MINISTERE DE L'ENSEIGNEMENT SUPERIEUR
ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE MOHAMED EL BACHIR EL-IBRAHIMI

BORDJ BOU ARRERIDJ

Faculté Des Mathématiques et d'informatique

Département des Mathématiques

MEMOIRE DE FIN D'ETUDES

Réalisé en vue de l'obtention du diplôme Master en Mathématiques

Spécialité : Méthodes et Outils pour la Recherche Opérationnelle

***Thème* :**

Le problème de routage de véhicule avec capacité par l'algorithme de colonies de fourmis CVRP

Présenté par :
BENNOUR Khawla
BOUBAAYA Zineb

Encadré par :
Dr MAZA Sofiane

Membre du jury:

- **ZOUACHE Djaafer** — **Président**
- **MAACH Salah** — **Examinateur -1-**
- **FILLALI Ferhat** — **Examinateur -2-**

Année universitaire : 2020-2021

Le Quotidien britannique « The Times » a publié un article le 6 février 2009, sous le titre (Hills are alive with the sound of ants - talking to each other) pour les fourmilières vivantes avec des voix, en parlant avec les uns des autres). L'article parle des nouvelles découvertes scientifiques sur le langage de communication et la conversation dans le royaume de fourmis. L'article mentionne que les découvertes récentes ont montré que la langue de communication chez les fourmis est sophistiquée et avancée significativement plus qu'on ne le pensait auparavant. En insérant des microphones et des enceintes miniatures dans les nids, les chercheurs ont découvert que la reine pouvait donner des instructions à ses ouvrières. Ils ont enregistré ses "discours", et ont également découvert que d'autres insectes imitent ces messages, pour faire des fourmis leurs esclaves.

Remerciements

Je remercie Dieu de m'avoir donné le courage et la volonté qui m'ont permis d'aborder ce modeste travail.

Je tiens à exprimer en tout premier lieu ma profonde gratitude à mon encadreur, Monsieur" MAZA Soufaine", de m'avoir proposé ce sujet, et de l'attention qu'il a portée à mon travail. J'ai découvert grâce à lui le monde de la recherche opérationnelle, et plus précisément celui du problème de tournée des véhicules, je le remercie vivement pour ces précieux conseils, pour sa disponibilité et surtout pour sa patience.

Je tiens à remercier tous les membres du jury qui m'ont fait l'honneur de bien vouloir juger ce travail et de l'enrichir par leurs remarques et critiques.

Mes remerciements vont aussi à l'ensemble des enseignants et du personnel du département de mathématique et informatique.

Enfin, je remercie du fond du cur ma famille et mes amies pour leur amour, leur tendresse et leur soutien inconditionnels.

Merci à tous

Dédicaces

Je dédie ce travail à ...

A ma mère pour son amour, ses encouragements et ses sacrifices.
A mon chère père pour son soutien , son affection et la confiance qu'il m'a accordé.

A tout les membres de ma famille....

En particulier, mon frère et ami Oussama Bennour pour m'avoir aidé pendant mon parcours d'études .

Ma mes chères soeurs Marwa , les frères Habib et les jumeaux Hassan et Hussein.

BENNOUR Khawla

Dédicaces

Je dédie ce travail à ...

À la fontaine qui ne se pas de donner a celle qui a tissé mon bonheur avec les fils tissés de son coeur A ma chére mère.

À celui qui s'efforce et lutte pour jouir du confort et du contentement qui n'épargne rien pour me pousser sur le chemin du succès qui est apprends moi à gravir les échelons de la vie avec sagesse et clarté A mon cher père.

À mes chères soeurs : Rayan , Amina et mes frères : Ayob , Ismail , Tahar ,

À mon cher grand père et ma chère grand-mère, À toute ma famille ,

mes oncles, mes tantes et mes cousins,

À toutes mes amies et camarades particulièrement ma chère binôme : "BENNOUR Khawla "

À toutes celles et ceux qui m'ont connu, soutenus et aimés.

BOUBAAYA Zineb

Table des matières :

Introduction générale	1
Chapitre 01 : le problème de routage de véhicule avec capacité CVRP	
1 Introduction……………………………………..…………………..	4
2 Formulation……………………………………...…………….…...	4
3 Détaille des équations……………………………………..…….	6
4 Variante du VRP……………………...………………………..….	8
5 Méthode de résolution du VRP…………………………………….	12
6 Exemple pour VRP…………….…………………………..…….	14
Conclusion……………………….…………………………………….	15
Chapitre 02 : les algorithmes métaheuristiques	
1 Introduction…………………………………………………………….	17
2 Présentation des métaheuristiques………………………………..	17
3 Caractéristique d'une métaheuristique…………………………...	18
3.1 L'aléatoire…………………………………………….....……….	18
3.2 Tolérance à la détérioration……………………………...……….	18
3.3 La mémoire………………………………………….....………….	18
3.4 La synchronisation entre la diversification et l'intensification.	19
4 Fonctionnement générale des métaheuristiques…………………...	19
6 Les algorithmes métaheuristiques………………………………..	22
6.1 Algorithmes génétiques…………………………………..…….	22
6.2 Les recuit simulé………………………………………………..	24
6.3 Les méthode recherche tabou……………………………………..	27
6.4 Les essaims particulaires (PSO)… ……………………………….	29
6.5 Les colonies de fourmis(ACO)……… ………………………...	32
7 Schéma générale…………………………………………………	35
8 Conclusion…………...	36
Chapitre 03 : Application de l'algorithme de colonie de fourmis sur Le CVRP	
1 Introduction…………………………………………………………	38
2 Historique…………………………………………………………….	39
3 Méthode de K- means………………………………………………	40
4 Les algorithmes de colonie de fourmis………………………….…	44
4.1 Les fourmis réelles………………………………………………..	44
4.1.1 L'intelligence collective des fourmis………………………….	45
4.1.2 La communication…………………………………………….	46
4.2 Les fourmis artificielles…………………………..……………..	47

4.2.1 Les algorithmes de colonie de fourmis…………………………	48
4.2.2 Optimisation naturelle………………………………………..	48
5 Données et notation………………………………………….……...	50
6 Les problèmes de routage de véhicule avec capacité…………….....	53
7 Algorithme proposé de colonie de fourmis…………………………	56
Conclusion ……………………………………………………………...	61
Tests et résultats..	62
Conclusion générale	73

TABLE DES FIGURES	Page
1.1: Classification des variantes du VRP…...………………………	12
1.2 : Classification des résoudre des méthodes de VRP……..………	13
2.1 : Schéma d'algorithme génétique………………..…………………	23
2.2 : Organigramme de l'algorithme de recuit simule………..………	26
2.3 : Organigramme de l'algorithme de tabou simple…………………	29
2.4 : Organigramme de l'algorithme de PSO…………………………..	31
2.5 : Organigramme de l'ACO-OPE………………………………	34
2.6 : Schéma générale de les méthodes d'optimisation………………..	35
3.1 : Schéma générale………………………………………………	39
3.2 : Exemple de déplacement de centre des classe. algorithme de kmeans…………………………………………………………	44
3.3 : Les fourmis réelles…………………………………………….	45
3.4 : La sélection par roulette ………………………………………	58
3.5 : Une fourmis suivent une piste de phéromone ……………………	48
3.6 : Expérience de sélection des branches les plus courtes par C.F…..	49
3.7 : Exemple de problème de tournées de véhicule VRP…………….....	53
3.8: Schéma détaillée……………………………………………….	60
3.9: Test pour véhicule 01 par ACO...	64
3.10: Test pour véhicule 02 par ACO..	65
3.11: Test pour véhicule 03 par ACO..	66
3.12: Test pour véhicule 04 par ACO..	67
3.13: Test pour véhicule 01 par ACO..	68
3.14: Test pour véhicule 02 par ACO...	69
3.15: Test pour véhicule 03 par ACO..	70
3.16: Test pour véhicule 01 par ACO..	71
3.17: Test pour véhicule 02 par ACO...	72

Chapitre 01

Le problème de routage de véhicule avec capacité

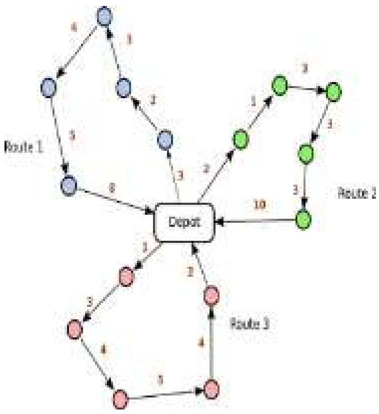
CVRP

Chapitre 02

Les algorithmes métaheuristiques

Chapitre 03

Application de l'algorithme de colonie de fourmis sur le CVRP



INTRODUCTION GENERALE

La recherche opérationnelle (RO) est une discipline scientifique récente datant tout au plus de la deuxième guerre mondiale. Elle est l'ensemble des méthodes et techniques rationnelles d'analyse scientifique (math et info) et de synthèses des phénomènes d'organisation qui traite de la maximisation d'un profit, d'une performance, d'un rendement ou bien de la minimisation d'un coût. Le champ d'application de la RO s'est élargi à des domaines comme l'économie, la finance, le marketing et la planification d'entreprise. La RO est avant tout un outil d'aide à la décision, sa vocation est donc de construire des modèles pour des problèmes généraux d'aide à la décision, particulièrement les problèmes d'optimisation, et de proposer de méthodes de résolution efficace de ces modèles.

Les problèmes de transport créent une partie essentielle des travaux de ce domaine, son problème de base est le problème de tournées probablement celui le plus étudié par plusieurs chercheurs.

Le problème classique de l'élaboration des tournées de véhicules (VRP) consiste à construire des routes avec un coût minimum pour que les véhicules puissent visiter exactement une fois chaque client géographiquement distribué. Le VRP est un sous problème important dans le domaine des systèmes de distribution et beaucoup d'efforts ont été consacrés en recherche sur divers aspects du VRP.

Il s'agit de déterminer les tournées d'une flotte de véhicules afin de livrer une liste de clients, ou de réaliser des tournées d'interventions (maintenance, réparation, contrôles) ou de visites (visites médicales, commerciales, etc.). L'objectif des problèmes de routage véhicule avec capacité (CVRP) est de minimiser le cout total, c-à-d la somme des distances et en conséquence 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 avec une dégradation maximale de cette dernière. Ce problème est une extension classique du problème du voyageur de commerce. En lui rajoutant des nouvelles contraintes comme la capacité et le nombre de véhicule où on cherche le plus court chemin avec une dégradation maximale de la charge de véhicule d’une à ville à une autre. Pour le résoudre plusieurs méthodes sont disponibles. Les méthodes méthaheuristiques comme Algorithme de colonies de fourmis prennent une grande place vis-à-vis des méthodes existantes. L’objectif de ce travail est de proposer un algorithme basé sur la colonie de fourmis.[39]

Les algorithmes métaheuristiques permettent de s'approcher d'une ou de plusieurs solutions

à des problèmes dits "difficiles" qui s'apparentent à des problèmes d'optimisations. Le principe d'une métaheuristiques est de minimiser ou de maximiser une fonction objectif. L’avantage des métaheuristiques est de trouver un minimum global à un problème de minimisation et de ne pas rester bloqué sur un minimum local .

Ce mémoire contient 3 chapitres organisés de la manière suivante :

Dans le premier chapitre, nous présenterons une vue globale de problème de routage de véhicule VRP .

Dans le deuxième chapitre, nous définissons d’abord les algorithmes métaheuristiques et nous mentionnons les principes des métaheuristiques les plus répondus (recuit simulé, recherche des tabou , les algorithmes génétiques GA, les colonies de fourmis ACO et les essaims particulaires PSO) en citant leurs définitions, leurs principes et leurs algorithmes de bases.

Dans le troisième chapitre, nous allons proposer un algorithme basé sur celui de colonie de fourmis pour résoudre le problème CVRP et nous allons faire une présentation générale des problèmes de routage véhicule avec capacité (CVRP). Dans l’algorithme de colonies de fourmis, nous avons introduit la notion de roulette qui permet de choisir le meilleur chemin de visite potentiel , avec une implémentation numérique d’application combiner d’algorithme de colonie de fourmis et l’algorithme de kmeans pour le regroupement des villes appliquées sur quelques exemples de CVRP.

Nous terminons notre travail par une conclusion générale, résumant ce qu’on a fait et les résultats obtenus.

Chapitre 01

Le problème de routage de véhicule Avec Capacité (CVRP)

1.1 Introduction

Une définition générale du VRP consiste à chercher un itinéraire optimal pour une flotte de véhicules, basée en un ou plusieurs dépôts, afin de desservir un ensemble de clients ayant des commandes connues, et dispersés géographiquement. Une tournée désigne l'ensemble des clients visités par un véhicule qui part et revient au même dépôt.

Le VRP fait partie des problèmes d'optimisation combinatoire et de recherche opérationnelle. La première fois, en 1959, que ce problème a été étudié par sous le nom de « The Truck Dispatching Problème » et a depuis fait l'objet de nombreux travaux qui ont donné de nombreuses variantes et différentes méthodes de résolution.

Le VRP appartient à la catégorie NP-difficile [40]et dans sa version de base, il modélise un problème de transport avec une contrainte de capacité (CVRP)[41] qui consiste à livrer des marchandises auprès des clients à l'aide d'une flotte de véhicules à capacité limitée. La résolution consiste à déterminer un ensemble de tournées qui minimise au mieux des objectifs comme le coût total, la distance totale parcourue, la somme des retards des clients [42].

La réduction des coûts de transport a intéressé les développeurs de software qui ont mis, sur le marché, des logiciels de plus en plus performants pour optimiser les tournées de véhicules. Cette question, mieux connue aujourd'hui, sous le nom de Véhicule Routin Problème (VRP), a également fait l'objet de nombreux travaux académiques, depuis plus de 50 ans, et reste d'actualité principalement pour la recherche des méthodes de résolution qui profitent de l'arrivée des ordinateurs plus puissants grâce aux progrès technologiques. Ainsi, les solutions issues de la recherche sont de plus en plus pertinentes, applicables à la réalité des entreprises et donc encourageantes. Ces solutions sont, dans la perspective, de fournir aux entreprises des outils d'aide à la décision qui peuvent leur permettre de réaliser des économies allant jusqu'à 20% .

1.2 Formulation

Rappelons que le problème VRP consiste à affecter chaque client à une tournée (i.e. route) effectuée par un seul véhicule et à trouver un ordre de visites des clients pour chaque véhicule de façon à satisfaire les contraintes de capacité des véhicules, et les quantités de produit demandé par chaque client, dans le cas d'un problème de livraison. Il convient de noter que la formulation proposée correspond également au VRP dans le cas d'un problème de collecte ou de ramassage.

L'objectif dans ce problème est de trouver l'ensemble des tournées qui minimisent la distance totale parcourue pour un nombre minimal de véhicules partant d'un dépôt et y retournant. Ce problème est une extension du problème classique du voyageur de commerce (TSP) .

Nous allons formuler le VRP selon un modèle de recherche opérationnelle dans la forme utilisée par Le Bouthillier. En fait, ces travaux ont formulé le problème VRPTW (VRP with Time Windows), dans cette section nous allons présenter la formulation du problème VRP classique sans contraintes temporelles et dans la section nous allons continuer à formuler le problème VRPTW en ajoutant ces contraintes.

Un graphe $G = (N;A)$ représente notre problème où :

– N représente les positions des clients et du dépôt,

– A représente les arcs entre deux clients $i; j \in N$.

Plus spécifiquement, nous avons un ensemble $C = \{1 \dots\dots\dots\dots n_c\}$ de clients qui doivent obtenir une livraison de marchandise provenant du dépôt. L'ensemble des positions de ces clients ou nœuds est défini comme l'ensemble $N = C \cup \{0, n_c + 1\}$ où 0 et $nc + 1$ représentent le dépôt (aller et retour). Une demande positive de produit di est associée à chaque client i appartenant à C.

Une flotte de véhicules $V = \{1, \dots\dots.., n_v\}$ est disponible au dépôt et chaque véhicule possède la même capacité (flotte homogène) Q telle que $Q \geq max\ di; \forall\ i \in N$. Pour tous clients i et $j, \forall i, j \in N$, nous connaissons le coût $ci;j$ 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 :

$$x_{i,j}^{v} = \begin{cases} 1\ si\ le\ véhicule\ v \in V\ visite\ le\ client\ j\ aprés\ le\ client\ i, \\ 0\ sinon \end{cases}$$

En définissant yi 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.

$$Min \sum_{v \in V} \sum_{i \in N} \sum_{j \in N} c_i x_{i,j}^{v} \qquad (1.1)$$

Avec les contraintes :

$$\sum_{v\in V}\sum_{j\in N} x_{i,j}^{v} = 1, \forall\, i \in C \quad (1.2)$$

$$\sum_{j\in N} x_{i,j}^{v} - \sum_{j\in N} x_{j,i}^{v} = 0, \forall i \in C, v \in V \quad (1.3)$$

$$\sum_{j\in N} x_{0,j}^{v} = 1; \forall v \in V \quad (1.4)$$

$$\sum_{j\in N} x_{j,n+1}^{v} = 1; \forall v \in V \quad (1.5)$$

$$x_{i,j}^{v} = 1 \rightarrow y_i - d_j = y_j\ ; \forall i,j \in N, v \in V \quad (1.6)$$

$$y_0 = Q, 0 \leq y_i \forall i \in C \quad (1.7)$$

$$x_{i,j}^{v} \in \{0,1\}, \forall i,j \in N, v \in V \quad (1.8)$$

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

$$coût(X) = \sum_{v\in V}\sum_{i\in N}\sum_{j\in N} c_{i,j}\, x_{i,j}^{v} \quad (1.9)$$

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

$$Nb\ véhicules(X) = \sum_{v\in V}\sum_{j\in C} x_{0,j}^{v} \quad (1.10)$$

1.2.1 Détail des équations

La fonction objectif (équation 1.1) représente le nombre de véhicules utilisés pour le routes et la somme des coûts de parcours.

La formulation du problème nécessite de satisfaire certaines contraintes :

– L'équation 1.2 assure qu'on part une et une seule fois de chaque client, avec un seul véhicule.

– L'équation 1.3 assure que le véhicule qui arrive chez un client est le même que celui qui part de ce client.

– L'équation 1.4 assure que chaque véhicule ne sort qu'une seule fois du dépôt.

– L'équation 1.5 assure le retour unique au dépôt pour chaque véhicule (ou tournée).

– Les équations 1.6 et 1.8 définissent les contraintes de capacité et d'intégrité.

– Les équations 1.9 et 1.10 sont des fonctions de mesure qui permettent respectivement de quantifier la solution selon la distance totale parcourue, ainsi que le nombre de véhicules utilisés. [6]

1.3 Paramètres

Outre la version basique du Capacité VRP, le problème de routage de véhicules a plusieurs autres variantes plus ou moins étudiées dans la littérature.

En effet, la définition la plus générale du VRP est la suivante : il s'agit de la conception de routes optimales par une flotte de véhicules, basée en un ou plusieurs dépôts, pour desservir un ensemble de clients (ou villes) dispersés géographiquement et ayant des demandes connues.

Cette définition généraliste met en évidence l'ensemble de paramètres qui caractérisent une variante du VRP : le réseau de transport, la clientèle et la flotte de véhicules.

Des contraintes auxiliaires peuvent éventuellement s'ajouter à ces trois paramètres principaux. Un dernier paramètre réside dans la fonction objectif à optimiser. Le réseau :

Le réseau routier peut être symétrique ou asymétrique ; en conséquence, le graphe associé G = (V; E) sera orienté ou non et les liaisons entre les sommets seront des arcs ou des arêtes.

Les tournées peuvent partir d'un seul dépôt ou de plusieurs dépôts.

La clientèle :

La principale caractéristique de la clientèle est sa demande en marchandise.

La livraison de celle-ci peut être contrainte à s'effectuer au cours de périodes de temps spécifiées, appelées fenêtres temporelles (time Windows). Ces contraintes peuvent être dures ou souples. Dans le cas de contraintes dures, une arrivée avant la fenêtre temporelle impose une attente, et les retards sont interdits ; alors que les contraintes souples peuvent être violées et induisent ainsi des pénalités.[2]

Les clients peuvent être livrés en marchandises mais peuvent aussi en remettre aux véhicules. On parle alors de tournées de livraison/ramassage ou de service mixte. Les temps de livraison et de ramassage peuvent être non négligeables et sont ainsi pris en compte dans le calcul des durées de tournées.

Finalement, l'accès à un client peut être limité à un sous-ensemble de véhicules seulement.

La flotte de véhicules :

Le nombre de véhicules disponibles peut être fixe ou non. Notons que dans le cas d'un seul véhicule, le problème de tournées reste tout de même différent d'un problème de voyageur de commerce car la livraison des différents clients peut s'effectuer en plusieurs tournées.

Un véhicule peut être associé à un dépôt particulier ou pas. Les véhicules peuvent avoir une capacité maximale en termes de marchandises transportées (volume, poids etc.). Dans le cas d'une flotte hétérogène, cette capacité peut différer selon le type de véhicules.

La fonction objectif :

Les objectifs les plus communs sont soit la minimisation du nombre de véhicules utilisés soit la minimisation de la distance totale parcourue par les véhicules. D'autres objectifs peuvent être considérés :

_ la minimisation de la durée totale des tournées

_ la minimisation du coût total des tournées (en prenant en compte les coûts des véhicules, des chauffeurs etc.)

