de nos connaissances sur le fonctionnement des systèmes aquifères a conduit à la création et à l'utilisation de modèles mathématiques complexes conçus pour décrire les données de terrain et les données expérimentales. Il est essentiel d'évaluer la capacité du modèle à représenter fidèlement les données afin d'obtenir des estimations réalistes des paramètres du modèle. Les techniques d'optimisation jouent un rôle important dans l'identification des variables qui ont un impact significatif sur les résultats du modèle et dans l'évaluation de sa qualité.

Dans cette thèse, nous présentons deux modèles d'optimisation, l'optimisation de Harris Hawks (HHO) et l'Optimisation du recuit simulé (RS). Le modèle HHO s'inspire du comportement coopératif et de la technique de poursuite des buses de Harris dans la nature, connue sous le nom de «bond-surprise». Quant à RS, il s'inspire du processus de recuit en métallurgie, où un matériau est chauffé puis lentement refroidi pour éliminer les défauts et réduire son état énergétique. Ces deux techniques ont été modélisées mathématiquement pour créer les algorithmes d'optimisation. L'efficacité des optimiseurs HHO et RS a été validée par des expériences qui montrent que les deux algorithmes donnent des résultats prometteurs et compétitifs, chaque technique ayant ses propres caractéristiques.

Enfin, nous traiterons le cas d'une nappe captive, le cas d'une nappe libre et le cas d'une nappe semi-captive. Nous testerons et vérifierons les codes mathématiques en mettant en œuvre des exemples théoriques de pompage et en interprétant certains essais de pompage à l'aide des deux algorithmes.

**Mots clés :** optimisation-HHO-RS-techniques-Algorithmes

# Abstract

The scientific development of our knowledge of how aquifer systems work has led to the creation and use of complex mathematical models designed to describe field and experimental data. It is essential to assess the model's ability to accurately represent the data in order to obtain realistic estimates of the model's parameters. Optimization techniques play an important role in identifying variables that have a significant impact on model results, and in assessing model quality.

In this thesis, we present two optimization models, Harris Hawks Optimization (HHO) and Simulated Annealing Optimization (SAO). The HHO model is inspired by the cooperative 2modelled2 and pursuit technique of Harris nozzles in nature, known as "leap-surprise". RS is inspired by the annealing process in metallurgy, where a material is heated and then slowly cooled to remove defects and reduce its energy state. Both techniques were mathematically 2modelled to create the optimization algorithms. The effectiveness of the HHO and RS optimizers has been validated by experiments which show that both algorithms deliver promising and competitive results, with each technique having its own unique characteristics.

Finally, we will deal with the confined water table case, the unconfined water table case, and the semi-confined water table case. We will test and verify the mathematical codes by implementing theoretical pumping examples and interpreting some pumping tests using the two algorithms.

**Keywords:** Optimization-SAO-HHO- techniques- Algorithms

## ملخص

أدى التطور العلمي لمعرفتنا بكيفية عمل أنظمة طبقات المياه الجوفية إلى إنشاء واستخدام نماذج رياضية معقدة مصممة لوصف البيانات الميدانية والتجريبية. ومن الضروري تقييم قدرة النموذج على تمثيل البيانات بدقة من أجل الحصول على تقديرات واقعية لبارامترات النموذج. تلعب تقنيات التحسين دوراً مهماً في تحديد المتغيرات التي لها تأثير كبير على نتائج النموذج وفي تقييم جودته.

في هذه الأطروحة، نقدم نموذجين للتحسين هما نموذج تحسين صقور هاريس ونموذج محاكاة التلدين الأمثل، نموذج هاريس هاوكس مستوحى من السلوك التعاوني وتقنية المطاردة لصقور هاريس في الطبيعة، والمعروفة باسم "القفزة المفاجئة" أما نموذج محاكاة التلدين فهو مستوحى من عملية التلدين في علم المعادن، حيث يتم تسخين المادة ثم تبريدها ببطء لإزالة العيوب وتقليل حالة الطاقة فيها. تمت نمذجة هاتين التقنيتين رياضياً لإنشاء خوارزميات التحسين. وقد تم التحقق من فعالية مُحسِنَات النموذجين من خلال التجارب التي أظهرت أن كلا الخوارزميتين تعطي نتائج واعدة وتنافسية.

وأخيراً، سنتعامل مع حالة منسوب المياه الجوفية المحصورة، وحالة منسوب المياه الجوفية غير المحصورة وحالة منسوب المياه الجوفية شبه المحصورة. سنقوم باختبار الرموز الرياضية والتحقق منها من خلال تنفيذ أمثلة نظرية للضحخ وتفسير بعض اختبارات الضحخ باستخدام الخوارزميتين.

الكلمات المفتاحية: التحسين-محاكاة التلدين-هاريس هاوكس- خوارزميات

# Sommaire

<b>Introduction Générale</b>	<b>1</b>
<b>I. Optimisation du Recuit Simulé</b>	<b>3</b>
I.1. Introduction	3
I.2. Historique et développement:	3
I.3. Fondements théoriques	4
I.3.1. Définition des termes	5
I.3.2. Représentation algorithmique du recuit simulé	6
I.3.3. Pseudocode	7
I.3.4. Fonction objectif	8
I.4. Stratégies d'optimisation	8
I.5. Techniques de perturbation	9
I.6. Améliorations ET variantes	10
I.6.1. Recuit simulé parallèle et distribué	10
I.6.2. Algorithme génétique	11
I.7. Techniques adaptatives	11
I.8. Evaluation des performances	12
I.8.1. Performance asymptotique	12
I.8.2. Performance en temps fini	20
I.9. Comparaison avec d'autres techniques d'optimisation	21
I.10. Orientations futures	22
I.11. Conclusion	23
<b>II. Optimisation de Harris Hawks</b>	<b>24</b>
II.1. Introduction	24
II.2. Contexte et inspiration	24
II.2.1. Introduction aux buses de Harris et à leur comportement de chasse	24
II.2.2. Vue d'ensemble de l'inspiration derrière HHO	25
II.3. Concepts fondamentaux de l'optimisation de Harris Hawks	26
II.3.1. Modèles mathématiques de HHO	26
II.3.1.1. Phase d'exploration	26
II.3.1.2. Transition de l'exploration à l'exploitation	26
II.3.1.3. Phase d'exploitation	27
II.4. Mécanique de l'optimisation de Harris Hawks	29
II.4.1. Discussion sur le rôle des différents paramètres et leur impact sur les performances de l'algorithme	29
II.4.2. Représentation algorithmique de HHO	30
II.4.3. L'organigramme de HHO	31
II.5. Variantes et extension de l'optimisation de Harris Hawks	32
II.5.1. Vue d'ensemble des variantes ou modifications de HHO	32
II.5.2. Applications et variantes du Harris Hawks	32
II.5.2.1. Optimisation de l'ingénierie	32
II.5.2.2. L'informatique	34

II.5.2.3.Médecine et santé publique	35
II.5.3.Comparaison avec d'autres algorithmes d'optimisation	35
II.6.Conclusion	36
<b>III.Applications des métaheuristiques à quelques problèmes</b>	<b>37</b>
<b>d'Hydraulique souterraine</b>	
III.1. Introduction	37
III.2. Essais de pompage	37
III.2.1 Nappe captive	37
III.2.2.Nappe libre	39
III.2.3. Nappe semi-captive	42
III.2.4. Essais de Pompage en régime transitoire	44
III.3. Tests et validations des codes	46
III.3.1. Problème 1	46
III.3.2 Problème 2	55
III.3.3 Problème 3	58
III.4. Interprétation des essais de pompage	61
III.4.1. Premier cas	61
III.4.2. Deuxième cas	63
III.5. Conclusion	65
<b>Conclusion Générale</b>	<b>67</b>
<b>Bibliographie</b>	

# Liste des figures

Fig.I.1: Algorithme simplifié du recuit simulé	6
Fig.I.2: Organigramme de l'algorithme du recuit simulé	7
Fig.II.1: Faucon (Buse) d'Harris ( <a href="https://www.gettyimages.fr/photos/buse-de-harris">https://www.gettyimages.fr/photos/buse-de-harris</a> )	25
Fig.II.2: Vecteurs lors d'un siège doux avec des plongées rapides progressives en 2D (Tripathy, B. K., et al (2022))	29
Fig.II.3: Vecteurs lors d'un siège dur avec des plongées rapides progressives en 3D (Tripathy, B. K., et al (2022))	29
Fig.II.4: Algorithme de Harris Hawks	30
Fig.II.5: Organigramme de L'algorithme de Harris Hawks	31
Fig.III.1: Coupe d'un puits pompant dans un aquifère captif (Holzbecher, E. (2009))	38
Fig.III.2: Abaissement de la hauteur piézométrique de la nappe d'eau dans un aquifère captif (confiné) dû au pompage (formule de Thiem)	39
Fig.III.3: Coupe d'un puits pompant dans un aquifère libre (Holzbecher, E. (2009))	40
Fig.III.4: Abaissement de la hauteur piézométrique de la nappe dans un aquifère libre	42
Fig.III.5: Vue schématique en coupe d'un puits pompant dans un aquifère semi-confiné aquifère (Holzbecher, E. (2009))	42
Fig.III.6: Abaissement de la hauteur piézométrique de l'eau souterraine dans un aquifère semi- captif dû au pompage, d'après de Glee (1930)	43
Fig.III.7: Abaissement régulier de la hauteur piézométrique des eaux souterraines dans un aquifère confiné, un aquifère semi- captif et un aquifère non confiné (Holzbecher, E. (2009))	43
Fig.III.8: Abaissement instable de la hauteur piézométrique de la nappe dans un aquifère captif dû au pompage selon Theis ; paramètres donnés dans le texte, pour les temps $t = 10^3, 10^4, 10^5, 10^6$ (Holzbecher, E. (2009))	44
Fig.III.9: Rabattement du niveau piézométrique dans un aquifère captif dû au pompage, d'après la formule de Theis	45
Fig.III.10: (a) - Comparaison des courbes de rabattements en fonction du temps entre les valeurs observées et calculées après optimisation l'aide du Recuit Simulé	47
(b) – Erreur relative entre les valeurs observées et calculées des rabattements	
Fig.III.11: Droite de corrélation entre les valeurs observées et calculées des rabattements	47
Fig.III.12: Valeurs des transmissivités obtenues pour 100 essais	48
Fig.III.13: Valeurs des coefficients d'emmagasinement obtenues pour 100 essais	48
Fig. III.14: Moyenne Mobile de la valeur de la transmissivité après 100 essais	48
Fig.III.15: Moyenne Mobile de la valeur du coefficient d'emmagasinement après 100 essais	49
Fig.III.16: Visualisation des paramètres de performance au moment de l'exécution du code (a-Convergence, b-Températures des variables, c-Fonction d'évaluation, d-Critère d'arrêt	50
Fig.III.17: Courbe de convergence de la fonction Objectif en fonction du nombre d'itérations	53
Fig.III.18: Comparaison des courbes de rabattements en fonction du temps entre les	54

valeurs observées et calculées après optimisation l'aide de Harris Hawks	
Fig.III.19: Erreur relative entre les valeurs observées et calculées des rabattements	54
Fig.III.20: Droite de corrélation entre les valeurs observées et calculées des rabattements	55
Fig.III.21: (a) -Comparaison des courbes de rabattements en fonction du temps entre les valeurs observées et calculées après optimisation à l'aide du Recuit Simulé. (b) – Erreur relative entre les valeurs observées et calculées des rabattements et (c) – Droite de corrélation entre les valeurs observées et calculées des rabattements	56
Fig.III.22: (a) – Courbe de convergence, (b) -Comparaison des courbes de rabattements en fonction du temps entre les valeurs observées et calculées après optimisation à l'aide du Recuit Simulé (c) – Erreur relative entre les valeurs observées et calculées des rabattements et (d) – Droite de corrélation entre les valeurs observées et calculées des rabattements	57
Fig.III.23: Courbes de convergence de la fonction Objectif pour les itérations 50, 100, 150 et 200	58
Fig.III.24: (a) - Comparaison des courbes de rabattements en fonction de la distance entre les valeurs observées et calculées après optimisation à l'aide du Recuit Simulé. (b) – Erreur relative entre les valeurs observées et calculées des rabattements et (c) – Droite de corrélation entre les valeurs observées et calculées des rabattements	59
Fig.III.25: (a) – Courbe de convergence, (b) -Comparaison des courbes de rabattements en fonction de la distance entre les valeurs observées et calculées après optimisation de Harris Hawks (c) – Erreur relative entre les valeurs observées et calculées des rabattements et (d) – Droite de corrélation entre les valeurs observées et calculées des rabattements	59
Fig.III.26: Courbes de convergence de la fonction Objectif pour les itérations 10, 20, 30, 40, 100 et 200	60
Fig.III.27: - (a) - Comparaison des courbes de rabattements en fonction du temps entre les valeurs observées et calculées après optimisation à l'aide du Recuit Simulé. (b) – Erreur relative entre les valeurs observées et calculées des rabattements et (c) – Droite de corrélation entre les valeurs observées et calculées des rabattements	61
Fig.III.28: Résultats obtenus par l'optimisation de Harris Hawks pour l'essai n° 4 correspondants à la courbe de convergence, aux courbes des rabattements observés et calculés, à l'erreur relative et à la corrélation entre les rabattements observés et calculés	63
Fig.III.29: Coupe géologique de la région de l'essai de pompage d'Oude Korendijk <b>(Kruseman, G.P et al ,1974)</b>	63
Fig.III.30: - (a) - Comparaison des courbes de rabattements en fonction du temps entre les valeurs observées et calculées après optimisation à l'aide du Recuit Simulé. (b) – Erreur relative entre les valeurs observées et calculées des rabattements et (c) – Droite de corrélation entre les valeurs observées et calculées des rabattements	64
Fig.III.31: - Résultats obtenus par l'optimisation de Harris Hawks pour l'essai n° 2 correspondants à la courbe de convergence, aux courbes des rabattements observés et calculés, à l'erreur relative et à la corrélation entre les rabattements observés et calculés	65

## Liste des Tableaux

Tab.III.1: Résultats obtenus pour différents essais correspondants aux Transmissivités T, aux coefficients d'emmagasinement S, à la racine carrée de de l'erreur quadratique moyenne (RMSE) et au temps de calcul en seconde	47
Tab.III.2: Valeurs des Moyennes Mobiles de la Transmissivité et du Coefficient d'emmagasinement pour différents essais	49
Tab.III.3: Estimation de la transmissivité et du coefficient d'emmagasinement par l'optimisation Harris Hawks pour trois essais différents	55
Tab.III.4: Résultats obtenus à l'aide du Recuit Simulé pour différents essais correspondants aux Transmissivités T, aux Résistance de la couche semi-perméable C, à la racine carrée de de l'erreur quadratique moyenne (RMSE) et au temps de calcul en seconde	56
Tab.III.5: Résultats obtenus par l'optimisation de Harris Hawks pour différents essais correspondants aux Transmissivités T, aux Résistance de la couche semi-perméable C, à la racine carrée de de l'erreur quadratique moyenne (RMSE) et au temps de calcul en seconde	57
Tableau III.6: Résultats obtenus à l'aide du Recuit Simulé pour différents essais correspondants aux Perméabilités K en fonction de la distance, à la racine carrée de de l'erreur quadratique moyenne (RMSE) et au temps de calcul en seconde	58
Tableau III.7: Résultats obtenus par l'optimisation de Harris Hawks pour différentes itérations essais correspondants aux Perméabilités K, à la racine carrée de de l'erreur quadratique moyenne (RMSE) et au temps de calcul en seconde	60
Tableau III.8: Résultats obtenus par l'optimisation de Harris Hawks pour quatre essais différents pour estimer la transmissivité et le coefficient d'emmagasinement de l'essai de pompage de Aïn Madhi	62
Tableau III.9: Rabattements dans les piézomètres entre 20 et 24 m de profondeur après 830 minutes de pompage (Essai d'Oude Korendjik) <b>(Kruseman, G.P, 1974)</b>	64
Tableau III.10: Résultats obtenus par l'optimisation de Harris Hawks pour les deux essais différents pour estimer la Perméabilité de l'essai de pompage d'Oude Korendjik	64

# Introduction Générale

---

L'interprétation des essais de pompage représente la meilleure solution de reconnaissance hydrogéologique pour estimer les paramètres hydrodynamiques. Classiquement l'interprétation est basée sur une interpolation manuelle utilisant des abaques et des tracés de courbe de rabattements qui laisse une part de subjectivité à l'opérateur ou utilisant parfois des méthodes d'approximation peu précises (Forkasiewicz, 1970 ; Kruseman et al., 1974). D'autres techniques numériques sont également utilisées basées sur des lois non linéaires utilisant le processus itératif par degrés successifs de Gauss-Newton sont assez répondues pour l'interprétation des essais de pompage (Mania et Sucche, 1978).

Notre problématique cherche donc à minimiser les écarts entre les rabattements observés et les rabattements calculés par un modèle mathématique d'essai de pompage à l'aide d'une méthode d'optimisation constituant ainsi, notre fonction **Objectif**.

Les algorithmes d'optimisation sont des outils essentiels dans divers domaines, visant à trouver la meilleure solution parmi un ensemble d'options possibles. Ces algorithmes se présentent sous diverses formes, chacune adaptée à des types spécifiques de problèmes d'optimisation.

Les algorithmes basés sur le gradient, tels que la descente de gradient et ses variantes, constituent une classe prédominante de méthodes d'optimisation. Ces algorithmes mettent à jour de manière itérative les paramètres d'un modèle dans le sens de la descente la plus raide de la fonction **Objectif** afin de la minimiser.

Une autre catégorie comprend les algorithmes métaheuristiques tels que le recuit simulé, l'optimisation par essais de particules, colonies de Fourmies, l'algorithme de recherche du corbeau, Loup gris, ainsi que de nombreuses autres méthodes qui explorent l'espace des solutions de manière plus approfondie, souvent en tirant parti du caractère aléatoire ou de l'intelligence en essaim. Ces algorithmes sont particulièrement utiles pour résoudre des problèmes d'optimisation complexes et multidimensionnels avec des fonctions objectives non linéaires et non convexes.

A ce titre, avec l'apport de l'intelligence artificielle, les méthodes d'optimisation métaheuristiques constituent actuellement une nouvelle voie prometteuse dans de nombreux domaines scientifiques et techniques et en particulier pour l'interprétation automatique des essais de pompage (Tadj et al., 2018).

L'objectif de ce travail est d'appliquer deux techniques métaheuristiques très différentes pour minimiser la fonction Objectif. La première méthode est celle du Recuit Simulé, qui est

une méthode d'optimisation inspirée du processus de recuit en métallurgie, où un matériau est chauffé puis refroidi lentement pour atteindre une structure minimale d'énergie.

La deuxième méthode est celle de l'optimisation des Harris Hawks, inspirée du comportement de chasse des faucons de Harris, c'est une méthode relativement nouvelle dans le domaine de l'optimisation métaheuristique.

La méthodologie adoptée dans cette étude est organisée en trois chapitres. Le premier chapitre est dédié à l'optimisation du Recuit Simulé en décrivant les fondements théoriques de la méthode, son algorithme et son organigramme. Nous avons également abordé les améliorations et les variantes recommandées de la méthode ainsi que l'évaluation de ses performances.

Le deuxième chapitre est consacré à l'optimisation de Harris Hawks, une méthode relativement nouvelle dans le domaine de l'optimisation où ces oiseaux de proie utilisent des stratégies de collaboration pour chasser d'une façon optimale. Dans ce chapitre, nous avons décrit le contexte et l'inspiration de la méthode et présenté les concepts fondamentaux de l'optimisation de Harris Hawks. Nous avons également présenté son algorithme et son organigramme. A la fin de ce deuxième chapitre, nous avons abordé les variantes et les extensions de l'optimisation de Harris Hawks.

Le troisième chapitre a été scindé en 3 parties, une première partie attribuée à la présentation de quelques modèles mathématiques permettant l'interprétation des essais de pompage. La deuxième partie destinée à tester les codes de calcul et à vérifier et valider les codes en effectuant des exemples théoriques de pompage. La troisième partie traite deux cas pratiques d'essais de pompage et leur l'interprétation à l'aide des métaheuristicques.

# Chapitre I

## Optimisation du Recuit Simulé

---

### I.1 Introduction :

Le recuit simulé est une technique probabiliste utilisée pour trouver une solution approximative à un problème d'optimisation. Cette technique s'inspire du processus de recuit en métallurgie, où un matériau est chauffé puis lentement refroidi pour éliminer les défauts et réduire son état énergétique. Le recuit simulé est un algorithme de recherche locale (*Métaheuristique*). Sa facilité de mise en œuvre, ses propriétés de convergence et son utilisation de mouvements d'escalade pour échapper à l'optimum local ont fait de cet algorithme une technique populaire au cours des deux dernières décennies. Elle est généralement utilisée pour résoudre des problèmes d'optimisation discrète et, dans une moindre mesure, continus. Sa capacité à équilibrer l'exploration et l'exploitation la rend particulièrement efficace pour les problèmes d'optimisation complexes avec des espaces de recherche non convexes. Lorsque le coefficient de température tombe à zéro, les mouvements de montée sont moins fréquents et la fonction objective est plus efficace.

L'algorithme est converti en une forme dans laquelle toutes les probabilités sont centrées sur l'ensemble des solutions globalement optimales, à condition que l'algorithme soit convergent ; dans le cas contraire, l'algorithme convergera vers un optimum local, qui peut ou non être globalement optimal. L'objectif de cette méthode est de générer un ensemble de solutions qui est une bonne approximation de l'ensemble des solutions efficaces. Dans ce chapitre, on s'attachera à établir une description mathématique de l'algorithme du recuit simulé et on décrira son comportement d'un point de vue théorique et pratique.

### I.2 Historique et développement :

Le recuit simulé est une méthode d'optimisation inspirée du processus de recuit en métallurgie, où un matériau est chauffé puis refroidi lentement pour atteindre une structure minimale d'énergie. Le terme "**recuit**" provient du processus métallurgique où un matériau est chauffé jusqu'à une température élevée pour ensuite être refroidi lentement. Cela permet aux atomes de se réorganiser et de minimiser l'énergie du système, atteignant ainsi un état plus stable.

La méthode s'inspire de ces principes physiques, où un "état de haute énergie" peut explorer différentes configurations et, en refroidissant lentement, le système se stabilise dans un état d'énergie minimale.

L'algorithme de base pour simuler le comportement des atomes lors du recuit a été décrit par N. Metropolis et ses collègues en 1953. Cet algorithme, connu sous le nom de l'algorithme de Metropolis, est utilisé pour simuler l'état d'équilibre des systèmes physiques à température constante. Il a fallu près de trente ans pour que Kirkpatrick et al. (1983) et indépendamment Cerny V. (1985) réalisent qu'il existe une profonde analogie entre la minimisation de la fonction de coût d'un problème d'optimisation combinatoire et le lent refroidissement d'un solide jusqu'à ce qu'il atteigne son état fondamental de faible énergie, et que le processus d'optimisation peut être réalisé en appliquant le critère de Metropolis. En remplaçant le coût par l'énergie et en exécutant l'algorithme de Metropolis à une séquence de valeurs de températures lentement décroissantes, Kirkpatrick et ses collègues ont obtenu un algorithme d'optimisation combinatoire qu'ils ont appelé "**Recuit Simulé**".

Depuis les travaux de Kirkpatrick et al. (1983), la méthode a été affinée et améliorée de diverses manières, notamment en ajustant les schémas de refroidissement (température), les critères d'acceptation des solutions, et en combinant le recuit simulé avec d'autres techniques d'optimisation. Le Recuit Simulé est considéré maintenant comme un algorithme stochastique impliquant une convergence asymptotique et permettant des mouvements aléatoires dans le voisinage recherché afin d'échapper aux minima locaux.

Le recuit simulé a été appliqué avec succès à de nombreux problèmes d'optimisation difficiles tels que le problème du voyageur de commerce, la conception de circuits intégrés, la planification et l'ordonnancement, et bien d'autres. Bien que proposé il y a plus de 40 ans, il attire toujours une certaine attention et est largement utilisé dans de nombreuses applications. En somme, le recuit simulé est une méthode polyvalente qui peut être appliquée à de nombreux domaines pour trouver des solutions proches de l'optimum global.

### **I.3 Fondements théoriques :**

La méthodologie du recuit simulé remonte à la thermodynamique et aux processus cinétiques de refroidissement, de recuit et de cristallisation de matériaux tels que les métaux et certains liquides. Lorsque la température initiale d'un matériau est élevée, les atomes qu'il contient ont une grande diffusivité, de sorte qu'ils peuvent sauter à différentes positions du réseau avec différentes énergies. Lorsque la température baisse lentement ou dans des conditions de refroidissement lent ou de recuit progressif, les atomes ont suffisamment de temps pour diffuser et trouver plus tard un point du réseau avec des énergies plus faibles. Même si un atome se stabilise à un équilibre stable avec une énergie locale moyenne, le refroidissement lent et la température initiale élevée du système physique garantissent que l'atome surmonte la barrière d'activation cinétique pour sortir du puits d'équilibre énergétique local et entrer dans le puits qui peut éventuellement contenir le minimum énergétique global.

Notez que tout matériel voudra toujours minimiser globalement son énergie libre de Gibbs et atteindre l'état d'équilibre le plus stable si un délai suffisant est prévu pour qu'un tel phénomène se produise.

### 1.3.1 Définition des termes :

Pour décrire les propriétés spécifiques de l'algorithme de recuit simulé pour les problèmes d'optimisation discrets, plusieurs définitions sont nécessaires. Soit  $\Omega$  l'espace des solutions (c'est-à-dire l'ensemble de toutes les solutions possibles). Soit  $f: \Omega \rightarrow \mathfrak{R}$  une fonction objective définie sur l'espace des solutions. Le but est de trouver un minimum global  $\omega^*$  (i.e.,  $\omega^* \in \Omega$  tel que  $f(\omega^*) \leq f(\omega)$  pour tout  $\omega \in \Omega$ ). La fonction objectif doit être bornée pour assurer que  $\omega^*$  existe. Définissons  $N(\omega)$  comme étant la fonction de voisinage pour  $\omega \in \Omega$ . Ainsi, associées à chaque solution,  $\omega \in \Omega$ , se trouvent des solutions voisines,  $N(\omega)$ , qui peuvent être atteintes en une seule itération d'un algorithme de recherche locale. Le recuit simulé commence par une solution initiale  $\omega \in \Omega$ . Une solution voisine  $\omega' \in N(\omega)$  est ensuite générée (soit de manière aléatoire, soit en utilisant une règle préétablie). Le recuit simulé est basé sur le critère d'acceptation de Metropolis (Metropolis, N et al, (1953)), qui modélise la façon dont un système thermodynamique passe de la solution (état) actuelle  $\omega \in \Omega$  à une solution candidate  $\omega' \in N(\omega)$ , dans laquelle le contenu énergétique est minimisé. La solution candidate,  $\omega'$ , est acceptée comme solution courante sur la base de la probabilité d'acceptation.

$$p\{\text{accept } \omega' \text{ entant que prochaine solution}\} = \begin{cases} \exp[-(f(\omega') - f(\omega))/t_k] & \text{if } f(\omega') - f(\omega) > 0 \\ 1 & \text{if } f(\omega') - f(\omega) \leq 0 \end{cases} \quad (1.1)$$

Définir  $t_k$  comme le paramètre de température à l'itération  $k$  (boucle extérieur), tel que

$$t_k > 0 \text{ pour tous } k \text{ et } \lim_{k \rightarrow \infty} t_k = 0. \quad (1.2)$$

Cette probabilité d'acceptation est l'élément de base du mécanisme de recherche dans le recuit simulé. Si la température est réduite suffisamment lentement, le système peut atteindre un état d'équilibre à chaque itération  $k$ . atteindre un équilibre (steady state) à chaque itération  $k$ . Soit  $f(\omega)$  et  $f(\omega')$  les énergies (valeurs de la fonction objectif) associées aux solutions  $\omega \in \Omega$  et  $\omega' \in N(\omega)$ , respectivement. Cet équilibre suit la distribution de Boltzmann, qui peut être distribuée comme la probabilité que le système soit dans l'état  $\omega \in \Omega$  avec l'énergie  $f(\omega)$  à la température  $T$  telle que

$$P\{\text{le système est dans l'état } \omega \text{ à la température } T\} = \frac{\exp(-f(\omega)/t_k)}{\sum_{\omega' \in \Omega} \exp(-f(\omega')/t_k)}. \quad (1.3)$$

Si la probabilité de générer une solution candidate  $\omega$  à partir des voisins de la solution  $\omega \in \Omega$  est  $g_k(\omega, \omega')$ , où

$$\sum_{\omega' \in N(\omega)} g_k(\omega, \omega') = 1, \text{ pour tous } \omega \in \Omega, k=1,2,\dots, \quad (1.4)$$

On peut alors définir une matrice stochastique carrée non négative  $p_k$  avec des probabilités de transition

$$p_{k(\omega, \omega')} = \begin{cases} g_k(\omega, \omega') \exp(-\Delta_{\omega, \omega'} / t_k) & \omega' \in N(\omega), \omega' \neq \omega \\ 0 & \omega' \notin N(\omega), \omega' \neq \omega \\ 1 - \sum_{\omega'' \in N(\omega), \omega'' \neq \omega} p_k(\omega, \omega'') & \omega' = \omega \end{cases} \quad (1.5)$$

Pour toutes les solutions  $\omega \in \Omega$  et toutes les itérations  $k=1,2, \dots$  et avec  $\Delta_{\omega,\omega'} \equiv f(\omega') - f(\omega)$ . Ces probabilités de transition définissent une séquence de solutions générées par la chaîne de Markov inhomogène (Romeo, F, et al, (1991)).

