



République Algérienne Démocratique et Populaire
Ministère de l'enseignement supérieur et de recherche scientifique
Université Mohamed El Bachir El Ibrahimi
Faculté des Mathématiques Et D'informatique
Département des Mathématiques

Mémoire de master

Le problème de la tournée de véhicule avec contrainte de capacité par l'algorithme génétique

Spécialité:
Recherche Opérationnelle

Présenté par :

- MEKHOUKH Lilia.
- Ghezali Yamina

Proposé et dirigé par :

- Dr. Maza Sofiane.

Devant le jury composé de :

- Dr. Moussawi Ali
- Dr. Fillali Ferhet
- Dr. Maach Salah

Année universitaire :2020-2021

Remerciements

Tous d'abord, nous tenons en premier lieu à remercier Dieu tout puissant de nous avoir donné la force, la santé et la volonté pour réaliser ce travail.

Nous tenons à remercier également notre encadreur **M^r** Maza Sofiane, Maître à l'Université El Bachir El Ibrahim de " Bordj Bou Arreridj " pour avoir bien voulu encadrer et de nous aider tout au long de la réalisation de ce mémoire, aussi ses conseils, ses commentaires et sa bienveillance.

Nous remercions aussi **M^r** **Ghezali Fatah** pour son aide et ses précieux conseils, et qui nous a prêté main forte pendant notre période de travail.

Nous exprimons notre grand respect également aux membres du jury d'avoir accepté d'examiner et d'évaluer ce travail.

Nous tenons de remercier tous les enseignants de la faculté des mathématiques et informatique pour tout le savoir que nous avons acquis grâce à eux durant notre cursus universitaire.

Enfin, nous tenons de remercier tous ceux qui nous ont soutenu et encouragé pendant la préparation de ce modeste travail.

Dédicace

Louange à dieu et que la prière et la paix soient sur le prophète.

Je dédie ce modeste travail à toute personne en quête du savoir, de science et de culture.

A celle qui ma soutenue avec ses prières ...celle qui a veillé pour éclair mon chemin

À celle qui a partagé mes joies À celle qui m'a donnée l'amour et la tendresse

Au plus beau sourire de ma vie à celle qui nous a quittés récemment

Et nous à laisser un grand vide et une douleur atroce

Celle qui nous manque et terriblement, ma "très chère mère" que

Dieu la bénisse et lui offre le paradis éternel.

A celui qui ma appris que la vie est une lutte permanente, le savoir et la science est son

arme à celui qui n'avait jamais épargne ses efforts pour me rendre heureuse,

à mon "très cher père" que Dieu le garde pour nous.

A mes frères le cadeaux du destin et qui connaissent la valeur de la fraternité .

Mes frères : Adel, Walid, Djalil, Rezki, Wafa, Sihem, Yasmina

Une dédicace spéciale aux poussins de la famille : Karim, Loudjain, Ayoub, Wassim.

A toute la famille qui ma toujours soutenue mes cousins , cousines toutes et oncles.

A mes collègues avec qui nous avons partagé le parcours du miel et de fiel que Dieu les
protèges.

Mes chères : Amina , Nadjewa ,Rahma ,Houda , Rania ,Safa ,Insaf , Echerak , Chaima

A mes professeurs et maitres qui nous ont comblés d'amour , de sacrifice de guidance

à qui je dois ma réussite et qui éclairent mon chemin.

Au département mathématique et informatique promotion 2021 de l'université El Bachir El
ibrahimi BBA.

A toute personne que j'estime et je respecte

Mekhoukh Lilia

Dédicace

Je dédie ce modeste travail a :

- ✚ Mes chers parents qui non pas cesse de prier pour moi ; pour leurs affectations et leur amour, et qui m'ont aidé durant toute la durée de mes études, que dieu les protège et les grande pour nous.
- ✚ Mes très chères frères et sœurs qui m'ont fourni du courage, du soutien, et tous leurs efforts et moyens pour que je termine mes études.
- ✚ Mes oncles, mes tantes.
- ✚ Mes cousins et mes cousines.
- ✚ Tous mes amis et camarades.

A tous ceux qui ont participé de près ou de loin dans la réalisation de ce travail.

Ghezali Yamina

L'optimisation combinatoire occupe une place très importante dans la recherche opérationnelle, en mathématique discrètes et en informatique. Le problème de tournée de véhicule avec contrainte de capacité (CVRP) est un problème avec la problématique : Quel est l'ensemble optimal de routes traverser afin de livrer a un ensemble donné de clients en respectant les capacités des véhicules et en retournant ou dépôt une fois termine ? L'objectif de ce travail est de minimiser le cout de distance, en appliquant les algorithmes génétiques comme heuristique et en utilisant le langage de programmation ' MATLAB ' pour le codage.

Mots-clés : Problème de tournée de véhicule avec contrainte de capacité CVRP, optimisation, algorithmes génétiques, MATLAB.

Liste des tableaux

Tableau 3.1 : Caractéristique de l’outil utiliser ; Matlab

Tableau 3.2 : Expérimentation pour mesurer l’influence de nombre des itérations

Tableau 3.3 : Expérimentation pour mesurer l’influence de nombre des itérations

Tableau 3.4 : Expérimentation pour mesurer l’influence de nombre des itérations

Liste des figures

Figure 1.1 : Représentation du problème de transports	15
Figure 1.2 : Représentation du Problème de voyageur de commerce	16
Figure 1.3 : Représentation de problème des postiers chinois	17
Figure 1.4 : Représentation graphique de problème de tournée des véhicules	18
Figure 1.5 : Variante la base de VRP avec contrainte de capacité	25
Figure 2.1 : La fonctionnement d’algorithme fireflies	31
Figure 2.2 : La fonctionnement d’algorithme de colonie de fourmis (ACO)	33
Figure 2.3 : La fonctionnement d’algorithme d’optimisation par essais particulières	36
Figure 2.4 : Schéma récapitulatif d’un algorithme génétique	42
Figure 3.1 : l’organigramme d’algorithme génétique.....	50
Figure 3.2 : langage de programmation Matlab.....	53
Figure 3.3 : l’interface de l’application Matlab.....	54
Figure 3.4 : Fenêtre graphique de l’exécution	55
Figure 3.5 : Fenêtre graphique de l’exécution	59

Figure 3.6 : Fenêtre graphique de l'exécution

Liste des abréviations

PVC	Le problème de voyageur de commerce
CPP	le problème de postier chinois.
VRP	le problème de tournée des véhicules.
CVRP	le problème de tournée des véhicules avec contrainte de capacité.
AF	algorithme firefly
ACF	algorithme de colonie de fourmis.
PSO	optimisation par essaims particulaires.
GA	génétique algorithme.

Table de matière

Liste des tableaux

Liste des figures

Liste des abréviations

1. Introduction générale	11
CHAPTRE I	13
2. Introduction.....	14
3. Problème des transports	15
4. Problème du voyageur de commerce.....	15
5. Problème du postier chinois	16
6. Problème de tournée des véhicules	17
6.1 Etude de présentation des problèmes	18
6.2 La formulation mathématiques VRP.....	19
7. Types de problèmes de la tournée de Véhicule (VRP).....	22
7.1 PTVC (Problème de tournées de véhicules avec capacité).....	22
7.2 PTVD (Problème de tournées de véhicule dynamique).....	22
7.3 VRPB (Prolongation du VRP classique).....	23
7.4 PTVMP (Problème de tournées de véhicules multi périodique).....	23
7.5 PTVS (Problème de tournée des véhicule stochastique)	24
8. Problème de la tournée de véhicule avec contrainte de capacité	24
9. Objectifs.....	25
10. Conclusion	26
Chapitre II	27
1. Introduction.....	28
2. L'algorithme fireflies	29
2.1 Algorithme fireflies	32
3. Algorithme de Colonie des fourmis	32
3.1 Variantes Principales	34
3.2 Algorithme de colonie de fourmis	34

4. Optimisation par essais particulières (PSO).....	35
4.1 Algorithme d’optimisation par essais particulières: [30]......	37
5. Algorithme génétique.....	37
5.1 Les éléments d’Un algorithme génétique.....	38
5.1.1 Génotype ou chromosome.....	38
5.1.2 Gène.....	38
5.1.3 Phénotype.....	38
5.2 Population initiale.....	39
5.3 Codage d’individus.....	39
5.3.1 Codage binaire.....	40
5.3.2 Codage réel.....	41
5.4 Algorithme génétique.....	42
5.5 Schéma récapitulatif d’Algorithme Génétique.....	42
5.6 Les opérateurs génétiques.....	43
5.6.1 L’évaluation.....	43
5.6.2 Operateur de sélection.....	43
5.6.2.1 La sélection par roulette.....	44
5.6.2.2 La sélection par tournoi.....	44
5.6.2.3 La selection Elitiste.....	45
5.7 L’opérateur croisement.....	45
5.7.1 Le croisement en un point.....	45
5.7.2 Le croisement en deux point.....	46
5.8 Operateur de mutation.....	46
5.9 Remplacement.....	46
5.9.1 Remplacement stationnaire.....	46
5.9.2 Remplacement élitiste.....	47
6. Conclusion.....	47
Chapitre III	48
1. Introduction.....	49
2. L’objectif du travail.....	49
3. L’organigramme de l’algorithme génétique.....	50

4. Description de l'approche proposée.....	51
4.1 Contrainte de tournée de véhicule avec capacité de véhicule	51
4.2 Paramètre des problèmes	51
4.2.1 La partie qui concerne les attributs des villes	51
4.2.2 la partie qui concerne les attributs des dépôts	51
4.3 Les paramètres d'algorithme génétique	52
5. Environnement matériel	52
6. Environnement de développement.....	52
7. Expérimentation	54
8. Quelques test sur l'implémentation	56
9. Discussion et résultat.....	63
10. Conclusion	64
Conclusion Générale	65
Bibliographie	66
ملخص	70
Résumé	70

1. Introduction générale

La recherche opérationnelle consiste à utiliser un ensemble des méthodes et des techniques rationnelles d'analyse et de synthèse des phénomènes pour résoudre des problèmes complexes d'organisation, de planification, de gestion de ressources, ou d'optimisations issus de nombreux secteurs d'activité comme le transport, l'industrie, les télécommunications, la finance ...etc.

L'optimisation joue un rôle important dans la recherche opérationnelle, est au cœur de tous problèmes liés à la prise de décisions.

L'optimisation cherchant à modéliser et à résoudre analytiquement ou numériquement les problèmes qui consistent à minimiser ou maximiser une fonction sur un ensemble.

Il existe deux types de méthodes d'optimisation, la première est l'optimisation mono-objectif qui se base sur la minimisation (ou la maximisation) d'une seule fonction objective ou le but est de trouver la meilleure solution appelé solution optimale qui est facilement définie suivant une seule performance du problème étudié. D'autre part, l'optimisation multi-objectif optimise simultanément plusieurs fonctions objectives qui souvent contradictoires, on cherche à trouver la meilleure solution suivant un ensemble de performances du problème.

Il y a plusieurs méthodes de résolution. Les méthodes exactes donnent des solutions optimales pour des problèmes de petite taille et les méthodes méta-heuristiques donnent des solutions optimales pour des problèmes de grande taille.

Si les méthodes de résolution exacte permettent d'obtenir une solution dont l'optimalité est garantie dans certaines situations, on peut cependant chercher des solutions de bonne qualité, sans garantie d'optimalité, mais au profit d'un temps de calcul plus réduit. Pour cela, on applique des méthodes appelées méta-heuristique, adaptées à chaque problème traité, avec cependant l'inconvénient de ne disposer en retour d'aucune information sur la qualité des solutions obtenues.

Les méta-heuristiques peuvent être classées de nombreuses façons. On peut distinguer celles qui travaillent avec une population de solutions de celles qui ne

manipulent qu'une seule solution à la fois. Les méthodes qui tentent itérativement d'améliorer une solution sont appelés méthodes de recherche locale ou méthode de trajectoire. Ces méthodes construisent une trajectoire dans l'espace des solutions en tentent de se diriger vers des solutions optimales.

Les exemples les plus connus de ces méthodes sont : Les algorithmes génétiques ; l'optimisation par essaim de particules ; les algorithmes fierfly est les algorithmes de colonies de fourmis présentent les exemples les plus connus des méthodes qui travaillent avec une population.

Le domaine dans lequel s'inscrit notre travail est le problème de la tournée de véhicules, il concerne l'élaboration des tournées de véhicules avec contrainte de capacité. C'est un problème très important qui consiste à affecter chaque client a une tournée effectuée par un seul véhicule de capacité finis.Ce véhicule commence et termine sa tournée au dépôt .

Le problème de la tournée de véhicules est un sous problème important dans le domaine des systèmes de distribution. Pour la résolution de ce problème on a choisi les algorithmes génétiques comme une solution.

On reparti ce mémoire en trois chapitres :

Le premier chapitre : nous présentons le problème de la tournée de véhicule : qui comporte un ensemble de vocabulaire et définition de problème vrp, les types de ce problème et enfin, nous présentons les objectifs de ce problème.

Le deuxième chapitre : les Méta heuristique : nous présentons les méthodes de résolution pour résoudre ce type de problèmes

Et puis, dans le troisième chapitre : on donne une solution de ce problème et en utilisant pour cela l'algorithme génétique ainsi des résultats expérimentaux.

Enfinement, on clôture ce mémoire par une conclusion générale.

CHAPTRE I

Le problème de la tournée de véhicule

2. Introduction

Ce chapitre consacré à la présentation des problèmes de bases de la tournée de véhicules (VRP). Le problème de tournée de véhicules (VRP) est un problème d'optimisation combinatoire et de recherche opérationnelle ; il fait partie de la catégorie des problèmes de transport tout comme le problème du voyageur de commerce (traveling salesman TSP) et le problème du postier chinois (chinese postman problème CPP).

Dans ces problèmes relevant du domaine de la logistique ; un ou plusieurs véhicules doivent couvrir un réseau de transport pour livrer des marchandises à des clients ou cour les route de ce réseau le (VRP) a été l'argenté étudié durant la 2^{émé} moitié du siècle dernier [1]. Considèrent que les années 1980 constituent l'apogée de ces études et (l'un des plus beaux succès de la recherche opérationnelle).

Les principales motivations de l'études de VRP sont d'une part la difficulté de sa réalisation et d'autre part ses nombreuses application pratiques en logistique ; ce 2^{émé} point concerne les votantes économique et environnementales liées à la minimisation des coute du système de transport.

Nous commencerons tout d'abord par présenter les problèmes de TSP ; CPP ; VRP en mettant l'accent sur la modélisation mathématiques du problème de routage de véhicules ainsi que sur ses variantes principales.

3. Problème des transports

Les problèmes de transport appelés aussi problèmes de routage ; modélisent des problèmes réels liés au transport de marchandises ou de personnes. Afin d'introduire le problème de routage de véhicules nous parlerons de deux autres problèmes de transport : le problème de voyageur du commerce et le problème du postier chinois [2].

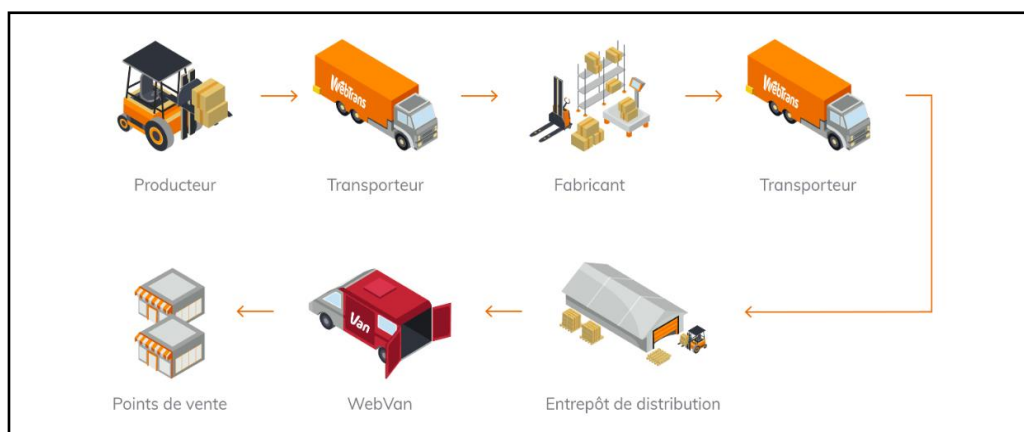


Figure 1.1 : représentation de problème des transports.

