

UNIVERSITE KASDI MERBAH OUARGLA

Faculté des Sciences Appliquées

Département de Génie Electrique



Mémoire

MASTER ACADEMIQUE

Domaine : Sciences et technologies

Filière : Electrotechnique

Spécialité : Réseaux électriques

Présenté par :

Ghettas Mohammed & Belhani Said

**Thème:**

Etude et optimisation de l'écoulement de puissance par  
l'algorithme de recherche de corbeaux

'Crow Serach Algorithm(CSA)'

Soumis au jury composé de :

M <sup>r</sup> Guehrrar Youcef	MAA	Président	UKM Ouargla
M <sup>r</sup> Larouci Benyekhlef	MCA	Encadreur/rapporteur	UKM Ouargla
M <sup>r</sup> Benbouza Naima	MAA	Examineur	UKM Ouargla

Année universitaire 2020/2021

# Remerciements

Avant tout, nous remercions Dieu le tout puissant de nous avoir donné courage, patience et force durant toutes ces années d'étude et que grâce à lui ce travail a pu être réalisé.

Nous tenons à exprimer notre remerciement et notre gratitude à nos encadreurs :

\* Mr : Larouci Benyekhle pour la confiance qu'il nous a prodigué pour la direction de ce travail, sans cesser de nous encourager et de nous pousser vers les horizons de recherche.

Nous remercions vivement Monsieur

Chef département de Génie électrique \* Mr : LOUAZENE MOHAMED LAKHDAR

Nos remerciements au département de Génie Electrique de l'université de Ouargla et à tous les enseignants qui nous enseignés durant les années du cursus.

Nos remerciements sont également adressés aux membres de jury qui ont accepté de juger ce travail.

Enfin nous remercions aussi toutes les personnes qui nous ont aidés de près ou de loin à la rédaction de ce travail. Nous vous remercions de la collection.

Ghettas Mohammed

Belhani Said

# *Dédicace*

*Je dédie ce modeste travail  
A ma très chère mère source de tendresse  
A mon très cher père, qui m'encourage  
Dans les instants délicats*

*A mes chers frères  
A mes chères sœurs  
A toute ma famille  
A tous mes amis*

*Ghettas Mohammed*



# *Dédicace*

*Je dédie ce modeste travail  
A ma très chère mère source de tendresse  
A mon très cher père, qui m'encourage  
Dans les instants délicats*

*A mes chers frères  
A mes chères sœurs  
A toute ma famille  
A tous mes amis*

*Belhani Said*

# Sommaire

Sommaire.....	I
Liste des figures .....	III
Liste des Tableaux .....	IV
Liste des symboles .....	V
Résumé .....	VII
Introduction générale .....	1
Chapitre I L'OBSTACLE DE LA REPARTITION OPTIMALE DES PUISSANCES .....	3
I.1 Introduction .....	3
I.2 Programme de marche des unités de production .....	3
I.3 Enonce du problème .....	4
I.4 Problème de la répartition optimale des puissances .....	4
I.4.1 Présentation de la fonction objective.....	6
I.4.2 Les contraintes .....	7
I.4.3 Classification des variables .....	7
I.4.4 Présentation du problème .....	8
I.5 Problème de l'écoulement de puissance.....	9
I.5.1 Introduction .....	9
I.5.2 Modélisation du réseau.....	9
I.5.3 Calcul de l'écoulement des puissances par la méthode de Gauss- Seidel .....	11
I.6 Formulation de la fonction objective [12] .....	14
I.7 Conclusion .....	16
Chapitre II LES METHODES D'OPTIMISATION.....	
II.1 Introduction .....	17
II.2 Les problèmes d'optimisation .....	17
II.3 Les éléments d'optimisation.....	18
II.4 Optimisation combinatoire .....	19
II.5 La démarche heuristique .....	20
II.6 Les méta-heuristiques.....	20
II.7 Organisation générale.....	22
II.8 Applications .....	22
II.8.1 Méta- heuristique à recuit simulé.....	22
II.8.2 Les méta- heuristiques évolutionnaires/génétiques.....	24
II.8.3 Les méta- heuristiques éthologiques/colonies de fourmis.....	26

II.8.4 Méthode de recherche Tabou .....	27
II.9 Méthodes de pénalité.....	27
II.9.1 Principe général des méthodes de pénalité.....	27
II.9.2 Méthode de Fiasco et Mc Cornick .....	28
II.9.3 Méthodes de pénalité extérieure.....	28
II.9.4 Méthodes de pénalité intérieure .....	29
II.9.5 Méthode de pénalité mixte .....	29
II.10 Méthodes métaheuristique proposées.....	30
II.10.1 Algorithme de recherche de corbeau(CSA) : [28] .....	30
II.10.2 Implémentation de CSA pour l'optimisation.....	32
II.10.3 Algorithme d'optimisation de mouette (SOA) [29].....	34
II.10.4 Méthode Tunicate Swarm Algorithm (TSA): [30].....	39
II.10.5 Algorithme de la luciole(FFA)[31] .....	44
II.11 Conclusion.....	46
Chapitre III RESULTATS ET SIMULATION .....	
III.1 Introduction.....	47
III.2 PARTIE N°1 .....	48
III.2.1 Dispatching Economique dynamique avec les pertes variables (ELD).....	49
III.2.2 Dispatching Environnemental dynamique avec les pertes variables (ED) .....	53
III.2.3 Dispatching Economique Environnemental Combiné dynamique avec les pertes variables (CEED) .....	58
III.3 PARTIE N°2 .....	62
III.3.1 Dispatching Economique dynamique (ELD) avec les ressources des énergies renouvelables (RES).....	65
III.3.2 Dispatching Environnemental dynamique (ED) avec les ressources des énergies renouvelables (RES).....	69
III.3.3 Dispatching Economique Environnemental combinée dynamique (CEED) avec les ressources des énergies renouvelables (RES).....	73
III.4 Conclusion .....	78
Conclusion générale.....	79
Références bibliographiques.....	80

## Liste des figures

<b>Fig.I.1</b> Fonction coût du générateur .....	7
<b>Fig.I.2</b> Réseau électrique simplifié à n-nœud .....	10
<b>Fig.I.3</b> Modélisation d'une ligne .....	10
<b>Fig.II.1</b> : Eléments indispensable .....	18
<b>Fig.II.2</b> :Organigramme de l'état 1 dans CSA (a) $fl < 1$ et (b) $fl > 1$ . .....	33
<b>Fig.II.3</b> Migration et comportements d'attaque des mouettes. ....	36
<b>Fig.II.4</b> . Comportement des essaims de tuniciers dans l'océan profond. ....	39
<b>Fig.II.5</b> Vecteurs de position 3D des tuniciers .....	42
<b>Fig III.1</b> courbe pour cout à la charge=740 MW .....	53
<b>Fig. III.2</b> . Courbe pour Emission à la charge=740 MW .....	57
<b>Fig III.3</b> . courbe pour Cout+Emission à la charge=740 MW .....	62
<b>Fig III.4</b> . Architecture d'un micro-réseau Isolé .....	63
<b>Fig.III.5</b> : Coût total de production à la charge=740 MW .....	69
<b>Fig.III.6</b> :Courbe des émissions à la charge=250 MW .....	73
<b>Fig.III.7</b> :Courbe de cout total à la charge=250 MW.....	77

## Liste des Tableaux

<b>Tableau III.1</b> Données de réseau à 05 générateurs.....	48
<b>Tableau III.2</b> Coefficients de perte B (système à 5 unités) .....	48
<b>Tableau III.3</b> Charge demandé pendant 24 heures.....	48
<b>Tableau III.4</b> Les résultats obtenus par la méthode CSA.....	49
<b>Tableau III.5</b> Les résultats obtenus pour la méthode SOA .....	50
<b>Tableau III.6</b> Les résultats obtenus pour la méthode TSA.....	51
<b>Tableau III.7</b> Les résultats obtenus pour la méthode FFA .....	52
<b>Tableau III.8</b> Résultats comparatifs .....	52
<b>Tableau III.9</b> Les résultats obtenus pour la méthode Algorithme de recherche de corbeau .....	54
<b>Tableau III.10</b> Les résultats obtenus pour la méthode SOA .....	55
<b>Tableau III.11</b> Les résultats obtenus pour la méthode TSA.....	56
<b>Tableau III.12</b> Les résultats obtenus pour la méthode FFA .....	57
<b>Tableau III.13</b> Résultats comparatifs .....	57
<b>Tableau III.14</b> Les résultats obtenus pour la méthode Algorithme de recherche de corbeau .....	59
<b>Tableau III.15</b> Les résultats obtenus pour la méthode SOA .....	60
<b>Tableau III.16</b> Les résultats obtenus pour la méthode TSA.....	60
<b>Tableau III.17</b> Les résultats obtenus pour la méthode FFA .....	61
<b>Tableau III.18</b> :Résultats du dispatching économique-environnemental CEED .....	62
<b>Tableau III.19</b> Demande de charge pendant 24 heures (réseau à 03 unités) .....	63
<b>Tableau III.20</b> Données pour le système à 3 unités .....	64
<b>Tableau III.21</b> Les résultats obtenus par la méthode CSA.....	66
<b>Tableau III.22</b> Les résultats obtenus pour la méthode SOA .....	66
<b>Tableau III.23</b> Les résultats obtenus pour la méthode TSA.....	67
<b>Tableau III.24</b> Les résultats obtenus pour la méthode FFA .....	68
<b>Tableau III.25</b> Résultats comparatifs ELD.....	68
<b>Tableau III.26</b> Les résultats obtenus pour la méthode Algorithme de recherche de corbeau .....	70
<b>Tableau III.27</b> Les résultats obtenus pour la méthode SOA .....	71
<b>Tableau III.28</b> Les résultats obtenus pour la méthode TSA.....	71
<b>Tableau III.29</b> Les résultats obtenus pour la méthode FFA .....	72
<b>Tableau III.30</b> Résultats comparatifs ED .....	73
<b>Tableau III.31</b> Les résultats obtenus pour la méthode Algorithme de recherche de corbeau .....	74
<b>Tableau III.32</b> Les résultats obtenus pour la méthode SOA .....	75
<b>Tableau III.33</b> Les résultats obtenus pour la méthode TSA.....	76
<b>Tableau III.34</b> Les résultats obtenus pour la méthode FFA .....	77
<b>Tableau III.35</b> Résultats du dispatching économique-environnemental CEED.....	77

## Liste des symboles

$N_G$  : Nombre total des nœuds producteurs

$N_c$  : Nombre total des nœuds consommateurs

$P_{Gi}$  : Puissance active générée au nœud  $i$

$P_{Chj}$  : Puissance active consommée au nœud  $j$

$P_{Gi}^{\max}$  : Puissance active maximale générée au nœud  $i$

$P_{Gi}^{\min}$  : Puissance active minimale générée au nœud  $i$

$P_L$  : Les pertes actives totales dans le réseau

$Q_{Gi}$  : Puissance réactive générée au nœud  $i$

$Q_{Chj}$  : Puissance réactive consommée au nœud  $j$

$Q_{Gi}^{\max}$  : Puissance réactive maximale générée au nœud  $i$

$Q_{Gi}^{\min}$  : Puissance réactive minimale générée au nœud  $i$

$Q_L$  : Les pertes réactives totales dans le réseau

$E_i$  : Module de la tension au nœud  $i$

$E_i^{\max}$  : Module de la tension maximale au nœud  $i$

$E_i^{\min}$  : Module de la tension minimale au nœud  $i$

$T_j$  : Rapport de transformation du transformateur

$T_{ij}^{\max}$  : Valeur maximale du rapport de transformation du transformateur

$T_{ij}^{\min}$  : Valeur minimale du rapport de transformation du transformateur

$\theta_{ij}$  : La phase du rapport de transformation du transformateur

$\theta_{ij}^{\max}$  : Valeur maximale de la phase de transformation du transformateur

$\theta_{ij}^{\min}$  : Valeur minimale de la phase de transformation du transformateur

$S_{ij}^{\max}$  : Puissance transitée entre les nœuds  $i$  et  $j$

$S_{ij}^{\min}$  : Puissance maximale transitée entre les nœuds  $i$  et  $j$

$r_K$  : Est le coefficient de pénalité.

$B$  : Est une constante, tel que

$h(P_G)$  : Est la contrainte égalité

$P_{ch}$  : Est la puissance active totale consommée.

$P_L$  : Représente les pertes actives totales

$y_{ij}$  : Admittance de la branche  $i-j$

$g_{ij}$  : Conductance de la branche  $i-j$

$b_{ij}$  : Suceptance de la branche  $i-j$

$I$  : Vecteur complexe des courants nodaux injectées dans le réseau ;

$Y$  : Matrice complexe des admittances nodales complexe ;

$E$  : Vecteur complexe des tensions nodales.

$n$  : Désigne le nombre de nœuds dans le réseau ;

$S_i^*$  : Le conjugué de la puissance apparente injectée au nœud  $i$  ;

$E_i^*$  : Le conjugué de la tension au nœud  $i$  ;

$P_i$  : la puissance active injectée au nœud  $i$  ;

$Q_i$  : La puissance réactive injectée au nœud  $i$ .

$u_i$  : Coefficient de coût de  $i^e$  générateur en [ $\$/ MW^2h$ ]

$v_i$  : Coefficient de coût de  $i^e$  générateur en [ $\$/ MW h$ ]

$w_i$  : Coefficient de coût de  $i^e$  générateur en [ $\$/ h$ ]

$h_i$  : facteur de pénalité de l'unité de production.

: La puissance fournir par les ressource des énergies renouvelables (RES)  $P_{RES}$

$I(r_K, g)$ : est le terme de pénalisation intérieur.

$A_i$ : est une constante.

## Résumé

La contribution essentielle de ce mémoire est l'application de quatre techniques métaheuristiques à la résolution du problème de dispatching économique on basant sur l'algorithme de recherche des corbeaux (CSA), donc plusieurs objectifs ont été envisagés à savoir optimisation de l'écoulement de puissance statique avec et sans pollution, minimisation du cout de production de l'énergie électrique avec et sans pertes de transmission ,détermination de l'état optimal de chaque générateur interconnecté dans le réseau électrique durant vingt-quatre heures autrement dit résolution du problème de dispatching économique dynamique tout en traitant dans ce cas plusieurs types de contraintes et fonctions objectifs telles que la fonction de cout quadratique ,la fonction de cout nonlinéaire « non linéaire function with valve points effects », et les contraintes de limitation inférieures et supérieures des générateurs (contraintes Rampes) et le calcul de dispatching Economique / environnemental dynamique sur 24h avec des ressources des énergies renouvelable (une centrale solaire, une ferme éolienne et une batterie de stockage d'énergie) sans et avec pertes.

## Summary

The essential contribution of this dissertation is the application of four metaheuristic techniques to the resolution of the problem of economic dispatching based on the search algorithm of the crows (CSA), therefore several objectives were considered namely optimization of the flow of static power with and without pollution, minimization of the production cost of the " electrical energy with and without transmission losses, determination of the optimal state of each generator interconnected in the electrical network for twenty four hours in other words resolution of the problem of dynamic economic dispatching while dealing in this case with several types of constraints and objective functions such as the quadratic cost function, the non-cost function linear "non-linear function with valve points effects", and the lower and upper limitation constraints of the generators (Ramp constraints) and the dynamic economic / environmental dispatching calculation over 24 hours with renewable energy resources (a solar power plant, a wind farm and an energy storage battery) without and with losses.

## ملخص

المساهمة الأساسية لهذه الرسالة هي تطبيق أربع تقنيات ميتاهورستية لحل مشكلة التقسيم الاقتصادي في الانظمة الكهربائية بناءً على خوارزمية البحث عن الغربان (CSA) ، لذلك تم النظر في عدة أهداف وهي تحسين تدفق الطاقة الساكنة مع وبدون تلوث ، وتقليل تكلفة إنتاج " الطاقة الكهربائية مع وبدون خسائر الإرسال ، تحديد الحالة المثلى لكل مولد مترابط في الشبكة الكهربائية لمدة أربع وعشرين ساعة بمعنى آخر حل مشكلة الإرسال الاقتصادي الديناميكي أثناء التعامل في هذه الحالة مع عدة أنواع من القيود والوظائف الموضوعية مثل دالة التكلفة التربيعية ، دالة التكلفة غير الخطية "وظيفة غير خطية مع تأثيرات فتح

الصمام" ، وقيود الحد الأدنى والعليا للمولدات وحساب التوزيع الاقتصادي / البيئي الديناميكي على مدار 24 ساعة باستخدام موارد الطاقة المتجددة بدون خسائر أو مع وجود خسائر.

### Introduction générale

L'étude et l'optimisation de l'écoulement de puissance dans ces version statique et dynamique en présence des ressources des énergies renouvelables occupe dans nos jours une place déterminante dans la stratégie concurrentielle de l'entreprise, qui se trouve face à face à la libéralisation du secteur d'électricité donc face à une concurrence acharnée, soit pour les nouvelles restrictions liées à l'environnement qu'elle doit les respecter.

Dans cette logique, un faible cout de production représente un challenge pour les sociétés productrices, vue notamment aux prix chers de combustibles jours après jours, et les fardeaux supplémentaires liés au traitement des déchets nucléaires qui demande une technologie plus fine.

Dans un autre coté la complexité grandissante du réseau d'aujourd'hui vis-à-vis de sa taille qui comporte des centaines de jeux de barres et de centaines de milliers de kilomètres de lignes de transmission, vue aussi de sa structure très complexe (réseau interconnecté) ; tous ses facteurs exigent qu'une optimisation de la répartition optimale de puissance active générée constitue une nécessité impérative et un faible cout représente son but primordial.

Notons qu'une optimisation de cette répartition ne doit pas garantir un faible cout de production mais aussi doit accompagner des pertes de transport les plus minimales que possible (dispatching économique avec pertes) vue ces raisons le problème de l'optimisation devient non linéaire, et les méthodes conventionnelles (mathématiques ou classiques) se trouvent incapables de le résoudre, car elles se bloquent généralement dans des minimas locaux non global avec un temps de calcul considérable inutile de les appliquées en temps réel ( on line).

L'apparition des méthodes globales (méta-heuristiques) les constituées une alternatif, elle avoir une recherche stochastique probabiliste guidé par un processus algorithmique inspiré de la nature et la théorie de l'éthologie chez les sociétés des animaux et des insectes, donc elle n'a voire rien avec les propriétés mathématiques de la fonction envisagée continuité, dérivé ...etc.

En plus elle fait dépassement pour le calcul des opérateurs mathématiques tels que le gradient le Hessien ...etc qui présente les méthodes conventionnelles et qui nécessite un processus itératif important dont les boucles (if,while) constituant ainsi leur noyau, et qui traduit enfin par un temps de calcul important parfois fastidieux.

Les algorithmes évolutionnaires à savoir les algorithmes génétiques (AG) ,les algorithmes d'optimisation par essaims de particules (PSO) ,les réseaux de neurones ont une recherche stochastique qui cherchent l'optimum dans une région de l'espace c'est-à-dire chaque fois qu'on

## Introduction générale

---

teste un point de l'espace avec les points de ses alentours contrairement aux méthodes classiques qui font une recherche points par points par les opérateurs gradient, Hessien qui conduit généralement à un blocage local du processus algorithmique.

Dans ce mémoire on a appliqué l'algorithme de la luciole (Firefly Algorithm: FFA), l'algorithme d'optimisation de mouette (Seagull optimization algorithm: SOA), l'algorithme d'essaim de tuniciers (Tunicate Swarm Algorithm: TSA), l'algorithme de recherche de corbeaux (Crow Search Algorithm: CSA) pour la résolution du problème de dispatching économique dynamique tout en envisageant plusieurs critères et objectifs.

Le premier chapitre intitulé l'obstacle de la répartition optimale des puissances donne des définitions de base sur la notion de l'écoulement de puissance optimale, le dispatching économique et fonction objectif, une représentation de la fonction quadratique du coût est montrée, quelques définitions importantes pour les réseaux électriques, enfin on a terminé par une représentation des différentes variantes de la caractéristique du coût de combustible et leurs modèles mathématiques.

Le deuxième chapitre intitulé méthodes d'optimisation globales consacré à l'étude des algorithmes génétiques, l'algorithme d'optimisation par essaims de particules (PSO), l'algorithme recherche Tabou, l'algorithme de la luciole (Firefly Algorithm: FFA), l'algorithme d'optimisation de mouette (Seagull optimization algorithm: SOA), l'algorithme d'essaim de tuniciers (Tunicate Swarm Algorithm: TSA), l'algorithme de recherche de corbeaux (Crow Search Algorithm: CSA), en effet des définitions de base ont été abordées les opérateurs de sélection de mutation et de croisement sont bien discutés ainsi le modèle mathématique et le diagramme de processus de recherche algorithmique de chacun de ces méthodes sont présentés, la notion de voisinage est aussi abordée, l'élitisme sont clairement exposées. Le troisième chapitre test est application représente la partie simulation de notre mémoire. Il traite en premier lieu d'une manière détaillée le problème de dispatching économique statique des comparaisons entre les approches proposées sont largement expliquées.

Le problème de dispatching économique multi objectifs est présenté afin de réduire le taux d'émission des gaz toxique, différentes solutions et caractéristiques sont examinées. Le problème de dispatching économique dynamique est nettement expliqué, que se soit sans perte ou avec pertes, différents résultats obtenus par Les trois approches sont alors discutés et ce en appliquant plusieurs réseaux test et fonctions objectifs.

## **Chapitre I**

# **L'OBSTACLE DE LA REPARTITION OPTIMALE DES PUISSANCES**

## **I.1 Introduction**

Le XXe siècle a progressivement vu l'électricité s'imposer comme le vecteur énergétique privilégié Presque tous les domaines domestiques et industriels. Les principaux piliers de cette énergie, et Les réseaux de transport et de distribution d'électricité d'aujourd'hui représentent l'économie et Grande évolution technologique. C'est l'une des composantes essentielles du développement et Le développement des sociétés humaines, que ce soit en termes d'amélioration des conditions de vie ou de développement des activités industrielles. Leur rôle est de fournir aux utilisateurs le produit électrique en Coût inférieur dans des conditions de qualité et de sécurité satisfaisantes. Les réseaux électriques de divers pays du pourtour méditerranéen (Europe, Maghreb et Asie) sont Ils sont interconnectés en un seul système de transmission d'énergie électrique, verrouillé et enroulé. le Moins de panne pour ce système vaste et complexe, il est toujours alimenté et contrôlé en temps réel. Adapter la production à la demande d'électricité serait une catastrophe industrielle, sociale et économique. Ainsi, le fonctionnement d'un tel réseau est toujours un compromis entre les limitations Contraintes économiques et techniques liées à la physique des systèmes et contraintes de sécurité, Conditions d'utilisation et restrictions liées à la qualité du produit "Electricité". Depuis, le développement ne s'est pas arrêté Depuis près de 100 ans, le réseau n'est pas figé : il évolue pour répondre à la demande constante Se développer, se transformer progressivement en un marché concurrentiel, s'adapter aux contraintes De nouveaux défis environnementaux... C'est ainsi que les lignes de distribution sont enterrées, mais aussi Moyens de production décentralisés. [1]

## **I.2 Programme de marche des unités de production**

La puissance appelée par les consommateurs est variable et correspond à l'activité économique et sociale, avec son rythme, jours de travail/week-end, périodes d'activité/vacances, ainsi que les conditions climatiques (température, nébulosité, vent).

Afin de définir un programme de marche des unités de production permettant une exploitation sûre économique, il convient de prévoir la charge du réseau de manière aussi précise que possible en se basant sur les relevés des périodes récentes similaires, sur son évolution générale et sur les prévisions météorologiques. Plusieurs méthodes algorithmiques ont été développées. Il faudra arrêter certains groupes de production en période de basse charge et en démarrer pour couvrir les pointes de charge. Le choix du programme de marche des unités de production devra tenir compte du coût de fonctionnement des unités, mais aussi de leurs caractéristiques techniques, comme par exemple, les temps de démarrage et d'arrêt, avec les coûts associés.

La détermination du programme de marche des unités conduisant au coût le plus faible est un problème difficile car il faut intervenir des variables continues (la modulation de la puissance des unités en service) et discontinues (décisions de marche ou d'arrêt). Sachant qu'il n'y a pas de transition continue entre les deux puisqu'il existe une puissance minimale de fonctionnement des groupes de production. Parmi les méthodes de calcul utilisées figurent la programmation dynamique, la méthode de relaxation de Lagrange, etc. Les algorithmes utilisés par les programmes de calcul sont généralement complétés par des règles heuristiques.

### I.3 Enonce du problème

Nous considérons un système électro-énergétique alimenté par un ensemble de centrales possédant chacune plusieurs machines. Les frais du combustible nécessaire pour la production de la puissance électrique pour chaque machine sont une fonction monotone de la puissance appelée

$$F_i = F_i(P_{Gi}) \quad (I.1)$$

La répartition optimale de puissance sert à minimiser soit le coût total de la production de la puissance active générée ou les pertes totales de puissance en respectant les contraintes de type égalité et d'inégalité.