### 1.3.2. Représentation algorithmique du recuit simulé :

```

- Choisir une solution initiale  $\omega \in \Omega$ 
- Sélectionner le compteur de changement de température  $k=0$ 
- Sélectionner un programme de refroidissement de la température,  $t_k$ 
- Sélectionner une température initiale  $T = t_0 \geq 0$ 
- Sélectionner un programme de répétition,  $M_k$  qui définit le nombre d'itérations exécutées à chaque température,  $t_k$ 
- Répéter
    ▪ Régler le compteur de répétition  $m=0$ 
    ▪ Répétition
    ▪ Générer une solution  $\omega' \in N(\omega)$ 
    ▪ Calculer  $\Delta_{\omega,\omega'} = f(\omega') - f(\omega)$ 
      Si  $\Delta_{\omega,\omega'} \leq 0$ , alors  $\omega \leftarrow \omega'$ 
      Si  $\Delta_{\omega,\omega'} > 0$ , alors  $\omega \leftarrow \omega'$  avec probabilité  $\exp(-\Delta_{\omega,\omega'} / t_k)$ 
     $m \leftarrow m+1$ 
    Jusqu'à ce que  $m = M_k$ 
   $k \leftarrow k+1$ 
Jusqu'à ce que le critère d'arrêt soit rempli.
  
```

Fig.I.1 : Algorithme simplifié du recuit simulé.

## I.3.3. Pseudocode:

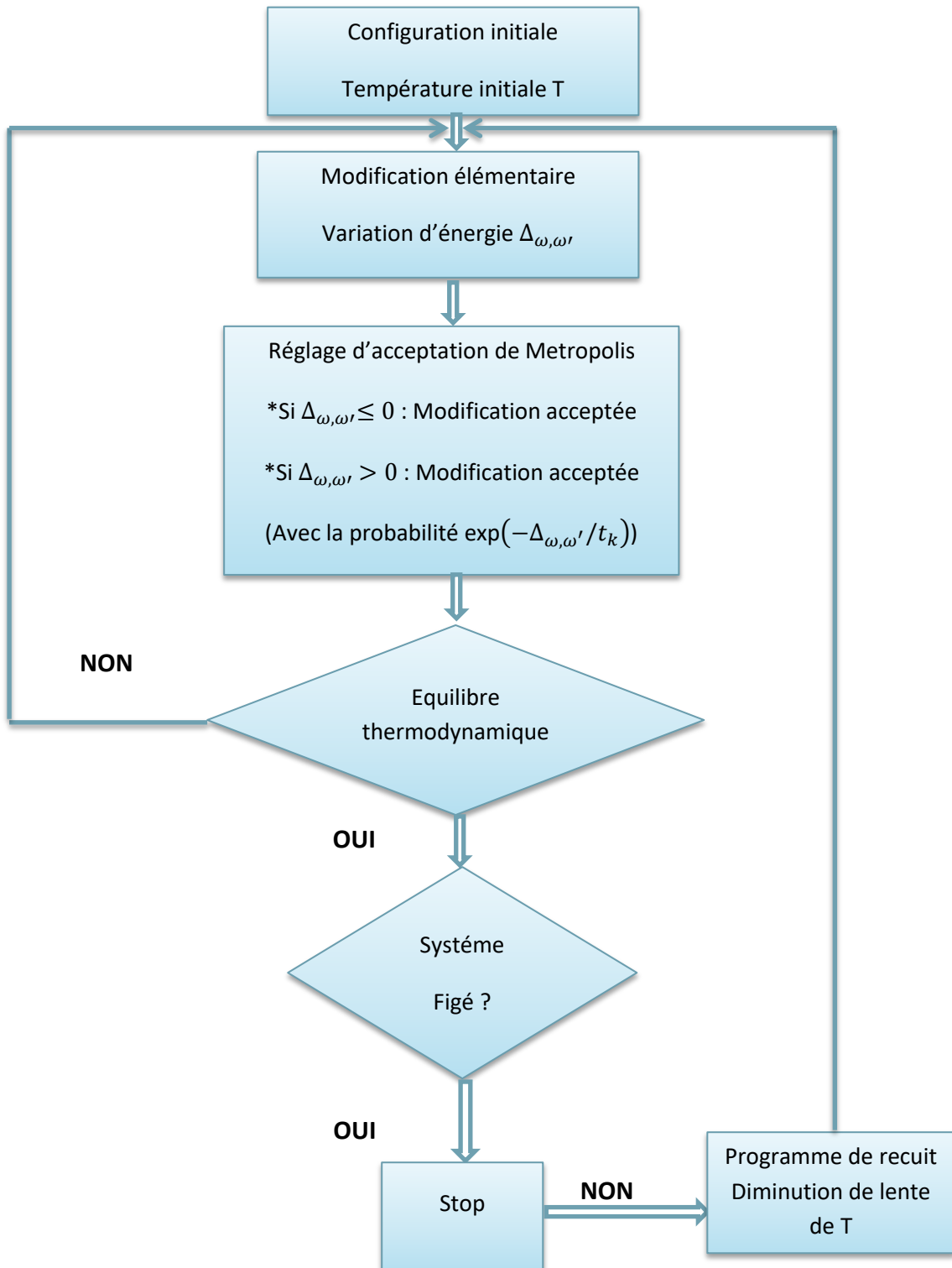


Fig. I.2 : Organigramme de l'algorithme du recuit simulé.

### I.3.4. Fonction objectif :

La fonction objectif est la fonction que vous souhaitez optimiser. Les algorithmes de la boîte à outils d'optimisation globale tentent de trouver le minimum de la fonction objectif. Écrivez la fonction objectif sous forme de fichier ou de fonction anonyme et transmettez-la au solveur sous forme de poignée de fonction. (Tan, C. M., (2008)).

- **Programme de la température :**

La température est un paramètre du recuit simulé qui affecte deux aspects de l'algorithme (Cohn, H, et al, 1999) :

- la distance entre un point d'essai et le point actuel ;
- la probabilité d'accepter un point d'essai ayant une valeur de fonction objective plus élevée.

La température peut être un vecteur avec des valeurs différentes pour chaque composante du point actuel. Généralement, la température initiale est un scalaire. Elle est le paramètre de recuit, qui est fonction du nombre d'itérations. Plus la température diminue lentement, plus les chances de trouver une solution optimale sont élevées.

- **Programmes de refroidissement :**

Les programmes de refroidissement (ou schémas de refroidissement) dans le recuit simulé sont des composantes cruciales qui déterminent comment la température diminue au fil des itérations. Ces programmes influencent l'efficacité de l'algorithme pour échapper aux optima locaux et atteindre un optimum global. Voici quelques-uns des schémas de refroidissement les plus courants utilisés en recuit simulé :

- **Refroidissement adaptatif :**

Ici, la manière de réduire la température est ajustée dynamiquement en fonction de l'état de la recherche. Par exemple, la température peut être maintenue ou réduite moins rapidement si l'on observe que l'algorithme est proche d'un optimum local.

- **Refroidissement géométrique :**

Ce schéma est une variante de l'exponentiel où la température est réduite en multipliant par une constante, mais la constante peut être ajustée de manière plus complexe en fonction des résultats obtenus à chaque étape. Par exemple, le facteur de réduction peut dépendre du nombre de changements acceptés ou rejetés lors de la dernière période de température stable, offrant ainsi une adaptabilité à la dynamique du processus d'optimisation.

- **Refroidissement non-monotone :**

Dans ce schéma innovant, la température peut temporairement augmenter pour permettre de sortir d'un optimum local profond. Ce genre de "réchauffement" contrôlé peut revitaliser la recherche en permettant l'exploration de nouvelles régions de l'espace de solutions. La gestion de ces augmentations de température nécessite une planification soignée pour éviter de compromettre la convergence globale de l'algorithme.

### I.4. Stratégies d'optimisation :

Le recuit simulé est un algorithme d'optimisation probabiliste inspiré du processus de recuit en métallurgie. Il est souvent utilisé pour trouver le minimum global d'une fonction complexe et multidimensionnelle (Kirkpatrick, S, et al, (1983)). Voici quelques stratégies pour améliorer son efficacité:

- **Solution initiale** : La qualité de la solution initiale peut avoir un impact significatif sur les performances de l'algorithme. L'utilisation d'une bonne solution initiale permet d'éviter les minimas locaux.
- **Tables de température constante** : Si la séquence de recuit simulé est exécutée à température constante, alors, sous des conditions simples sur la matrice de génération, le "meilleur jusqu'à présent", avec une probabilité de 1, finira par devenir un minimum global, et la matrice de génération sera finie. Le temps attendu et la variance du temps nécessaire pour atteindre le minimum global seront finis. Cela découle de la théorie classique des chaînes de Markov finies. Nous verrons que pour certains problèmes de petite et moyenne taille, les tables de température constante semblent mieux fonctionner que les algorithmes de simulation basés sur la table de refroidissement.
- **Générer des voisins** : Le choix des voisins pour une solution donnée affecte grandement les performances de l'algorithme. Veiller à ce que la génération de voisins soit à la fois diversifiée et efficace.
- **Critères d'acceptation** : Le critère d'acceptation détermine si une nouvelle solution est acceptée ou rejetée. Le critère le plus courant est le critère de Metropolis, mais des variantes telles que le critère de Gibbs peuvent également être efficaces.
- **Vitesse de refroidissement** : La vitesse à laquelle la température diminue peut être ajustée pendant le processus de recuit. Les taux de refroidissement plus lents permettent une exploration plus poussée, tandis que les taux plus rapides mettent l'accent sur l'exploitation.
- **Critères d'arrêt** : Décidez d'un critère d'arrêt en fonction du problème et des ressources informatiques disponibles. Il peut s'agir d'un nombre maximal d'itérations, ou de l'atteinte d'une certaine température.

## I.5. Techniques de perturbation :

Les techniques de perturbation dans le recuit simulé sont utilisées pour améliorer l'exploration de l'espace des solutions, ce qui peut aider l'algorithme à échapper aux optima locaux et à trouver de meilleures solutions. Voici quelques techniques de perturbation courantes.

- **Sauts aléatoires** : Parfois, au lieu de générer une solution voisine basée sur une recherche locale, une solution aléatoire est générée. Cela permet d'explorer des régions éloignées de l'espace de solution.
- **Perturbation** : Après avoir atteint un minimum local, la solution actuelle est perturbée (modifiée) pour introduire un caractère aléatoire. Cela peut aider l'algorithme à échapper au minimum local et à poursuivre son exploration.
- **Augmentation de la température** : Occasionnellement, la température est augmentée pour permettre à l'algorithme d'explorer plus librement. Cela peut s'avérer utile lorsque la température a baissé trop rapidement, limitant ainsi l'exploration.
- **Recherche Tabu** : La recherche Tabu peut être combinée avec le recuit simulé pour éviter de revenir sur des solutions explorées précédemment. Cela peut aider l'algorithme à explorer de nouvelles régions de l'espace de solution.

- **Voisinage dynamique** : au lieu d'utiliser un voisinage fixe pour générer des solutions de voisinage, la taille ou la structure du voisinage peut être ajustée en fonction de la progression de l'algorithme.
- **Mécanismes de mémoire** : Conserver une mémoire des solutions ou des mouvements passés afin d'éviter d'y revenir trop tôt. Cela peut aider l'algorithme à explorer différentes parties de l'espace de solution.

- 

## I.6. Améliorations et variantes:

Le recuit simulé peut être amélioré et adapté de diverses manières afin d'améliorer ses performances et son applicabilité à différents types de problèmes d'optimisation. Voici quelques améliorations et variantes notables.

### I.6.1. Recuit simulé parallèle et distribué :

La parallélisation de l'algorithme de recuit simulé a fait l'objet de nombreuses expériences récemment, qui ont montré la difficulté d'obtenir des résultats satisfaisants. Les limites imposées par le modèle mathématique sous-jacent sont illustrées par le fort couplage entre les processus, qui oblige à de nombreuses synchronisations coûteuses en temps. Les méthodes existantes peuvent être divisées en deux sous-catégories : les algorithmes asynchrones (imprécis) réduisent le nombre d'opérations de synchronisation en acceptant le calcul des modifications de la fonction de coût à partir de données non mises à jour. Dans l'état actuel de la technique, il semble difficile de contrôler correctement l'erreur acceptée, sauf pour certaines questions particulières] ; les algorithmes synchronisés (exacts), au contraire, garantissent un calcul précis de la fonction de coût à l'aide de données cohérentes, (Mahfoud, S et al, (1995)).

- **Les algorithmes synchrones :**

Ce type de parallélisme est limité par le parallélisme que l'on peut extraire des fonctions de mouvement et d'évaluation. Il peut être appliqué avec succès à de grandes structures de données irrégulières, telles que les graphes. Cette approche conduit à des parallélisations à grain moyen ou fin. Elle est particulièrement intéressante lorsque le problème traité implique de grandes structures de données régulières (traitement d'images par exemple).

- **Recuits multiples et indépendants :**

Dans le cas séquentiel, lorsque l'on manque de temps, l'exécution de plusieurs opérations de recuit courtes et indépendantes avec la même table de température au lieu d'une longue montre que ces opérations de recuit courtes et indépendantes peuvent être exécutées en parallèle.

- **Recuit à températures multiples :**

Une autre méthode consiste à répartir la séquence de température entre les processeurs. Ainsi, chaque processeur effectue une opération de recuit à une température fixe. Les processeurs se synchronisent et interagissent périodiquement, et cette classe de parallélisme dans le recuit simulé est appelée recuit multi-température. Il convient de distinguer deux types de parallélisme.

-Propagation déterministe des états : Lors de la synchronisation, les meilleurs états sont propagés de manière déterministe d'une température plus élevée à une température plus basse.

-L'échange probabiliste d'états : Lors de la synchronisation, on échange des états obtenus selon un critère probabiliste qui vise à s'approcher d'un quasi-équilibre thermique.

### **I.6.2. Algorithme génétique :**

Les algorithmes génétiques (AG) sont basés sur les mécanismes de la sélection naturelle et des traits génétiques naturels. L'AG fonctionne avec une population initiale composée d'une série de variantes appelées chromosomes qui portent des paramètres ou des traits et mesurent la population. Il y a trois déclencheurs en particulier, la sélection, le croisement et la mutation pour créer de nouvelles populations à partir d'anciennes. Dans la sélection, un ordre de chromosomes est choisi comme parents initiaux au stade de la génération en fonction de leur aptitude. Les plus aptes ont une plus grande chance de survie ainsi qu'une plus grande probabilité de reproduire un plus grand nombre de descendants. Par l'accouplement. La mutation, un changement discrétionnaire dans les gènes, est effectuée pour protéger les différents traits génétiques d'une population (Mahfoud, S, et al. (1995)).

- **Hybrid GA-SA :**

Un hybride entre l'AG et l'AS peut parvenir à équilibrer la puissance des algorithmes pour permettre à l'AG d'explorer un vaste espace de recherche et à l'AS d'exploiter les zones de recherche locales. Alors que l'AS standard commence généralement par une configuration initiale aléatoire des paramètres, dans la solution proposée, la meilleure solution de l'AG est utilisée comme configuration initiale dans l'AS (Fatyanosa, T. N. et al, (2016)).

Cette combinaison a déjà fait l'objet d'une étude qui a prouvé que l'hybride GA-SA est supérieur à l'AG et à l'AS ordinaires. Une autre étude a montré que l'hybride GA-SA peut résoudre le problème de l'ordonnement des magasins à flux continu. Les chercheurs mettent en œuvre une hybridation entre l'AG pour explorer un vaste espace de recherche et la recherche de voisinage variable (VNS) pour exploiter les zones de recherche locales. Ils ont constaté que l'hybridation donne de meilleurs résultats que l'approche séquentielle.

- **La recherche Tabou :**

La recherche Tabou est une technique de haut niveau ou une méta-stratégie pour résoudre les problèmes d'optimisation combinatoire (Glover, 1989, 1990). Il s'agit d'une procédure itérative qui part de certaines solutions initiales possibles et tente d'identifier une meilleure solution. La technique effectue plusieurs déplacements voisins et choisit celui qui produit la meilleure solution parmi tous les déplacements candidats pour l'itération en cours. Cette meilleure solution candidate peut ne pas optimiser la solution actuelle. Les contraintes tabou permettent à la recherche de contourner les points d'optimisation locaux tout en effectuant le meilleur mouvement possible à chaque itération. Les contraintes taboues sont appliquées par une liste tabou qui stocke les attributs de mouvement afin d'éviter les inversions de mouvement.

Les attributs de mouvement afin d'éviter les inversions de mouvement. La liste taboue est associée à une taille et peut être visualisée comme une fenêtre sur les mouvements acceptables. Les mouvements qui ont tendance à se rétracter. Les mouvements à l'intérieur de cette fenêtre sont interdits.

### **I.7. Techniques adaptatives :**

Les programmes de refroidissement adaptatifs sont un élément essentiel du recuit simulé, car ils régissent la vitesse à laquelle la température diminue au cours du processus d'optimisation. Les programmes de refroidissement traditionnels, tels que la décroissance géométrique ou exponentielle, diminuent la température selon un schéma fixe. Les programmes de refroidissement adaptatifs, en revanche, ajustent le taux de refroidissement en fonction de l'état actuel du système ou de la progression de la recherche. Cet ajustement dynamique peut améliorer les performances du recuit simulé en équilibrant mieux l'exploration et

l'exploitation, ce qui conduit à une convergence plus rapide ou à de meilleures solutions. Voici quelques approches courantes du refroidissement adaptatif dans le recuit simulé :

- **Refroidissement adaptatif basé sur le taux d'acceptation :**

Mécanisme : La vitesse de refroidissement est ajustée en fonction du taux d'acceptation des nouvelles solutions. Si de nombreuses nouvelles solutions sont acceptées, la température diminue lentement pour encourager la poursuite de l'exploration. Si peu de solutions sont acceptées, la température est diminuée plus rapidement pour accélérer la convergence.

- **Programmation adaptative basée sur le retour d'information :**

Mécanisme : L'algorithme utilise le retour d'information du processus d'optimisation, tel que les changements dans la qualité des solutions ou d'autres indicateurs heuristiques, pour ajuster la température.

- **Longueur de la chaîne de Markov adaptative.**

- **Refroidissement adaptatif basé sur l'énergie.**

Le réglage des paramètres de l'AS consisté à rechercher les valeurs optimales qui conduisent à une bonne solution dans un temps de traitement raisonnable pour chacun des paramètres suivants :

- Température initiale (maximale)  $t_{max}$  (pour atteindre le point de fusion du métal) .
- Fonction de réduction de la température.
- La vitesse de refroidissement, qui est le temps de recuit, (nombre d'itérations) à la température T.
- le processus d'exploration des voisins qui permet de calculer le voisin de l'état  $x_0$  à la température T.

## I.8. Evaluation des performances :

Lors de l'évaluation des performances du recuit simulé, plusieurs paramètres clés doivent être pris en compte :

### I.8.1. Performance asymptotique :

Les résultats de convergence asymptotique pour le recuit simulé ont généralement pris deux directions : l'algorithme a été modélisé soit comme une séquence de chaînes de Markov homogènes, soit comme une seule chaîne de Markov inhomogène.

- **Approche par chaîne de Markov homogène :**

L'approche de la chaîne de Markov homogène suppose que chaque température  $t_k$  est fixée pour un nombre suffisant d'itérations  $m$  de la matrice aléatoire  $p_k$  afin d'atteindre sa distribution stationnaire (état d'équilibre)  $\pi_k$ . Toutefois, l'indice  $k$  doit être interprété comme l'indice dual  $k, m$ , où une série d'itérations de recuit simulé  $m=1, 2, \dots, M_k$  se produit pour chaque  $k$  fixé. Il s'ensuit qu'il existe une distribution stationnaire à chaque  $k$  itérations. L'existence d'une distribution stationnaire à chaque itération  $k$  découle de ce qui suit

**Remarque :** pour s'assurer que le théorème 1 est compatible avec l'algorithme de recuit simulé décrit à la section I.3.2, sans perte de généralité, laissons  $t_k$  être une fonction uniquement de chaque itération  $k$  de la boucle extérieure et laissons le nombre d'itérations  $M_k$  de la boucle intérieure et le nombre d'itérations  $k$  de la boucle extérieure s'approcher de l'infini).

**Théorème 1** Soit  $P_k(\omega, \omega')$  la probabilité de passer de la solution  $\omega$  à la solution  $\omega'$  en une seule itération dans la boucle extérieure  $k$ , et que  $p_k^m(\omega, \omega')$  soit la probabilité de passer de la solution  $\omega$  à la solution  $\omega'$  en  $m$  boucles intérieures. Si la chaîne de Markov associée à  $p_k^m(\omega, \omega')$  est irréductible et acyclique avec un nombre fini de solutions, alors la borne

$\lim_{m \rightarrow \infty} p_k^m(\omega, \omega') = \pi_k(\omega')$  existe pour tous les  $\omega, \omega' \in \Omega$  et  $k$  itérations. De plus,  $\pi_k(\omega')$  est la seule solution strictement positive.

$$\pi_k(\omega') = \sum_{\omega \in \Omega} \pi_k(\omega) P_k(\omega, \omega'), \text{ pour tous } \omega' \in \Omega, \quad (1.6)$$

Et

$$\sum_{\omega \in \Omega} \pi_k(\omega) = 1. \quad (1.7)$$

Les conditions essentielles à l'existence des distributions stationnaires et à la convergence de la séquence de vecteurs  $\pi_k$  sont les suivantes :

1. irréductibilité de la matrice de transition (pour toute boucle extérieure finie  $k$ , la matrice de transition est irréductible). Peut attribuer un chemin de probabilité non nulle entre deux solutions  $\omega, \omega' \in \Omega$ ,
2. apériodicité (à partir de la solution  $\omega' \in \Omega$ ), il est possible de se ramener à  $\omega, \omega'$  de période 1.
3. Une distribution de probabilité stationnaire non nulle, lorsque le nombre de boucles extérieures  $k$  s'approche de l'infini.

Notez que toutes les preuves de convergence du recuit simulé dans la littérature basée sur la théorie des chaînes de Markov homogènes, qu'elles soient explicites ou implicites, utilisent la condition suffisante de réversibilité (également appelée équilibre détaillé ; définie comme la condition de réversibilité de la chaîne de Markov). de réversibilité (également appelée équilibre détaillé ; définie comme

$$\pi_k(\omega) P_k(\omega, \omega') = \pi_k(\omega') P_k(\omega', \omega), \text{ pour tous } \omega, \omega' \in \Omega, \text{ et toutes les itérations } k. \quad (1.8)$$

La réversibilité est une condition suffisante pour qu'une solution unique existe pour les équations (1.6) et (1.7) à chaque itération de boucle extérieure  $k$ . Une condition nécessaire pour la réversibilité est la multiplicativité. C'est-à-dire, pour tout trois solutions  $\omega, \omega', \omega'' \in \Omega$  telles que  $f(\omega) \leq f(\omega') \leq f(\omega'')$  et pour toutes les itérations  $k$ ,

$$k_{k(\Delta_{\omega, \omega''})} = k_k(\Delta_{\omega, \omega'}) k_k(\Delta_{\omega', \omega''}), \quad (1.9)$$

Où  $k_k(\Delta_{\omega, \omega'})$  est la probabilité d'accepter la transition de la solution  $\omega$  à la solution  $\omega'$  à l'itération  $k$  de la boucle extérieure. La réversibilité est assurée en supposant des conditions de symétrie sur les probabilités de génération de solutions  $g_k$  et soit en exprimant directement la probabilité d'acceptation à l'aide d'une forme exponentielle, soit en exigeant la condition multiplicative de l'équation (1.9).

Les preuves de convergence de la chaîne de Markov homogène dans la littérature (implicitement ou explicitement) exigent que la condition de l'équation (1.9) s'applique à la fonction d'acceptation et traitent ensuite des conditions suffisantes sur la matrice de génération de solutions  $P_k$ . Par exemple, les preuves de convergence homogènes originales exigent la condition multiplicative pour la fonction d'acceptation, puis supposent que la matrice de génération de solutions est symétrique et constante pour toutes les itérations  $k$  de la boucle externe. Rossier, y, et al(1986). Partitionnent l'espace de solution en blocs composés

de solutions voisines de valeur égale de la fonction objective, puis exigent que les probabilités de génération de solutions soient symétriques entre ces blocs. Rossier et al. expriment ensuite la fonction d'acceptation comme un rapport des probabilités de distribution stationnaire. Faigle et Schrader (1988) et Faigle et Kern (1991) utilisent une approche fondée sur la théorie des graphes pour assouplir la condition de symétrie de la matrice de génération de solutions. Cependant, cela nécessite que la fonction de probabilité d'accepter la solution satisfasse à l'équation (1.9).

Granville et coll, (1994). Une procédure de recuit simulé est proposée pour filtrer les images binaires, où la fonction d'acceptabilité dépend de la probabilité de la solution actuelle, plutôt que de la modification de la valeur de la fonction objective. La fonction de probabilité que Granville et al. existe pour accepter une solution candidate dans (la boucle externe) l'itération  $k$  dépend du rapport entre la probabilité constante de la solution actuelle de l'itération  $k - 1$  et la probabilité constante de la solution initiale (qui dépend de l'estimation du maximum de vraisemblance). La probabilité d'acceptation est

$$\xi_k = q \pi_0(\omega) / \pi_{k-1}^{\phi(k)}(\omega') \quad (1.10)$$

Où  $q = \inf_{\omega \in \Omega} \pi(\omega) / \sup_{\omega' \in \Omega} \pi(\omega')$  ( $q$  doit aussi être estimé), et  $\phi(k)$  est une fonction lentement croissante. Par conséquent, la probabilité d'une transition de solution ne tient pas compte de la valeur de la fonction objective de la solution candidate. Granville et al, (1994). Fournissent une preuve de la convergence asymptotique de cette approche, mais notent que la méthodologie de la preuve ne montre pas que l'ensemble des solutions globalement optimales sont asymptotiquement uniformément distribués.

Le recuit simulé et la théorie de la convergence homogène sont basés sur les travaux de Metropolis et al. qui traitent des problèmes de mécanique statistique à l'équilibre. Pour comprendre cette relation, considérons un système en équilibre thermique avec son environnement, en solution (état)  $S$  avec une énergie  $F(S)$ . La densité de probabilité dans l'espace des phases du point représentant  $S$  est proportionnelle à

$$\exp(-f(s)/b_t). \quad (1.11)$$

Où  $b$  est la constante de Boltzmann et  $T$  la température absolue de l'environnement. Par conséquent, la proportion de temps que le système passe dans la solution  $S$  est proportionnelle à l'équation (1.11), et la densité de probabilité d'équilibre pour tous les  $S \in \Omega$  est donc la suivante

$$\pi_s = \frac{\exp(-F(s)/bT)}{\int \exp(-F(s)/bT) ds}. \quad (1.12)$$

L'espérance de toute fonction solution valide  $f(S)$  est donc la suivante

$$E[f] = \frac{\int f(s) \exp(-F(s)/bT) ds}{\int \exp(-F(s)/bT) ds}. \quad (1.13)$$

Malheureusement, pour de nombreuses fonctions solutions, l'équation (1.13) ne peut pas être évaluée analytiquement. Hammersley et Handscomb (1964) notent que l'on pourrait théoriquement utiliser des techniques naïves de Monte Carlo pour estimer la valeur des deux intégrales de l'équation (1.13). Toutefois, cette méthode échoue souvent dans la pratique, car le facteur exponentiel signifie qu'une partie importante des intégrales est concentrée dans

une très petite région de l'espace des solutions  $\Omega$ . Ce problème peut être surmonté en utilisant l'échantillonnage d'importance, en générant des solutions avec la densité de probabilité de l'équation (1.12).