4. Problème du voyageur de commerce

On appelle problème du voyageur de commerce le problème informel suivant : étant donné une carte avec des villes, des chemins et des distances (ou temps de trajet) entre ces villes fournir un itinéraire qui passe par toutes les villes, revient à la ville de départ ; ne visite chaque ville qu'une fois et qui réalise la distance (ou le temps minimal) [3].

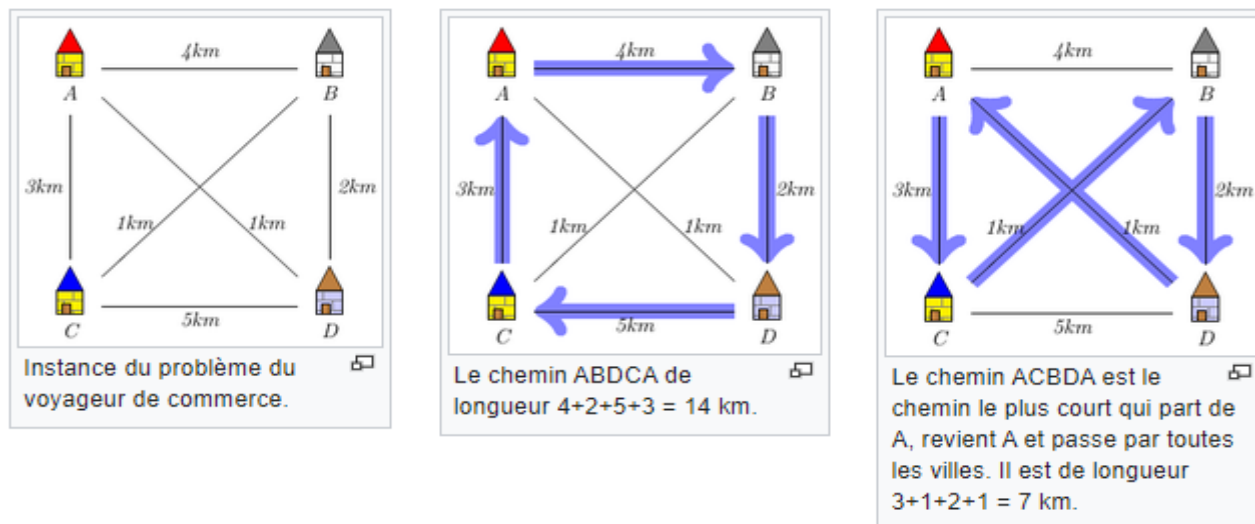


Figure 1.2 : représentation du problème de voyageur de commerce.

5. Problème du postier chinois

Le problème de postier chinois (en anglais : chinese postman problème) a été étudié et modélisé par le mathématicien chinois Ming-Kao Wang en 1962 .Ce problème qui a beaucoup d'applications pratiques (ramassage des ordures ménagères, ramassage solaire ,....etc.) a connu plusieurs évolutions au fil du temps par l'adjonction de nouvelles contraintes ce qui le rend de plus en plus réaliste et complexe en même temps .pour simplifier la définition de ce problème, supposons que dans une ville constitué d'un ensemble de quartiers on veuille organiser la collecte des ordures ménagères ; le problème qui se pose est déterminer un itinéraire qui passe par tous les quartiers une seule fois [4].

Le problème de postier chinois consiste à :

Etant donné un graphe, il s'agit de trouver une chaîne (cycle) Eulérienne dans G si une telle chaîne existe. Dans le cas où la chaîne n'existe pas et si les arêtes sont pondérées le problème

revient à trouver un ensemble d'arêtes du graphe qu'il faut parcourir plus d'une fois de manière à rendre notre graphe eulérien et rendre le coût total de la tournée minimum.

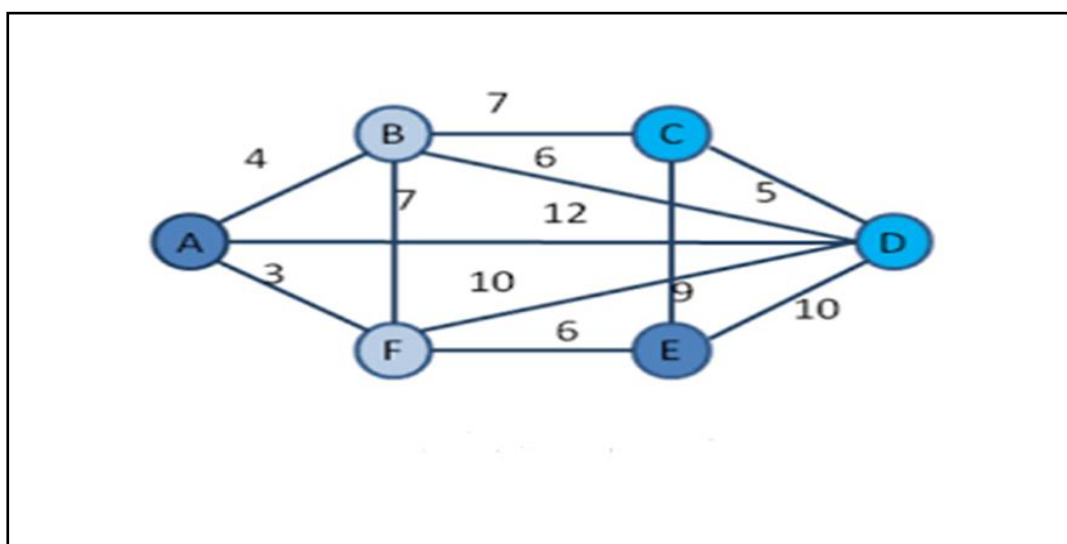


Figure 1.3 : représentation de problème postier chinois.

6. Problème de tournée des véhicules

Le problème de tournées de véhicules (véhicule routing problème VRP) est une extension du problème du voyageur du commerce [5]. Il a été introduit pour la première fois par Dantzig en 1954 sous le nom de (Truck Dispatching Problème) [6].

Autrement-dit dans la version basique dit (CVRP) ou (VRP) avec contrainte de capacité, une flotte des véhicules de capacité fini basée dans un dépôt doit assurer des tournées entre plusieurs clients (villes) ayant demandé chacun une

certaine quantité de marchandises. L'ensemble des clients visités par un véhicule désigne la tournée de celui-ci chaque client doit être desservi une seule fois et chaque tournée commence et se termine au dépôt [7].

6.1 Etude de présentation des problèmes

La figure représente un exemple de problème de VRP avec 25 clients, résolu avec 5 véhicules.

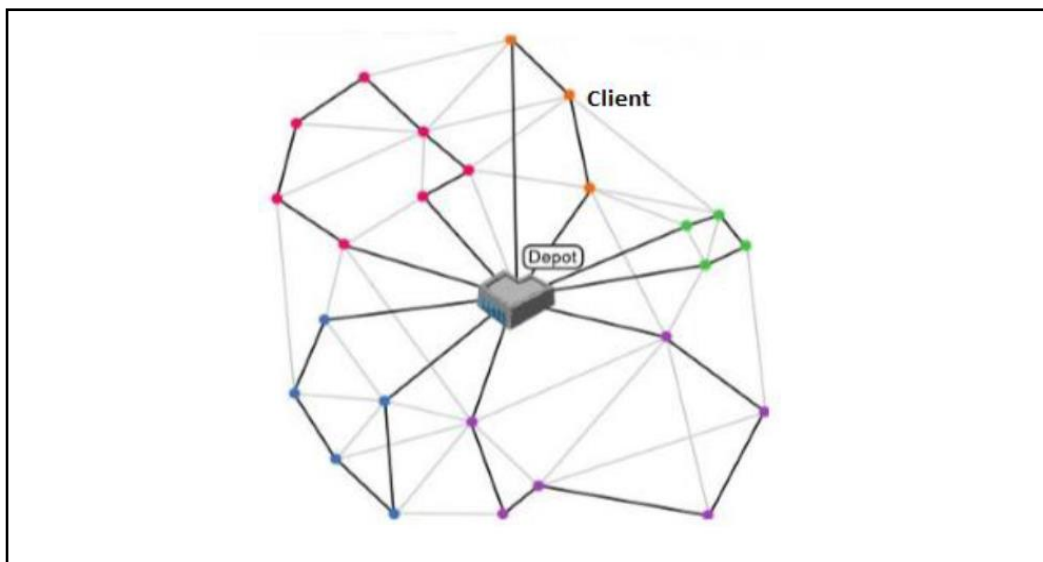


Figure 1.4 : Présentation graphique du problème de tournées de véhicules VRP [7].

6.2 La formulation mathématiques VRP

Formulation La version standard du VRP considère une contrainte de capacité (**PTVC** , Problème De Tournée Des Véhicules Avec Capacité) Il peut être représenté sous la forme d'un graphe orienté et value $G=(N ,A)$

Ou :

- N représente les positions des clients et du depot,
- A représente les arcs entre deux clients $i, j \in N$.

Plus spécifiquement, nous avons un ensemble $C = \{1, \dots, n_c\}$ de clients qui doivent obtenir une livraison de merchandise provenant du dépôt. L'ensemble des positions de ces clients ou nœuds est défini par l'ensemble $N = C \cup \{0, n_{c+1}\}$ ou 0 et n_{c+1} représente le depot (aller et retour).

Une demande positive de produit d_i est associée à chaque client i appartenant à C .

Une flotte de véhicules $V = \{1, \dots, n_v\}$ est disponible au dépôt et chaque véhicule possède la même capacité (flotte homogène) Q telle que $Q \geq \max d_i, \forall i \in N$. Pour tous les clients i et $j, \forall i, j \in N$, nous connaissons le cout c_{ij} de transport direct entre i et j (proportionnel à la distance à parcourir). Pour trouver l'ordre de visite des clients, nous définissons les variables de décisions comme suit :

$$x_{ij}^v = \begin{cases} 1 & \text{Si le Véhicule } v \in V \text{ visite le client } j \text{ après le client } i, \\ 0 & \text{Sinon} \end{cases}$$

En définissant y_i comme étant la charge résiduelle du véhicule après avoir desservi le client $i \in C$. Il nous est possible d'écrire formellement le modèle de VRP. Il s'agit d'optimiser la fonction suivante :

$$\text{Min} \sum_{v \in V} \sum_{i \in N} \sum_{j \in N} C_{i,j} x_{i,j}^v \quad (1)$$

Avec les constraints :

$$\sum_{v \in V} \sum_{j \in N} x_{i,j}^v = 1, \forall i \in C \quad (2)$$

$$\sum_{i \in N} x_{i,h}^v - \sum_{j \in N} x_{h,j}^v = 0, \forall h \in C, v \in V \quad (3)$$

$$\sum_{j \in N} x_{0,j}^v = 1, \forall v \in V \quad (4)$$

$$\sum_{j \in N} x_{j,n+1}^v = 1, \forall v \in V \quad (5)$$

$$x_{ij}^v = 1 \rightarrow y_i - d_j = y_j, \forall i, j \in N, v \in V \quad (6)$$

$$y_0 = Q, 0 \leq y_i, \forall i \in C \quad (7)$$

$$x_{ij}^v \in \{0,1\}, \forall i, j \in N, v \in V \quad (8)$$

La fonction de cout de la solution $X = (x_{i,j}^v), \forall i, j \in N, \forall v \in V$ est définie par :

$$\text{cout}(X) = \sum_{v \in V} \sum_{i \in N} \sum_{j \in N} c_{i,j} x_{i,j}^v \quad (9)$$

Le nombre de véhicule utilisés par la solution X , est défini par :

$$\text{Nb véhicule}(X) = \sum_{v \in V} \sum_{j \in C} x_{0,j}^v \quad (10)$$

La fonction objective (équation 1) représente le nombre de véhicules utilisés pour les trajets effectués et la somme des couts s'y rapportant.

La formulation du problème nécessite de satisfaire certaines contraintes :

- ✓ L'équation 2 assure qu'on part une et une seule fois de chaque client, avec un seul véhicule.
- ✓ L'équation 3 assure que le véhicule qui arrive chez un client est le même que celui qui part de ce client.
- ✓ L'équation 4 assure que chaque véhicule ne sort qu'une seule fois du dépôt.
- ✓ L'équation 5 assure le retour unique au dépôt pour chaque véhicule (ou tournée). Il n'y a pas de restriction sur le nombre de véhicule, mais un cout c_v est affecté à chaque véhicule utilisé. On impose une valeur de c_v suffisamment grande pour réduire principalement le nombre de véhicule et pour réduire au minimum dans un deuxième temps les couts de transport.
- ✓ Les équations 6 et 8 définissent les contraintes de capacité et d'intégrité.

- ✓ Les équations 9 et 10 sont des fonctions de mesure qui permettent respectivement de quantifier la solution selon la distance totale parcourue. Ainsi que le nombre de véhicules utilisés [7].

Le problème d'élaboration de tournées de véhicule est un problème NP-difficile, c'est-à-dire qu'il n'existe pas à ce jour un algorithme déterministe pouvant résoudre ce problème en temps polynomial.

Pour des problèmes comportant un grand nombre de clients (<100 clients), des méthodes approchées sont nécessaires pour les résoudre.

7. Types de problèmes de la tournée de Véhicule (VRP)

Durant des années de recherche sur le VRP d'autres dérivées de ce problème sont apparues. Ces apparitions sont dues essentiellement aux activités des chercheurs qui travaillent de plus sur les problèmes de transport et de distribution que rencontrent les sociétés. Dans ce qui suit nous allons présenter les principaux problèmes dérivés du PTVFT :

7.1 PTVC (Problème de tournées de véhicules avec capacité)

Comme défini précédemment, un problème PTVC consiste à affecter chaque client à une tournée effectuée par un seul véhicule de capacité finie. Ce véhicule commence et termine sa tournée au dépôt [8].

7.2 PTVD (Problème de tournées de véhicule dynamique)

Contrairement au VRP classique où toutes les demandes des clients sont connues et planifiées à l'avance, le PTVD illustre mieux la réalité. Dans les problèmes de transports réels dans la société d'aujourd'hui la composante dynamique est très importante, ou de pannes des véhicules de transport. Dans ce

genre de situation, le décideur doit réorganiser, en un temps acceptable, la planification des itinéraires de sa flotte de véhicules pour répondre aux nouvelles demandes urgentes [9].

7.3 VRPB (Prolongation du VRP classique)

Ce problème est une prolongation du VRP classique. Il comporte des clients auxquels des produits doivent être livrés et des fournisseurs dont les marchandises doivent être amenées au dépôt. Donc, pour éviter de réarranger les produits transportés dans le véhicule sur une tournée, les clients doivent être visités avant les fournisseurs [10].

7.4 PTVMP (Problème de tournées de véhicules multi périodique)

Le problème de tournées de véhicules multi périodique (PTVMP) consiste à livrer pour un ensemble de clients, la quantité demandée d'un ou de plusieurs produits sur un horizon de temps donné. Dans ce problème, la quantité de produits livrée à un client, permet à ce dernier de subvenir à ses besoins en attendant la prochaine visite du véhicule. Le but principal de ce problème se divise en deux parties : la première consiste à planifier les horaires de livraison de chaque client sur un horizon de temps prédéterminé ; la deuxième consiste à organiser les tournées des véhicules afin d'effectuer les livraisons nécessaires tout en optimisant le coût total de transport. Outre la connaissance et l'utilisation des contraintes connues d'un problème de transport, le PTVMP nécessite l'intégration des éléments de gestion de stock afin de respecter les besoins en quantité et en temps des clients [11].