### I.4 Problème de la répartition optimale des puissances

Le problème de la répartition optimale des puissances actives peut être de la manière suivante [2] :

$$\text{Minimiser } F = \sum_{i=1}^{NG} F_i(P_{Gi}) \quad (I.2)$$

Sous les contraintes :

$$\sum_{i=1}^{NG} P_{Gi} - \sum_{j=1}^{NG} P_{Chi} - P_L = 0 \quad (I.3)$$

$$\sum_{i=1}^{NG} Q_{Gi} - \sum_{j=1}^{NG} Q_{Chi} - Q_L = 0 \quad (I.4)$$

$$P_{Gi\min} \leq P_{Gi} \leq P_{Gi\max} \quad (I.5)$$

$$Q_{Gi\min} \leq Q_{Gi} \leq Q_{Gi\max} \quad (I.6)$$

$$E_i\min \leq E_i \leq E_i\max \quad (I.7)$$

$$T_{ij\min} \leq T_{ij} \leq T_{ij\max} \quad (I.8)$$

$$Q_{ij}^{min} \leq Q_{ij} \leq Q_{ij}^{max} \quad (I.9)$$

$$|S_{ij}|^2 - |S_{ij}^{max}|^2 \leq 0 \quad (I.10)$$

Où:

$N_G$  : nombre total des nœuds producteurs ;

$N_c$ : nombre total des nœuds consommateurs ;

$P_{Gi}$  : puissance active générée au nœud  $i$  ;

$P_{Chj}$  : puissance active consommée au nœud  $j$  ;

$P_{Gi}^{max}$  : puissance active maximale générée au nœud  $i$  ;

$P_{Gi}^{min}$  : Puissance active minimale générée au nœud  $i$  ;

$P_L$ : les pertes actives totales dans le réseau ;

$Q_{Gi}$ : puissance réactive générée au nœud  $i$  ;

$Q_{Chj}$ : puissance réactive consommée au nœud  $j$  ;

$Q_{Gi}^{max}$  : puissance réactive maximale générée au nœud  $i$  ;

$Q_{Gi}^{min}$  : Puissance réactive minimale générée au nœud  $i$  ;

$Q_L$ : les pertes réactives totales dans le réseau ;

$E_i$  : module de la tension au nœud  $i$  ;

$E_i^{max}$  : Module de la tension maximale au nœud  $i$  ;

$E_i^{min}$  : Module de la tension minimale au nœud  $i$  ;

$T_j$ : rapport de transformation du transformateur ;

$T_{ij}^{max}$ : Valeur maximale du rapport de transformation du transformateur ;

$T_{ij}^{min}$ : Valeur minimale du rapport de transformation du transformateur ;

$\theta_{ij}$ : La phase du rapport de transformation du transformateur ;

$\theta_{ij}^{max}$ : Valeur maximale de la phase de transformation du transformateur ;

$\theta_{ij}^{min}$ : Valeur minimale de la phase de transformation du transformateur ;

$S_{ij}^{max}$ : Puissance transitée entre les nœuds i et j ;

$S_{ij}^{min}$ : Puissance maximale transitée entre les nœuds i et j ;

La résolution d'un tel problème avec toutes les contraintes reste difficile à réaliser pour des réseaux complexes comportant généralement beaucoup de nœuds et de lignes, d'interconnexion séparément la répartition optimale des puissances active et réactive, tandis que la seconde des puissances active et réactive simultanément. On s'est limité, dans notre cas, à la répartition optimale des puissances actives.. Ce n'est pas toujours rentable d'inclure toutes ces contraintes. Il est donc, nécessaire de simplifier le problème.

La complexité et la taille des problèmes posés ont permis d'élaborer deux méthodes distinctes de résolution. La première consiste à traiter

#### I.4.1 Présentation de la fonction objective

Dans le problème du dispatching économique, la fonction objective à minimiser est le coût total de production des groupes thermiques. Les courbes donnant le coût de production de chaque groupe (en Da/h ou MBtu/h) en fonction de la puissance qu'il débite (en MW) ont été déterminées expérimentalement (fig.1.1).

Chaque groupe va produire sa propre puissance selon une fonction coût convexe donnée par la fonction quadratique suivante [3]

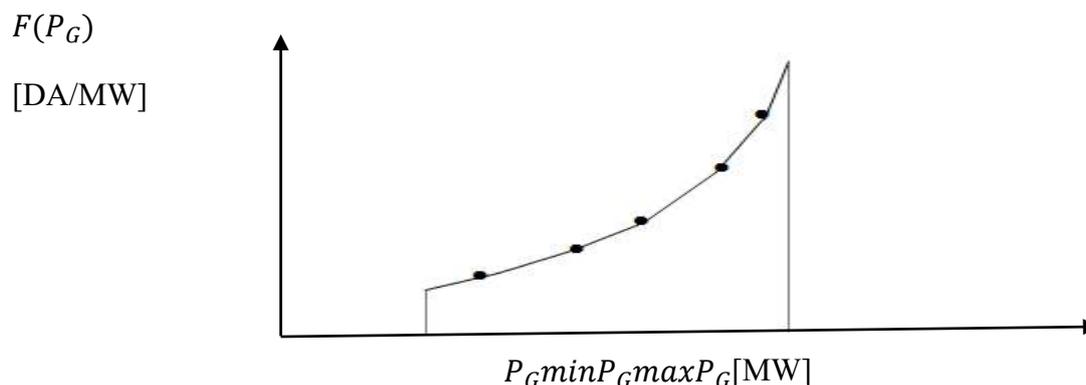
$$F_i(P_{Gi}) = a_i + b_i P_{Gi} + c_i P_{Gi}^2 \quad (I.11)$$

Où: Les coefficients  $a_i$ ,  $b_i$  et  $c_i$  sont numériquement connus.

La fonction du coût totale de production de l'énergie électrique exprimée en dinars par heure est donnée par l'expression suivante :

$$F = \sum_{i=1}^{NG} F_i(P_{Gi}) \quad (I.12)$$

Où:  $N_G$  est le nombre total de nœuds générateurs dans le réseau.



**Fig.I.1** Fonction coût du générateur

### I.4.2 Les contraintes

Les puissances actives générées optimales correspondant à un coût minimal, doivent obligatoirement respecter les contraintes de type égalité et inégalité. Les contraintes de sécurité expriment qu'en cas de perte d'un ou plusieurs ouvrages de production et/ou de transport, le réseau continue à fonctionner dans des limites admissibles.

#### I.4.2.1 Contraintes de type égalité

Le choix optimal des puissances générées doit obéir à l'équilibre statique de l'énergie dans le système électrique. Ces contraintes sont représentées par des équations non linéaires de l'écoulement de puissance. Il faut que la somme des puissances actives produites dans le réseau soit égale à la somme des puissances actives consommées, Plus les pertes.

#### I.4.2.2 Contraintes de type inégalité

Ces contraintes traduisent les limites de fonctionnement des ouvrages (groupe de production, lignes de transport, transformateurs...).

### I.4.3 Classification des variables

Généralement, le fonctionnement du système électrique peut être décrit en fonction de six variables pour chaque nœud considéré ( $P_{Gi}, Q_{Gi}, P_{Chi}, Q_{Chi}, E_i, \theta_i$ ). Ces variables sont généralement divisées en trois catégories :

- Variables incontrôlables : Ces variables peuvent être considérées comme des valeurs numériques constantes ou comme des paramètres. Les puissances active et réactive ( $P_{Chi}, Q_{Chi}$ ) demandées sont des exemples de ce type de variables.
- Variables de contrôle : Ce sont généralement les puissances active et réactive générées ( $P_{Gi}, Q_{Gi}$ ). On peut aussi, selon des cas, considérer les tensions aux nœuds de génération ou les

rappports de transformation des transformateurs avec régleur en charge, comme des variables de contrôle.

- Variables d'état : Ces variables ne sont pas directement contrôlées dans le processus d'optimisation. Leurs valeurs sont dépendantes du choix des variables de contrôle. Le vecteur des variables d'état peut inclure les tensions des jeux de barres de charge et les angles de phase des jeux de barres générateurs.

#### I.4.4 Présentation du problème

La première formulation, concerne la minimisation du coût totale de production de l'énergie électrique en fonction des puissances actives générées, en tenant compte de certaines contraintes. Le problème peut être posé de la manière suivante [4, 5, 6] :

Minimiser.

$$F = \sum_{i=1}^{NG} F_i(P_{Gi}) \quad (I.13)$$

Sous les contraintes :

$$\sum_{i=1}^{NG} P_{Gi} - \sum_{j=1}^{NC} P_{chj} - P_L = 0 \quad (I.14)$$

$$P_{Gi}min \leq P_{Gi} \leq P_{Gi}max \quad (I.15)$$

Les contraintes inégalités seront prises en considération dans l'algorithme de Recherche Tabou. Par contre, la contrainte égalité n'est pas prise en considération, pour cette raison qu'on va utiliser une méthode basée sur la transformation du problème original avec contraintes en un problème auxiliaire sans contraintes. Nous appliquons la pénalisation, on aura une fonction sans contraintes ( $F_m$ ), cette dernière tient compte de la fonction coût  $F$  et la contrainte égalité.

$$F_m(P_G, r_K) = \sum_{i=1}^{NG} F_i(P_{Gi}) + \frac{1}{r_K} \cdot B \cdot h^2(P_G) \quad (I.16)$$

Où:

$r_K$ : est le coefficient de pénalité.

$B$ : est une constante, tel que

$$\begin{cases} B > 0, & \text{si } h(P_G) \neq 0 \\ B = 0, & \text{si } h(P_G) = 0 \end{cases} \quad (I.17)$$

$h(P_G)$ : est la contrainte égalité

$$h_{(P_G)} = \sum_{j=1}^{NG} P_{Gi} - P_{Ch} - P_L = 0 \quad (I.18)$$

$P_{ch}$ : est la puissance active totale consommée.

$P_L$ : représente les pertes actives totales dans le réseau, elles sont déterminées par le calcul de l'écoulement statique des charges

## I.5 Problème de l'écoulement de puissance

### I.5.1 Introduction

Le calcul de l'écoulement de puissance (load flow ou power flow) dans un réseau électrique est un outil fondamental tant en contrôle opérationnel qu'en planification. Il a pour but de déterminer, en régime triphasé permanent, les tensions en tout point du réseau et les puissances actives et réactive transitant sur toutes les lignes. Ceci explique, une littérature riche qui est consacrée à ce sujet et relatant depuis le début de l'ère des ordinateurs, l'évolution de la question des points de vue mathématique et informatique [7, 8, 9, 10].

### I.5.2 Modélisation du réseau

Un état du réseau est en lieu défini par sa topologie, c'est-à-dire, d'une part, la liste des ouvrages en service à l'instant étudié, et d'autre part, les connexions entre les ouvrages. On distingue la topologie élémentaire (ou détaillée) et la topologie nodale.

La topologie nodale peut être par un graphe dont les sommets sont les nœuds électriques, et les arêtes les ouvrages du réseau (lignes, câbles, transformateurs).

Dans les calculs de la répartition des charges, le réseau est représenté en topologie nodale.

Soit un réseau électrique simplifié est représenté par la figure I.2 :

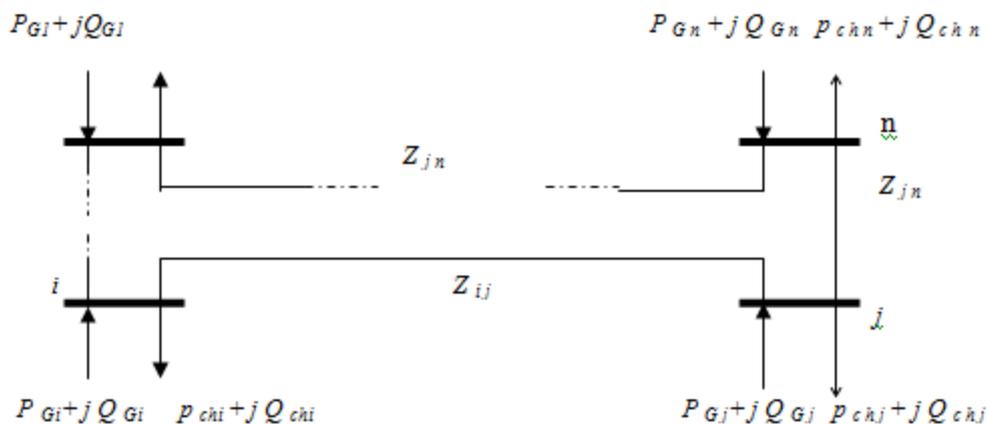


Fig.I.2 Réseau électrique simplifié à n-nœud

**I.5.2.1 Modélisation de la ligne**

Une ligne reliant deux nœuds  $i$  et  $j$ , est modélisée par un schéma en  $\pi$  dont l'impédance série est (figure I.3) :

$$Z_{ij} = r_{ij} + jX_{ij} \tag{I.19}$$

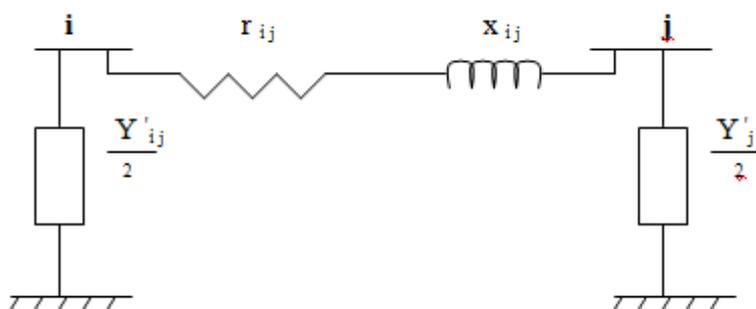


Fig.I.3 Modélisation d'une ligne

Pour utiliser la topologie nodale, nous devons transformer les impédances des branches du réseau en admittances ; pour cela, nous posons :

$$y_{ij} = \frac{1}{Z_{ij}} = \frac{1}{R_{ij} + jX_{ij}} = \frac{R_{ij}}{R_{ij}^2 + X_{ij}^2} - j \frac{X_{ij}}{R_{ij}^2 + X_{ij}^2} \tag{I.20}$$

Où:

$$y_{ij} = g_{ij} - jb_{ij} \tag{I.21}$$

Et :

$$\begin{cases} g_{ij} = \frac{R_{ij}}{R_{ij}^2 + X_{ij}^2} \\ b_{ij} = \frac{X_{ij}}{R_{ij}^2 + X_{ij}^2} \end{cases} \quad (\text{I.22})$$

$y_{ij}$  : admittance de la branche  $i-j$

$g_{ij}$  : conductance de la branche  $i-j$

$b_{ij}$  : susceptance de la branche  $i-j$

### I.5.2.2 Matrice admittance nodale

Les éléments de la matrice admittance nodale  $Y$  seront :

- Les éléments diagonaux  $Y_{ii}$  :

$$Y_{ii} = \sum_{j=1}^n \left( y_{ij} + \frac{y_{ij}}{2} \right) \quad (\text{I.23})$$

- Les éléments non diagonaux  $Y_{ij}$  :

$$Y_{ii} = -Y_{ij} \quad (\text{I.24})$$

### I.5.3 Calcul de l'écoulement des puissances par la méthode de Gauss- Seidel

Le calcul de l'écoulement de puissance en régime permanent établi se base sur le système d'équations linéaires suivant :

$$I = Y \cdot E \quad (\text{I.25})$$

Où :

$I$  : vecteur complexe des courants nodaux injectés dans le réseau ;

$Y$  : matrice complexe des admittances nodales complexe ;

$E$  : vecteur complexe des tensions nodales.

Si le réseau admet  $N$  nœuds ;  $I$  et  $E$  auront  $N$  composantes complexes et  $Y$  sera une matrice complexe de dimension  $N \times N$ . Les éléments de  $y_{ij}$  seront calculés à partir des caractéristiques des composants du réseau.

Cette méthode dérivant de la méthode itérative de Gauss permet de résoudre un système d'équations non linéaires en utilisant la matrice admittance. On suppose initialement des tensions pour tous les nœuds excepté le nœud de bilan (nœud pris comme nœud de référence) où la tension est maintenue constante [7, 11].

On peut exprimer les courants pour chaque nœud comme suit :

$$I_i = \frac{S_i^*}{E_i^*} = \frac{P_i - jQ_i}{E_i} \quad i=1.2.....n \quad (I.26)$$

Où:

$n$  : désigne le nombre de nœuds dans le réseau ;

$S_i^*$  : le conjugué de la puissance apparente injectée au nœud  $i$  ;

$E_i^*$  : le conjugué de la tension au nœud  $i$  ;

$P_i$  : la puissance active injectée au nœud  $i$  ;

$Q_i$  : la puissance réactive injectée au nœud  $i$ .

En remplaçant l'équation (I.25) dans (I.26), on aura :

$$I_i = \frac{P_i - jQ_i}{E_i^*} = Y_{ii} \cdot E_i + \sum_{\substack{j=1 \\ j \neq i}}^n Y_{ij} \cdot E_j \quad i \neq s \quad (I.27)$$

Où:  $s$  est le numéro du nœud de référence.

L'expression de la tension pour chaque nœud est :

$$E_i = \frac{1}{Y_i} \left[ \frac{P_i - jQ_i}{E_i^*} - \sum_{j=1}^{i-1} Y_{ij} \cdot E_j - \sum_{j=i+1}^n Y_{ij} \cdot E_j \right] \quad (I.28)$$

En posant :

$$\begin{cases} KL_i = \frac{P_i - jQ_i}{Y_{ij}} \\ YL_{ij} = \frac{Y_{ij}}{Y_{ij}} \end{cases} \quad (I.29)$$

D'où l'expression finale de la tension pour chaque nœud :

$$E_i^{K+1} = \frac{KL_i}{[E_i^K]^*} - \sum_{j=1}^{i-1} YL_{ij} \cdot E_j^{K+1} - \sum_{j=i+1}^n YL_{ij} \cdot E_j^K \quad i \neq s \quad (I.30)$$

Pour accélérer la convergence de la méthode, on introduit un facteur d'accélération

$\alpha$  ( $1 \leq \alpha \leq 1.4$ )

$$E_i^{K+1} = E_i^K + \alpha \cdot \Delta E_i^K \quad (I.31)$$

Où:

$$\Delta E_i^K = E_i^{K+1} - E_i^K \quad (\text{I.32})$$

### Algorithme de Gauss-seidel

#### Etape 1

Formation de la matrice admittance [Y]

#### Etape 2

Estimation des valeurs initiales des tensions nodales  $E_i^0 \ i=1, \dots, n$

#### Etape 3

Détermination des paramètres  $KL_i$  et  $YL_{ij} \begin{cases} i = 1, \dots, n \\ j = 1, \dots, n \end{cases}$

Initiation des itérations  $k=0$

#### Etape 4

Calcul itératif des tensions pour chaque nœud suivant la relation :

$$E_i^{K+1} = \frac{KL_i}{[E_i^K]^*} - \sum_{j=1}^{i-1} YL_{ij} \cdot E_j^{K+1} - \sum_{j=i+1}^n YL_{ij} \cdot E_j^K \quad (\text{I.33})$$

On calcule l'écart entre les valeurs d'une même tension trouvée aux itérations qui se suivent :

$$\Delta E_i^K = E_i^{K+1} - E_i^K \quad (\text{I.34})$$

On introduit le facteur d'accélération  $\alpha$  pour réduire le nombre d'itérations.

$$E_i^{K+1} = E_i^K + \alpha \cdot \Delta E_i^K \quad (\text{I.35})$$

#### Etape 5

Une fois, le test de convergence est vérifié ( $Max \Delta E_i^K \leq \varepsilon$ ), les valeurs des tensions de la dernière itération sont retenues, on calcule :

- Les puissances transitées :

$$S_{ij}^* = E_i^* (E_i - E_j) y_{ij} + E_i^* \cdot E_i \cdot \frac{y'_{ij}}{2} \quad (\text{I.36})$$

- Les puissances injectées :

$$S_i = \sum_{j=1}^n S_{ij} \quad (\text{I.37})$$

- Les pertes :

$$S_L = \sum_{i=1}^n S_i \quad (\text{I.38})$$

Sinon aller à l'étape 4

### I.6 Formulation de la fonction objective [12]

Nous considérons un système électro-énergétique alimenté par un ensemble de centrales possédant chacune plusieurs machines. Les frais du combustible nécessaire pour la production de la puissance électrique pour chaque machine sont une fonction monotone de la puissance. Les courbes donnant le coût de production de chaque groupe (en \$/h ou MBtu/h) en fonction de la puissance qu'il débite (en MW) sont déterminées expérimentalement.

Chaque groupe va produire sa propre puissance selon une fonction coût (ELD) convexe.

La fonction du coût totale de production de l'énergie exprimée en unité de monnaie soit en dinars ou en dollar par heure est donnée par la fonction quadratique suivante:

$$F(P) = \sum_{t=1}^{24} \sum_{i=1}^g \{u_i P_i^2(t) + v_i P_i(t) + w_i\} \quad (\text{I.39})$$

Les coefficients  $u_i$ ,  $v_i$  et  $w_i$  sont numériquement connus.

Chaque centrale va produire sa propre puissance selon une fonction d'émission convexe donnée par la fonction quadratique. Le problème consiste à minimiser la fonction d'émission (ED) totale de gaz pour la production d'énergie électrique qui est donnée par l'expression suivante en kg/h:

$$E(P) = \sum_{t=1}^{24} \sum_{i=1}^g \{x_i P_i^2(t) + y_i P_i(t) + z_i\} \quad (\text{I.40})$$

Où les coefficients  $x_i$ ,  $y_i$  et  $z_i$  sont numériquement connus.

Le calcul de dispatching économique 'DE' doit aussi respecter les contraintes environnementales. Le dispatching économique et environnemental combinée (CEED) a pour objectif, non seulement de produire à un coût de combustible réduit mais aussi de réduire les émissions de gaz de ces combustibles.

L'étude du dispatching économique-environnemental consiste à la minimisation simultanée des deux fonctions données par (I.41). Nous transformons donc le problème d'optimisation bi-objectifs en un problème d'optimisation mono-objectif, en introduisant un facteur de pénalité de prix. Ce facteur  $hi$  est défini comme étant le rapport entre le coût maximal et les

émissions maximales de chaque générateur. Après détermination de ce facteur, nous pouvons représenter la fonction décrivant le Dispatching Economique-Environnemental (CEED) par l'équation suivante en \$/h:

$$C(P) = \sum_{t=1}^{24} \sum_{i=1}^g [\{u_i P_i^2(t) + v_i P_i(t) + w_i\} + h_i \times \{x_i P_i^2(t) + y_i P_i(t) + z_i\}] \quad (I.41)$$

: facteur de pénalité de l'unité de production.  $h_i$  ou

### Formulation du problème Dispatching économique (ELD) de Micro réseau isolé en présence de la ressource des énergies renouvelables (RES):

Dans cette partie, nous avons étudié un micro-réseau isolé avec l'intégration des parcs éoliens et des système photovoltaïque

(PV) pour minimiser les coûts de carburant, d'émissions et également pour augmenter l'efficacité et maintenir une alimentation électrique ininterrompue. Le coût d'exploitation

et de maintenance du parc éolien et du système photovoltaïque est de 0,016 \$ / kW investi à une avec un taux de 9% pendant 20 ans. Le ratio de coût d'investissement pour établir la puissance est de 5000 \$/kW pour le système PV et de 1400 \$/kW pour le parc éolien.

Ainsi, la fonction de coût de PV devient  $F_{PV} = 547,7483 * P_{PV}$  et la fonction de coût du éolien est  $F_{WIND} = 153,3810 * P_{WIND}$ . (I.42)

Par conséquent, avec l'intégration de RES, la fonction dispatching économique devient:

$$ELD(P) = \sum_{i=1}^g (a_i P_i^2 + b_i P_i + c_i) + 547.7483 * P_{PV} + 153.3810 * P_{WIND} \quad (I.43)$$

Et avec l'intégration de RES, la fonction de cout des émissions économique combinée ce réécrire sous la forme suivante :

$$ELD(P) = \sum_{i=1}^g [(a_i P_i^2 + b_i P_i + c_i) + h_i (l_i P_i^2 + m_i P_i + n_i)] + 547.7483 * P_{PV} + 153.3810 * P_{WIND} \quad (I.44)$$

La fonction de coût de la batterie peut être calculée comme suit:

$$(I.45) P_{Batterie} * = 119(P_{Batterie})F$$

Sous les contraintes suivantes:

$$P_{i,min} \leq P_i \leq P_{i,max} \quad (I.46)$$

$$P_{RES,max} \quad (I.47) \leq P_{RES} \leq P_{RES,min}$$

$$(I.48) \quad g_{...} = 1,2,3,i P_{RES} + P_i = P_{charge}$$

Avec :

: La puissance générée par les générateurs.  $P_i$

: La puissance fournir par les ressource des énergies renouvelables (RES)  $P_{RES}$

### I.7 Conclusion

Dans ce chapitre, nous avons décrit le problème de l'écoulement de puissance optimale. Plusieurs critères sont modélisés et pris en considération à savoir le critère économique, environnemental (simple et dynamique) économique/environnemental et le coût d'investissement des ressource des énergies renouvelables (RES), et le dispatching économique avec effet d'ouverture de vannes.

Nous proposerons dans le chapitre suivant, l'application de l'algorithme de recherche de Crow Serach Algorithm (CSA)' et en l'on compare avec trois algorithmes à savoir 'corbeaux (CSA), ( TSA),( SOA) et l'algorithme (FFA), sur différents réseaux test en tenant compte de tous les critères cités.

## **Chapitre II**

# **LES METHODES D'OPTIMISATION**

## II.1 Introduction

Parmi les problèmes rencontrés par les chercheurs, les problèmes d'optimisation occupent à notre époque, une place prépondérante. L'objectif principal de l'optimisation est de déterminer une solution qui minimise (ou maximise) une fonction, appelée dans la littérature fonction objective ou fonction d'adaptation tout en vérifiant un certain nombre de contraintes. Cette fonction correspond à une relation algébrique entre une ou plusieurs variables de sortie du système étudié que l'on appelle " critères". Pratiquement, toutes les méthodes d'optimisation opèrent par itération successive, à partir d'une estimation initiale  $x^0$  qui est progressivement améliorée. La différence entre les méthodes réside dans le choix de la procédure adoptée pour passer d'une estimation  $x^i$  à la nouvelle estimation  $x^{k+1}$ .