Cette approche semble également vouée à l'échec en raison de l'intégrale au dénominateur de l'équation (1.12). Cependant, Metropolis et al (1953). Résolvent ce problème en discrétisant d'abord l'espace des solutions, de sorte que les intégrales dans les équations (1.12) et (1.13) sont remplacées par des sommations sur l'ensemble des solutions discrètes  $\omega' \in \Omega$ , puis en construisant une chaîne de Markov irréductible et apériodique avec des probabilités de transition  $P(\omega, \omega')$  telles que

$$\pi(\omega') = \sum_{\omega \in \Omega} \pi(\omega) P(\omega, \omega') \text{ pour tous } \omega' \in \Omega. \quad (1.14)$$

Où

$$\pi(\omega') = \frac{\exp(-F(\omega')/bT)}{\sum_{\omega \in \Omega} \exp(-F(\omega)/bT)} \text{ pour tous } \omega' \in \Omega. \quad (1.15)$$

Notez que pour calculer la distribution d'équilibre  $\pi$ , il n'est pas nécessaire de calculer le dénominateur de l'équation (1.13) (une constante de normalisation). Au lieu de cela, il suffit de calculer les rapports  $\pi(\omega')/\pi(\omega)$  et de définir une matrice de transition  $P$  qui satisfait à l'équation (1.14). Hammersley et Handscomb(1964) montrent que Metropolis et al(1953). Y parviennent en définissant  $P$  comme le produit des probabilités symétriques de génération de solutions  $g(\omega, \omega')$  et des ratios d'équilibre  $\pi(\omega')/\pi(\omega)$ ,

$$p(\omega, \omega') = \begin{cases} g(\omega, \omega')\pi(\omega')/\pi(\omega) & \text{si } \pi(\omega')/\pi(\omega) < 1, \omega' \neq \omega \\ g(\omega, \omega') & \text{si } \pi(\omega')/\pi(\omega) \geq 1, \omega' \neq \omega \\ g(\omega, \omega') + \Delta & \text{si } \omega' = \omega \end{cases} \quad (1.16)$$

Avec  $\Delta = \sum_{\omega'' \in \Omega, \pi(\omega'') < \pi(\omega)} g(\omega, \omega'') (1 - (\pi(\omega'')/\pi(\omega)))$ , où

$$g(\omega, \omega') \geq 0, \sum_{\omega' \in \Omega} g(\omega, \omega') = 1 \text{ et } g(\omega, \omega') = g(\omega', \omega) \text{ pour tous } \omega, \omega' \in \Omega. \quad (1.17)$$

L'utilisation de rapports de probabilité stationnaires pour définir les probabilités d'acceptation des solutions, combinée à des probabilités symétriques de génération de solutions, permet à Metropolis et al. D'utiliser la condition de réversibilité de l'équation (1.8) pour montrer que les équations (1.16) et (1.17) satisfont à l'équation (1.14). Les preuves homogènes de convergence pour le recuit simulé deviennent plus difficiles à établir lorsque la condition de réversibilité n'est pas satisfaite. Notez que l'existence d'une distribution stationnaire unique pour chaque itération de boucle extérieure  $k$  est facilement démontrée en spécifiant que chaque matrice de transition  $P_k$  est irréductible et apériodique. D'autre part, il devient très difficile de dériver une expression explicite de forme fermée pour chaque distribution stationnaire  $\pi_k$  qui reste analytiquement traitable lorsque l'espace des solutions du problème devient grand. On ne peut plus utiliser l'équation (1.8) pour décrire chaque distribution stationnaire puisque, en général, la condition multiplicative n'est pas remplie. Décrire chaque distribution stationnaire, puisque, en général, la condition multiplicative n'est pas remplie. A la place, il faut résoudre directement le système d'équations formé par les

équations (1.6) et (1.7). Par exemple, Davis tente d'obtenir une expression de forme fermée pour  $\pi_k$  en utilisant la règle de Cramer et en réécrivant les équations (1.6) et (1.7) comme suit

$$\pi_k(I - P_k) = 0 \quad (1.18)$$

ET

$$\pi_k e^T = 1. \quad (1.19)$$

$I$  est la matrice identité et  $e_t$  est une vectrice colonne de uns. On notera que la matrice de transition  $P_k$  associée à l'équation (1.18) est une matrice de transition d'ordre  $\text{Card}(\Omega) \times \text{Card}(\Omega) - 1$ . Par conséquent, en supprimant une équation quelconque de l'équation (1.18) et en la remplaçant par l'équation (1.19), on obtient l'ensemble des équations linéairement indépendantes pour  $\text{card}(\Omega)$

$$\pi_k(I - P_k)^{[i]} = e_i, \quad (1.20)$$

Où la matrice carrée  $(I - P_k)^{[i]}$  est obtenue en remplaçant la  $i$ ème colonne de la matrice  $(I - P_k)$  par une vectrice colonne de uns. Le vecteur  $e_i$  est une vectrice ligne de zéros, à l'exception d'un un à la  $i$ ème position. Puisque  $(I - P_k)^{[i]}$  est de plein rang, son déterminant (écrit  $\det((I - P_k)^{[i]})$  est non nul. Définir  $(I - P_k)\omega$  comme étant la même matrice que  $(I - P_k)$ , sauf que les éléments de la  $i$ ème ligne de  $(I - P_k)$  sont remplacés par le vecteur  $e_\omega$ . Par conséquent, pour toutes les itérations  $k$ ,

$$\pi_k(\omega) = \frac{\det((I - P_k)^{[i]\omega})}{\det((I - P_k)^{[i]})}, \text{ pour tous } \omega \in \Omega. \quad (1.21)$$

On tente de résoudre l'équation (1.21) pour chaque  $\omega \in \Omega$  par le biais d'une expansion en série de Taylor multi variée de chaque déterminant, mais la méthode n'a pas permis de produire une expression analytique en forme fermée. Dans l'ensemble, la difficulté d'exprimer explicitement les distributions stationnaires pour les grands espaces de solution, combinée à la limitation du nombre de conditions de la matrice de transition pour un grand  $k$ , suggère qu'il est très difficile de prouver la convergence asymptotique de l'algorithme de recuit simulé en appliquant la méthode de la matrice de transition à la matrice de transition. L'algorithme de recuit simulé en traitant les équations (1.5) et (1.6) comme un problème d'algèbre linéaire. Lundy et Mees (1986) notent que pour chaque itération de boucle extérieure fixe  $k$ , la convergence vers le vecteur de distribution de probabilité de l'équilibre de la solution  $\pi_k$  (en termes de la distance euclidienne entre  $P(m)_k$  et  $\pi_k$ , comme  $m \rightarrow +\infty$ ) est géométrique puisque l'espace de solution est fini et que le facteur de convergence est donné par la deuxième plus grande valeur propre de la matrice de transition  $P_k$ . Ce résultat est basé sur un théorème de convergence standard pour les chaînes de Markov homogènes irréductibles et aperiodiques. Il convient de noter qu'un vaste espace de solution empêche le calcul pratique de cette valeur propre. Lundy et Mees (1986) supposent que lorsque la température  $t_k$  est proche de zéro, la deuxième valeur propre la plus importante sera proche de un pour les problèmes avec optima locaux, et donc la convergence vers la distribution d'équilibre sera très lente (rappelons que la valeur propre dominante pour  $P_k$  est 1, avec une multiplicité algébrique). Lundy et Mees (1986) [utilisent cette conjecture pour justifier le fait que le recuit

simulé doit être initié avec une température relativement élevée. L'hypothèse de stationnarité pour chaque itération  $k$  de la boucle extérieure limite l'application pratique de la théorie des chaînes de Markov homogènes. Romeo et Sangiovanni-Vincentelli (1991) montrent que si l'équilibre (pour une chaîne de Markov qui satisfait à la condition de réversibilité) est atteint en un nombre fini d'étapes, alors il peut être atteint en une seule étape. Ainsi, Romeo et Sangiovanni-Vincentelli (1991) conjecturent qu'il n'y a pratiquement aucun espoir pour les versions les plus utilisées du recuit simulé d'atteindre l'équilibre en un nombre fini d'itérations.

- **Approche par chaîne de Markov inhomogène :**

La deuxième approche de convergence pour le recuit simulé, est basée sur la théorie de la chaîne de Markov inhomogène. Dans cette approche, il n'est pas nécessaire que la chaîne de Markov atteigne une distribution stationnaire (par exemple, la boucle interne du recuit simulé ne doit pas être infiniment longue) pour chaque boucle externe  $k$ . En revanche, une séquence infinie d'itérations  $k$  (boucles externes) doit encore être examinée, à condition que le paramètre de température  $t_k$  se refroidisse suffisamment lentement. La preuve donnée par Mitra et al (1986). Est basée sur la satisfaction des conditions d'ergodicité faible et forte de la chaîne de Markov inhomogène. La preuve nécessite quatre conditions :

1. La chaîne de Markov du recuit simulé inhomogène doit être faiblement ergodique (c'est-à-dire que la dépendance à l'égard de la solution initiale disparaît dans la limite).
2. Il doit exister un vecteur propre  $\pi_k$  avec une valeur propre de 1 de sorte que les équations (1.6) et (1.7) s'appliquent à chaque itération  $k$ .
3. La chaîne de Markov doit être fortement ergodique (c'est-à-dire que la chaîne de Markov doit être faiblement ergodique et que la séquence des vecteurs propres  $\pi_k$  doit converger vers une forme limite), c'est-à-dire,

$$\sum_{k=0}^{\infty} \|\pi_k - \pi_{k+1}\| < +\infty. \quad (1.22)$$

4. La séquence des vecteurs propres doit converger vers une forme où toute la masse de probabilité est concentrée sur l'ensemble des solutions globalement optimales  $\omega^*$ . Par conséquent

$$\lim_{K \rightarrow \infty} \pi_K = \pi^{opt}, \quad (1.23)$$

Où  $\pi^{opt}$  est la distribution d'équilibre avec seulement les optima globaux ayant des probabilités supérieures à 0. (Notez que l'ergodicité faible et forte est équivalentes pour la théorie des chaînes de Markov homogènes). Mitra et al (1986). Satisfont à la condition 1 (ergodicité faible) en établissant d'abord une borne inférieure sur la probabilité d'atteindre une solution à partir d'un minimum local et en montrant ensuite que cette borne ne s'approche pas trop rapidement de zéro. Par exemple, ils définissent la limite inférieure des probabilités de transition du recuit simulé dans l'équation (1.5) comme suit

$$p^{(m)}(\omega, \omega') \geq w^m \exp(-m\Delta_L / t_{km-1}), \quad (1.24)$$

pour tout entier  $k$  supérieur ou égal à un entier fixe  $k_0$ , où  $m$  est le nombre de transitions nécessaires pour atteindre n'importe quelle solution à partir de n'importe quelle solution de valeur de fonction objective non maximale,  $w > 0$  est la limite inférieure de la génération de

solutions en une étape. Valeur de la fonction objective,  $w > 0$  est une borne inférieure sur les probabilités de génération de solutions en une étape,  $\Delta_L$  est l'augmentation maximale en une étape de la valeur de la fonction objective entre deux solutions, et  $t_{km-1}$  est la température à l'itération  $km-1$ . Mitra et al, (1986). Montrent que la chaîne de Markov est faiblement ergodique si, pour tout entier fixe  $k_0$

$$\sum_{k=k_0}^{\infty} \exp(-m\Delta_L/t_{km-1}) = +\infty. \quad (1.25)$$

Par conséquent, l'ergodicité faible est obtenue si la température  $t_k$  est réduite suffisamment lentement à zéro pour que l'équation (1.25) soit satisfaite. En général, la séquence (infinie) de températures  $\{t_k\}$ ,  $k = 1, 2, \dots$ , doit satisfaire aux conditions suivantes

$$t_k \geq \frac{\beta}{\log(k)}, \quad (1.26)$$

Où  $\lim_{k \rightarrow \infty} t_k = 0$ ,  $\beta$  est une constante dépendant du problème et  $k$  est le nombre d'itérations. Mitra et al (1986) montrent que les conditions (1.2), (1.3) et (1.4) sont satisfaites en utilisant la théorie de la chaîne de Markov homogène développée pour les probabilités de transition de l'équation (1.5), à condition que la fonction de génération de solution soit symétrique. Romeo et Sangiovanni-Vincentelli(1991) notent que si le programme de refroidissement logarithmique dans l'équation (1.26) est une condition de convergence suffisante, il n'existe que quelques valeurs de  $\beta$  qui rendent la règle logarithmique également nécessaire. De plus, il existe un choix unique pour  $\beta$  qui rend la règle logarithmique à la fois nécessaire et suffisante pour la convergence du recuit simulé vers l'ensemble des optima globaux], Hajek(1988) a été le premier à montrer que le programme de refroidissement logarithmique (équation (1.26)) est à la fois nécessaire et suffisant, en développant une borne inférieure étroite pour  $\beta$ , à savoir la profondeur du minimum local le plus profond qui n'est pas un minimum global, sous une hypothèse de réversibilité faible (notez que Hajek exige que la profondeur des optima globaux soit infiniment grande). Hajek(1988) définit une chaîne de Markov comme étant faiblement réversible si, pour toute paire de solutions  $\omega, \omega' \in \Omega$  et pour tout nombre réel non négatif  $h$ ,  $\omega$  est atteignable à la hauteur  $h$  à partir de  $\omega'$  si et seulement si  $\omega'$  est atteignable à la hauteur  $h$  à partir de  $\omega$ . Notons que Hajek ne cherche pas à satisfaire les conditions d'ergodicité faible et forte, mais utilise plutôt une approche probabiliste plus générale pour développer une borne inférieure sur la probabilité d'échapper à des optima locaux, mais pas globaux. Connors et Kumar(1989) justifient les conditions nécessaires et suffisantes de Hajek en utilisant les ordres de récurrence,

$$B_i \equiv \sup \left\{ x \geq 0: \sum_{k=0}^{\infty} \exp\left(-\frac{x}{t_k}\right) \pi_k(\omega) = +\infty \right\} \text{ pour tous } i \in \Omega. \quad (1.27)$$

Connors et Kumar(1989) montrent que ces ordres de récurrence quantifient le comportement asymptotique de la probabilité de chaque solution dans la distribution des solutions. Le résultat clé est que la chaîne de Markov inhomogène du recuit simulé converge, au sens de Cesaro, vers l'ensemble des solutions ayant les ordres de récurrence les plus élevés. Borkar (1992) améliore ce résultat de convergence en utilisant un résultat de dichotomie convergence/oscillation pour les martingales. Tsitsiklis (1989) utilise des bornes et

des estimations pour des chaînes de Markov approximativement stationnaires et singulièrement perturbées afin de développer une théorie de la convergence. Qui subsume la condition de réversibilité faible dans. Notons que Tsitsiklis (1989) définit  $N(h) \subset \Omega$  comme l'ensemble de tous les minima locaux (en termes de valeur de la fonction objective) de profondeur  $h + 1$  ou plus. Par conséquent,  $\beta$  est le plus petit  $h$  tel que tous les minima locaux (mais pas globaux) ont une profondeur inférieure ou égale à  $h$ . Tsitsiklis (1989) conjecture que sans une certaine forme de réversibilité, il n'existe pas de  $h$  tel que les optima globaux soient contenus dans l'ensemble des optima locaux. Notons que la condition multiplicative (1.9) est requise (explicitement ou implicitement) pour les preuves de convergence. Anily et Federgruen,(1987) utilisent des techniques d'analyse des perturbations pour prouver la convergence d'un algorithme stochastique particulier d'escalade en limitant les écarts de la séquence des distributions stationnaires de l'algorithme particulier d'escalade par rapport à la séquence des distributions stationnaires connues correspondant à un algorithme de recuit simulé. En général, cette approche de preuve de convergence n'est utile que pour une classe restrictive d'algorithmes de recuit simulé, puisque le nombre de conditions de la matrice de transition croît de manière exponentielle lorsque le nombre d'itérations  $k$  devient important. Anily et Federgruen(1987) présentent également une preuve de convergence pour le recuit simulé avec des fonctions de probabilité d'acceptation générales. En utilisant la théorie de la chaîne de Markov inhomogène, ils prouvent la convergence sous les conditions nécessaires et suffisantes suivantes :

1. La fonction de probabilité d'acceptation doit, pour toute itération  $k$ , permettre à toute transition de montée de se produire avec une probabilité positive.
2. La fonction de probabilité d'acceptation doit être limitée et asymptotiquement monotone, avec une limite de zéro pour les transitions de solution de type "hillclimbing".
3. Dans la limite, la distribution de probabilité stationnaire doit avoir une masse de probabilité de 0 pour toute solution non optimale au niveau mondial.
4. La probabilité d'échapper à toute solution localement (mais non globalement) optimale ne doit pas s'approcher trop rapidement de 0.

Anily et Federgruen (1987) utilisent la condition (3) pour assouplir la condition multiplicative de la fonction d'acceptation (1.9). Toutefois, dans la pratique, la condition (3) serait très difficile à vérifier sans supposer que l'équation (1.9) est valable. La condition (4) est la condition nécessaire pour que le taux de probabilité des transitions en montée approche 0. La condition (4) s'exprime quantitativement comme suit : supposons que  $t_k$  soit défini par l'équation (1.2) et définissons la probabilité d'acceptation minimale en une étape comme suit

$$\underline{a}_k = \min_{\omega \in \Omega, \omega' \in N(\omega)} at_k(\omega, \omega'). \quad (1.28)$$

Définir l'ensemble des optima locaux  $L \subset \Omega$  tels que  $\omega \in L$  implique que  $f(\omega) \leq f(\omega')$  pour tout  $\omega' \in N(\omega)$ , et laisser

$$\bar{a}_k = \min_{\omega \in L, \omega' \in N(\omega)/L} at_k(\omega, \omega'). \quad (1.29)$$

Enfin, toute solution  $\omega' \in \Omega$  doit pouvoir être atteinte à partir de toute solution  $\omega \in \Omega$  en  $q$  transitions ou moins. S'il existe des solutions localement optimales (non globales),

$$\sum_{K=1}^{\infty} (\underline{a}_k)^q = +\infty, \quad (1.30)$$

Et que les conditions (1), (2) et (3) sont remplies, alors l'algorithme de recuit simulé convergera asymptotiquement vers l'ensemble des optima globaux avec une probabilité de 1. Cependant, si solutions localement optimales (non globales) existent ET que

$$\sum_{k=1}^{\infty} \bar{a}_k < +\infty, \quad (1.31)$$

Alors l'algorithme de recuit simulé ne convergera pas toujours vers l'ensemble des optima globaux avec une probabilité de 1. Johnson ET Jacobson, (1991) assouplissent les conditions suffisantes trouvées en utilisant un argument de chemin entre les optima globaux et les optima locaux (mais non globaux). Yao et Li, (2002) discutent également des algorithmes de recuit simulé avec des probabilités d'acceptation générales, bien que leur principale contribution concerne les distributions générales de génération de voisinage. Schuur, (1997) fournit une description des fonctions d'acceptation garantissant la convergence de l'algorithme de recuit simulé associé vers l'ensemble des optima globaux.

Le concept de preuve hétérogène est plus fort que l'approche homogène en ce qu'il fournit des conditions nécessaires pour le taux de convergence, mais sa nature approximative suggère que l'application pratique n'est peut-être pas possible. Romeo et Sangiovanni-Vincentelli (1991) notent qu'"il n'y a aucune raison de croire que la troncature de la séquence logarithmique de température aboutira à une bonne configuration, puisque la queue de la séquence est l'élément clé de la preuve". En outre, le programme de refroidissement logarithmique impose un taux de convergence très lent. C'est pourquoi les travaux les plus récents se sont concentrés sur les moyens d'améliorer le comportement du recuit simulé en temps fini, en modifiant ou en combinant l'algorithme avec d'autres méthodes de recherche.

### 1.8.2. Performance en temps fini :

Au cours de la dernière décennie, un nombre croissant de travaux a été consacré au comportement en temps fini du recuit simulé. Jacobson, S.H. et al (2004), présentent des conditions de convergence nécessaires et suffisantes (asymptotiques) pour les algorithmes d'escalade de colline généralisés qui incluent le recuit simulé comme un cas spécial. Ils introduisent également de nouvelles mesures de performance qui peuvent être utilisées pour évaluer et comparer les algorithmes d'escalade de collines généralisés convergents et non convergents avec la recherche locale à redémarrage aléatoire.

Desai, M.P. (1999), se concentre sur les performances en temps fini en incorporant les informations approximatives sur la taille fournies par certaines valeurs propres associées à la matrice de transition.

Pour évaluer le temps d'exécution prévu requis par un algorithme de recuit simulé pour atteindre une solution d'une qualité prédéfinie, Wood, G.R., (2004), présentent une approche de la modélisation et de l'analyse d'un algorithme d'optimisation globale stochastique générique utilisant une série de processus stochastiques, qui se termine par un processus de recherche adaptative avec retour en arrière.

## I.9.Comparaison avec d'autres techniques d'optimisation :

- **Algorithmes génétiques (AG):**

Algorithmes génétiques : inspirés de l'évolution biologique, ils utilisent des mécanismes tels que la sélection, le croisement et la mutation. Ils travaillent sur une gamme de solutions potentielles, permettant une recherche plus diversifiée de l'espace de solution.

Points forts L'AG fonctionne efficacement sur des problèmes complexes avec des espaces de recherche très difficiles ; il maintient simultanément plusieurs solutions candidates, ce qui peut conduire à une recherche plus robuste, (Fatyanosa, T. N, et al, (2016)).

Faiblesses L'AG peut être coûteux en termes de calcul en raison du maintien et de l'évaluation d'un ensemble de solutions ; il nécessite un réglage minutieux des paramètres tels que les taux de croisement et de mutation.

Le recuit simulé les points forts Simplicité de mise en œuvre ; bon pour les problèmes où une solution approximative est acceptable ; efficace pour les problèmes comportant un grand nombre de minima locaux.

Faiblesses le RS Peut être lent à converger vers une solution ; les performances dépendent fortement du programme de refroidissement.

- **Descente en gradient (GD):**

RS : Ne nécessite pas d'informations sur le gradient et peut traiter des variables discrètes et continues, (Cai, Z. (2021)).

Points forts Peut traiter des fonctions non différentiables et des problèmes d'optimisation discrets.

Faiblesses Moins efficace dans la recherche de solutions précises que les méthodes basées sur le gradient.

Descente de gradient : Suit le gradient de la fonction pour trouver le minimum ou le maximum, utilisé principalement lorsque le gradient peut être calculé.

Points forts Très efficace pour les problèmes lisses et continus ; très efficace lorsqu'il est combiné avec la rétro propagation dans la formation des réseaux neuronaux.

Faiblesses A tendance à se bloquer dans des minima locaux ; nécessite que la fonction soit différentiable.

- **Optimisation par essais de particules (PSO):**

Recuit simulé : Utilise des variations aléatoires et accepte les moins bonnes solutions de manière probabiliste, (Wang, R, (2018)).

Points forts Simple et facile à mettre en œuvre ; permet d'échapper aux optima locaux.

Faiblesses le RS Peut nécessiter beaucoup de temps pour affiner le programme de refroidissement.

Optimisation par essaimage de particules : Utilise une population de solutions candidates (particules) qui ajustent leur position dans l'espace de recherche en fonction de leur propre expérience et de celle de leurs voisins.

Points forts Naturellement parallèle et pouvant être facilement mis en œuvre sur des architectures parallèles ; bonne capacité à converger rapidement vers une solution raisonnable.

Faiblesses Vulnérable à l'enlèvement dans des optima locaux ; les paramètres tels que la vitesse et la taille de la population doivent être réglés avec soin.

- **Recherche Tabu (RT) :**

Recuit simulé : Perturbations aléatoires avec acceptation sur la base d'un programme de refroidissement, (Youssef, H, et al (2001).

Points forts RS concept plus simple et plus facile à mettre en œuvre.

Faiblesses RS convergence plus lente dans certains cas

Recherche Tabu : Utilise des structures de mémoire qui décrivent les solutions visitées ou des règles définies par l'utilisateur pour éviter de revenir aux solutions visitées précédemment.

RT : Permet souvent une convergence plus rapide vers des solutions de haute qualité en évitant intelligemment les cycles.

RT : Plus complexe à mettre en œuvre ; nécessite une gestion minutieuse de la mémoire et des structures taboues.

## I.10. Orientations futures:

Le recuit simulé (RS) est une technique d'optimisation de base depuis sa popularisation dans les années 1980 pour résoudre des problèmes d'optimisation complexes. Bien qu'il s'agisse d'une technique ancienne, elle continue d'évoluer avec des adaptations qui améliorent son efficacité et élargissent son champ d'application. Voici plusieurs tendances émergentes dans la recherche sur le recuit simulé qui reflètent les innovations actuelles et les intégrations avec les nouvelles technologies :

- **Programmes de refroidissement adaptatifs:**

L'AS traditionnel utilise un programme de refroidissement prédéfini, qui consiste souvent en une diminution linéaire ou exponentielle de la température. Les recherches récentes se concentrent sur les stratégies de refroidissement adaptatives qui ajustent la température en fonction de l'état de la recherche, comme le nombre d'itérations sans amélioration ou l'ampleur des améliorations. Cette adaptabilité peut améliorer considérablement l'efficacité de l'algorithme en équilibrant dynamiquement l'exploration et l'exploitation tout au long du processus d'optimisation.

- **Améliorations spécifiques aux applications:**

L'AS est adapté à des applications spécifiques, ce qui conduit à des versions personnalisées de l'algorithme qui sont nettement plus efficaces dans des contextes particuliers. Par exemple, dans le domaine de l'apprentissage automatique, l'AS est utilisé pour optimiser les architectures de réseaux neuronaux ou les hyper paramètres. Dans le domaine de l'ingénierie, elle est appliquée à des problèmes complexes de planification et d'agencement. Chaque domaine d'application peut conduire à des modifications et à des améliorations uniques de la méthodologie de l'AS.

- **Intégration avec l'apprentissage automatique:**

Il existe un intérêt croissant pour l'utilisation de l'AS dans les pipelines d'apprentissage automatique, soit en tant qu'optimiseur pour les modèles d'apprentissage automatique, soit en utilisant des techniques d'apprentissage automatique pour améliorer l'AS lui-même. Par exemple, l'apprentissage automatique peut être utilisé pour prédire les résultats des solutions potentielles ou pour ajuster dynamiquement les paramètres de l'algorithme d'AS afin d'améliorer les performances. Cette intégration peut considérablement élargir le champ d'application et les capacités de l'AS et de l'apprentissage automatique.

- **L'informatique économe en énergie:**

L'efficacité énergétique devenant un aspect crucial de l'informatique, l'AS est adaptée pour réduire les coûts de calcul et la consommation d'énergie. Il s'agit notamment de concevoir des programmes de refroidissement respectueux de l'énergie ou d'optimiser

l'algorithme afin de réduire le nombre d'opérations gourmandes en énergie, telles que les évaluations inutiles de la fonction objective.

### **I.11.Conclusion :**

En conclusion, le recuit simulé reste un outil précieux dans la boîte à outils de l'optimiseur. Sa capacité à trouver des solutions quasi-optimales dans un large éventail de contextes, associée à des recherches et des améliorations continues, garantit sa pertinence et son efficacité dans la résolution des problèmes d'optimisation classiques et de pointe. Au fur et à mesure que la technologie informatique évolue, la portée et la capacité du recuit simulé évoluent également, s'adaptant et prospérant dans le paysage des méthodes informatiques modernes.

## Chapitre II

# Optimisation de Harris Hawks

### II.1. Introduction :

L'optimisation de Harris Hawks ou encore l'optimisation des Buses de Harris (HHO pour Harris Hawks Optimization) est un algorithme d'optimisation sans gradient, basé sur un essaim, avec plusieurs phases d'exploration et d'exploitation actives et variables dans le temps. Cet algorithme (Heidari, et al., 2019) dès les premiers jours de sa publication a attiré une attention croissante des chercheurs en raison de sa structure flexible, de ses hautes performances et de ses résultats de haute qualité. La logique principale de l'algorithme HHO est conçue sur la base du comportement coopératif et des styles de poursuite des faucons de Harris dans la nature appelés « *bond-surprise* ». Actuellement, il existe de nombreuses suggestions sur la façon d'améliorer les fonctionnalités du HHO, et il existe également plusieurs variantes améliorées du HHO dans les principaux journaux de transactions Elsevier et IEEE.

L'algorithme de Harris Hawks peut simuler différents styles de poursuite basé sur différents scénarios et comportements de fuite des proies. Plus spécifiquement, quatre stratégies de poursuite sont développées dans HHO, Il s'agit d'un siège doux, d'un siège dur, d'un siège doux avec des plongées rapide progressives et d'un siège dur avec des plongées rapides progressives. Les résultats testés sur des fonctions de référence et plusieurs problèmes d'optimisation technique confirment que HHO surpasse de nombreuses approches si bien connues telles que les algorithmes de : Essaim de particules (PSO), Loups gris (GWO), Biogéographie (BBO), Lucioles (FA), colonie de Fourmies (AC). De plus, les résultats également montrent que HHO atteint un bon équilibre entre exploration et exploitation, améliorant ainsi l'évolutivité de HHO et la capacité à obtenir des solutions de haute qualité.

HHO a été exploité pour de nombreuses applications et problèmes d'ingénierie en raison de ses fonctionnalités d'optimisation et performances compétitives. Dans ce chapitre, Nous présenterons en détail le concept des algorithmes d'optimisation de Harris Hawks.

### II.2. Contexte et inspiration :

#### II.2.1. Introduction aux buses de Harris et à leur comportement de chasse :

Les buses de Harris (*Parabuteo unicinctus*) sont d'un rapace bien connu qui survit en groupes quelque peu stables trouvés dans la moitié sud d'Arizona, États-Unis (Bednarz, J. C. (1988)). La buse de Harris se caractérise par ses activités uniques de recherche de nourriture en coopération avec d'autres membres de la famille vivant dans le même groupe stable. Selon Bednarz en 1998, ces oiseaux commencent leur mission en équipe au crépuscule du matin, souvent perchés sur des arbres géants ou des poteaux électriques dans leur domaine vital. Ils sont conscients des mouvements de leur groupe pendant l'attaque. Une fois qu'ils se sont rassemblés et que la fête a commencé, certains faucons se lèvent un par un, décrivent de courts cercles et se posent ensuite sur des perchoirs assez élevés. De cette manière, les faucons « sautent » parfois sur l'ensemble du site visé, puis se regroupent et se séparent plusieurs fois pour rechercher activement l'animal couvert, généralement un lapin.

La principale tactique des faucons de Harris pour capturer une proie est la « bond-surprise », également connue sous le nom de stratégie des « quatre attaques ». Dans cette stratégie intelligente, plusieurs faucons tentent d'attaquer en coopération depuis différentes directions et convergent simultanément vers un lapin qui s'échappe à l'extérieur de la couverture. L'attaque peut être réalisée rapidement en attrapant la proie surprise en quelques secondes, mais parfois la proie peut avoir une chance de s'échapper grâce à ses capacités. Une tactique de changement se produit lorsque le meilleur faucon (chef) s'abaisse devant la proie et se perd, et la poursuite sera poursuivie par l'un des les membres du parti. Ces activités de commutation peuvent être observées dans différentes situations car ils sont bénéfiques pour confondre le lapin qui s'enfuit. Le principal avantage de ces coopératives la tactique est que les faucons de Harris peuvent poursuivre le lapin détecté à l'épuisement, ce qui augmente sa vulnérabilité. De plus, par perplexe la proie en fuite, il ne peut pas récupérer ses capacités et enfin, il ne peut échapper à l'équipe confrontée assiéger depuis l'un des faucons, qui est souvent le plus puissant et expérimenté, capture sans effort le lapin fatigué et le partage avec d'autres membres du parti.



Fig. II.1 : Faucon (Buse) d'Harris

### II.2.2. Vue d'ensemble de l'inspiration derrière HHO :

La principale source d'inspiration de HHO est le travail collaboratif et le style de chasse coopératif des buses de Harris dans la nature, appelée ((bond surprise)). Le modèle mathématique HHO génère des modèles et des comportements dynamiques pour le développement d'un algorithme d'optimisation. (Tripathy, B. K., et al, (2022)).

L'optimiseur Harris Hawks a reçu une large attention parmi les chercheurs en termes de performances, de qualité des résultats et de convergence acceptable dans le traitement de différentes applications dans des problèmes du monde réel (Alabool, H. M et al, (2021)). Bien que l'algorithme HHO en soit encore à ses débuts, il a été clairement observé qu'il

surpassait de nombreux méta-algorithmes bien établis en termes de vitesse et de précision dans la résolution de divers problèmes de référence et de nombreux problèmes d'optimisation.

## II.3. Concepts fondamentaux de l'optimisation de Harris Hawks :

### II.3.1. Modèles mathématiques de HHO :

Cette section traite le modèle mathématique de l'algorithme HHO, qui comprend une phase d'exploration, une phase d'exploitation et un nombre d'approches d'attaque de Harris Hawks. HHO est un algorithme inspiré de la nature qui peut être appliqué à n'importe quel problème d'optimisation. Cette section présente toutes les phases de HHO, qui sont expliquées plus en détail dans les sous-sections suivantes, (Tripathy, B. K., et al (2022)).