7.5 PTVS (Problème de tournée des véhicule stochastique)

Un problème VRP est dit stochastique si au moins un de ses éléments est aléatoire, c'est à dire avec une certaine incertitude. Ces éléments peuvent être une ou plusieurs demandes des clients, les temps ou les couts de transport. Le problème avec les demandes stochastique est celui le plus étudié dans la littérature. La majorité des recherches suppose que les demandes aléatoires suivent une loi de distribution << normale >> [12].

8. Problème de la tournée de véhicule avec contrainte de capacité

Le problème de routage du véhicule capacité (CVRP) est un problème de routage du véhicule avec contraintes sur les capacités des véhicules. Dans un programme CVRP, chaque emplacement a une demande d'articles ramassé ou livré là bas.

Chaque fois qu'un véhicule visite un lieu, le montant total des articles véhicule est porteur d'augmentations (pour une prise en charge) ou de diminutions (pour une livraison) par la demande à ce emplacement. Un graphique peut être utilisé pour représenter un CVRP dans lequel l'arête correspond aux distances et aux nœuds correspondent aux demandes.

Si les clients ont chacun une demande fixe et les véhicules ont une capacité de stockage, puis nous parlons du CVRP (Capacité VRP).

Le problème va donc s'aggraver contrainte de limitation de la demande Cumulée du circuit à la capacité du véhicule à travers elle [13].

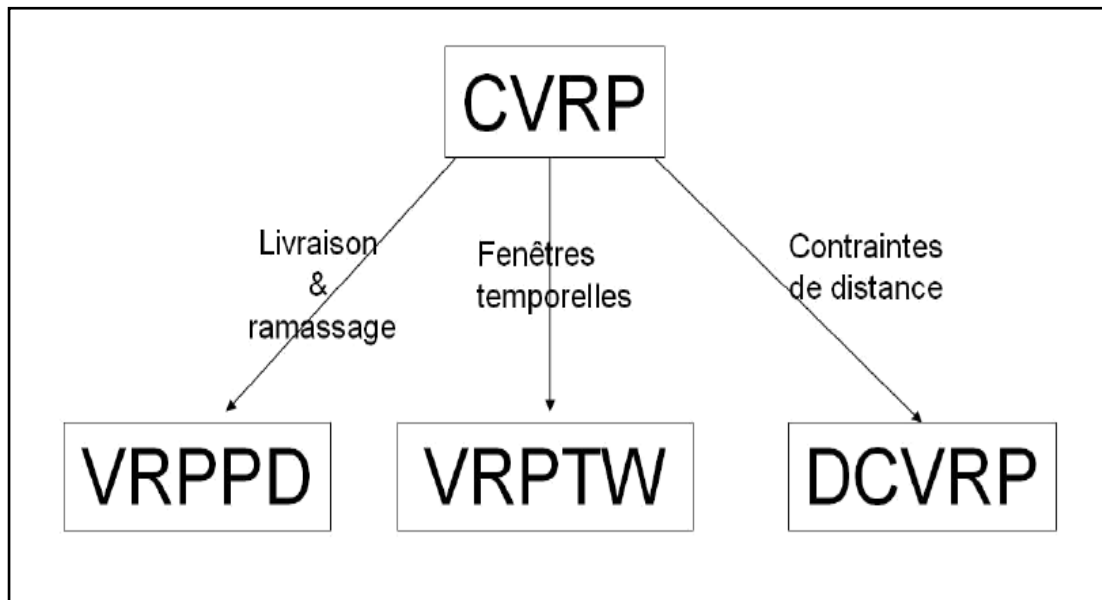


Figure 1.5 : variante de base VRP avec contrainte de capacité [13].

9. Objectifs

L'objectif du PRV et du PRVC est de réduire au minimum le coût total du voyage.

Le PRVC n'est équivalent au PRV que si le total de toutes les demandes d^2 est dynamique si la capacité de chaque véhicule n'est pas limitée à de nouvelles exigences [14].

Le **VRP** en général est l'un des problèmes les plus difficiles qui a attiré beaucoup d'attention et qui a nécessité de nombreuses études. Il a de nombreuses applications dans la vie réelle et trouver une solution optimale est un effort continu.

Autrement-dit : est de minimiser le coût total, la somme des distances ou des temps de parcours des tournées, tout en respectant la contrainte de capacité des véhicules : la quantité de marchandises livrées sur une tournée ne doit pas dépasser la capacité du véhicule qui l'assure [15].

Le but de cette recherche est de trouver une solution optimale pour le problème de la tournée
objectifs suivants pour atteindre avec succès le but de la recherche.

Le VRP dans cette étude est le Problème d'acheminement des véhicules capacités
(CVRP).

L'objectif est de minimiser la distance parcourue tout en assurant la satisfaction des clients. En outre, la capacité du véhicule ne doit pas dépasser la plage du nombre total de clients. Chaque véhicule provient d'un seul et même dépôt, sert les clients à des endroits connus et revient au dépôt.

Le problème de routage est décrit comme suit :

Chaque véhicule sert un ensemble de clients et revient à :

- Un certain nombre de véhicules proviennent d'un seul dépôt.
- L'objectif est de réduire la distance totale parcourue.
- La demande totale des clients desservie par un véhicule doit être le même dépôt
- Tous les clients sont desservis par un seul véhicule.
- Ne pas dépasser la capacité du véhicule [16].

10. Conclusion

Nous avons présente dans ce chapitre les principaux concepts de problème de la tournée des véhicules et les travaux de recherche sur ce problème, nous avons vu que cette problématique est divisée en plusieurs types. Le chapitre suivant on présentera les méthodes de résolution du problème de VRP, et on va entamer quelques algorithmes a sa voire (fierfly ; colonis de fourmis, optimisation par essais particuliers et génétique algorithme).

Chapitre II

Les algorithmes Méta-heuristique

1. Introduction

Méta heuristique est une méthode générique pouvant optimiser une large gamme de problème différents, sans nécessite de changements profonds dans l'algorithme employé.

Autrement dit :

Les Méta heuristiques sont une forme d'algorithmes d'optimisation stochastique hybrides avec une recherche locale.

Le terme méta est donc pris au sens où les algorithmes peuvent regrouper plusieurs heuristiques.

Des heuristiques plus poussées ont été mises au point et ont données naissance à une nouvelle famille d'algorithme (les Meta heuristiques).

Le but d'un méta heuristique est de réussir à trouver un optimum global pour cela ; l'idée est à la fois de parcourir l'espace de recherche ; et d'Explorer les zones qui paraissent prometteuse mais sans être « piégé par un optimum local.

Les Méta heuristique sont souvent inspirées des processus naturels est sont de plus hybridées avec d'autres méthodes de recherche opérationnelle ; et généralement une marche aléatoire dans l'espace de recherche guide par les heuristiques [17].

Dans ce chapitre nous allons présenter des nombreux Meta heuristique on été proposés pour résoudre le (VRP) ; ces Meta heuristique permis d'avoir des résultat très compétitifs avec des instances larges, plusieurs algorithmes bases sur la (optimisation par essais particulières (OEP) ; colonies de fourmis ; les lucioles lampyrides ou fireflies (en anglais) et l'algorithme génétique (AG)) [18].

2. L'algorithme fireflies

L'algorithme fireflies est un algorithme évolutif ; les lucioles (lampyrides) ou fireflies (en anglais) ; sont des insectes qui appartiennent à la famille des abeilles. Les lucioles produisant une lumière froide sans fréquences infrarouge ou ultra-violet.

Cette lumière chimique est générée à partir de l'abdomen inférieur du corps de ces insectes.

La lumière peut être jaune, verte ou rouge pâle ; avec une longueur d'onde entre 510 à 610 nanomètres [19].

Environ 2000 espèces de lucioles se trouvent dans les régions tropicales, la plupart dans les zones humides couvertes de bois où les larves de ces insectes trouvent des sources de nourriture.

Ces larves appelées (GLOWWORMS) sont aussi capables d'émettre de la lumière pour la plupart des espèces de cet insecte, le mâle et la femelle sont capables de voler ; à part quelques espèces où les femelles sont incapables de le faire [20].

Les lucioles hibernent lors de l'hiver à l'étape de larve, après plusieurs semaines d'alimentation, elles émergent en tant qu'adultes. Les larves de la plupart des espèces de cet insecte sont prédatrices qui se nourrissent sur les autres larves, les escargots et les lomaxes. Tandis que les autres se nourrissent sur le pollen ou le nectar des plantes. La plupart de ces insectes sont toxiques aux autres prédateurs [21].

La production de la lumière dans ces insectes est due à un type de réaction chimique appelé bioluminescence. Ce processus s'effectue dans un organe spécial, au niveau du corps de la luciole, qui est l'abdomen inférieur de la luciole.

L'enzyme luciférase réagit sur le luciférien, dans la présence du magnésium, L'ATP et l'oxygène pour produire la lumière.

Pendant l'étape de larve, la bioluminescence sert comme un signal d'alerte aux autres prédateurs, afin de protéger la luciole de ces dernières. Tandis que ce phénomène a une autre utilisation lors de l'âge adulte de cet insecte, les éclats lumineux représentent un moyen de sélection de partenaires [22].

Des espèces de lucioles sont distinguées grâce à leur modèle spécifique des flashes de lumière que les mâles diffusent dans le but de trouver des femelles. D'autre part, il existe une espèce dont les femelles reproduisent les signaux des autres espèces afin d'attirer les mâles de ces espèces comme proies.

L'intensité de la lumière à un point de distance r à partir d'une source de lumière oblique diminue avec la croissance de la distance r , dont le terme est $(I \text{ ALPHA } 1/R^2)$, due à l'absorption de la lumière par l'air. L'absorption de la lumière par l'air rend les lucioles visibles que pour une distance limitée [23].

L'algorithme firefly est un méta heuristique, bio inspirée, introduite par Dr Xin-She Yan à l'Université Cambridge en 2007. L'algorithme est basé sur le principe d'attraction entre les lucioles et simule le comportement d'un essaim de lucioles dans la nature ce qui lui donne beaucoup de similarités avec d'autres méta heuristiques basées sur l'intelligence collective du groupe.

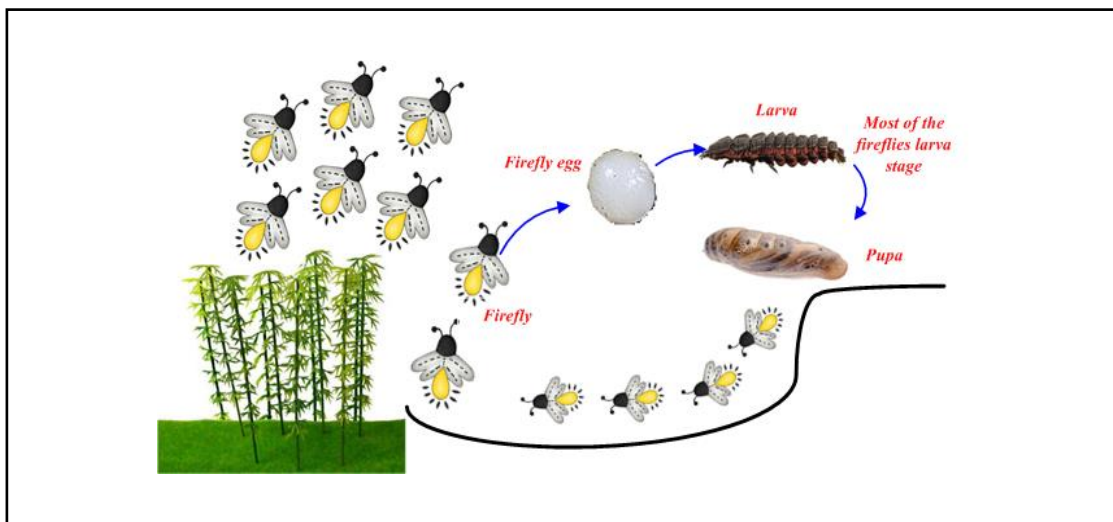


Figure 2.1 : le fonctionnement d’algorithme fireflies.

L’algorithme prend en considération les trois points suivants : [24] [25].

1. Toutes les lucioles sont unisexes, ce qui fait l’attraction entre celles-ci n’est pas en fonction de leur sexe.
2. L’attraction est proportionnelle à leurs luminosités, donc pour deux lucioles, la moins lumineuse se déplacera vers la plus lumineuse. Si aucune luciole n’est lumineuse qu’une luciole particulière, cette dernière se déplacera aléatoirement.
3. La luminosité des lucioles est déterminée en fonction d’une fonction objective (à optimiser).

2.1 Algorithme fireflies

```
Définir une fonction objective  $f(\mathbf{x})$ ,  $\mathbf{x} = (x_1, \dots, x_d)^T$   
Générer une population de lucioles  $\mathbf{x}_i$  ( $i = 1, 2, \dots, n$ )  
Définir l'intensité de lumière  $I$  à un point  $x_i$  par la fonction objective  $f(\mathbf{x}_i)$   
Déterminer le coefficient d'absorption  $\gamma$   
Tant que ( $t < \text{Max Génération}$ )  
  Pour  $i = 1$  jusqu'à  $n$   
    Pour  $j = 1$  jusqu'à  $n$   
      Si ( $I_i < I_j$ )  
        Déplacer la luciole  $j$  vers la luciole  $i$   
      Fin Si  
    Varier l'attraction en fonction de la distance  $r$  via  $\exp[-\gamma r]$   
    Evaluation des Nouvelles solutions et mettre à jour l'intensité de lumière  
  Fin Pour  $j$   
  Fin Pour  $i$   
  Classifier les lucioles et trouver la meilleure solution  
Fin Tant que  
Visualiser les résultats
```

3. Algorithme de Colonie des fourmis

Les insectes, comme les fourmis, les abeilles ou les termites sont généralement imaginées d'une manière simple, des animaux non intelligents, néanmoins ils exhibent collectivement des compétences impressionnantes pour résoudre les problèmes. Inspirant de ces insectes, les recherches dans les décennies passées sont guidées en quelques progrès fascinants dans le champ d'algorithmes naturels. Ces algorithmes imitent la nature d'une manière ou d'une autre.

Les réseaux de neurones imitent la structure de notre cerveau humain et les algorithmes génétique simulent l'évolution ils sont caractérisés par le parallélisme inhérent, l'adaptation, le feed-back positif et quelques éléments de l'aléatoire [26].

La recherche guidée par Marco Dorigo produisit un nouveau membre de cette classe d'algorithmes : l'algorithme du système de fourmis. Ici, la manière dont ces insectes trouvent le chemin le plus court à partir d'une ressource alimentaire à leur nid est copiée dans une tentative de résoudre les problèmes durs d'optimisation combinatoire a l'origine, l'optimisation par colonie de fourmis a été conçus pour résoudre le problème du voyageur de commerce (pvc).

L'objectif de ce problème est de trouver la tournée la plus courte pour un ensemble de villes données. Une matrice fournissant les distances ; $d(ij)$ entre toutes les paires de villes est utilisée pour estimer la longueur d'une tournée [27].

Les colonies de fourmis ; et plus généralement les insectes sociaux sont des systèmes distribués qui, dans L'esper de la simplicité de leurs individus, présentent une forte structure d'organisation sociale.

Comme un résultat de cette organisation, les colonies de fourmis peuvent accomplir les tâches complexes qui sont en quelques sortes loin dépasser les capacités d'individu pour une seule fourmi.

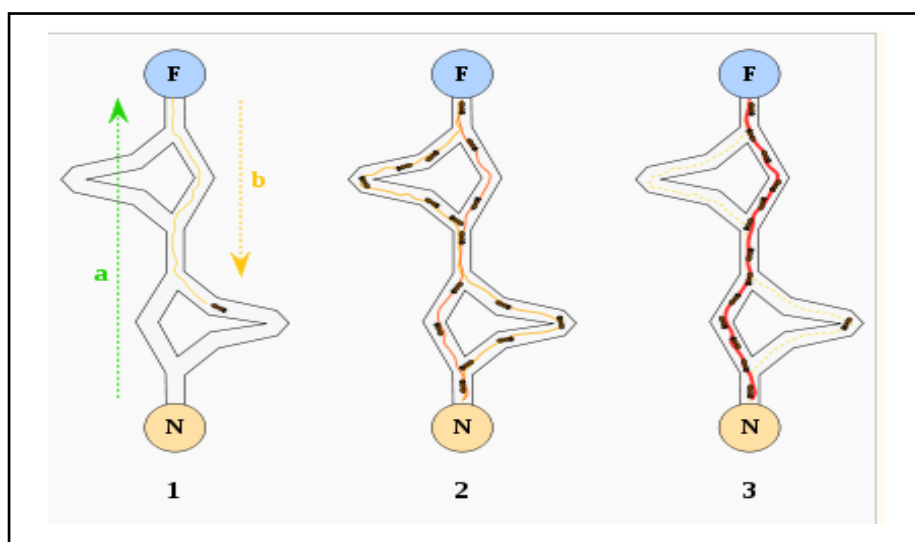


Figure 2.2 : La fonctionnement d'algorithme de colonies de fourmis.