La méthode de programmation non linéaire a été la première méthode à connaître un essor remarquable, attirant ainsi l'attention des chercheurs et des ingénieurs ; les solutions qu'elles offrent couvrent un large champ d'application. Dans les années quatre-vingt. Le développement rapide de l'outil informatique a permis d'élaborer d'autres méthodes [2] :

- Des méthodes de programmations successives.
- Des méthodes de lagrangien augmentée.
- Des méthodes de programmation quadratique, mentionnons les méthodes de Newton et quasi-Newton.

Les spécialistes de l'optimisation combinatoire ont ensuite orienté leur recherche vers le développement des méthodes stochastiques tel que : le recuit simulé, la recherche taboue et les algorithmes évolutionnistes. Depuis quelques années, un nombre croissant de méthodes d'optimisation ont été proposées inspirées d'un phénomène naturel ou bien biologique. Actuellement, poussées par les performances générales de tels algorithmes, un nombre croissant d'études proposent ce type d'approche.

## II.2 Les problèmes d'optimisation

La résolution des problèmes d'optimisation est utilisée dans un grand nombre de domaines [14, 31]. A l'origine, ce sont les militaires qui se sont intéressés à ces questions au cours de la seconde guerre mondiale. C'était en fait un nouveau domaine de recherche en mathématiques appliquées qui a vu le jour avec la recherche opérationnelle. Le développement de l'informatique a ouvert de nouveaux horizons à la résolution de ces problèmes, et a permis un élargissement massif des champs d'application de ces techniques.

La résolution d'un problème d'optimisation et un problème complexe, carde nombreux facteurs interviennent et interagissent entre eux. Néanmoins, l'optimisation appliquée au domaine d'électrotechnique permet de résoudre des problèmes qui étaient insolubles auparavant et aboutit souvent à des solutions originales.

Dans ce chapitre, nous présentons différentes méthodes de résolution. L'ensemble de ces méthodes est tellement vaste qu'il est impossible de tout exposer. Ainsi, nous présentons les principales méthodes de résolution.

### II.3 Les éléments d'optimisation

L'optimisation est une des mathématiques consacrées à l'étude du (ou des) minimum(s)/maximum(s) d'une fonction à une ou plusieurs variables sur un certain domaine de définition, de l'étude de leur existence à leur détermination, en général par la mise en œuvre d'un algorithme et par suite un programme. Pour mener à bien une opération, plusieurs éléments sont indispensables et conditionnent la solution trouvée. La figure suivante présente les quatre éléments essentiels à la résolution d'un problème d'optimisation.

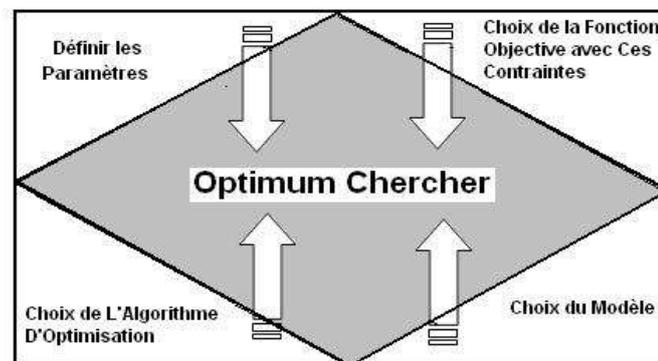


Fig.II.1 : Eléments indispensable

En général, un grand nombre de paramètres sont indispensables, il faut être capable de définir les paramètres utiles à l'optimisation. Certains paramètres ont une influence sur la fonction choisie, d'autres pas. Etant donné le coût des simulations, seuls les paramètres influents sont à retenir :

**Une fonction objective** : définit l'objectif à atteindre. La définition de cette fonction est en fait un problème délicat. Car le problème est formulé en un problème d'optimisation par l'intermédiaire de la fonction objective. C'est elle qui est au centre de l'optimisation, c'est donc elle que dépend la pertinence de la solution.

**Un modèle** : précis, robuste et malléable du système étudié est indispensable. Ce modèle doit être utilisable sur un domaine d'étude le plus large possible.

**Un algorithme d'optimisation** : permet de trouver la solution.

Différentes méthodes d'optimisation existent et en sont présentées.

## II.4 Optimisation combinatoire

L'optimisation combinatoire [32] occupe une place très importante en recherche opérationnelle, en mathématiques discrètes et en informatique. Son importance se justifie d'une part par la grande difficulté des problèmes d'optimisation et d'autre part par de nombreuses applications pratiques pouvant être formulées sous la forme d'un problème d'optimisation combinatoire. Bien que les problèmes d'optimisation combinatoire soient souvent faciles à définir, ils sont généralement difficiles à résoudre. En effet, la plupart de ces problèmes appartiennent à la classe des problèmes *NP-difficiles* et ne possèdent donc pas à ce jour de solution algorithmique efficace valable pour toutes les données. L'optimisation combinatoire est minimisée (ou maximisée) une fonction souvent appelée fonction coût, d'une ou plusieurs variables soumises à des contraintes. Le sujet de l'optimisation combinatoire dans un domaine discret. Il faut trouver parmi toutes les possibilités, souvent en nombre fini, la possibilité optimale. Ceci paraît facile mais devient infaisable dès que la taille du problème est suffisamment grande. La taille pour laquelle la recherche d'un optimum devient infaisable est petite, très souvent plus petite que la taille des problèmes pratiques. En général, la difficulté d'un problème grandit très vite avec le nombre des variables. Il n'est pas alors faisable d'examiner toutes les possibilités.

Les méthodes d'optimisation peuvent être réparties en deux catégories :

1. Méthodes exactes.
2. Méthodes approchées.

Les méthodes exactes fournissent systématiquement une solution (optimale) au problème traité si une telle solution existe. Dans le cas contraire, ce type de méthode permet d'affirmer qu'il n'existe pas de solution au problème traité. Les méthodes approchées fournissent une solution approchée au problème traité. Elles sont en général conçues de manière à ce que la solution obtenue puisse être située par rapport à la valeur optimale : de telles méthodes permettent d'obtenir des bornes inférieures ou supérieures de la valeur optimale tel que :

- 1- Méthodes Heuristiques ;
- 2- Méthodes Méta heuristiques.

### **II.5 La démarche heuristique**

L'heuristique [14,33] est une méthode, une technique ou un critère de guidage ou de décision, en général empirique ou obtenu par approximation, permettant de choisir la voie la plus prometteuse de recherche de la solution au problème posé, ou d'éliminer les voies les moins intéressantes, sans garantie sur la validité ou la précision de l'information ainsi fournie.

Entrer dans le domaine des heuristiques, c'est se départir d'emblée des schémas classiques. En effet, alors que la démarche classique mathématique est centrée sur l'objet de l'étude, sur la compréhension de sa structure et de sa logique, la démarche heuristique repousse le problème lui-même au rang d'illustration pour dégager des schémas plus généraux et donc originaux.

Les heuristiques disposent d'une simplicité et donc d'une rapidité dans leur exécution plus élevée que les algorithmes classiques. Ces règles s'appliquant à un ensemble particulier la recherche des faits se voit simplifiée et accélérée (moins de possibilités).

D'où une analyse des situations améliorée. Mais une méthode heuristique trop simplifiée ou au contraire trop générale peut conduire à des biais cognitifs, générant des erreurs de décision.

L'utilisation de plus de ces éléments simples (les heuristiques) afin de créer des éléments plus complexes (les méta- heuristiques) permet donc de réduire considérablement l'ensemble de recherche global de l'algorithme. L'une de leurs caractéristiques principales et à première vue défaut, dont héritent également les méta- heuristiques, est qu'ils peuvent dans certains cas ne pas proposer de solution optimale au problème. Mais au résultat s'y approchant d'assez près pour qu'il soit considéré comme correct, on parle alors de garantie de performance.

### **II.6 Les méta-heuristiques**

Les méta- heuristiques sont apparues dans les années 1980 et forment une famille d'algorithmes d'optimisation visant à résoudre des problèmes d'optimisation difficile, pour lesquels on ne connaît pas de méthode classique plus efficace. Elles sont généralement utilisées comme des méthodes génériques pouvant optimiser une large gamme de problèmes différents, sans nécessiter de changements profonds dans l'algorithme employé [14,15,34,16]. Etymologiquement parlant ce mot est composé dans un premier temps du préfixe méta qui signifie «au-delà » ou «plus haut» en grec puis de « heuristique » qui signifie « trouver ».

Cette décomposition permet facilement de comprendre le but premier de ces algorithmes : trouver des solutions à des problèmes en utilisant plusieurs (méta) heuristiques. Méta- heuristiques utilisent des processus aléatoires comme moyens de récolter de l'information et de faire face à des problèmes comme l'explosion combinatoire. En plus de cette base stochastique, les méta-heuristiques sont généralement itératives, c'est-à-dire qu'un même schéma de recherche est appliqué plusieurs fois au cours de l'optimisation, et directes, c'est-à-dire qu'elles n'utilisent pas l'information du gradient de la fonction objectif. Elles tirent en particulier leur intérêt de leur capacité à éviter les optima locaux, soit en acceptant une dégradation de la fonction objective au cours de leur progression, soit en utilisant une population de points comme méthode de recherche.

Les méta- heuristiques, du fait de leur capacité à être utilisées sur un grand nombre de problèmes différents, se prêtent facilement à des extensions. Pour illustrer cette caractéristique, citons notamment :

- L'optimisation multi objectif (dites aussi multicritère) [17], où il faut optimiser plusieurs objectifs contradictoires. La recherche vise alors non pas à trouver un optimum global, mais un ensemble d'optima « au sens de Pareto » formant la « surface de compromis » du problème.
- L'optimisation multimodale, où l'on cherche un ensemble des meilleurs optima globaux et/ou locaux.
- L'optimisation de problèmes bruités, où il existe une incertitude sur le calcul de la fonction objectif. Incertitude dont il faut alors tenir compte dans la recherche de l'optimum.
- L'optimisation dynamique, où la fonction objectif varie dans le temps. Il faut alors approcher au mieux l'optimum à chaque pas de temps.
- La parallélisation, où l'on cherche à accélérer la vitesse de l'optimisation en répartissant la charge de calcul sur des unités fonctionnant de concert. Le problème revient alors à adapter les méta- heuristiques pour qu'elles soient distribuées.
- L'hybridation, qui vise à tirer parti des avantages respectifs de méta- heuristiques différentes en les combinant [17,15].

Enfin, la grande vitalité de ce domaine de recherche ne doit pas faire oublier qu'un des intérêts majeurs des méta- heuristiques est leur facilité d'utilisation dans des problèmes concrets. L'utilisateur est généralement demandeur de méthodes efficaces permettant d'atteindre un optimum avec une précision acceptable dans un temps raisonnable. Un des enjeux de la conception des

métas- heuristiques est donc de faciliter le choix d'une méthode et de simplifier son réglage pour l'adapter à un problème donné.

## II.7 Organisation générale

D'une manière générale, les méta- heuristiques s'articule autour de trois notions [17] :

**Diversification /exploration** : désigne les processus visant à récolter de l'information sur le problème optimisé.

**L'intensification/exploitation** : vise à utiliser l'information déjà récoltée pour définir et parcourir les zones intéressantes de l'espace de recherche.

**La mémoire** : est le support de l'apprentissage, qui permet à l'algorithme de ne tenir compte que des zones où l'optimum global est susceptible de se trouver, évitant ainsi les optimums locaux. Les métas- heuristiques progressent de façon itérative, en alternant des phases d'intensification, de diversification et d'apprentissage. L'état de départ est souvent choisi aléatoirement, l'algorithme se déroulant ensuite jusqu'à ce qu'un critère d'arrêt soit atteint.

## II.8 Applications

Les métas- heuristiques sont souvent inspirés par des systèmes naturels, qu'ils soient pris en physique (les méthodes de voisinage comme le recuit simulé et la recherche tabou), en biologie de l'évolution (les algorithmes évolutifs comme les algorithmes génétiques et les stratégies d'évolution) ou encore en étiologie (les algorithmes de colonies de fourmis).

### II.8.1 Méta- heuristique à recuit simulé

La méthode de recuit simulé s'inspire du processus de recuit physique [15,18]. Ce processus utilisé en métallurgie pour améliorer la qualité d'un solide cherche un état d'énergie minimale qui correspond à une structure stable du solide. Les origines du recuit simulé remontent aux expériences réalisées par Metropolis et al dans les années 50 pour simuler l'évolution d'un tel processus de recuit physique. Metropolis et al utilisent une méthode stochastique pour générer une suite d'états successifs du système en partant d'un état initial donné.

Tout nouvel état est obtenu en faisant subir un déplacement (une perturbation) aléatoire à un atome quelconque. L'utilisation d'un tel processus du recuit simulé pour résoudre des problèmes d'optimisation combinatoire a été reportée dans, Le recuit simulé peut être vu comme une version étendue de la méthode de descente. Le processus du recuit simulé répète une procédure itérative qui cherche des configurations de coût plus faible tout en acceptant de manière contrôlée des

configurations qui dégradent la fonction de coût. A chaque nouvelle itération, un voisin de la configuration courante est généré de manière aléatoire. Selon les cas, ce voisin sera soit retenu pour remplacer celle-ci, soit rejeté. Si ce voisin est de performance supérieure ou égale à celle de la configuration courante, il est systématiquement retenu. Dans le cas contraire, il est accepté avec une probabilité qui dépend de deux facteurs : d'une part l'importance de la dégradation (Les dégradations plus faibles sont plus facilement acceptées) d'autre part un paramètre de contrôle, la température (une température élevée correspond à une probabilité plus grande d'accepter des dégradations). La température est contrôlée par une fonction décroissante qui définit un schéma de refroidissement. Les deux paramètres de la méthode définissent la longueur des paliers et la fonction permettant de calculer la suite décroissante des températures. En pratique, l'algorithme s'arrête et retourne la meilleure configuration trouvée lorsque aucune configuration voisine n'a été acceptée pendant un certain nombre d'itérations à une température ou lorsque la température atteint la valeur zéro.

La performance du recuit simulé dépend largement du schéma de refroidissement utilisé. De nombreux schémas théoriques et pratiques ont été proposés. De manière générale, les schémas de refroidissement connus peuvent être classés en trois catégories :

- Réduction par paliers : chaque température est maintenue égale pendant un certain nombre d'itérations, et décroît ainsi par paliers.
- Réduction continue : la température est modifiée à chaque itération.
- Réduction non- monotone : la température décroît à chaque itération avec des augmentations occasionnelles.

Il existe des schémas qui garantissent la convergence asymptotique du recuit simulé. En pratique, on utilise des schémas relativement simples même s'ils ne garantissent pas la convergence de l'algorithme vers une solution optimale.

Le recuit simulé constitue, parmi les méthodes de voisinage, l'une des plus anciennes et des plus populaires. Il a acquis son succès essentiellement grâce à des résultats pratiques obtenus sur de nombreux problèmes NP- difficiles. La preuve de convergence a également contribué à cette popularité, bien que cette preuve n'ait pas de portée en pratique.

## II.8.2 Les méta- heuristiques évolutionnaires/génétiques

### II.8.2.1 Origines

Les algorithmes génétiques appartiennent à une famille d'algorithmes appelés méta-heuristique dont le but est d'obtenir une solution approchée [19,20], en un temps correct, à un problème d'optimisation, lorsqu'il n'existe pas de méthode exacte pour le résoudre. Les algorithmes génétiques utilisent la notion de sélection naturelle développée par le scientifique Charles Darwin au XIXème siècle.

Dans cette théorie, une population d'individus évolue grâce au mécanisme de la reproduction sexuée. Les individus les plus adaptés à leur milieu se reproduisent plus que les autres, favorisant les caractères les plus adaptés. Ainsi une girafe avec un cou plus long que les autres aura accès à plus de nourriture, et aura donc plus de chances de survivre et de se reproduire. Ses descendants auront un cou plus long, et en moyenne la population de girafe aura un cou plus long.

L'utilisation d'algorithmes génétiques dans la résolution de problèmes est à l'origine des recherches de John Holland dès 1960. La nouveauté introduite a été la prise en compte de l'opérateur *crossing over* en complément des mutations, et c'est cet opérateur qui permet le plus souvent de se rapprocher de l'optimum d'une fonction en combinant les gènes contenus dans les différents individus de la population [17,21, 22].

### II.8.2.2 Principe

Les algorithmes génétiques classiques introduits par Holland s'appuient fortement sur un codage universel sous forme de chaînes 0/1 de longueur fixe et un ensemble d'opérateurs génétiques : les sélections, les *crossing over* ou recombinaison et les mutations. Un individu sous ce codage, appelé *chromosome*, représente une configuration du problème. Les opérateurs «génétiques» sont définis de manière à opérer aléatoirement sur un ou deux individus sans aucune connaissance sur le problème.

La génétique a mis en évidence l'existence de plusieurs opérateurs au sein d'un organisme donnant lieu au brassage génétique. Ces opérations interviennent lors de la phase de reproduction lorsque les chromosomes de deux organismes fusionnent.

Ces opérations sont imitées par les algorithmes génétiques afin de faire évoluer les populations de solutions de manière progressive.

### II.8.2.3 Les sélections

Pour déterminer quels individus sont plus enclins à obtenir les meilleurs résultats, une sélection est opérée. Ce processus est analogue à un processus de sélection naturelle, les individus les plus adaptés gagnent la compétition de la reproduction tandis que les moins adaptés meurent avant la reproduction, ce qui améliore globalement l'adaptation

Il existe plusieurs techniques de sélection, les principales sont :

1. Sélection par rang,
2. Probabilité de sélection proportionnelle à l'adaptation,
3. Sélection par tournoi,
4. Sélection uniforme.

### II.8.2.4 Lescrossant over ou recombinaison

Lors de cette opération, deux chromosomes s'échangent des parties de leurs chaînes, pour donner de nouveaux chromosomes. Ces crossant over peuvent être simples ou multiples. Dans le premier cas, les deux chromosomes se croisent et s'échangent des portions d'ADN en un seul point. Dans le deuxième cas, il y a plusieurs points de croisement. Pour les algorithmes génétiques, c'est cette opération qui est prépondérante. Sa probabilité d'apparition lors d'un croisement entre deux chromosomes est un paramètre de l'algorithme génétique.

- **Les mutations**

D'une façon aléatoire, un gène peut, au sein d'un chromosome être substitué à un autre. De la même manière que pour les crossant over, on définit ici un taux de mutation lors des changements de populations qui est généralement compris entre 0.001 et 0.01. Il est nécessaire de choisir pour ce taux une valeur relativement faible de manière à ne pas tomber dans une recherche aléatoire et conserver le principe de sélection et d'évolution. La mutation sert à éviter une convergence prématurée de l'algorithme.

- **Codage**

Pour les algorithmes génétiques, un des facteurs les plus importants, si ce n'est le plus important, est la façon dont sont codés les solutions, c'est-à-dire les structures de données qui coderont les gènes.

- **Codage binaire**

Le principe est de coder la solution selon une chaîne de bit. Ce type de codage est le plus utilisé car il présente plusieurs avantages [20, 23, 21, 22]. Il existe au moins un côté négatif qui fait que d'autres existent. Ce codage est peu naturel par rapport à un problème donné.

- **Codage à caractère multiple**

Ce type de codage est plus naturel que le codage binaire. Il est utilisé dans de nombreux cas poussés [20, 23, 21, 22, 24]. Ce codage utilise une structure arborescente avec une racine de laquelle peuvent être issus un ou plusieurs fils. Un de leurs avantages est qu'ils peuvent être utilisés dans le cas de problèmes où les solutions n'ont pas une taille finie. Les arbres de tailles quelconques peuvent être formés par le biais de croisement et de mutations. Le problème de ce type de codage est que les arbres résultants sont souvent difficiles à analyser et que l'on peut se retrouver avec des arbres dont la taille est importante. Pour le choix du type de codage, il suffit de choisir celui qui semble le plus naturel en fonction du problème à traiter et développer ensuite l'algorithme de traitement. Bien que les algorithmes génétiques soient considérés aujourd'hui comme une méthode d'optimisation, l'objectif initial consistait à concevoir des systèmes d'apprentissage généraux, robustes et adaptatifs, applicables à une large classe de problèmes. L'universalité d'un tel algorithme pose évidemment des problèmes d'efficacité en pratique. En effet, en tant que méthode d'optimisation, un algorithme génétique classique se base uniquement sur des opérateurs « aveugles ». Une autre voie intéressante pour améliorer l'efficacité des algorithmes génétiques consiste à combiner le cadre génétique avec d'autres méthodes de résolution [21].

### **II.8.3 Les méta- heuristiques éthologiques/colonies de fourmis**

Cette méta- heuristique s'inspire des comportements collectifs des fourmis dans leur découverte de nouvelles sources de nourriture. En effet, ces insectes utilisent des phéromones afin de marquer les informations qu'ils ont recueillies sur leur environnement. On appelle cela stigmergie. L'utilisation de ces phéromones leur permet de repérer les plus courts chemins entre une source de nourriture et leur nid. Car malgré leur capacité cognitive limitée, elles sont collectivement capables de résoudre des problèmes complexes.

### II.8.4 Méthode de recherche Tabou

La recherche Tabou a été introduite par Glover [25] et a montré ses performances sur de nombreux problèmes d'optimisation. C'est une technique d'exploration locale combinée avec un certain nombre de règles et de mécanismes permettant à celle-ci de surmonter l'obstacle des optima locaux, tout en évitant de cycliser.

Le principe de l'algorithme est le suivant : à chaque itération, le voisinage (complet ou sous-ensemble de voisinage) de la solution courante est examiné et la meilleure solution est sélectionnée, même si elle est moins bonne que la solution, la méthode interdit les mouvements aboutissant à une solution récemment visitée. Pour cela, une liste taboue contenant les attributs des dernières solutions visitées est tenue à jour. Chaque nouvelle solution considérée enlève de cette liste la solution la plus anciennement visitée. Ainsi, la recherche de la solution appartenant à la liste taboue. Dans certains cas, on mémorise les mouvements réalisés plutôt que les solutions complètes, essentiellement dans le but de mémoriser le moins d'informations possibles.

## II.9 Méthodes de pénalité

### II.9.1 Principe général des méthodes de pénalité

Les méthodes d'optimisation que nous allons utiliser, sont des méthodes de minimisation sans contraintes. Or, notre problème est avec contraintes. C'est pour cette raison qu'on va utiliser une méthode basée sur la transformation du problème original avec contraintes en un problème auxiliaire sans contraintes où le minimum est le même que celui du problème original. Le principe de base de cette méthode consiste à modifier le critère en lui ajoutant une fonction de pénalisation  $p(x)$ . C'est-à-dire, qu'on ramène le problème de programmation avec contraintes en un problème de programmation sans contraintes.

Les méthodes de pénalité constituent une famille d'algorithmes particulièrement intéressants du double point de vue de la simplicité de principe et de l'efficacité pratique.

Il existe plusieurs possibilités du choix de la fonction de pénalité :

### II.9.2 Méthode de Fiasco et Mc Cornick

Cette méthode consiste à ramener le problème d'optimisation minimisation de la fonction de pénalisée suivante [26] :

$$f_m(x, r_K) = f_{obj}(x) + r_K \sum_{i=1}^n g_i^2(x) + \frac{1}{r_K} \sum_{j=1}^m h_j^2(x) \quad (\text{II.1})$$

Où  $r_K$  est une constante de réglage de calcul (coefficient de pénalité). Elle est choisie de telle sorte que :

$r_K > 0$  et limite de  $r_K = 0$  quand  $K \rightarrow \infty$

Avec :

$$r_K = \frac{r_{K-1}}{p} \text{ et } r_0 = 1 \quad (\text{II.2})$$

Où  $p$  est une constante choisie.

### II.9.3 Méthodes de pénalité extérieure

Dans cette méthode, on introduit les deux types de contraintes égalités et inégalités. La fonction objectives  $f_m$  s'écrit sous la forme suivante [26] :

$$f_m = f_{obj}(x) = E(r_K, g, h) \quad (\text{II.3})$$

$$f_m = f_{obj}(x) + \frac{1}{r_K} \sum_{i=1}^n D_i g_i^2(x) + \frac{1}{r_K} \sum_{j=1}^m B_j h_j^2(x) \quad (\text{II.4})$$

Où  $E(r_K, g, h)$  est le terme de pénalisation extérieure.

$$E(r_K, g, h) = \frac{1}{r_K} \sum_{i=1}^n D_i g_i^2(x) + \frac{1}{r_K} \sum_{j=1}^m B_j h_j^2(x) \quad (\text{II.5})$$

On aura donc  $f_m$  sous la forme suivante :

Avec

$$\begin{cases} D_i > 0, \text{ si } g_i(x) < 0 \\ D_i = 0, \text{ si } g_i(x) \geq 0 \end{cases} \quad (\text{II.6})$$

Et

$$\begin{cases} B_j > 0, \text{ si } h_j(x) \neq 0 \\ B_j = 0, \text{ si } h_j(x) = 0 \end{cases} \quad (\text{II.7})$$

$D_i$  et  $B_j$  sont des constantes.

### II.9.4 Méthodes de pénalité intérieure

Le principal inconvénient de la méthode de pénalité extérieure est que l'optimum  $x^*$  est approché vers l'extérieur, ce qui a conduit de chercher une autre méthode de pénalité dans laquelle l'optimum est approché vers l'intérieur (d'où le nom de pénalité intérieure) [27].