_ la minimisation des pénalités liées aux violations des contraintes, notamment dans le cas de fenêtres temporelles

_ la maximisation des gains engendrés par les tournée [4], les objectifs de minimisation du nombre de véhicules et de la distance (ou durée) totale des tournées sont conflictuels : la diminution du nombre de véhicules engendre le plus souvent une augmentation de la distance totale parcourue.[1] , [4]

Notons que dans des approches multi-objectifs, une somme pondérée de ces objectifs peut être considérée .[3]

1.4.Variante du VRP

Après l'introduction du problème du VRP par Dantzig et Ramser [44] en 1959, la multitude et la variété d'applications du problème a obligé sa soumission à un certain nombre de contraintes qui a, par conséquent fait apparaître plusieurs variantes dont nous citerons les plus importantes Problème de tournée de véhicules avec contrainte de capacité (CVR).

- **Problème de tournée de véhicules avec contrainte de capacité (CVRP) :**

Le problème de routage de véhicule avec capacité est la variante basique et la plus répandue du problème. Elle se distingue par la contrainte de capacité des véhicules. En effet, les véhicules d'une instance du CVRP peuvent supporter une charge de marchandise limitée et préalablement fixée.

- **Problème de tournée de véhicules avec fenêtre de temps (VRPTW)**
  Cette alternative associe à chaque client une fenêtre de temps, ce nouveau paramètre consiste en un intervalle de temps au cours duquel la demande du client doit être accomplie. La flexibilité des contraintes est néanmoins variable d'un cas à un autre et cette variation fait place à deux sous catégories :
- **VRPTW avec constraints « strictest » (Hard time window constraints):**

Les intervalles de temps peuvent être perçus comme les horaires d'ouverture et de fermeture des services de réception de la marchandise ou comme les heures de travail du personnel, et sont par conséquent complètement inviolables. En effet, les véhicules arrivant avant cette fenêtre de temps doivent impérativement attendre d'être dans l'intervalle spécifié pour être servis, quant à ceux arrivant trop tard, ils devront faire face à l'impossibilité de satisfaire leurs demandes .[45]

- **VRPTW avec contraintes « souples » (Soft time Windows contraints) :**

Cette version plus souple permet aux clients d'être servis en dehors de la fenêtre de temps contre une certaine forme de pénalité. Etant plus réaliste cette version est la plus répandue dans la littérature.

Problème de tournées de véhicules stochastique (SVRP) :

Cette classification concerne les formulations du problème dont au moins un des paramètres est aléatoire, cette définition regroupe selon Cordeau J.F et Savelsbergh[46] les cas suivants :

_Le VRP with Stochastic Customers VRPS : qui se distingue par l'introduction d'un paramètre Pi qui exprime la possibilité qu'un client i émette une demande.	**_Le VRP with Stochastic Demands (VRPSD) :** est de loin la variante stochastique la plus répandue où la quantité demandée par un client est évaluée par une variable aléatoire.	**_Le VRP with Stochastic Travel Times (VRPSTT)**

Problème de tournées de véhicules avec dépôts multiples (MDVRP) :

Le MDVRP comporte plusieurs dépôts, chacun d'eux accueillant une partie de l'ensemble des véhicules disponibles. Il est également régit par la contrainte imposant que chaque tournée doit impérativement commencer puis se terminer par le même dépôt, ainsi que celle attribuant aux clients le privilège de choisir parmi les dépôts duquel un des véhicules aura par la suite l'obligation de le visiter une et une seule fois .[47]

Problème de tournées de véhicules Split-Delivery (SDVRP ou VRPSD) :

Le SDVRP exclut la nécessité de ne visiter chaque client qu'une seule fois. De cette nouvelle flexibilité résulte la possibilité de diviser la demande de service d'un seul client sur plusieurs tournées, cette liberté modifie néanmoins l'appréhension de la contrainte de capacité et alloue donc aux clients de formuler des demandes de quantité supérieur à la capacité des véhicules.[48]

Problème de tournées de véhicules Dynamique (DVRP) :

Le routage dynamique de véhicules est le problème dont les données nécessaires à la résolution ne sont pas forcément connues dès le départ du processus d'attribution des routes, et peuvent notamment changer après qu'une partie des tournées ait été planifiée. Il découle ainsi naturellement de certains scénarios une inédite dépendance au temps, notamment les cas d'apparition de nouveaux nœuds de passage, ou la mise à jour des exigences des clients relatives aux temps de service .[49]

Problème de tournée de véhicules avec Backhauls (VRPB) :

Comporte deux types de clients, livreurs et receveurs (resp. Backhauls et Linehauls), les marchandises à livrer aux clients doivent être prises du dépôt, et celles collectées des livreurs doivent également être rendues au dépôt.

Cette classe comporte elle-même un certain nombre de sous catégories, la nécessité de cette nouvelle subdivision s'explique par la multitude de formulations du problème et de ses contraintes, toutes différentes les unes des autres, à l'image des situations réelles auxquelles elles peuvent s'appliquer. Le croquis général suivant est lui-même selon [50] une subdivision du VRPB :

-Le VRP with Clustered Backhauls (VRPCB)

Où la tâche de livraison est antérieur à celle du ramassage. Ainsi toutes les livraisons doivent être effectuées avant la première collecte.[51]

_Le VRP with Mixed Linehauls and Backhaul (VRPMB)

Autorise le traitement des services de collecte et de livraison dans une même tournée .[52]

- **Le VRP with Pick-up and Delivery (VRPPD)**

La variante notée VRPPD concerne les problèmes où les marchandises doivent être prises des clients de collecte et transportées aux clients de livraison.

_Le VRP with Simultaneous Pick-up and Delivery (VRPSPD)

S'intéresse aux clients demandant en même temps des livraisons et collectes.[53]

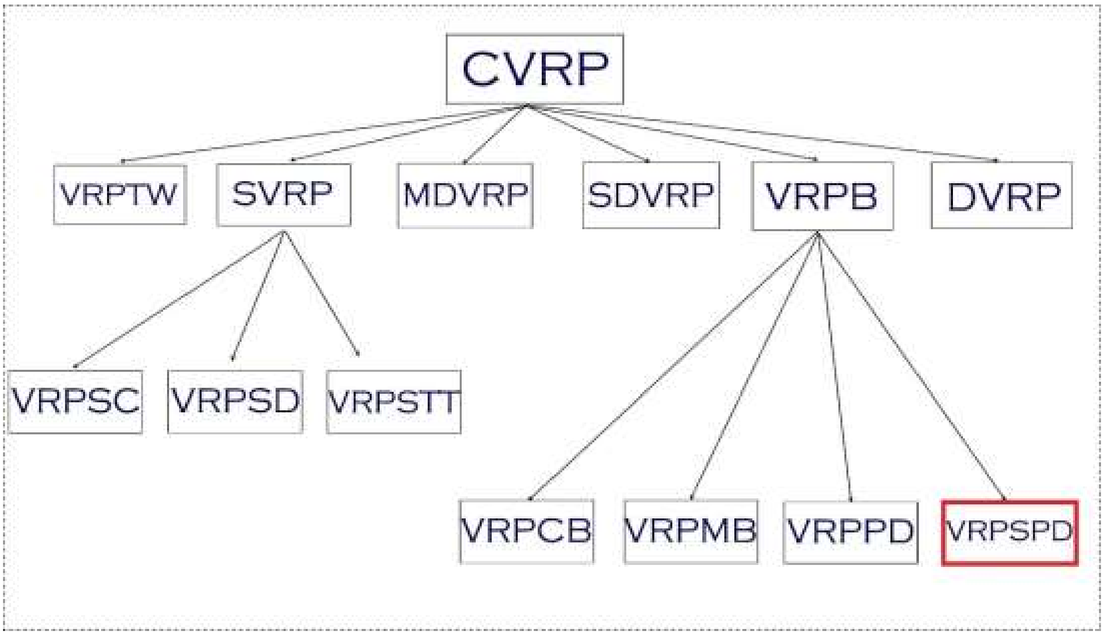


Figure1.1 : classification des variantes du VRP [30]

1.5 Méthodes de résolution du VRP

Dans cette section nous proposons une vue d'ensemble sur les méthodes de résolution du problème de routage de véhicules. notre but n'est pas de détailler fonctionnement de ces méthodes, mais plutôt d'avoir un aperçu sur leur principe de base et leurs classifications afin d'identifier les méthodes adaptées à une résolution interactive.

1.5.1Classification générale

Comme les autres problèmes d'optimisation combinatoire, le problème de routage de véhicules a été étudié et résolu par des méthodes exactes, des heuristiques spécifiques ainsi que par des métaheuristiques .Ces trois familles correspondant à la classification générale des méthodes de résolution [43] . On a les méthodes suivante :

- -Méthodes exactes
- -Méthodes approchées : Les heuristiques et Les métaheuristiques .
- -Méthodes d'amélioration.
- -Méthodes à deux phases.

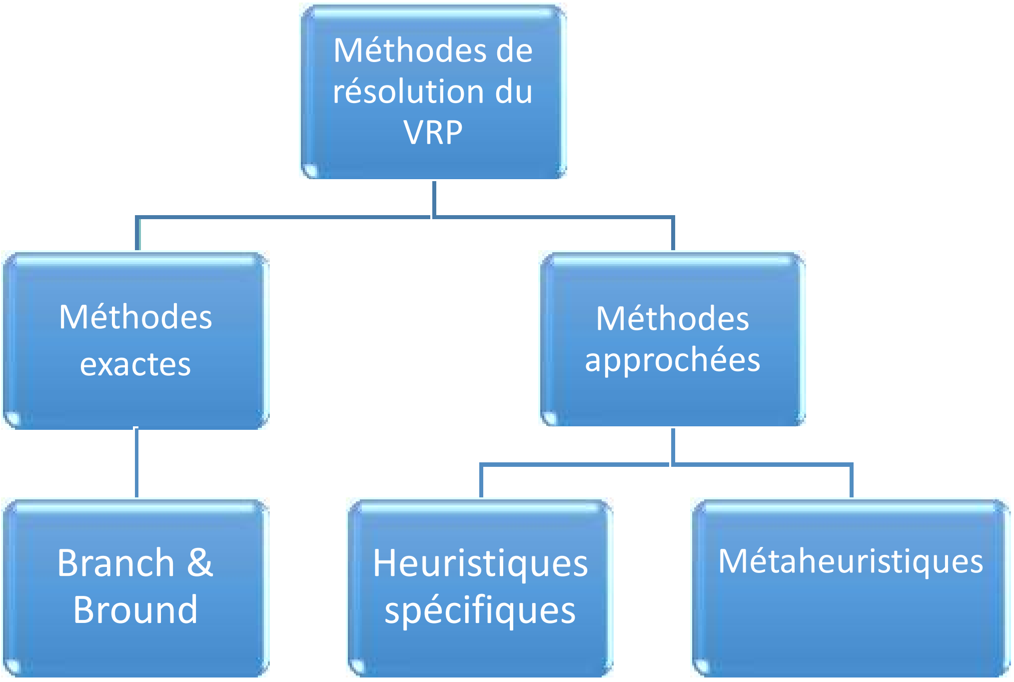
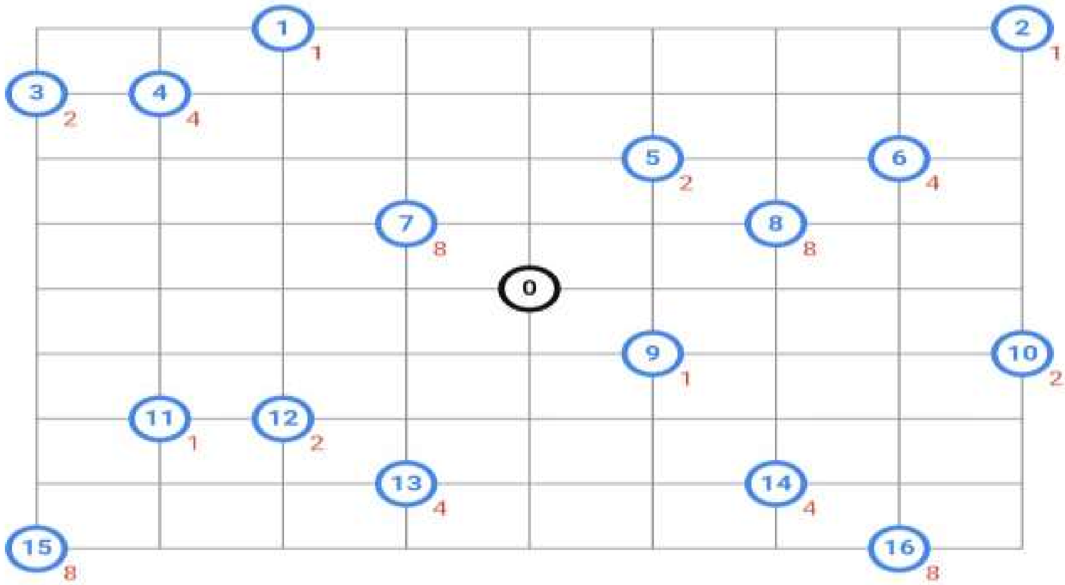


Figure1.2 : classification de résoudre des méthodes de VRP.

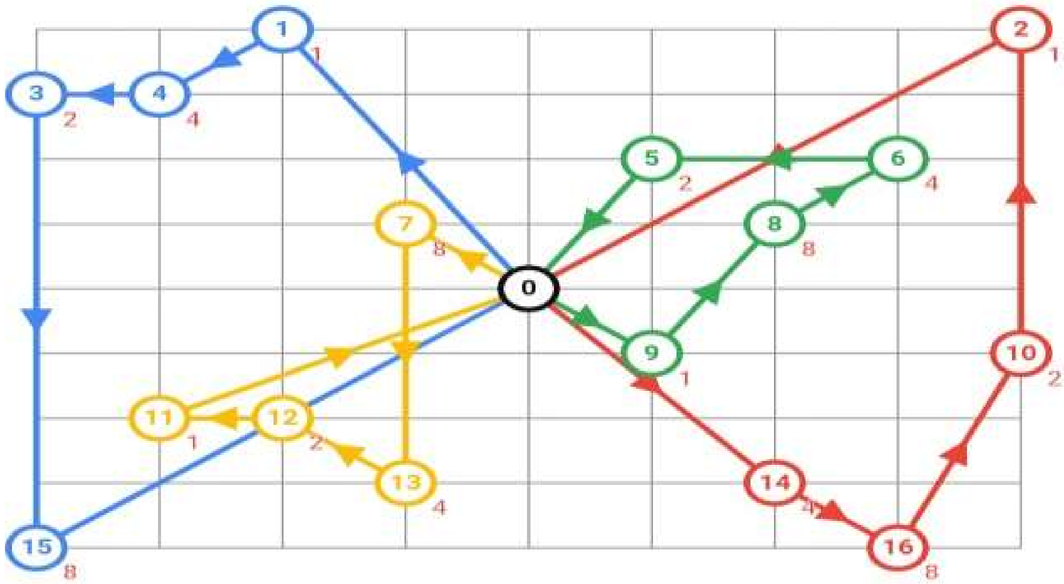
1.6 Exemple :

A chaque emplacement il y a une demande correspondant à la quantité de l'article à ramasser .de plus , chaque véhicule a une capacité maximale de 15.(nous ne spécifions pas d'unités pour les demandes ou la capacité).

La grille ci-dessous montre les lieux à visiter en bleu et le site de l'entreprise en noir .les demandes sont affichées en bas à droite de chaque emplacement . voir coordonnées d'emplacement dans la section VRP pour plus de détails sur la façon dont les emplacement sont définis. Le problème est de trouver une attribution d'itinéraire aux ayant la distance totale plus courte, et tell que la quantité totale transportée par un véhicule ne dépasse jamais sa capacité. [28]



Après avoir appliqué l'algorithme nécessaire pour résoudre ce problème, nous trouvons l'image:



Conclusion

Nous avons présenté dans ce chapitre le problème de routage de véhicule avec contrainte de capacité (CVRP), ainsi sa formulation mathématique, variantes et paramètres.

Comme nous l'avant déjà ce problème appartient à la classe NP difficiles. Donc sa résolution n'est possible que d'une manière approchée car de nos jours il n'y a aucun algorithme qui puisse nous permettre leur résolution en un temps polynomial.

Dans le chapitre suivant, nous allons présenter une description générale sur les métaheuristiques ainsi que le principe de quelques méthodes de résolution.

Chapitre 02

Les algorithmes Métaheuristiques

Introduction

L'optimisation combinatoire (OC) occupe une place très importante en recherche opérationnelle (RO), 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 qui peuvent être formulées sous la forme d'un problème d'optimisation.

On distingue généralement deux grandes familles de méta heuristiques : celles qui manipulent en parallèle toute une population de solutions (on peut citer les algorithmes génétiques, la méthode des colonies de fourmis, l'optimisation par essaim particulaire, etc.) et les autres qui se basent sur l'évolution itérative d'une solution unique (la méthode Tabou et le Recuit Simulé sont des exemples typiques de ces méthodes). Nous invitons le lecteur intéressé par plus de détails à consulter notre article [7].

2 – Présentation des métaheuristiques :

Les métaheuristiques sont des algorithmes d'optimisation de type stochastique et progressant vers un optimum par échantillonnage d'une fonction objectif dont le but est la résolution de problèmes d'optimisation difficile.

Les métaheuristiques marquent une réconciliation de domaines: en effet, celles-ci s'appliquent à toutes sortes de problèmes discrets, et elles peuvent s'adapter aussi aux problèmes continus.

Ces méthodes ont en commun, en outre, les caractéristiques suivantes :

-elles sont, au moins pour partie, stochastiques: cette approche permet de faire face à l'explosion combinatoire des possibilités;

-souvent d'origine discrète, elles ont l'avantage, décisif dans le cas continu, d'être directes, c'est-à-dire qu'elles ne recourent pas au calcul, souvent problématique, des gradients de la fonction objectif.

Elles sont inspirées par des analogies: avec la physique (recuit simulé, diffusion simulée...), avec la biologie (algorithmes évolutionnaires, recherche tabou...) ou avec l'éthologie (colonies de fourmis...).

Elles partagent aussi les mêmes inconvénients: les difficultés de réglage des paramètres de la méthode, et le temps de calcul élevé.

L'utilisation du mot métaheuristiques signifie les algorithmes regroupent en réalité plusieurs heuristiques. Cette utilisation implique qu'on ne peut pas réellement classer les métaheuristiques dans la catégorie des algorithmes d'intelligence artificielle, puisqu'ils sont principalement guides par le hasard. Cependant ils sont souvent combines a d'autre algorithmes afin d'accélérer la convergence en obligeant entre autre le métaheuristiques a ne pas prendre en compte les solutions trop (extravagantes).

Une notion à bien cerner découlant de l'utilisation d'heuristique est l'ensemble des solutions possibles ou espace global. Car bien que les métaheuristiques fonctionnent plus ou moins de façon hasardeuse cette tare est compensée par la diminution de l'espace de travail local à chaque itération [8].

3-Caractéristiques d'une métaheuristiques :

Les principales caractéristiques d'une métaheuristiques sont [15] :

3.1-L'aléatoire :

La majorité des métaheuristiques sont des algorithmes stochastiques non déterministes. L'aléatoire est considéré comme l'acteur principal lors de toutes les opérations de prise de décision. Ainsi, l'algorithme garantit le respect de la diversification durant toutes les phases de son exécution.

3.2-Tolérance à la détérioration :

La tolérance à la détérioration. Une tolérance motivée par le fait que la structure de l'espace de recherche d'un problème d'optimisation ne permet pas de garantir avec certitude d'arriver à la solution optimale même en partant d'une très bonne solution au départ. D'où l'intérêt de tolérer une détérioration contrôlée et bornée dans le but d'augmenter les chances d'arriver à l'optimum.

3.3-La mémoire

Une métaheuristiques peut faire usage de la mémoire sous plusieurs formes et ceci dans le but de stocker une partie de l'historique de la recherche. Un stockage qui se fait de deux manières différentes à savoir :

Stockage à court terme : un stockage qui se fait dans le but de garder une trace de la meilleure position d'une solution dans le but d'y retourner en cas d'échec d'amélioration.

Stockage à long terme : un stockage sélectif où les solutions jugées prometteuses seront stockées dans le but d'être exploitées plus tard.

3.4-La synchronisation entre la diversification et l'intensification :

L'équilibre entre l'exploration et l'exploitation d'un espace de recherche permet la diversification des solutions et ainsi donner plus d'opportunités à la métaheuristiques.

4- Fonctionnement général des métaheuristiques :

Les métaheuristiques ne nécessitent pas une connaissance particulière sur les problèmes d'optimisation à résoudre. Il suffit d'associer une ou plusieurs variables à une ou plusieurs solutions (optimum).

Les habituelles de dispersion de recherche métaheuristiques reprend les cinq composantes suivantes : [9]

• la diversification,
• d'amélioration,
• Référence set mise à jour,
• la génération de sous-ensemble, et la combinaison de solutions.
• L'idée de base derrière la méthode de recherche de dispersion. Est de combiner les bonnes solutions qui sont découverts par la métaheuristiques.