Notre modèle pour ce comportement est :

1. La première fourmi trouve la source de nourriture (F), via un chemin quelconque (a), puis revient au nid (N) en laissant derrière elle une piste de phéromone (b).
2. Les fourmis empruntent indifféremment les quatre chemins envisageables, mais le renforcement de la piste rend plus attractif le chemin le plus court.
3. Les fourmis empruntent le chemin le plus court, les portions longues des autres chemins perdent leur piste de phéromones.

3.1 Variantes Principales

L'algorithme de colonies de fourmis a été à l'origine en particulier utilisé pour produire des solutions quasi –optimales au problème du voyageur de commerce, puis d'une façon plus générale, aux problèmes d'optimisation combinatoire.

3.2 Algorithme de colonie de fourmis

<p>Initialisation des pistes de phéromone ;</p> <p>Boucler tant que critère d'arrêt non atteint :</p> <p>Construire les solutions composants par composants ,</p> <p>Utilisation (facultative) d'une heuristique ,</p> <p>mise à jour des pistes de phéromone ;</p> <p>Fin de la boucle .</p>
--

Des algorithmes de colonie de fourmis ont été appliqués à la plupart des problèmes d'optimisation combinatoire. Comme énormément de Meta heuristiques, L'algorithme de base a été adapté aux problèmes dynamiques, en variable réelles, aux problèmes stochastique, multi-objectifs ou aux implémentation parallèles.

L'algorithme (ACO) optimisant le problème du voyageur des commerce :

1. Une fourmi choisit un trajet, et trace une piste de phéromone.
2. La totalité des fourmis parcourt un certain nombre de trajet, chaque fourmi déposant une quantité de phéromone proportionnelle à la qualité du parcours.
3. Chaque arête du meilleur chemin est plus renforcée que les autres.
4. L'exploration fait disparaître les mauvaises solutions.

Et enfin il est intéressant de noter que bien qu'une seule fourmi soit capable de construire une solution seul le comportement de toute la colonie qui crée le chemin le plus courts [28].

4. Optimisation par essais particulières (PSO)

L'optimisation par essais particulières (Particle Swarm Optimization) est une méthode qui s'inspire de la biologie pour résoudre des problèmes d'optimisation.

Comme les réseaux de neurones artificiels, les Algorithmes génétique ou les Algorithme de colonie de fourmis.

Le Particle Swarm Optimization (PSO) est un Algorithme bio-inspiré .il repose sur les principes d'auto-organisation qui permettent à un groupe

d'organisme vivants d'agir ensemble de manière complexe, à partir de “ règles “ simples.

Le PSO s'inspire du modèle développé par Craig Reynolds pour simuler le déplacement grégaire de certains animaux (troupeaux de bovins, volées d'oiseaux ...).

Dans ce modèle chaque oiseau artificiel au “ boi ” (bird-oid Object) se déplace aléatoirement en suivant trois règles simples :

La cohésion : les boids sont attirés vers la position moyenne du groupe.

L'alignement : les boids suivant le même chemin que leur voisin.

La séparation : pour éviter les collisions les boids gardent une certaine distance entre eux.

Le PSO introduit un autre principe :

Les boids ne se déplacent pas aléatoirement, ils ont un objectif à atteindre. Celui-ci est déterminé par une fonction à optimiser Ou “ fonction objectif “ qui est fournie par l'utilisateur, et qui dépend de l'application concernée.

On a fonctionné cet Algorithme comme des vecteurs de nombre à dimension, les positions, les Vitesses initiales étant souvent défini aléatoirement.

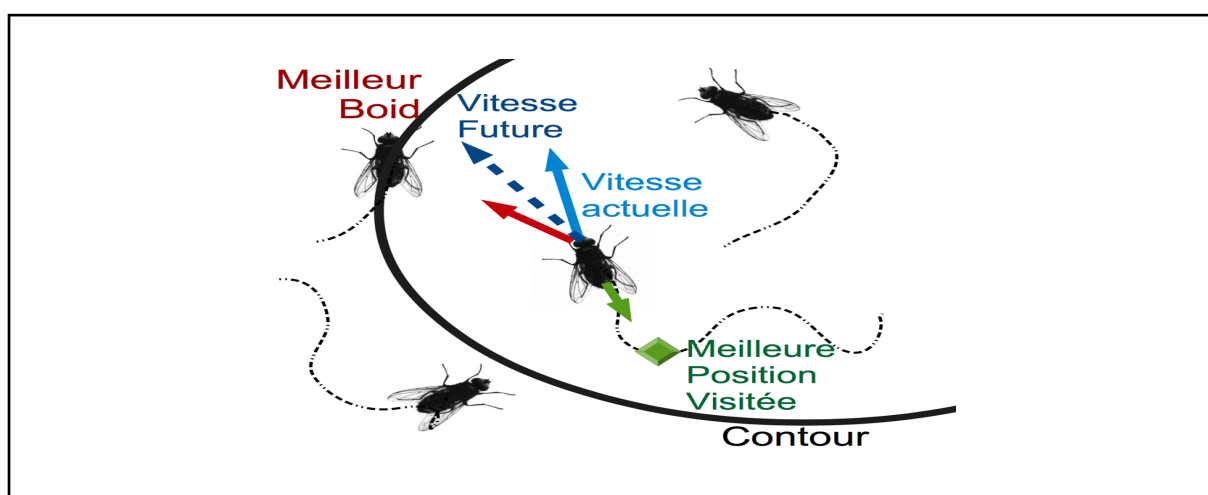


Figure 2.3 : La fonctionnement d'algorithme d'optimisation par des essaims particulières.

- Un vecteur Vitesse partant de X et allant vers le meilleur boid de l'essai (flèche Rouge sur le schéma).
- Un vecteur Vitesse allant vers la meilleure position que le boid a visité (flèche verte).
- Le vecteur Vitesse précédent (flèche bleue) [29].

4.1 Algorithme d'optimisation par essais particulières: [30].

1. **Initialiser** aléatoirement N particules : position et Vitesse.
2. **Evaluer** les positions des particules
3. **Pour** chaque particule i , $\overrightarrow{p_{best_i}} = \overrightarrow{x_i}$
4. **Calculer** $\overrightarrow{G_{best}}$ selon (1.3.4)
5. **Tant que** le critère d'arrêt n'est pas satisfait **faire**
6. **Déplacer** les particules selon (1.3.1) et (1.3.2)
7. **Evaluer** les positions des particules
8. **Mettre à jour** $\overrightarrow{p_{best_i}}$ et $\overrightarrow{G_{best}}$ selon (1.3.3) et (1.3.4)
9. **Fin**

5. Algorithme génétique

Les algorithmes génétiques sont des algorithmes d'exploration fondés sur les mécanismes de la sélection naturelle et de la génétique. Ils sont basés sur les principes de survie de structures les mieux adaptées et les échanges d'information.

A chaque génération un nouvel ensemble de créatures artificielles (codées sous forme de chaîne de caractère) est construit à partir des meilleurs éléments de la génération précédente.

Bien que reposant fortement sur le hasard (et donc sur un générateur de nombres aléatoires) ces algorithmes ne sont pas purement aléatoires [31].

5.1 Les éléments d'Un algorithme génétique

Les algorithmes génétiques sont inspirés de la génétique classique : on considère une (population)de points repartis dans l'espace. Avant d'expliquer en détail le fonctionnement d'un algorithme génétique, nous allons présenter quelques mots de vocabulaire relatifs à la génétique.

Ces mots sont souvent utilisés pour décrire un algorithme génétique : [32].

5.1.1 Génotype ou chromosome

Le génotype est constitué de gènes situés sur des chromosomes stockés dans le noyau des cellules sous la forme d'une longue chaîne d'acide désoxyribonucléique (ADN). Dans la nature, L'ADN est un polymère constitué par l'enchaînement de quatre molécules, les nucléotides adénine (A), cytosine (C), guanine (G), et la thymine (T).

On peut donc décrire L'ADN par des chaînes de quatre caractères ACGT. L'ADN constitue l'ensemble des chromosomes, ou les génomes d'un individu. C'est une autre façon de dire (individu) [31].

5.1.2 Gène

Un chromosome est composé de gènes. Dans le codage binaire, un gène vaut soit 0 soit 1.

5.1.3 Phénotype

Le phénotype est l'ensemble des protéines et des enzymes qui peuvent être fabriqués à partir de L'ADN. En fait L'ADN est copiée par un messenger (ARN) qui au niveau du ribosome, se traduit en chaîne d'acides aminés formant les protéines et les enzymes.

Compte une protéine (un enzyme) par gène. Ce sont les protéines et les enzymes qui dictent la structure et le comportement des cellules qui permettent à un individu de :

- Réaliser des tâches dans son environnement.
- Survive.
- Reproduire a des taux différents [31].

Chaque génotype représente une solution potentielle a un problème d'optimisation .la valeur de cette solution potentielle est appelée le phénotype [32].

5.2 Population initiale

L'AG démarre avec une population composée de N individus dans le codage retenu. Le choix des individus conditionne fortement la rapidité de L'algorithme. Si la position de L'optimum dans l'espace de recherche est totalement inconnue, il est intéressant que la population soit répartie sur tout l'espace de recherche. Si par contre des informations a priori sur le problème sont disponible, il parait évident de générer les individus dans un espace particulier afin d'accélérer la convergence. Disposant d'une population initiale souvent non homogène, la diversité de la population doit être entretenue aux cours des générations afin d'explorer le plus largement possible l'espace de recherche. C'est le rôle des opérateurs de croisement et de mutation.

5.3 Codage d'individus

Avant de passer à l'explication des différents processus génétiques, il faut tout d'abord expliquer le codage des individus. La procédure normale pour coder un algorithme génétique ayant plusieurs paramètres est de coder chaque paramètre comme séquence de bites.

Les séquences sont ensuite tronquées lune après l'autre pour former une grande séquence, le chromosome qui représente le vecteur des paramètres. Chaque séquence du vecteur total représente un gène, et la valeur de chaque gène est un allèle.

Exemple :

Soit un vecteur \bar{x} compose de trois paramètre x_1 , x_2 et x_3 , codes a 4 bits

$$x_1=0011$$

$$x_2=1010$$

$$x_3=0100$$

La séquence totale serait la suivante :

$$x_1 \quad x_2 \quad x_3$$

$$\bar{x} = (0011 / 1010 / 0100)$$

Un chiffre code en binaire dans ce cas ne représente pas une valeur spécifique mais plutôt un intervalle.

Par exemple, si on code des valeurs de 0 à 1 a l'aide de 5 bits, Le chiffre 11111 représente l'intervalle de 31/32/ a 32/32, plutôt que la valeur de 31/32 exactement [33].

5.3.1 Codage binaire

Ce codage a été le premier a être utiliser dans le domaine des AG. Il présente plusieurs avantages : alphabet minimum 0,1, facilite de mise au point d Operateurs génétiques et existence de fondements théoriques (théorie sur les schémas).

Néanmoins ce type de codage présente quelques inconvénients :

1. Les performances de l'algorithme sont dégradées devant les problèmes d'optimisation de grande dimension a haute précision numérique pour de tels problèmes, les AG bases sur les chaines binaires on de faibles performances.

2. La distance de Hemming entre ces deux nombres voisin (nombre de bits différents) peut être assez grande dans le codage binaire : l'entier 7 correspond à la chaîne 0111 et la chaîne 1000 correspond à l'entier 8. Or la distance de Hemming entre ces deux chaînes est de 4, ce qui crée bien souvent une convergence, et non pas l'obtention de la valeur optimale. Cet inconvénient peut être évité en utilisant un codage de Gray par exemple.

5.3.2 Codage réel

La représentation des solutions dans le cadre des AG n'est pas nécessairement réduite à un alphabet de faible cardinalité (0,1), il existe toute une école pour laquelle la représentation la plus efficace est celle qui s'appuie sur des nombres réels. Cette représentation est à la base de l'approche évolutionnaire (Evolution stratégique). Ce type de codage présente certains avantages par rapport au codage binaire :

- Le codage réel est robuste pour les problèmes considérés comme difficile pour le codage binaire.
- Ce codage nécessite une adaptation des opérateurs de croisement et mutation [34].

5.4 Algorithme génétique

- 1) Initialiser la population initiale \mathbf{P} .
- 2) Evaluer \mathbf{P} .
- 3) Tant Que (Pas Convergence) faire :
 - a) $\mathbf{P}' =$ Sélection des parts dans \mathbf{P}
 - b) $\mathbf{P}' =$ Appliquer Operateur de Croisement sur \mathbf{P}'
 - c) $\mathbf{P}' =$ Appliquer Operateur de Mutation sur \mathbf{P}'
 - d) $\mathbf{P} =$ Remplacer les anciens de \mathbf{P} par leurs Descendants de \mathbf{P}'
 - e) Evaluer \mathbf{P}

Fin Tant Que

5.5 Schéma récapitulatif d'Algorithme Génétique

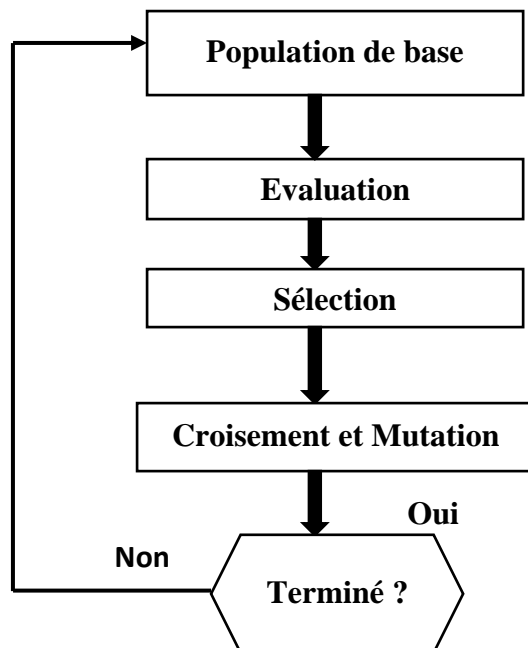


Figure 2.4 : Schéma récapitulatif

5.6 Les opérateurs génétiques

5.6.1 L'évaluation

L'évaluation permet de s'assurer que les individus performants seront conservés, alors que les individus peu adaptés seront progressivement éliminés de la population. L'évaluation d'un individu ne dépend pas de celle des autres individus, pour calculer le coût d'un point de l'espace de recherche, on utilise une fonction d'évaluation f , cette fonction est souvent une transformation g de la fonction objectif $f(x) = g(\phi(x))$, appelée aussi la fonction fitness, le résultat fourni par cette fonction va permettre de sélectionner ou de refuser un individu pour ne garder que l'individu ayant le meilleur coût en fonction de la population courante.

L'évaluation de chaque individu consiste à transformer la chaîne binaire 0 et 1 du chromosome en une valeur réelle, appelée valeur d'adaptation [35].

5.6.2 Opérateur de sélection

Comme son nom indique, la sélection permet d'identifier statistiquement les individus de la population courante qui seront autorisés à se reproduire. Cette opération est fondée sur la performance des individus, estimée à l'aide de la fonction d'adaptation. Cet opérateur est peut-être le plus important puisqu'il permet aux individus d'une population de survivre, de se reproduire ou de mourir. En règle générale la probabilité de survie d'un individu sera directement liée à son efficacité relative au sein de la population.