La pénalisation intérieure peut être appliquée uniquement dans le cas, Ou on a des contraintes de type inégalité.

La fonction objective  $f_m$  s'écrit sous la forme suivante :

$$f_m = f_{obj}(x) + I(r_K, g) \quad (\text{II.8})$$

Où:

$I(r_K, g)$ : est le terme de pénalisation intérieure.

Le terme  $I(r_K, g)$  peut être donné par l'expression suivante :

$$I(r_K, g) = r_K \sum_{i=1}^n \frac{A_i}{g_i(x)} \quad (\text{II.9})$$

On aura donc  $f_m$  sous la forme suivante :

$$f_m = f_{obj}(x) + r_K \sum_{i=1}^n \frac{A_i}{g_i(x)} \quad (\text{II.10})$$

Avec

$$\begin{cases} A_i > 0, \text{ si } g_i(x) \geq 0 \\ A_i = 0, \text{ si } g_i(x) < 0 \end{cases} \quad (\text{II.11})$$

Où:

$A_i$ : est une constante.

### II.9.5 Méthode de pénalité mixte

Cette méthode englobe les termes de pénalisation intérieure représentés par  $I(r_K, g)$  et les termes de pénalisation extérieure représentés par  $E(r_K, g, h)$ . La fonction pénalisée s'écrit sous la forme suivante [26] :

$$f_m = f_{obj}(x) + r_K \sum_{i=1}^n \frac{A_i}{g_i(x)} + \frac{1}{r_K} \sum_{i=1}^n D_i g_i^2(x) + \frac{1}{r_K} \sum_{j=1}^m B_j h_j^2(x) \quad (\text{II.12})$$

## II.10 Méthodes métaheuristique proposées

### II.10.1 Algorithme de recherche de corbeau(CSA) : [28]

Cette méthode a été proposée en 2016 par A.Askarzadeh. Les corbeaux sont un genre d'oiseaux largement répandu qui est maintenant considéré comme l'un des animaux les plus intelligents du monde.

En tant que groupe, les corbeaux montrent des exemples remarquables d'intelligence et obtiennent souvent de très bons résultats aux tests d'intelligence. Ils peuvent mémoriser des visages, utiliser des outils, communiquer de manière sophistiquée et cacher et récupérer de la nourriture au fil des saisons.

Dans un troupeau de corbeaux, il existe un comportement qui présente de nombreuses similitudes avec un processus d'optimisation. Selon ce comportement, les corbeaux cachent leur excès de nourriture dans certaines positions (cachettes) de l'environnement et récupèrent la nourriture stockée lorsque cela est nécessaire. Les corbeaux sont des oiseaux gourmands car ils se succèdent pour obtenir de meilleures sources de nourriture. Trouver une source de nourriture cachée par un corbeau n'est pas un travail facile car si un corbeau en trouve un autre qui le suit, le corbeau essaie de tromper ce corbeau en se rendant à une autre position de l'environnement.

Du point de vue de l'optimisation, les corbeaux sont des chercheurs, l'environnement est un espace de recherche, chaque position de l'environnement correspond à une solution réalisable, la qualité de la source de nourriture est une fonction objective (fitness) et la meilleure source de nourriture de l'environnement est la solution globale du problème. Sur la base de ces similitudes, CSA tente de simuler le comportement intelligent des corbeaux pour trouver la solution aux problèmes d'optimisation. Dans cette approche, basée sur les comportements intelligents mentionnés ci-dessus, un algorithme métaheuristique basé sur la population, CSA, est développé. Les principes de la CSA sont énumérés comme suit :

\*Les corbeaux vivent sous forme de troupeau.

\*Les corbeaux mémorisent la position de leurs cachettes.

\*Les corbeaux se succèdent pour commettre des vols. Les corbeaux protègent leurs caches contre le vol par une probabilité.

On suppose qu'il existe un environnement de dimension  $d$  y compris un certain nombre de corbeaux. Le nombre de corbeaux (taille du troupeau) est  $N$  et la position du corbeau  $i$  au temps

(itération) iter dans le l'espace de recherche est spécifié par un vecteur  $X^{i.iter}$

où  $X^{i.iter} = [X_1^{i.iter} . X_2^{i.iter} \dots X_d^{i.iter}]$  et  $iter_{max}$  est le nombre ( $i = 1, 2, \dots, N; iter = 1, 2, \dots, iter_{max}$ )

maximum d'itérations. Chaque corbeau a une mémoire dans laquelle la position de sa cachette est mémorisée. A l'itération iter, la position de la cachette du corbeau i est indiquée par  $m^{i.iter}$ . C'est la meilleure position que le corbeau a obtenue jusqu'à présent.

En effet, en mémoire de chaque corbeau la position de sa meilleure expérience a été mémorisée. Les corbeaux se déplacent dans l'environnement et recherchent de meilleures sources de nourriture (cachettes).

Supposons qu'à l'itération iter, le corbeau j veuille visiter sa cachette,  $m^{j.iter}$ . A cette itération, le corbeau i décide de suivre le corbeau j pour s'approcher de la cachette du corbeau j. Dans ce cas, deux états peuvent se produire :

État 1 : le corbeau j ne sait pas que le corbeau i le suit. En conséquence, le corbeau i s'approchera de la cachette du corbeau j. Dans ce cas, la nouvelle position du corbeau i est obtenue comme suit :

$$(II.13) \quad (m^{j.iter} - X^{i.iter}) fl^{i.iter} \times r_i + X^{i.iter} = X^{i.iter}$$

où  $r_i$  est un nombre aléatoire avec une distribution uniforme entre 0 et 1 et  $fl^{i.iter}$  désigne la longueur de vol du corbeau i à l'itération iter

La figure 1 montre le schéma de cet état et l'effet de fl sur la capacité de recherche. Les petites valeurs de fl conduisent à une recherche locale (à le voisinage de  $X^{i.iter}$ ) et les valeurs élevées entraînent une recherche globale (loin de  $X^{i.iter}$ ). Comme le montre la figure 1(a), si la valeur de fl est sélectionnée moins que 1, la position suivante du corbeau i est sur la ligne en tirets entre  $X^{i.iter}$  et  $m^{j.iter}$ . Comme l'indique la figure 1(b), si la valeur de fl est sélectionnée plus de 1, la prochaine  $m^{j.iter}$  position du corbeau i est sur la ligne en tirets qui peut dépasser  $m^{j.iter}$ .

État 2 : le corbeau j sait que le corbeau i le suit. En conséquence, dans afin de protéger sa cache d'être chapardé, corbeau j va tromper corbeau i en allant à une autre position de l'espace de recherche.

Au total, les états 1 et 2 peuvent être exprimés comme suit :

$$X^{i.iter+1} = \begin{cases} X^{i.iter} + r_i \times fl^{i.iter} (m^{j.iter} - X^{i.iter}), & r_j \geq AP^{j.iter} \\ arandomposition & otherwise \end{cases} \quad (II.14)$$

où  $r_j$  est un nombre aléatoire avec une distribution uniforme entre 0 et 1 et  $AP^{j.iter}$  désigne la probabilité de prise de conscience du corbeau  $j$  à l'itération  $iter$ .

Les algorithmes métaheuristiques devraient fournir un bon équilibre entre diversification et intensification. En CSA, L'intensification et la diversification sont principalement contrôlées par le paramètre de probabilité de conscience (PA). Par diminution de la valeur de probabilité de prise de conscience, CSA a tendance à effectuer la recherche sur une région locale où une bonne solution actuelle est trouvée dans cette région. En conséquence, l'utilisation de petites valeurs de PA augmente l'intensification. D'autre part, par augmentation de la valeur de probabilité de conscience, la probabilité de rechercher au voisinage des bonnes solutions courantes diminue et CSA tend à explorer l'espace de recherche à une échelle globale (randomisation). En conséquence, l'utilisation de grandes valeurs de PA augmente la diversification.

### II.10.2 Implémentation de CSA pour l'optimisation

Le pseudo-code de CSA est illustré à la Fig II. 2. La procédure par étapes pour la mise en œuvre de CSA est donnée dans cette section.

Étape 1 : Initialiser le problème et les paramètres réglables

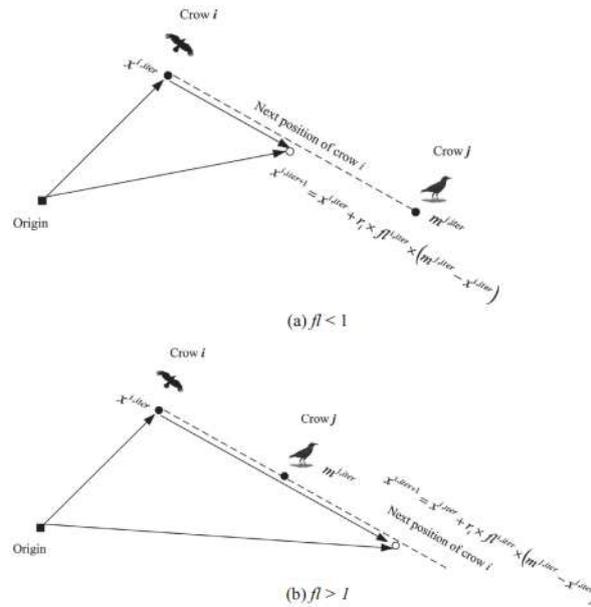
Le problème d'optimisation, les variables de décision et les contraintes sont définis. Ensuite, les paramètres ajustables de CSA (taille du troupeau (N), nombre maximum d'itérations (itermax), longueur de vol (fl) et probabilité de conscience (AP)) sont valorisés

Étape 2 : Initialiser la position et la mémoire des corbeaux

N corbeaux sont positionnés au hasard dans un espace de recherche à  $d$  dimensions comme les membres du troupeau. Chaque corbeau désigne une solution réalisable du problème et  $d$  est le nombre de variables de décision.

$$\text{Corbeaux} = \begin{bmatrix} x_1^1 & x_2^1 & \dots & x_d^1 \\ x_1^2 & x_2^2 & \dots & x_d^2 \\ \vdots & \vdots & \ddots & \vdots \\ x_1^N & x_2^N & \dots & x_d^N \end{bmatrix} \quad (\text{II.15})$$

La mémoire de chaque corbeau est initialisée. Comme à l'itération initiale, les corbeaux n'ont pas d'expériences, on suppose qu'ils ont caché leurs aliments à leurs positions initiales.



**Fig.II.2 :** Organigramme de l'état 1 dans CSA (a)  $fl < 1$  et (b)  $fl > 1$ .

Le corbeau peut aller à n'importe quelle position sur la ligne en tirets.

**Crow search algorithm**

**Randomly initialize the position of a flock of N crows in the search space**

**Evaluate the position of the crows**

**Initialize the memory of each crow**

**while** iter < iter<sub>max</sub>

**for** i = 1 : N (all N crows of the flock)

        Randomly choose one of the crows to follow (for example j)

**Define an awareness probability**

**if**  $r_j \geq AP^{i,iter}$ ,

$x^{i,iter} = x^{i,iter} + r_i \times fl^{i,iter} \times (m^{i,iter} - x^{i,iter})$

**else**

$x^{i,iter} =$  a random position of search space

**end if**

**end for**

Check the feasibility of new positions

Evaluate the new position of the crows

Update the memory of crows

**end while**

$$\text{Memory} = \begin{bmatrix} m_1^1 & m_2^1 & \dots & m_d^1 \\ m_1^2 & m_2^2 & \dots & m_d^2 \\ \vdots & \vdots & \ddots & \vdots \\ m_1^N & m_2^N & \dots & m_d^N \end{bmatrix} \quad (\text{II.16})$$

Étape 3 : Évaluer la fonction de condition physique (objectif)

Pour chaque corbeau, la qualité de sa position est calculée en insérant les valeurs des variables de décision dans la fonction objective.

Étape 4 : Générer une nouvelle position

Les corbeaux génèrent une nouvelle position dans l'espace de recherche comme suit : supposons que le corbeau veuille générer une nouvelle position. Pour cet objectif, ce corbeau sélectionne au hasard l'un des corbeaux du troupeau (par exemple corbeau  $j$ ) et le suit pour découvrir la position des aliments cachés par ce corbeau ( $m^j$ ). La nouvelle position du corbeau  $i$  est obtenue par l'équation. (2). Cette le processus est répété pour tous les corbeaux

Étape 5 : Vérifier la faisabilité de nouvelles positions

La faisabilité de la nouvelle position de chaque corbeau est vérifiée. Si la nouvelle position d'un corbeau est possible, le corbeau met à jour sa position. Sinon, le corbeau reste dans la position actuelle et ne passer à la nouvelle position générée.

Étape 6 : Évaluer la fonction de remise en forme des nouveaux postes

La valeur de la fonction fitness pour la nouvelle position de chaque corbeau est calculée.

Étape 7 : Mettre à jour la mémoire

Les corbeaux mettent à jour leur mémoire comme suit :

$$(II.17) \begin{cases} X^{i.iter+1} \\ m^{i.iter} \end{cases} \begin{cases} f(X^{i.iter+1}) \text{ is better than } f(m^{i.iter}) \\ o.w \end{cases} = m^{i.iter+1}$$

où  $f(x)$  désigne la valeur de la fonction objectif. On voit que si la valeur de la fonction de fitness de la nouvelle position de un corbeau est meilleure que la valeur de la fonction de fitness de la mémoire position, le corbeau met à jour sa mémoire par la nouvelle position.

Étape 8 : Vérifier le critère d'arrêt

Les étapes 4 à 7 sont répétées jusqu'à ce que  $iter_{max}$  soit atteint. Lorsque le critère d'arrêt est satisfait, la meilleure position de la mémoire en termes de la valeur de la fonction objectif est indiquée comme la solution du problème d'optimisation.

### II.10.3 Algorithme d'optimisation de mouette (SOA) [29]

Cette méthode a été proposée par Gaurav Dhiman et Vijay Kumar en 2018, L'inspiration et la modélisation mathématique de l'algorithme proposé est discuté en détail:

### a. Paradigme biologique

Les mouettes, appelées scientifiquement Laridés, sont des oiseaux marins qui peuvent être trouvés partout sur la planète. Il existe une large gamme d'espèces de mouettes avec des masses et des longueurs différentes. Les mouettes sont omnivores et mangent des insectes, des poissons, des reptiles, des amphibiens, des vers de terre, etc. Le corps de la plupart des mouettes est couvert de plumage blanc. Les mouettes sont très oiseaux intelligents. Ils utilisent de la chapelure pour attirer les poissons et produisent du bruit de pluie avec leurs pieds pour attirer les vers de terre cachés sous le sol. Les mouettes peuvent boire aussi bien de l'eau douce que de l'eau salée. La plupart des animaux sont incapables de le faire. Cependant, les mouettes ont une paire spéciale de glandes juste au-dessus de leurs yeux qui est spécifiquement conçue pour chasser le sel de leurs systèmes par les ouvertures du bec. Généralement, les mouettes vivent en colonies. Ils utilisent leur intelligence pour trouver et attaquer la proie. La chose la plus importante à propos des mouettes sont leurs comportements de migration et d'attaque. La migration est définie comme le mouvement saisonnier des mouettes d'un endroit à un autre pour trouver les sources de nourriture les plus riches et les plus abondantes qui fourniront une énergie adéquate. Ce comportement est décrit comme suit:

- Pendant la migration, ils voyagent en groupe. Les positions initiales des mouettes sont différentes pour éviter les collisions entre chaque autre.
- En groupe, les mouettes peuvent voyager dans la direction la plus apte à la survie, c'est-à-dire une mouette dont la valeur de fitness est faible par rapport aux autres.
- Sur la base de la mouette la plus apte, d'autres mouettes peuvent mettre à jour leurs positions initiales.

Les mouettes attaquent fréquemment les oiseaux migrateurs au-dessus de la mer. Quand ils migrent d'un endroit à un autre. Ils peuvent faire un mouvement en spirale naturel pendant l'attaque. Conceptuellement un exemple de ces comportements. Ces comportements peuvent être formulés de telle manière qu'ils puissent être liés à la fonction objective à améliorer. Cela permet de formuler un nouvel algorithme d'optimisation. Cet article se concentre sur deux comportements naturels des mouettes.

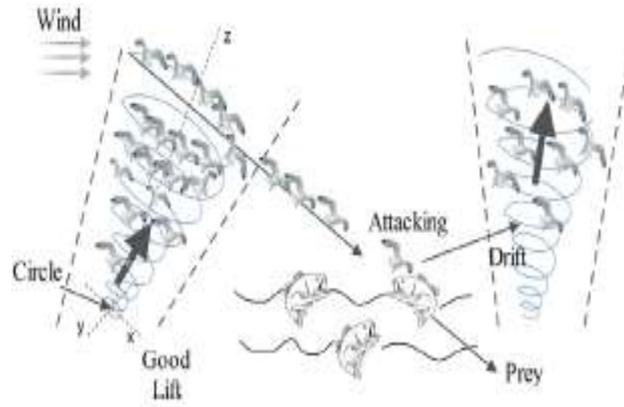


Fig.II.3 Migration et comportements d'attaque des mouettes.

## b.Modèle mathématique

Les modèles mathématiques de migration et d'attaque de la proie sont discutés:

### b.1.Migration (exploration)

Pendant la migration, l'algorithme simule comment le groupe de les mouettes se déplacent d'une position à l'autre. Dans cette phase, une mouette doit remplir trois conditions :

- Éviter les collisions : pour éviter les collisions entre Voisins (i.e. autres mouettes), une variable supplémentaire A est utilisée pour calculer la position du nouvel agent de recherche

$$\vec{C}_s = A \times \vec{P}_s(x) \quad (\text{II.18})$$

où  $\vec{C}_s$  représente la position de l'agent de recherche qui ne pas entrer en collision avec un autre agent de recherche,  $\vec{P}_s$  représente le courant position de l'agent de recherche, x indique l'itération en cours, et A représente le comportement de mouvement de l'agent de recherche dans un espace de recherche donné.

$$A = f_c - (x \times (f_c / \text{Max}_{iteration}))$$

$$\text{Où } : x = 0, 1, 2, \dots, \text{Max}_{iteration} \quad (\text{II.19})$$

Où  $f_c$  est introduit pour contrôler le rythme d'embauche La variable A qui décroît linéairement de  $f_c$  à 0. dans ce travail, f.valeur de  $f_c$  Réglé sur 2. Sensibilité détaillée. Analyse de  $f_c$

- Mouvement vers la direction du meilleur voisin : après avoir évité la collision entre voisins, les agents de recherche sont se déplacer vers la direction du meilleur voisin

$$\vec{M}_s = B \times (\vec{P}_{bs}(x) - \vec{P}_s(x)) \quad (\text{II.20})$$

où  $\vec{M}_s$  représente les positions d'agent de recherche  $\vec{P}_s$  vers l'agent de recherche le mieux adapté  $\vec{P}_{bs}$  (c'est-à-dire la mouette la plus apte). Le comportement de B est randomisé, ce qui est responsable du bon équilibre entre l'exploration et l'exploitation. B est calculé comme :

$$B = 2 \times A^2 \times rd \quad (\text{II.21})$$

où rd est un nombre aléatoire se situe dans la plage de [0, 1].

- Restez proche du meilleur agent de recherche : enfin, la recherche L'agent peut mettre à jour sa position concernant le meilleur agent de recherche

$$\vec{D}_s = |\vec{C}_s + \vec{M}_s| \quad (\text{II.22})$$

$D_s$  représente la distance entre l'agent de recherche et l'agent de recherche le mieux adapté (c'est-à-dire la meilleure mouette dont la valeur de fitness est moins).

### b.2. Attaquer (exploiter)

L'exploitation vise à exploiter l'histoire et l'expérience du processus de recherche. Les mouettes peuvent changer l'angle d'attaque en continu ainsi que la vitesse pendant la migration. Ils gardent sa hauteur à l'aide de ses ailes et de son poids. En attaquant Proie, le comportement de mouvement en spirale se produit dans l'air. Ce comportement est décrit dans les plans x, y et z comme suit :

$$x' = r \times \cos(k) \quad (\text{II.23})$$

$$y' = r \times k \quad (\text{II.24})$$

$$r = u \times e^{kv} \quad (\text{II.25})$$

où r est le rayon de chaque tour de la spirale, k est un nombre aléatoire dans la plage  $[0 \leq k \leq 2\pi]$ . u et v sont des constantes pour définir le forme de spirale, et e est la base du logarithme népérien. La mise à jour la position de l'agent de recherche est calculée à l'aide des équations. (25)–(20).

$$\vec{P}_s(x) = (\vec{D}_s \times x' \times y' \times z') + \vec{P}_{bs}(x) \quad (\text{II.26})$$

Où  $\vec{P}_s(x)$  enregistre la meilleure solution et met à jour une position d'autres agents de recherche.

La SOA proposée commence par une population générée aléatoirement.

Les agents de recherche peuvent mettre à jour leurs positions par rapport aux meilleurs agents de recherche pendant le processus d'itération.  $A$  est diminué linéairement de  $fc$  à 0. Pour une transition en douceur entre l'exploration et l'exploitation, la variable  $B$  est responsable. Par conséquent, la SOA est considérée comme un optimiseur global (voir Algorithme) en raison de sa meilleure exploration et la capacité d'exploitation.

### Algorithm 1 Seagull Optimization Algorithm

```

Input: Seagull population  $\vec{P}_s$ 
Output: Optimal search agent  $P_{bs}$ 
1: procedure SOA
2: Initialize the parameters  $A$ ,  $B$ , and  $Max_{iteration}$ 
3:   Set  $fc \leftarrow 2$ 
4:   Set  $u \leftarrow 1$ 
5:   Set  $v \leftarrow 1$ 
6:   while ( $x < Max_{iteration}$ ) do
7:      $\vec{P}_{bs} \leftarrow \text{ComputeFitness}(\vec{P}_s)$  /* Calculate the fitness values of each search agent
using ComputeFitness function */ /* Migration behavior */
8:      $rd \leftarrow \text{Rand}(0, 1)$  /* To generate the random number in range [0, 1] */
9:      $k \leftarrow \text{Rand}(0, 2\pi)$  /* To generate the random number in range [0, 2 $\pi$ ] */
/* Attacking behavior */
10:     $r \leftarrow u \times e^{kv}$  /* To generate the spiral behavior during migration */
11:    Calculate the distance  $\vec{D}_s$  using Eq. (9)
12:     $P \leftarrow x' \times y' \times z'$  /* Compute  $x$ ,  $y$ ,  $z$  planes using Eqs. (10)–(13) */
13:     $\vec{P}_s(x) \leftarrow (\vec{D}_s \times P) + \vec{P}_{bs}$ 
14:     $x \leftarrow x + 1$ 
15:  end while
16:  return  $\vec{P}_{bs}$ 
17: end procedure

1: procedure ComputeFitness( $\vec{P}_s$ )
2:   for  $i \leftarrow 1$  to  $n$  do /* Here,  $n$  represents the dimension of a given problem */
3:      $FITs[i] \leftarrow \text{FitnessFunction}(P_s(\vec{i}, :))$  /* Calculate the fitness of each
individual */
4:   end for
5:    $FIT_{sbest} \leftarrow \text{BEST}(FITs [])$  /* Calculate the best fitness value using BEST
function */
6:  return  $FIT_{sbest}$ 
7: end procedure

1: procedure BEST( $FIT_s []$ )
2:    $Best \leftarrow FIT_s[0]$ 

```

```

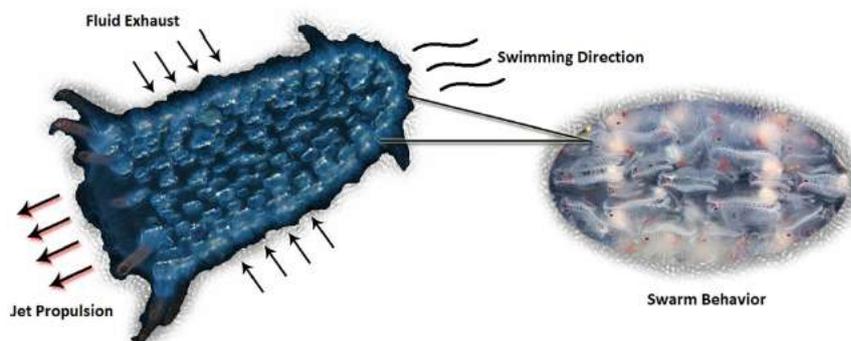
3:   for i ← 1 to n do
4:     if(FITs [i] < Best ) then
5:       Best ← FITs [i]
6:     end if
7:   end for
8:   return Best /* Return the best fitness value */
9: end procedure

```

SOA est un algorithme d'optimisation bio-inspiré donc il y a pas besoin de processus évolutifs, c'est-à-dire de croisement et de mutation. On peut donc conclure que l'efficacité de calcul de l'algorithme proposé est bien meilleure que l'autre concurrent approches.

#### II.10.4 Méthode Tunicate Swarm Algorithm (TSA): [30]

Dans cette approche a été proposée par **Satnam Kaur et al en 2020**. Elles présentent une nouvelle méthode méta-heuristique bio-inspiré, appelé Tunicate Swarm Algorithm (TSA), est proposé pour l'optimisation des problèmes contraints non linéaires. Il est inspiré de l'essaim comportement du tunicier pour survivre avec succès dans les profondeurs de l'océan. Dans cette section, l'inspiration et la modélisation mathématique du l'algorithme proposé sont décrits en détail.