Les cinq composantes de la métaheuristiques sont définies comme suit:

4.1 Diversification:

La composante diversification cherche à générer un ensemble de solutions diversifiées qui est représentatif de la rechercher l'espace. Dans l'approche présentée dans cet article, la méthode de diversification consiste à générer de la population aléatoirement.

4.2 Amélioration :

Le composant amélioration commence par la recherche d'un vecteur de paramètres qui réduit le nouveau courant la somme carrée moyenne des erreurs (MSE), généralement en effectuant une recherche locale autour d'une bonne solution. Les procédures de recherche locale sont le problème de dépendance.

4.3 Mise à jour des références :

L'ensemble de référence est composé d'un sous-ensemble solution soumise à un processus évolutif dans un tenter de déterminer la solution optimale. Dans cet article, l'ensemble de référence est composé d'un petit ensemble de meilleures solutions trouvé pour la MSE la somme carrée moyenne des erreurs. Il s'agit d'une référence élitiste que défini dans les meilleures solutions sont maintenues. Soit R la référence ensemble, organisé selon les valeurs décroissantes non-MSE

$$MSE(X) = \frac{\sum_{t=1}^{n}(Y(t) - \hat{Y}(t))^2}{n}$$

4.4 Génération de sous-ensemble :

L'étape de génération de sous-ensemble fonctionne de combiner des solutions de R pour créer des nouveaux sous-ensembles. Il est de nombreuses façons de combiner ces solutions, mais les résultats généralement bons peuvent être obtenus même en combinant un petit nombre des solutions.

4.5 Combinaison Solution

L'étape de combinaison encourage la combinaison des solutions contenues dans chaque sous-ensemble.

Les métaheuristiques, 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) [9], où il faut optimiser plusieurs objectifs contradictoires. La recherche vise alors non pas à trouver un optimum global, mais un ensemble d'optima 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 bruite, où il existe une incertitude sur le calcul de la fonction objectif. Incertitude dont il faut alors tenir comptes dans la recherche de l'optimum.

- L'optimisation dynamique, où la fonction objective varie dans le temps. Il faut alors approcher au mieux l'optimum à chaque pas de temps.

- La parallélisassions, 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 métaheuristiques pour qu'elle soit distribuée.

-L'hybridation, qui vise à tirer parti des avantages respectifs de métaheuristiques différentes en les combinant [10].

Les métaheuristiques sont inspirées par des systèmes naturels et utilisées dans de nombreux domaine :En physique (recuit simule), en biologie de l'évolution (algorithmes évolutionnaires) en éthologie (algorithmes de colonie de fourmis) [8].

5-Classification des métaheuristiques :

On peut classe les métaheuristiques selon leur fonctionnements :

5.1 Métaheuristiques a parcours

Les métaheuristiques les plus classiques sont ceux fondes sur la notion de parcours. Dans ce type d'algorithme ce dernier fait évoluer une seule fonction objectif sur l'espace de recherche local a chaque itération puis la compare aux optimums. La compréhension de la notion de voisinage est alors nécessaire.

Les plus connes dans cette classe sont le recuit simule et la recherche de tabous [8].

5.2. Métaheuristiques à population

Dans cette famille les métaheuristiques utilisent la notion de population :

Ils manipulent un ensemble de solutions en parallèle. Chaque élément de la population parcours un certain nombre de solutions dans l'ensemble local. On peut les assimiler les métaheuristiques a des méta parcours. Parmi les algorithmes inclus dans cette classification on peut citer les algorithmes génétiques et les algorithmes de colonie de fourmis. [8]

5.3 Métaheuristiques à méthodes implicites

Lors de l'utilisation des méthodes implicites, avec les algorithmes génétiques, la distribution de probabilité n'est pas connue ou n'est pas utilisée le choix de l'échantillonnage entre deux itérations ne suit pas une loi donnée, mais est fonction de règles locales. [10]

5.4 Métaheuristiques a méthodes explicites

Ces méthodes utilisent une distribution de probabilité choisie à chaque itération. C'est le cas des algorithmes a estimation de distribution, comme leur nom indique, estime a chacune de leur itération et via une distribution de probabilité l'espace de recherche local optimal [10].

6 . Les algorithmes métaheuristiques :

On a plusieurs méthodes métaheuristiques et on va choisir les 5 méthodes suivantes :

6.1 Algorithmes génétiques

Les algorithmes génétiques (AG) sont des techniques de recherche inspires par l'évolution biologique des espaces. Introduits par J.H. Holland au début des années 1970, ils ont d'abord suscité un intérêt limité, du fait de leur important coût d'exécution. Ils connaissent, depuis les années 1990, un développement considérable, notamment suite l'apparition des architectures massivement parallèles, qui exploitent leur (parallélisme intrinsèque) [8].

6.1.1 Principe de base

Le principe d'un AG se décrit simplement, dans le cas de la minimisation d'une fonction f. Un ensemble de N points, qui peuvent • être choisis au hasard, constitue la population initiale ; chaque individu x de la population possède une certaine compétence, qui mesure son degré d'adaptation l'objectif visé, x est d'autant plus comptent que f (x) est petit. Un AG consiste faire évoluer progressivement, par générations successives, la composition de cette population, en maintenant sa taille constante, d'une génération a la suivante, la (compétence) de la population doit globalement s'améliorer ; un tel résultat est obtenu en mimant les deux principaux mécanismes qui régissent l'évolution des êtres vivants la sélection naturelle (qui détermine quels membres d'une population survivent et se reproduisent) et la reproduction (qui assure le brassage et la recombinaison des, pour former des descendants aux potentialités nouvelles).

En pratique, chaque individu est codé par une chaîne de caractères de longueur donnée (de même Qu'un chromosome est forme d'une chaîne de gènes).

Le passage d'une génération la suivante se déroule en deux phases : une phase de reproduction et une phase de remplacement. La phase de reproduction consiste à appliquer des operateurs, dits génétiques, sur les individus de la population courante, pour engendrer de nouveaux individus ; les opérateurs les plus utilisés sont le croisement, qui produit deux descendants partir de deux parents, et la mutation, qui produit un nouvel individu a partir d'un seul individu. Les individus de la population prenant part à la reproduction sont préalablement sélectionnés, en respectant le principe suivant : plus un individu est compétent, plus sa probabilité de sélection est élevée. La phase de remplacement consiste ensuite a choisir les membres de la nouvelle génération : on peut, par exemple, remplacer les plus mauvais individus (au sens de la fonction objectif) de la population courante par les meilleurs individus produits (en nombre égal). L'algorithme est interrompu après un nombre donné de générations.

Nous avons représenté en figure suivante le principe d'un algorithme génétique.

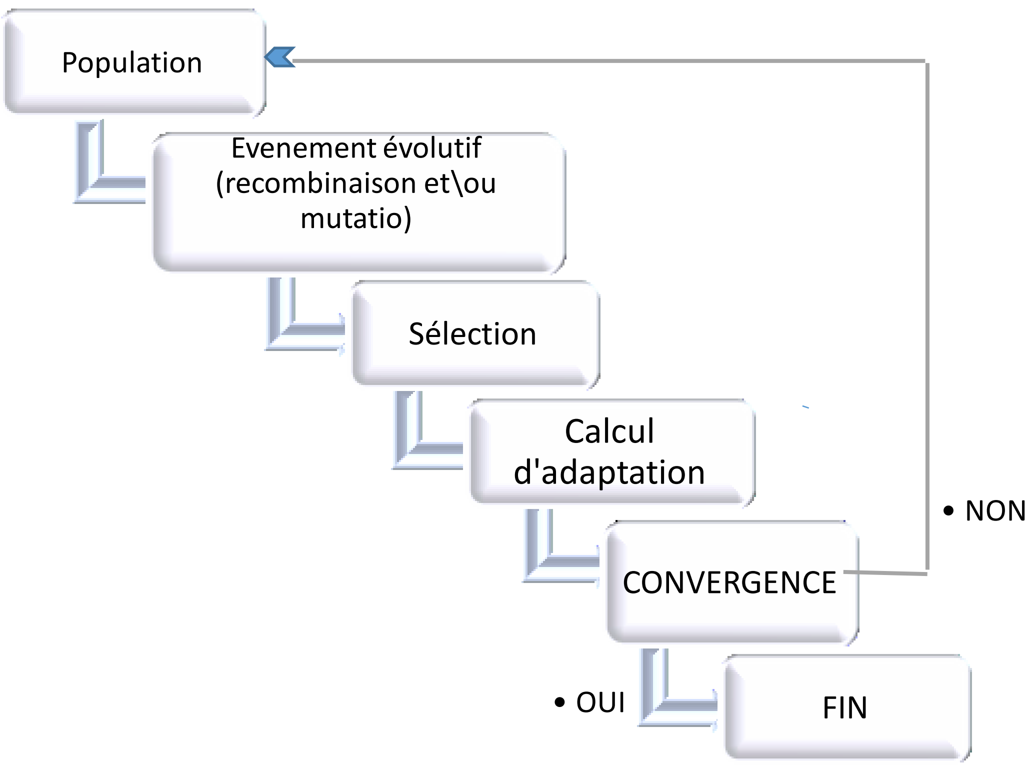


Figure2.1 : schéma d'un algorithme génétique [31]

[8] Voici l'algorithme génétique de base [11].

Début
1 : Générer une population aléatoire de n chromosomes
2 : Evaluer la fitness des chromosomes avec la fonction f (x)
3 : Répéter
4 : Calcule la fonction fitness f (x), pour tout chromosome x
5 : Appliquer l'opération de sélection
6 : Appliquer l'opération de croisement avec une probabilité PC
7 : Appliquer l'opération de mutation avec une probabilité PM
8 : Ajouter les nouveaux chromosomes à la nouvelle population
9 : Calcule la fonction fitness f (x), pour tout chromosome x
10 : Appliquer l'opération de remplacement
11 : Jusqu'a la satisfaction des conditions de terminaison
Fin

L'algorithme génétique comporte trois phases distinctes [12].

La production de la population d'individus la mieux adaptée pour contribuer a la reproduction de la génération suivante (version artificielle de la sélection naturelle) elle peut être mise en œuvre sous plusieurs formes algorithmique; la phase de reproduction, qui exploite essentiellement les operateurs de croissement et de mutation ;la stratégie de remplacement des populations parent et enfant par la génération suivante. Elle pourra être mise en œuvre sous plusieurs formes.

6.2- Le recuit simulé

6.2.1 Principe de base

Le recuit simulé est une métaheuristiques probabiliste permettant d'approximer l'optimum global d'une fonction objectif. Cette technique est souvent utilisée lorsque le calcul de la solution optimale exacte demanderait un temps de calcul trop important, le recuit simulé est alors utilisé pour trouver une solution approchée dans un temps raisonnable.

Le nom et le principe de cette métaheuristiques sont inspirés directement de la technique du recuit en métallurgie. Le recuit permet de diminuer les défauts d'un matériau en augmentant sa température puis en contrôlant le refroidissement par un apport externe de chaleur. Un refroidissement naturel ne permet pas d'atteindre la configuration optimale des atomes pour obtenir la meilleure stabilité. De manière analogue, pour le recuit simulé, nous parlerons de l'énergie du système, nous cherchons à atteindre la configuration du système qui possède l'énergie la plus faible.

A chaque itération du recuit simulé, la solution actuelle est remplacée par une autre solution issue du voisinage de la solution initiale. Cette nouvelle solution est acceptée avec une probabilité dépendant à la fois de la différance entre les valeurs de cout des deux solutions et d'un paramètre T appelé température. Cette température décroit progressivement au cours du temps selon une loi déterminée préalablement. Il y a donc toujours une probabilité de dégrader la solution lors d'une itération.

Cette stratégie permet d'´éviter de rester coincer dans un minimum local.

Le recuit simulé est une technique générique qui peut être appliquée aussi bien à des problèmes discrets que des problèmes continus. Généralement, le recuit simulé est plus souvent utilisé dans des cas discrets, on peut par exemple citer également le problème du voyageur de commerce ou encore le problème du flow shop.[13]

1) Choisir, aléatoirement, une solution initiale **x** du système à optimiser et évaluer la valeur de la fonction objectif f = f(**x**) ;
2) Choisir une température initiale "élevée" T.
3) Perturber cette solution pour obtenir une nouvelle solution **x'** = **x** + **Δx** ;
4) Calculer Δf =f(**x'**) - f(**x**) ;
5) Accepter ou refuser la solution x', en appliquant une certaine "règle d'acceptation" (généralement, la règle de Metropolis.
6) Sauver le meilleur point rencontré ;
7) Si l'"équilibre thermodynamique" du système à la température T est atteint,
   Alors abaisser légèrement la température T ;
   Sinon Aller à l'étape **3)** ;
8) Si le "système est figé" (par exemple, la température T est inférieure à une température seuil voisine de 0),
   Alors Aller à l'étape **9)** ;
   Sinon Aller à l'étape **3)** ;
9) Solution = meilleur point trouvé ; Arrêt du programme.

6.2.2 Pseudo code

Voici le pseudo code de la version la plus simple du recuit simulé:

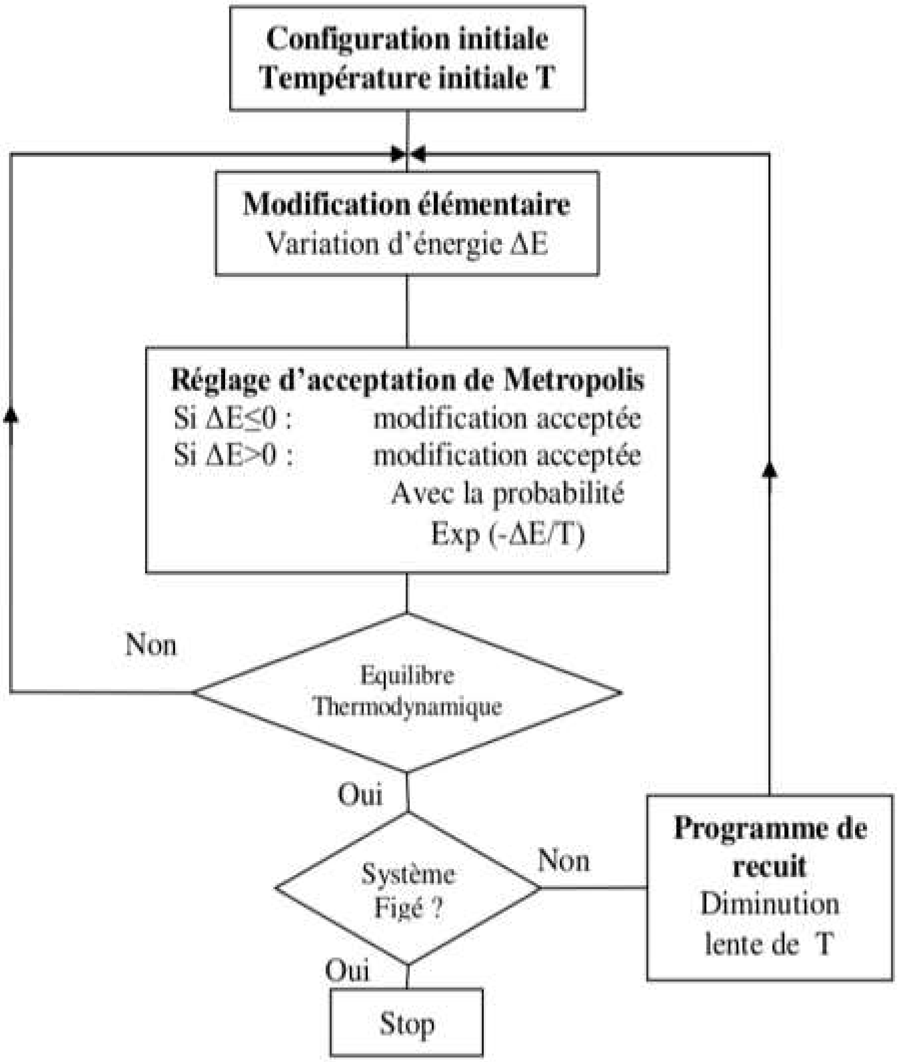


Figure 2.2 : organigramme de l'algorithme de recuit simule[32]

6.3 La méthode recherche Tabou

La recherche Tabou a été appliquée pour résoudre un problème d'affectation de fréquences [29]. L'algorithme est exécuté avec un seul jeu de poids qui est fixé suivant les priorités des objectifs Tabou, un sujet qu'il est préférable de ne pas aborder si l'on veut respecter les codes de la société.

La recherche Tabou dans le domaine de la recherche opérationnelle .La recherche Tabou est une méthode d'optimisation mathématique de la famille des techniques de recherche locale présentée pour la première fois en 1986, et elle est devenue très classique en optimisation combinatoire [12].

Elle se distingue des méthodes de recherche locale simples par l'introduction de la notion d'historique dans la politique d'exploration des solutions pour diriger au mieux la recherche dans l'espace. Cette méthode s'est révélée particulièrement efficace et a été appliquée avec succès à de nombreux problèmes difficiles.

6.3.1 Principe de la méthode

L'idée de départ

Se déplacer de solution en solution (en visitant éventuellement des solutions moins bonnes) en s'interdisant de revenir à une solution déjà rencontrée.

A chaque itération, on examine V(i) et nous allons sur la meilleure solution i'
même si le coup remonte (F (i) > F(i)).

Donc : La recherche Tabou ne s'arrête pas au premier optimum trouvé.

Le danger serait alors de revenir à i immédiatement, puisque i est meilleure que i'. Pour éviter de tourner ainsi en rond, on crée une liste T qui mémorise les dernières solutions visitées et qui interdit tout déplacement vers une solution de cette liste. Cette liste T est appelée liste Tabou.

On conserve en cours de route la meilleure solution trouvée .

On stoppe dès que le critère de fin est vérifié.

6.3.2Principe de algorithme

Générer une solution initiale S de manière aléatoire
S*← S ; C*← F(S) / S* est la meilleure solution rencontrée, C* est sont cout et F la
fonction objectif
Ajouter S a la liste Tabou ; K← 0
Répéter tant qu'un critère de fin n'est pas vérifié
Choisir parmi le voisinage de S_K. $V(S_K)$, le mouvement qui minimise F et qui n'appartient pas à la liste Tabou, meilleur (S_K)
S_{K+1} ← meilleur(S_K)
Si la liste Tabou est pleine **alors**
Remplacer le dernier élément de la liste Tabou par S_{K+1}
Si non
Ajouter S_{K+1} a la liste Tabou
Fin si
Si (C (S_{K+1}) < C*) **alors**
S*← S_{K+1}, C*← C (S_{K+1})
Fin si
Fin d'algorithme

Lorsque la mémoire est pleine, elle est gérée comme une liste circulaire en FIFO (First In First Out)

On élimine le plus vieux point Tabou et on insère la nouvelle solution. La taille de la mémoire permet de ne pas saturer rapidement les ressources disponibles pour la recherche et permet de surcroît d'adapter facilement la méthode à un espace de recherche dynamique.

Voici l'organigramme simple de l'algorithme tabou.

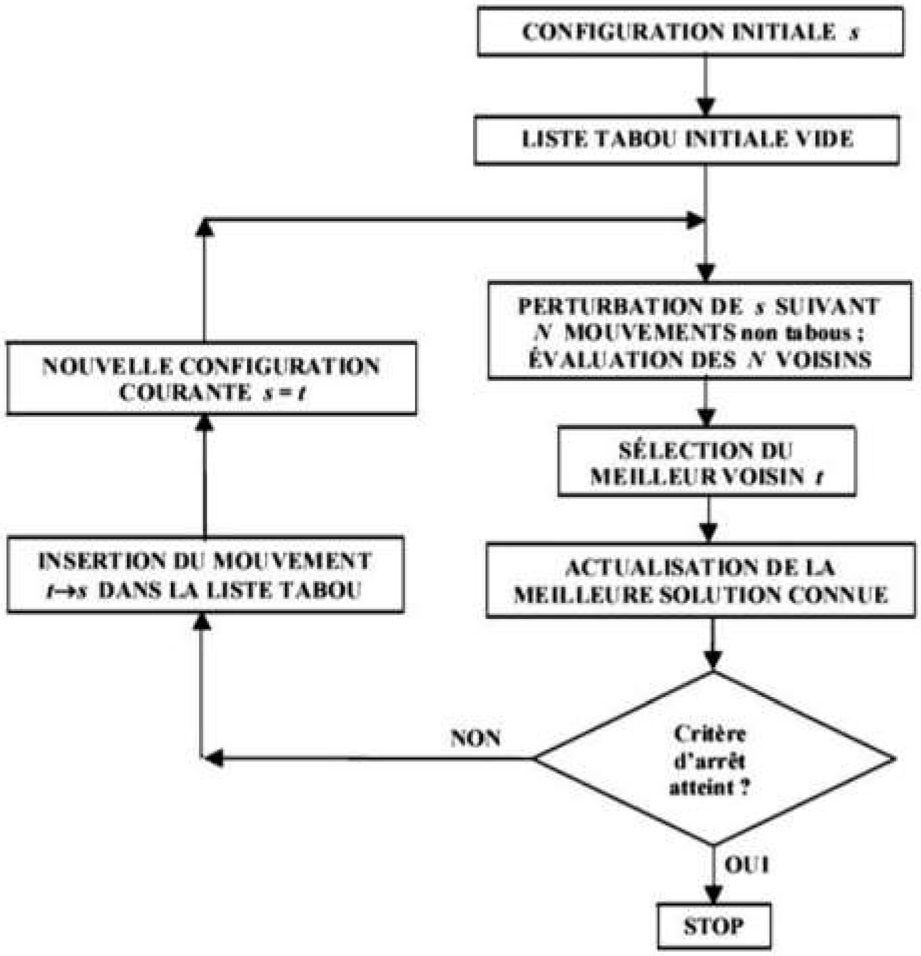


Figure 2.3 : organigramme de l'algorithme tabou simple [33]

6.4 Les essaims particulaires (Particle Swarm Optimization: PSO)

PSO est une métaheuristiques proposée par Kennedy et Eberhart (1995). Elle est basée sur la métaphore des interactions et communications sociales. Au départ de l'algorithme, un essaim est réparti au hasard dans l'espace de recherche, chaque particule ayant également une vitesse aléatoire. Ensuite, à chaque pas de temps : particule est capable d'évaluer la qualité de sa position et de garder en mémoire sa meilleure performance, c'est-à-dire la meilleure position qu'elle a atteinte jusqu'ici (qui peut en fait être parfois la position courante) et sa qualité (la valeur en cette position de la fonction optimiser).

chaque particule est capable d'interroger un certain nombre de ses voisins et d'obtenir de chacune d'entre elles sa propre meilleure performance (et la qualité afférente).