#### II.3.1.1. Phase d'exploration :

Dans HHO, les faucons sont considérés comme des solutions candidates et, à chaque itération, la proie est considérée comme la solution optimale. Les faucons se perchent à des endroits spécifiques et observent continuellement l'environnement pour identifier les proies à l'aide de deux stratégies représentées par l'équation (2.1). Si  $p < 0.5$ , les faucons se perchent en fonction de l'emplacement des membres de la famille. Si  $p \geq 0.5$ , les faucons se perchent dans un espace aléatoire au sein de la zone de perchage.

$$A(x+1) = \begin{cases} A_r(x) - a_1 |A_a(x) - 2a_2 A(x)|, p \geq 0.5, \\ (A_{rabbit}(x) - A_p(x)) - a_3, \\ L_B + a_4(U_B - L_B), p < 0.5. \end{cases} \quad (2.1)$$

Dans l'équation (2.1), où  $A(x+1)$  désigne la position du lapin à la prochaine itération.  $A_{rabbit}(x)$  désigne la position de lapin.  $A(x)$  désigne la position actuelle des faucons.  $a_1, a_2, a_3, a_4$  et  $p$  sont des variables aléatoires allant de 0 à 1.  $L_B$  et  $U_B$  sont la limite inférieure et la limite supérieure des variables aléatoires.  $A_p(x)$  indique la position moyenne du faucon, qui est représenté dans l'équation suivante, où  $A_i(x)$  désigne l'emplacement de chaque faucon à la  $i$ ème itération et  $H$  désigne le nombre de faucons dans l'espace de recherche.

$$A_p(x) = \frac{1}{H} \sum_{i=1}^H A_i(x). \quad (2.2)$$

#### II.3.1.2. Transition de l'exploration à l'exploitation :

Cette sous-section explique la transformation de l'exploration phase à la phase d'exploitation, en fonction du niveau d'énergie de la proie à s'échapper, qui est mathématiquement définie comme suit :

$$P = 2P_0 \left(1 - \frac{x}{I}\right), \quad (2.3)$$

Où  $P$  désigne l'énergie de la proie pour s'échapper à l'itération  $x$ ,  $I$  désigne le nombre total d'itérations, et  $P_0$  désigne l'énergie initiale de la proie. Lors de chaque évacion, le niveau d'énergie de la proie tombe considérablement. Pour chaque itération,  $P_0$  va changer la valeur de  $(-1, 1)$ . Quand  $P_0$  descend de 0 à  $-1$ , la proie est épuisée ; de même, lorsque la valeur  $P_0$  est en passant de 0 à 1, la proie est renforcée. Quand  $|P| \geq 1$ , l'exploration a eu

lieu, et quand  $|P| < 1$ , l'exploitation se produit. La figure II.2.représente le comportement énergétique qui s'échappe.

### II.3.1.3.Phase d'exploitation :

Cette sous-section explique la phase d'exploitation au cours de laquelle les éperviers attaquent la proie ciblée. Ensuite, la proie tente d'échapper à l'attaque. Sur la base du comportement d'attaque des éperviers et du comportement de fuite des proies, quatre approches sont examinées dans les sous-sections suivantes.

- **Siège doux :**

Dans HHO, un siège doux se produit lorsqu'un  $a \geq 0.5$  et  $|P| \geq 0.5$ , et la proie a suffisamment d'énergie pour s'échapper de l'attaque, mais ne peut pas échapper car les éperviers encerclent la proie et l'énergie de la proie s'épuise.

$$A(x + 1) = \Delta A(x) - P|UA_{rabbit}(x) - A(x)|, \quad (2.4)$$

Et

$$\Delta A(x) = A_{rabbit}(x) - A(x). \quad (2.5)$$

$\Delta A(x)$  Désigne la différence entre le vecteur position du lapin et position actuelle à la  $x^{ième}$  itération.  $U$  représente la force de saut aléatoire du lapin lors de sa fuite, où  $U$  est mis à jour comme  $U = 2(1 - a_5)$  et  $a_5$  est un nombre aléatoire compris entre  $[0, 1]$ .

- **siège dur :**

Dans le cas de HHO, le siège dur se produit lorsque,  $a \geq 0.5$  et  $|P| < 0.5$ , et la proie est épuisée et n'a pas assez d'énergie. Les faucons encerclent la proie et exécutent le bond-surprise. La position mise à jour des faucons est affichée dans l'équation (2.6).

$$A(x + 1) = A_{rabbit}(x) - P|\Delta A(x)|. \quad (2.6)$$

- **Siège doux avec plongées rapides progressives :**

Cette sous-section traite des blocages doux avec des plongées rapides et graduelles où la proie a suffisamment d'énergie  $|P| \geq 0,5$  pour échapper à l'attaque, mais où le faucon construit un blocus doux  $a < 0,5$ . Dans cette étape, le faucon doit choisir la meilleure position pour cibler la proie.

1. Effectuer divers mouvements.

2. Analyser et réfléchir à un nouveau mouvement en utilisant équation (2.7).

3. Évaluer le mouvement avec la plongée précédente pour la proie et se rendre compte si le mouvement est favorable ou pas.

$$T = A_{rabbit}(x) - P|UA_{rabbit}(x) - A(x)|. \quad (2.7)$$

4. Si le mouvement n'est pas favorable pour attaquer la proie, la plongée est sélectionnée sur la base d'un vol de prélèvement LF à l'aide de l'équation suivante :

$$L = T + V \times LF(z), \quad (2.8)$$

Où  $Z$  est représenté comme la dimension du problème,  $V$  est considéré comme un vecteur aléatoire de taille  $1 * Z$ , et la fonction LF est calculée à l'aide de l'équation suivante :

$$LF(f) = 0.01 \times \frac{m \times \psi}{|n|}, \quad (2.9)$$

$$\psi = \left( \frac{\xi(1 + \chi) \times \sin(\pi\chi/2)}{\xi(1 + \chi/2) \times \chi \times 2^{(x-1/2)}} \right)^{1/\chi},$$

Où  $m$  et  $n$  sont des valeurs aléatoires comprises entre  $(0, 1)$  et  $\chi$  est une constante fixée à 1,5. L'équation (2.10) est utilisée pour mettre à jour les positions des faucons pendant la phase de siège doux.

$$A(x + 1) = \begin{cases} T & \text{if } F(T) < F(A(x)), \\ L & \text{if } F(L) < F(A(x)), \end{cases} \quad (2.10)$$

Où  $T$  et  $L$  sont acquis à l'aide des équations (2.7) et (2.8), La figure II.3 représente les Vecteurs lors d'un siège doux avec des plongées rapides progressives.

- **siège dur avec des plongées rapides progressives :**

Cette sous-section traite du siège dur avec des plongées rapides et progressives. Quand  $|P| < 0,5$  et  $a < 0,5$  lorsque la proie n'a pas assez d'énergie pour échapper à l'attaque et le faucon construit un siège dur pour attraper et tuer la proie. Dans cette phase, la proie La condition est similaire à celle du siège doux, mais les faucons ont l'intention de minimiser la distance entre leurs emplacements vers une proie en fuite. L'équation explique la dure condition de siège.

$$A(X + 1) = \begin{cases} T & \text{if } F(x) < F(A(x)), \\ L & \text{if } F(x) < F(A(x)), \end{cases} \quad (2.11)$$

Où  $T$  et  $L$  sont dérivés à l'aide des équations suivantes.

$$T = A_{rabbit}(x) - P|UA_{rabbit}(x) - A_P(x)|, \quad (2.12)$$

Et

$$L = T + V \times LF(z). \quad (2.13)$$

Les figures II.2 et II.3 représentent des vecteurs lors d'un siège dur avec plongées rapides progressives dans l'espace 2D et 3D.

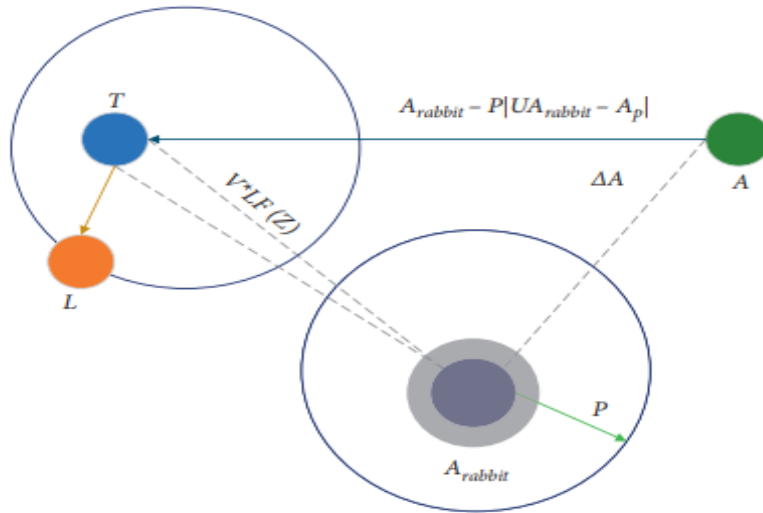


Fig.II.2: Vecteurs lors d'un siège doux avec des plongées rapides progressives en 2D.

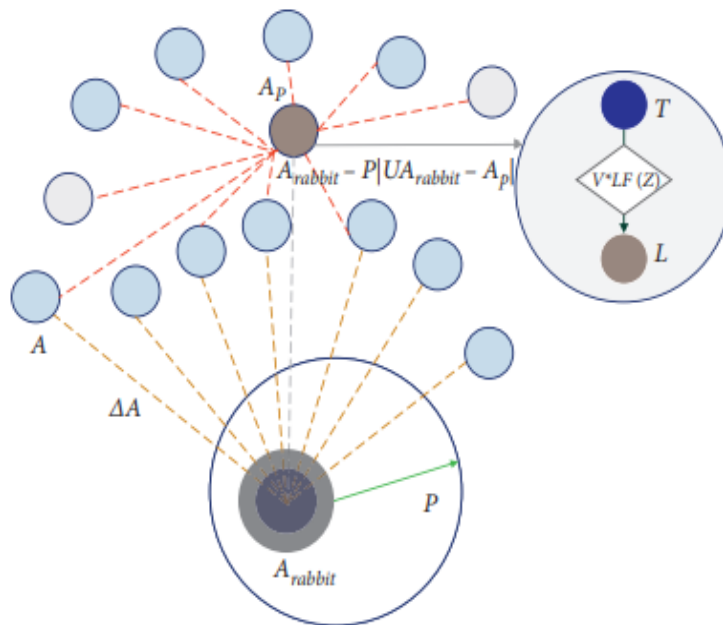


Fig.II.3: Vecteurs lors d'un siège dur avec des plongées rapides progressives en 3D.

## II.4. Mécanique de l'optimisation de Harris Hawks :

### II.4.1. discussion sur le rôle des différents paramètres et leur impact sur les performances de l'algorithme :

Diverses études ont prouvé que l'une des raisons de l'intérêt porté à HHO par rapport à d'autres concurrents est sa performance. Pour plusieurs raisons telles que les composants dynamiques, la sélection gourmande et les règles de recherche en plusieurs étapes, HHO peut afficher des résultats de haute qualité dans les mêmes conditions que celles où d'autres méthodes sont appliquées. C'est l'une des principales raisons pour lesquelles HHO est utilisé dans les papiers suivis. La deuxième raison est la nécessité d'améliorer les performances.

Dans la plupart des variantes, les auteurs affirment que la stabilité de l'optimiseur doit être améliorée pour atteindre des niveaux de convergence plus élevés et moins de problèmes de stagnation. Cette exigence est une nécessité dans toutes les approches basées sur la population pour parvenir à un équilibre plus stable entre les tendances de recherche locales et mondiales. La plupart des études ont renforcé un sentiment d'équilibre entre les tendances exploratoires et exploitatrices des HHO. La précision des résultats et la vitesse de convergence sont les caractéristiques optimisées les plus courantes dans la littérature jusqu'à présent. Ainsi, nous observons que les auteurs ont appliqué HHO et ses variantes à de nombreux nouveaux problèmes et ensembles de données. En conséquence, ils peuvent étudier différents aspects de la performance de l'algorithme HHO face à de nombreux problèmes du monde réel. Enfin, l'objectif est d'optimiser davantage la qualité des résultats de HHO sur une base évolutive plus profonde, telle que la technologie coévolutive, les méthodes multi-populations, les méthodes de conservation et le calcul parallèle. Ces règles peuvent contribuer à harmoniser les tendances globales et locales, ce qui permet d'obtenir de meilleures variantes de HHO. (Alabool, H. M., et al, (2021)).

#### II.4.2. Représentation algorithmique de HHO :

Représentation de l'algorithme qui simplifie la technique HHO :

```

intrants : H et I
sorties : position de la proie et valeur de la condition physique
(i)Ai (i=1,2,...,H)
tant que le critère d'arrêt n'est pas atteint, faire
    calculer la valeur de l'aptitude de l'épervier
    définir Arrabit comme la meilleure position du lapin
    pour chaque faucon Ai faire
        mettre à jour l'énergie p0 et la capacité de saut U
        mettre à jour p à l'aide de l'équation 2.3
    Fin
    si |p| ≥ 1 alors
        mettre à jour le vecteur position avec l'équation 2.1
    Fin
    Si |p| < 1 alors
        Si a ≥ 0.5 et |p| ≥ 0.5 alors
            mettre à jour le vecteur position avec l'équation 2.4
        sinon si a ≥ 0.5 et |p| < 0.5 alors
            mettre à jour le vecteur position avec l'équation 2.6
        sinon si a < 0.5 et |p| ≥ 0.5 alors
            mettre à jour le vecteur position avec l'équation 2.10
        sinon si a < 0.5 et |p| < 0.5 alors
            mettre à jour le vecteur position avec l'équation 2.(11,12,13)
    Fin
Fin
Fin
Retour Arrabit

```

Fig.II.4: Algorithme de Harris Hawks

II.4.3. L'organigramme de HHO :

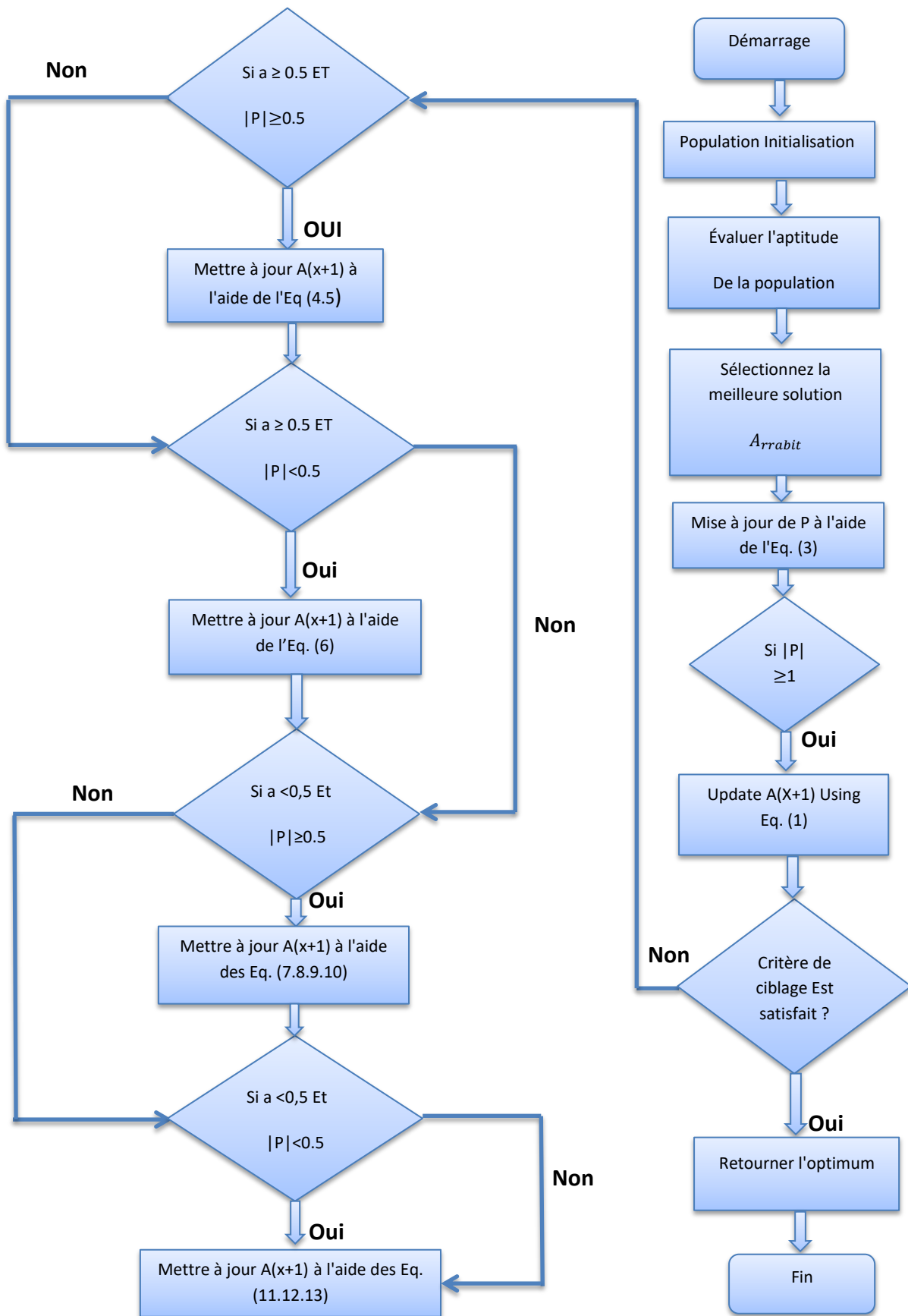


Fig.II.5 : Organigramme de L'algorithme de Harris Hawks

## II.5. Variantes et extension de l'optimisation de Harris Hawks :

### II.5.1. Vue d'ensemble des variantes ou modifications de HHO :

Dans la catégorie de modification du HHO, six études ont proposé différentes versions modifiées de HHO. Par exemple, Devarapalli, R., et al, (2019), ont proposé différentes versions modifiées de HHO, qui visaient à augmenter la diversité de la population, à améliorer le taux de convergence et à contrôler l'équilibre entre les phases d'exploration et d'exploitation de HHO.

Dans le domaine de l'informatique, trois catégories des versions HHO sont prises en compte, comme la binarisation de HHO, HHO hybride et HHO chaotique.

Dans le domaine de la médecine et de la santé publique, il existe deux versions de HHO, un HHO hybride développé par Houssein, E. H., et al, (2020) et HHO chaotique. Cependant, un résumé des principales variables du HHO est mentionné dans les étapes suivantes.

### II.5.2. Applications et variantes du Harris Hawks :

#### II.5.2.1. Optimisation de l'ingénierie :

Dans le domaine de l'optimisation de l'ingénierie, différentes versions de HHO (par exemple, HHO chaotique, HHO hybride, modification de HHO et HHO multi objectif (Alabool, H.M., (2021))).

- **HHO améliorée :**

Les travaux de Fan, Q., et al, (2020) ont proposé un algorithme hybride appelé algorithme de Harris quasi-réfléchi de Hawks (QRHHO) qui comprend HHO et Quantum-Behaved Rao-Blackwellized Particle Swarm Optimization (QRBL). L'algorithme hybride QRHHO vise à aborder l'optimisation globale. L'algorithme proposé vise à améliorer la performance du HHO en termes d'augmentation de la diversité de la population, améliorant le taux de convergence et contrôlant l'équilibre entre les phases d'exploration et d'exploitation.

Pour résoudre des problèmes de conception technique et d'optimisation, les travaux de Kamboj, V. K., et al, (2020) ont proposé un algorithme hybride Harris Hawks sinus-cosinus (hHHO-SCA). Il vise à maintenir et contrôler l'équilibre entre l'exploration et les phases d'exploitation. L'algorithme hybride amélioré comprenait deux algorithmes principaux qui sont HHO et SCA.

Dans les travaux de Dhawale, D., et al, (2020) les auteurs ont introduit le HHO hybride pour le faucon de Harris et le loup gris (hHHO-IGWO). L'algorithme HHO-IGWO vise à améliorer l'équilibre entre les phases d'exploration et d'exploitation pour des problèmes standards en augmentant la diversité de la population.

Pour résoudre des problèmes du monde réel, les travaux de Chen, H., et al, (2020), ont amélioré l'algorithme HHO en combinant Cuckoo Search - Multiple-Path Search (CS, MPS) et d'autres algorithmes. L'algorithme chaotique HHO introduit est appelé Chaos Map and Differential Evolution Harris Hawks Optimisation (CMDHHO). Il vise à maintenir l'équilibre entre les phases d'exploration et d'exploitation en termes d'augmentation de la diversité de la population et d'amélioration du taux de convergence. Les résultats de l'expérience indiquent que le CMDHHO joue un rôle essentiel pour empêcher le HHO de tomber dans l'optimum local, améliorant ainsi les capacités de recherche globales et locales nécessaires de HHO, et il surpasse les autres algorithmes.

Le travail de Gupta, S., et al, (2020), a modifié HHO pour être utilisé à des fins générales de problèmes de conception technique. La version modifiée de HHO appelée m-HHO, qui a été modifiée selon les algorithmes NEP-E, PRD, OBL, GSM. Les résultats globaux ont montré

que m-HHO est significatif dans la diminution du nombre de solutions opposées ; éviter l'OL ; augmenter le taux de convergence ; et surperformance par rapport aux autres algorithmes. Les travaux de Zheng-Ming, G. A. O., et al, (2019) ont proposé l'algorithme chaotique CHHO qui vise à améliorer les performances de HHO. Il combine une stratégie de chaos sur une carte de tente. Dix-huit fonctions de référence ont été utilisées pour valider le nouvel algorithme CHHO. Résultats a constaté que le CHHO proposé peut trouver la solution optimale plus rapidement dans la phase d'exploration.

Dans Les travaux de Qu, C., et al, (2020), pour aborder l'ingénierie d'optimisation contrainte problèmes, algorithme chaotique HHO appelé IEHHO qui la combinaison NEP-E et CS a été mise en œuvre. Les résultats de l'analyse ont révélé que L'algorithme chaotique IEHHO offre une meilleure solution qualité et vitesse de convergence plus rapide, est capable d'améliorer le processus de recherche d'optimisation, est suffisant dans la capacité d'exploration globale, et est capable d'éviter de tomber dans un optimum local.

Le travail d'Ewees, A. A., et al, (2020) a proposé un CMVHHO chaotique algorithme qui vise à améliorer et modifier le multivers méthode d'optimisation dans le contexte de problèmes d'ingénierie. Il combine deux algorithmes principaux : MVO, cartes chaotiques et HHO.

- **Industrie manufacturière :**

Pour résoudre les problèmes de reconnaissance de formes, Golilarz, N. A., et al (2019) a développé Algorithme HHO-ConvNet. Il vise à contrôler le modèle des graphiques pour surveiller le processus de production. L'amélioration l'algorithme hybride est composée de deux algorithmes principaux qui sont HHO et ConvNet.

En termes d'optimisation de la conception technique, les travaux de Shehabeldeen, T. A., et al, (2019) a utilisé l'algorithme HHO pour améliorer les performances d'ANFIS. L'algorithme hybridé a été nommé ANFISHHO. L'objectif principal de l'ANFIS-HHO est de maximiser qualité et résistance du soudage par friction malaxage.

Dans le cadre de l'optimisation de la fabrication, les travaux de Yıldız, A. R., et al, (2019) a proposé H-HHONM, qui hybride HHO algorithme et méthode Nelder-Mead qui vise à optimiser les paramètres de processus dans les opérations de fraisage.

Un hybride de HHO, associé à un recuit simulé, a été présenté Kurtuluş, E., et al, (2020) pour sélectionner les meilleurs paramètres de conception de systèmes de garde-corps routiers, connus sous le nom de HHOSA.

Dans les travaux des auteurs, (Yıldız, A. R., et al, (2019)) algorithme hybride HHO composé d'algorithmes HHO, GOA et MVO est proposé pour résoudre le problème d'optimisation de la fabrication. L'algorithme proposé a été utilisé pour optimiser le processus paramètres pour les opérations de fabrication.

- **Qualité environnementale :**

Dans le contexte de la prévision de la pollution atmosphérique, de la recherche de Du, P., et al, (2020) en a proposé un modèle de prévision de la pollution atmosphérique connu sous le nom de MOHHO, qui hybride les algorithmes ELM et HHO. Le MOHHO proposé vise à surmonter les lacunes de l'optimisation HHO à objectif unique.

Les travaux de Yu, Z., et al, (2020) ont introduit le HHO hybride qui est composé d'algorithmes HHO et RF. HHO-RF vise à optimiser la prédiction des vibrations du sol induites par le souffle.

Pour optimiser la prédiction de la compression du sol coefficient, un HHO hybride avec ANN et GOA des algorithmes ont été suggérés.

Dans le contexte de la qualité de l'environnement, Moayedi, H., et al, (2021), a proposé L'algorithme est connu sous le nom de HHO-MLP qui hybride HHO algorithme et

méthode MLP pour examiner le roulement capacité en situation de problème de classification dans le problème de génie géotechnique.

Pour l'analyse de susceptibilité aux glissements de terrain, il a été proposé algorithme hybride appelé HHO – ANN composé de HHO et ANN sont implémentés.

- **Solaire photovoltaïque :**

Dans le domaine des problèmes d'ingénierie, Moayedi, H., et al(2021), a proposé algorithme hybride HHO avec l'algorithme ANN, nommé comme HHO\_ANN, pour optimiser la précision des prévisions de productivité d'alambic solaire actif.

Pour mesurer les paramètres des modèles de cellules solaires du modèle photovoltaïque à trois diodes, la recherche de Chen, H., et al, (2020), introduit l'algorithme chaotique HHO en le combinant avec mécanisme OBL et stratégie CLS. Le suggéré L'algorithme est connu sous le nom d'EHHO et concerne le monde réel, Problèmes d'ingénierie.

- **Systèmes électriques :**

Algorithme HHO conventionnel, qui vise à optimiser la surcharge harmonique de la composante dépendante de la fréquence, a été présenté par Aleem, S.H.A., et al, (2019). Trois objectifs différents les fonctions du filtre harmonique de type C ont été utilisés pour valider l'algorithme HHO conventionnel.

L'article de Devarapalli, R., et al, (2019), a proposé un algorithme connu sous le nom de LogHHO, qui utilisait la fonction logarithmique pour modifier l'Algorithme HHO.

L'algorithme proposé vise à sélectionner la meilleure allocation des unités DG dans le système de distribution radiale.

Les auteurs Tayab, U. B., et al, (2020) ont introduit l'hybride HHO, qui est obtenu par un algorithme intégré robuste appelé FNN pour HHO. L'algorithme proposé, connu sous le nom de HHO/FNN, vise à réduire l'erreur de prédiction en améliorant les performances de FNN.

- **Autres domaines d'ingénierie :**

Pour la reconnaissance et la prévision des cibles radar à synthèse d'ouverture le marché boursier, la recherche de Hu, H., et al,( 2020) a proposé un nouveau algorithme connu sous le nom d'IHHO-SVM, qui hybride quatre principaux algorithmes, à savoir HHO, ATA, SVM et ELM. L'objectif principal de l'algorithme proposé est d'ajouter de la vitesse dans la phase d'exploration.

### II.5.2.2.L'informatique :

- **Exploration et traitement des données :**

Dans le contexte de la planification de tâches dans le cloud computing, Attiya, I., et al, (2020) a présenté une version modifiée de HHO qui vise à trouver une optimisation de la solution de planification des travaux. Le Un algorithme amélioré nommé HHOSA a été introduit dans améliorer les performances de HHO en améliorant le taux de convergence et en améliorant la recherche locale.

Les travaux de Thaher, T., et al,(2020) ont présenté l'algorithme d'hybridation HHO connu sous le nom de BHHO qui vise à optimiser les caractéristiques tâches de sélection dans le cadre du data mining.

- **Seuil d'image :**

Les problèmes de segmentation d'images ont été représentés en déterminant les seuils optimaux de l'image couleur, résolu en utilisant une version améliorée du HHO algorithme. L'algorithme amélioré comprend le DE algorithme et HHO nommé HHO-DE. Il vise à améliorer les capacités de recherche globales et locales de base préservées dans la phase d'exploration du standard HHO.

Le travail de Rodríguez-Esparza, E., et al, (2020), a présenté la version modifiée de l'Algorithme HHO. L'algorithme modifié nommé MCETHHO.2 Il vise à optimiser la solution pour la segmentation d'image et amélioration des performances de HHO en termes d'amélioration du taux de convergence et de la recherche.

Les travaux de Shahid, M., et al, (2020), ont hybridé l'algorithme HHO en combinant l'algorithme SSBS. L'algorithme amélioré est nommé HHO-SSBS. Il vise à traiter le d'ébruité données d'image pour trouver les paramètres améliorés des fonctions de seuillage. Pour évaluer l'efficacité du HHO-SSBS, trois images satellite ont été utilisées. Aussi, pour mesurer son performances, l'algorithme a été comparé à cinq algorithmes de pointe ; JADE, TNN de Sahraeian, NoorbakhshTNN, Sahraeian-HHO et Noorbakhsh-HHO. Résultats de l'analyse a révélé que l'algorithme HHO effectue beaucoup mieux que les autres algorithmes visuellement et quantitativement.

- **La mise en réseau :**

Pour résoudre un problème d'optimisation combinatoire complexe, a amélioré l'algorithme chaotique HHO en combinant les algorithmes NCPS, ROBLS et HHO. Le L'algorithme proposé connu sous le nom de NOL-HHO vise à estimer la limite inférieure optimale de la séquence d'ADN contrainte et maintenir l'équilibre entre exploration et exploitation phases en termes d'augmentation de la diversité de la population et améliorant le taux de convergence du HHO conventionnel.

Le travail de Li, C., et al, (2020), a présenté l'algorithme chaotique HHO en le combinant avec la stratégie LCSDP. L'algorithme présenté est connu sous le nom de CHHO. Il vise à améliorer la recherche locale de HHO en phase d'exploration afin de répondre au problème de composition du service web QoSaware.