Il existe différentes méthodes de sélection. En les plus célèbres :

- la sélection par roulette.
- la sélection par tournois.
- la sélection Elitiste.

5.6.2.1 La sélection par roulette

Elle consiste à associer à chaque individu un secteur de la roulette dont la surface est proportionnelle à sa performance. On tire alors un nombre aléatoire de distribution uniforme entre 0 et 1, puis on regarde quel est le secteur sélectionné et on reproduit individus correspondants. Illustre une roulette dans chaque individu se voit attribuer un secteur dont l'angle est proportionnel à sa performance donnée par l'équation :

$$P_s(X_i) = \frac{F(x_i)}{\sum_{K=1}^N F(x_K)}$$

Où P_s est la proportion qui fait la valeur de la fonction objectif $F(x_i)$ d'un individu x_i , par rapport à la somme de tous les objectifs des N individus de la population. De cette façon, plus un individu a une forte qualité F par rapport au reste de la population ; plus celui-ci a de chances de se produire. Toutefois ; comme la sélection se fait aléatoirement, Les moins bons individus ont tout de même une chance de faire partie de l'ensemble de reproduction [36].

5.6.2.2 La sélection par tournoi

Cette méthode qui utilise des comparaisons entre individus ; au même titre que la sélection par rang, ne nécessite en revanche aucun tri de la population. Le principe consiste à choisir uniformément T individus et à sélectionner le meilleur de ces T individus. Le choix de T ; qui représente la taille du tournoi ; permet de faire varier la pression sélective [36].

5.6.2.3 La selection Elitiste

Elle consiste à copier un ou plusieurs des meilleurs chromosomes dans la nouvelle génération ; Ensuite on génère le reste de la population selon l'algorithme de reproduction usuel.

Cette méthode améliore considérablement les algorithmes génétiques, car elle permet de ne pas perdre les meilleures solutions [36].

5.7 L'opérateur croisement

Le croisement est appliqué avec une P_c à deux individus parents préalablement sélectionnés, qui échangent des parties de leurs chaînes ; pour former deux nouveaux individus enfants.

Usuellement, le croisement se fait selon deux modes :

- Le croisement en un point.
- Le croisement en deux points.

5.7.1 Le croisement en un point

Consiste à choisir aléatoirement un point de coupure identique pour les chromosomes des deux parents ; par la suite les sous chaînes situées après sont interchangées pour former les deux enfants.

5.7.2 Le croisement en deux point

Dans ce type de croisement, deux points de coupure sont choisis au hasard et le contenu entre ces points est inter-changer pour former les chaînes des descendants. Cet opérateur est généralement considéré comme plus efficace que le précédent [36].

5.8 Operateur de mutation

Selon une probabilité P_m les descendants sont mutés ; en modifiant aléatoirement la valeur d'un ou plusieurs composants de leur chaîne. Il ne crée pas forcément de meilleures solutions au problème, mais garantit la diversité de la population et évitent l'établissement de populations uniformes incapables d'évoluer.

La mutation classique consiste à remplacer dans un chromosome binaire un bit par son inverse.

5.9 Remplacement

Cette opération appliquée après l'application successive des opérateurs de sélection de croissent et de mutation.

On trouve essentiellement deux méthodes de remplacement différente :

1. Remplacement stationnaire
2. Remplacement élitiste

5.9.1 Remplacement stationnaire

Les enfants remplacement automatiquement les parents sans tenir compte de leurs performances respective.

5.9.2 Remplacement élitiste

On garde au moins les individus possédant les meilleures performances d'une génération a la suivante [37].

6. Conclusion

Nous avons intéressé en général aux méthodes génériques de résolution des problèmes d'optimisation. Tous d'abord, les Méta- euristique pour la résolutions de ces problèmes. Et on va présentons dans le chapitre suivant sera consacré à l'étude D'algorithme génétique pour la résolution du problème de la tournée des véhicules avec contrainte de capacité.

Chapitre III

CVRP Par Génétique Algorithme

1. Introduction

Ce chapitre est consacré à la réalisation et l'implémentation de différents fonctionnalités de notre programme ; nous expliquerons notre méthode pour résoudre le problème de tournée de véhicule avec contrainte de capacité. Nous avons choisi d'utiliser les algorithmes génétiques pour résoudre ce problème. Des résultats expérimentaux sont présents afin de mesurer l'efficacité de notre solution.

2. L'objectif du travail

L'objectif de ce travail est l'utilisation d'une méta-heuristique pour résoudre le problème de la tournée de véhicule avec contrainte de capacité. Dont la fonction objective est la minimisation des coûts de transport, en termes de distance. Il consiste à visiter un nombre N de villes (clients) en un minimum de distance sans passer deux fois par la même ville, nous avons choisis : **les Algorithme Génétiques.**

3. L'organigramme de l'algorithme génétique

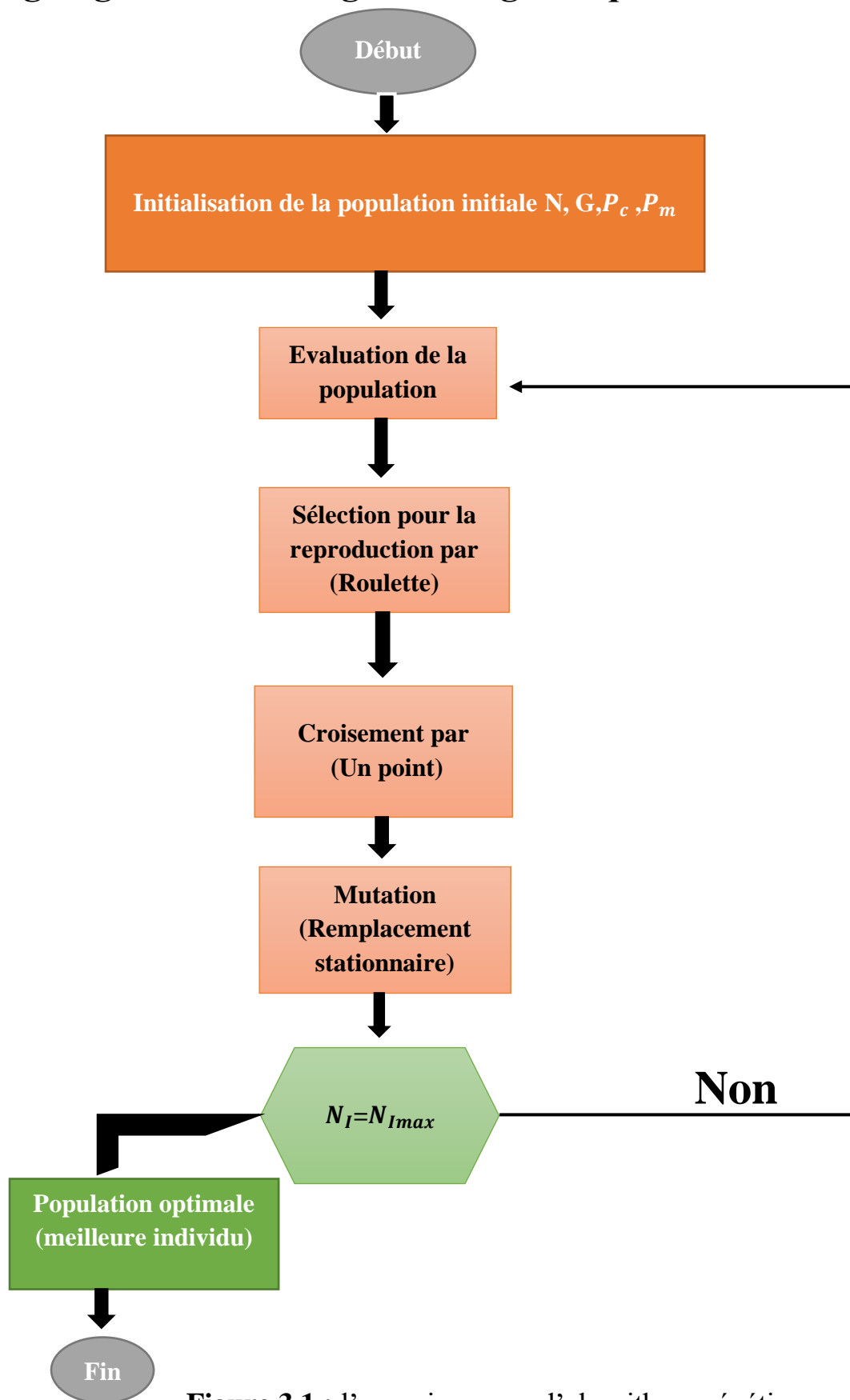


Figure 3.1 : l'organigramme d'algorithme génétique.

4. Description de l'approche proposée

Pour résoudre le problème, nous avons choisi les algorithmes Génétiques comme Meta-heuristique la motivation principale est les avantages suivants :

- 1- Les algorithmes génétiques sont parmi les premières méthodes utilisées.
- 2- Ils sont simples à implémenter.
- 3- Permet de traiter des espaces de recherche importants beaucoup de solutions, pas de parcourir exhaustivement (envisager).
- 4- Pour un nombre de solutions importants, les algorithmes génétiques permettent d'éliminer les solutions non valides.

4.1 Contrainte de tournée de véhicule avec capacité de véhicule

Dans notre approche nous proposons les contraintes suivantes :

- 1- Un dépôt de distribution logistique fournit des distributions pour (8 jusqu'à 40) villes.
- 2- La capacité des véhicules est différente.
- 3- Le dépôt de distribution dispose de (3 jusqu'à 6) véhicules.

4.2 Paramètre des problèmes

Les paramètres des problèmes concernent deux axes : les villes et le centre dépôt.

4.2.1 La partie qui concerne les attributs des villes

- n nombre des villes.
- T est la quantité demandée par chaque ville.

4.2.2 la partie qui concerne les attributs des dépôts

- m le centre de distribution.

- K nombre de véhicules.
- c la capacité de véhicule qui demande par chaque ville.

4.3 Les paramètres d'algorithme génétique

L'espace de recherche est l'ensemble de permutations de $\{1, 2, \dots ; n\}$ et la population est un ensemble des individus, chaque individu représente une solution.

La solution : est codée sous forme d'un vecteur, chaque case représente une ville dans l'ordre de la tournée, alors la taille des vecteurs représente la solution (l'ordre des villes par une tournée des véhicules).

La fonction fitness ou la fonction d'adaptation qui calcule le coût, la distance, et le coût de temps (la pénalité de retard) entre les villes pendant la tournée.

Fonction de fitness = minimiser (le coût de distance).

5. Environnement matériel

Ce travail a été implémenté sur un PC, caractérisé comme suit :

- Un Processeur : Intel(R) Core (TM) I3-3217U CPU @ 1.80GHz.
- Une RAM: 4 GO.
- Sous un système d'exploitation 64 bits.

6. Environnement de développement

Le langage de programmation **Matlab** est un langage de script, émulé par un environnement de développement du même nom ; il est utilisé à des fins de calcul numérique. Développé par la société the MathWorks.

Matlab permet de manipuler des matrices d'afficher des courbes et des données, de mettre en œuvre des algorithmes, de créer des interfaces des utilisateurs, et peut interfacer avec d'autres langages comme le C, C++, Java et Fortran [38].

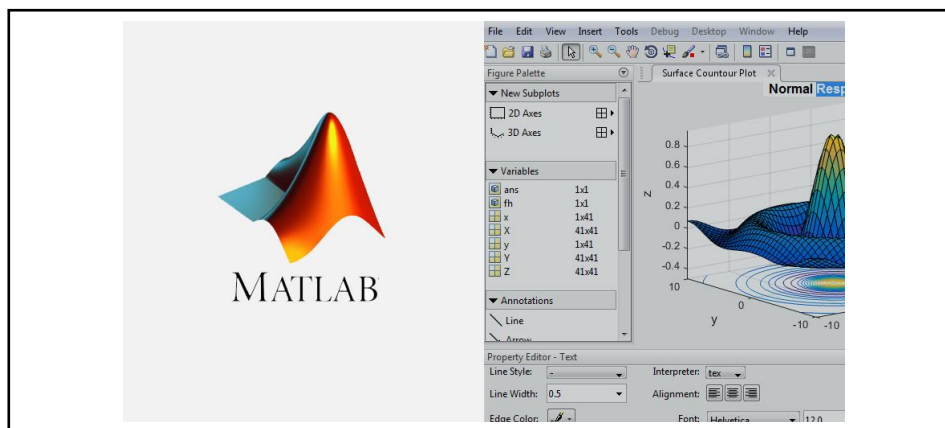


Figure 3.2 : langage de programmation Matlab.

Tableau 3.1 : Caractéristique de l’outil utilisé, MATLAB

Créateur	Cleve Moler
Développé par	MathWorks
Première version	1984
Dernière version	R2021a (17 mars 2021)
Écrit en	C, C++, Fortran et Java
Système d’exploitation	Microsoft Windows, macOS et GNU/Linux
Environnement	Linux, Unix, MacOS et Windows
Format lus	MATLAB M-File, MAT4, MAT-file, level5, version6, MAT-file, level5, version7, MAT-file, level5, version7.3 et BigTIFF
Format écrit	MATLAB M-File et Hierarchical Data Format
Type	Calcul numérique
Licence	Propriétaire
Site web	www.fr.mathworks.com

7. Expérimentation

Test 01 :

Dans la **figure (3.3)**, Nous avons affichons l'interface de l'application avec quelque lignes de code source.

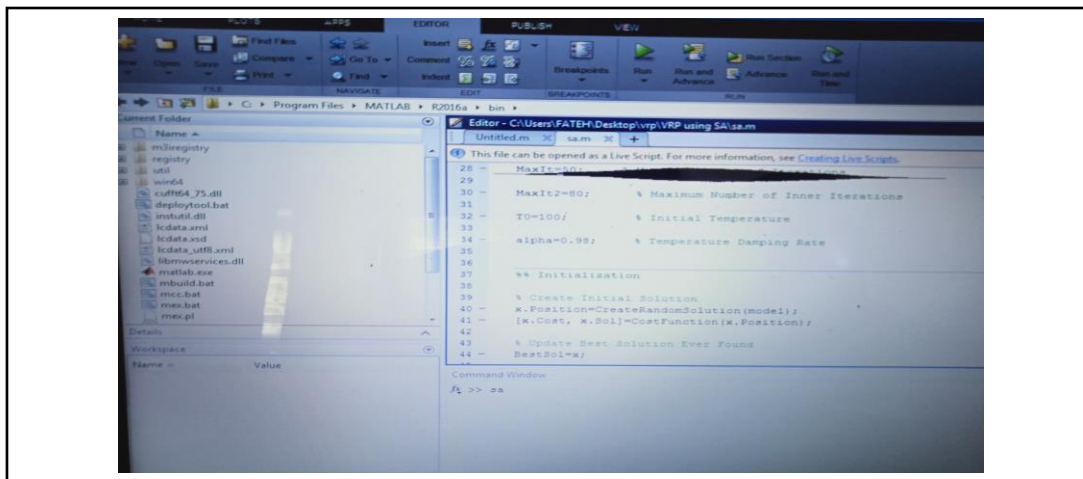


Figure 3.3 : L'interface de l'application Matlab.

Après l'exécution de l'application avec les paramètres suivants :

Nombre de villes = **8**.

Nombre de véhicules = **3**.

Nombre des itération $N_{Imax} = [10 \text{ jusqu'à } 20]$.