A chaque pas de temps, chaque particule choisit la meilleure des meilleures performances dont elle a connaissance, modifie sa vitesse en fonction de cette information et de ses propres données et se déplace en conséquence [14].

A chaque itération, La vélocité et la position sont mise à jour suivant deux forces best local et global. La première l'attire au best local qui est la position qui a donné la meilleure fitness pour la particule (i.e c'est la position la plus proche de l'objectif que cette particule a pu atteindre). L'autre best est le global best, c'est la meilleure position trouvée par la particule et ses voisins *Bli* , et *Bg* sont les best locaux et globaux.

n est la taille des files d'attente.

Le pseudo code de l'algorithme de cette métaheuristiques adapté pour résoudre notre problème est le suivant :

S'il y'a une place libre dans la station de chargement alors
    Initialiser la population.
    Initialiser les paramètres.
    Tant que (critère d'arrêt est non atteint)
        Pour chaque particule *Xi*
        Calculer le produit des charges de routages de cette particule.
            Si *F(xi) > F(Bli)* alors : (mise à jour du best local)
                Bli=xi
                Fin si
            Si *F (xi) > F (Bg)* alors : (mise à jour du best global)
                Bg= xi
            Fin si
        Fin pour
    Pour chaque particule *Xi*
    $\boldsymbol{X_i(t) = c_2 \oplus F_2(c_1 \oplus F_2(w \oplus F_1(X_i(t-1)), Bl_i(t-1), B_g(t-1))}$
    Fin pour
    Fin Tant que
Fin si

F1 représente un opérateur qui modifie les routages de quelques pièces parmi les premières pièces de la file infinie avec une probabilité *w*, un nombre aléatoire uniforme *r* est généré entre 0 et 1. Si *r* est inférieur à *w* alors *F1* est appliqué pour produire une permutation perturbée. De la même manière, on applique *F2* et *F3* qui représentent les croisements avec les best locaux et globaux selon des probabilités *C1* et *C2.*

Voici l'organigramme simple de l'algorithme PSO

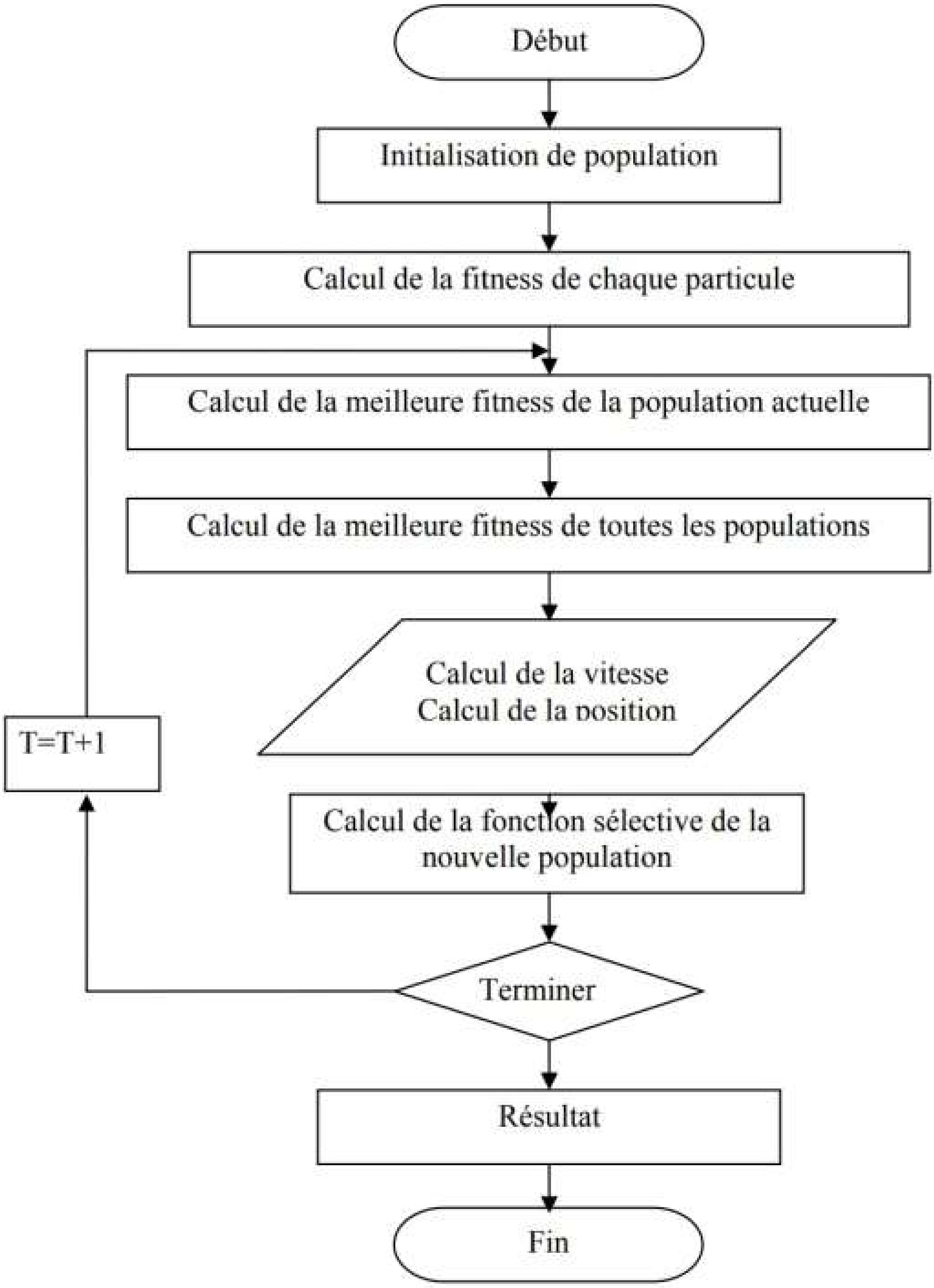


Figure 2.4 : organigramme de l'algorithme PSO[34]

6.5 Les colonies de fourmis (Ant Colony Optimization: ACO) [16]

Cette métaheuristiques a été introduite pour la première fois par Dorigo (1992) et a été inspirée des études du comportement de fourmis réelles pour résoudre naturellement des problèmes relativement complexes.

-Le premier algorithme de cette métaheuristiques a été appliqué pour résoudre le problème du voyageur de commerce, le principe de cet algorithme est simple. Lorsqu' une fourmi k se déplace de la ville i à la ville j, elle laisse une trace sur le chemin. De plus, elle choisit la prochaine ville à visiter à l'aide d'une probabilité basée sur un compromis entre l'intensité de la trace et la visibilité qui représente l'inverse de la distance entre i et j.(dij).

-L'importance relative des deux éléments est contrôlée par deux coefficients α et β. Chaque fourmi k possède une forme de mémoire $tabou_K$, lui rappelant la liste ordonnée des villes déjà visitées afin d'obliger celle-ci à former une solution admissible. Après un tour complet, chaque fourmi laisse une certaine quantité de phéromone qui dépend de la qualité de la solution trouvée sur l'ensemble de son parcours.

-L'algorithme original a été adapté à notre problème en remplaçant la ville i par la pièce i et la ville j par le routage j.

-Pour chaque pièce i, le choix du routage j est basé sur un compromis entre l'intensité de la trace $\Delta\tau_{ij}^{K}$(t) et la visibilité ηij (dépend du nombre de pièces dans la file d'entrée de la première machine de ce routage et sa charge).

- L'importance relative des deux éléments est toujours contrôlée par deux coefficients $\boldsymbol{\alpha}$ et $\boldsymbol{\beta}$. Si le nombre total de fourmis est $\boldsymbol{m}$ et la taille de la station de chargement est $\boldsymbol{n}$, un cycle est réalisé lorsque chacune des $\boldsymbol{m}$ fourmis affecte les $\boldsymbol{n}$ premières pièces de la file infini à des routages $\boldsymbol{j}$.

- Après un tour complet (l'affectation de toutes les n premières pièces de la file infini aux routages par les fourmis), chaque fourmi laisse une certaine quantité de phéromone $\Delta\tau_{ij}^{K}$(t) qui dépend de la qualité de la solution trouvée (le produit des charge de routages) sur l'ensemble des routages sélectionnés pour les pièces.

S'il y'a une place libre dans la station de chargement alors

Pour t = 1 à t_{max}

Pour chaque fourmi k = 1 à m

Choisir pour la première pièce de la file infini un routage au hasard suivant son type.

Pour chaque pièce i contenu dans la deuxième place jusqu' au la n^{eme} place de la file infini.

Choisir un routage j, parmi les routages possibles selon une probabilité dépendant de l'intensité de la trace et du nombre de pièces dans la file d'entrée de la première machine de ce routage et sa charge.

Fin Pour

Evaluation de la fonction objectif. (Produit des charges de routages)

Déposer une $\Delta\tau_{ij}^{K}(t)$piste sur le trajet $T^{K}(t)$ (pour chaque routage *j* choisi pour la pièce *i* par la fourmis *k*).

Fin Pour.

Évaporer les pistes et modifier les intensités.

Fin Pour.

Finsi.

Le pseudo code de l'algorithme est :

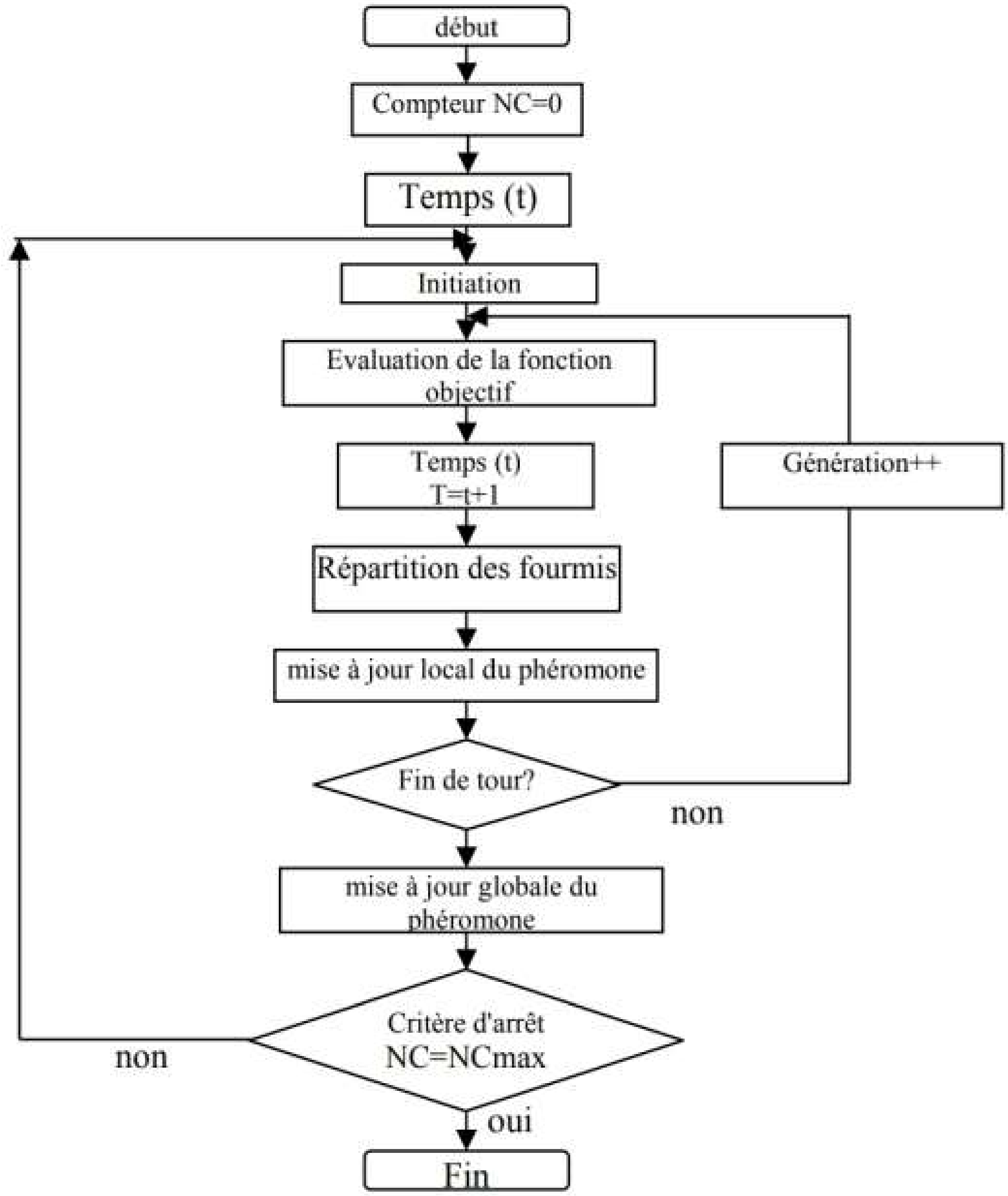


Figure 2.5 : organigramme de l'algorithme de ACO

7-Schéman Général :

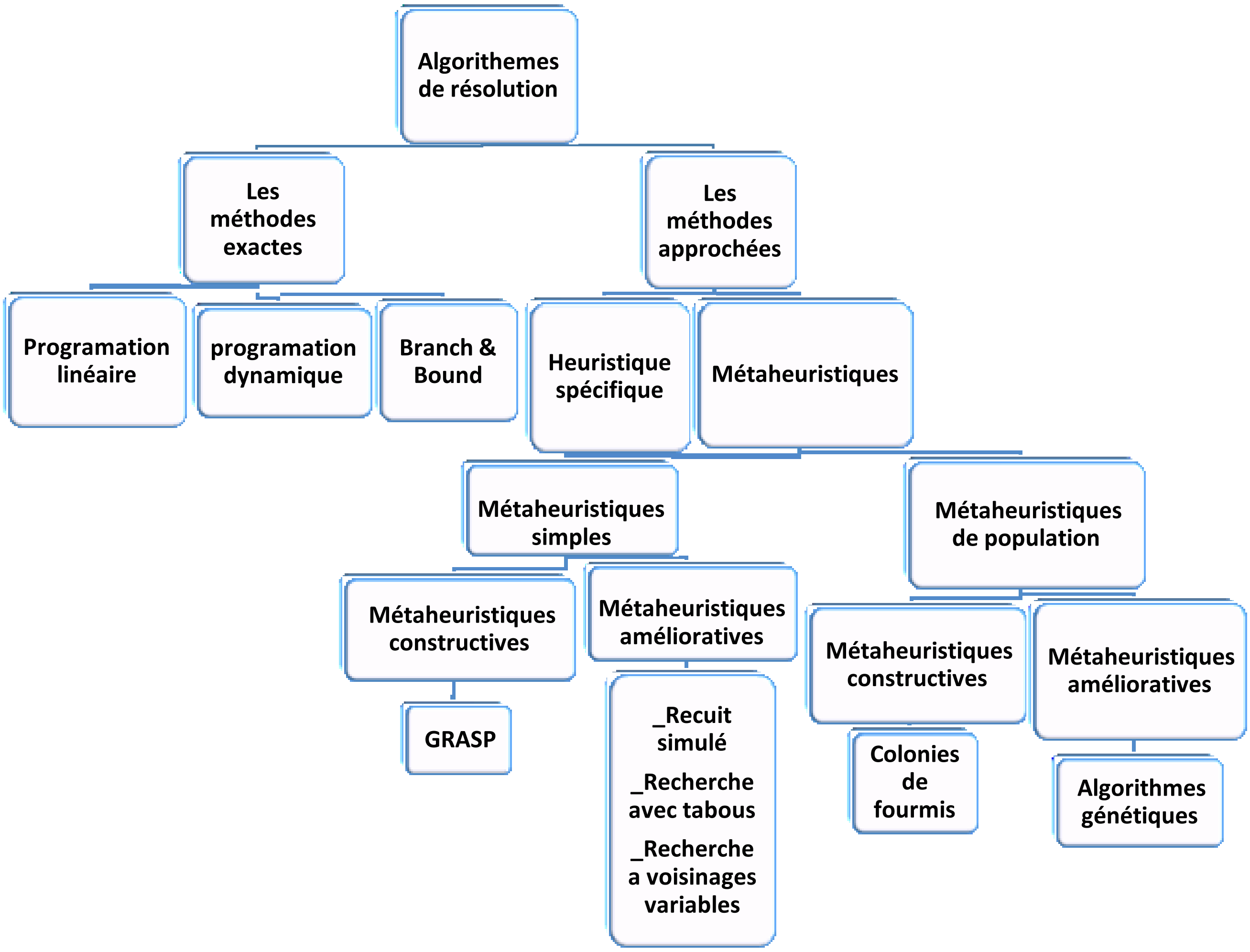


Figure 2.6 : schéma général de les méthodes d'optimisation

Conclusion

Dans ce chapitre, nous pouvons conclure que les méthodes exactes ne peuvent pas toujours être applicables notamment dans le cas où le problème NP complet. par conséquent, les méthodes approchées peuvent être un bon terrain pour travailler sur ce type de problème.

Généralement, les métaheuristiques sont des méthodes très puissantes permettant de trouver de bon solutions approchées à ces problèmes et un temps raisonnable.

Dans le chapitre suivant ,nous mettons l'accent d'une manière détaillée sur la méthode de colonies de fourmis appliquée à un problème de véhicule avec capacité.

Chapitre 03 : Application de l'algorithme de colonie de fourmis sur le CVRP

1.Introduction

Dans ce chapitre, on s'intéresse à la résolution du problème CVRP par les méthodes de colonie de fourmis visant à chercher le plus court chemin avec une dégradation maximale de la charge de véhicule d'une ville à une autre. Une technique de roulette est ajoutée dans l'algorithme de colonies de fourmis avec une agrégation des deux objectifs définie dans le choix de transition, une technique de k-mans pour regrouper les villes plus approché .

Nous allons tout d'abord présenter le principe de la méthode de colonie de fourmis, puis on donne la formulation du problème CVRP et on présente la description de l'algorithme obtenu.

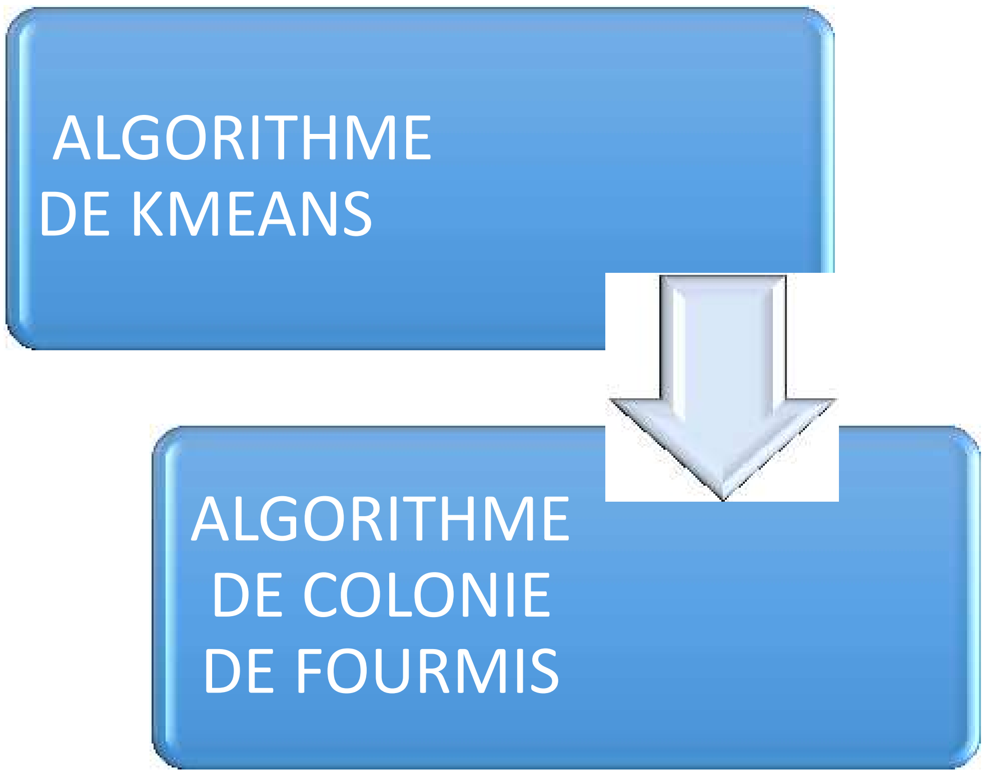


Figure 3.1 : Schéma générale

2 :Historique

Les algorithmes de colonie de fourmis ont été proposés par Colorni, Dorigo et Maniezzo en 1992 et appliquées la première fois au problème de voyageur de commerce. Ce sont des algorithmes itératifs à population où tous les individus partagent un savoir commun qui leur permet d'orienter leur futur choix et d'indiquer aux autres individus des choix à suivre ou à éviter.