**Fig.II.4.** Comportement des essaims de tuniciers dans l'océan profond.

##### a. Modèle mathématique et algorithme d'optimisation

Le tunicier a la capacité de trouver l'emplacement de la source de nourriture en mer. Cependant, il n'y a aucune idée de la source de nourriture dans la recherche donnée espace. Dans cet algorithme, deux comportements de tunicier sont utilisés pour trouver la source de nourriture, c'est-à-dire optimale. Ces comportements sont la propulsion par le jet et l'essaim d'intelligence. Pour modéliser mathématiquement le comportement de propulsion du jet, un tunicier doit remplir trois conditions à savoir éviter les conflits entre agents de recherche, évolution vers la position de meilleur agent de recherche, et reste proche du meilleur agent de recherche. Alors que le

comportement de l'essaim mettra à jour les positions des autres agents de recherche sur la meilleure optimale solution. La modélisation mathématique de ces comportements est décrite dans les sous-sections précédentes

### a.1. Éviter les conflits entre les agents de recherche

Pour éviter les conflits entre les agents de recherche (c'est-à-dire d'autres tuniciers), le vecteur  $\vec{A}$  est utilisé pour le calcul de la nouvelle position de l'agent de recherche

$$\vec{A} = \frac{\vec{G}}{\vec{M}} \quad (\text{II.27})$$

$$\vec{G} = c_2 + c_3 - \vec{F} \quad (\text{II.28})$$

$$\vec{F} = 2.c_1 \quad (\text{II.29})$$

Cependant,  $\vec{G}$  est la force de gravité et  $\vec{F}$  montre l'advection du débit d'eau dans l'océan profond. Les variables  $c_1, c_2$  et  $c_3$  sont des nombres aléatoires se trouvant dans la plage de  $[0, 1]$ .  $\vec{M}$  représente les forces sociales entre les agents de recherche. Le vecteur  $\vec{M}$  est calculé comme suit:

$$\vec{M} = [P_{min} + c_1.P_{max} - P_{min}] \quad (\text{II.30})$$

où  $P_{min}$  et  $P_{max}$  représentent les vitesses initiale et subordonnée à faire une interaction sociale. Dans ce travail, les valeurs de  $P_{min}$  et  $P_{max}$  sont considérés comme 1 et 4, respectivement. Notez que la sensibilité détaillée l'analyse de ces paramètres est discutée dans la section 5.5.

### a.2. Mouvement vers la direction du meilleur voisin

Après avoir évité le conflit entre voisins, les agents de recherche sont se déplacer vers la direction du meilleur voisin

$$\vec{PD} = |\vec{FS} - r_{and}.\vec{P}_P(x)| \quad (\text{II.31})$$

où  $\vec{PD}$  est la distance entre la source de nourriture et l'agent de recherche, c'est-à-dire, tunicier,  $x$  indique l'itération actuelle,  $\vec{FS}$  est la position de la nourriture source, c'est-à-dire optimale. Le vecteur  $\vec{P}_P(x)$  indique la position du tunicier et  $rand$  est un nombre aléatoire dans la plage  $[0, 1]$

### a.3. Converger vers le meilleur agent de recherche

L'agent de recherche peut maintenir sa position vers la meilleure recherche agent (c.-à-d. source de nourriture)

$$\vec{P}_P(x) = \begin{cases} \vec{FS} + \vec{A}.\vec{PD}, & \text{if } r_{and} \geq 0.5 \\ \vec{FS} + \vec{A}.\vec{PD}, & \text{if } r_{and} < 0.5 \end{cases} \quad (\text{II.32})$$

où  $\vec{P}_P(\hat{x})$  est la position mise à jour du tunicier par rapport au position de la source de nourriture  $\vec{FS}$ .

#### a.4. Comportement de l'essaim

Afin de simuler mathématiquement le comportement en essaim de tuniciers, les deux premières meilleures solutions optimales sont enregistrées et mettent à jour les positions des autres agents de recherche en fonction de la position de la meilleure recherche agents. La formule suivante est proposée pour définir le comportement de l'essaim de tunicier :

$$P_P(\vec{x} + 1) = \frac{\vec{P}_P(x) + P_P(\vec{x} + 1)}{2 + c_1} \quad (\text{II.33})$$

La figure II.5 montre comment les agents de recherche peuvent mettre à jour leurs positions en fonction à la position de  $\vec{P}_P(x)$ . La position finale serait dans un endroit aléatoire, dans un cylindre ou en forme de cône, qui est défini par la position de tunicier. Le pseudo code de l'algorithme TSA proposé est affiché dans Algorithme 1. Il y a quelques points importants sur l'algorithme TSA qui sont décrits comme :

- $\vec{A}, \vec{G}$  et  $\vec{F}$  aident les solutions à se comporter aléatoirement dans un espace de recherche et responsable d'éviter les conflits entre les différents agents de recherche.
- La possibilité de meilleures phases d'exploration et d'exploitation est fait par les variations des vecteurs  $\vec{A}, \vec{G}$  et  $\vec{F}$ .
- La propulsion à réaction et les comportements en essaim du tunicier dans un l'espace de recherche définit le comportement collectif de l'algorithme TSA.

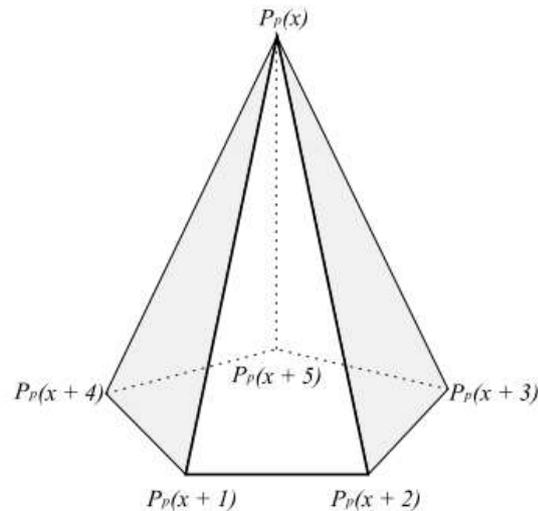


Fig.II.5 Vecteurs de position 3D des tuniciers

### b. Algorithme de TSA

Les étapes et l'algorithme du TSA proposé sont donnés au dessous:

Étape 1: Initialisez la population de tuniciers  $\overrightarrow{P_p}$ .

Étape 2: Choisissez les paramètres initiaux et le nombre maximum d'itérations.

Étape 3: Calculez la valeur de fitness de chaque agent de recherche.

Étape 4: Après avoir calculé la valeur de fitness, le meilleur agent de recherche est explorée dans l'espace de recherche donné.

Étape 5: Mettez à jour la position de chaque agent de recherche à l'aide de l'Eq.(7).

Étape 6: Ajustez l'agent de recherche mis à jour qui dépasse la limite dans un espace de recherche donné.

Étape 7: calculez la valeur de fitness mise à jour de l'agent de recherche. S'il y a un meilleure solution que la solution optimale précédente, puis mettez à jour  $P_p$ .

Étape 8: Si le critère d'arrêt est satisfait, l'algorithme s'arrête. Sinon, répétez les étapes 5 à 8.

Étape 9: retournez la meilleure solution optimale obtenue à ce jour.

**Algorithm Tunicate Swarm Algorithm**

```

Input :Tunicate population  $\vec{P}$ 
Output :Optimal fitness value  $\vec{P}S$ 
1 : procedure TSA
2 : Initialize the parameters  $\vec{A}, \vec{G}, \vec{F}, \vec{M}$ , and  $M_{ax\_iterations}$ 
3 : Set  $\leftarrow 1$ 
4 : Set  $\leftarrow 4$ 
5 : Set Swarm  $\leftarrow 0$ 
6 : While ( $x < Max_{iterations}$ ) do
7 :     for  $i \leftarrow 1$  to 2 do /*Looping for compute swarm brhavior*/
8 :          $\vec{FS} \leftarrow ComputeFitness(\vec{P}_p)$  /*Calulate the fitness values of each agent using.
ComputeFitness function*/ /*Jet propulsion behavior*/
9 :          $c_1, c_2, c_3, r_{and} \leftarrow Rand()$  /*Rand() is a function to generate the random number in
range [0, 1] */
10:          $\vec{M} \leftarrow [P_{min} + c_1 \times P_{max} - P_{min}]$ 
11:          $\vec{F} \leftarrow 2 \times c_1$ 
12:          $\vec{G} \leftarrow c_2 + c_3 - \vec{F}$ 
13:          $\vec{A} \leftarrow \vec{G} / \vec{M}$ 
14:          $\vec{PD} \leftarrow ABS(\vec{FS} - r_{and} \times \vec{P}_p(x))$  /*Swarm behavior */
15:         if ( $r_{and} \leq 0.5$ ) then
16:              $Sicarm + \vec{FS} + \vec{A} \times \vec{PD}$ 
17:         else
18:              $Sicarm \leftarrow Sicarm + \vec{FS} - \vec{A} \times \vec{PD}$ 
19:         end if
20:     end for
21:  $\vec{P}_p(x) \leftarrow Sicarm / (2 + c_1)$ 
22:  $Sicarm \leftarrow 0$ 
23: Update the parameters  $\vec{A}, \vec{G}, \vec{F},$  and  $\vec{M}$ 
24:  $x \leftarrow x + 1$ 
25: end while
26: return  $\vec{FS}$ 
27: end procedure
28: procedure COMPUTEFTTNESS( $\vec{P}_p$ )
29:     for  $I \leftarrow 1$  to  $n$  do /*Here,  $n$  represents the dimension of a given problem*/
30:          $FIT_p(i) \leftarrow FinessFunction(P_p(\vec{i}, :))$  /*Calculate the fitness of each individual*/
31:     end for
32:  $FIT_{Pbest} \leftarrow BEST(FIT_{PII})$ 
33: return  $FIT_{Pbest}$ 
34: end procedure

```

```

35: procedure BEST(FITpII)
36: Best ← FITp[0]
37: for i ← 1 to n do
38:   if (FITp[i] < Best) then
39:     Bset ← FITp[i]
40:   end if
41: end for
42: return Best /*Return the best fitness value*/
43: end procedure

```

### II.10.5 Algorithme de la luciole(FFA)[31]

Pour simplifier la description de l'algorithme de luciole (FFA), nous utilisons maintenant les trois règles idéalisées suivantes :

- 1) toutes les lucioles sont unisexes afin qu'une luciole soit attirée par d'autres lucioles indépendamment de leur sexe.
- 2) L'attractivité est proportionnelle à leur luminosité, donc pour deux des lucioles clignotantes, la moins brillante se déplacera vers la plus brillante. l'attractivité est proportionnelle à la luminosité et ils diminuent tous les deux à mesure que leur la distance augmente. S'il n'y en a pas de plus brillant qu'une luciole en particulier, il déplacera au hasard ;
- 3) La luminosité d'une luciole est affectée ou déterminée par l'évaluation de la fonction objectif. Pour un problème de maximisation, la luminosité peut être simplement proportionnelle à la valeur de la fonction objective. Autre formes de luminosité peuvent être définies d'une manière similaire à la fonction de fitness dans algorithmes génétiques ou l'algorithme de recherche de nourriture bactérienne (BFA) [6, 10]. Dans l'algorithme Luciole, il y a deux problèmes importants : la variation de la lumière l'intensité et la formulation de l'attractivité. Pour simplifier, on peut toujours supposer que l'attrait d'une luciole est déterminé par sa luminosité qui à son tour est associée à la fonction objective codée.

Dans le cas le plus simple pour des problèmes d'optimisation maximale, la luminosité  $I$  d'une luciole à un endroit particulier  $x$  peut être choisi comme  $I(x)=f(x)$ . Cependant, l'attractivité  $\beta$

est relative, elle doit être vue aux yeux du spectateur ou jugé par les autres lucioles. Ainsi, il variera avec la distance  $r_{ij}$  entre luciole  $i$  et luciole  $j$ . De plus, l'intensité lumineuse diminue avec la distance de sa source, nous devrions donc permettre l'attractivité varie avec le degré d'absorption. Dans la forme la plus simple, l'intensité lumineuse  $I(r)$  varie selon la loi du carré inverse  $I(r) = \frac{I_s}{r^2}$

Où  $I_s$  est l'intensité à la source. Pour un support donné avec une lumière fixe coefficient d'absorption, l'intensité lumineuse  $I$  varie avec la distance rest comme suit :

$$I = I_0 e^{-\gamma r} \quad (\text{II.34})$$

Où  $I_0$  est l'intensité lumineuse d'origine. Comme l'attractivité d'une luciole est proportionnelle à l'intensité lumineuse vue par lucioles adjacentes, nous pouvons maintenant définir l'attractivité  $\beta$  d'une luciole par l'équation suivante :

$$\beta = \beta_0 e^{-\gamma r^2} \quad (\text{II.35})$$

où  $\beta_0$  est l'attractivité à  $r = 0$ .

#### Algorithme de la luciole

Begin

Objective function  $f(x)$ ,  $x = (x_1, \dots, x_d)^T$

Generate initial population of fireflies  $x_i$  ( $i = 1, 2, \dots, n$ )

Light intensity

$I_i$  at  $x_i$  is determined by  $f(x_i)$

Define light absorption coefficient  $\gamma$

while ( $t < \text{MaxGeneration}$ )

for  $i = 1 : n$  all  $n$  fireflies

for  $j = 1 : i$  all  $n$  fireflies

if ( $I_j > I_i$ )

Move firefly  $i$  towards  $j$  in  $d$ -dimension via Levy flights

end if

Attractiveness varies with distance  $r$  via  $\exp[-\gamma r]$

Evaluate new solutions and update light intensity

end for  $j$

end for  $i$

Rank the fireflies and find the current best

end while

Postprocess results and visualization

End

### **II.11 Conclusion**

Les méthodes de résolution sont extrêmement nombreuses, elles sont basées sur des principes totalement différents, chacune explore et exploite l'espace de recherche selon des techniques qui lui sont propres.

Pour notre étude, nous avons retenu l'algorithme de recherche de corbeau(CSA) parce qu'elle est extrêmement performante dans de nombreux domaines. C'est une méthode très efficace lorsqu'il s'agit d'exploiter une zone de l'espace de recherche. D'autre part, elle s'adapte assez bien au problème posé.

**Chapitre III**  
**RESULTATS ET SIMULATION**

### III.1 Introduction

L'algorithme de recherche de corbeau a été appliqué à de nombreux problèmes technologiques avec succès. Comme nous l'avons dit précédemment, la méthode de recherche de corbeau est une heuristique qui permet de contrer le problème des optimums locaux.

Nous proposons dans cette partie une application de la méthode de l'algorithme de recherche de corbeau dans la répartition optimale des puissances actives et on le compare avec trois méthodes métaheuristiques (SOA, TSA et FFA).

Dans le premier cas, nous avons appliqué notre approche le réseau à 05 générateurs, pour résoudre le problème du Dispatching Economique dynamique avec les pertes variables (ELD), Dans le second cas, nous avons adopté le réseau à 05 générateurs pour résoudre le problème du Dispatching Environnemental dynamique avec les pertes variables (ED),.

Dans la troisième application, nous avons examiné le réseau le réseau à 05 générateurs, les pour résoudre le problème du Dispatching Economique Environnemental combiné dynamique avec les pertes variables et l'effet d'ouverture des vannes sont pris en compte (CEED).

Les simulations ont été réalisées en utilisant l'environnement de calcul Matlab, sur un micro-ordinateur Acer Intel Core 2 i5-3210M (2,5 GHz) avec 4 Go de mémoire totale.

Dans le premier cas, elle était consacrée aux bénéfices économiques et environnementaux sans ressources énergétiques renouvelables (ELD), et le second cas était d'envoyer des émissions environnementales et sans ressources énergétiques renouvelables (ED). La troisième étude de cas représente les émissions économiques et les avantages environnementaux sans ressources énergétiques renouvelables (CEED). Et la quatrième étude de cas sur la répartition économique des ressources énergétiques renouvelables ( $ELD_{RES}$ ). Quant au cinquième cas, il traite de l'envoi des émissions et de l'environnement avec des ressources énergétiques renouvelables ( $ED_{RES}$ ). La sixième étude de cas sur les émissions économiques et l'enrichissement de l'environnement avec des ressources énergétiques renouvelables ( $CEED_{RES}$ ).

### III.2 PARTIE N°1

Nous avons appliqué les algorithmes CSA, SOA, TSA et FFA sur un réseau à 5 générateurs pour minimiser le coût de production de l'énergie électrique et les émissions des gaz toxique en tenant compte l'effet d'ouverture des vannes.

Quantités	Unité 1	Unité 2	Unité 3	Unité 4	Unité 5
$a_i$ (\$/(MW)2h)	0.0080	0.0030	0.0012	0.0010	0.0015
$b_i$ (\$/MWh)	2.0	1.8	2.1	2.0	1.8
$c_i$ (\$/h)	25	60	100	120	40
$e_i$ (\$/h)	100	140	160	180	200
$f_i$ (rad/MW)	0.042	0.040	0.038	0.037	0.035
$\alpha_i$ (lb/MW2hr)	0.0180	0.0150	0.0105	0.0080	0.0120
$\beta_i$ (lb/MW hr)	-0.805	-0.555	-1.355	-0.600	-0.555
$\gamma_i$ (lb/hr)	80	50	60	45	30
$\eta_i$ (lb/hr)	0.6550	0.5773	0.4968	0.4860	0.5035
$\delta_i$ (1/MW)	0.02846	0.02446	0.02270	0.01948	0.02075
$P_i, \min$ (MW)	10	20	30	40	50
$P_i, \max$ (MW)	75	125	175	250	300
$h_i$	1.820062	1.543604	3.491129	1.727848	0.757816

**Tableau III.1** Données de réseau à 05 générateurs

$$B = \begin{bmatrix} 0.000049 & 0.000014 & 0.000015 & 0.000015 & 0.000020 \\ 0.000014 & 0.000045 & 0.000016 & 0.000020 & 0.000018 \\ 0.000015 & 0.000016 & 0.000039 & 0.000010 & 0.000012 \\ 0.000015 & 0.000020 & 0.000010 & 0.000040 & 0.000014 \\ 0.000020 & 0.000018 & 0.000012 & 0.000014 & 0.000035 \end{bmatrix} \text{ per MW}$$

**Tableau III.2** Coefficients de perte B (système à 5 unités)

Temps (h)	charge (MW)						
1	410	7	626	13	704	19	654
2	435	8	654	14	690	20	704
3	475	9	690	15	654	21	680
4	530	10	704	16	580	22	605
5	558	11	720	17	558	23	527
6	608	12	740	18	608	24	463

**Tableau III.3** Charge demandé pendant 24 heures.

### III.2.1 Dispatching Economique dynamique avec les pertes variables (ELD)

Les fonctions objectives des coûts exprimées en dollars par heures sont données par les expressions suivantes:

$$F_{i,t}(P_{i,t}) = (a_i P_{i,t}^2 + b_i P_{i,t} + c_i + |e_i \times \sin(f_i \times (P_{i,min} - P_{i,t}))|) \quad (III.1)$$

#### a. Algorithme de recherche de corbeau (CSA) :

Nous présentons dans le tableau III.4 les résultats de l'algorithme CSA:

H	P1(MW)	P2(MW)	P3(MW)	P4(MW)	P5(MW)	Pcharge	Perte	Cout	Emission
1	84,8038	98,5398	50,6524	40,00002	139,759	410	3,75588	1363,64	546,6043
2	48,0748	98,5398	112,673	40	139,759	435	4,04790	1380,41	510,7425
3	88,8761	98,5398	112,673	40	139,759	475	4,84925	1423,79	584,1222
4	59,9544	98,5398	112,673	124,9079	139,759	530	5,83546	1584,71	590,8647
5	84,1388	98,5398	112,673	40	229,519	558	6,87173	1604,79	969,8663
6	55,0790	98,5398	112,673	209,8158	139,759	608	7,86793	1777,14	784,0622
7	73,5278	98,5398	112,673	209,8158	139,759	626	8,31671	1783,77	814,0872
8	97,4492	98,5398	112,673	124,9079	229,519	654	9,09003	1882,45	1071,515
9	49,6195	98,5398	112,673	209,8158	229,519	690	10,1682	1977,66	1175,437
10	64,0107	98,5398	112,673	209,8158	229,519	704	10,5594	1996,59	1194,648
11	80,4834	98,5398	112,673	209,815000	229,519	720	11,0321	1989,97	1226,653
12	101,112	98,5398	112,673	209,8158	229,519	740	11,6616	2106,45	1282,648
13	64,0107	98,5398	112,673	209,8158	229,519	704	10,5594	1996,59	1194,648
14	49,6195	98,5398	112,673	209,8158	229,519	690	10,1682	1977,66	1175,437
15	97,4492	98,5398	112,673	124,9079	229,519	654	9,09003	1882,45	1071,515
16	84,8003	40,0583	112,673	209,8158	139,759	580	7,10783	1746,07	745,1312
17	84,1388	98,5398	112,673	40	229,519	558	6,87173	1604,79	969,8663
18	55,0790	98,5398	112,673	209,8158	139,759	608	7,86793	1777,14	784,0622
19	97,4492	98,5398	112,673	124,9079	229,519	654	9,09003	1882,45	1071,515
20	64,0107	98,5398	112,673	209,8158	229,519	704	10,5594	1996,59	1194,648
21	39,3528	98,5398	112,673	209,8158	229,519	680	9,9016	1944,59	1166,578
22	52,0075	98,5398	112,673	209,8158	139,759	605	7,79645	1771,65	780,3514
23	56,8894	98,5398	112,673	124,9079	139,759	527	5,77048	1581,46	586,5843
24	81,4318	98,5398	112,673	124,9079	50	463	4,55307	1392,53	468,9682
Cout totale=4242.5\$; Emission totale= 21960.6 Ib ; Perte totale =193.393 MW									

**Tableau III.4** Les résultats obtenus par la méthode CSA

**b. Algorithme d'optimisation de mouette (SOA):**

Nous présentons dans le tableau III.5 les résultats de l'algorithme SOA:

H	P1(MW)	P2(MW)	P3(MW)	P4(MW)	P5(MW)	Pcharge	Perte	Cout	Emission
1	24,49812	20	30	40	300	410	4,498118	1439,552	1380,76
2	10,3527	120,4612	35,29645	40	233,5489	435	4,659233	1460,504	966,4851
3	10,21545	77,4949	122,3647	40	229,9784	475	5,053457	1552,942	866,245
4	10	123,506	128,6482	223,9115	50,40415	530	6,469847	1826,172	749,1873
5	10,22221	120,5032	175	208,4027	50,87208	558	7,000133	1829,171	787,1396
6	53,82849	24,33493	31,4282	207,1216	300	608	8,713256	2005,719	1656,352
7	74,97303	53,20565	30	248,2254	228,6667	626	9,070738	2121,684	1259,816
8	15,44137	125	175	48,18346	300	654	9,624834	2268,399	1704,771
9	57,51427	125	175	205,6716	136,8298	690	10,01568	2272,747	968,7669
10	75	125	175	40,01101	300	704	11,01101	2380,153	1757,259
11	56,95511	95,13386	30	250	300	720	12,08896	2412,28	1920,724
12	62,44823	125	52,83638	212,1447	300	740	12,42928	2499,689	1856,381
13	10	20	175	209,5766	300	704	10,57658	2221,791	1788,619
14	10	90,56174	173,4356	207,9374	218,1138	690	10,04851	2143,106	1189,367
15	73,76849	20,16804	60,51551	209,3162	300	654	9,768276	2203,987	1684,653
16	10	125	175	40	237,5079	580	7,507881	1924,915	1152,246
17	62,28653	98,38161	146,1875	207,4164	50,46947	558	6,741461	1863,652	676,9082
18	16,97101	20	30	250	300	608	8,97101	2056,313	1804,423
19	56,53749	91,97935	175	40	300	654	9,516847	2210,395	1630,194
20	74,75111	125	175	40,26622	300	704	11,01733	2374,637	1756,766
21	10	20	110,3121	250	300	680	10,3121	2206,885	1821,054
22	11,19715	61,22635	121,2513	119,3662	300	605	8,041026	2065,32	1497,056
23	10	20,10138	30	245,9073	227,7316	527	6,740282	1702,657	1172,686
24	18,6322	20	172,3484	206,6299	50,17711	463	4,787584	1567,1	605,002
Cout totale=48609.8\$ ; Emission totale= 32652.9 Ib ; Perte totale =204.664 MW									

**Tableau III.5** Les résultats obtenus pour la méthode SOA

**c. Algorithme d'essaim de tuniciers (TSA):**

Nous présentons dans le tableau III.6 les résultats de l'algorithme TSA:

H	P1(MW)	P2(MW)	P3(MW)	P4(MW)	P5(MW)	Pcharge	Perte	Cout	Emission
---	--------	--------	--------	--------	--------	---------	-------	------	----------