Nous obtiendrons alors les résultats suivants :

Les résultats sont montrer sur le graphe figure (3.3) suivant ; et reporter sur le tableau (3.2) suivant :

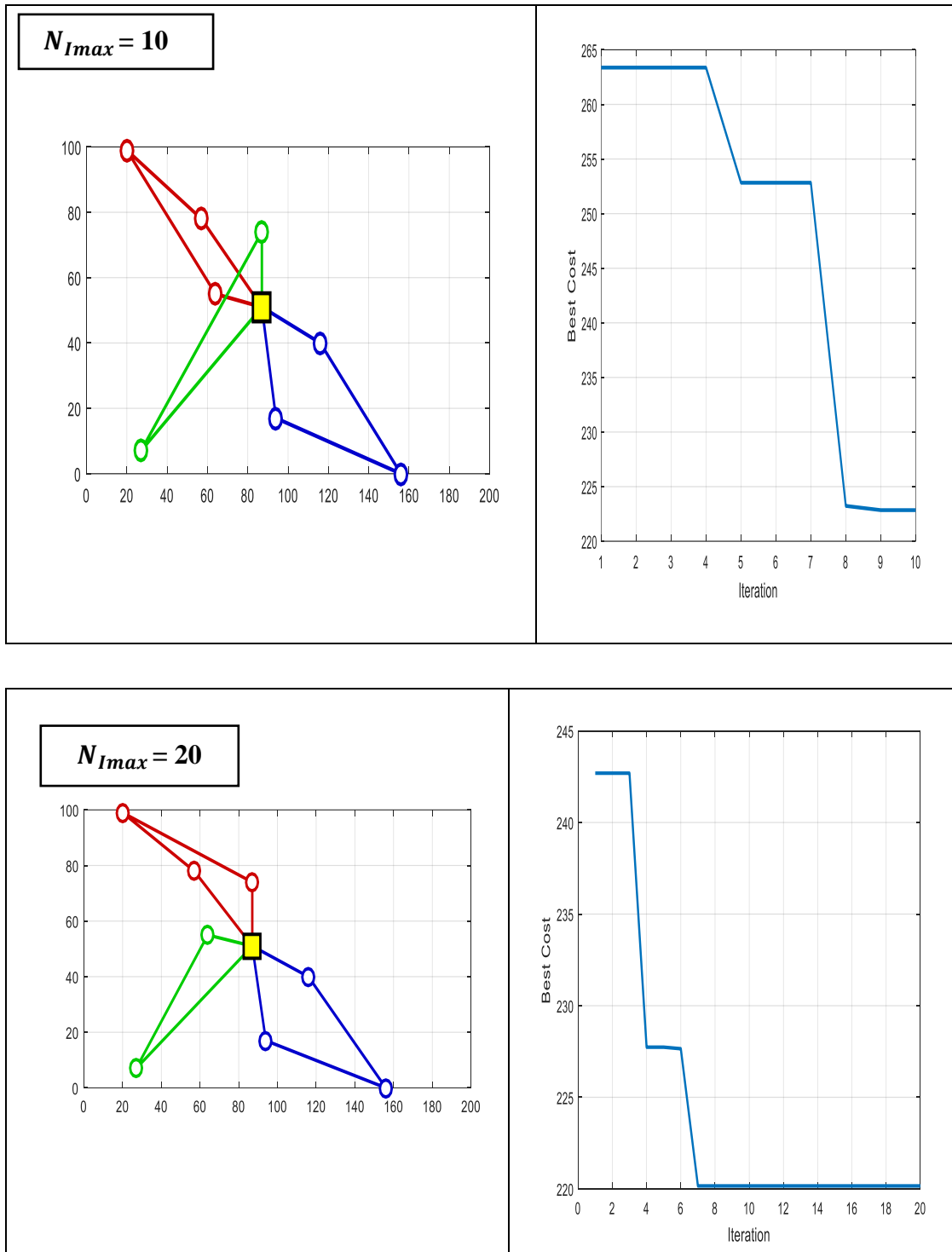


Figure 3.4 : Fenêtre graphique de l'exécution.

Nombre des itérations $N_{I_{max}}$	Solution optimal
10	222.16
20	222.12

Tableau 3.2 : Expérimentation pour mesurer l'influence de nombre des itérations.

8. Quelques test sur l'implémentation

Test 02 :

Le but du test est de distinguer l'influence de nombre des itérations sur la distance minimal (la solution optimale) et la performance.

Les paramètres sont choisis comme suit :

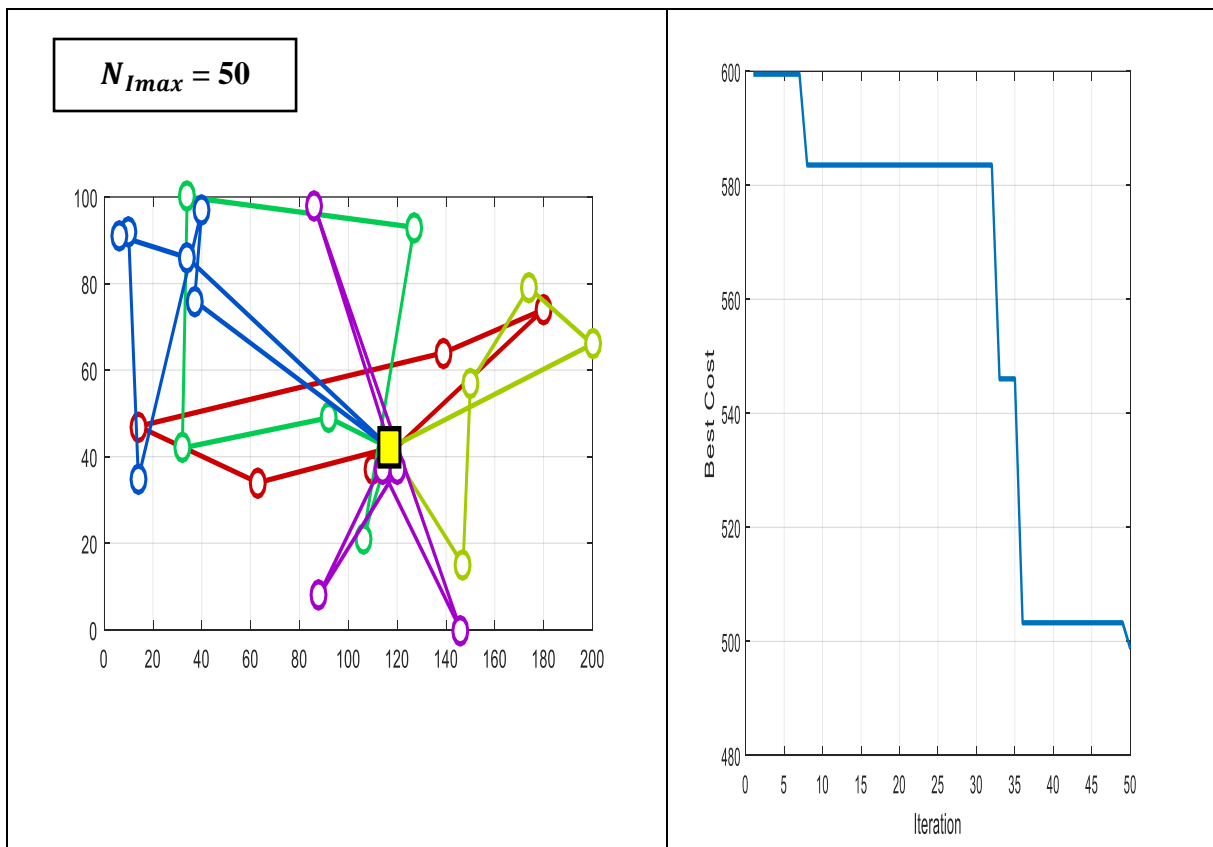
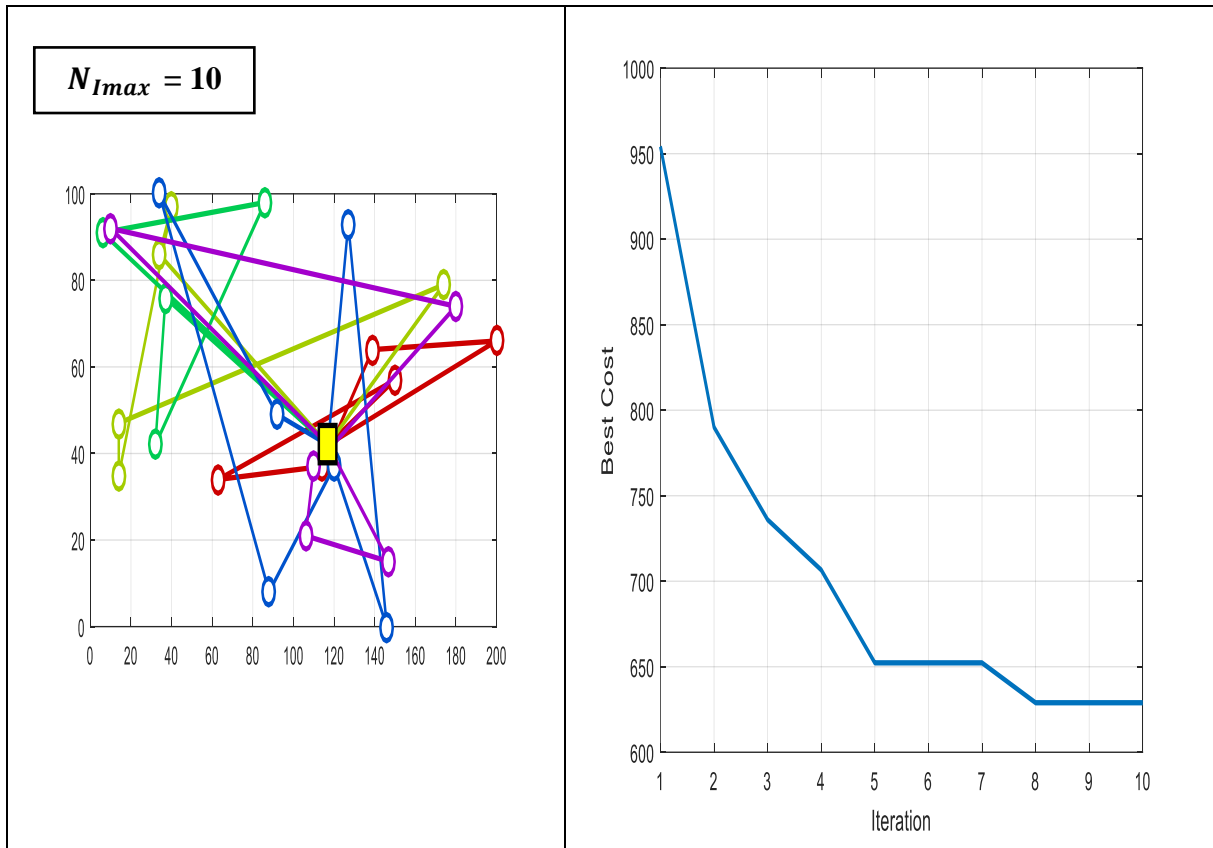
Nombre de villes =**25**.

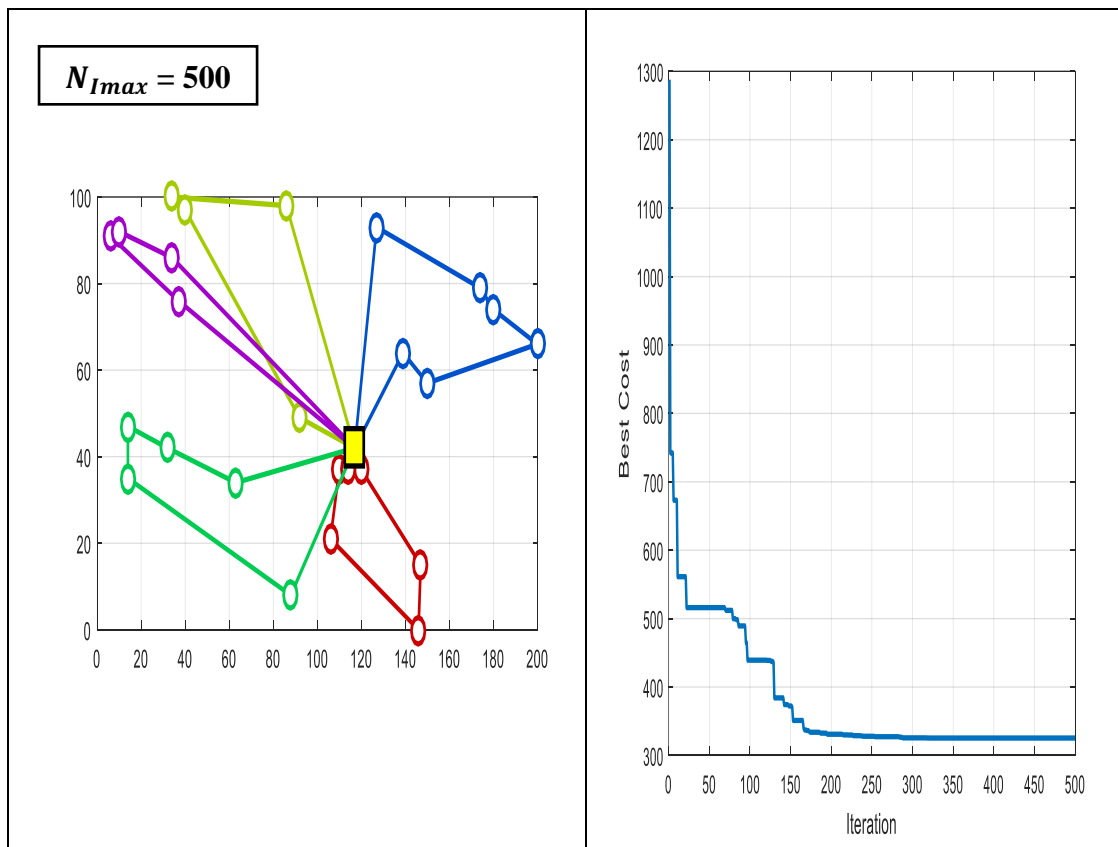
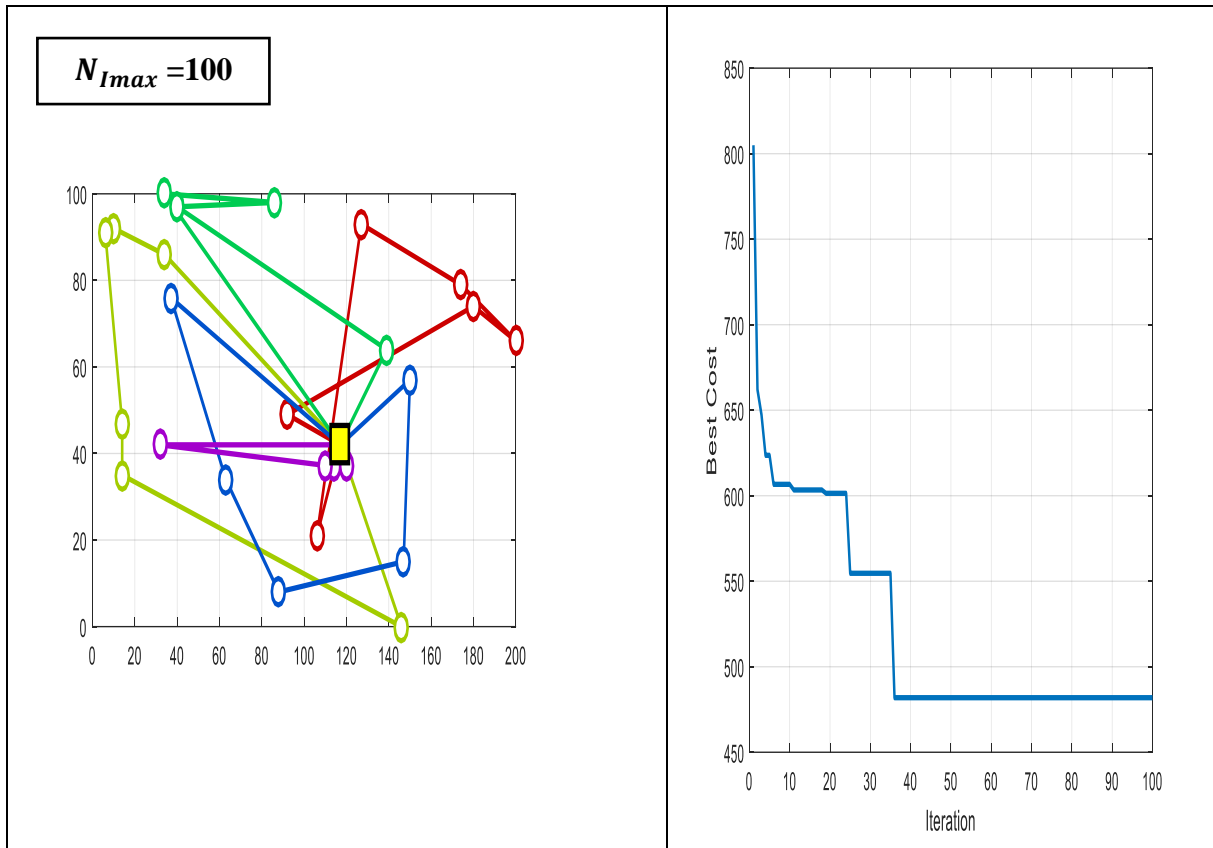
Nombre de véhicules =**5**.