Une colonie de fourmis ayant le choix entre deux chemins d'inégale longueur menant à une source de nourriture avait tendance à utiliser le chemin le plus court.

Un modèle expliquant ce comportement est le suivant [17] :

✓ Une fourmi (appelée « éclaireuse ») parcourt plus ou moins au hasard l'environnement autour de la colonie.

✓ Si celle-ci découvre une source de nourriture, elle rentre plus ou moins directement au nid, en laissant sur son chemin une piste de phéromones.

✓ Ces phéromones étant attractives, les fourmis passant à proximité vont avoir tendance à suivre, de façon plus ou moins directe, cette piste.

✓ Si deux pistes sont possibles pour atteindre la même source de nourriture, celle étant la plus courte sera, dans le même temps parcoure par plus de fourmis que la longue piste.

✓ La piste courte sera donc de plus en plus renforcée, et donc de plus attractive.

✓ La longue piste finira par disparaitre, les phéromones étant volatiles.

✓ A terme, l'ensemble des fourmis a donc déterminé et « choisi » la plus courte.

3. Méthode de k-means :

La méthode d'agrégation autour des centres mobiles est la méthode la plus utilisée de par sa simplicité et la qualité des résultats qu'elle offre. On la retrouve sous d'autres appellations ou d'autres variantes à savoir la méthode des mans, des c-means, la méthode ISODATA, abréviation de Itérative Self Organizing Data Analysis [23], la méthode des fuzzy c-mans utilisant la logique floue, Ces algorithmes se ressemblent dans le principe général, mais différent dans la façon de procéder pour aboutir à une partition finale. Elles procèdent par groupement de données selon une distance comme la distance euclidienne. Chaque classe est représentée par son centroïde. La procédure de partitionnement s'effectue par rapport aux centroïdes des classes. La méthode d'agrégation autour des centres mobiles ou k-means à laquelle nous nous intéressons particulièrement, permet de construire une partition finale qui est censée être la meilleure à partir d'une partition initiale qui peut être fixée par l'utilisateur d'une manière aléatoire. Le choix de cette partition initiale reste toutefois délicat. Cependant, il existe des méthodes qui permettent de la déterminer automatiquement. Comme nous le verrons plus bas, les classes obtenues peuvent caractérisées à l'aide des moments interclasse et interclasse qui nous renseignent sur la qualité de la partition finale. On se réfère, donc, au moment d'ordre deux d'un ensemble I de N points par rapport à leur centre de gravité G. Ce moment est donné par la formule générale suivante :

où mi sont des masses ponctuelles ou des pondérations, et d(i,G) les distances des points i au centre de gravité. Lorsque les observations sont décrites par M variables (espace à M dimensions), ces distances, lorsqu'elles ont euclidiennes, sont données par la formule suivante:

Ainsi l'expression donnant le moment centré d'ordre 2 devient :

Dans le repère à M dimension, la .... coordonnée du centre de gravité G du nuage des N observations est donnée par:

Si les observations ne sont pas pondérées, (m_i=1), nous aurons :

Ainsi, le moment centré d'ordre deux M^2 (IG) correspond à l'écart type d'une distribution gaussienne. L'écart type étant un paramètre de dispersion, on remarque que si les points sont très concentrés autour du centre de gravité, la valeur de M^2 (I,G) devient faible. Dans le cas où ces points sont très dispersés, M^2 (I,G) prend une valeur plus élevée.

Le moment centré d'ordre deux peut ainsi nous renseigner sur la qualité des classes obtenues en classification.

Le nuage I peut être décomposé en L sous nuage ou sous-ensembles disjoints in n=1,2..., L. Chaque sous-nuage in correspond à un nombre Kn d'éléments tel que K N , n=1,2,...L. Pour chaque sous-nuage In=1,2,...,, on calcule le moment centré d'ordre deux. En notant Les barycentres de ces sous nuages, le moment du sous-nuage In par rapport à son centre de gravité Gm est calculé par la formule suivante:

Par rapport au barycentre G de l'ensemble des points ou du nuage global En, ce moment est: En appliquant le théorème de Huyghens pour le sous ensemble I_n on obtient:

Le terme représente la dispersion intra classée représente la dispersion interclasses.

La formule donnant M^2 (I,G) montre que le moment centré d'ordre deux du nuage I_n est la somme des moments centrés d'ordre deux des sous-nuages augmentée du moment interclasse. En d'autres termes, la dispersion totale est décomposée en dispersion à l'intérieur des sous-ensembles ou des classes.

La dispersion intra classe est de valeur élevée si les barycentres des classes formées sont éloignés du barycentre du nuage Quant aux dispersions intra classes, elles sont de valeurs faibles si les points constituant chaque classe sont très rapprochés du centre de gravité correspondant. En classification automatique, on cherche à minimiser la dispersion intra classe et à maximiser la dispersion intra classe. En effectuant le rapport entre le moment interclasse et le moment interclasse, on peut constituer l'indice R qui est le rapport entre ces deux moments Cet indice R est calculé par la formule suivante : où L est le nombre de classes ou sous-nuages disjoints in de barycentres C_n, K_n, le nombre d'éléments de la classe In M, le nombre de variables décrivant chaque observation et xi, Xj. xG_j, respectivement les coordonnées des objets dans l'espace à M dimensions, le centre de gravité du sous-nuage In de la classe k et le centre de gravité de tout le nuage de points.

On note que dans cet algorithme, le choix des i classes ou centres initiaux s'effectue sur la base d'un tirage aléatoire sans remise à partir de la population à classifier. La partition des classes est modifiée à chaque affectation d'un individu i à l'une des classes après le calcul des distances euclidiennes par rapport à chaque centre un des classes initiales. L'individu est affecté à la classe correspondant à la distance la plus faible. Cette opération est effectuée pour chaque élément du nuage . Les centres de nouvelles classes ainsi formées sont recalculés. Le calcul à nouveau de toutes les distances de chaque élément à ces nouveaux centres est effectué encore et l'affection de ces éléments est aussi effectuée et de nouvelles classes sont ainsi formées. L'opération se poursuit ainsi jusqu'à l'arrêt de la procédure.

Le déroulement de la méthode des k-means s'effectue selon l'algorithme

Etape 0 : on choisit par un tirage aléatoire sans remise k individus permis n individus composant l'ensemble I. ces k centre notés $\{c_1^0, c_2^0 \ldots \ldots \ldots \ldots c_k^0\}$ sont provisoires.

Etape 1 : chaque individu i de I est affecté à une classe et une seule . chacune de ces classes est localisée par son centre. La procédure d'affectation est la suivante : l'élément i est affecté à la classe notée p_1^0 de centre c_1^0 si et seulement si :

$$d(i, c_1^0) < d(i, c_k^0), k\#1\,; k = 1,2, \ldots \ldots, K$$

Après avoir affecté tous les individus on obtient notés ($p_1^0, p_2^0 \ldots \ldots \ldots \ldots, p_k^0$) de centre respectifs $\{c_1^0, c_2^0 \ldots \ldots \ldots, c_k^0\}$.

Etape 2 : en considérant les k classes obtenues à l'étape 1, on recalcule les centre de gravité . On obtient donc k nouveaux centre notés $c_1^1, c_2^1 \ldots \ldots, c_k^1$. On utilise la meme règle d'affectation qu'à l'étape 1.on obtient k nouvelles classes ($p_1^1, p_2^1, \ldots \ldots \ldots, p_k^1$) de centre respectifs ($c_1^1, c_2^1, \ldots \ldots \ldots, c_k^1$).

Etape 3 : on détermine k nouvelles classes en calculant les centres de gravité des classes obtenues à l'étape (h-1). La règle d'affectation reste la même qu'à l'étape précédente et on obtient, par la suite, une nouvelle répartition ($\mathrm{p}_1^{\mathrm{h}}, \mathrm{p}_2^{\mathrm{h}}, \ldots, \mathrm{p}_k^{\mathrm{h}}$) de centre respectifs ($\mathrm{c}_1^{\mathrm{h}}, \mathrm{c}_2^{\mathrm{h}} \ldots \ldots, \mathrm{c}_k^{\mathrm{h}}$).

Arrêt de l'algorithme

L'arrêt de l'algorithme de la méthode des { k-means } se fait

- Lorsque deux itérations successives conduisent à une même partition.

Lorsqu'on fixe un critère d'arrêt tel que le nombre maximal d'itérations.

Illustre un exemple de déplacement de deux entrées centres d'inertie de deux classes de données et la formation des classes.

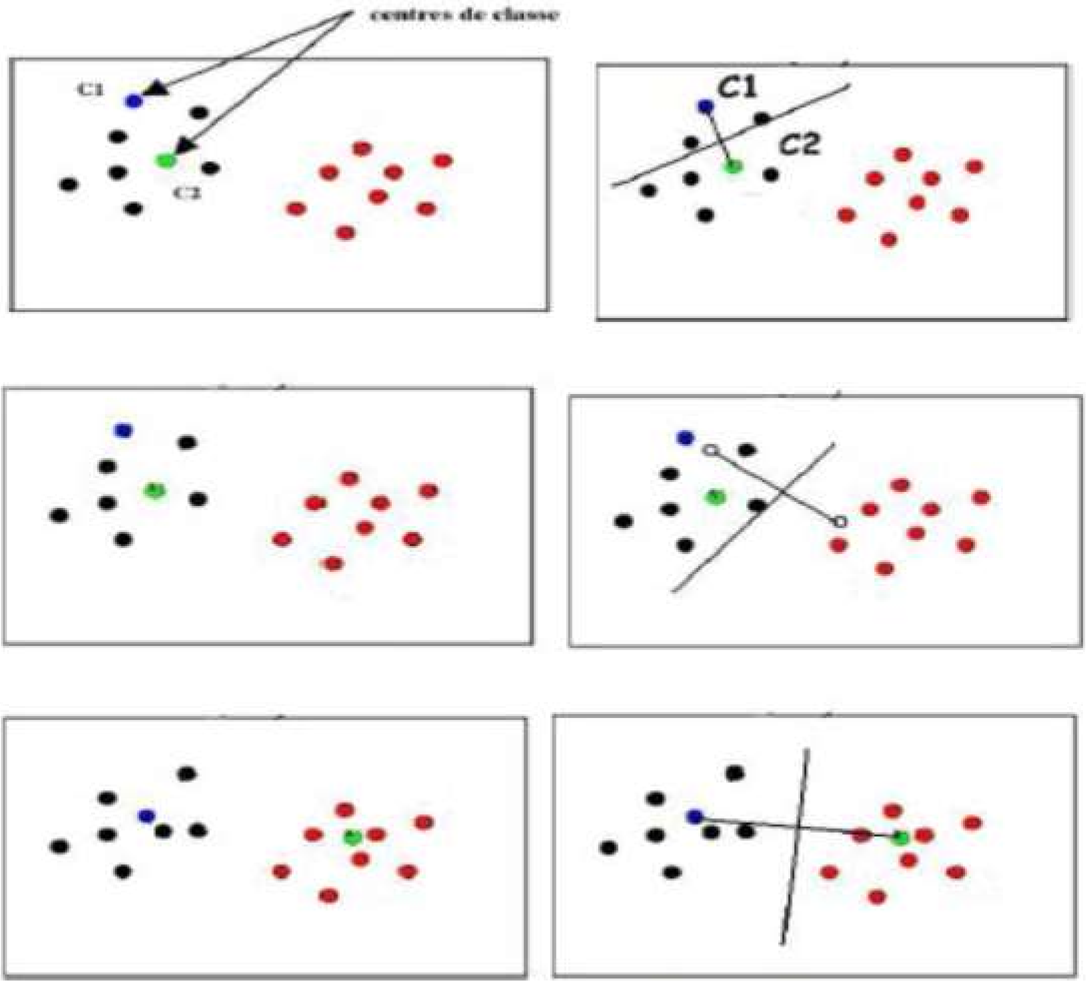


Figure 3.2 :exemple de déplacement des centres des classe

4.Les algorithmes de colonies de fourmis

4.1. Les fourmis réelles

L'étude des fourmis a longtemps été négligée par les entomologistes. Jusqu'à ce que, Hölldobler et Wilson ont corrigé cette lacune en 1990 en publiant un ouvrage concentrant tout ce que l'on connaissait alors des fourmis [19].

Les fourmis constituent à l'heure actuelle un support majeur pour les théories développées en écologie comportementale et en sociobiologie. On peut citer plusieurs raisons à cette inspiration:

a) l'influence des fourmis sur leur environnement naturel est extrêmement importante. Il a par exemple été montré qu'elles déplacent plus de terre en forêt tropicale que les vers de terre, ou encore que le poids total des fourmis sur terre est du même ordre de grandeur que le poids des humains. De plus, la domination des fourmis est une preuve de leur adaptation à des environnements très variés)

b) l'étude des fourmis se fait assez facilement en laboratoire car elles s'adaptent sans trop de difficultés à des environnements différents de leur habitat d'origine .

c) les fourmis possèdent une gamme de comportements très variés, collectifs ou individuels.

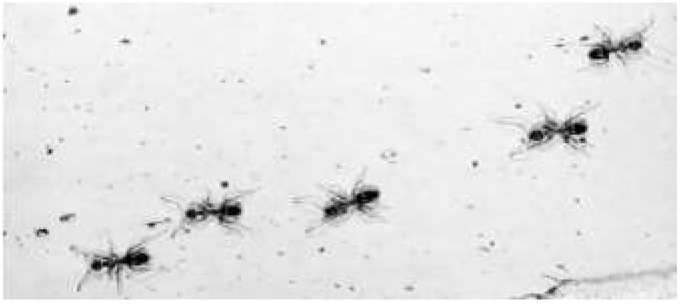


Figure 3.3 : Les fourmis réelles[35]

4.1.1. L'intelligence collective des fourmis

Malgré que certaines espèces des fourmis aient des capacités individuelles étonnantes telles que des capacités visuelles inhabituelles et des capacités d'apprentissage mais la plupart des caractéristiques qui nous intéressent sont cependant collectives. On parle d'intelligence collective quand un groupe social peut résoudre un problème dans un cas où un agent isolé en serait incapable. Cette intelligence est basée sur les processus d'auto-organisation.

L'auto-organisation se parant bien à l'étude des insectes sociaux montrent des comportements collectifs complexes issus de comportements individuels simples. On peut regrouper les processus d'auto-organisation chez les insectes sociaux en quatre groupes tant leur diversité est importante [18].

– la division du travail et l'organisation des rôles sociaux : à l'intérieur d'une même société,

On peut observer différentes catégories spécialisées dans un certain nombre de tâches (la recherche de nourriture, la défense du nid, ...);

- l'organisation de l'environnement : la construction du nid est un symbole de l'organisation distribuée des insectes. Le nid est construit sans que les insectes soient dirigés, ils répondent à un certain nombre de stimuli provenant de leur environnement :

- la reconnaissance interindividuelle : chaque fourmi est capable d'identifier ses congénères tout en participant elle-même à l'identité de sa colonie (par exemple l'échange d'aliments entre les individus d'une même colonie 'trophallaxie')

- le recrutement et l'exploitation collective des sources de nourriture : le fourragement met à jour des stratégies qui permettent aux insectes une grande adaptation à leur milieu.

Les capacités des fourmis en matière de coopération, de communication, de compétition et d'apprentissage, entre autres, peuvent être mises à profit pour la conception d'algorithmes de résolution des problèmes d'optimisation [20].

4.1.2. La communication

Les insectes sociaux en général, et les fourmis en particulier, ont développé des mécanismes de communication très élaborés. Il a été défini douze types de réponse mettant en œuvre une forme de communication [26] [19]:

01. l'alarme;

02. l'attraction simple:

03. le recrutement (pour une source de nourriture ou un site de nidification) :

04. l'entretien et l'amélioration;

05. la trophallaxie (échange de liquides);

06. l'échange d'aliments solides;

07. les effets de groupe (augmentation ou inhibition d'une activité);

08. la reconnaissance des apparentés ou de caste;

09. la compétition pour la reproduction ;

10. la détermination de catégorie: la compétition pour la reproduction:

11. le marquage du territoire et du nid;

12. la reproduction (différenciation du sexe, de l'espèce, de la colonie...).

La communication chimique est de loin la plus présente chez les fourmis. Les phéromones (mélange d'hydrocarbures) sont à la base de la communication de nombreuses espèces.

La chémoréception présente les avantages suivants [18]:

- la diversité des molécules pouvant intervenir permet de fournir des informations qualitatives
- la stabilité du signal pour une molécule peu volatile permet d'assurer une certaine permanence. Par contre, les principaux inconvénients de la communication chimique sont les suivants:
- elle n'offre que peu d'informations sur la direction ;
- sa propagation est relativement lente et elle est peu adaptée pour la transmission de messages urgents ou pour l'intégration de deux stimulations successives sous une forme temporelle.

La communication entre les individus peut se faire directement ou indirectement L'utilisation des phéromones est majoritairement une forme indirecte puisque l'échange d'information se fait grâce au support du sol. Quand deux individus interagissent indirectement en modifiant l'environnement on parle de stigmergie.

4.2. Les fourmis artificielles

Le terme « fourmi » est un mot qui se profile plusieurs domaines : celui de la biologie ou plus précisément de la MYR écologie qui est l'étude du comportement naturel des fourmis, celui de la robotique qui utilise leur comportement pour concevoir de nouvelle machines, et celui de l'informatique où ces créatures sont modélisées pour la simulation ou la création d'algorithme. Les différentes applications informatiques qui découlent des capacités de communication des fourmis se retrouvent par exemple en optimisation combinatoire où la coopération stigmergique s'applique parfaitement à la recherche du plus court chemin dans un graphe [20].

4.2.1. Les algorithmes de colonies de fourmis

Les algorithmes de colonies de fourmis forment une classe des métaheuristiques récemment proposée pour des problèmes d'optimisation difficiles. Ces algorithmes s'inspirent des comportements collectifs de dépôt et de suivi de piste observés dans les colonies de fourmis. Une colonie d'agents simples (les fourmis) communiquent indirectement via des modifications dynamiques de leur environnement (les pistes de phéromones) et construisent ainsi une solution à un problème en s'appuyant sur leur expérience collective.

4.2.2. Optimisation naturelle : pistes de phéromones

Les algorithmes de colonies de fourmis sont nés à la suite d'une constatation : les insectes sociaux en général, et les colonies de fourmis en particulier, résolvent naturellement des problèmes relativement complexes. Les biologistes ont étudié comment les fourmis arrivent à résoudre collectivement des problèmes trop complexes pour un seul individu, notamment les problèmes de choix lors de l'exploitation de sources de nourriture. Les fourmis ont la particularité d'employer pour communiquer des substances volatiles appelées phéromones. Elles sont très sensibles à ces substances, qu'elles perçoivent grâce à des récepteurs situés dans leurs antennes. Ces substances sont nombreuses et varient selon les espèces. Les fourmis peuvent déposer des phéromones au sol, grâce à une glande située dans leur abdomen, et former ainsi des pistes odorantes, qui pourront être suivies par leurs congénères (figure 3.5) [21].