1	75	20	50,26042	128,3952	139,9875	410	3,643151	1421,475	494,6614
2	19,01762	105,4373	112,9574	63,0177	138,6278	435	4,057863	1503,052	518,7742
3	22,67442	96,08062	111,4859	199,8302	50	475	5,071116	1514,253	561,1997
4	74,68807	20	171,6779	40	229,8404	530	6,206444	1716,187	963,4745
5	25,54276	23,23529	175	109,8291	230,9619	558	6,569029	1880,342	971,9729
6	23,58568	93,1622	149,3554	210,1541	139,5601	608	7,817454	1932,59	809,77
7	75	25,48874	109,4993	124,6413	300	626	8,629319	1993,525	1514,466
8	75	106,8242	31,505	220,8109	229,6448	654	9,784825	1996,483	1257,83
9	75	112,5888	66,34475	213,774	232,7914	690	10,49891	2221,343	1259,471
10	25,16173	109,5287	117,7358	210,741	251,5513	704	10,71844	2224,463	1349,401
11	36,73379	77,63359	109,878	207,0471	300	720	11,29253	2328,642	1710,688
12	65,48024	59,20995	113,0158	214,158	300	740	11,864	2416,47	1741,111
13	26,93598	20	157,7717	209,8993	300	704	10,60693	2302,538	1742,461
14	27,20152	100,5523	115,7805	212,8253	243,9058	690	10,26543	2099,658	1274,508
15	23,29678	100,0722	30	210,7026	300	654	10,07149	2015,732	1754,5
16	75	113,0276	45,33705	125,1817	228,995	580	7,541261	1859,927	1027,302
17	12,87276	100,2285	112,1672	40	300	558	7,268459	1757,69	1504,522
18	69,59884	97,90876	102,999	207,1494	138,2372	608	7,893116	1851,373	781,6156
19	71,96935	22,32406	115,8615	214,2703	238,6899	654	9,115088	2015,594	1185,103
20	59,7922	113,7473	175	134,4451	231,3607	704	10,34528	2289,457	1181,669
21	35,83584	108,1251	115,4153	206,5162	224,0036	680	9,895987	2074,23	1149,67
22	11,89167	88,6857	171,6191	202,6829	137,8311	605	7,710509	1947,043	826,3481
23	21,56562	97,81125	114,3146	160,1217	139,0553	527	5,868464	1718,464	625,6428
24	75	21,77261	30	41,77049	300	463	5,543096	1591,948	1435,076
Cout totale=46672.5\$ ; Emission totale= 27641.2 Ib ; Perte totale =198.278 MW									

Tableau III.6 Les résultats obtenus pour la méthode TSA

**d.Méthode Algorithme de luciole (FFA):**

Nous présentons dans le tableau III.7 les résultats de l'algorithme de la luciole FFA:

H	P1(MW)	P2(MW)	P3(MW)	P4(MW)	P5(MW)	Pcharge	Perte	Cout	Emission
1	19,20393	101,3583	110,1013	40,63259	142,3517	410	3,647865	1306,438	509,9129
2	13,5152	47,20116	113,1773	212,4193	52,97922	435	4,29223	1467,648	524,8231
3	22,29915	99,93399	171,2436	45,79406	140,607	475	4,877811	1602,979	617,3354
4	18,96785	60,91442	105,0217	123,5033	227,5903	530	5,99763	1735,221	865,8232

5	25,50881	101,5741	90,5326	209,1015	138,0938	558	6,810818	1767,446	744,025
6	45,12719	99,25409	113,8785	125,9156	231,6809	608	7,856204	1812,383	989,4732
7	39,52219	37,69733	113,712	213,2453	230,1935	626	8,370342	1948,229	1088,084
8	25,24217	24,20355	112,6441	209,8147	291,4548	654	9,359288	2062,222	1126,804
9	70,45106	94,1332	105,1593	201,5458	228,8651	690	10,15449	2075,081	1156,315
10	24,79972	79,33228	102,3969	209,811	298,5417	704	10,88164	2272,749	1694,113
11	60,81106	94,29771	70,19573	209,7278	296,515	720	11,54738	2386,643	1714,292
12	44,68981	108,99	173,2861	124,7362	299,999	740	11,7011	2423,428	1722,198
13	13,11853	101,3345	125,13	244,5067	230,671	704	10,76076	2193,506	1325,838
14	39,57878	87,06837	131,838	211,4361	230,1586	690	10,07994	2153,62	1175,689
15	60,21785	73,85592	174,3162	124,9697	229,5257	654	8,885355	2121,855	1057,853
16	50,58127	22,63828	112,6961	209,8132	191,4094	580	7,13835	1932,397	871,6703
17	48,76286	79,96294	81,47027	125,0031	229,5163	558	6,715431	1905,084	908,4722
18	19,30277	111,5123	129,4418	125,3102	230,3285	608	7,895433	1888,124	1025,413
19	56,0381	48,53486	117,5775	211,3928	229,5283	654	9,071555	2060,774	1107,205
20	61,41316	54,89936	159,131	209,3771	229,5029	704	10,32353	2306,248	1190,22
21	73,64079	98,54764	112,7614	124,8966	280,1375	680	9,983987	2123,421	1403,31
22	39,79293	82,2176	114,2229	124,9378	251,6498	605	7,820974	2003,977	1081,438
23	39,99508	104,7255	32,79423	126,0108	229,8263	527	6,351866	1646,117	968,0881
24	17,00453	61,82148	125,6119	124,7839	138,2	463	4,421789	1585,045	503,8562
Cout totale=45474.2\$ ; Emission totale= 24862.3Ib ; Perte totale=191.298 (MW)									

**Tableau III.7** Les résultats obtenus pour la méthode FFA

Le tableau III.8 regroupe les résultats comparatifs entre les valeurs optimisées.

Les méthodes	Cout totale (\$)	Emission totale(Ib)	Perte totale (MW)
<b>Méthode (CSA)</b>	42425.455	21960.553	193.393
<b>Méthode (SOA)</b>	48609.77	32652.86	204.66
<b>Méthode (TSA)</b>	46672.4787	27641.23	198.278
<b>Méthode (FFA)</b>	45474.198	24862.338	191.298

**Tableau III.8**Résultats comparatifs

En analysant les résultats donnés par le tableau III.8, nous remarquons que le coût total trouvé par l'algorithme CSA qui est égal à est de 42425.455\$ est plus réduit par rapport à celui trouvé par les algorithmes SOA TSA et FFA qui est estime à 48609.77 \$, 46672.4787\$, 45474.198\$.

D’après les résultats donnés par le tableau III.8, nous remarquons que les émissions total trouvé par l’algorithme CSA qui est égal à est de 21960.553 Ib est plus réduit par rapport à celui trouvé par les algorithmes SOA, TSA et FFA qui est estime à 32652.86 Ib, 27641.23 Ib, 24862.338 Ib.

L’algorithme CSA qui offre le meilleur coût de production, offre en contre partie des pertes de transmission relativement plus élevées que FFA, soit 193.393 MW pour l’algorithme de CSA , 204.66MW pour SOA, 198.278 MW pour TSA et 191.298 MW pour FFA.

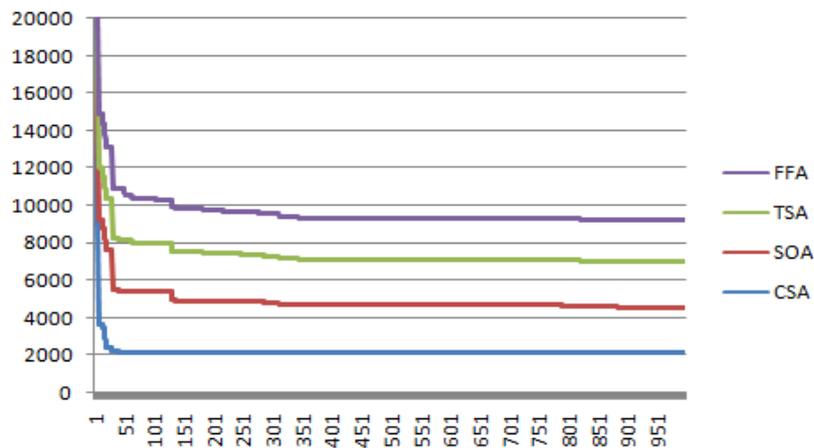


Fig III.1 courbe pour cout à la charge=740 MW

### III.2.2 Dispatching Environnemental dynamique avec les pertes variables (ED)

Les fonctions objectives des émissions exprimées en Ib par heures sont données par les expressions suivantes:

$$E_{i,t}(P_{i,t}) = (\alpha_i P_{i,t}^2 + \beta_i P_{i,t} + \gamma_i + \eta_i \exp(\delta_i P_{i,t})) \tag{III.2}$$

#### a. Méthode Algorithme de recherche de corbeau (CSA) :

Nous présentons dans le tableau III.9 les résultats de l’algorithme CSA:

H	P1(MW)	P2(MW)	P3(MW)	P4(MW)	P5(MW)	Pcharge	Perte	Emission	Cout
1	54,67862	58,23559	116,5715	110,5982	73.364	410	3,447958	352,4528	1723,627
2	58,06713	62,38344	121,8515	117,9818	78.601	435	3,885448	385,9599	1784,169
3	63,52586	69,08035	130,2206	129,7507	87.063	475	4,64133	446,6424	1912,114
4	71,12038	78,4295	141,5517	145,8015	98.890	530	5,793613	544,6478	2135,506
5	75,03238	83,26247	147,2326	153,8953	105,008	558	6,430753	601,2083	2203,616
6	82,10686	92,0343	157,2178	168,1757	116,1191	608	7,653807	713,8008	2241,188
7	84,68439	95,24133	160,7598	173,2525	120,183	626	8,121003	758,078	2229,487

8	88,729	100,2861	166,2125	181,0719	126,5767	654	8,876185	831,0191	2240,807
9	93,99597	106,8797	173,1187	190,9702	134,9336	690	9,898208	932,292	2270,804
10	96,06553	109,4783	175,7725	194,7682	138,2267	704	10,31124	974,0215	2274,187
11	98,44592	112,4724	178,7834	199,0718	142,0205	720	10,794	1023,365	2305,246
12	101,4445	116,2528	182,5145	204,3931	146,8088	740	11,41358	1087,577	2365,552
13	96,06556	109,4785	175,7724	194,7681	138,2266	704	10,31124	974,0215	2274,19
14	93,99597	106,8798	173,1187	190,9701	134,9336	690	9,898208	932,292	2270,804
15	88,72879	100,2861	166,2126	181,0718	126,5769	654	8,876184	831,0191	2240,804
16	78,13039	87,09846	151,652	160,2085	109,866	580	6,955391	648,8916	2233,356
17	75,03251	83,26242	147,2327	153,8951	105,008	558	6,430753	601,2083	2203,615
18	82,10695	92,03412	157,2178	168,1758	116,1191	608	7,653807	713,8008	2241,189
19	88,72897	100,2861	166,2125	181,0718	126,5768	654	8,876185	831,0191	2240,807
20	96,06563	109,4783	175,7724	194,7682	138,2268	704	10,31124	974,0215	2274,187
21	92,52516	105,0358	171,212	188,2392	132,5964	680	9,608534	903,2978	2265,782
22	81,67887	91,50247	156,6248	167,3259	115,4452	605	7,577329	706,6168	2241,906
23	70,70337	77,91507	140,9394	144,931	98,23858	527	5,727376	538,858	2126,253
24	61,88312	67,06283	127,7207	126,2268	84,51384	463	4,407324	427,5144	1850,281
Cout totale=51149.5 \$ ; Emission totale= 17733.6 Ib ; Perte totale =187.901 MW									

**Tableau III.9** Les résultats obtenus pour la méthode Algorithme de recherche de corbeau**b. Algorithme d'optimisation de mouette (SOA):**

Nous présentons dans le tableau III.10 les résultats de l'algorithme SOA:

H	P1(MW)	P2(MW)	P3(MW)	P4(MW)	P5(MW)	Pcharge	Perte	Emission	Cout
1	56,80166	77,33698	168,1004	47,22855	64,1807	410	3,648264	435,6455	1724,603
2	10	61,42367	121,8705	166,9573	78,80467	435	4,05613	456,8399	1817,803
3	75	44,67558	160,5058	77,46035	122,0379	475	4,679715	516,6437	2002,468
4	73,85888	109,4026	175	127,113	50,60292	530	5,977386	619,7103	1772,674
5	73,23114	82,24195	159,4532	182,1581	67,47565	558	6,560062	633,7188	1994,614
6	75	69,59599	161,4064	183,0695	126,5554	608	7,627297	727,6797	2263,074
7	64,68968	93,39199	175	196,3184	104,7863	626	8,186361	783,023	2232,087
8	75	101,9043	155,5031	200,7885	129,7372	654	8,933097	842,4547	2157,095
9	75	108,3677	171,6668	206,4414	138,4728	690	9,948731	958,8998	2150,044
10	74,58614	125	175	214,7268	125,1289	704	10,44186	1002,685	2344,968
11	74,52077	104,9059	175	216,9163	159,4582	720	10,8012	1045,847	2333,441
12	75	114,4371	173,3374	223,0289	165,6491	740	11,45238	1116,07	2504,33

13	75	115,6677	174,9069	224,0869	124,7782	704	10,43969	1004,073	2370,264
14	73,75475	91,35562	175	203,3775	156,3736	690	9,861509	954,2253	2244,522
15	75	104,1984	175	175,9484	132,6925	654	8,839301	859,0802	2215,624
16	62,40935	90,83947	159,1414	132,4058	142,115	580	6,911073	682,3087	1968,234
17	62,60168	72,65742	150,6241	175,237	103,3214	558	6,441642	621,8534	2298,012
18	70,446	113,1693	162,1669	142,0179	127,8705	608	7,670479	735,3927	2172,466
19	74,2609	90,1974	159,5851	199,1525	139,6887	654	8,884657	845,6656	2123,564
20	72,04745	108,1765	168,1249	229,1019	136,9771	704	10,4278	1011,293	2311,295
21	55,42704	87,85559	175	237,7282	133,7291	680	9,739868	977,1968	2326,101
22	15,31108	125	173,4614	165,0245	133,9637	605	7,760671	828,6102	2157,097
23	70,89269	68,53032	157,4312	176,9506	59,05787	527	5,862771	574,295	2096,969
24	11,95141	83,16798	163,3074	145,3992	63,78788	463	4,613911	509,9076	1803,953
Cout totale=51385.3 \$ ; Emission totale= 18743.1 Ib ; Perte totale =189.766 MW									

Tableau III.10 Les résultats obtenus pour la méthode SOA

**c. Algorithme d'essaim de tuniciers (TSA):**

Nous présentons dans le tableau III.11 les résultats de l'algorithme TSA:

H	P1(MW)	P2(MW)	P3(MW)	P4(MW)	P5(MW)	Pcharge	Perte	Emission	Cout
1	25,91916	40,10573	117,3242	126,4625	103,6315	410	3,443112	387,8494	1605,69
2	25,89697	66,0971	106,6346	136,4707	103,8356	435	3,934976	428,3494	1764,519
3	75	73,38502	144,3405	103,243	83,70104	475	4,669547	470,8061	2016,49
4	75	75,74827	137,5107	147,8684	99,66794	530	5,795319	545,52	2129,551
5	52,27358	99,00597	139,2901	168,7723	105,1687	558	6,510574	620,0196	2169,803
6	70,80055	87,42708	161,8729	156,5098	138,9919	608	7,602177	727,4557	2134,696
7	75	101,0449	145,7464	182,1376	130,2265	626	8,155422	767,1428	2166,817
8	75	81,7734	175	183,2326	147,8082	654	8,814195	849,9285	2254,575
9	70,10642	112,0273	175	203,6995	139,1004	690	9,933655	948,5334	2182,917
10	75	125	175	212,6286	126,7942	704	10,4229	995,4907	2319,088
11	75	123,092	169,7297	223,9407	139,1571	720	10,91944	1052,724	2358,382
12	75	115,5667	175	213,8232	172,0323	740	11,42222	1115,217	2467,091
13	59,3802	116,6789	167,8331	234,0376	136,5588	704	10,48863	1028,933	2407,578
14	63,70816	125	145,2238	219,103	147,0492	690	10,08417	982,5467	2352,122
15	45,83753	85,93059	175	192,5166	163,5574	654	8,842136	906,9634	2325,859
16	27,76229	109,3399	140,0134	191,8435	118,1962	580	7,155388	722,8087	2123,591
17	75	83,72602	141,0007	135,8926	128,7656	558	6,384966	627,8271	1983,869
18	64,45975	103,1383	154,038	162,6533	131,3825	608	7,671784	737,8577	2186,914

19	75	93,08616	175	159,7027	160,0138	654	8,802665	858,5933	2294,213
20	64,29375	114,8233	175	218,6207	141,6667	704	10,40444	1013,093	2270,718
21	75	85,51024	148,9881	200,5205	179,5992	680	9,618063	969,0975	2392,676
22	36,1936	52,95419	175	208,3331	140,1371	605	7,617982	807,522	2026,562
23	66,48809	27,02027	149,5637	165,3027	124,3287	527	5,703444	600,7798	2058,392
24	58,31418	85,8049	126,705	108,4238	88,18007	463	4,427902	437,7679	1886,609
Cout totale=51878.7 \$ ; Emission totale=18602.8 Ib ; Perte totale =188.825 MW									

Tableau III.11 Les résultats obtenus pour la méthode TSA

**d.Méthode Algorithme de luciole (FFA):**

Nous présentons dans le tableau III.12 les résultats de l'algorithme de la luciole FFA:

H	P1(MW)	P2(MW)	P3(MW)	P4(MW)	P5(MW)	Pcharge	Perte	Emission	Cout
1	43,60271	61,15663	99,7588	94,31998	114,5988	410	3,43697	382,2801	1844,563
2	35,25743	33,08445	146,8149	140,5947	83,15972	435	3,911208	422,4015	1876,908
3	27,72626	91,41255	145,1788	158,8472	56,68221	475	4,847024	501,9786	1851,856
4	50,56428	93,636	123,3218	136,1698	132,1228	530	5,814583	576,5831	1808,614
5	36,46583	96,95828	152,2766	146,0946	132,6604	558	6,455678	644,6478	1992,042
6	56,41988	106,4025	148,9569	211,8681	92,23396	608	7,881302	760,8966	2201,711
7	53,71132	38,97318	155,5166	222,0711	163,9046	626	8,176781	884,1328	2315,641
8	67,80126	53,60999	151,4631	209,0526	180,9499	654	8,876863	931,0491	2369,31
9	42,94346	104,4151	134,8563	234,0032	183,9136	690	10,13172	1068,43	2466,537
10	21,14631	98,08323	167,755	242,561	184,9788	704	10,5244	1155,329	2465,426
11	61,96011	44,77608	173,9061	211,9732	238,1678	720	10,78335	1284,139	2359,787
12	37,0138	119,0104	164,4168	235,9464	195,2104	740	11,59782	1230,397	2675,673
13	46,20626	66,45373	157,9158	235,4571	208,3637	704	10,39662	1160,033	2589,019
14	50,8126	92,60171	152,8327	201,5278	202,1578	690	9,932581	1050,528	2387,185
15	71,99817	62,22002	140,4111	214,2174	174,0742	654	8,920839	916,2278	2350,721
16	26,32747	100,4619	154,1994	159,7438	146,2837	580	7,016224	722,5204	2059,318
17	39,70661	91,35604	163,2771	164,64	105,5098	558	6,489647	631,8468	2214,47
18	56,41988	106,4025	148,9569	211,8681	92,23396	608	7,881302	760,8966	2201,711
19	43,39738	95,54834	144,3114	214,1224	165,6137	654	8,993286	915,0649	2238,012
20	21,14631	98,08323	167,755	242,561	184,9788	704	10,5244	1155,329	2465,426
21	53,77089	110,6339	132,4731	180,856	211,9925	680	9,726535	1054,69	2405,526
22	32,42514	118,2766	122,2273	165,4871	174,3317	605	7,747794	832,0366	2269,25
23	25,35644	42,25496	128,5336	221,4854	115,3307	527	5,961174	668,013	1977,457
24	64,22946	83,22532	107,6578	68,20295	144,163	463	4,4786	512,8564	1721,865

Cout totale=51263.5 \$ ; Emission totale= 19840 Ib ; Perte totale =187.07 MW

**Tableau III.12** Les résultats obtenus pour la méthode FFA

Le tableau III.13 regroupe les résultats comparatifs entre les valeurs optimisées.

Les méthodes	Emission totale (Ib)	Cout totale (\$)	Perte totale (MW)
Méthode (CSA)	17733.6	51149.5	187.901
Méthode (SOA)	18743.1	51385.3	189.766
Méthode (TSA)	18602.8	51878.7	188.825
Méthode (FFA)	19840.2	51263.5	187.07

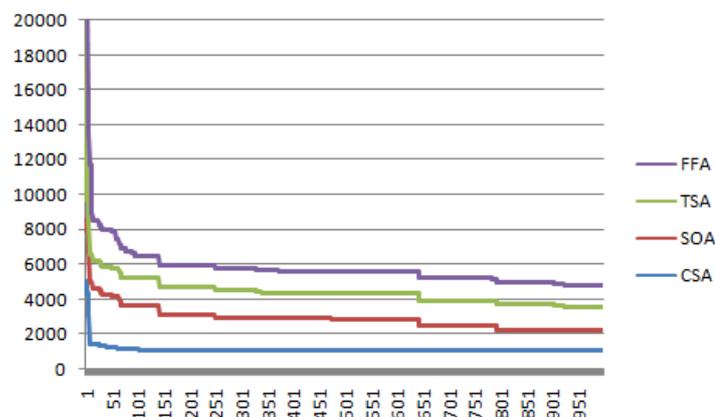
**Tableau III.13** Résultats comparatifs

En analysant les résultats données par le tableau III.13, nous remarquons que les émissions total trouvé par l'algorithme CSA qui est égal à est de 17733.6 Ib est plus réduit par rapport à celui trouvé par les algorithmes SOA TSA et FFA qui est estime à 18743.1 Ib , 18602.8 Ib et 19840.2 Ib.

D'après les résultats données par le tableau III.13, nous remarquons que le cout total calculé par l'algorithme CSA qui est égal à est de 51149.5\$est plus réduit par rapport à celui trouvé par les algorithmes SOA, TSA et FFA qui est estime à 51385.3 \$ , 51878.7 \$ et 51263.5 \$.

L'algorithme CSA qui offre le meilleur coût de production, offre en contre partie des pertes de transmission relativement plus élevées que FFA, soit 193.393 MW pour l'algorithme de CSA , 204.66MW pour SOA, 198.278 MW pour TSA et 191.298 MW pour FFA.

L'algorithme CSA qui offre le meilleure émission de gaz toxique, offre en contre partie des pertes de transmission relativement plus élevées que FFA, soit 187.901 MWpour l'algorithme de CSA , 204.66MW pour SOA, 188.825 MW pour TSA et 187.07 MW pour FFA.



**Fig. III.2.** Courbe pour Emission à la charge=740 MW

### III.2.3 Dispatching Economique Environnemental Combiné dynamique avec les pertes variables (CEED)

Dans ce cas, nous traitons le dispatching économique/environnemental combinées avec les pertes variable en introduisant le facteur de pénalité de prix sous l'effet des ouvertures des vannes. Les pertes électriques sont variables en fonctions des puissances générées.

Les fonctions objectives des coûts et des émissions combinées exprimées en dollars par heures sont données par les expressions suivantes:

$$F_{i,t}(P_{i,t}) + E_{i,t}(P_{i,t}) = (a_i P_{i,t}^2 + b_i P_{i,t} + c_i + |e_i \times \sin(f_i \times (P_{i,min} - P_{i,t}))|) + (\alpha_i P_{i,t}^2 + \beta_i P_{i,t} + \gamma_i + \eta_i \exp(\delta_i P_{i,t}))$$

(III.3)

#### a. Méthode Algorithme de recherche de corbeau (CSA) :

Les valeurs optimales des puissances générées, des pertes actives transmises, du coût total, du cout de production, des émissions sont données par le tableau III.14.