Nombre des itération $N_{I_{max}} = [10 \text{ jusqu'à } 1000]$.

Nous obtiendrons alors les résultats suivants :

Les résultats sont montrés sur le graphe **figure (3.4)** suivant ; et reporter sur le **tableau (3.3)** suivant :





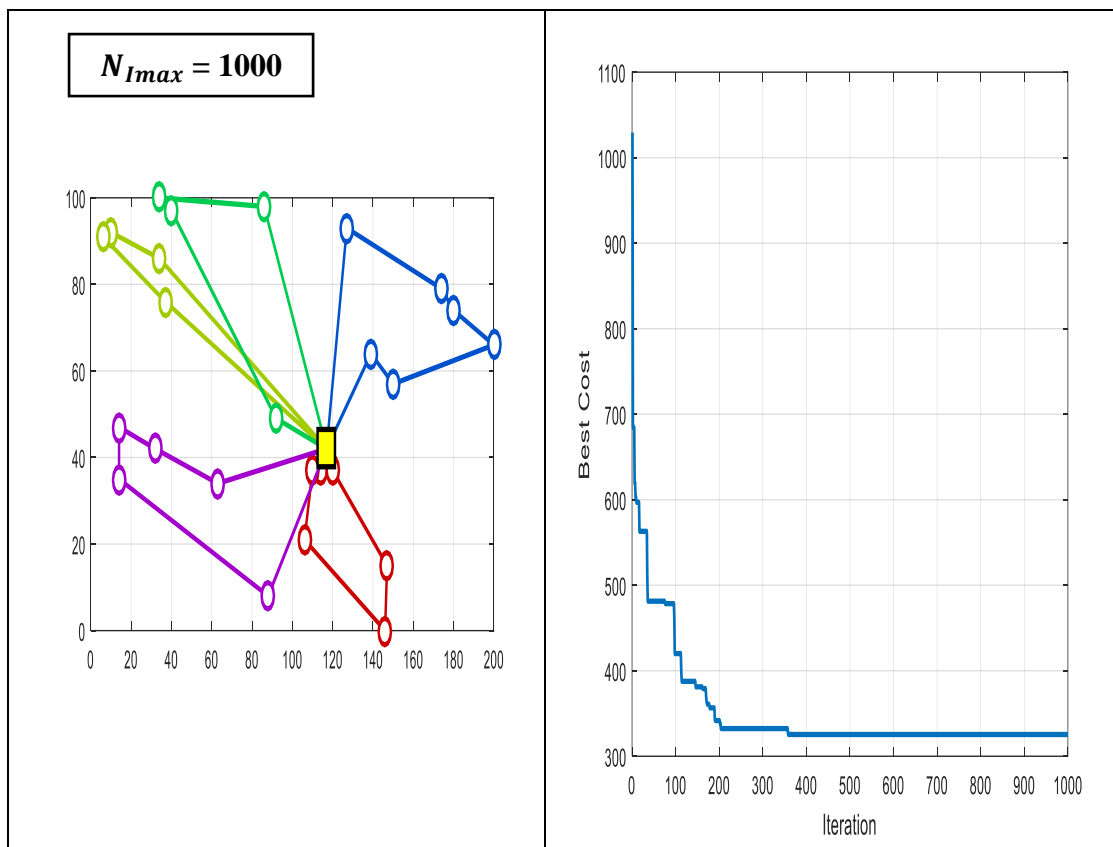


Figure 3.5 : Fenêtre graphique de l'exécution.

Nombre des itérations $N_{I_{max}}$	Solution optimal
10	628.95
50	498.75
100	481.89
500	325.44
1000	325.44

Tableau 3.3 : Expérimentation pour mesurer l'influence de nombre des itérations.

Test 03 :

Le but du test est de distinguer l'influence de nombre des itérations sur la distance minimal (la solution optimale) et la performance.

Les paramètres sont choisis comme suit :

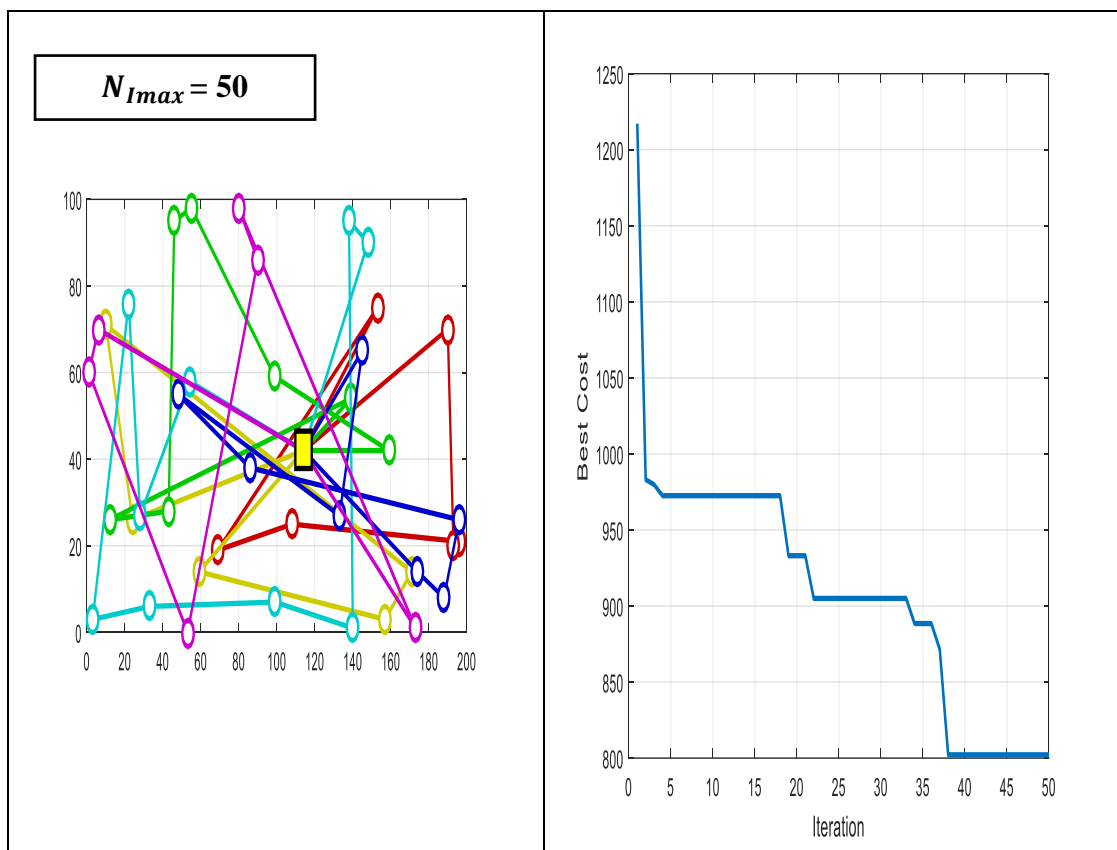
Nombre de villes =**40**.

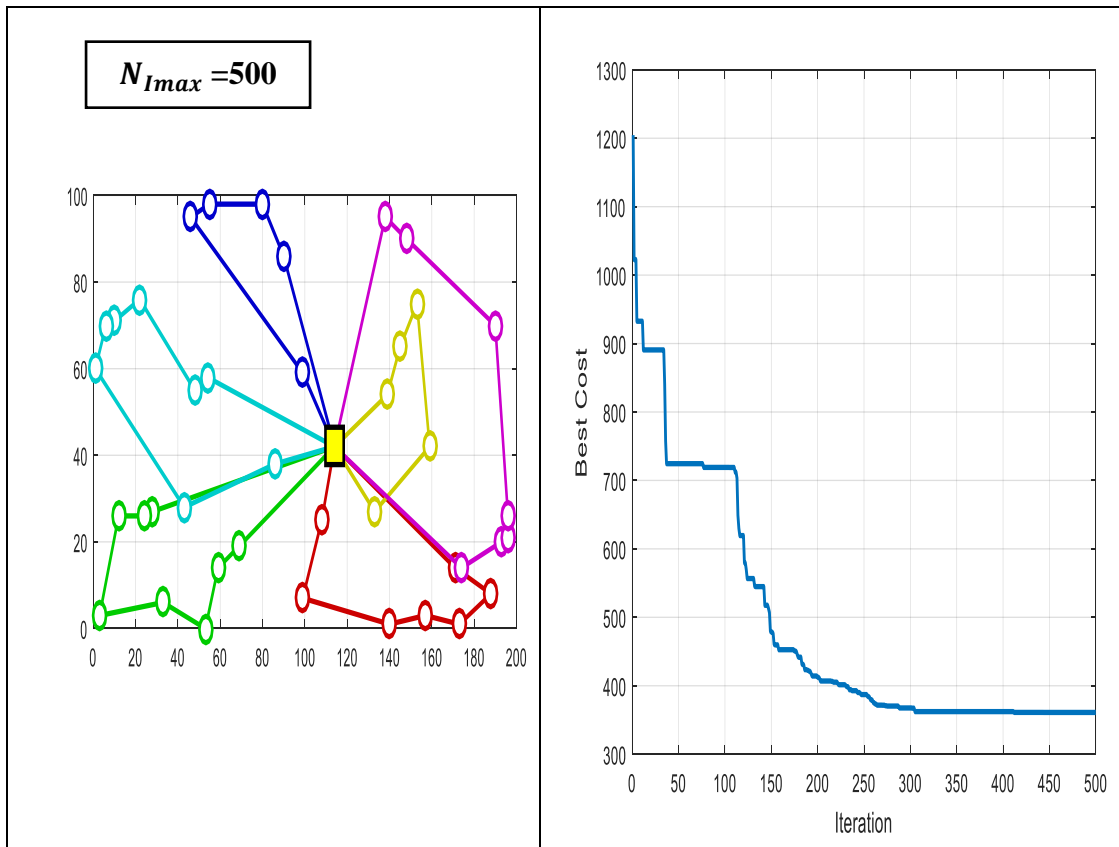
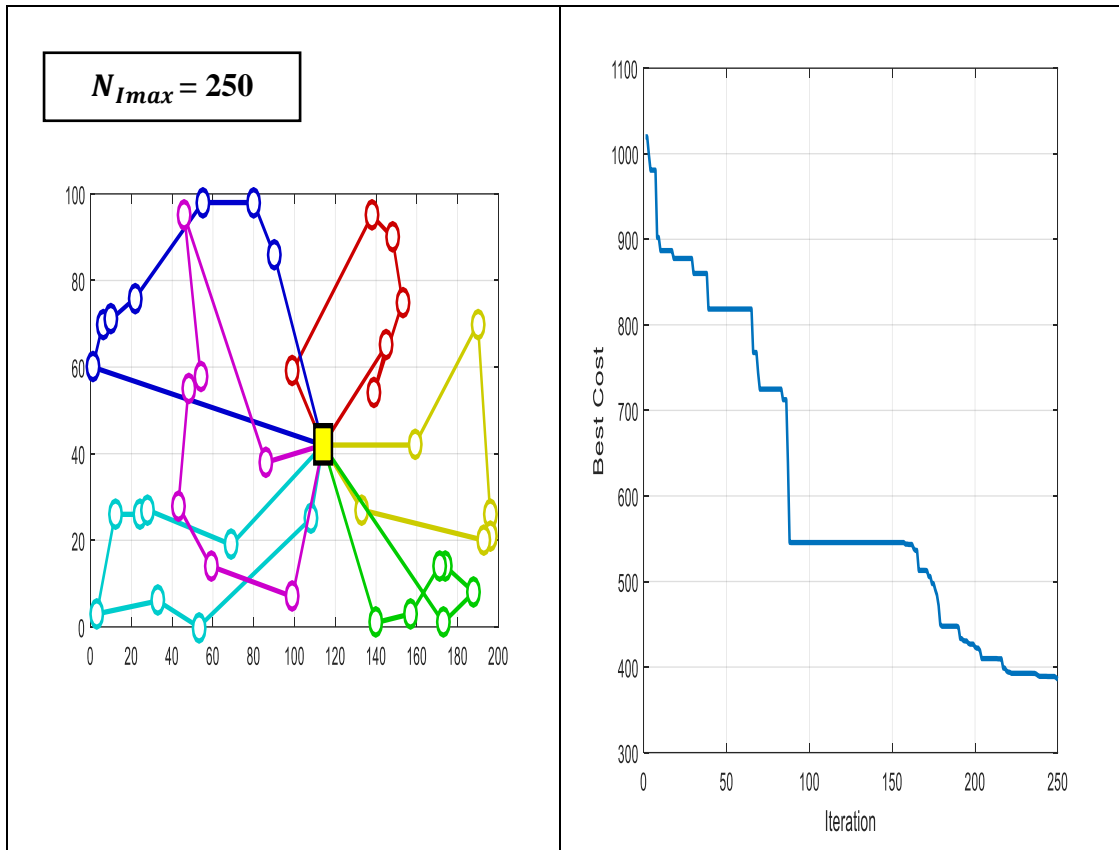
Nombre de véhicules =**6**.

Nombre des itération $N_{I_{max}} = [50 \text{ jusqu'à } 1200]$.