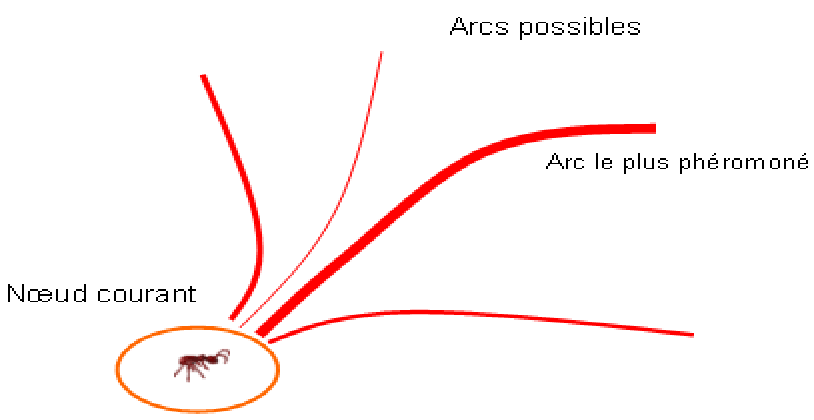
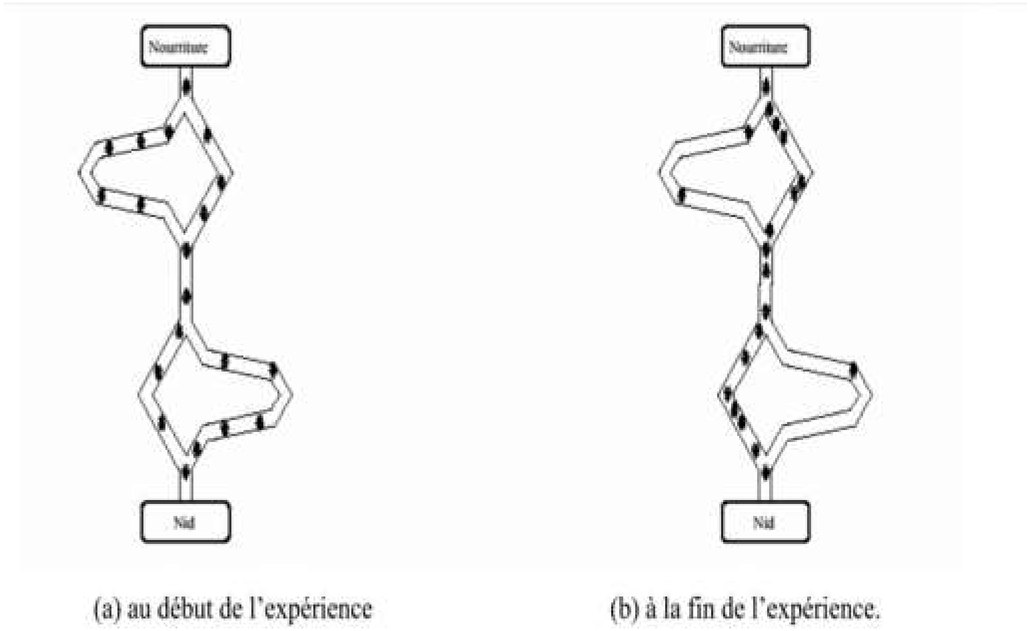


Figure 3.5 : une fourmis suivent une piste de phéromone[36]

Les fourmis utilisent les pistes de phéromones pour marquer leur trajet (entre le nid et une source de nourriture). Une colonie est ainsi capable de choisir (sous certaines conditions) le plus court chemin vers une source à exploiter, sans que les individus aient une vision globale du trajet.



Figue 3.6 : Expérience de sélection des branches les plus courtes par une C.F.[37]

Effet, comme illustré sur la figure 3.6, les fourmis le plus rapidement arrivées au nid, après avoir visité la source de nourriture, sont celles qui empruntent les deux branches les plus courtes. Ainsi, la quantité de phéromone présente sur le plus court trajet est légèrement plus importante que celle présente sur le chemin le plus long. Or, une piste présentant une plus grande concentration en phéromones est plus attirante pour les fourmis, elle a une probabilité plus grande d'être empruntée. La piste courte va alors être plus renforcée que la longue, et, à terme, sera choisie par la grande majorité des fourmis. Mais à tout moment, la probabilité existe qu'un individu quitte la trace puis se déplace plus ou moins au hasard. À cette occasion, l'individu n «égaré» peut éventuellement découvrir une source de nourriture beaucoup plus riche que celle qui exploite ses saveurs. En déposant alors une trace de phéromones plus intense encore, elle va les attirer vers cette nouvelle ressource, formant une nouvelle boucle de rétroaction positive [22].

Nous allons maintenant présenter l'algorithme initial d'Ant Colony System (ACS)[27] s'appelant Ant Colony Cycle, consistant à trouver le cycle hamiltonien le plus court dans un graphe, appliqué au problème de voyageur de commerce (TSP) qui a été proposé pour résoudre le VRP et ces extensions, le problème visant à trouver le cycle hamiltonien le plus court, chaque fourmi est initialement placée sur une ville (sommet de graphe), à l'instant t+ 1 toutes les fourmis choisissent une destination et ce déplacent, ainsi à l'instant n toute la colonie aura réalisé un tour complet sur l'ensemble des sommets du graphe, chacune possède une mémoire qui stocke la solution partielle quelle a construite auparavant.

5.1 Données et Notation

— L'ensemble fini X de sommets représente les villes.

— L'ensemble fini U = {(i,j)/i,j ∈ X} d'arcs ralliant les sommets entre eux.

— L'ensemble dij représentant les poids de arcs(i,j) ∈ U (la distance entre le sommet i et le sommet j).

— La langueur du circuit μ représentant la distance totale parcourue :

$$L(u) = d_{1,p} + \sum_{i=1}^{q-1} d_{i,i+1}$$

— τij(t) la valeur de τij à l'instant t.

— n = |X| le nombre total de ville.

— $\eta ij \ = \frac{1}{d_{ij}}$ la constante indiquant la visibilité de la ville j à partir de la ville i.

Choix de transition

Chaque fourmi possède une mémoire qui sera vidée une fois le cycle terminé lui permettant d'éviter de revenir à une ville déjà visitée. Une fourmi k ce trouvant dans la ville i a l'instant t choisis la ville j comme destination en fonction de deux paramètres : la visibilité ηij et la densité de phéromone τij(t) de l'arc les reliant, suivant la formule :

$$p_{ij}^k(t) = \begin{cases} \dfrac{(\tau_{ij}(t))^\alpha \times \eta_{ij}^\beta}{\sum_{l \in N_i^k(t)} (\tau_{il}(t))^\alpha \times \eta_{il}^\beta} & si\ j \in N_i^k \\ 0 & sinon \end{cases}$$

Où p_{ij}^k est la probabilité que donne la fourmi k à un arc (i,j), N_i^kest l'ensemble des voisins non visités du sommet i par la fourmi k dans le cycle courant, α et β sont des paramètres qui contrôlent l'importance que donne la fourmi a un arc.

Mise à jour des phéromones

A la fin de chaque cycle (chaque fourmi a parcouru les n sommets qui composent le graphe),

les variables des phéromones sont mises à jour selon la formule :

$$\tau_{ij}(t+1) = (1-p) \times \tau_{ij}(t) + \Delta\tau_{ij}(t) \qquad (3.2)$$

Où p est un coefficient tel que $0 < p < 1$

$(1 - p)$ est l'évaporation de la phéromone .

$$\Delta\tau_{ij}(t) = \sum_{k=1}^{n} \Delta\tau_{ij}^k(t) \qquad (3.3)$$

Représente la quantité des phéromones par unité de longueur déposée sur l'arête (i,j) par la k^{eme} fourmi entre t et t + n, elle est donnée par :

$$\Delta\tau_{ij}^k(t) = \begin{cases} \frac{Q}{L_k} & si\ la\ k^{eme}\ fourmi\ \ traverse\ l'arete(i,j) dans\ son\ tour \\ 0 & sinon \end{cases} \qquad (3.4)$$

Avec,

L_k : la longueur du tour de la k^{eme} fourmi.

Q : un nombre positif constant.

$N\ I_{max}$: représente le nombre maximum de tours autorisé, si à partir d'un certain nombre de tours toutes les fourmis font le même trajet on dit que l'algorithme stagne, par conséquent s'interrompt et affiche la meilleure solution mémorisée jusque-là.

Algorithme 4 : Algorithme de colonie de fourmis (ACO classique)

Début

Lecture des paramètres d'exécution : $nv, nf, \alpha, \beta\ , Q, \rho, NI_{max}$

Calcul de la matrice des distances d_{ij}

Initialisation des matrices ηij et τ_{ij}

tant que t = 1 à NI_{max} faire

    Placer les fourmis sur les villes

    pour k = 1 **à n f faire**

        pour v = 1 à **nv faire**

            Nk: l'ensemble des villes non visitées par la fourmi k

            Choisir la plus grande valeur de p_{ij}^{k}selon la formule (3.1)

        fin

        Calculer le coût de la tournée L^k

        Calculer $\Delta\tau_{ij}^{k}$selon la formule (3.4)

        Mettre à jour les traces de phéromone selon la formule (3.2)

    fin

        Calculer L ← min(L^k) on trouve la longueur minimum pour chaque tour de chaque fourmis

fin

Fin

4.6 Le problème de routage véhicule avec capacité CVRP

6.1 Définition

Le problème de routage de véhicule est un problème d'optimisation combinatoire qui consiste à chercher un itinéraire optimal pour une flotte de véhicules, avec une contrainte de capacité, basée en un aux plusieurs dépôts, doit assurer les tournées entre plusieurs clients (plusieurs villes), chacun des clients a demandé une certaine quantité de demandes. Le groupe des clients visité par un même véhicule désigne la tournée commencée et se termine au dépôt et chaque client doit être desservi une seule fois [24].

L'objectif du CVRP est de minimiser le cout total, c-à-d 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.

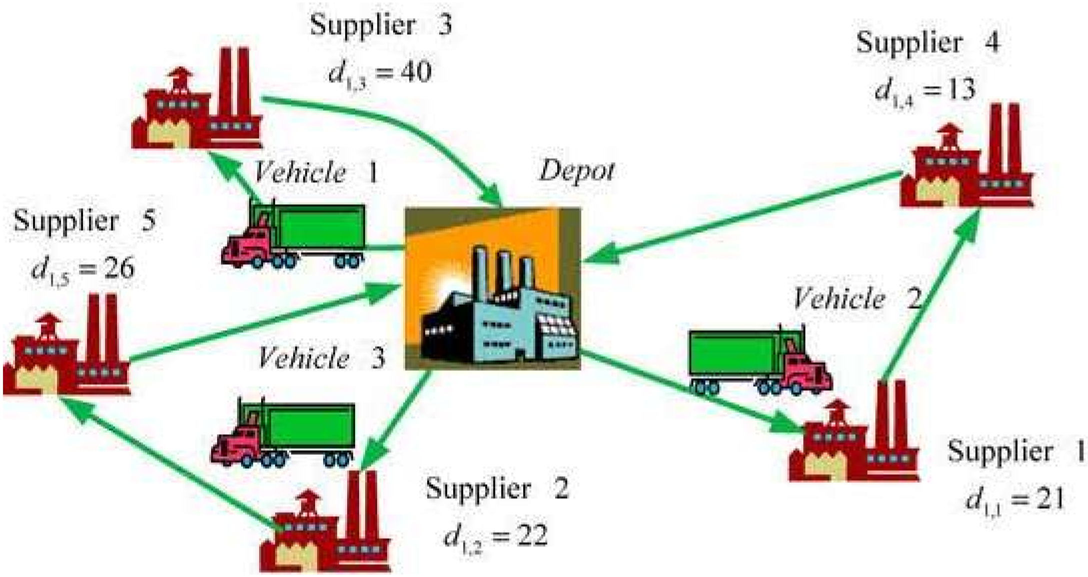


Figure 3.7 : exemple de problème de tournées de véhicules VRP [38]

6.2 Champs d'application

Les applications du VRP sont aussi diverses que variées. Toute entreprise industrielle désire améliorer l'efficacité de sa chaine logistique, pour assurer une production de biens ou de services au moindre coût et une fluidité d'écoulement de sa marchandise En effet, le problème de tournée des véhicules est un maillon principal dans le domaine de la logistique Le problème de tournées des véhicules fait partie de notre quotidien, à commencer par le ramas- sage scolaire, le ramassage du personnel, le ramassage des déchets ménagers, la distribution des produits alimentaires telles que le lait, le pain, l'eau, etc. Aussi, les services et la distribution urgente de produits pharmaceutiques font partie des problèmes de tournées des véhicules.

6.3 Modélisation mathématique du problème routage de véhicule avec capacité CVRP :

La formulation du VRP que nous présentons ici dans sa version la plus basique (CVRP), correspond à la formulation mathématique utilisée en programmation linéaire en nombres entiers. Elle traduit la modélisation naturelle du problème par la définition d'une variable binaire X_{ij}^{k}égale à 1 si le véhicule k parcourt l'arête (i,j). Cette formulation est la plus utilisée dans la littérature, et a ainsi été adoptée par Toth and Vigo[25].

Les Données

La définition la plus générale du VRP est la suivante : il s'agit de la conception de routes optimales par une flotte de véhicules, basée en un ou plusieurs dépôts, pour desservir un ensemble de clients (ou villes) dispersés géographiquement.

Cette définition généraliste met en évidence l'ensemble des paramètres qui caractérisent le VRP.

Le Réseau

Le réseau routier peut être symétrique ou asymétrique ; en conséquence, le graphe associé G =(V;E) sera orienté ou non et les liaisons entre les sommets seront des arcs ou des arêtes. Les tournées peuvent partir d'un seul dépôt ou de plusieurs dépôts. Dans la suite de ce mémoire, nous étudierons le cas ou le réseau est non-orienté et complet.

La clientèle

Les caractéristiques de la clientèle sont définies par le type de variante développées, la variante de base (le VRP classique), la caractéristique de la clientèle est sa position par rapport aux autres et par rapport aux dépôts, pour le CVRP la caractéristique principale est la demande du client.

La flotte des véhicules

Le nombre de véhicules disponibles peut être fixe ou non et peut être associé à un dépôt parti- culier ou pas. Dans le cas du CVRP, la capacité des véhicules peut être homogène ou inégale pour toute la flotte (on parle de flotte hétérogène : les véhicules n'ont pas la même capacité).

Dans notre travail, nous utilisons une seule véhicule.

La fonction objectif

Les objectifs les plus communs sont soit la minimisation du nombre de véhicules utilisés soit la minimisation de la distance totale parcourue par les véhicules. D'autres objectifs peuvent être considérés en fonction de la variante développée (minimisation de la durée totale des tournées, la minimisation du coût total des tournées...)

Notre objectif est de trouver le plus court chemin avec une décharge maximale de la demande.

Paramètre :

n = nv : le nombre de villes.

m : le nombre de véhicules utilisé.

cap : capacité d'un véhicule.

Dem_i : demande du client i.

D_{ij} : le coût de l'arête entre les deux sommets i et j (distance de parcours).

variables de décision

$$X_{ij}^{k} = \begin{cases} 1 & si\ le\ véhicule\ k\ voyage\ du\ noeud\ i\ vers\ le\ noeud\ j \\ 0 & sinon \end{cases} \quad (3.5)$$

La fonction objectif :

$$Min \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{m} d_{ij} \times x_{ij}^{k} \quad (3.6)$$

les contraintes du problème :

$$\sum_{i=1}^{n} \sum_{k=1}^{m} x_{ij}^{k} = 1 \qquad \forall\ 1 \leq\ j \leq n \quad (3.7)$$

$$\sum_{j=1}^{n} \sum_{k=1}^{m} x_{ij}^{k} = 1 \qquad \forall\ 1 \leq i \leq n \quad (3.8)$$

$$\sum_{i=1}^{n} \sum_{l=1}^{n} x_{i,l}^{k} - \sum_{l=1}^{n} \sum_{j=1}^{n} x_{l,j}^{k} = 0 \qquad \forall 1 \leq k \leq m \quad (3.9)$$

$$\sum_{j=1}^{n} x_{0,j}^{k} = 1 \qquad \forall 1 \leq k \leq m \quad (3.10)$$

$$\sum_{i=1}^{n} x_{i,0}^{k} = 1 \qquad \forall 1 \leq k \leq m \quad (3.11)$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} dem_{i} \times x_{i,j}^{k} \leq cap \qquad \forall 1 \leq k \leq m \quad (3.12)$$

(3.6) signifie que l'objectif du problème d'optimisation est de minimiser la distance de toutes les tournées . Les équations (3.7) et (3.8) assurent que chaque nœud n'est servi qu'une seule fois par un et un seul véhicule.

L'équation (3.9) assure la continuité d'une tournée par un véhicule : le nœud visité doit impérativement être quitté.

Les équations (3.10) et (3.11) assurent que chaque tournée commence et se termine au dépôt.

L'équation (3.12) assure le respect de la contrainte de capacité du véhicule.

3.7 Algorithme proposé :

3.7.1 Algorithme de colonies de fourmis avec agrégation

Dans l'algorithme classique, pour trouver le plus court chemin, la fourmi dépendait de la densité de phéromones et la visibilité pour choisir la transition selon la formule (3.1). Par contre, pour trouver le plus court chemin avec une dégradation maximale de la charge de véhicule, nous avons introduit une agrégation des deux objectifs en modifiant la formule (3.1) comme suit :

$$p_{ij}^{k}(t) = \begin{cases} \dfrac{(\tau_{ij}(t))^{\alpha} \times \eta_{ij}^{\beta} \times dem_i^{\gamma}}{\sum_{l \in N_i^k(t)} [(\tau_{il}(t))^{\alpha} \times \eta_{il}^{\beta} \times dem_i^{\gamma}]} & si\ j \epsilon N_i^k \\ 0 & sinon \end{cases} \quad (3.13)$$

- P_{ij}^{k} : la probabilité que donne la fourmi k à un arc (i,j).
- $\alpha\ ,\beta\ ,\gamma$: sont des paramètres qui contrôlent l'importance que donne la fourmi à un arc.
- $\eta_{ij} = \frac{1}{d_{ij}}$: la valeur de la visibilité.
- $\tau_{ij(t)}$: la densité de phéromone.
- dem_i: la demande de chaque ville.

3.7.2 Amélioration de l'algorithme de colonies de fourmis

Pour améliorer l'algorithme des colonies de fourmis, nous avons ajouté une sélection par roulette, cette sélection est appliquée juste après les calculs des probabilités.

La sélection par Roulette

La sélection par roulette est la technique de sélection commune et la plus simple. Elle vise à sélectionner un candidat particulier avec la probabilité d'être sélectionné est p_{ij}^{k} définie dans la formule (3.13).

Lorsque les fourmis choisissent le chemin pour la première fois, elles ne connaissent pas les informations sur les phéromones dans chaque chemin. Cette sélection permet aux fourmis de choisir le meilleur chemin de visite potentiel. Le calcul de la sélection par roulette est basé sur la valeur heuristique ηij. Plus la valeur de ηij est élevée, plus la probabilité d'être choisie est élevée.

La sélection par roulette permet toujours à la fourmis de trouver un autre chemin à partir des résultats de la sélection par roulette.

Exemple

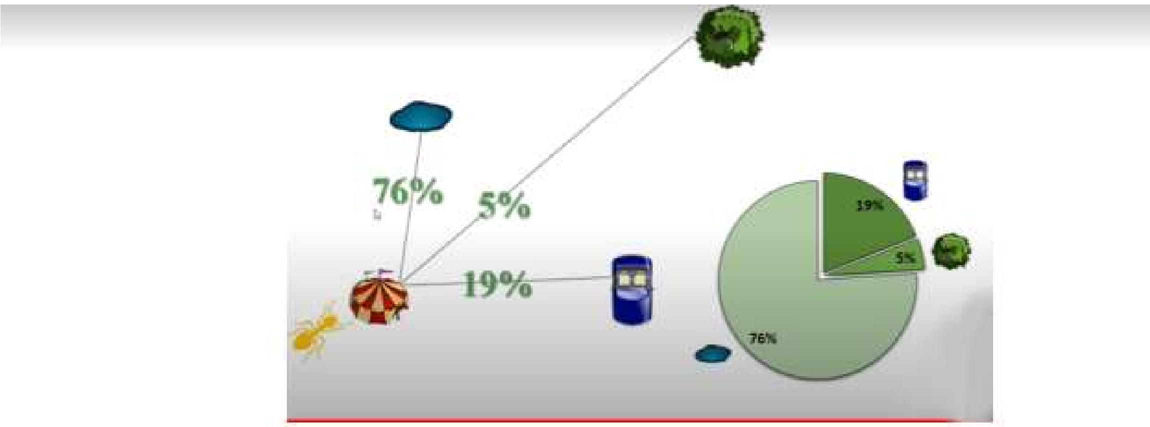


FIGURE 3.4 – la sélection par roulette

la probabilité	0.76	0.19	0.05

somme cumulative	1	0.24	0.05

r est un nombre aléatoire entre [0,1] $\begin{cases} \mathbf{0.24 < r \leq 1} \\ \mathbf{0.05 < r \leq 0.24} \\ \mathbf{0 \leq r \leq 0.05} \end{cases}$

On tire une variable aléatoire

$\mathbf{r = 0.12}$

$\mathbf{0.05 < r \leq 0.24}$

donc on choisit la ville dont la somme cumulée est égale à **0.24**

Algorithme 1: Sélection par roulette

1 Fonction

2 r est un nombre aléatoire entre [0, 1];

3 calculer la somme cumulative de la probabilité ;

4 trouver la première valeur cumulée de la probabilité supérieure à **r;**

5 Fin

3.7.3 Algorithme de colonie de fourmis ACO pour CVRP

L'algorithme ACO modifié suivant, prend en considération l'agrégation des deux objectifs du CVRP à savoir le plus court chemin avec une décharge maximale traduit par la probabilité P selon la formule(3.13) moyennant la technique de roulette selon l'algorithme1.