H	P1(MW)	P2(MW)	P3(MW)	P4(MW)	P5(MW)	Pcharge	Perte	Cout+Emission	Cout	Emission
1	23,221	95,066	118,179	125,507	51,646	410,000	3,619	1814,930	1343,493	400,897
2	42,464	84,193	112,348	124,800	75,142	435,000	3,946	2078,248	1607,846	399,951
3	46,363	94,690	111,484	87,447	139,713	475,000	4,697	2254,467	1670,469	519,509
4	26,838	99,530	114,280	126,507	168,745	530,000	5,899	2475,641	1745,194	669,218
5	33,564	92,891	123,185	175,142	139,753	558,000	6,536	2649,852	1902,752	665,000
6	73,336	98,983	157,530	146,016	139,760	608,000	7,625	2893,762	2031,153	728,731
7	57,750	99,767	123,604	210,658	142,515	626,000	8,294	2838,928	1914,203	816,010
8	51,884	88,674	103,432	204,515	214,625	654,000	9,131	3248,435	2131,606	1040,647
9	53,142	97,776	114,207	209,557	225,462	690,000	10,144	3260,318	2019,883	1153,740
10	54,142	109,352	112,662	209,810	228,632	704,000	10,598	3370,372	2075,861	1204,781
11	73,636	100,102	117,640	210,105	229,527	720,000	11,010	3381,405	2052,002	1223,114
12	72,069	116,167	123,841	210,056	229,502	740,000	11,635	3613,856	2224,386	1274,535
13	66,908	98,680	173,744	145,451	229,517	704,000	10,300	3523,693	2236,929	1154,978
14	61,029	98,536	102,648	209,815	228,176	690,000	10,204	3293,254	2036,995	1171,356
15	66,581	117,798	115,698	213,160	149,886	654,000	9,123	3098,415	2078,304	903,466
16	22,005	101,970	112,492	124,576	226,155	580,000	7,200	2700,991	1701,781	948,532
17	24,788	95,120	109,230	195,916	139,614	558,000	6,667	2521,975	1740,035	706,614

18	66,807	83,120	156,707	124,906	184,080	608,000	7,620	3116,141	2199,492	803,606
19	61,025	110,762	116,453	209,863	164,984	654,000	9,087	3145,552	2122,958	915,351
20	61,190	96,517	118,054	209,294	229,473	704,000	10,528	3333,294	2049,371	1189,705
21	61,279	77,640	112,693	208,694	229,519	680,000	9,825	3280,867	2055,372	1140,486
22	66,280	38,471	112,702	209,818	185,459	605,000	7,729	3013,830	2051,944	873,120
23	47,683	93,663	126,697	124,929	139,771	527,000	5,741	2357,800	1693,775	582,538
24	41,562	70,450	112,609	124,880	117,903	463,000	4,404	2212,678	1696,097	452,880
(Cout+Emission )totale=69478.7 ; Cout totale=463881.9 \$ ; Emission totale= 20938.8 Ib ; Perte=191.563MW										

**Tableau III.14** Les résultats obtenus pour la method Algorithme de recherche de corbeau

**b. Algorithme d'optimisation de mouette (SOA):**

Nous présentons dans le tableau III.15 les résultats de l’algorithme SOA:

H	P1(MW)	P2(MW)	P3(MW)	P4(MW)	P5(MW)	Pcharge	Perte	Cout+Emission	Cout	Emission
1	29,820	24,525	175,000	40,000	144,308	410	3,652574	2106,61	1477,15	535,54
2	22,870	99,305	31,426	208,711	77,283	435	4,59495	2184,56	1514,50	608,30
3	61,587	125,000	31,440	212,466	50,000	475	5,492908	2420,18	1632,59	701,36
4	72,448	92,820	31,899	199,340	139,844	530	6,350623	2514,48	1673,98	746,22
5	10,000	32,977	175,000	208,954	137,637	558	6,567746	2680,73	1772,19	781,84
6	58,294	99,122	172,287	146,574	139,345	608	7,622273	2919,44	2036,81	741,72
7	11,931	89,961	172,762	217,229	142,401	626	8,282553	3026,37	1966,70	894,52
8	65,403	102,605	175,000	199,297	120,624	654	8,929393	3216,25	2207,43	848,03
9	60,813	87,536	165,898	213,120	172,499	690	9,865438	3537,71	2377,28	988,98
10	12,681	99,787	113,109	249,833	239,452	704	10,86289	3649,16	2181,40	1386,92
11	26,974	121,718	149,604	216,640	216,089	720	11,02516	3769,08	2419,26	1234,47
12	72,548	20,053	174,731	250,000	234,189	740	11,52101	3997,78	2413,71	1418,81
13	63,704	85,402	175,000	250,000	140,338	704	10,44453	3591,03	2372,23	1049,32
14	75,000	71,831	175,000	211,233	166,784	690	9,846898	3484,29	2344,13	980,36
15	75,000	125,000	175,000	205,286	82,822	654	9,108651	3377,31	2314,46	879,15
16	26,169	125,000	174,616	211,024	50,666	580	7,474805	2911,45	1950,54	808,03
17	19,761	101,756	170,101	139,243	133,663	558	6,523931	2695,48	1881,31	687,16
18	63,148	113,921	175,000	212,051	51,901	608	8,019919	2999,55	2017,59	809,26
19	10,000	125,000	175,000	215,069	138,123	654	9,192217	3204,99	2075,54	981,92
20	53,974	95,859	166,957	208,188	189,326	704	10,30337	3578,12	2375,07	1056,65

21	46,861	80,740	175,000	245,929	141,234	680	9,764575	3512,33	2350,48	1002,47
22	71,100	86,466	106,827	211,088	140,364	605	10,84461	2977,82	1860,07	780,63
23	75,000	101,469	174,214	128,687	53,504	527	5,87401	2513,86	1751,38	592,40
24	10,000	26,075	175,000	206,721	50,000	463	4,795757	2278,76	1534,95	616,03
(Cout+Emission) totale=73147.3 ; Cout totale=48500.8 \$ ; Emission totale= 21130.1 Ib ; Perte=196.961MW										

**Tableau III.15** Les résultats obtenus pour la méthode SOA

**c. Algorithme d'essaim de tuniciers (TSA):**

Nous présentons dans le tableau III.16 les résultats de l'algorithme TSA:

H	P1(MW)	P2(MW)	P3(MW)	P4(MW)	P5(MW)	Pcharge	Perte	Cout+Emission	Cout	Emission
1	15,849	103,096	37,234	208,041	50,000	410	4,22	1978,33	1334,07	585,52
2	26,887	100,421	112,565	124,988	74,149	435	4,01	2010,09	1500,94	429,65
3	75,000	105,154	113,134	125,212	61,259	475	4,76	2158,61	1555,61	482,74
4	59,295	94,756	120,037	123,170	138,537	530	5,79	2356,89	1671,45	583,43
5	12,231	92,987	113,315	211,220	135,007	558	6,76	2466,58	1641,31	742,28
6	44,933	105,181	113,196	211,601	140,970	608	7,88	2761,90	1837,90	801,14
7	54,473	106,807	116,059	213,633	143,371	626	8,34	2889,43	1938,01	833,53
8	25,912	94,715	108,796	203,312	230,467	654	9,20	3138,52	1941,06	1131,97
9	25,386	99,283	130,068	215,006	230,401	690	10,14	3392,41	2076,37	1209,74
10	48,170	105,104	115,813	209,795	235,714	704	10,60	3432,92	2111,25	1234,38
11	25,053	92,515	173,833	212,331	227,166	720	10,90	3592,97	2192,10	1260,48
12	43,790	96,899	174,600	210,252	225,911	740	11,45	3668,79	2251,26	1267,87
13	75,000	110,942	171,889	126,646	229,905	704	10,38	3512,13	2180,87	1169,21
14	38,214	92,944	117,208	222,798	229,021	690	10,18	3405,36	2112,92	1199,94
15	14,371	115,419	174,998	215,115	143,221	654	9,13	3177,77	2069,19	964,16
16	36,394	94,031	30,000	204,142	223,165	580	7,73	2952,40	1810,85	1079,63
17	61,748	106,636	109,781	59,181	227,423	558	6,77	2869,63	1855,31	935,65
18	13,824	99,241	156,922	206,057	139,885	608	7,93	2896,38	1881,93	830,21
19	13,320	102,125	175,000	209,920	162,650	654	9,01	3210,72	2078,76	973,53
20	75,000	101,863	144,323	212,721	180,470	704	10,38	3521,07	2351,98	1030,77
21	75,000	48,259	116,680	221,314	228,557	680	9,81	3426,70	2148,52	1166,36
22	28,044	102,547	56,195	206,210	220,256	605	8,25	3129,91	1982,46	1076,72
23	10,000	29,713	147,105	117,465	228,597	527	5,88	2721,51	1758,37	902,19
24	12,907	97,581	175,000	130,989	51,212	463	4,69	2188,32	1533,79	533,75
(Cout+Emission) totale=70859.3 ; Cout totale=45816.3 \$ ; Emission totale= 22424.8 Ib ; Perte=194.204										

**Tableau III.16** Les résultats obtenus pour la méthode TSA**d.Méthode Algorithme de luciole (FFA):**

Nous présentons dans le tableau III.17 les résultats de l'algorithme de la luciole FFA:

H	P1(MW)	P2(MW)	P3(MW)	P4(MW)	P5(MW)	Pcharge	Perte	Cout+Emission	Cout	Emission
1	48,870	78,518	112,933	123,216	50,000	410	3,54	1884,81	1442,66	368,02
2	27,423	89,520	147,199	124,898	50,012	435	4,05	2084,34	1556,39	434,74
3	35,474	99,681	31,947	209,812	103,403	475	5,32	2382,32	1666,50	660,13
4	28,745	94,802	112,489	160,092	139,781	530	5,91	2440,00	1886,71	609,54
5	49,720	54,457	172,891	147,540	139,764	558	6,37	2831,79	2059,04	653,58
6	44,224	98,897	117,894	125,107	229,717	608	7,84	2856,37	1814,56	979,25
7	60,545	30,586	106,964	208,713	227,563	626	8,37	3079,16	1938,05	1068,33
8	54,169	111,041	112,780	155,571	229,529	654	9,09	3266,71	2120,27	1070,58
9	49,317	97,181	116,310	208,359	228,975	690	10,14	3413,56	2020,60	1167,87
10	41,438	90,033	145,063	209,817	228,080	704	10,43	3504,57	2214,63	1189,40
11	50,583	98,516	143,205	209,248	229,362	720	10,91	3530,60	2203,29	1220,63
12	69,792	103,846	163,603	209,810	204,363	740	11,41	3721,70	2404,82	1170,39
13	45,865	98,832	120,386	219,306	230,187	704	10,58	3433,57	2125,72	1219,03
14	28,202	98,028	142,775	209,522	221,547	690	10,07	3401,18	2147,45	1157,69
15	22,642	113,327	172,946	124,905	229,208	654	9,03	3268,32	2044,00	1118,31
16	15,057	98,525	142,320	209,579	121,740	580	7,22	2759,39	1896,17	758,75
17	55,173	29,291	113,452	135,962	230,705	558	6,58	2766,90	1804,38	908,50
18	15,802	96,735	92,218	209,786	201,536	608	8,08	3026,88	1984,61	982,66
19	40,332	106,765	124,801	161,658	229,499	654	9,05	3324,60	2175,54	1073,32
20	33,000	112,932	133,094	208,005	227,524	704	10,55	3512,82	2205,03	1214,07
21	18,766	98,541	153,149	189,826	229,478	680	9,76	3429,26	2166,87	1167,00
22	56,897	102,567	103,913	209,786	139,672	605	7,84	2727,16	1844,00	787,44
23	38,339	93,312	110,462	150,947	139,747	527	5,81	2431,20	1762,89	598,41
24	19,972	93,917	109,819	124,905	118,886	463	4,50	2097,85	1546,56	491,98
(Cout+Emission ) totale=71175.1 ; Cout totale=47030.7 \$ ; Emission totale= 22069.6 Ib ; Perte=192.451										

**Tableau III.17** Les résultats obtenus pour la méthode FFA

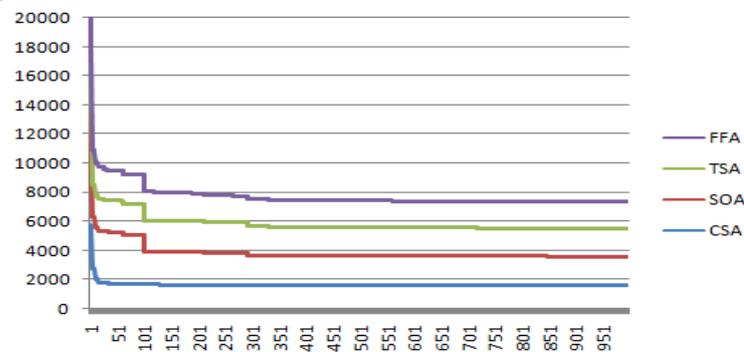
Les méthode	(Cout +Emission )totale	Cout totale (\$)	Emission totale (Ib)	Perte totale (MW)
<b>Méthode (CSA)</b>	69478.7	46381.9	20938.8	192.386
<b>Méthode(SOA)</b>	73147.3	48500.8	21130.1	196.961

<b>Méthode (TSA)</b>	70859.3	45816.3	22424.8	194.204
<b>Méthode (FFA)</b>	71175.1	47030.7	22069.6	192.451

**Tableau III.18 :** Résultats du dispatching économique-environnemental CEED

En analysant les résultats données par le tableau III.18 de cas de dispatching économique/environnemental, nous pouvons constater que le coût total est réduit lors de l'application de CAA (69478.7\$) par rapport à celui trouvé par les algorithmes SOA, TSA et FFA qui est estimé à 73147.3\$ , 70859.3\$ et 71175.1\$.

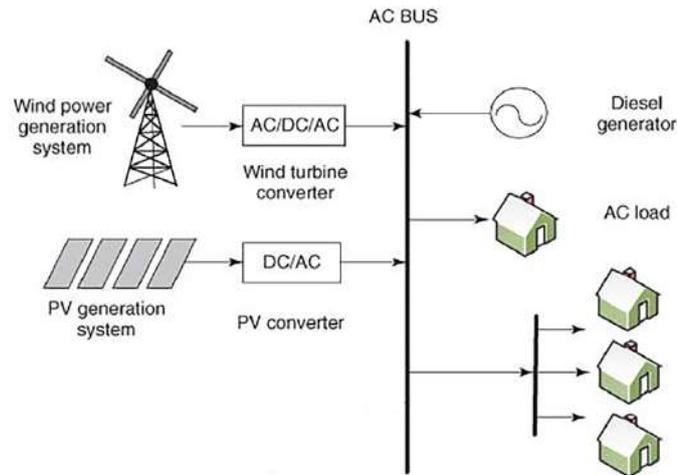
D'après les résultats donnés par le tableau V.18, nous déduisons que la minimisation du coût de production et celle des émissions sont des objectifs contradictoires. Les émissions sont maximales lorsque le coût de production est minimal.



**Fig III.3.** courbe pour Cout+Emission à la charge=740 MW

### III.3 PARTIE N°2

Dans cette partie application, la méthode de l'algorithme de recherche de corbeau est appliqué pour résoudre le problème du dispatching environnemental dynamique en tenant compte les ressources des énergies renouvelable (PV, éolienne et batterie) intégrés dans un micro réseau isolé (an islanded microgrid.) à trois générateurs dont le schéma unifilaire est illustré par la figure III.4 et on le compare les résultats trouve rpar CSA avec trois méthodes métaheuristiques (SOA, TSA et FFA). Le tableau III.20 regroupe les valeurs des coefficients des fonctions coût, les émissions, des 03 générateurs et les puissances limites  $P_{max}$  et  $P_{min}$ .



**Fig III4.** Architecture d'un micro-réseau Isolé

Le tableau III.19 représente la courbe de charge, la puissance générée par la centrale PV, la puissance générée par la centrale éolienne et la puissance fournie par la batterie avec la variation journalière (24h).

H	Pwind	Ppv	Pbatt	Pcharge	H	pwind	Ppv	Pbatt	Pcharge
1	1,70	0	0,4	140	13	14,35	31,94	0,15	240
2	8,5	0	0,275	150	14	10,35	26,81	0	220
3	9,27	0	0,15	155	15	8,26	10,08	0	200
4	16,66	0	0	160	16	13,71	5,3	0,4	180
5	7,22	0	0	165	17	3,44	9,57	0,275	170
6	4,91	0,03	0,40	170	18	1,87	2,31	0,15	185
7	14,66	6,27	0,275	175	19	0,75	0	0	200
8	25,56	16,18	0,15	180	20	0,17	0	0	240
9	20,58	24,05	0	210	21	0,15	0	0,4	225
10	17,85	39,37	0	230	22	0,31	0	0,275	190
11	12,8	7,41	0,4	240	23	1,07	0	0,15	160
12	18,65	3,65	0,275	250	24	0,58	0	0	145

**Tableau III.19** Demande de charge pendant 24 heures (réseau à 03 unités) .

Le tableau III.20 regroupe les valeurs des coefficients des fonctions coût, les émissions, des 03 générateurs et les puissances limites  $P_{max}$  et  $P_{min}$ .

Donnée	Générateur 1	Générateur 2	Générateur 3
$U_i$	0.0024	0.0029	0.0210
$V_i$	21.00	21.16	20.40

$w_i$	1530	992	600
$x$	0.0105	0.008	0.012
$y_i$	-1.355	-0.6	-0.555
$z_i$	60	45	90
$P_i, \min$ (MW)	37	40	50
$P_i, \max$ (MW)	150	160	190
$h_i$	25.1597	11.9948	4.6750

**Tableau III.20** Données pour le système à 3 unités

La fonction de cout est donnée sous la forme suivante:

$$F(P) = \sum_{i=1}^{Ng} U_i P^2 + V_i P + w_i \quad \frac{\$}{h} \quad (\text{III.4})$$

La fonction des émissions est donnée sous la forme suivante:

$$E(P) = \sum_{i=1}^{Ng} x P^2 + y_i P + \frac{z_i \text{kg}}{h} \quad (\text{III.5})$$

La fonction de dispatching économique/ environnemental combinée est donnée sous la forme suivante:

$$C(P) = \sum_{i=1}^{Ng} (u_i P^2 + v P + w_i) + h_i \times (E(P)) \quad \text{kg/h} \quad (\text{III.6})$$

La fonction de cout de la centrale photovoltaïque est donnée sous la forme suivante:

$$F_{PV} = 547.7483 * p_{pv}$$

( III.7) La fonction de cout de la centrale éolienne est donnée sous

la forme suivante:

$$F_{WIND} = 153.3810 * p_{wind} \quad (\text{III.8})$$

La fonction de cout de la batterie est donnée sous la forme suivante:

$$F_{battery} = 119 * P_{battery} \quad (\text{III.9})$$

### III.3.1 Dispatching Economique dynamique (ELD) avec les ressources des énergies renouvelables (RES)

Les fonctions objectives des coûts exprimées en dollars par heures sont données par les expressions suivantes:

$$ELD_{RES} = \sum_{i=1}^{Ng} F(P) + F_{pv} + F_{wind} + F_{battery} \quad (\text{III.10})$$

#### a. Algorithme de recherche de corbeau (CSA) :

Nous présentons dans le tableau III.21 les résultats de l'algorithme CSA:

H	P1(MW)	P2(MW)	P3(MW)	Pcharge	Cout
1	47,82455	40,07502	50,00042	140	6365,307
2	51,16851	40,00016	50,05633	150	7464,097
3	55,55829	40,02164	50,00007	155	7659,829
4	53,32875	40,00702	50,00423	160	8727,844
5	65,88657	41,89334	50,00001	165	7587,502
6	74,61	40,05	50,00	170	7443,91
7	63,78038	40,00115	50,01347	175	12110,69
8	47,99552	40,11403	50,00045	180	18862,26
9	75,05039	40,31956	50,00005	210	22971,88
10	79,75287	43,02712	50,00002	230	31103,1
11	108,3551	61,03481	50,00001	240	13870,13
12	113,6156	63,80856	50,00082	250	12865,99
13	93,89194	49,66779	50,00026	240	26959,47
14	87,15793	45,68194	50,00013	220	23288,36
15	87,24313	44,41677	50,00001	200	13778,69
16	69,73796	40,85153	50,00051	180	11593,46
17	66,71082	40,0033	50,00088	170	12259,55
18	86,18544	44,48454	50,00001	185	8539,224
19	97,44185	51,80537	50,00274	200	7482,691
20	122,0877	67,74209	50,00019	240	8266,925
21	109,4654	64,98457	50,00007	225	7979,96
22	92,36339	47,05155	50,00006	190	7236,911
23	67,54989	41,22982	50,00029	160	6683,326
24	54,17399	40,24588	50,00013	145	6284,461

Cout totale=297385.55 \$
--------------------------

**Tableau III.21** Les résultats obtenus par la méthode CSA**b. Algorithme d'optimisation de mouette (SOA):**

Nous présentons dans le tableau III.22 les résultats de l'algorithme SOA:

H	P1(MW)	P2(MW)	P3(MW)	Pcharge	Cout
1	47,61976	40,27938	50	140	6366,176
2	43,15114	45,74963	52,32787	150	7471,866
3	46,07748	49,50439	50	155	7663,407
4	39,05622	54,28275	50	160	8731,867
5	57,75159	50,03089	50	165	7591,088
6	73,91	40,00	50,75	170	7444,90
7	63,20578	40,59003	50	175	12111,55
8	48,05747	40	50,05162	180	18863,2
9	62,38938	52,98059	50	210	22973,18
10	80,72557	40	52,05526	230	31106,28
11	128,4767	40,39231	50,52196	240	13874
12	98,64408	70,78416	57,99626	250	12876,02
13	86,55155	57,00841	50	240	26959,77
14	92,78681	40,05134	50	220	23290,3
15	61,03335	70,62894	50	200	13784,64
16	59,0837	51,50341	50	180	11597,55
17	66,06724	40	50,64809	170	12260,65
18	64,68588	65,98335	50	185	8542,521
19	109,2434	40	50	200	7489,953
20	145,8297	44,00536	50	240	8275,911
21	134,2535	40	50,19843	225	7985,084
22	78,46238	60,95834	50	190	7243,631
23	37,28888	71,49415	50	160	6693,531
24	54,30366	40,1165	50	145	6284,606
Cout totale=297481.69 \$					

**Tableau III.22** Les résultats obtenus pour la méthode SOA**c. Algorithme d'essaim de tuniciers (TSA):**

Nous présentons dans le tableau III.23 les résultats de l'algorithme TSA:

H	P1(MW)	P2(MW)	P3(MW)	Pcharge	Cout
1	39,49759	48,4026	50	140	6367,226
2	49,11755	42,10884	50	150	7465,77
3	38,0493	57,53022	50	155	7664,121
4	37	56,34015	50	160	8731,63
5	63,12008	44,66005	50	165	7587,914
6	72,70	41,96	50,00	170	7444,63
7	61,91102	41,88433	50	175	12111,22
8	39,40648	48,70429	50	180	18864,83
9	53,45297	61,91645	50	210	22975,64
10	77,32563	45,45402	50	230	31103,54
11	45,70609	123,6867	50	240	13893,35
12	130,3625	47,06354	50	250	12868,76
13	74,05163	69,50421	50	240	26965,57
14	79,26368	53,57592	50	220	23289,12
15	80,89096	50,76844	50	200	13779,48
16	70,58837	40	50	180	11595
17	64,14598	42,57009	50	170	12260,86
18	90,67124	40	50	185	8540,575
19	63,46555	85,78336	50	200	7489,585
20	85,80112	104,0301	50	240	8273,934
21	48,57582	120,5004	55,37652	225	8007,089
22	66,57263	72,84077	50	190	7241,732
23	57,57922	51,20178	50	160	6685,627
24	54,41931	40	50	145	6285,104
Cout totale=297492.31 \$					

**Tableau III.23** Les résultats obtenus pour la méthode TSA

**d. Méthode Algorithme de luciole (FFA):**

Nous présentons dans le tableau III.24 les résultats de l'algorithme de la luciole FFA:

H	P1(MW)	P2(MW)	P3(MW)	Pcharge	Cout
1	40,57711	43,58259	53,74031	140	6371,09
2	44,90219	44,37787	51,94495	150	7467,348
3	47,69546	47,79935	50,08518	155	7661,236

4	37,96796	55,12727	50,24477	160	8731,432
5	57,12192	47,1142	53,54389	165	7592,688
6	51,52	55,93	57,21	170	7455,80
7	60,85749	42,53292	50,40459	175	12111,42
8	43,80887	43,98508	50,31605	180	18863,38
9	70,32899	45,04032	50,00069	210	22972,16
10	54,66614	65,54547	52,56838	230	31109,7
11	58,06789	98,24656	63,07554	240	13896,4
12	46,79955	106,0614	74,56401	250	12917,36
13	78,59363	62,80216	52,1642	240	26962,87
14	73,15464	59,53021	50,15514	220	23289,65
15	57,57098	59,77678	64,31227	200	13801,27
16	44,79896	59,39853	56,39251	180	11605,42
17	60,44538	46,03769	50,23192	170	12260,46
18	58,10328	63,73801	58,82871	185	8553,506
19	84,94445	61,09772	53,20783	200	7486,736
20	104,1782	77,42387	58,22791	240	8276,624
21	92,03101	82,41034	50,00865	225	7981,78
22	73,72075	65,66827	50,02599	190	7238,583
23	51,15155	53,21376	54,41469	160	6690,891
24	49,66707	44,75293	50	145	6285,171
Cout totale=297582.97					

**Tableau III.24** Les résultats obtenus pour la méthode FFA

Le tableau III.25 regroupe les résultats comparatifs entre les valeurs optimisées.

Les méthodes	Cout totale (\$)
<b>Méthode (CSA)</b>	297385.55
<b>Méthode (SOA)</b>	297481.69
<b>Méthode (TSA)</b>	297492.31
<b>Méthode (FFA)</b>	297582.97

**Tableau III.25** Résultats comparatifs ELD

En analysant les résultats données par le tableau III.25, nous remarquons que le coût total trouvé par l'algorithme CSA qui est égal à est de 297385.55 \$est plus réduit par rapport à celui trouvé par les algorithmes SOA TSA et FFA qui est estime à 297481.69\$, 297492.31\$, 297582.97\$.