- **Génie logiciel :**

L'algorithme HHO conventionnel a été appliqué pour sélectionner le meilleur cas de test dans le contexte de la minimisation des tests de régression problèmes de dépenses]. Cinq objets différents de SIR ont été utilisés pour valider l'algorithme HHO conventionnel. C'était par rapport à l'algorithme HS pour mesurer ses performances. En termes de couverture des défauts.

Les travaux de Thaher, T., & Arman, N. (2020), ont développé l'hybridation HHO qui est composé d'algorithmes HHO et EB connus comme EBHHO. EBHHO vise à optimiser la solution pour prédire les composants défectueux dans un projet logiciel. Le L'hybridation HHO vise à préserver l'équilibre entre phases d'exploration et d'exploitation, pour accroître la diversité de la population et améliorer le taux de convergence des HHO.

### **II.5.2.3 Médecine et santé publique :**

Les travaux de Houssein, E. H., (2020) ont proposé un SVM et un KNN intégrés algorithme avec HHO hybridé pour sélectionner la meilleure fonctionnalité pour obtenir une précision de classification élevée des produits chimiques sélection des descripteurs et activités des composés chimiques, en termes de problèmes de sélection de descripteurs chimiques de conception de médicaments.

### **II.5.3.Comparaison avec d'autres algorithmes d'optimisation :**

Le choix entre HHO et d'autres algorithmes dépend en grande partie du problème spécifique à résoudre, des caractéristiques du paysage de la solution, des ressources informatiques disponibles et de l'expertise des praticiens. Harris Hawks Optimisation se distingue par son inspiration unique de la nature, son équilibre dynamique entre exploration

et exploitation et son adaptabilité à des problèmes complexes. Cependant, comme tous les algorithmes, il n'est pas universellement supérieur et doit être sélectionné après un examen attentif des exigences du problème et une analyse comparative des performances. (Heidari, A. A., et al (2019)).

HHO : Comme la plupart des métaheuristiques, elle nécessite un réglage de ses paramètres (par exemple, l'énergie de fuite) pour différents problèmes, ce qui peut être considéré à la fois comme une flexibilité et une complexité, selon la perspective de mise en œuvre.

Autres algorithmes : le réglage des paramètres est une exigence commune à tous ces algorithmes pour obtenir des performances optimales. La facilité ou la difficulté du réglage peut affecter la convivialité globale et l'efficacité de l'algorithme pour des applications spécifiques.

HHO : A fait preuve d'une grande efficacité dans la recherche d'optimums globaux avec un bon équilibre entre exploration et exploitation. Il est particulièrement efficace pour les paysages complexes et multimodaux, mais peut nécessiter un réglage des paramètres pour des performances optimales.

GA, PSO, ACO, SA : l'efficacité varie en fonction du problème et de la mise en œuvre. Les GA et les PSO conviennent généralement à un large éventail de problèmes, mais peuvent avoir des difficultés avec les optima locaux. ACO excelle dans les problèmes d'optimisation discrets comme le routage. SA est connu pour sa simplicité et sa capacité à échapper aux optima locaux, mais peut être lent à converger.

## **II.6. Conclusion :**

L'algorithme Harris Hawks d'Optimisation s'appuie sur les stratégies de chasse intelligentes des faucons de Harris pour fournir une méthode d'optimisation efficace et adaptative. Sa combinaison de phases d'exploration et d'exploitation, inspirée des comportements naturels, lui permet de résoudre efficacement des problèmes d'optimisation complexes.

HHO a été appliqué avec succès à divers problèmes d'optimisation, notamment la conception technique, l'apprentissage automatique, la sélection de caractéristiques et l'ordonnancement. Sa capacité à équilibrer l'exploration et l'exploitation en fait un outil d'optimisation polyvalent et efficace.

## Chapitre III

# Applications des métaheuristiques à quelques problèmes d'Hydraulique souterraine

### III.1. Introduction :

Ce chapitre est organisé en 3 parties. Une première partie réservée à la présentation de quelques problèmes d'Hydraulique souterraine pour identifier les caractéristiques hydrodynamiques des aquifères. Nous traiterons le cas d'une nappe captive, le cas d'une nappe libre et le cas d'une nappe semi-captive.

La deuxième partie de ce chapitre sera dédiée à l'application des deux métaheuristiques : l'algorithme du Recuit Simulé et l'algorithme de Harris Hawks aux trois problèmes Hydrauliques et à la validation des codes de calcul pour des cas théoriques.

La troisième partie traitera l'interprétation de quelques essais de pompage de trois cas réels à l'aide des deux métaheuristiques testées et validées.

### III.2. Essais de pompage :

Longtemps considérées comme des ressources inépuisables, les eaux souterraines sont en réalité fragiles, limitées et difficiles à appréhender. Leur disponibilité à chaque endroit du globe est liée à certaines conditions climatiques, géologiques et environnementales. Les eaux souterraines sont largement utilisées dans de nombreuses régions du monde pour des besoins industriels, agricoles et l'approvisionnement en eau potable. Bien que la composition chimique des eaux souterraines puisse varier considérablement, leur qualité est souvent conforme aux normes les plus rigoureuses.

Afin d'évaluer la capacité des aquifères et d'estimer les réserves d'eau souterraine, les essais de pompage représentent le moyen direct pour estimer les caractéristiques hydrodynamiques.

Dans cette section, nous étudions l'évolution des hauteurs piézométriques  $h$  en fonction du temps de pompage au niveau de piézomètres à proximité du puits de pompage. Des solutions analytiques existent pour  $h$  en fonction de la distance par rapport au centre du puits  $r$ , et elles peuvent être calculées à l'aide de nos programmes programmés sous MATLAB.

#### III.2.1 Nappe captive :

Un aquifère captif est une couche d'eau souterraine perméable située entre deux couches imperméables (aquitards), comme le montre la figure III.1. Dans des situations idéales, qui sont traitées dans ce chapitre, les eaux souterraines s'écoulent dans une couche perméable, l'aquifère, de tous les côtés radialement vers un puits de pompage installé. On suppose que la situation est totalement égale dans toutes les directions radiales, ce qui permet d'utiliser le rayon  $r$  comme variable d'espace. On suppose également qu'il n'y a pas de différences dans la direction verticale : le puits est blindé sur l'ensemble de l'aquifère et il

n'y a pas de différences concernant les propriétés hydrauliques à l'intérieur de la couche perméable. L'aquifère reste saturé en eau, c'est-à-dire qu'il n'y a pas de parties qui s'assèchent à cause du pompage.

Dans la situation idéalisée de la figure III.1, l'aquifère est caractérisé par une épaisseur  $H$  [m] et une transmissivité  $T$  [ $m^2/s$ ]. Le paramètre de transmissivité représente la conductivité hydraulique  $K$  du matériau poreux et l'épaisseur de l'aquifère  $H$  :

$$T = K.H \quad (3.1)$$

$T$  augmente avec l'épaisseur ;  $T$  est plus élevé pour les aquifères plus perméables. On suppose que le puits prélève de l'eau à un taux constant  $Q$  [ $m^3/s$ ], ce qui permet de décrire l'écoulement de l'eau souterraine en régime permanent. La variable pertinente pour l'analyse de l'écoulement des eaux souterraines est la hauteur piézométrique  $h$ , qui varie en fonction de la distance  $r$  par rapport à la position du puits. La hauteur piézométrique est la variable clé de l'écoulement, quantifiant la hauteur de la nappe phréatique au-dessus d'un niveau de référence mesuré par un piézomètre. Un piézomètre est un tuyau ouvert aux deux extrémités et qui pénètre dans l'aquifère par l'extrémité inférieure.  $H$  diminue si l'on s'approche du puits et peut être calculée en utilisant la formule de Thiem (1906) :

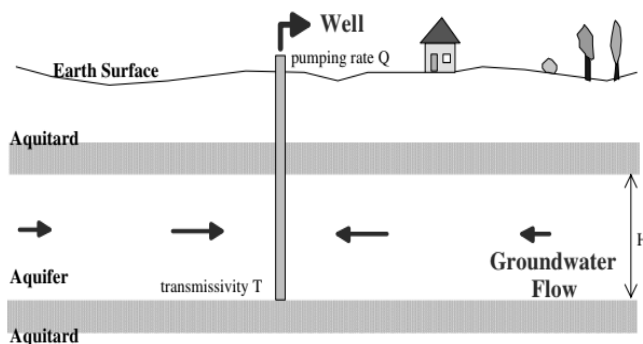


Fig. III.1 : Coupe d'un puits pompant dans un aquifère captif.

$$h(r) = h_0 + \frac{Q}{2\pi T} \log\left(\frac{r}{r_0}\right) \quad (3.2)$$

Avec:  $h_0$  hauteur piézométrique au-dessus de la base au rayon  $r_0$  [m].

$Q$  débit de pompage [ $m^3/s$ ].

$T$  transmissivité de l'aquifère [ $m^2/s$ ].

$r_0$  Rayon [m].

Nous décrivons ici une courte séquence de commandes permettant de calculer les valeurs de la hauteur piézométrique à proximité du puits. Tout d'abord, les valeurs des paramètres d'entrée doivent être données :

```
h0 = 5 ;
T = 5.e-6;
Q = 1.e-4;
r0 = 0.1;
```

Le vecteur de rayon  $r$  est ensuite spécifié. Pour chaque rayon du vecteur, l'abaissement de la tête piézométrique  $h$  est calculé :

```
r = [0.1:0.1:20];
```

La commande suivante lance le calcul des valeurs souhaitées selon la formule (3.2) :

```
h = h0 + (Q/ (2 *pi*T)) *log(r/r0);
```

Dans **MATLAB**,  $\log$  est le logarithme naturel de la base  $e$ . Dans le vecteur  $h$ , nous trouvons maintenant toutes les têtes piézométriques pour les rayons, donnés dans le vecteur rayon  $r$ .

Pour une illustration, il est préférable d'utiliser un graphique qui est produit le plus facilement par la commande :

```
plot (r,h);
ylabel ('hauteur piézométrique [m]'); xlabel ('distance [m]');
```

Le résultat est illustré à la figure III.2

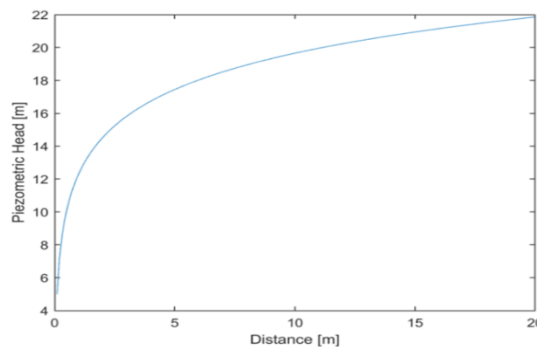


Fig. III.2 : Abaissement de la hauteur piézométrique de la nappe d'eau dans un aquifère captif (confiné) dû au pompage (formule de Thiem).

### III.2.2. Nappe libre :

Contrairement à un aquifère captif, un aquifère libre (également appelé aquifère phréatique) n'est pas limité par une couche imperméable située au-dessus. La limite supérieure d'un aquifère non confiné est donnée par la nappe phréatique. Entre la nappe phréatique et la surface de la terre se trouve la zone non saturée, où l'espace poreux à l'intérieur du matériau poreux est rempli d'eau et d'air.

Dans l'aquifère, seule l'eau circule dans l'espace poreux. La situation est représentée schématiquement par la Fig.III.3. Si elle est mesurée par rapport à la base de l'aquifère, la variable  $h$  est l'épaisseur saturée en eau de l'aquifère, c'est-à-dire la distance entre la position de la nappe phréatique et la base de l'aquifère situé en dessous. Contrairement à l'aquifère captif, dans l'aquifère libre, la hauteur piézométrique  $h$  représente la position de la nappe phréatique. La formule suivante adapte l'équation de Thiem à la situation d'un aquifère non confiné. Elle est dérivée de l'équation (3.2.)Et fournit la hauteur piézométrique  $h$  à la distance  $r$  d'un puits :

$$h^2(r) = h_0^2 + \frac{Q}{\pi K} \log \frac{r}{r_0} \quad (3.3)$$

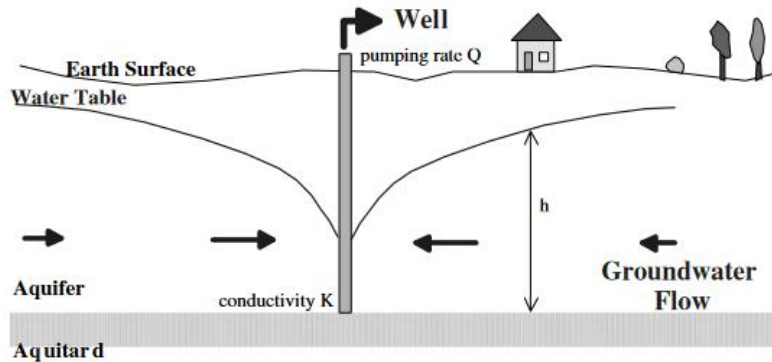


Fig. III.3 : Coupe d'un puits pompant dans un aquifère libre.

Avec:  $h_0$  : niveau de l'eau dans le puits [m].  
 $Q$  : débit de pompage [ $m^3/s$ ].  
 $K$  : conductivités hydraulique de l'aquifère [m/s].  
 $r_0$  : Rayon du puits [m].

Dans l'aquifère confiné, l'écoulement horizontal vers un puits en régime permanent doit satisfaire à l'équation de conservation du volume :

$$2\pi r H v_r = Q \quad (3.4)$$

Pour tous les rayons  $r$  avec une vitesse  $v_r$  dépendant du rayon, une profondeur d'aquifère  $H$  et un taux de pompage  $Q$ . La loi de Darcy s'applique :

$$v_r = K \frac{\partial h}{\partial r} \quad (3.5)$$

Les deux équations combinées donnent une équation différentielle pour  $h(r)$  :

$$r \frac{\partial h}{\partial r} = \frac{Q}{2\pi T} \quad (3.6)$$

Avec  $T = KH$ . Le membre de droite étant une constante, l'équation différentielle peut également s'écrire comme suit :

$$\frac{\partial}{\partial r} \left( r \frac{\partial h}{\partial r} \right) = 0 \quad (3.7)$$

Afin d'obtenir une formule de solution, nous procédons à une reformulation de l'équation :

$$\frac{\partial h}{\partial r} = \frac{Q}{2\pi T} \frac{1}{r} \quad (3.8)$$

La solution peut être obtenue simplement par intégration :

$$h = \frac{Q}{2\pi T} \log(r) + C \quad (3.9)$$

Avec la constante d'intégration C. Si la tête  $h_0$  à une position  $r_0$  est donnée, la constante d'intégration peut être déterminée :

$$C = h_0 - \frac{Q}{2\pi T} \log(r_0) \quad (3.10)$$

La formule (3.2) donnée ci-dessus en résulte.

Dans la situation non confinée, on commence par analogie avec le principe de conservation du volume :  $2\pi H v_r = Q$

Au lieu de la hauteur totale de la couche d'eau souterraine, il faut considérer la hauteur de la nappe phréatique  $h$  au-dessus de la base. L'utilisation de la loi de Darcy, comme indiqué ci-dessus, donne :

$$r h \frac{\partial h}{\partial r} = \frac{Q}{2\pi K} \quad (3.11)$$

Où

$$r \frac{\partial h^2}{\partial r} = \frac{Q}{\pi K} \quad (3.12)$$

Avec la solution générale :

$$h^2 = \frac{Q}{\pi K} \log(r) + C \quad (3.13)$$

Comme ci-dessus, la connaissance d'une paire  $(r_0, h_0)$  permet de déterminer la constante d'intégration C et la formule (3.3) en résulte.

$h$  comme change avec le rayon  $r$ , comme expliqué ci-dessus. Nous calculons les valeurs changeantes dans une courte séquence de commandes. Dans la fenêtre de commande MATLAB, on doit spécifier d'abord le nouveau paramètre d'entrée.

```
K = 1.e-4;
```

La ligne suivante lance le calcul du vecteur des hauteurs piézométriques :

```
h = sqrt(h0*h0 + (Q/(pi*K))*log(r/r0)); \quad (3.14)
```

Sqrt désigne la racine carrée, avec la commande suivante, les résultats sont affichés sous la forme d'une ligne brisée verte :

```
plot (r, h, '--g');
```

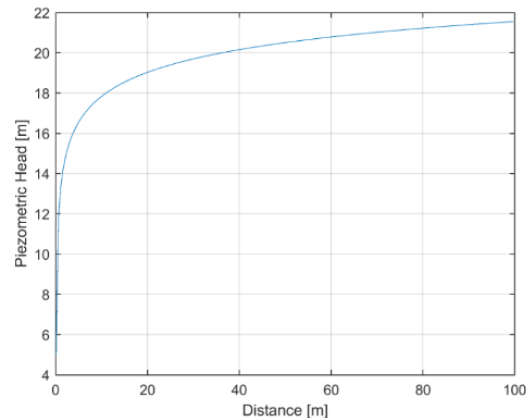


Fig. III. 4 : Abaissement de la hauteur piézométrique de la nappe dans un aquifère libre

### III.2.3. Nappe semi-captive :

La situation d'un aquifère semi-confiné est illustrée à la figure III.5 un aquifère est recouvert d'une couche semi-perméable. Ainsi, l'eau pompée provient en partie de l'aquifère lui-même, et en partie des couches sus-jacentes, qui sont reliées par la couche semi-perméable.

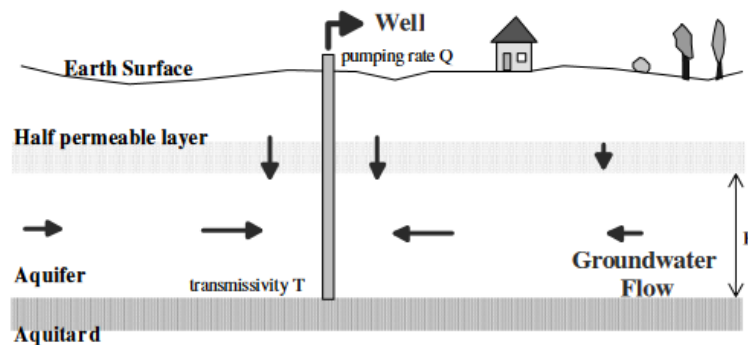


Fig. III. 5: Vue schématique en coupe d'un puits pompant dans un aquifère semi-confiné aquifère.

Pour un aquifère épais semi-captif, De Glee (1930) a dérivé une formule décrivant le rabattement  $s(r)$  de la hauteur piézométrique à une distance  $r$  d'un puits :

$$s(r) = \frac{Q}{2\pi T} k_0 \left( \frac{r}{\sqrt{TC}} \right) \quad (3.15)$$

Avec :  $Q$  débit de pompage [ $m^3/s$ ].  
 $T$  transmissivité [ $m^2/s$ ].  
 $C$  résistance de la couche semi-perméable [ $s$ ].  
 $k_0$  Fonction de Bessel modifiée.

En introduisant le nouveau paramètre  $c$  :

$$c = 1.e7;$$

On calcul ainsi le vecteur des rabattements (drawdowns) en fonction de :

$$s = (Q / (2 * \pi * T)) * \text{besselk}(0, r / \sqrt{T * c}); \quad (3.16)$$

Le résultat est affiché à l'aide de la commande plot :

```
plot (r, -s);
```

La figure III.6 représente la sortie graphique. Dans La figure III.7 les rabattements d'un aquifère captif, d'un aquifère libre et d'un aquifère semi-captif sont comparés. Le rabattement pour la situation de semi-captive se situe entre le rabattement pour l'aquifère confiné et l'aquifère non confiné. L'utilisateur peut facilement trouver des valeurs de paramètres pour lesquelles ce résultat raisonnable n'est pas vrai. La raison de cette incompatibilité apparente est que les trois formules sont valables dans des conditions différentes. La formule de de Glee est dérivée pour le demi-espace situé sous la couche semi- perméable, c'est-à-dire en supposant que l'aquifère est trop étendu, ce qui rend la valeur de son épaisseur non pertinente.

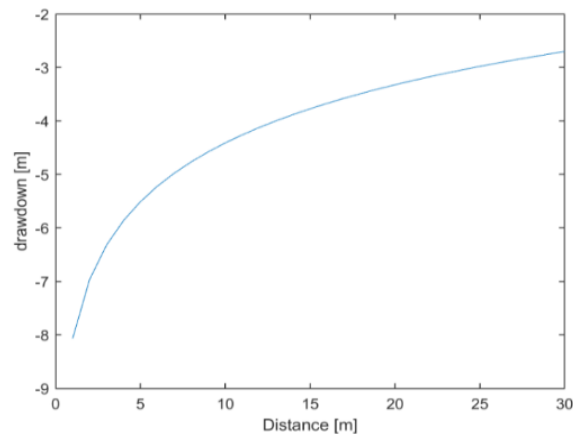


Fig. III.6 : Abaissement de la hauteur piézométrique de l'eau souterraine dans un aquifère semi- captif dû au pompage, d'après de Glee (1930).

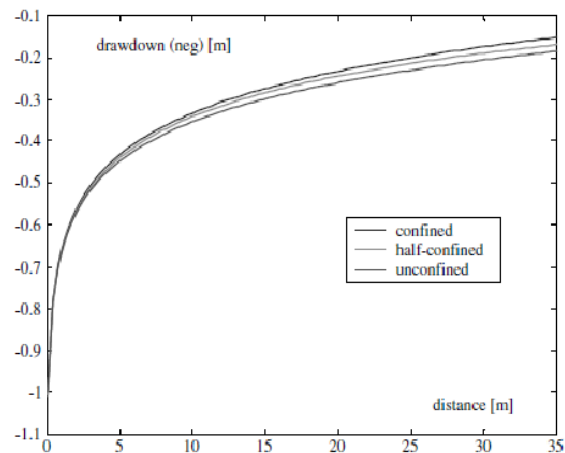


Fig. III.7 : Abaissement régulier de la hauteur piézométrique des eaux souterraines dans un aquifère confiné, un aquifère semi- captif et un aquifère non confiné.

### III.2.4. Essais de Pompage en régime transitoire:

Dans un aquifère captif, le rabattement de la hauteur piézométrique  $s$  est donné par la formule de Theis (1983).  $S$  est fonction de la distance du puits  $r$  et du temps  $t$  après le début du pompage :

$$s(r) = \frac{Q}{4\pi T} W\left(\frac{Sr^2}{4Tt}\right) \quad (3.17)$$

Avec:  $Q$  débit de pompage [ $m^3/s$ ].

$T$  transmissivité de l'aquifère [ $m^2/s$ ].

$S$  coefficient d'emmagasinement de l'aquifère

Dans la formule apparaît la fonction  $W$ , avec un argument qui est généralement abrégé en  $u$ .  $W(u)$  est si important pour le rabattement du puits qu'il est appelé fonction du puits. Dans la littérature mathématique, la même fonction est appelée intégrale exponentielle et est généralement désignée par  $E_1(u)$ . Ce nom s'explique par la définition de  $E_1(u)$  :

$$W(u) = E_1(u) = \int_u^{\infty} \frac{\exp(-\zeta)}{\zeta} d\zeta \quad (3.18)$$

La figure III.8 Réprésente les rabattements obtenus à quatre moments différents :

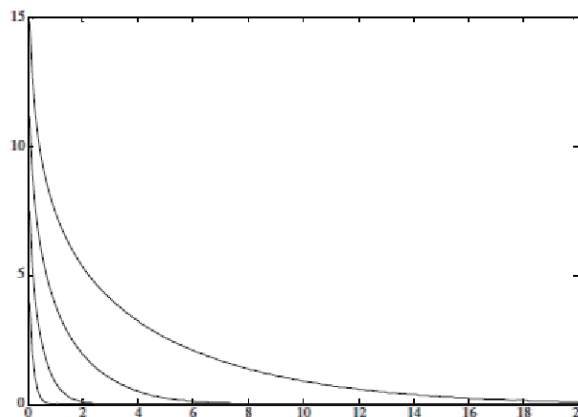


Fig. III.8 : Abaissement instable de la hauteur piézométrique de la nappe dans un aquifère captif dû au pompage selon Theis ; paramètres donnés dans le texte, pour les temps  $t = 10^3, 10^4, 10^5, 10^6$

Sous **MATLAB**, la fonction de puits peut être trouvée sous sa notation mathématique. Elle est appelée par **expint**. Pour calculer la formule donnée ci-dessus, on doit spécifier le paramètre de stockage (Coefficient d'emmagasinement) comme nouveau paramètre :

```
S = 0.1;
```

Spécifiez ensuite la durée pour laquelle le rabattement doit être calculé :

```
t = [1000000];
```

Calculez et tracez les résultats à l'aide des commandes suivantes :

$$s = (Q / (4 * \pi * T)) * \text{expint}(S * r * r / (4 * T * t)); \quad (3.19)$$

```
plot (r, -s);
```

La figure III.9 représente la variation du rabattement croissant en fonction de la distance :

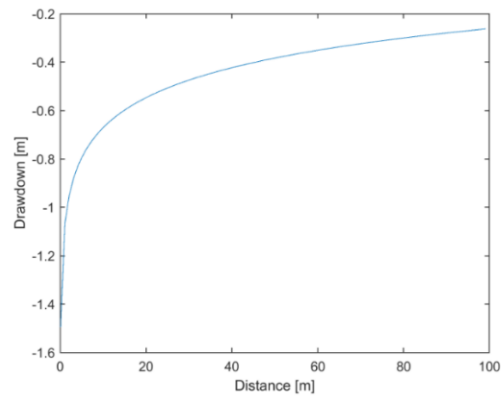


Fig. III.9 : Rabattement du niveau piézométrique dans un aquifère captif dû au pompage, d'après la formule de Theis.

### III.3. Tests et validations des codes :

#### III.3.1 Problème 1 :

Le Problème 1 correspond à l'interprétation d'un essai de pompage fictif pour une nappe captive selon le modèle de Theis. Pour ce problème on considère, une nappe captive de Transmissivité  $T= 0.020 \text{ m}^2/\text{s}$  et un coefficient d'emmagasinement  $S=0.1$ . On suppose une durée de pompage d'une heure (60 mn) et les mesures des rabattements au niveau du puits de pompage.

##### a) – Optimisation par Recuit Simulé :

Après avoir défini la fonction **Objectif** qui correspond à minimiser les écarts entre les rabattements calculés et les rabattements observés selon la relation ci-dessous :

$$\text{RMSE} = \text{sqrt}(\text{mean}((s_{\text{cal}} - s_{\text{obs}})^2));$$

Nous avons imposé les conditions indiquées dans le programme principal (Main Programm) suivant :

```

%*****Ob
jectiveFunction = @simple_objective5;
x0 = [0.001 0.01]; % Starting point
startingPoint= [0.001 0.01];

[x,fval,exitFlag,output] = simulannealbnd(ObjectiveFunction,x0);

lb = [0.0001 0.001];
ub = [0.05 0.3];

[x,fval,exitFlag,output]= simulannealbnd(ObjectiveFunction,x0,lb,ub);

fprintf('The number of iterations was : %d\n', output.iterations);

fprintf('The number of function evaluations was : %d\n',
output.funccount);

fprintf('The best function value found was : %g\n', fval);
%*****

```

Après exécution du code de calcul, et après 2810 itérations et 2831 fonctions d'évaluation le meilleur résultat obtenu pour la fonction Objectif est de 0.000144302. La transmissivité T calculée est de 0.0200 et le coefficient d'emmagasinement S est de 0.102. Ces résultats sont très satisfaisants, les figures III.10 et III.11 illustrent ces résultats.

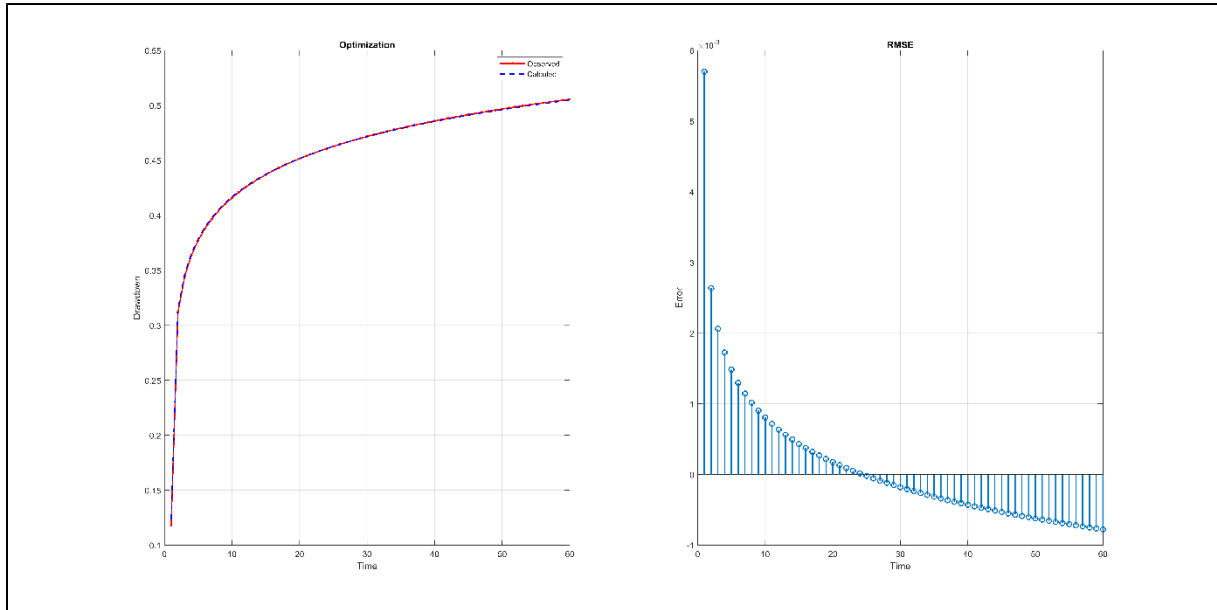


Fig.III.10 : (a) - Comparaison des courbes de rabattements en fonction du temps entre les valeurs observées et calculées après optimisation l'aide du Recuit Simulé.  
 (b) – Erreur relative entre les valeurs observées et calculées des rabattements

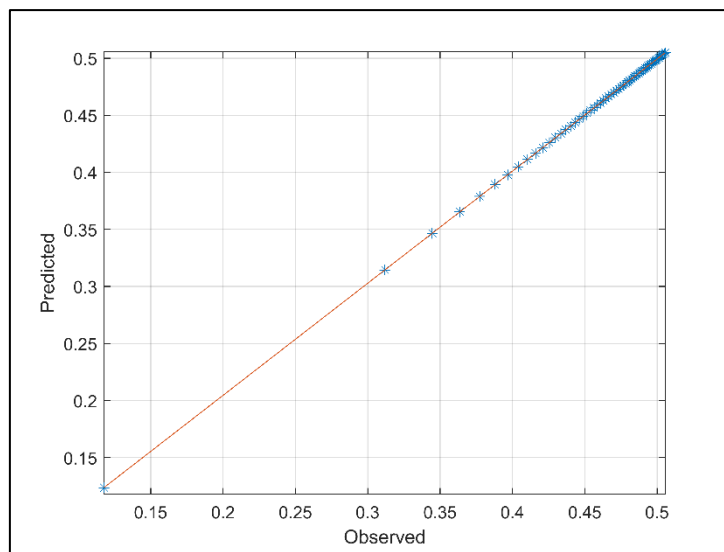


Fig.III.11 : Droite de corrélation entre les valeurs observées et calculées des rabattements

Cependant, pour les mêmes conditions on peut avoir d'autres résultats très proches mais légèrement différents. Le tableau III.1 donne les résultats de dix essais.