Algorithme : Algorithme de colonie de fourmis (ACO)

```
Début
Lecture des paramètres d'exécution : nv, n f , dem ,cap, α, β, γ, Q, ρ, N Imax
Calcul de la matrice des distances dij
Initialisation des matrices ηij et τij
tant que t = 1 à N Imax faire
    Placer les fourmis sur les villes
    pour k = 1 à n f faire
        pour v = 1 à nv faire
            N^k: l'ensemble des villes non visitées par la fourmi k
            Calculer la probabilité Pij selon la formule (3.13)
            Sélection par roulette
        fin
        Calculer le coût de la tournée L
        Calculer Δτ ij selon la formule (3.4)
        Mettre à jour les traces de phéromone selon la formule (3.2)
    fin
    Calculer L ← min(Lk) on trouve la longueur minimum pour chaque tour de chaque
    fourmi
fin
Calculer la capacité restante dans le véhicule après chaque décharge
Fin
```

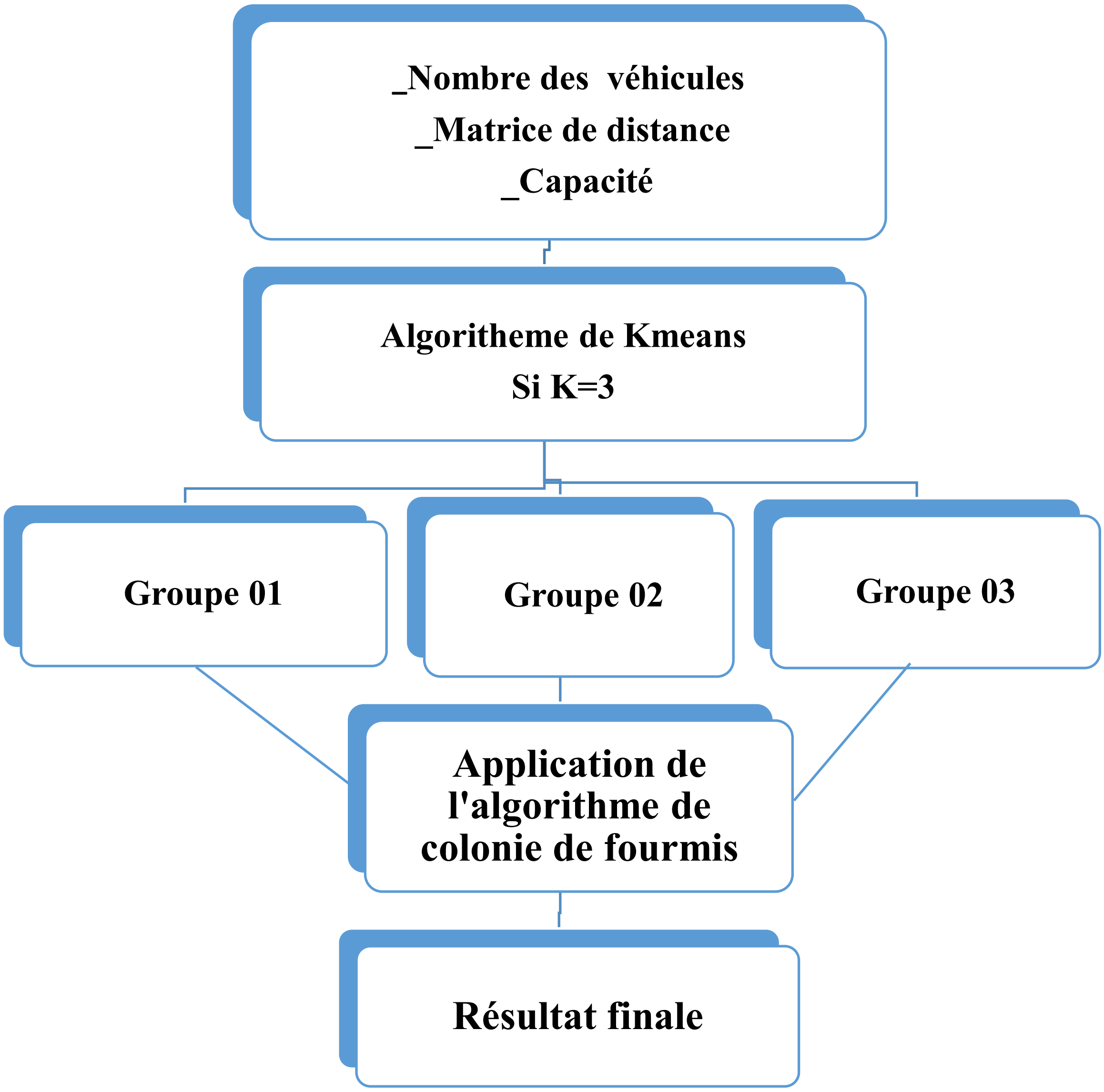


Figure3.8 : Schéma détaillé

Tests et résultats

Dans cette partie, nous présentons les résultats des tests numériques effectués sur l'algorithme ACO (plus court chemin avec dégradation maximale de la capacité) basé sur la technique de la roulette. Nous présentons dans chaque test un tableau contenant :

_ Utilise Algorithme de kmeans pour regrouper les villes .

_ Dépôt D= (456,320)

_Chaque tournée (groupe de ville) effectuée par le véhicule,

— la charge restante dans le véhicule après chaque passage par une ville

— La distance parcourue par le véhicule

— le temps d'exécution de l'algorithme.

De même des schémas illustratifs d'algorithmes ACO sont présentés.

Nous utilisons les paramètres suivants dans tous les tests :

nv	m	α	β	γ	ϱ	ρ	τ_0
nf	1	1	1	1	0.05	1	1

TEST I:

Nombre de ville : 16

Nombre de véhicule : 4

Capacité des véhicules : 2000 .

N Imax :200

Un transporteur doit livrer du fioul à un certain nombre de clients à partir de l'usine.

Le tableau suivant représenté les demandes de chaque ville :

Ville	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16
Demande	**60**	**45**	**50**	**35**	**30**	**40**	**45**	**25**	**70**	**80**	**28**	**56**	**36**	**21**	**73**	**46**

La matrice des distances en kilomètres séparant les villes est donnée comme suit :

\	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16
V1	0	684	308	194	502	730	354	696	742	1084	594	480	674	1016	868	1210
V2	684	0	922	878	502	274	810	468	742	400	1278	1164	1130	788	1552	754
V3	308	992	0	114	650	878	502	844	890	1232	514	628	822	1164	560	1358
V4	194	878	114	0	536	764	388	730	776	1118	400	514	708	1050	670	1244
V5	502	502	560	536	0	228	308	194	240	582	776	622	628	514	1050	708
V6	730	274	878	764	228	0	536	194	468	354	1004	890	856	514	1278	480
V7	354	810	502	388	308	536	0	342	388	730	468	354	320	662	742	856
V8	696	468	844	730	194	194	342	0	274	388	810	696	662	320	1084	514
V9	742	742	890	776	240	468	388	227	0	342	536	422	388	274	810	468
V10	1084	400	1232	1118	582	354	730	388	342	0	878	764	730	388	1152	354
V11	594	1278	514	400	776	1004	468	810	536	878	0	114	308	650	274	844
V12	480	1164	628	541	622	890	354	696	422	764	114	0	194	536	388	730
V13	674	1130	822	708	628	856	320	662	888	730	308	194	0	342	422	536
V14	1016	788	1164	1050	514	514	662	320	274	388	650	536	342	0	764	194
V15	868	1552	560	674	1050	1278	742	1084	810	1152	274	388	422	764	0	798
V16	1210	754	1358	1244	708	480	856	514	468	354	844	730	536	194	798	0

Pour faire ses livraisons , on va classé les villes et on veut déterminer la tournée à réaliser pour livrer tous les clients de façon à minimiser le nombre de kilomètre parcours et décharger de façon maximale .

Après algorithme de kmeans ,on obtient les groupes suivants :

On a 4 véhicule donc :

Groupe01 :v1 ; v3 ; v4 ; v7.

Groupe02 :v2 ; v5 ; v6 ; v8.

Groupe 03 : v9 ; v10 ; v14 ; v16.

Groupe04 :v11 ; v12 ; v13 ; v15.

Alors on appliquer ACO pour chaque groupe :

Test 01 :

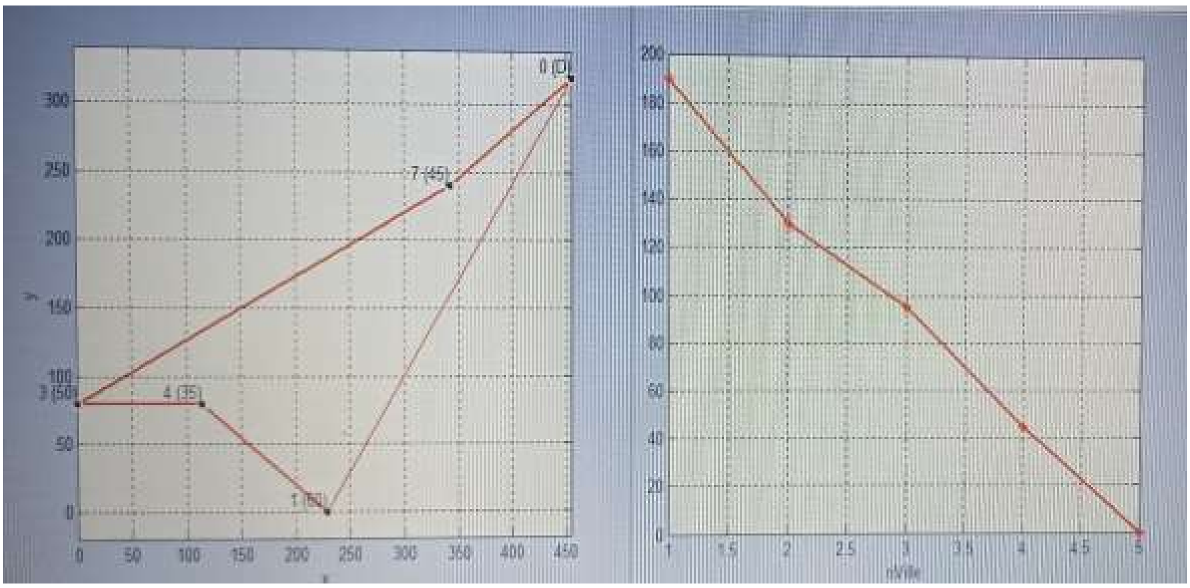


Figure 3.9 : test pour véhicule 01 par ACO.

Résultat :

NV	Tournée\ décharge	longueur	Temps d'exécution
5v	D – 1 – 4 – 3 – 7 – D 190 - 130 - 95 - 45 - 0	1163.0328 km	31.879264 seconds

Test 02 :

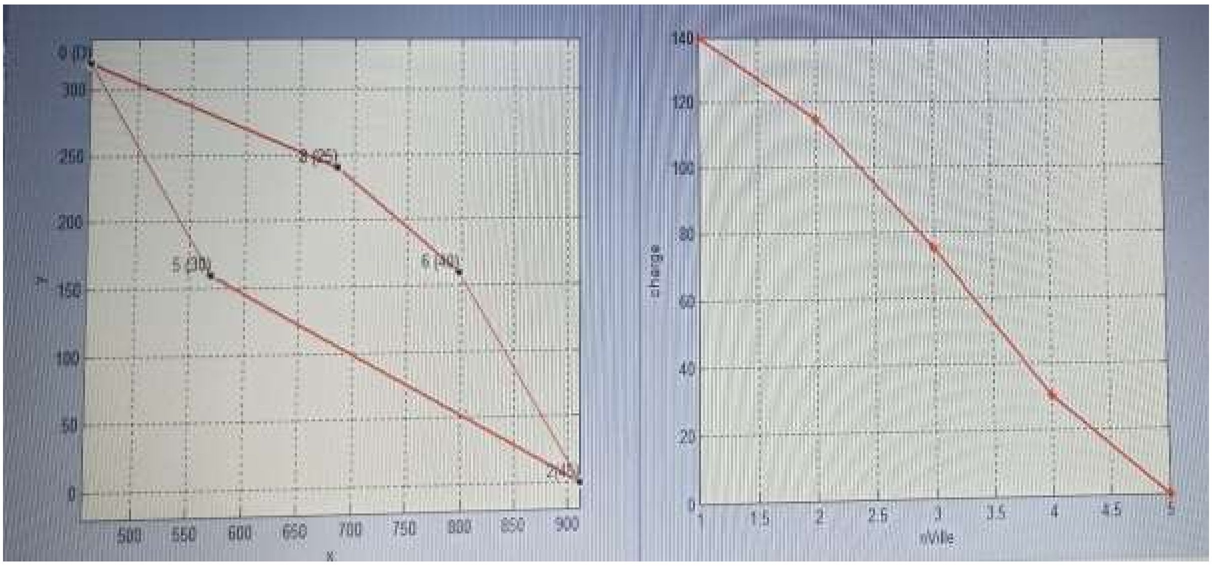


Figure3.10 : test pour véhicule 02 par ACO.

Résultat :

NV	Tournée \ décharge	longueur	Temps d'exécution
5v	D – 5 – 2 – 6 – 8 – D 140 – 115 – 75 – 30 – 0	1151.3911 km	32.089998 seconds

Test 03 :

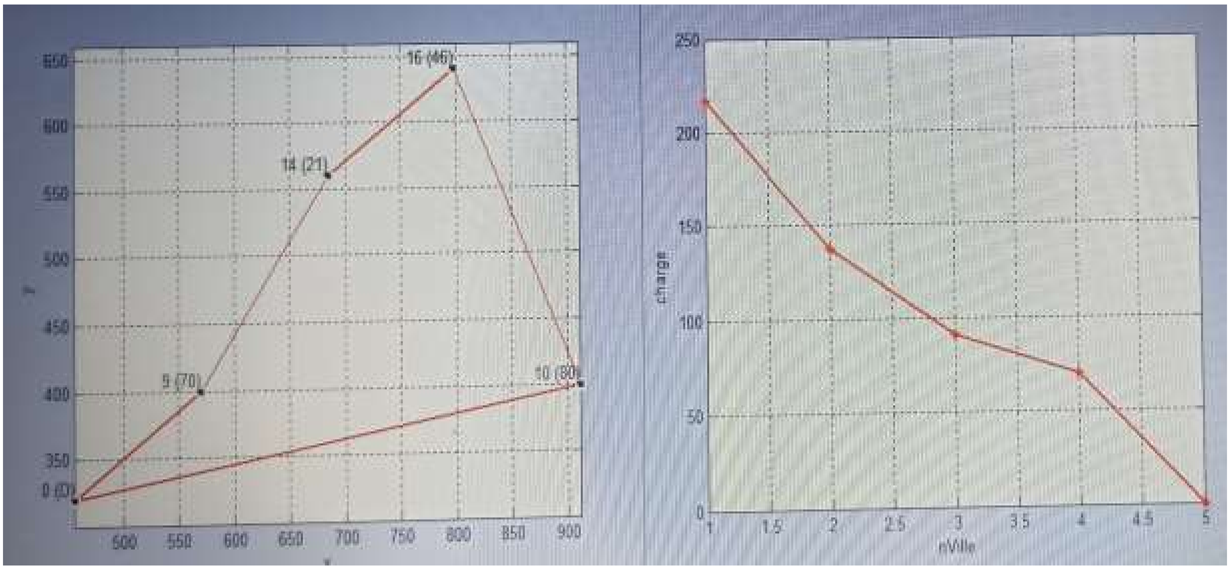


Figure3.11 :test pour véhicule 03 par ACO.

Résultat :

NV	Tournée \ décharge	longueur	Temps d'exécution
5v	**D – 10 – 16 – 14 – 9 – D** **217 – 137 – 91 – 70 – 0**	**1203.6611 km**	**31.287007 seconds**

Test 04 :

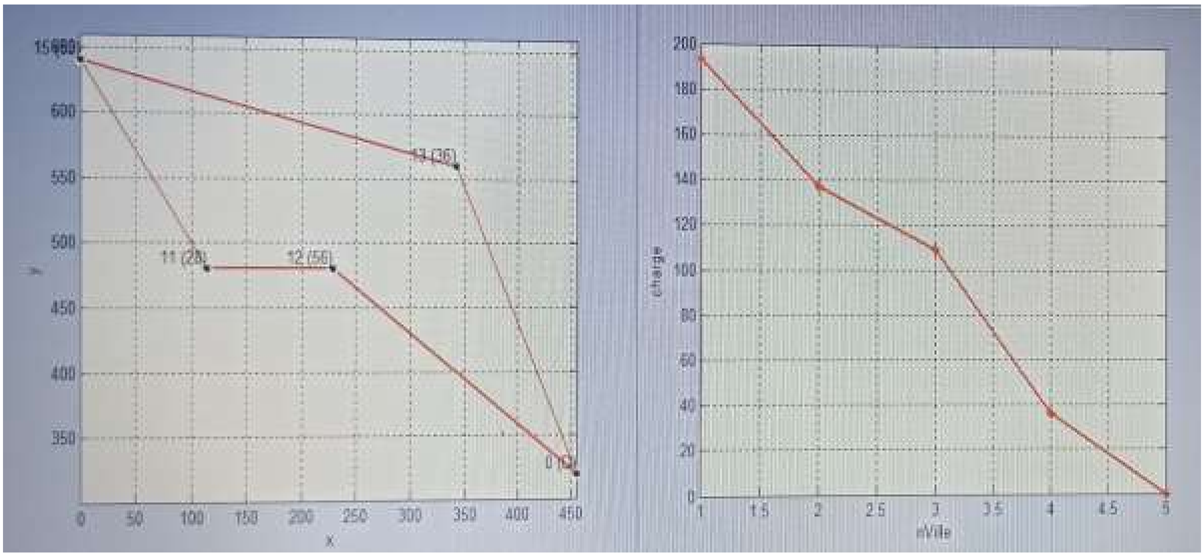


Figure 3.12 : test pour véhicule 04 par ACO.

Résultat :

NV	Tournée \ décharge	longueur	Temps d'exécution
5v	D – 12 – 11 – 15 – 13 – D 193 – 137 – 109 – 36 – 0	1205.9289 km	32.039920 seconds

TEST II :

nombre de ville : 16

nombre de véhicule : 3

capacité des véhicules : 2000 .

N Imax : 400

Un transporteur doit livrer du fioul à un certain nombre de clients à partir de l'usine.

Le tableau suivant représenté les demandes de chaque ville :

Ville	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16
Demande	60	45	50	35	30	40	45	25	70	80	28	56	36	21	73	46

Pour faire ses livraisons ,on va classé les villes et on veut déterminer la tournée à réaliser pour livrer tous les clients de façon à minimiser le nombre de kilomètre parcours et décharger de façon maximale.

Après algorithme de kmeans ,on obtient les groupes suivants :

On a 3 véhicules donc :

Groupe01 : v1 ; v3 ; v4 ; v7.

Groupe02 : v2 ; v5 ; v6 ; v8 ; v9 ; v10 ; v14 ; v16.

Groupe 03 : v11 ; v12 ; v13 ; v15 .

Alors on appliquer ACO pour chaque groupe :

Test 01 :

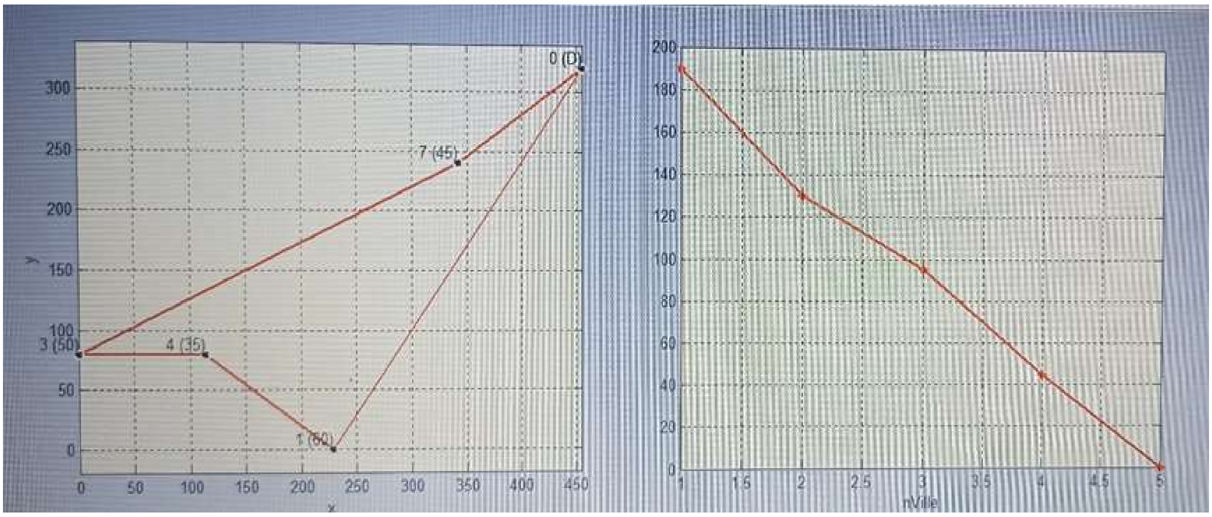


Figure 3.13 : test pour véhicule 01 par ACO.