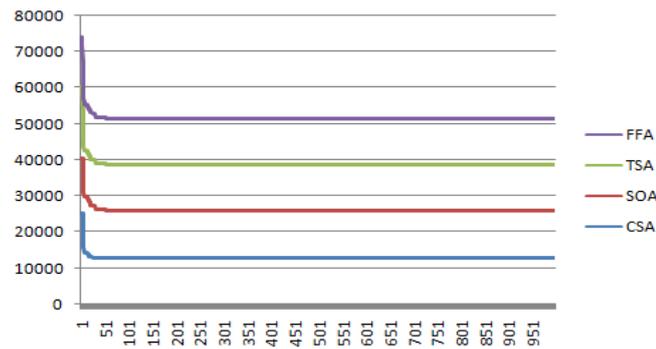


Fig.III.5 : Coût total de production à la charge=740 MW

### III.3.2 Dispatching Environnemental dynamique (ED) avec les ressources des énergies renouvelables (RES)

Les fonctions objectives des émissions exprimées en Kg par heures sont données par les expressions suivantes:

$$ED_{RES} = \sum_{i=1}^{Ng} E(P) + F_{pv} + F_{wind} + F_{battery} \quad (III.11)$$

#### a. Méthode Algorithme de recherche de corbeau (CSA) :

Nous présentons dans le tableau III.26 les résultats de l'algorithme CSA:

H	P1(MW)	P2(MW)	P3(MW)	Pcharge	Emission
1	37,0001	40,0038	60,8961	140	1118,67
2	37,0004	40,0106	64,2139	150	2168,29
3	37,0001	40,188	68,3919	155	2301,53
4	37	40,0017	66,3383	160	3401,46
5	37	44,7126	76,0673	165	2061,34
6	37,00	47,19	80,47	170	1827,64
7	37,0001	43,2498	73,545	175	6638,36
8	37,0002	40,0051	61,1048	180	13612,5
9	37,0001	47,5569	80,8131	210	17346,5
10	37,0001	50,1957	85,5842	230	25384,3
11	37,0002	67,462	114,928	240	7649,67
12	37,0001	70,3185	120,106	250	6573,99
13	37	57,9686	98,5913	240	20998,7
14	37	53,9187	91,9213	220	17448,8
15	37,0001	53,5379	91,122	200	7952,92
16	37	45,739	77,851	180	6030,15

17	37	44,3955	75,3194	170	6747,78
18	37	53,1319	90,5381	185	2725,12
19	37,0001	60,0243	102,226	200	1460,79
20	37	74,9498	127,88	240	1873,12
21	37,0001	69,3657	118,084	225	1713,94
22	37	56,3688	96,0462	190	1322,09
23	37	45,096	76,684	160	1143,91
24	37,0001	40,0354	67,3846	145	942,605
Emission totale=130444.15 Kg					

**Tableau III.26** Les résultats obtenus pour la méthode Algorithme de recherche de corbeau

**b. Algorithme d'optimisation de mouette (SOA):**

Nous présentons dans le tableau III.27 les résultats de l'algorithme SOA:

H	P1(MW)	P2(MW)	P3(MW)	Pcharge	Emission
1	47,89927	40	50	140	1628,514
2	51,23481	40	50	150	2639,348
3	55,49387	40	50,08636	155	2708,147
4	53,34164	40	50	160	3836,419
5	63,46971	44,27211	50,03891	165	2358,777
6	70,67	43,99	50,00	170	2079,60
7	63,79497	40	50	175	6962,265
8	48,11493	40,0019	50	180	14125,23
9	75,3719	40	50	210	17609,34
10	67,47751	43,83237	61,47446	230	25601
11	78,84434	66,79674	73,74139	240	7562,796
12	74,40023	96,67035	56,35725	250	6524,378
13	77,91572	40	75,64572	240	21123,42
14	71,49222	61,34786	50	220	17585,93
15	72,11729	49,80985	59,73567	200	8101,303
16	67,95238	40,30945	52,33708	180	6319,476
17	66,71592	40	50	170	7050,918
18	69,11533	50,40332	61,14967	185	2879,617
19	71,97871	74,4352	52,84548	200	1524,868
20	70,08091	68,2997	101,4503	240	1675,604
21	81,66823	79,41511	63,37023	225	1616,581

22	78,13162	60,58514	50,69093	190	1435,57
23	66,08059	42,68229	50,01905	160	1432,996
24	54,41716	40	50	145	1365,203
Emission totale=135747.29 Ib					

**Tableau III.27** Les résultats obtenus pour la méthode SOA**c. Algorithme d'essaim de tuniciers (TSA):**

Nous présentons dans le tableau III.28 les résultats de l'algorithme TSA:

H	P1(MW)	P2(MW)	P3(MW)	Pcharge	Emission
1	47,9003	40	50	140	453,85
2	45,8698	45,3537	50	150	1484,36
3	55,5803	40	50	155	1583,19
4	53,3393	40	50	160	2699,71
5	67,7798	40	50	165	1250,09
6	51,08	63,58	50,00	170	966,85
7	58,5177	45,2777	50	175	5859,18
8	48,1107	40	50	180	12946,7
9	61,8919	52,9121	50,5661	210	16474,7
10	75,7091	47,0712	50	230	24447,4
11	90,1767	78,5571	50,6504	240	6238,59
12	79,2536	98,1697	50	250	5068,28
13	85,3885	58,171	50	240	19864,7
14	52,3237	80,5139	50	220	16433,7
15	81,0463	50,6138	50	200	6934,87
16	59,9063	50,6823	50	180	5198,83
17	61,5177	43,5034	51,6936	170	5946,37
18	74,6392	40,0697	65,9606	185	1727,26
19	37	96,2193	66,0293	200	307,647
20	96,6105	93,2186	50	240	204,902
21	91,9476	67,3842	65,1186	225	240,785
22	74,0949	57,2535	58,0632	190	235,944
23	58,6103	50,1636	50	160	331,969
24	53,565	40,8553	50	145	232,866
Emission totale=137132.64 Ib					

**Tableau III.28** Les résultats obtenus pour la méthode TSA

**d. Méthode Algorithme de luciole (FFA):**

Nous présentons dans le tableau III.29 les résultats de l'algorithme de la luciole FFA:

H	P1(MW)	P2(MW)	P3(MW)	Pcharge	Emission
1	47,89956	40	50	140	454,0227
2	51,22668	40	50	150	1482,337
3	53,58691	41,99319	50	155	1583,492
4	37,94421	55,39659	50	160	2708,394
5	55,32609	52,45422	50	165	1252,687
6	59,32	54,22	51,11	170	962,96
7	63,79491	40	50	175	5858,101
8	47,37691	40	50,73203	180	12947,8
9	49,33592	66,03397	50	210	16481,26
10	56,95631	62,51384	53,3099	230	24452,9
11	86,34342	83,04598	50	240	6234,164
12	96,52854	67,0218	63,8752	250	5064,378
13	60,6038	82,95604	50	240	19873,08
14	55,82817	77,01151	50	220	16428,51
15	69,90205	59,02183	52,73644	200	6936,705
16	41,38903	63,53798	55,66333	180	5211,235
17	64,57727	42,13885	50	170	5945,885
18	59,40639	71,26506	50	185	1723,099
19	86,35683	62,89282	50	200	267,8368
20	111,076	64,7859	63,96614	240	210,3423
21	70,32795	81,16425	72,95661	225	250,8223
22	80,56308	48,77034	60,08198	190	234,3897
23	53,49036	50,00606	55,28202	160	332,0849
24	54,41981	40	50	145	232,5588
Emission totale=137129.05 Ib					

**Tableau III.29** Les résultats obtenus pour la méthode FFA

Le tableau III.30 regroupe les résultats comparatifs entre les valeurs optimisées.

Les méthode	Emission totale (Ib)
<b>Méthode (CSA)</b>	130444.15
<b>Méthode (SOA)</b>	135747.29
<b>Méthode (TSA)</b>	137132.64

Méthode (FFA)	137129.05
---------------	-----------

Tableau III.30 Résultats comparatifs ED

En analysant les résultats données par le tableau III.45, nous remarquons que les émissions total trouvé par l’algorithme CSA qui est égal à est de 130444.15Kg est plus réduit par rapport à celui trouvé par les algorithmes SOA TSA et FFA qui est estime à 13574.29Kg, 137132.64 Kg et 137129.05 Kg.

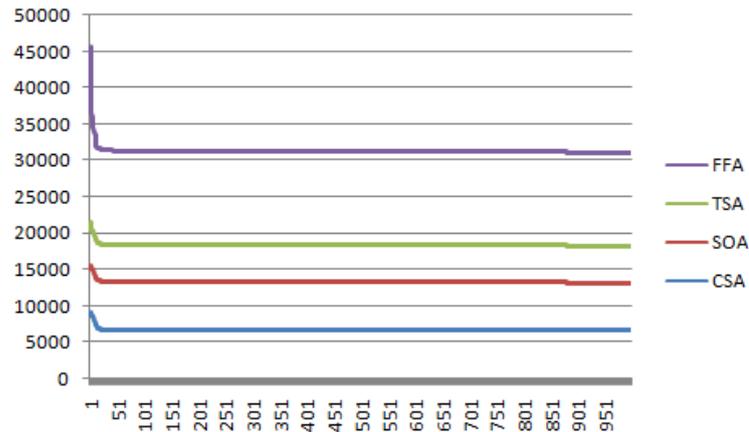


Fig.III.6 : Courbe des émissions à la charge=250 MW

### III.3.3 Dispatching Economique Environnemental combinée dynamique (CEED) avec les ressources des énergies renouvelables (RES)

Dans ce cas, nous traitons le dispatching économique/environnemental combinées dynamique (CEED) avec les ressources des énergies renouvelables (RES) en introduisant le facteur de pénalité de prix.

Les fonctions objectives des coûts et des émissions combinées avec les couts des ressources des énergies renouvelables exprimées en dollars par heures sont données par les expressions suivantes:

$$CEED_{RES} = \sum_{i=1}^{Ng} C(P) + F_{pv} + F_{wind} + F_{battery} \tag{III.12}$$

#### a. Méthode Algorithme de recherche de corbeau (CSA) :

Les valeurs optimales des puissances générées, des pertes actives transmises, du coût total, du cout de production, des émissions sont données par le tableau III.31.

H	P1(MW)	P2(MW)	P3(MW)	Pcharge	Cout+Emission+RES
1	47,8982	40,0012	50,0006	140	7684,74

2	51,2246	40,0002	50,0002	150	8757,2
3	55,5781	40,0014	50,0005	155	8927,4
4	53,3382	40,0007	50,001	160	10007,3
5	66,2099	41,57	50,0001	165	8835,64
6	68,05	46,61	50,00	170	8701,44
7	63,7892	40,0057	50,0001	175	13357,2
8	48,1091	40,0005	50,0004	180	20179,8
9	68,2484	47,1216	50	210	24230,7
10	70,2162	52,5635	50,0003	230	32380
11	77,8003	72,9584	68,6312	240	15386,9
12	78,9232	75,9873	72,5145	250	14440,3
13	74,19	63,2216	56,1484	240	28322,8
14	72,692	59,1789	50,9692	220	24602,1
15	72,5249	58,7357	50,3994	200	15087,4
16	66,8722	43,7178	50,0001	180	12844,7
17	65,8726	40,8423	50,0001	170	13507
18	72,3716	58,2967	50,0017	185	9843,9
19	74,9855	65,3676	58,897	200	8875,66
20	80,6575	80,663	78,5095	240	9939,15
21	78,508	74,8655	71,0764	225	9532,46
22	73,6102	61,6603	54,1445	190	8580,07
23	66,4401	42,3398	50,0001	160	7932,46
24	54,4183	40,0017	50	145	7557,83
(Cout+Emission+RES) totale=329514.28					

**Tableau III.31** Les résultats obtenus pour la méthode Algorithme de recherche de corbeau

**b. Algorithme d'optimisation de mouette (SOA):**

Nous présentons dans le tableau III.32 les résultats de l'algorithme SOA:

H	P1(MW)	P2(MW)	P3(MW)	Pcharge	Emission
1	47,90002	40	50	140	7684,739
2	51,13004	40,02532	50,06933	150	8758,457
3	54,07375	41,50145	50	155	8940,932
4	53,33071	40	50	160	10016,45
5	66,15079	40	51,62635	165	8844,297
6	70,34	44,04	50,00	170	8851,08

7	63,79819	40	50	175	13360,49
8	48,10334	40	50	180	20186,42
9	67,44692	47,93198	50	210	24240,06
10	70,32413	50,9291	51,53159	230	32387,16
11	70,41542	72,31331	76,65675	240	15410,88
12	76,64607	89,42748	61,33713	250	14483,1
13	70,49394	71,80204	51,26082	240	28338,67
14	67,0796	65,45412	50,30299	220	24617,58
15	69,05926	55,70253	56,90336	200	15100,1
16	70,59448	40	50	180	12854,14
17	65,23741	41,47766	50	170	13507,21
18	70,55738	52,13164	57,97717	185	9857,352
19	78,05917	68,78677	52,4051	200	8883,652
20	86,34978	72,4189	81,06108	240	9955,245
21	80,46542	74,31222	69,67139	225	9534,594
22	73,74953	61,33746	54,32143	190	8586,499
23	68,77994	40	50	160	7934,51
24	54,41515	40	50	145	7562,59
(Cout+Emission) totale=329896.20					

**Tableau III.32** Les résultats obtenus pour la méthode SOA

**c. Algorithme d'essaim de tuniciers (TSA):**

Nous présentons dans le tableau III.33 les résultats de l'algorithme TSA:

H	P1(MW)	P2(MW)	P3(MW)	Pcharge	Emission
1	47,9	40	50	140	7684,77
2	51,2253	40	50	150	8757,48
3	55,5798	40	50	155	8927,57
4	53,3393	40	50	160	10008
5	66,9296	40,8504	50	165	8835,91
6	67,69	46,97	50,00	170	8702,09
7	63,7951	40	50	175	13357,3
8	48,1101	40	50	180	20180
9	69,3725	45,1073	50,8899	210	24233,9
10	72,0144	50,7641	50	230	32382,7
11	77,7005	69,5208	72,1678	240	15390

12	77,4785	73,329	76,6198	250	14445,2
13	72,9683	66,554	54,0366	240	28325,7
14	74,7078	58,132	50	220	24603,5
15	72,5254	59,1332	50	200	15088,8
16	68,4114	42,1773	50	180	12846,7
17	65,8671	40,847	50	170	13507,8
18	72,4854	56,7216	51,4579	185	9849,24
19	73,7844	70,4614	55,0038	200	8880,2
20	77,4929	80,1481	82,1884	240	9943,46
21	78,2112	77,785	68,4508	225	9536,73
22	72,535	62,3433	54,5358	190	8581,26
23	65,2872	43,4933	50	160	7933,45
24	54,419	40	50	145	7558,81
(Cout+Emission) totale=329560.64					

**Tableau III.33** Les résultats obtenus pour la méthode TSA

**d. Méthode Algorithme de luciole (FFA):**

Nous présentons dans le tableau III.34 les résultats de l'algorithme de la luciole FFA:

H	P1	P2	P3	Pcharge	Emission
1	47,9	40	50	140	7684,723
2	51,22498	40	50	150	8757,185
3	53,35466	42,21774	50,00763	155	8941,105
4	52,98688	40,34964	50,00352	160	10009,7
5	64,58213	43,19786	50	165	8836,567
6	64,80	48,25	51,61	170	8708,48
7	59,77948	43,81172	50,2038	175	13367,53
8	47,9567	40,15124	50,00204	180	20181,29
9	66,60501	48,765	50	210	24231,69
10	70,20996	51,00159	51,56844	230	32382,25
11	76,91034	77,46447	65,01519	240	15390,17
12	79,19588	68,18866	80,04047	250	14450,74
13	75,0194	61,85302	56,68759	240	28323,19
14	71,34894	60,16996	51,32111	220	24602,66
15	73,3955	58,25965	50,00486	200	15087,68
16	64,98374	44,75664	50,84961	180	12848,34

17	64,88574	41,82926	50	170	13507,35
18	74,89152	54,74251	51,03599	185	9846,974
19	74,59123	62,15515	62,50363	200	8877,728
20	79,08987	84,06012	76,68002	240	9941,211
21	79,50638	73,0309	71,91273	225	9533,121
22	74,5056	63,02102	51,88838	190	8580,859
23	64,30739	44,46534	50,00728	160	7934,147
24	54,21304	40,20692	50	145	7559,105

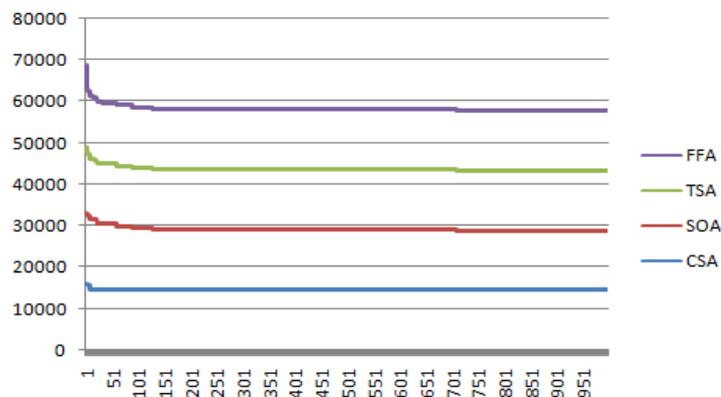
**Tableau III.34** Les résultats obtenus pour la méthode FFA

Les méthodes	(Cout+Emission) totale
Méthode (CSA)	329514.28
Méthode (SOA)	329896.20
Méthode (TSA)	329660.64
Méthode (FFA)	329583,81

**Tableau III.35** Résultats du dispatching économique-environnemental CEED

En analysant les résultats données par le tableau III.50 de cas de dispatching économique/environnemental avec RES, nous pouvons constater que le coût total est réduit lors de l'application de CSA (329514.28\$) par rapport à celui trouvé par les algorithmes SOA, TSA et FFA qui est estimé à 329896.20\$ , 329660.64\$ et 329583.81\$.

D'après les résultats donnés par le tableau III.50, nous déduisons que la minimisation du coût de production et celle des émissions sont des objectifs contradictoires. Les émissions sont maximales lorsque le coût de production est minimal.



**Fig.III.7** :Courbe de cout total à la charge=250 MW

### III.4 Conclusion

Dans ce chapitre, nous avons présenté des tests de l'algorithme de recherche de corbeaux 'Crow Search Algorithm (CSA)' sur différents réseaux à savoir le réseau à 05 générateurs et un micro réseau isolé (islanded microgrid).

Les résultats obtenus en appliquant l'algorithme de la recherche de corbeaux aux problèmes de l'écoulement de puissance dit le dispatching économique, environnemental, dispatching économique-environnemental et en tenant compte des coûts d'investissement des énergies renouvelables sont très probants et montrent qu'ils présentent une forte applicabilité à la résolution de ces problèmes. Ces résultats sont aussi comparables avec d'autres résultats obtenus dans le même domaine et satisfont nos contraintes à savoir la contrainte de type égalité et les contraintes de fonctionnement.

### Conclusion générale

L'algorithme de recherche Crow (CSA) a été récemment développé comme Un algorithme qui simule le comportement des corbeaux en stockage et reprendre de la nourriture. Ce dernier a été d'un grand intérêt pour les chercheurs En raison de ses excellentes caractéristiques, dans notre thèse, nous avons fourni une revue complète de CSA. Et nous l'avons comparé avec d'autres méthodes (Algorithme d'optimisation Seagull (SOA) , Méthode Tunicate Swarm Algorithm (TSA), Méthode Algorithme de luciole (FFA)) qui sont utilisées pour être les meilleures en termes de coût, d'émission et de pertes, où les résultats de la comparaison de CSA avec (Algorithme d'optimisation Seagull (SOA) , Méthode Tunicate Swarm Algorithm (TSA), Méthode Algorithme de luciole (FFA)) ont prouvé que la recherche du corbeau L'algorithme donne les meilleurs résultats en termes de moindre (coût + envoi + pertes). Les limites et les forces de la CSA sont discutées dans des détails. Une gamme complète d'applications a été étudiée. Malgré le succès et la popularité de l'ASC, de nombreux domaines et Les défis futurs doivent être relevés. Plusieurs régions Ce qui peut être traité à l'avenir est la liste ci-dessous

- Aucun travail dans la littérature n'a étudié les paramètres de contrôle ASC.
- Aucun travail n'a été entré dans la littérature pour l'adaptation L'ASC pour travailler dans une dynamique et multi-objectifs Problèmes.
- CSA peut être hybridé avec de nombreux autres algorithmes.
- Aucun travail n'a été entré dans la littérature pour l'adaptation CSA pour travailler sur un problème d'optimisation bruité.
- L'ASC a besoin de plus d'attention aux études théoriques.

## Références bibliographiques

---

### Références bibliographiques

- [1] UNIVERSITE A/MIRA-BEJAIA- Modélisation et simulation des réseaux électriques
- [2] M. Rahli « Contribution à l'Etude de la Répartition Optimale des puissance actives dans unRéseau d'Energie Electrique », Thèse d'Etat, Soutenu à L'USTO, 1998.
- [3] N.V.Aravantidis ,J .Rosing, the use of objective function in real power dispatching ,I.E.E.E Trans on PAS, vol pas 90, July-Auguest1975
- [4] J.F.Dopazo, Member IEEE, J.D .Schaffer."Real-Coded Genetic Algorithms and IntervalSchemata ‘’, Foundation of Genetic Algorithms 2. San Mateo: L Darrel whitley(Morgan Kaufmann Publishers), 187-202-1993
- [5] R. Gnandass , P. Venkatesh ,T. G. Palanivelu , K Manivannan. « EvolutionaryProgramming Solution of Economic Load Dispatch with Combined Cycle Co-generation Effect »IE (I) Journal-EL, Vol 85,pp124-128,September 2004.
- [6] K. S.Swarup. «Economic Dispatch Solution using Hopfied Neural Network » IE(I) Journal-EL, Vol 84,pp.77-82,September 2004
- [7] G.W.Stagg ,A.H. Elabiadh. « Computer Methods in Power System »,Mc Graw-HillInternational Book Company
- [8] L.P. Singh. «Advanced power system analysis and dynamic », Wiley Eastern limited , 1986
- [9] Y. Wallach .«Calculation and program for power system network », Prentice- Hall, Inc, Englewood cliffs, 1986.
- [10] T.Gonen «Modern power system analysis », John Wiley Sons, 1988.
- [11] A. Chaker &M. Laouer & H. Bouzeboudja. «Analyse comparative de la répartition optimale des Puissances par les méthodes des fonctions implicites et les équations de coordination », Bulletin scientifique de L'ENSET Oran, N°4 –juillet 1997.
- [12] Solving multi-objective economic emission dispatch of a renewable integrated microgrid using latest bio-inspired algorithms - Bishwajit Dey, Shyamal Krishna Roy, Biplab Bhattacharyya - Department of Electrical Engineering, IIT(ISM), Dhanbad, IndiaDepartment of Electrical Engineering, IIT(ISM), Dhanbad, India
- [13] Economic Dispatch of Renewable Energy Sources in a Microgrid using Ant Lion Optimizer-Conference Paper. March 2016
- [14] Meziane. R, « Optimisation de la structure d'un réseau de production d'énergieélectrique et amélioration de sa performance », thèse de doctorat, USTO 2007.
- [15] Méta heuristique, Wikipédia encyclopédie, 20 décembre 2005

## Références bibliographiques

---

- [16] B. Bullnheimer, R.F. Hartl, and C. Strauss, A new rank-based version of the ant system: a computational study, *Central European Journal of Operations Research* 7 (1) (1999), 25–38.
- [17] Jin-Kao Hao, Philippe Galinier, Michel Habib, Méta heuristiques pour l'optimisation combinatoire et l'affectation sous contraintes, *Revue d'Intelligence Artificielle*, Vol : No. 1999.
- [18] J. Dreco, A. Petrowski, P. Siarry et E. Taillard, « Métaheuristiques pour L'optimisation difficile », Eyrolles, 2003.
- [19] S., Martello S., Osman I.H., Roucairol C. (eds.) *Meta- Heuristics: Advances and Trends in Local Search Paradigms for Optimization*, Kluwer, Boston, 1999.
- [20] Algorithmes génétiques, Wikipedia, encyclopédie, 20 décembre 2005.
- [21] M. Dorigo, *Optimization, learning and natural algorithms*, Ph.D. Thesis, Politecnico di Milano, Milano, 1992
- [22] E. Bonabeau, M. Dorigo, G. Theraulaz, *Nature*, Volume 406, Number 6791, Pag. 39 - 42(2000)
- [23] J.H. HOLLAND, *Adaptation in natural and artificial systems*. The University of Michigan Press, Ann Arbor, 1975.
- [24] S. Chen, S. Smith., *Commonality and genetic algorithms*. Technical Report CMU-RITR- 96-27, The Robotic Institute, Carnegie Mellon University, Pittsburgh, PA, USA, 1996.
- [25] Glover, F., Kochenberger, G.A.: *Handbook of Metaheuristics*. Kluwer (2003).
- [26] J. C. Dodu , P. Huard. « La méthode de Quasi-Newton sous contraintes non linéaires : Algorithmes à convergence globale super linéaire », *Bulletin de la Direction des Etudes et Recherches*, Electricité de France, Série C N°2, 1988
- [27] M. Minoux. « *Programmation Mathématique Théorie et Algorithmes* », Tome 1, Dunod, 1983
- [28] novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm  
Alireza Askarzadeh  
Department of Energy Management and Optimization, Institute of Science and High Technology and Environmental Sciences, Graduate University of Advanced Technology Kerman, Iran
- [29] Seagull optimization algorithm: Theory and its applications for large-scale industrial engineering problems  
Gaurav Dhiman \*, Vijay Kumar  
Computer Science and Engineering Department, Thapar Institute of Engineering & Technology, Patiala 147004, Punjab, India

## Références bibliographiques

---

- [30] Tunicate Swarm Algorithm: A new bio-inspired based metaheuristic paradigm for global optimization ☆ Satnam Kaur a, Lalit K. Awasthi, A.L. Sangal, Gaurav Dhiman,  
a Department of Computer Science and Engineering, Dr. B. R. Ambedkar National Institute of Technology, Jalandhar 144011, Punjab, India  
b Department of Computer Science, Government Bikram College of Commerce, Patiala 147001, Punjab, India
- [31] Firefly Algorithm, Levy Flights and Global Optimization-Article· March 2010
- [32] M. Yehia, R. Ramadan, Z. El-Tawail, K. Tarhini, «An Integrated Technico- Economical Methodology for Solving Reactive Power Compensation Problem», IEEE Trans. On PAS, vol. 13, No. 1, pp. 54–59, Feb. 1998.
- [33] O. Roux, La mémoire dans les algorithmes à colonie de fourmis, applications à optimisation et à la programmation automatique, thèse de doctorat, Université du Littoral Côte, Opale, 2001
- [34] E. Housos & G .D .Irissarri . «Real and Reactive Power System Security Dispatch Using a Variable Weights Optimization Method », IEEE Transactions on Power Apparatus and System. Vol PAS-102, N°.5, pp.1260.1983.