N <sup>o</sup> Essai	1	2	3	4	5	6	7	8	9	10
T	0.019	0.020	0.019	0.019	0.0201	0.0200	0.197	0.0200	0.0201	0.0202
S	0.106	0.085	0.119	0.149	0.0978	0.102	0.1146	0.108	0.0991	0.0902
RMSE	0.003	0.001	0.001	0.002	0.00015	0.00014	0.00095	0.00100	0.00107	0.000697
Temps	2.495	2.522	2.367	2.657	4.69016	5.88637	3.10507	2.47092	2.34671	2.74608

Tableau III.1 : Résultats obtenus pour différents essais correspondants aux Transmissivités T, aux coefficients d'emmagasinement S, à la racine carrée de de l'erreur quadratique moyenne (RMSE) et au temps de calcul en seconde.

La figure III.12 présente les résultats obtenus pour 100 essais des valeurs des transmissivités calculées à l'aide du Recuit Simulé. Les valeurs sont très proches et très significatives. De même pour les valeurs des coefficients d'emmagasinement, la figure III.13 présente les valeurs calculées pour les 100 essais.

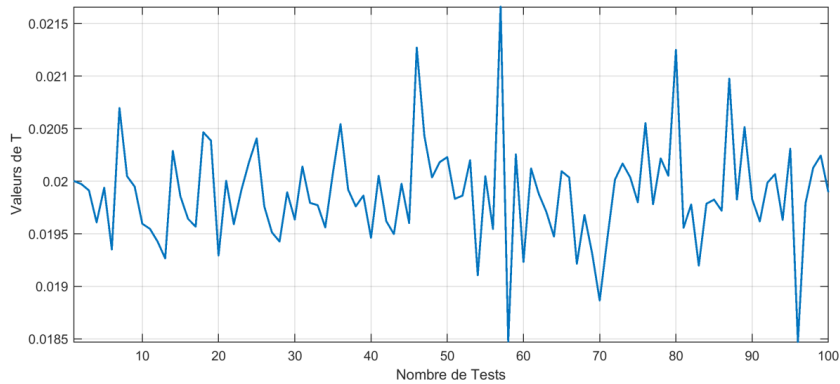


Fig. III.12 : Valeurs des transmissivités obtenues pour 100 essais

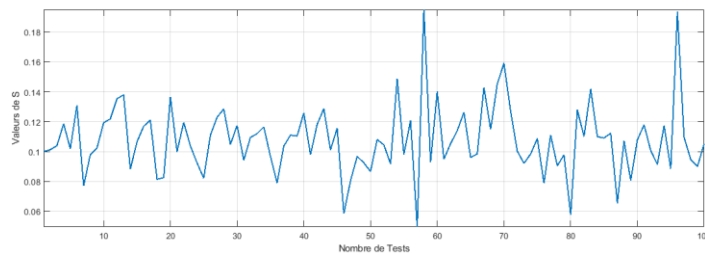


Fig.III.13: Valeurs des coefficients d'emmagasinement obtenues pour 100 essais

Les figures III.14 et III.15 montrent les moyennes mobiles des transmissivités et des coefficients d'emmagasinement pour les 100 essais, elles convergent vers les meilleures valeurs estimées. Le tableau III.2 donne les valeurs de la moyenne mobile des transmissivités et des coefficients d'emmagasinement pour 10, 50, 100 et 200 essais. Les valeurs obtenues après 200 essais sont très satisfaisantes.

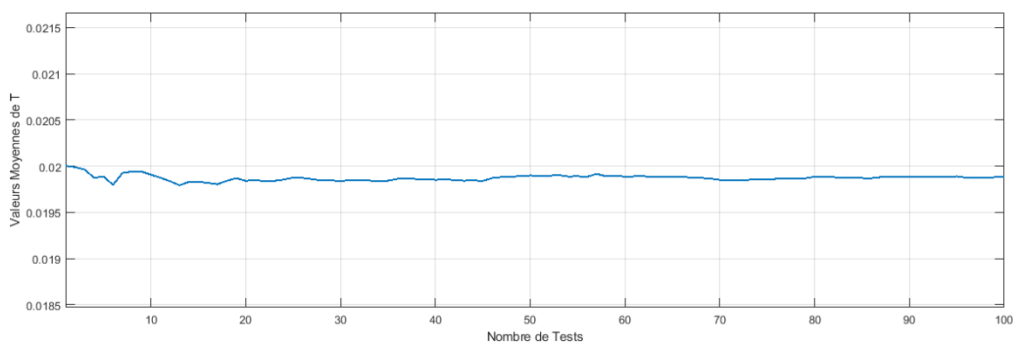


Fig. III.14 : Moyenne Mobile de la valeur de la transmissivité après 100 essais.

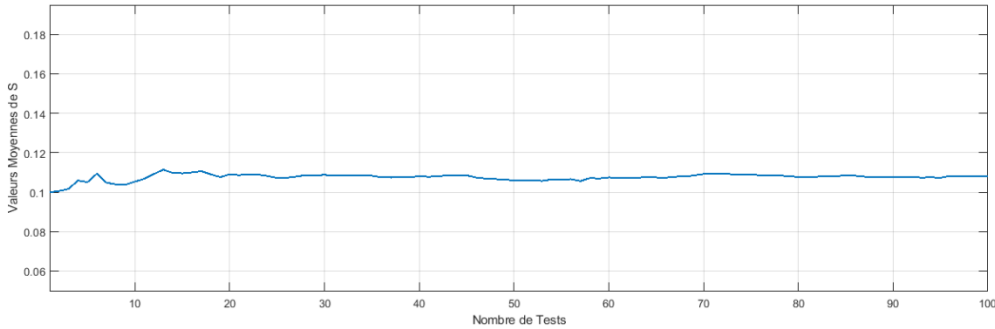


Fig. III.15: Moyenne Mobile de la valeur du coefficient d’emmagasinement après 100 essais.

Essais	10	50	100	200
T Moyenne	0.0199	0.0199	0.0199	0.200
S Moyen	0.1063	0.1074413	0.1078	0.1041

Tableau III.2 : Valeurs des Moyennes Mobiles de la Transmissivité et du Coefficient d’emmagasinement pour différents essais.

Cependant, on peut ajouter plusieurs options au programme principal parmi lesquelles :

- La visualisation des performances ;
- La spécification des options de Température ;
- La reproduction des résultats ;
- Les critères d’arrêt.

- **Visualisation des performances** : Le solveur **simulannealbnd** peut accepter une ou plusieurs fonctions en 'options'. La fonction **optimoptions** par exemple est utile pour visualiser les performances du solveur au moment de l'exécution.

Pour sélectionner plusieurs fonctions présentées graphiquement, on doit définir l'option **PlotFcn** via la fonction **optimoptions**. En sélectionnant par exemple **saplotbestf**, on représentera la meilleure valeur de la fonction **Objectif** à chaque itération, **saplottemperature** représente la température actuelle dans chaque dimension à chaque itération, **saplotf** représente la valeur actuelle de la fonction (qui n'est pas nécessairement la meilleure), et **saplotstopping** représente le pourcentage de critères d'arrêt satisfaisant toutes les dix itérations.

En introduisant à titre d'exemple dans le programme principal la séquence suivante :

```

%*****
options = optimoptions(@simulannealbnd, ...
'PlotFcn',{@saplotbestf,@saplottemperature,@saplotf,@saplotstopping}) ;
simulannealbnd(ObjectiveFunction,startingPoint,lb,ub,options) ;
%*****
    
```

On obtient après exécution du code les résultats suivants :

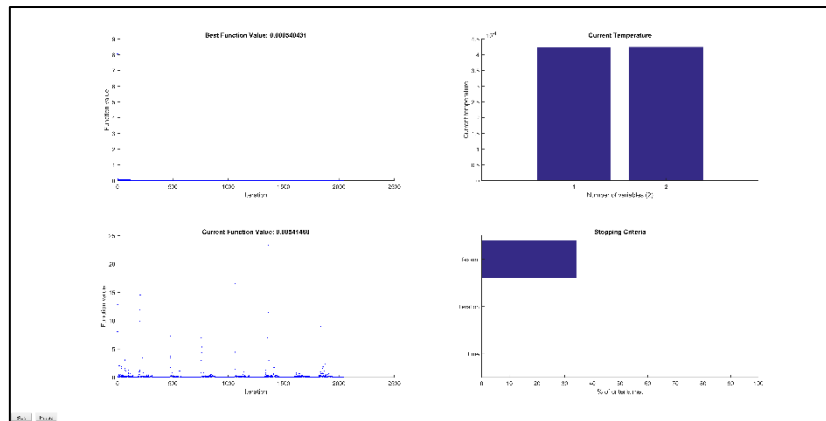


Fig.III.16 : Visualisation des paramètres de performance au moment de l'exécution du code (a-Convergence, b-Températures des variables, c-Fonction d'évaluation, d-Critère d'arrêt.

- Spécification des options de température** : Le paramètre de température utilisé dans le recuit simulé contrôle les résultats globaux de la recherche. La température de chaque dimension est utilisée pour limiter l'étendue de la recherche dans cette dimension. La boîte à outils vous permet de spécifier la température initiale ainsi que les moyens de mettre à jour la température au cours du processus de solution. Les deux options relatives à la température sont **InitialTemperature** et **TemperatureFcn**.

- Spécification de la température initiale** : La température initiale par défaut est fixée à 100 pour chaque dimension (paramètres à optimiser). Si on souhaite que la température initiale soit différente selon les dimensions, on doit spécifier un vecteur de températures. Cela peut s'avérer nécessaire lorsque le problème est mis à l'échelle différemment dans chaque dimension. Par exemple :

```
options = optimoptions(@simulannealbnd,'InitialTemperature',
[300 50]) ;
```

InitialTemperature peut être défini comme un vecteur de longueur inférieure au nombre de variables (dimension) ; le solveur étend le vecteur aux dimensions restantes en prenant le dernier élément du vecteur de température initiale. Dans cet exemple, on souhaite que la température initiale soit la même dans toutes les dimensions, on a donc besoin de spécifier qu'une seule température.

```
options.InitialTemperature = 100 ;
```

- Spécification d'une fonction de température** : La fonction de température par défaut utilisée par **simulannealbnd** est appelée **temperatureexp**. Dans le schéma **temperatureexp**, la température à un pas donné est 0,95 fois la température au pas précédent. Cela permet à la température de baisser lentement au début, mais de

se refroidir plus rapidement que les autres schémas. Si un autre schéma est souhaité, par exemple le schéma de **Boltzmann** ou le schéma de recuit "*rapide*", alors **temperatureboltz** ou **temperaturefast** peuvent être utilisés respectivement. Pour sélectionner le schéma de température rapide, on peut mettre à jour nos options créées précédemment, en modifiant directement **TemperatureFcn**.

```
options.TemperatureFcn = @temperaturefast ;
```

- **Spécification du Recuit** : Le recuit fait partie du processus de recuit. Après l'acceptation d'un certain nombre de nouveaux points, la température est augmentée à une valeur plus élevée dans l'espoir de relancer la recherche et de sortir d'un minima local. Effectuer un recuit trop tôt peut ne pas aider le solveur à identifier un minimum, c'est pourquoi un intervalle relativement élevé doit être choisi. L'intervalle auquel le recalage a lieu peut être défini à l'aide de l'option **ReannealInterval**. Ici, on réduit l'intervalle de recalage par défaut à 50 car la fonction semble être plate dans de nombreuses régions et le solveur pourrait se bloquer rapidement.

```
options.ReannealInterval = 50 ;
```

- **Reproduction des résultats** : **simulannealbnd** est un algorithme non déterministe. Cela signifie que l'exécution du solveur plus d'une fois sans changer aucun paramètre peut donner des résultats différents. Ceci est dû au fait que **simulannealbnd** utilise des générateurs de nombres aléatoires MATLAB® lorsqu'il génère des points ultérieurs et lorsqu'il détermine s'il faut ou non accepter de nouveaux points. Chaque fois qu'un nombre aléatoire est généré, l'état des générateurs de nombres aléatoires change. Pour s'en convaincre, deux exécutions du solveur de **simulannealbnd** donnent des résultats :

```
[x, fval]=simulannealbnd(ObjectiveFunction, startingPoint, lb, ub, options);
```

**simulannealbnd** s'arrête car la variation de la valeur de la meilleure fonction est inférieure à **options.FunctionTol**

On peut reproduire nos résultats si nous réinitialisons les états des générateurs de nombres aléatoires entre les exécutions du solveur en utilisant les informations renvoyées par **simulannealbnd**. **simulannealbnd** renvoie les états des générateurs de nombres aléatoires au moment où **simulannealbnd** est appelé dans l'argument de sortie. Ces informations peuvent être utilisées pour réinitialiser les états. Ici, dans ce cas nous réinitialisons les états entre les exécutions en utilisant ces informations de sortie afin que les résultats des deux prochaines exécutions soient identiques.

```
[x, fval, exitFlag, output]=simulannealbnd(ObjectiveFunction, startingPoint, lb, ub, options);
```

Dans ce cas, **simulannealbnd** s'arrête car la variation de la valeur de la meilleure fonction est inférieure à **options.FunctionTol**.

Nous réinitialisons l'état du générateur de nombres aléatoires :

```
strm = RandStream.getGlobalStream ;
strm.State = output.rngstate.State ;
```

L'exécution à nouveau de **simulannealbnd** par la séquence suivante :

```
[x, fval] = simulannealbnd(ObjectiveFunction, startingPoint, lb, ub, options) ;
```

Dans ce cas, **simulannealbnd** s'est arrêté parce que le changement de la valeur de la meilleure fonction est inférieur à **options.FunctionTol**

- **Modification des critères d'arrêt : simulannealbnd** utilise six critères différents pour déterminer quand arrêter le solveur. **simulannealbnd** s'arrête lorsque le nombre maximum d'itérations ou d'évaluations de fonctions est dépassé. Le nombre maximum d'itérations est fixé à **Inf** et le nombre maximum d'évaluations de fonctions est de  $3000 \times \text{nombre De Variables}$ . **simulannealbnd** garde la trace du changement moyen de la valeur de la fonction pour les itérations **MaxStallIterations**. Si le changement moyen est inférieur à la tolérance de la fonction **FunctionTolerance**, l'algorithme s'arrête. Le solveur s'arrête également lorsque la valeur de la fonction objective atteint **ObjectiveLimit**. Enfin, le solveur s'arrête après avoir fonctionné pendant **MaxTime** secondes. Dans l'exemple, nous avons fixé la **FunctionTolerance** à  $10^{-5}$ .

#### b) – Optimisation de Harris Hawks :

Ce code de calcul est établi par Ali Asghar Heidari et ses collaborateurs du département des sciences informatiques du School of Computing, National University of Singapore, le 03 Mars 2019 (Heidari, A. et al, 2019) a été exploité dans cette étude. Quelques modifications ont été introduites pour adapter le code aux problèmes hydrauliques. La première partie du code de calcul regroupant les principaux paramètres est donnée ci-dessous.

```

%% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
tic
clear all
close all
clc

N=50; % Number of search agents

Function_name='F1'; % Name of the test function
%pause;
T=100; % Maximum number of iterations

% Load details of the selected benchmark function
[lb,ub,dim,fobj]=Get5_Functions_details(Function_name);
%pause;
[Rabbit_Energy,Rabbit_Location,CNVG]=HHO(N,T,lb,ub,dim,fobj);
%pause;

%Draw objective space
figure,
hold on
semilogy(CNVG,'Color','b','LineWidth',1.5);
title('Convergence curve')
xlabel('Iteration');
ylabel('Best fitness obtained so far');
axis tight
grid off
box on
legend('HHO')
grid on;

display(['The best location of HHO is: ', num2str(Rabbit_Location)]);
display(['The best fitness of HHO is: ', num2str(Rabbit_Energy)]);
toc
%*****

```

Ce code a été testé dans un premier temps sur le Problème 1 pour évaluer l'efficacité de la méthode de Harris Hawks. Ainsi, pour un nombre d'agents de recherche  $N=50$  et un nombre d'itérations  $T=100$ , les résultats obtenus sont présentés par les figures III.17, III.18, III.19 et III.20 et regroupés dans le Tableau III.3. La convergence après 100 itérations est très satisfaisante. La figure III.18 montre une presque parfaite restitution des rabattements. L'erreur relative reste faible de quelques millimètres seulement. La droite de corrélation montre une parfaite corrélation entre les rabattements observés et calculés. Cependant, le résultat est encore meilleur pour un nombre d'itérations supérieur et un nombre d'agents de recherche plus grand (Tableau III.3), mais ce résultat est au détriment du temps de calcul, qui augmente avec le nombre d'itérations.

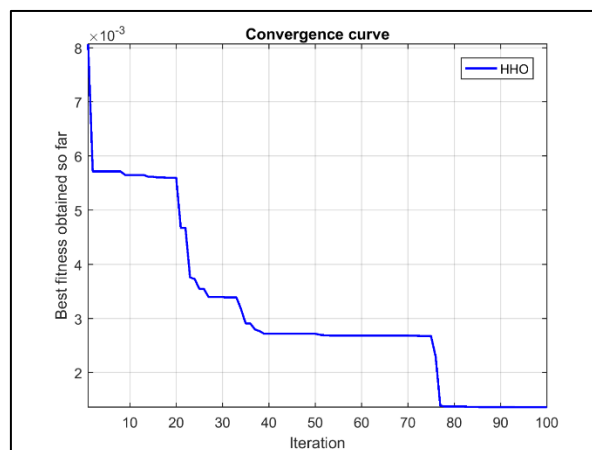


Fig. III.17: Courbe de convergence de la fonction Objectif en fonction du nombre d'itérations

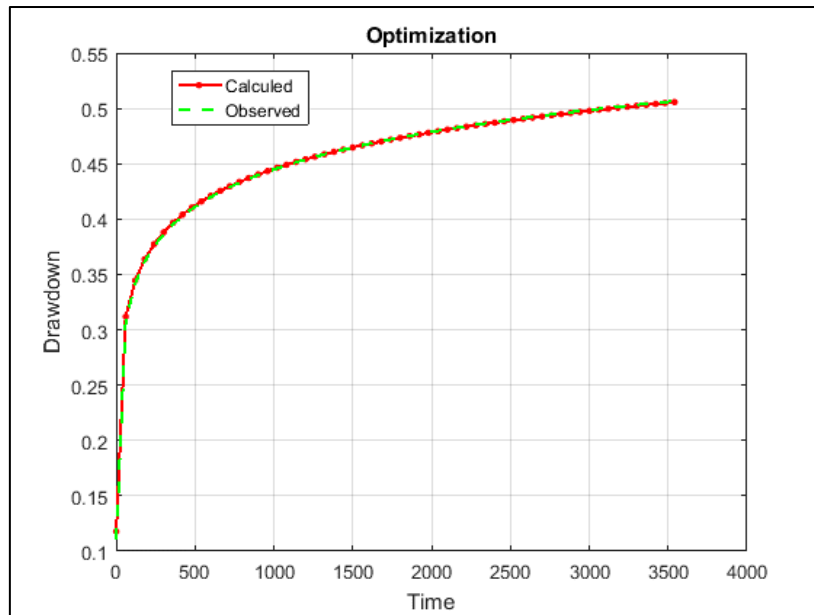


Fig.III.18 : Comparaison des courbes de rabattements en fonction du temps entre les valeurs observées et calculées après optimisation l'aide de Harris Hawks

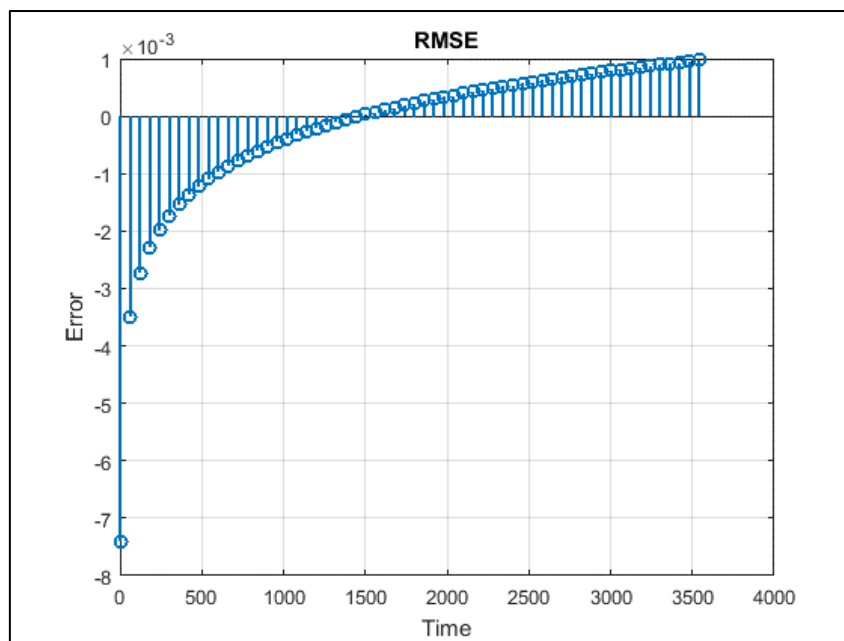


Fig.III.19 : Erreur relative entre les valeurs observées et calculées des rabattements

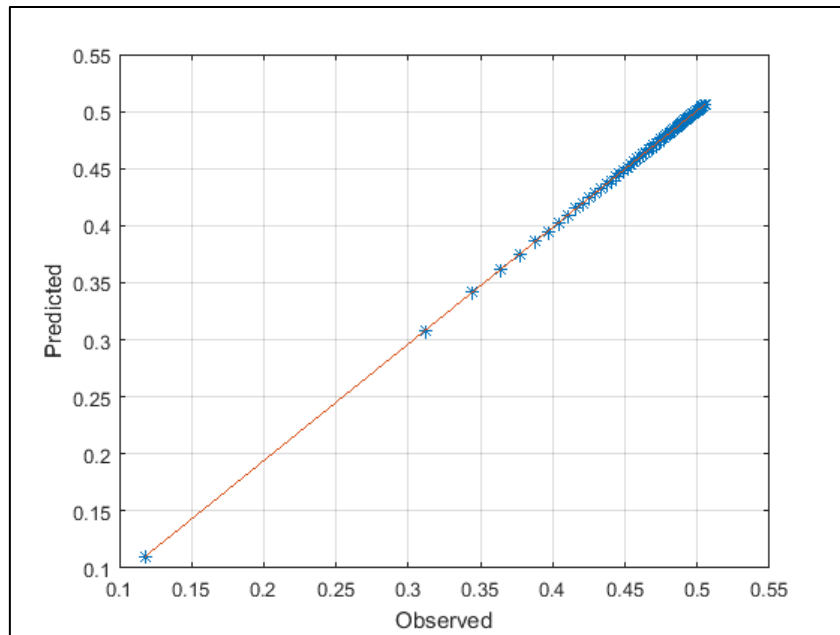


Fig. III.20: Droite de corrélation entre les valeurs observées et calculées des rabattements

Essais	1	2	3
N <sup>b</sup> itération	100	500	800
N	50	500	900
T	0.019549	0.020186	0.02000
S	0.12168	0.092217	0.10008
RMSE	0.0013645	5.4763 10 <sup>-4</sup>	5.6381 10 <sup>-6</sup>
Temps de calcul (s)	12.3013	616.9259	1809.9861

Tableau III.3 : Estimation de la transmissivité et du coefficient d'emménagement par l'optimisation Harris Hawks pour trois essais différents.

### III.3.2 Problème 2 :

Le Problème 2 correspond à l'interprétation d'un essai de pompage fictif réalisé dans une nappe semi-captive. Le pompage est réalisé dans le forage de captage et les observations des rabattements au niveau de 10 piézomètres distants respectivement de 5, 10, 20, 30, 40, 50, 60, 80 et 100 m. Le débit de pompage est de 0.015 m<sup>3</sup>/s et la transmissivité est supposée de 0.050 m/s<sup>2</sup>. Pour cet exemple, on a considéré la résistance de la couche semi-perméable C égale à 10<sup>4</sup> s comme indiqué dans les équations 3.15 et 3.16.

a) – **Optimisation par Recuit Simulé** : Pour le recuit simulé, ces données ont été mentionnées dans la fonction **Objectif** suivante comme suit :

```

%*****
function [RMSEs]=simple_objective7(x)
%*****
c = 10000;
%tf = 1:60:3600;
Q=0.015;
T=0.050;
r=[5 10 15 20 30 40 50 60 80 100];
s = (Q/(2*pi*T))*besselk(0,r/sqrt(T*c));
DeltaObserv=s;
%*****
    
```

Dix essais ont été effectués pour estimer la transmissivité et la résistance de la couche semi-perméable. Ces informations ont été insérées dans le programme principal et dans la fonction RMSEs comme indiqué ci-dessous :

```

%*****
x1 = x(1);
x2 = x(2);
Q=0.015;
r=[5 10 15 20 30 40 50 60 80 100];
ss = (Q/(2*pi*x1))*besselk(0,r/sqrt(x1*x2));
DeltaPred=ss;
RMSEs = sqrt(mean((DeltaPred-DeltaObserv).^2));
end
%*****
    
```

Tableau III.4 ci-dessous regroupe les résultats des 10 essais effectués. Pour des paramètres hydrodynamiques, les résultats sont très satisfaisants et très encourageants, l'erreur est suffisamment faible et le temps de calcul est très court. A titre d'exemple, pour l'essai N° 7 montre le meilleur résultat obtenu, le nombre d'itération était de 1008, le nombre de fonction d'évaluation était de 1021, l'erreur était de  $1.18295 \cdot 10^{-5}$  et le temps de calcul était uniquement de 0.75 seconde. La figure III.21 illustre ce résultat, elle montre une parfaite restitution des courbes de rabattements

Essais	1	2	3	4	5	6	7	8	9	10
T	0.0502	0.0499	0.0500	0.498	0.0500	0.0499	<b>0.05000</b>	0.0498	0.0500	0.0501
C	$1.01 \cdot 10^4$	$9.927 \cdot 10^3$	$1.002 \cdot 10^4$	$9.9636 \cdot 10^3$	$9.9888 \cdot 10^3$	$9.9135 \cdot 10^3$	<b><math>1.007 \cdot 10^4</math></b>	$9.9797 \cdot 10^3$	$1.009 \cdot 10^4$	$1.004 \cdot 10^4$
RMSE	$1.01 \cdot 10^{-4}$	$6.66 \cdot 10^{-5}$	$3.135 \cdot 10^{-5}$	$4.722 \cdot 10^{-5}$	$1.012 \cdot 10^{-5}$	$7.318 \cdot 10^{-5}$	<b><math>1.182 \cdot 10^{-5}</math></b>	$6.948 \cdot 10^{-5}$	$1.262 \cdot 10^{-4}$	$3.17 \cdot 10^{-5}$
Temps	0.8903	0.8466	0.7806	0.7709	0.9072	0.7634	0.7582	0.7764	0.7244	0.7601

Tableau III.4 : Résultats obtenus à l'aide du Recuit Simulé pour différents essais correspondants aux Transmissivités T, aux Résistance de la couche semi-perméable C, à la racine carrée de de l'erreur quadratique moyenne (RMSE) et au temps de calcul en seconde.

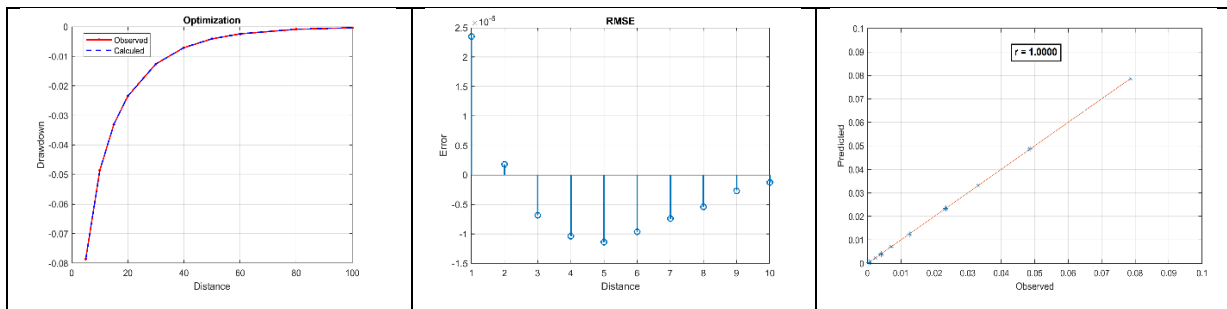


Fig.III.21 : (a) -Comparaison des courbes de rabattements en fonction du temps entre les valeurs observées et calculées après optimisation à l'aide du Recuit Simulé. (b) – Erreur relative entre les valeurs observées et calculées des rabattements et (c) – Droite de corrélation entre les valeurs observées et calculées des rabattements.