Nous obtiendrons alors les résultats suivants :

Les résultats sont montrés sur le graphe **figure (3.5)** suivant ; et reporter sur le **tableau (3.4)** suivant :





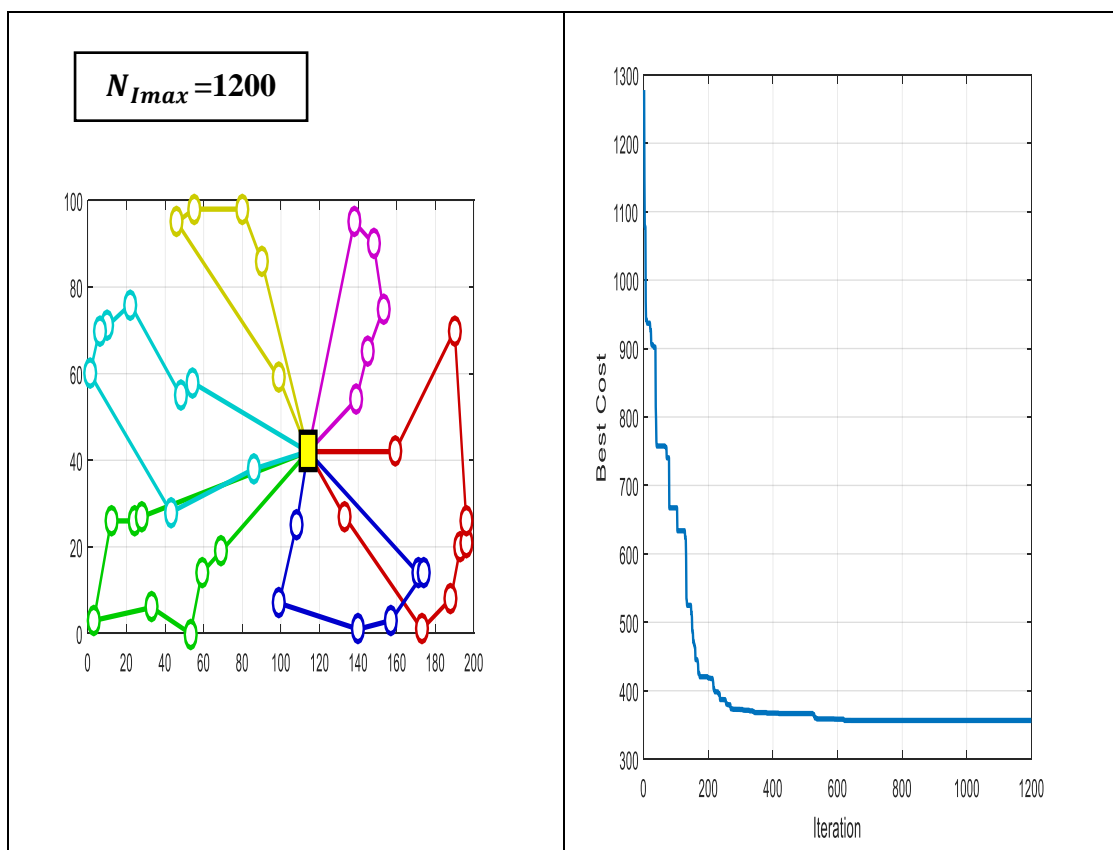
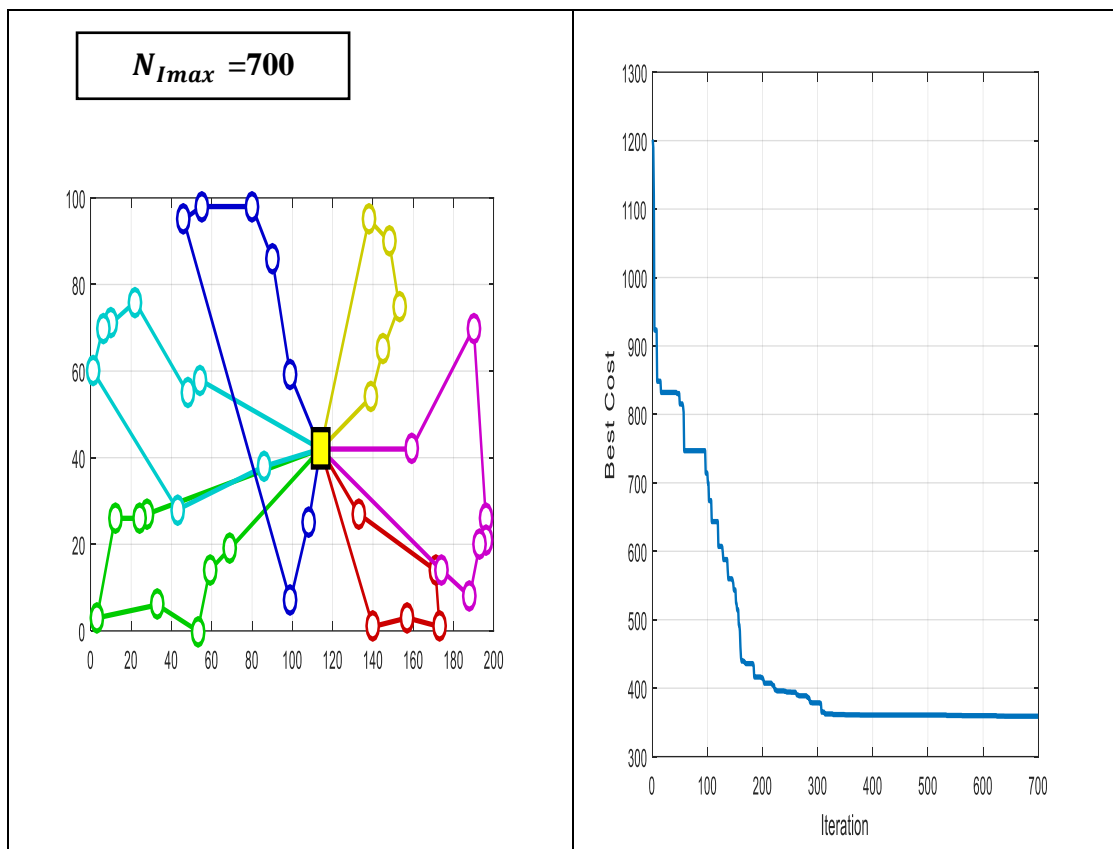


Figure 3.6 : Fenêtre graphique de l'exécution.

Nombre des itérations $N_{I_{max}}$	Solution optimal
50	802.15
250	385.69
500	361.00
700	359.02
1200	356.62

Tableau 3.4 : Expérimentation pour mesurer l'influence de nombre des itérations.

9. Discussion et résultat

Les trois graphes suivants présentent juste ou dessus, sont bien visible est bien expliquées dans ces trois tableaux et on remarque que :

- **Dans le test 01 présente :** une amélioration légèrement jusqu'à $n = 8$ a $n = 20$; la courbe stabilise dans la solution optimale.
- **Dans le test 02 présente :** décroissance accentues vers une solution qui améliorée et contenue légèrement s'améliore de $n = 400$ jusqu'à $n = 1000$ qui restera stable.
- **Dans le test 03 :** on constate une légère amélioration en $n = 80$ Jusqu'à $n = 180$.
En $n = 700$ une brusque chute de la courbe améliorer nettement la solution optimale.

10. Conclusion

Nous avons mené une étude sur l'influence des paramètres sur la performance de l'algorithme génétique à trouver des solutions optimales au problème considéré qui est le **CVRP**. On n'est arrivée a la conclusion que la taille de population et le nombre des itération et génération ont une influence sensible sur la performance de l'algorithme néanmoins leurs valeurs doivent être bien étudiées pour sélectionner les meilleures qui peuvent apporter plus de performance a l'algorithme génétique.

Conclusion Générale

Parmi les problèmes des transports on cite le problème de tournée de véhicules ; ce problème est intrigant de la recherche opérationnelle qui a été étudié pour plusieurs années.

Dans le cadre de ce mémoire nous sommes intéressés par une variante importante de ces problèmes de tournée de véhicule avec une contrainte de capacité (**CVRP**) consiste à effectuer chaque client à une tournée effectuée par seul véhicule de capacité finie.

Ce véhicule commence et termine sa tournée au dépôt.

Nous avons proposé notre approche pour résoudre le (**CVRP**) principalement basé sur les **algorithmes génétiques**.

Notre algorithme génétique développe et approuve son efficacité en :

- 1- La minimisation de la route de véhicule (la distance).
- 2- Donne la meilleure solution (solution optimale).
- 3- Relativité de la qualité de solution selon le degré de précision demandée

Respectivement, cette étude peut être étendue à d'autres types de problèmes logistiques à travers les différents types d'algorithmes méta-heuristique.

Bibliographie

- [1] **Rego and C. Roucairol.** (1994) le problème de tournées de véhicules : Etude de résolution approchée. Technical Report 2197, INRIA-Institut National de recherche en Informatique et en Automatique, 1994.
- [2] **Sahbi Ben Ismail** (Télécom Bretagne, Labsticc) François Legros (Deev Intraction). Gilles coppin (télécom Bertagne , labsticc) Synthèse du problème de routage de véhicules RR-2011-03-lussi.
- [3] **David Monnaiux et Bruno Salvy** le problème du voyageur de commerce, Ecole polytechnique ___X2013___ INF412 Fondements de L'informatique.
- [4] **Douglas B. west,** Introduction to graphe theory ,2001, Prentice_ Hall, Memoire Master universite mosta-dz, 2016.
- [5] **Dhaenens, M.L.Espinouse , and Bernard Penz .**Problèmes combinatoires classiques.In Recherche opérationnelle et réseaux : méthodes d'analyse spatiale Hermès Science Publications , 2002.
- [6] **Dantzig, R. Fulkerson ; and S. Johnson.** Solution of a large –scale traveling – salesman problem. Journal of the Operations Research Society of America, 2(4): pp.393-410,1954.
- [7] **MONTOYA JA,** Electric Vehicul Routing Problem: models and solution approaches; Thèse Doctorat en informatique; university d'angrs 2016.
- [8] **CADET David Joseph,** Optimisation des flux : application aux problèmes de distribution en nutrition animale, thèse Doctorat en optimisation des systèmes, Université de Technologie de Troyes, 2013.
- [9] **MARSI Loubna,** Etude de cas d'un problème de tournées des véhicules à la société SNTL, Mémoire licence en sciences et techniques ; Université Sidi Mohamed Ben Abdellah. Faculté des Sciences et Technique,2015.

- [10] **HAMANI Amir**, Development of a tool for optimization vehicle routing problem, Mémoire Master en informatique décisionnelle et optimisation, Université m'sila, Algérie ,2018.
- [11] **MICHALLET Julien**, problème de tournées de véhicules périodiques Avec contraintes de sécurité ou de qualité de service, thèse Doctorat en optimisation, Troyes, 2013.
- [12] **AkIL, Meriem**, Problème de tournées de véhicules avec contraintes et fenêtre de temps. Mémoire Magister en informatique, UMMTO, 2013.
- [13] **Sahbi Ben Ismail** (Télécom Bretagne, Labsticc) François Legros (Deev Intraction) Gilles coppin (télécom Bretagne , labsticc) Synthèse du problème de routage de véhicules RR-2011-03-lussi.
- [14] **Sahbi Ben Ismail** (Télécom Bretagne, Labsticc) François Legros (Deev Intraction) Gilles coppin (télécom Bertagne , labsticc) Synthèse du problème de routage de véhicules RR-2011-03-lussi.
- [15] **MONTOYA JA**, Electric Vehicul Routing Problem: models and solution approaches; Thes Doctorat en informatique ; university d'angrs 2016 .
- [16] **Mazin Abed Mohammed, Mohd Sharifuddin Ahmad, Salma A. Mostafa**
Using Genetic Algorithm in Implementing Capacitated Vehicle Routing Problem. International Conference on computer and information science (ICCIS) ,2012.
- [17] **Derkaoui Orkia**, heuristique et méta heuristique, université Dr Moulay Tahar de saida, 26/02/2014.
- [18] HAL Id : hal -01542507 <https://hal.archives-ouvertes.fr/hal-01542507>.
- [19] <http://www.cirrusimage.com/beetle-firefly-photuris-lucicresens.htm>.
- [20] <http://www.sciencedaily.com/releases/2011/04/110405093656.htm>.
- [21] **Eisner, Thomas; Wiemer, David ;Haynes,Leroy; Meinwald ,Jerrold (1978)**, Lucibufagins :Defensive steroids from the fireflies **photinus ignites and**

P.marginellus (Coleoptera :Lampyridae), The National Academy of Sciences of the USA

[22] **Stanger-Hall; K.F.; Lloyd, J.E.; Hillis, D.M. (2007)**, “Phylogeny of North American fireflies (Coleoptera: Lamyridae): implications for the evolution of light signals “, *Molecular Phylogenetics and evolution* 45 (1): 33a 49, doi: 10.1016/j.ympev.2007.05.013, PMID 17644427

[23] **Xin-She Yang**. Firefly algorithm for multimodal optimization.*Stochastic Algorithms : Foundation and Application* ,5th,2009 .

[24] **yang, X.S.**, “Firefly Algorithm, Stochastic Test Functions and Desing Optimization “, *Int.J.Bio-Inspired Computation* , Vol .2,No .2,2010,pp.78—84

[25] **X.-S. Yang**, “Firefly Algorithm, Levy Flights and Global Optimization “, *Research and Development in Intelligent Systems XXVI* “(Eds M. Bramer, R.Ellis , Petridis), Springer London , 2010, pp. 209-218.

[26] **D. GAERTNER**, *Natural Algorithms for Optimization Problem*, Outsourcing Report, 16 janvier 2004

[27] **C. GAGNE, M. GRAVEL ET W. L. PRICE**, *Optimization par Colonie de Fourmis pour un problème d’ordonnancement Industriel avec un temps de Réglage Dépendant de la séquence*, 3^{ème} Conférence francophone de Modélisation et Simulation Conception, Analyse et Gestion de Système Industriels, MOSIM01, Troyes, France, 2001.

[28] [http : //www.kimonte.com/algorithm_de_colonie_de_fourmis.php](http://www.kimonte.com/algorithm_de_colonie_de_fourmis.php)

[29] <https://dataanalyticspost.com/Lexique/particle-swarm-optimization-psy/#:~:text=Comment%20fonctionne%20cet%20algorithme%20%3F,potentielle%20au%20probl%C3%A8me%20d'optimisation>.

[30] **J.Kennedy and R.C.Eberhat**. “Particle Swarm Optimization “. In: *Proceedings of the IEEE International Conference on Neural Networks IV*, pp.1942-1948, perth, Australia, November 1995.

[31] **GEORGE NEMHAUSER ET LAURENCE WOLSEY**, *Integre and Combinatorial Optimisation* .Weley Interscience,1988,

- [32] **VINCENT BARICHARD**, Approches hybrides pour les problèmes multi objectifs, Thèse de doctorat, Ecole Doctorale d'Angers, 24 Novembre 2003
- [33] **GOLDBERG**, D.E Genetic Algorithms in Search, Optimization and Machine Learning. Addison – Wesley: Reading, MA, 1989.
- [34] **C. HAMZACEBI ET F. KUTAY**, A heuristic approach for finding the global minimum: Adaptive random search technique. Applied Mathematics and Computation, 173 :1323-1333,2006
- [35] **BENKADOUR H, Aribi R**, Meta-heuristiques parallèles pour la résolution des problèmes difficiles, Mémoire Master en informatique, Université Kasdi Merabah Ouargla, 2013.
- [36] **HAJ-RACHID, Mais et al**, Différentes opérateurs évolutionnaires de permutation : sélections, croisements et mutation, Rapport de recherche, Laboratoire D'informatique, Université de Franche –Comte ; 2010.
- [37] **DOUIRI Mohamed**, Méthodes de Résolution Exacte Heuristiques et Meta heuristiques, Mémoire Master en informatique. Université d'Oran, 2008.
- [38] http://www.yorku.ca/nmw/facs1939f13/javascript_all/js_scriptingVSprogramming.html.

ملخص

يحتل التحسين التوافقي مكانة مهمة للغاية في بحوث العمليات و الرياضيات و علوم الكمبيوتر .
ان مشكلة توجيه السيارة مع احترام القدرة المحددة (CVRP) هي مشكلة تحسين توافق مع الإشكالية التالية :
☞ ماهي المجموعة المثلى من المسارات لأسطول من المركبات من أجل التسليم الى مجموعة معينة من العملاء مع احترام القدرة المطلوبة للمركبات و العودة الى المستودع عند الانتهاء
هدفنا في هذا العمل هو تقليل المسافة التي تقطعها المركبات مع احترام القدرة المحددة بتطبيق الخوارزميات الجينية باستعمال لغة البرمجة "مطلب" .
الكلمة المفتاحية مشكلة توجيه السيارة مع احترام القدرة ، التحسين ، الخوارزميات ، الجينية ،

Matlab

Résumé

L'optimisation combinatoire occupe une place très importante dans la recherche opérationnelle, en mathématique discrètes et en informatique. Le problème de tournée de véhicule avec contrainte de capacité (CVRP) est un problème avec la problématique : Quel est l'ensemble optimal de routes traverser afin de livrer a un ensemble donné de clients en respectant les capacités des véhicules et en retournant ou dépôt une fois termine ? L'objectif de ce travail est de minimiser le cout de distance, en appliquant les algorithmes génétiques comme heuristique et en utilisant le langage de programmation ' MATLAB ' pour le codage

Mots-clés : Problème de tournée de véhicule avec contrainte de capacité CVRP, optimisation, algorithmes génétiques, MATLAB.

Abstract

Combinatorial optimization occupies a very important place in operations research ,discrete mathematics and computer science .The capacitated problem (CVRP) is combinatorial optimization that answers : what is the optimal set of routes for a fleet of vehicles to traverse in order to deliver to given set of customers with respect delivery capacitated and returning to the warehouse once completed ? Our goal in this work is to minimize the cost of distance traveled and the time spend , by applying genetic algorithms as heuristics and by using the ' MATLAB ' programming language for coding .

Keywords : Capacitated vehicle routing problem (CVRP) , optimization , genetic algorithms , MATLAB .