**b) – Optimisation de Harris Hawks :**

Pour le Problème 2, l'application de l'optimisation de Harris Hawks pour les mêmes caractéristiques hydrodynamiques, c'est-à-dire de transmissivité T égale à  $0.050 \text{ m/s}^2$  et de résistance de la couche semi-captive c égale  $10^4 \text{ s}$ , et en supposant un nombre d'agents de recherche N=10 et un nombre d'itérations égal à 10, le résultat obtenu est illustré par la figure III.22. Elle montre une convergence rapide, une très bonne restitution des rabattements, une erreur très faible de l'ordre du millimètre et une parfaite corrélation.

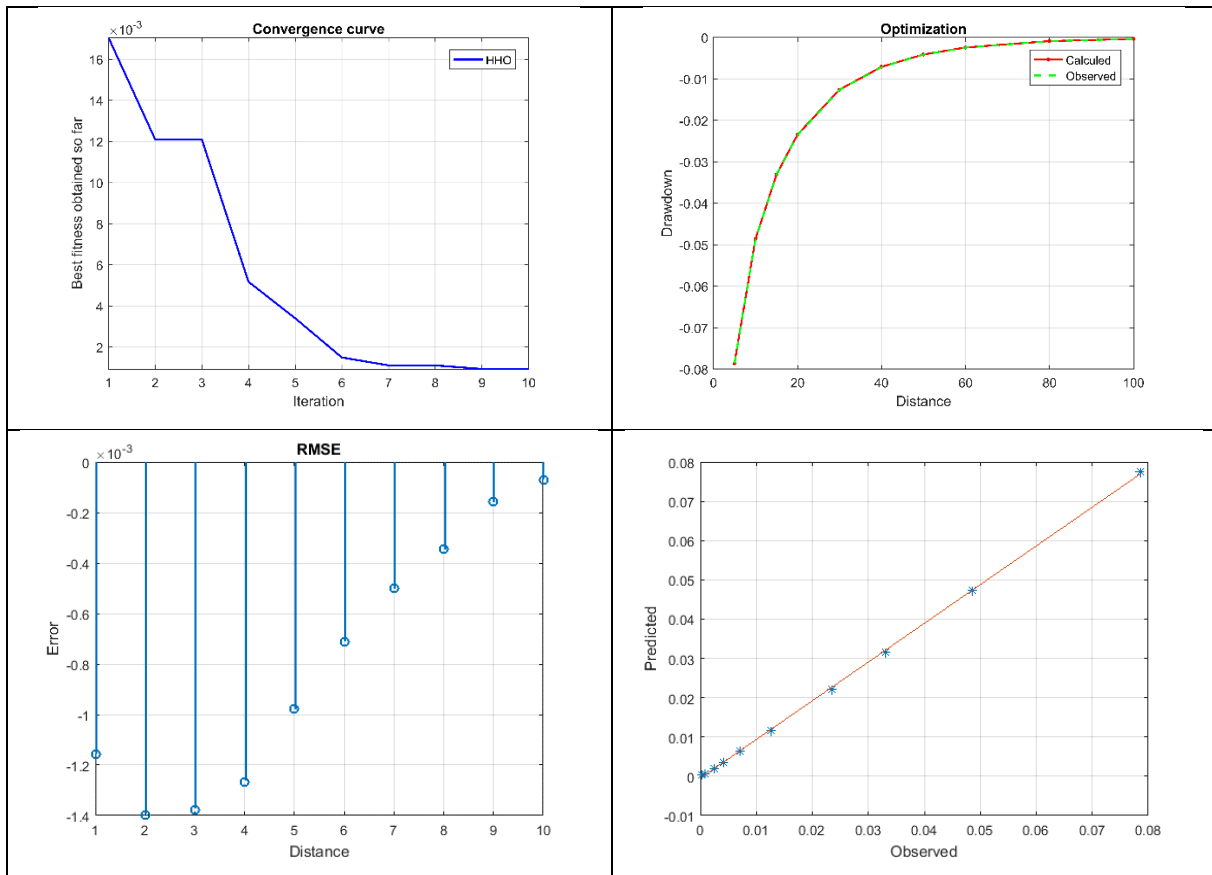


Fig.III.22 : (a) – Courbe de convergence, (b) -Comparaison des courbes de rabattements en fonction du temps entre les valeurs observées et calculées après optimisation à l’aide du Recuit Simulé (c) – Erreur relative entre les valeurs observées et calculées des rabattements et (d) – Droite de corrélation entre les valeurs observées et calculées des rabattements.

Afin de valider la méthodologie, cinq essais ont été effectués avec un nombre d’itérations respectivement égal à 10, 50, 100, 150 et 200 et un nombre d’agents de recherche fixe égal à 10, les résultats obtenus sont résumés dans le tableau III.5. Nous constatons la bonne performance qui augmente avec le nombre d’itération. Le temps de calcul reste relativement très faible avec le nombre d’itérations. L’application de cette exemple montre la haute performance de l’optimisation de Harris Hawks, avec un nombre d’agents assez réduit, les résultats obtenus sont très significatifs et très concluants. La figure III.23 montre les courbes de convergence des quatre autres essais. Les courbes montrent la rapidité de la convergence et sa stabilité.

Essais	1	2	3	4	5
N <sup>b</sup> Itérations	10	50	100	150	200
T	0.0492	0.0499	0.0501	0.0500	0.0500
C	9.1337 10 <sup>-3</sup>	9.9529 10 <sup>-3</sup>	1.0067 10 <sup>-4</sup>	9.9804 10 <sup>-3</sup>	9.9946 10 <sup>-3</sup>
RMSE	9.3028 10 <sup>-4</sup>	4.3762 10 <sup>-5</sup>	6.1441 10 <sup>-5</sup>	1.8175 10 <sup>-5</sup>	5.0842 10 <sup>-6</sup>
Temps	0.79456	0.65530	0.67281	0.70777	0.75109

Tableau III.5 : Résultats obtenus par l’optimisation de Harris Hawks pour différents essais correspondants aux Transmissivités T, aux Résistance de la couche semi-perméable C, à la racine carrée de de l’erreur quadratique moyenne (RMSE) et au temps de calcul en seconde.

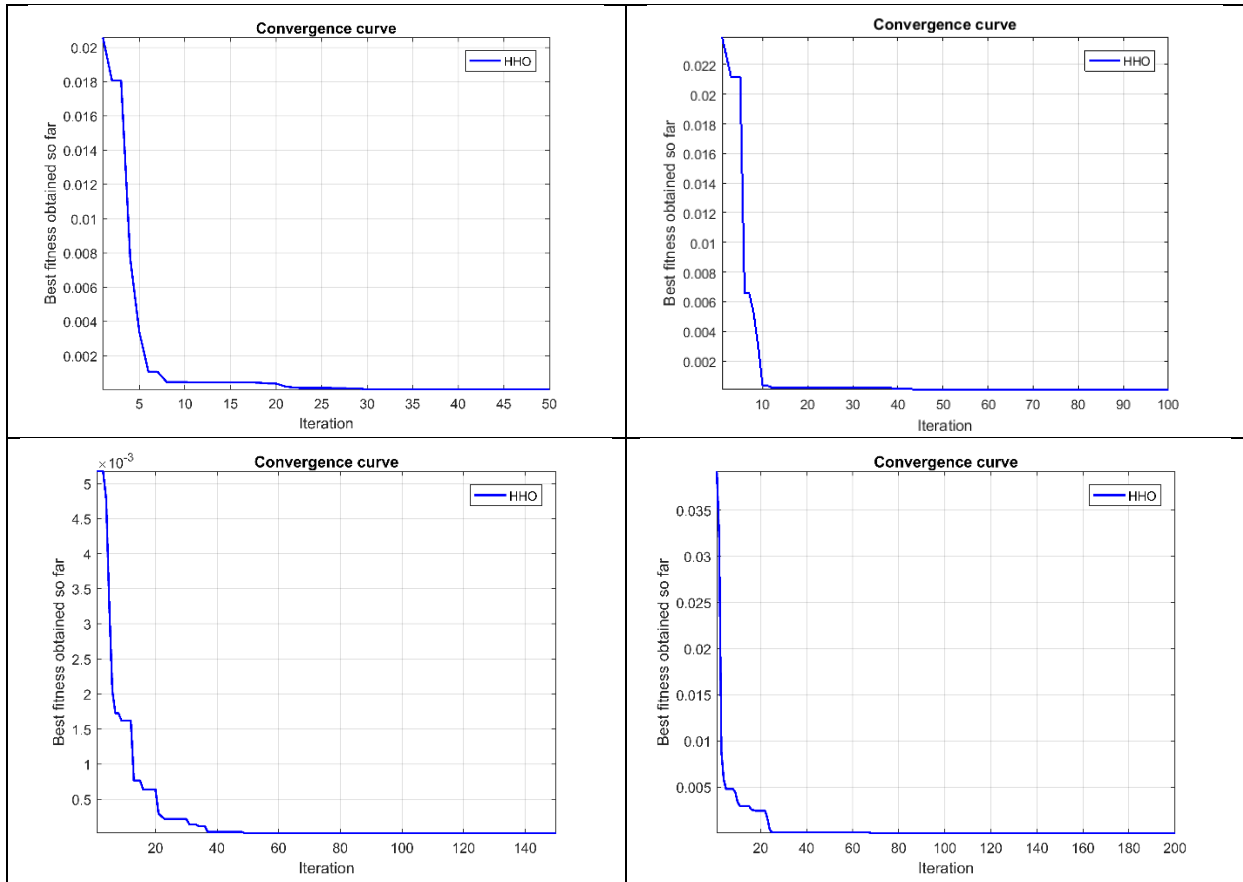


Fig.III.23 : Courbes de convergence de la fonction Objectif pour les itérations 50, 100, 150 et 200.

**III.3.3 Problème 3 :**

Le Problème correspond à l'interprétation d'un essai de pompage fictif réalisé dans une nappe libre. Le pompage est réalisé dans le forage de captage de 40 cm de diamètre avec un débit Q égal à 0.015 m<sup>3</sup>/s, et les observations des rabattements au niveau de 10 piézomètres distants respectivement de 5, 10, 20, 30, 40, 50, 60, 80, 100, 150 et 200 mètres. Pour cet exemple, la perméabilité proposée K est de 10<sup>-4</sup> m/s. En utilisant les relations de Thiem décrites précédemment (Equations 3.13 et 3.14), nous utiliserons l'optimisation par Recuit simulé et l'optimisation de Harris Hawks pour estimer la perméabilité K.

**a) – Optimisation par Recuit Simulé :**

Dix essais ont été réalisés pour estimer la perméabilité K et les caractéristiques de performance. Le tableau III.6 résume les résultats obtenus. L'essai n° 8 donne les meilleurs résultats une estimation parfaite de la perméabilité K (10<sup>-4</sup> m/s) et la meilleure erreur minimale. Ces résultats sont représentés par la figure III.24 qui montre la bonne superposition des courbes de rabattement observés et calculés, l'erreur très faible et la bonne corrélation.

Essai	1	2	3	4	5	6	7	8	9	10
K	9.9.10 <sup>-5</sup>	9.8.10 <sup>-5</sup>	9.9.10 <sup>-4</sup>	9.9.10 <sup>-5</sup>	1..10 <sup>-4</sup>	9.9.10 <sup>-5</sup>	9.980.10 <sup>-5</sup>	1.0.10 <sup>-4</sup>	9.9.10 <sup>-5</sup>	1.0.10 <sup>-4</sup>
RMSE	8.86.10 <sup>-5</sup>	0.0019	6.69.10 <sup>-5</sup>	6.18.10 <sup>-5</sup>	1.61.10 <sup>-4</sup>	7.10.10 <sup>-4</sup>	3.61.10 <sup>-4</sup>	5.59.10 <sup>-5</sup>	7.88.10 <sup>-4</sup>	7.37.10 <sup>-4</sup>
Temps	0.436869	0.529747	0.616286	0.589043	0.627828	0.549147	0.478401	0.821352	0.621600	0.577387

Tableau III.6 : Résultats obtenus à l'aide du Recuit Simulé pour différents essais correspondants aux Perméabilités K en fonction de la distance, à la racine carrée de de l'erreur quadratique moyenne (RMSE) et au temps de calcul en seconde.

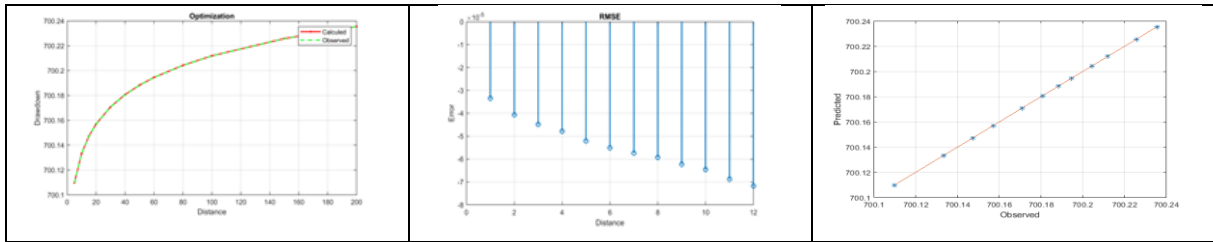


Fig.III.24 : (a) - Comparaison des courbes de rabattements en fonction de la distance entre les valeurs observées et calculées après optimisation à l'aide du Recuit Simulé. (b) – Erreur relative entre les valeurs observées et calculées des rabattements et (c) – Droite de corrélation entre les valeurs observées et calculées des rabattements.

**b) – Optimisation de Harris Hawks :**

L'application de l'optimisation de Harris Hawks pour le Problème 3 a été lancée pour un nombre d'agents de recherche égal à 30 et un nombre d'itérations égal à 10. Les résultats obtenus sont présentés par la figure III.25. Elle montre une courbe de convergence qui décroît très rapidement et une bonne superposition des courbes de rabattements avec une erreur relative très très faible ainsi qu'une bonne corrélation.

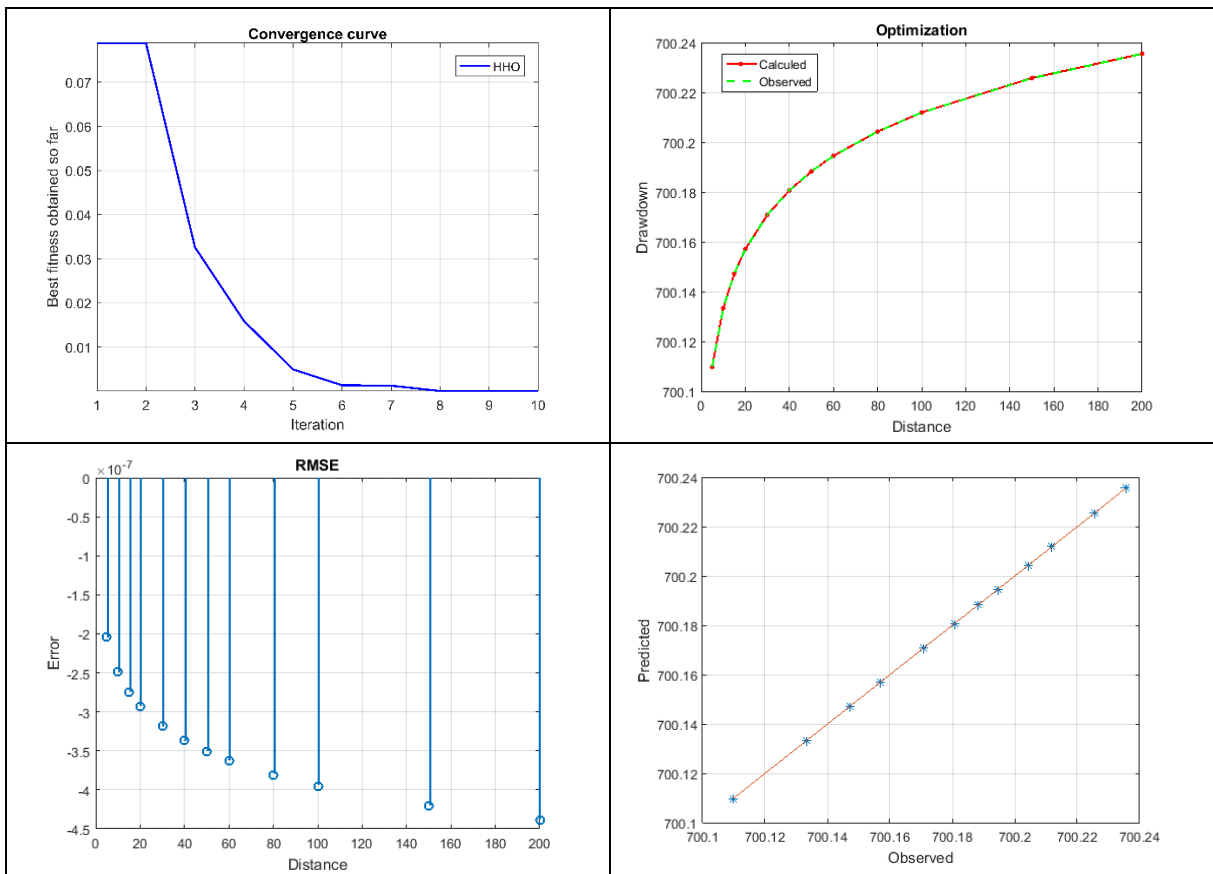


Fig.III.25 : (a) – Courbe de convergence, (b) -Comparaison des courbes de rabattements en fonction de la distance entre les valeurs observées et calculées après optimisation de Harris Hawks (c) – Erreur relative entre les valeurs observées et calculées des rabattements et (d) – Droite de corrélation entre les valeurs observées et calculées des rabattements.

Le tableau III.7 regroupe les principaux résultats obtenus pour des itérations 10, 20, 30, 40, 100 et 200. La perméabilité a été estimée parfaitement à partir d'un nombre d'itérations égal 20. Le temps de calcul est également très court montrant la robustesse de la méthode. L'erreur est très faible voire nulle. La figure III.26 confirme cette

conclusion en montrant des courbes de convergence très rapide puis progressive et en fin stable.

N <sup>o</sup> itérations	10	20	30	40	100	200
K	$9.9961 \cdot 10^{-5}$	$1.0 \cdot 10^{-4}$	$1.0 \cdot 10^{-4}$	$1.0 \cdot 10^{-4}$	$1.0 \cdot 10^{-4}$	$1.0 \cdot 10^{-4}$
RMSE	$7.1549 \cdot 10^{-5}$	$1.8368 \cdot 10^{-7}$	$3.4308 \cdot 10^{-9}$	$3.2884 \cdot 10^{-10}$	$3.2819 \cdot 10^{-14}$	0.000
Temps	0.610871	0.706365	0.715378	0.732615	0.688708	0.766530

Tableau III.7 : Résultats obtenus par l'optimisation de Harris Hawks pour différentes itérations essais correspondants aux Perméabilités K, à la racine carrée de de l'erreur quadratique moyenne (RMSE) et au temps de calcul en seconde.

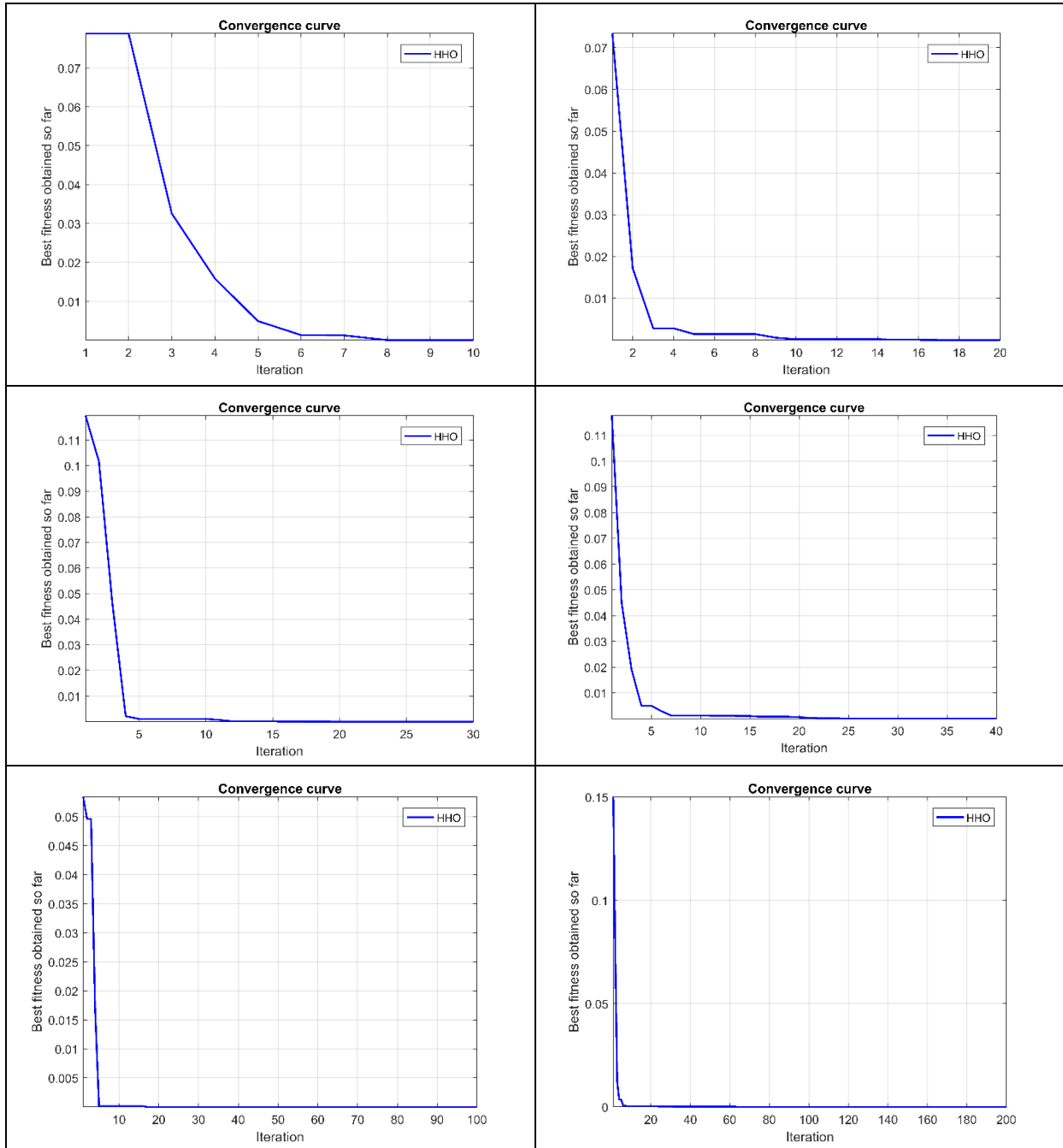


Fig.III.26 : Courbes de convergence de la fonction Objectif pour les itérations 10, 20, 30, 40, 100 et 200.

### III.4. Interprétation des essais de pompage :

Cette section constitue la phase ultime de ce travail d'interprétation des essais de pompage de quelques cas pratiques par les deux techniques d'optimisation du recuit simulé et de Harris Hawks.

#### III.4.1. Premier cas :

Le premier cas correspond à un essai de pompage d'une durée de 24 heures réalisé dans la région de Aïn Madhi pour une nappe captive. Le débit de pompage est de 0.009 m<sup>3</sup>/s, et les mesures des rabattements au niveau du forage d'exploitation. Les données utilisées importées du fichier de données sont présentées ci-dessous :

```

*****
s = [0.68 0.78 0.84 0.88 0.90 0.95 0.97 0.98 0.99 1.07 1.05 1.06 1.08 1.11 ...
1.12 1.14 1.18 1.19 1.22 1.23 1.24 1.26 1.27 1.28 1.29 1.31 1.31 1.31 ...
1.32 1.33 1.34 1.35 1.37 1.38 1.39 1.40 1.41 1.41 1.42 1.43 1.44 1.45 ...
1.45 1.46 1.47 1.48 1.48 1.48 1.48 1.49 1.50 1.51 1.52 1.53 1.53 1.56 ...
1.55 1.55 1.56 1.57 1.57 1.58 1.58 1.59 1.59 1.60 1.60 1.61 1.61 1.62 ...
1.63 1.63 1.64 1.65 1.66 1.68 1.68 1.68 1.70 1.70 1.70 1.71 1.71 1.72 ...
1.72];
DeltaObserv=s;
*****
t=[60 120 180 240 300 360 420 480 540 600 720 840 960 1080 1200 1500 1800 ...
2100 2400 2700 3000 3300 3600 3900 4200 4500 4800 5100 5400 5700 6000 ...
6600 7200 7800 8400 9000 9600 10200 10800 11400 12000 12600 13200 13800 ...
14400 15000 15600 16200 16800 17400 18000 19200 20400 21600 22800 24000 ...
25200 26400 27600 28800 30000 31200 32400 33600 34800 36000 37200 38400 ...
39600 40800 42000 45600 49200 52800 56400 60000 61800 63600 67200 70800 ...
74400 78000 79800 81600 84600];
Q=0.009;
r=0.20 ;
*****
    
```

Les résultats obtenus par l'optimisation du recuit simulé sont présentés par la figure III.27 La figure montre une très bonne restitution des rabattements, une erreur très faible et une très bonne corrélation. Pour un débit de 0.009 m<sup>3</sup>/s, et après 1629 itérations et 1638 Fonctions d'évaluation, l'erreur quadratique moyenne n'est que **0.00774** et le coefficient de corrélation entre rabattements observés et calculés est de 0.9995 et le temps de calcul effectué est de **6.405** secondes, la transmissivité estimée est alors de **0.0050** m<sup>2</sup>/s et le coefficient d'emmagasinement est de **0.148**.

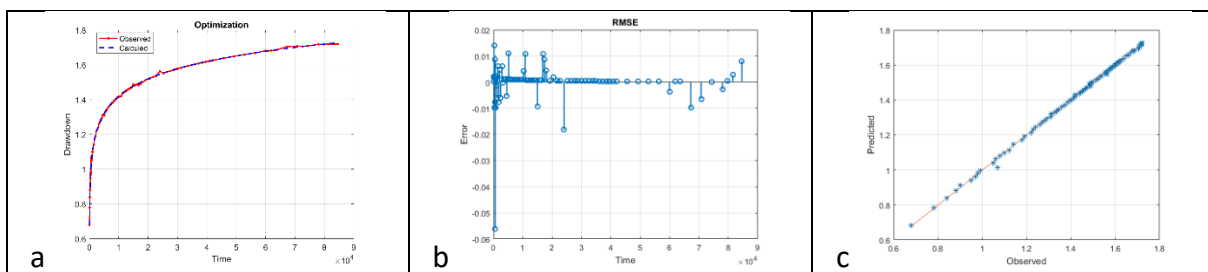


Fig.III.27 : - (a) - Comparaison des courbes de rabattements en fonction du temps entre les valeurs observées et calculées après optimisation à l'aide du Recuit Simulé. (b) – Erreur relative entre les valeurs observées et calculées des rabattements et (c) – Droite de corrélation entre les valeurs observées et calculées des rabattements.

L'application de l'optimisation de Harris Hawks a été réalisée pour quatre essais différents, afin de valider les résultats obtenus. Ainsi, le tableau III.8 regroupe les résultats des différentes simulations obtenues par l'optimisation de Harris Hawks.

Essais	Paramètres	Résultats
1	Nombre d'agents de Recherche	50
	Nombre d'itérations	100
	Transmissivité	0.0058
	Coefficient d'emmagasinement	0.0340
	RMSE	0.0358
	Coefficient de corrélation	0.9995
	Temps de calcul en secondes	8.822666
2	Nombre d'agents de Recherche	100
	Nombre d'itérations	300
	Transmissivité	0.0049
	Coefficient d'emmagasinement	0.1640
	RMSE	0.0085
	Coefficient de corrélation	0.9995
	Temps de calcul en secondes	53.050025
3	Nombre d'agents de Recherche	100
	Nombre d'itérations	400
	Transmissivité	0.0052
	Coefficient d'emmagasinement	0.1036
	RMSE	0.0119
	Coefficient de corrélation	0.9995
	Temps de calcul en secondes	70.282960
4	Nombre d'agents de Recherche	150
	Nombre d'itérations	400
	Transmissivité	<b>0.0050</b>
	Coefficient d'emmagasinement	<b>0.1328</b>
	RMSE	<b>0.0081</b>
	Coefficient de corrélation	0.9995
	Temps de calcul en secondes	<b>104.218991</b>

Tableau III.8 : Résultats obtenus par l'optimisation de Harris Hawks pour quatre essais différents pour estimer la transmissivité et le coefficient d'emmagasinement de l'essai de pompage de Aïn Madhi.

Les résultats obtenus montrent que pour le premier essai, la transmissivité et le coefficient d'emmagasinement sont grossièrement évalués, mais lorsqu'on augmente le nombre d'agents de recherche et le nombre d'itération suffisamment, on converge vers les bonnes valeurs. Ainsi, pour un nombre d'agents de recherche égal à 150 et un nombre d'itérations égal à 400, la transmissivité est de **0.050** m<sup>2</sup>/s égale à la valeur estimée par le recuit simulé, et le coefficient d'emmagasinement égal à **13.28%** légèrement inférieur à celui du recuit simulé estimé à 14.80%. Cependant, ce résultat est obtenu au détriment du temps de calcul suffisamment long relativement au temps de calcul très réduit dans le cas du recuit simulé. Les meilleurs résultats obtenus par l'optimisation de Harris Hawks sont illustrés par la figure III.28.

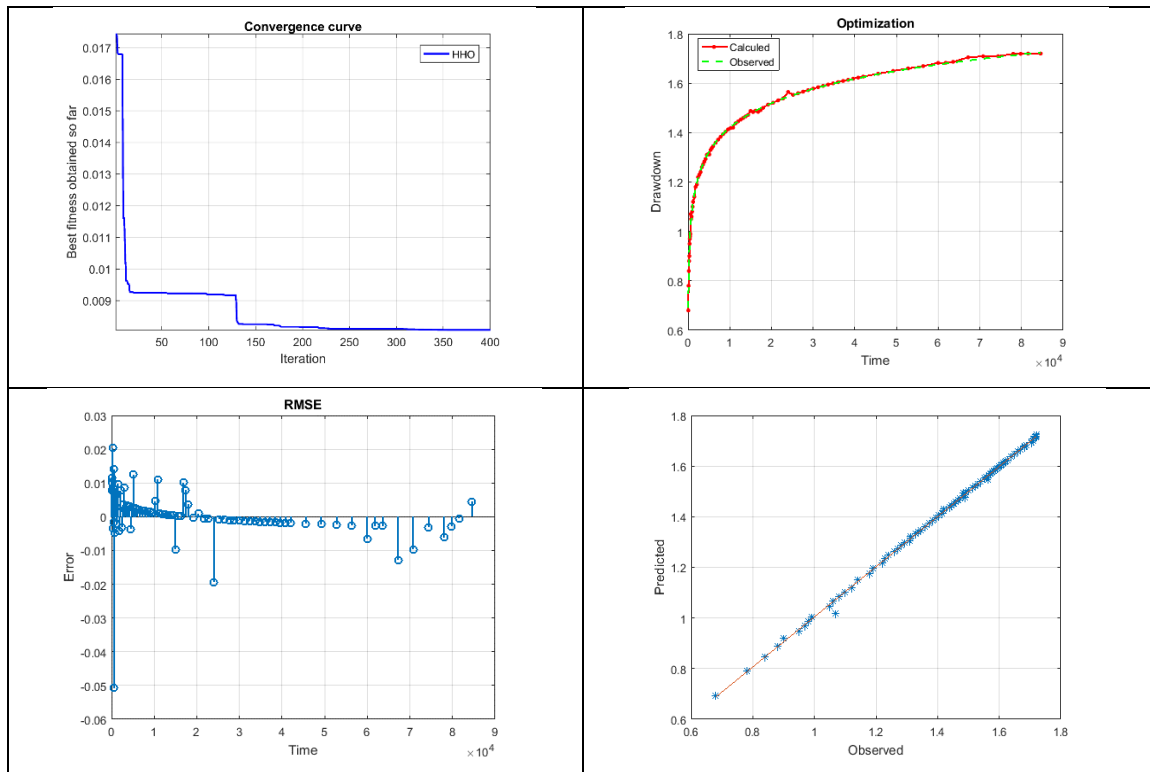


Fig.III.28 : - Résultats obtenus par l'optimisation de Harris Hawks pour l'essai n° 4 correspondant à la courbe de convergence, aux courbes des rabattements observés et calculés, à l'erreur relative et à la corrélation entre les rabattements observés et calculés.

**III.4.2. Deuxième cas :**

Le deuxième cas est celui d'un essai de pompage exécuté dans une nappe captive dans le polder "Oude Korendijk", au Sud de Rotterdam aux Pays-Bas. L'essai de pompage a été contrôlé par l'Institut de recherche du génie rural et de l'hydrologie agricole Hollandais.

La Figure III.29 donne une coupe géologique de la région, obtenue à partir des coupes des sondages. Le pompage est réalisé au niveau du forage et les mesures des rabattements au niveau de quatre piézomètres situés à 0.80, 30, 90 et 215 mètres du forage de pompage.

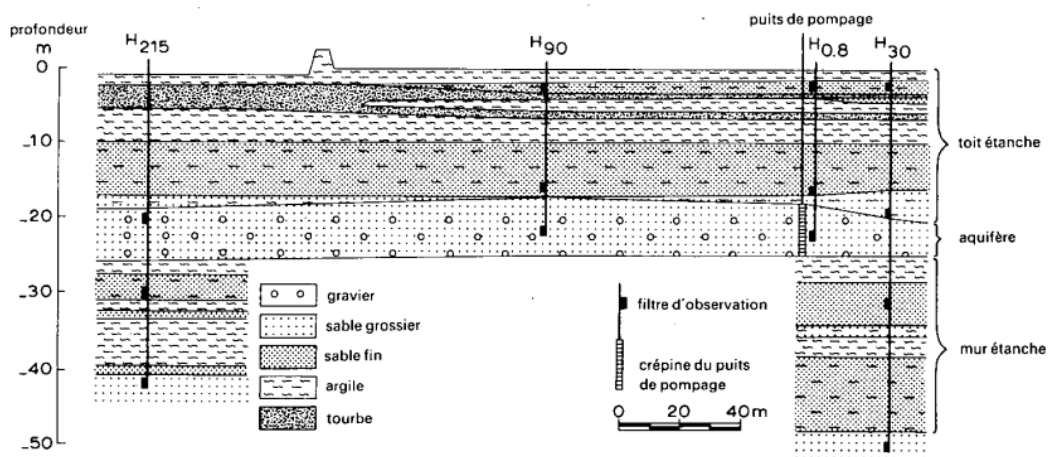


Fig.III.29 : Coupe géologique de la région de l'essai de pompage d'Oude Korendijk.

Le Tableau III.9 résume les rabattements obtenus après environ 14 heures de pompage au débit constant  $Q = 0.00912 \text{ m}^3/\text{s}$ .

Piézomètres	H <sub>0,8</sub> 0.8 (m)	H <sub>30</sub> 30 (m)	H <sub>90</sub> 90 (m)	H <sub>215</sub> 215 (m)
Rabattements (m)	2.236	1.088	0.716	0.250

Tableau III.9 : - Rabattements dans les piézomètres entre 20 et 24 m de profondeur après 830 minutes de pompage (Essai d'Oude Korendjik)

Les résultats obtenus par l'optimisation du recuit simulé en utilisant le modèle de Thiem (équations 3.13 et 3.14).

Ainsi, un seul essai a été réalisé pour une tolérance de  $10^{-5}$  pour la fonction **Objectif**, la perméabilité K obtenue est de  $4.3772 \cdot 10^{-5}$ , soit une transmissivité de  $3.064 \cdot 10^{-4} \text{ m}^2/\text{s}$  si on considère l'épaisseur saturée entre 18 et 25 m. L'erreur quadratique moyenne (RMSE) obtenue dans ce cas est de 0.0735, et le coefficient de corrélation est de 0.09953 pour un temps de calcul très court de 0.5737 seconde.

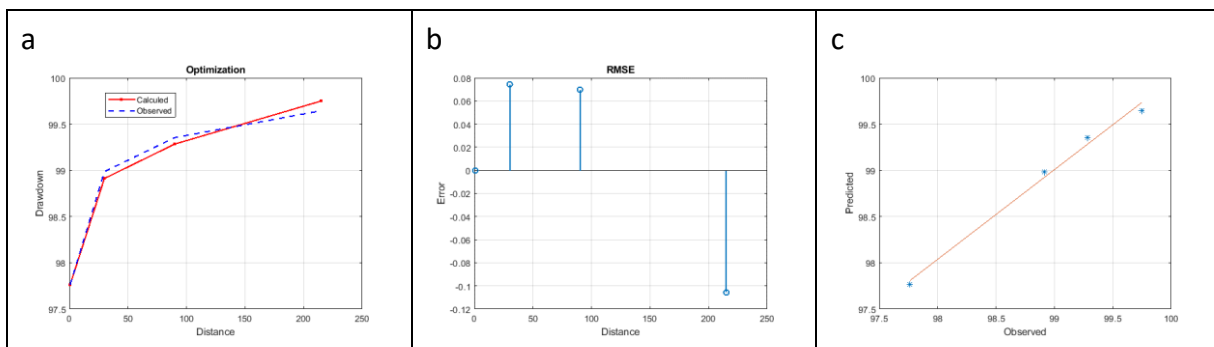


Fig. III.30 : - (a) - Comparaison des courbes de rabattements en fonction du temps entre les valeurs observées et calculées après optimisation à l'aide du Recuit Simulé. (b) – Erreur relative entre les valeurs observées et calculées des rabattements et (c) – Droite de corrélation entre les valeurs observées et calculées des rabattements

Pour, l'optimisation de Harris Hawks, deux essais ont été réalisés pour deux itérations 10 et 100. Ainsi, les résultats obtenus sont présentés dans le tableau III.10. Nous avons obtenu les mêmes résultats pour les deux itérations, le temps de calcul est la seule différence. L'optimisation de Harris Hawks donne la même valeur de perméabilité que dans le cas de l'optimisation du recuit simulé. La figure III.31 illustre le résultat du deuxième essai.

Essais	1	2
Nombre d'itérations	10	100
K (perméabilité m/s)	$4.3772 \cdot 10^{-5}$	$4.3772 \cdot 10^{-5}$
Erreur	0.0735	0.0735
Coefficient de corrélation	0.9953	0.9953
Temps de calcul	0.147237	0.573746

Tableau III.10 : - Résultats obtenus par l'optimisation de Harris Hawks pour les deux essais différents Pour estimer la Perméabilité de l'essai de pompage d'Oude Korendjik.

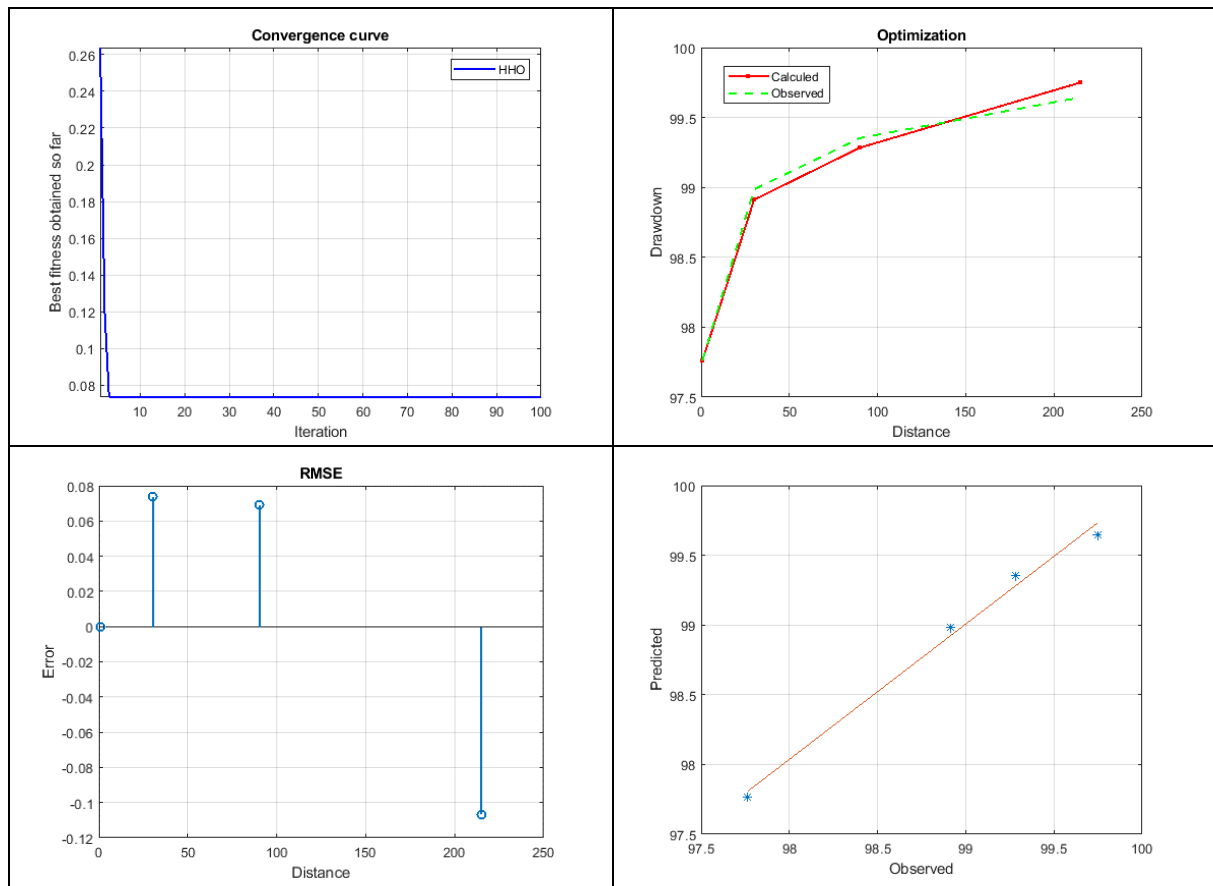


Fig.III.31: - Résultats obtenus par l'optimisation de Harris Hawks pour l'essai n° 2 correspondants à la courbe de convergence, aux courbes des rabattements observés et calculés, à l'erreur relative et à la corrélation entre les rabattements observés et calculés.

### III.5. Conclusion

L'interprétation des essais de pompage par les méthodes d'optimisation métaheuristiques a révélé des résultats très satisfaisants et encourageants. Les deux méthodes utilisées : l'optimisation par recuit simulé et l'optimisation de Harris Hawks dans cette étude ont donné des résultats significatifs mais peu différents à causes des paramètres introduits dans les codes de calcul à savoir la tolérance, le nombre d'itérations, les valeurs de températures, le nombre d'agents de recherche, etc.

Chacune des méthodes à ses propres avantages et ses propres inconvénients. Ainsi, le recuit simulé est une méthode d'optimisation robuste et flexible, son principal avantage est sa capacité à échapper aux minima locaux et à explorer une vaste gamme de solutions possibles, augmentant ainsi la probabilité de trouver la solution optimale. Cependant, la méthode du recuit simulé peut être lente pour des problèmes de grande taille et nécessite un ajustement minutieux des paramètres pour chaque application spécifique (comme la température initiale, le taux de refroidissement, etc.).

Par ailleurs, la méthode d'optimisation de Harris Hawks qu'elle soit prometteuse et présente certains avantages, elle n'est pas sans inconvénients. Son efficacité dépend fortement de la

manière dont elle est mise en œuvre et des caractéristiques du problème à résoudre. Comme de nombreux algorithmes d'optimisation, la performance de la méthode des Harris Hawks peut diminuer considérablement avec l'augmentation de la dimensionnalité de l'espace de recherche. Cela peut limiter son efficacité pour résoudre des problèmes de grande dimension. Ce problème n'a pas été abordé dans le cas d'interprétation des essais de pompage, mais il peut être très important dans le cas d'un calage automatique pour optimiser les paramètres hydrodynamiques au cours d'une modélisation hydrogéologique.

---

# Conclusion Générale

---

L'interprétation des essais de pompage à l'aide des techniques d'optimisation métaheuristiques en l'occurrence, le recuit simulé et l'optimisation de Harris Hawks a montré une bonne performance pour l'estimation des paramètres hydrodynamiques.

L'application du recuit simulé a permis de constater qu'il peut trouver l'optimum global de la fonction Objectif, même en présence de multiples optima locaux. La méthode est caractérisée par une très bonne flexibilité et peut être appliquée à un large éventail de problèmes d'optimisation et caractérisée également par sa forte robustesse et moins susceptible de rester bloquée dans des optima locaux que d'autres techniques d'optimisation. Par ailleurs, le recuit simulé peut être plus lent que d'autres algorithmes d'optimisation et ses performances dépendent du choix du schéma de refroidissement et d'autres paramètres, qui peuvent être difficiles à déterminer. Le recuit simulé peut nécessiter une grande quantité de mémoire pour stocker les solutions actuelles et candidates, en particulier pour les problèmes dont l'espace de recherche est vaste.

La méthode des Harris Hawks est basée sur un modèle naturel inspiré du comportement des oiseaux de proie, ce qui lui confère une certaine robustesse et une capacité à gérer des problèmes non linéaires et non convexes. Elle est souvent plus rapide que le recuit simulé dans certains cas, en particulier pour les problèmes où une exploration efficace de l'espace de recherche est cruciale. La méthode d'optimisation des Harris Hawks, bien qu'elle ait ses avantages, elle présente également certains inconvénients parmi lesquels : la complexité de mise en œuvre, la sensibilité aux paramètres, les problèmes de convergence, la dépendance à la dimension et le manque de garanties théoriques. Ces inconvénients peuvent faire l'objet d'autres projets de recherche approfondis.

Les résultats obtenus par ces deux métaheuristiques semblent très encourageants et prometteurs et permettent d'être exploitables pour estimer les réserves et utilisés pour lancer des modèles mathématiques de gestion de nappes. Les deux méthodes peuvent être testées également sur des problèmes hydrogéologiques forts différents comme les milieux fissurés et karstiques.

Cependant, la méthodologie a révélé que les caractéristiques hydrodynamiques qui présentent une forte sensibilité aux modèles hydrodynamiques comme la transmissivité et la perméabilité sont les mieux estimées, alors que les caractéristiques les moins sensibles comme le coefficient d'emmagasinement sont sommairement estimées. Ce type de problème peut être résolu en augmentant les propriétés de performances des deux algorithmes.

Les résultats obtenus par ces deux algorithmes peuvent être confrontés à d'autres méthodes d'optimisation employées dans le cadre de l'ajustement de modèles aux données expérimentales comme la méthode de Gauss-Levenberg-Marquardt qui constitue une technique d'optimisation non linéaire largement utilisée pour résoudre des problèmes de moindres carrés non linéaires.

## Références Bibliographiques:

- Alabool, H. M., Alarabiat, D., Abualigah, L., & Heidari, A. A. (2021). Harris hawks optimization: a comprehensive review of recent variants and applications. *Neural computing and applications*, 33, 8939-8980.
- Al-Betar, M. A., Awadallah, M. A., Heidari, A. A., Chen, H., Al-Khraisat, H., & Li, C. (2021). Survival exploration strategies for Harris hawks optimizer. *Expert Systems with Applications*, 168, 114243.
- Aleem, S.H.A., Zobaa, A.F., Balci, M.E. and Ismael, S.M., 2019. Harmonic overloading minimization of frequencydependent components in harmonics polluted distribution systems using Harris hawks op.
- Attiya, I., Abd Elaziz, M., & Xiong, S. (2020). Job scheduling in cloud computing using a modified Harris hawks optimization and simulated annealing algorithm. *Computational intelligence and neuroscience*.
- Bednarz, J. C. (1988). Cooperative hunting Harris' hawks (*Parabuteo unicinctus*). *Science*, 239(4847), 1525-1527.
- Borkar, V.S.: Pathwise recurrence orders and simulated annealing. *J. Appl. Probab.* 29, 472–476 (1992).
- Cai, Z. (2021). Sa-gd: Improved gradient descent learning strategy with simulated annealing. *arXiv preprint arXiv:2107.07558*.
- Černý, V. (1985). Thermodynamical approach to the traveling salesman problem: An efficient simulation algorithm. *Journal of optimization theory and applications*, 45, 41-51.
- Chen, H., Heidari, A. A., Chen, H., Wang, M., Pan, Z., & Gandomi, A. H. (2020). Multi-population differential evolution-assisted Harris hawks optimization: Framework and case studies. *Future Generation Computer Systems*, 111, 175-198.
- Chen, H., Jiao, S., Wang, M., Heidari, A. A., & Zhao, X. (2020). Parameters identification of photovoltaic cells and modules using diversification-enriched Harris hawks optimization with chaotic drifts. *Journal of Cleaner Production*, 244, 118778.
- Cohn, H., & Fielding, M. (1999). Simulated annealing: searching for an optimal temperature schedule. *SIAM Journal on Optimization*, 9(3), 779-802.
- Connors, D. P., & Kumar, P. R. (1989). Simulated annealing type Markov chains and their order balance equations. *SIAM Journal on Control and Optimization*, 27(6), 1440-1461.
- Delamarre, D., & Viot, B. (1998). Simulated annealing algorithm: technical improvements. *RAIRO-Operations Research*, 32(1), 43-73.
- Devarapalli, R., & Bhattacharyya, B. (2019, December). Application of modified Harris hawks optimization in power system oscillations damping controller design. In 2019 8th international conference on power systems (ICPS) (pp. 1-6). IEEE.
- Devarapalli, R., & Bhattacharyya, B. (2019, December). Optimal parameter tuning of power oscillation damper by MHHO algorithm. In 2019 20th International conference on intelligent system application to power systems (ISAP) (pp. 1-7). IEEE.
- Dhawale, D., & Kamboj, V. K. (2020, January). HHHO-IGWO: A new hybrid Harris hawks optimizer for solving global optimization problems. In 2020 international conference on computation, automation and knowledge management (ICCAKM) (pp. 52-57). IEEE.
- Du, P., Wang, J., Hao, Y., Niu, T., & Yang, W. (2020). A novel hybrid model based on multi-objective Harris hawks optimization algorithm for daily PM<sub>2.5</sub> and PM<sub>10</sub> forecasting. *Applied Soft Computing*, 96, 106620.
- Ewees, A. A., & Abd Elaziz, M. (2020). Performance analysis of chaotic multi-verse Harris hawks optimization: a case study on solving engineering problems. *Engineering Applications of Artificial Intelligence*, 88, 103370.

- Faigle, U., Schrader, R.: On the convergence of stationary distributions in simulated annealing algorithms. *Inf. Process. Lett.* 27, 189–194 (1988).
- Fan, Q., Chen, Z., & Xia, Z. (2020). A novel quasi-reflected Harris hawks optimization algorithm for global optimization problems. *Soft Computing*, 24, 14825-14843.
- Fatyanosa, T. N., Sihananto, A. N., Alfarisy, G. A. F., Burhan, M. S., & Mahmudy, W. F. (2016). Hybrid genetic algorithm and simulated annealing for function optimization. *Journal of Information Technology and Computer Science*, 1(2), 82-97.
- Forkasiewicz, J., (1970). Guide bibliographique méthodique sur l'interprétation des données de pompages d'essai (Evaluation des paramètres aquifères). *Bull. Bur. Rech. Géol. Minières (Fr.)*, Sect. 2, 3'(1): 67-72.
- Golilarz, N. A., Addeh, A., Gao, H., Ali, L., Roshandeh, A. M., Munir, H. M., & Khan, R. U. (2019). A new automatic method for control chart patterns recognition based on ConvNet and Harris hawks meta heuristic optimization algorithm. *Ieee Access*, 7, 149398-149405.
- Granville, V., Krivánek, M., & Rasson, J. P. (1994). Simulated annealing: A proof of convergence. *IEEE transactions on pattern analysis and machine intelligence*, 16(6), 652-656.
- Gupta, S., Deep, K., Heidari, A. A., Moayedi, H., & Wang, M. (2020). Opposition-based learning Harris hawks optimization with advanced transition rules: principles and analysis. *Expert Systems with Applications*, 158, 113510.
- Hajek, B. (1988). Cooling schedules for optimal annealing. *Mathematics of operations research*, 13(2), 311-329.
- Hammersley, J. M., & Handscomb, D. C. (1964). *Monte Carlo Methods* (Methuen and Co. Ltd., London).
- Heidari, A. A., Mirjalili, S., Faris, H., Aljarah, I., Mafarja, M., & Chen, H. (2019). Harris hawks optimization: Algorithm and applications. *Future generation computer systems*, 97, 849-872.
- Houssein, E. H., Hosney, M. E., Oliva, D., Mohamed, W. M., & Hassaballah, M. (2020). A novel hybrid Harris hawks optimization and support vector machines for drug design and discovery. *Computers & Chemical Engineering*, 133, 106656.
- Holzbecher, E. (2009). *Environmental Modeling: Using MATLAB®*. (n.p.): Springer Berlin Heidelberg.
- Hu, H., Ao, Y., Bai, Y., Cheng, R., & Xu, T. (2020). An improved Harris's Hawks optimization for SAR target recognition and stock market index prediction. *IEEE Access*, 8, 65891-65910.
- Johnson, A. W., & Jacobson, S. H. (2002). On the convergence of generalized hill climbing algorithms. *Discrete applied mathematics*, 119(1-2), 37-57.
- Kamboj, V. K., Nandi, A., Bhadoria, A., & Sehgal, S. (2020). An intensify Harris Hawks optimizer for numerical and engineering optimization problems. *Applied Soft Computing*, 89, 106018.
- Karabin, M., & Stuart, S. J. (2020). Simulated annealing with adaptive cooling rates. *The Journal of Chemical Physics*, 153(11).
- Kirkpatrick, S., Gelatt Jr, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *science*, 220(4598), 671-680.
- Kruseman, G.P., De Ridder, N.A. et Meilhac, A., (1974). *Interpretation et discussion des pompages d'essai*. ILRI, Wageningen, 213 pp.
- Kurtuluş, E., Yıldız, A. R., Sait, S. M., & Bureerat, S. (2020). A novel hybrid Harris hawks-simulated annealing algorithm and RBF-based metamodel for design optimization of highway guardrails. *Materials testing*, 62(3), 251-260.
- Li, C., Li, J., & Chen, H. (2020). A meta-heuristic-based approach for QoS-aware service composition. *IEEE Access*, 8, 69579-69592.

- Lundy, M., & Mees, A. (1986). Convergence of an annealing algorithm. *Mathematical programming*, 34, 111-124.
- Mahfoud, S. W., & Goldberg, D. E. (1995). Parallel recombinative simulated annealing: A genetic algorithm. *Parallel computing*, 21(1), 1-28.
- Mania, J. et Sucche, M., (1978). Analyse automatique des données de pompage d'essai -- Application aux hypothèses de Boulton et d'Hantush. (Automatic analysis of pumping test data -- Application to the Boulton and Hantush hypotheses.) *J. Hydrol.*, 37: 185--194.
- Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., & Teller, E. (1953). Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6), 1087-1092.
- Mitra, D., Romeo, F., & Sangiovanni-Vincentelli, A. (1986). Convergence and finite-time behavior of simulated annealing. *Advances in applied probability*, 18(3), 747-771.
- Moayedi, H., Abdullahi, M. A. M., Nguyen, H., & Rashid, A. S. A. (2021). Comparison of dragonfly algorithm and Harris hawks optimization evolutionary data mining techniques for the assessment of bearing capacity of footings over two-layer foundation soils. *Engineering with Computers*, 37, 437-447.
- Qu, C., He, W., Peng, X., & Peng, X. (2020). Harris hawks optimization with information exchange. *Applied mathematical modelling*, 84, 52-75.
- Rahbari, M., & Jahed, A. (2017, May). A hybrid simulated annealing algorithm for travelling salesman problem with three neighbor generation structures. In *10th International Conference of Iranian Operations Research Society (ICIORS 2017)*.
- Rodríguez-Esparza, E., Zanella-Calzada, L. A., Oliva, D., Heidari, A. A., Zaldivar, D., Pérez-Cisneros, M., & Foong, L. K. (2020). An efficient Harris hawks-inspired image segmentation method. *Expert Systems with Applications*, 155, 113428.
- Romeo, F., & Sangiovanni-Vincentelli, A. (1991). A theoretical framework for simulated annealing. *Algorithmica*, 6, 302-345.
- Rossier, Y., Troyon, M., & Liebling, T. M. (1986). Probabilistic exchange algorithms and Euclidean traveling salesman problems. *Operations-Research-Spektrum*, 8(3), 151-164.
- Shahid, M., Li, J. P., Golilarz, N. A., Addeh, A., Khan, J., & Haq, A. U. (2019, December). Wavelet based image de-noising with optimized thresholding using HHO algorithm. In *2019 16th International Computer Conference on Wavelet Active Media Technology and Information Processing* (pp. 6-12). IEEE.
- Shehabeldeen, T. A., Abd Elaziz, M., Elsheikh, A. H., & Zhou, J. (2019). Modeling of friction stir welding process using adaptive neuro-fuzzy inference system integrated with Harris hawks optimizer. *Journal of Materials Research and Technology*, 8(6), 5882-5892.
- Tadj, W., Chettih, M., and Mouattah, K., (2018). Automatic interpretation of pumping tests data using metaheuristics. *Arabian Journal of Geosciences* (2018) 11:393. <https://doi.org/10.1007/s12517-018-3730-0>.
- Tan, C. M. (Ed.). (2008). *Simulated annealing*. BoD—Books on Demand.
- Tayab, U. B., Zia, A., Yang, F., Lu, J., & Kashif, M. (2020). Short-term load forecasting for microgrid energy management system using hybrid HHO-FNN model with best-basis stationary wavelet packet transform. *Energy*, 203, 117857.
- Thaher, T., & Arman, N. (2020, April). Efficient multi-swarm binary Harris hawks optimization as a feature selection approach for software fault prediction. In *2020 11th International conference on information and communication systems (ICICS)* (pp. 249-254). IEEE.
- Thaher, T., Heidari, A. A., Mafarja, M., Dong, J. S., & Mirjalili, S. (2020). Binary Harris Hawks optimizer for high-dimensional, low sample size feature selection. *Evolutionary machine learning techniques: algorithms and applications*, 251-272.

- Tripathy, B. K., Maddikunta, P. K. R., Pham, Q. V., Gadekallu, T. R., Dev, K., Pandya, S., & ElHalawany, B. M. (2022). Harris Hawk optimization: a survey on variants and applications. *Computational Intelligence and Neuroscience*, 2022, 1-20.
- Tsitsiklis, J. N. (1989). Markov chains with rare transitions and simulated annealing. *Mathematics of Operations Research*, 14(1), 70-90.
- Wang, R., Chen, B., Qiu, S., Ma, L., Zhu, Z., Wang, Y., & Qiu, X. (2018). Hazardous source estimation using an artificial neural network, particle swarm optimization and a simulated annealing algorithm. *Atmosphere*, 9(4), 119.
- Yao, X., & Li, G. (1991). General simulated annealing. *Journal of Computer Science and Technology*, 6(4), 329-338.
- Yıldız, A. R., Yıldız, B. S., Sait, S. M., & Li, X. (2019). The Harris hawks, grasshopper and multi-verse optimization algorithms for the selection of optimal machining parameters in manufacturing operations. *Materials Testing*, 61(8), 725-733.
- Youssef, H., Sait, S. M., & Adiche, H. (2001). Evolutionary algorithms, simulated annealing and tabu search: a comparative study. *Engineering Applications of Artificial Intelligence*, 14(2), 167-181.
- Yu, Z., Shi, X., Zhou, J., Chen, X., & Qiu, X. (2020). Effective assessment of blast-induced ground vibration using an optimized random forest model based on a Harris hawks optimization algorithm. *Applied Sciences*, 10(4), 1403.
- Zhang, Y., Zhou, X., & Shih, P. C. (2020). Modified Harris Hawks optimization algorithm for global optimization problems. *Arabian Journal for Science and Engineering*, 45, 10949-10974.
- Zheng-Ming, G. A. O., Juan, Z. H. A. O., Yu-Rong, H. U., & Chen, H. F. (2019, October). The improved Harris hawk optimization algorithm with the Tent map. In 2019 3rd International conference on electronic information technology and computer engineering (EITCE) (pp. 336-339). IEEE.