Résultat :

NV	Tournée\ décharge	Longueur	Temps d'exécution
5v	D – 1 – 4 – 3 – 7 – D 190 - 130 - 95 - 45 - 0	1163.0328 km	31.879264 seconds

Test 02 :

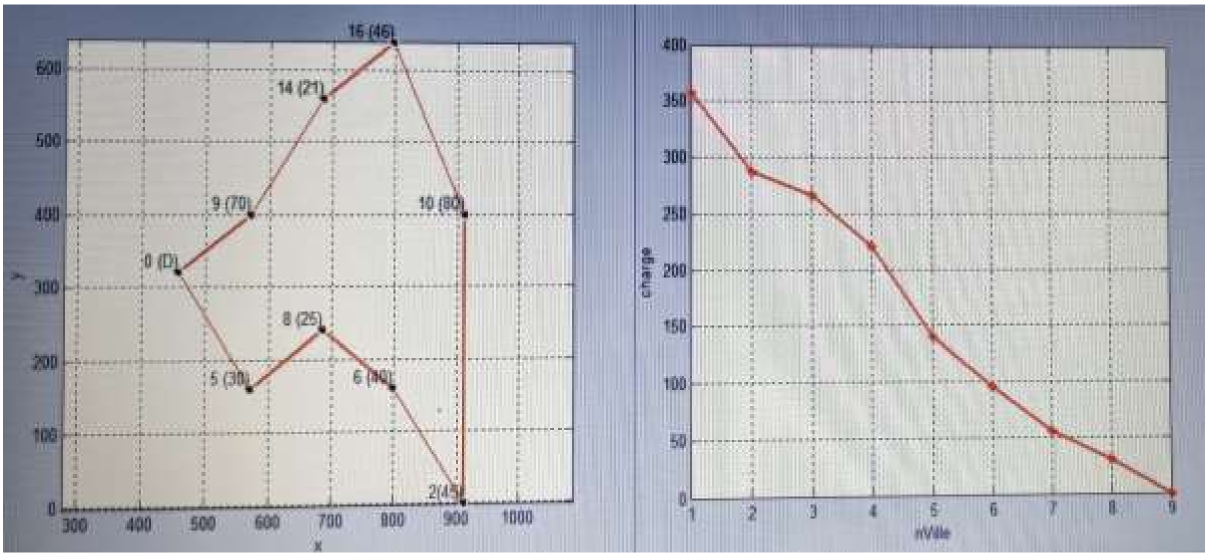


Figure 3.14 : test pour véhicule 02 par ACO.

Résultat :

NV	Tournée\ décharge	longueur	Temps d'exécution
9v	D – 9 – 14 – 16 – 10 – 2 – 6 – 8 – 5 – D 357 – 287 – 266 – 220 – 140 – 55 – 30 – 0	1812.1531 km	32.547862 seconds

Test03 :

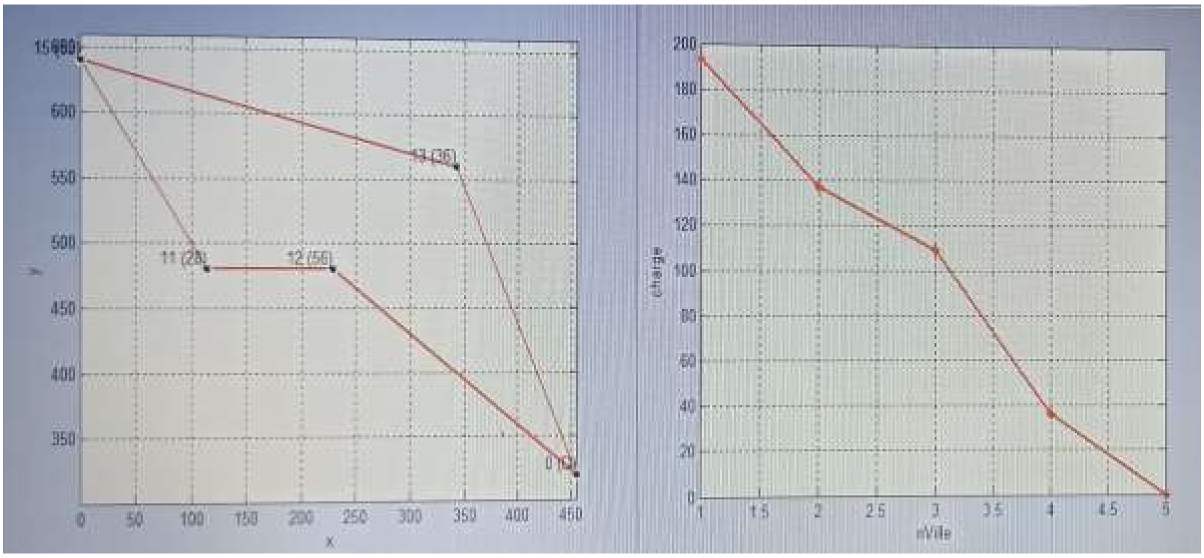


Figure 3.15: test pour véhicule 03 par ACO.

Résultat :

NV	Tournée \ décharge	longueur	Temps d'exécution
5v	**D – 12 – 11 – 15 – 13 – D** **193 – 137 – 109 – 36 – 0**	**1205.9289** **km**	**32.039920 seconds**

TEST III :

nombre de ville : 16

nombre de véhicule : 2

capacité des véhicules : 2000 .

N Imax : 200

Un transporteur doit livrer du fioul à un certain nombre de clients à partir de l'usine.

Le tableau suivant représenté les demandes de chaque ville :

Ville	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16
Demande	60	45	50	35	30	40	45	25	70	80	28	56	36	21	73	46

Pour faire ses livraisons ,on va classé les villes et on veut déterminer la tournée à réaliser pour livrer tous les clients de façon à minimiser le nombre de kilomètre parcours et décharger de façon maximale.

Après algorithme de kmeans ,on obtient les groupes suivants :

On a 2 véhicules donc :

Groupe01 : v1 ; v2 ; v3 ; v4 ; v5 ; v6 ; v7 ; v8 ; v9 ; v10.

Groupe02 : v11 ; v12 ; v13 ; v14 ; v15 ; v16.

Alors on appliquer ACO pour chaque groupe.

Test 01 :

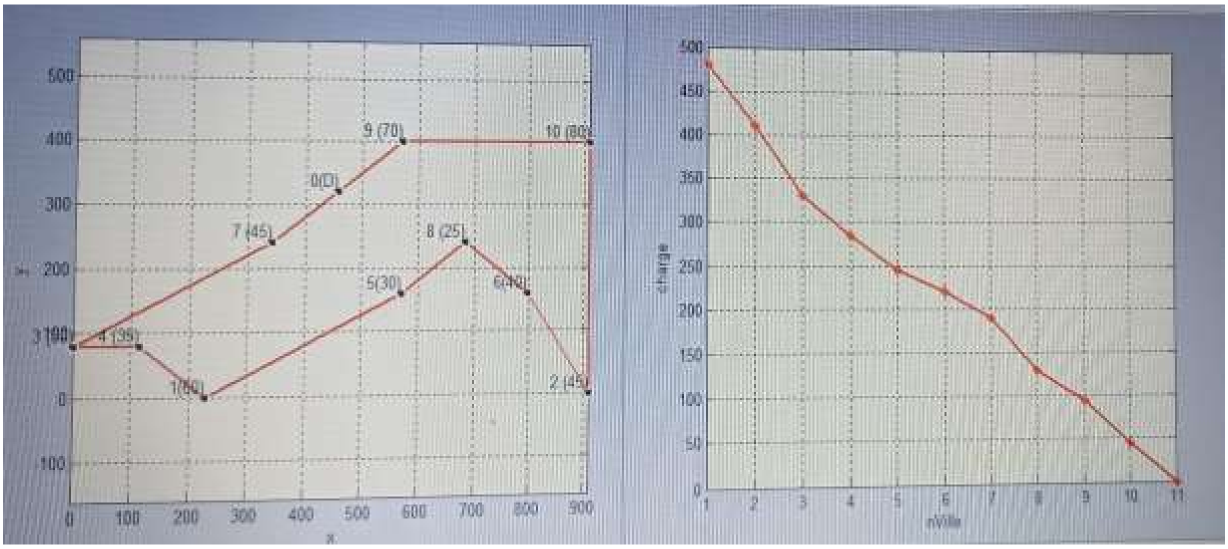


Figure 3.16: test pour véhicule 01 par ACO.

Résultat :

NV	Tournée\ décharge	longueur	Temps d'exécution
11v	**D – 9 – 10 – 2 – 6 – 8 – 5 – 1 –4 – 3 – 7 – D** **480 – 410 – 330 – 285 – 245 – 220 – 130 – 95 – 45 – 0**	**2503.9592 km**	**33.354101seconds**

Test 02 :

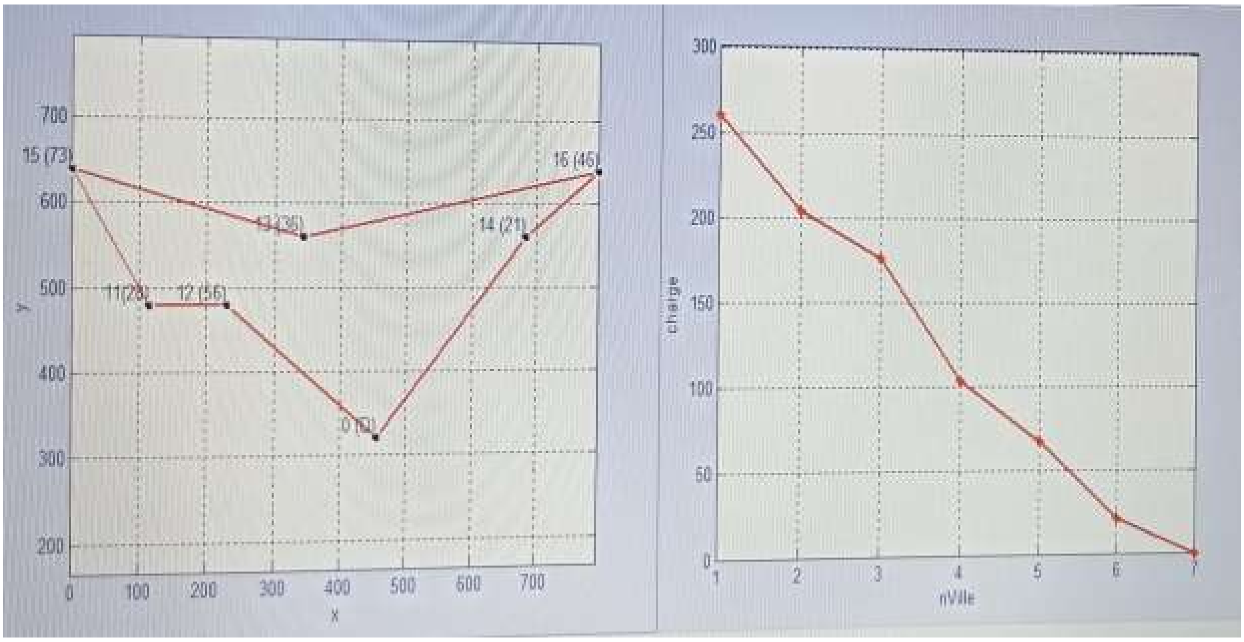


Figure 3.17: test pour véhicule 02 par ACO.

Résultat :

NV	Tournée\ décharge	longueur	Temps d'exécution
7v	D – 12 – 11 – 15 – 13– 16 – 14 – D 260 – 239 – 193 – 157 – 84 – 56 – 0	1873.4984 km	29.163619seconds

Conclusion

Nous avons présenté l'algorithme de colonies de fourmis modifié moyennant la notion de roulette dans le but de sélectionner le plus court chemin toute en dégradant au maximum la charge de véhicule, ce qui permet d'une part de minimiser le temps de la tournée et aillerez le véhicule par les charges les plus pesant. Pour consolider nos propos, nous présentons dans le chapitre suivant des simulations numériques illustratives sur quelques exemples de différente taille.

CONCLUSION GENERALE

Le sujet traité dans ce mémoire concerne le problème de tournées de véhicules avec capacité par l'algorithme colonie de fourmis.

L'objectif du CVRP est de minimiser le cout , la somme des distances et en conséquence des temps de parcours des tournées. en respectant la capacité des véhicules : la quantité de marchandises livrées sur une tournée ne doit pas dépasser la capacité du véhicule qui l'assure avec une dégradation maximale de cette dernière.

Nous avons introduit une agrégation des deux objectifs avec la technique de roulette qui ont joué un rôle très important dans les résultats numériques.

Cette étude nous avons permis d'appliqué l'algorithme de colonie de fourmis sur le problème CVRP avec plusieurs véhicules on combiner ACO avec l'algorithme de kmeans pour le regroupement des villes.

Dans une perspective d'amélioration nous proposons d'utiliser une métaheristique (algorithme de colonie de fourmis ACO) pour avoir meilleur résultat comme le montre les testes numériques .

Ce travail avait pour objectif de se familiariser avec le monde de la recherche et les techniques d'optimisation les plus récentes.

L'état de l'art sur les problèmes de tournées des véhicules que nous avons traité tout au long de ce mémoire nous a permis de mettre en évidence la difficulté du problème : les méthodes de résolutions de ce problème exactes, approchées, heuristiques et méta-heuristiques, généralités sur les problèmes des tournées des véhicules,

Nous considérons que les résultats obtenus dans notre travail sont très encourageants et méritent une étude plus approfondie.

Nous avons beaucoup appris en réalisant ce mémoire : la rigueur, la patience, l'effort, l'organisation.

Nous souhaitons avoir atteint le but que m'a assigné mon encadreur et nous espérons que ce travail trouvera un prolongement futur.

BIBLIOGRAPHIE

- [1] Bräysy and M. Gendreau. Metaheuristics for the vehicle routing problem with time windows. Report STF42 A, 1025, 2001a
- [2] B. Kallehauge, J. Larsen, O.B.G. Madsen, and M. Solomon.Vehicle routing problem with time windows. Column Generation, pages 67_98, 2001
- [3] Jozefowiez. Modélisation et résolution approchées de problèmes de tournée multi-objectif. PhD thesis, Laboratoire d'Informatique Fondamentale de Lille, Université Sciences et Technologies de Lille, Villeneuve d'Ascq, France, December 2004.
- [4] Bräysy. Local search and variable neighborhood search algorithms for the vehicle routing problem with time windows.PhD thesis, University of Vaasa,Finland, 2001.
- [5] Résolution du problème de tournées de véhicules avec collecte et livraison simultanées avec une approche coopérative de métaheuristiques. https://www.researchgate.net/publication/337840545.
- [6] *Président :* Anne-Marie Jolly-Desodt ;Professeur à l'ENSAIT ,une approche génétique pour la résolution du problème VRPTW dynamique.
- [7] Boussaïd et al 2013 c.
- [8] CLERC Et SI ARRY,une nouvelle metaheristique pour l'optimisation difficile la méthode des essaims particuliers, Vol 3-7 France Télécom R Et D : université Paris 12 2004.
- [9] Laptoule K évine ,les algorithmes métaheristiques, Édition springer 1 juin 2006
- [10] RACHid Chelouach ,chapitre 1généralité sur les méthodes d'optimisation, université de France 15 Décembre 1999.
- [11] LAYEB Abdeslem ,les Cergy méthodes de locale ,university Mentouri of Constantine Algeria computer science Département 2009.

- [12] BENSTRIRA_K et Bouarouj,recherche tabau ,rapport de recherche, module recherche opération avancée ,Master 02,2010.
- [13] F ,tangou et p.borne présentation of same metaheristic for the optimization of complex systems ,volume 17 ,No 2 Jun 2008.
- [14] Clerc et siarry 2004 .
- [15] Fermin Alfredo Tang Montane Roberto Dieguez Galvao .A tabu search algorithm for the vehicl routin problem with semultaneous pick-up and delivery service .PHD theis 2006.
- [16] Souier Mahdi ,Hassam Ahmed ,Sari Zaki , les algorithmes base sur les principes des metaheristique pour les ordonnancement temps réel dans un Fms avec flexibilité de routage.
- [17] SOUAR HMID.résolution du problème de multi-flot compatible à coutminimal par l'approche génération de colonnes, Mémoire de Magister, Université des sciences et la technologie d'Oran ; Mohamed Boudiaf 2016.
- [18] Hölldobler.B.and Wilson,E."the Ants ,springer verlage ",Berlin, Germany,1990.
- [19] Hölldobler,B.and.wilson, E."Voyage chez les fourmis", seuil,1996.
- [20] Bonabeau, E.and.Theraulaz,G" Intelligence collective ". Hermès .1994.
- [21] Brossut; Phéromones ;la communication chimique .chez les animaux ;CNRS éditions .
- [22]M. Dorigo ,"optimization les learning,ans natural algorithms",ph.D.dissertatio .Italien,Dipartimentodi Elettronica ,Politecnico dimilano;Italy;1992.
- [23] A.K.Jainet R.C.Dubes .Algorithms for clustering data . Éditions Prentice. Hall Advanced Référence .séries .1988.

- [24] C.NGUYEN, Modélisation d'un problème d'affectation pour la production d'un plan de semence et résolution par une approche de type programmation linéaire mixte , Mémoire de master 2ème année Odile.september 2009.
- [25] M. Dorigo, V. Maniezzo, et A. Colorni, Ant system : optimization by a colony of cooperating agents, IEEE Transactions on Systems, Man, and Cybernetics–Part B, 26(1), 1996.
- [26] Dorigo M., Caro G. D., « The Ant Colony Optimization Meta-Heuristic », in D Corne, M. Dorigo, F. Glover (eds), New Ideas in Optimization, McGraw-Hill, London, pp. 11-32, 1999.
- [27] D.ViGOP.TOTH, The vehicle Routing problem,société for Industrial Mathématics (SIAM) Monographs an Discrète Mathématics and applications philadelphia,2001.
- [28]Vehicle Routing Problem,developers.google.com.
- [29] Dahl et al, 1995.
- [30]PDF Résolution du problème de tournées de véhicules avec collecte et livraison simultanées avec une approche coopérative de métaheuristiques.
- [31] www.researchgate . net
- [32] www.Cloud school.org
- [33] Clustring de textes par la recherche tabou et recuit
- [34] Flowchart of PSO Algorithm-Download Scientific Diagram...
- [35] Des fourmis suivante une piste des phéromone
- [36] Résolution du problème du voyageur des commerce
- [37] Expérience de sélection des branches les plus courtes par C.F.
- [38] An Enhanced Approch for the Multiple vehicles Routing problem with Hetero geneaaus vehicles and a soft time window.

- [39] Mémoire : Algorithme de colonie de fourmis pour l'optimisation des chaines logistiques problème (CVRP)2019/2020.
- [40]Christofides. 1979, Lenstra,1981.
- [41]T.K.Raplhs, 2002.
- [42]Laporte 1999.
- [43] Hao et al.1999, Gendreau and Potvin,2005.
- [44] G. B Dantzig et Ramser. A branch-and-price algorithm for a simultaneous pickup and delivery problem. working paper. *The truck dispatching problem. Management Science*,1959.
- [45] B.J. Ross Ombuki B.M. et F. Hanshar. *A Multi-Objective Genetic Algorithm Approach to the Vehicle Routing Problem with Time Windows*. PhD thesis, 2006
- [46] D. Vigo Cordeau J.F, G. Laporte et M.W.P. Savelsbergh. *Vehicle Routing Handbooks in Operations Research and Management Science*. PhD thesis, 2005.
- [47] Mingozzi A. *The multi-depot periodic vehicle routing problem*. PhD thesis, 2005.
- [48] Archetti C. A. Hertz et M.G. Speranza. *A Tabu Search Algorithm for the Split Delivery Vehicle Routing Problem. Mathematics of Information Technology and Complex Systems,*. PhD thesis,2002.
- [49] Bianchi L. Notes on dynamic vehicle routing - the state of the art. Septembre2000.
- [50] K.F. Doerner Parragh S. N. et R. F. Hartl. *survey on pickup and delivery models Part I : Transportation between customers and depot.* PhD thesis, 2006.

- [51] Brandao J. A new tabu search algorithm for the vehicle routing problem with backhauls. *European Journal of Operational Research*,173.540 à 555,2006.
- [52] Crispim J. et J. Brandao. Metaheuristics applied to mixed and simultaneous extensions of vehicle routing problems with backhauls. *Journal of the OR Society*, 56 : 1296â1302., novembre 2005.
- [53] A.Hoff . et A. LÃ¸kketangen. Creating lassosolutions for the traveling salesman problem with pickup and delivery by tabu search. *Central European Journal of Operations Research*, 14 :125–140, 2006.

Résumé :

Le problème de tournnées de véhicules est un des problèmes d'optimisation combinatoire les plus connus et les plus difficules .Il s'agit de déterminer les tournées optimales pour une flotte de véhicules afin de servir un ensemble donnéde clients.

Les métaheuristiques sont utilisées pour résoudre les problèmes d'optimisation difficile qui sont des problèmes pour lesquelles aucune méthode exacte n'est capable de résoudre exactement en un temps raisonnable.

Parmis les algorithmes métaheuristiques nous avons utilisé algorithmede colonie de fourmis (ACO).

Dans ce travail, on s'intéresse à la résolution numérique d'un problème de tournée de véhicules avec capacité (CVRP). A cet égard, nous avons suggéré la métaheuristique d'optimisation par colonie de fourmis grâce à leur simplicité et leur efficacité. Cette mémoire est achevé par la réalisation des expérimentations numériques sur quelques exemples pratiques .

Mots clés : Métaheuristique , Colonie de formis , Tournée de véhicule avec capacité .

Abstract :

The vehicle routing problem is one of the best known and most difficult combinatorial optimization problems. This involves determining the optimal routes for a fleet of vehicles to serve a given set of customes.

Métaheuristics are used to solve difficult optimization problems which are problems for which no exact method is able.To solve exactly in a reasonable time .

Among the metaheuristic algorithms we used the ant colony algorithms(ACO).
In this work, we are interested in numerical resolution of a vehicles routing problem with capacity (CVRP).In this regard, we have suggested the ant colony optimization metaheuristic due to their simlicaty and efficiency.

This memory is completed by the realization of digital experiments on some practical examples .

Keywords: Métaheuristic ,Ant colony ,Vehicle routing with capacity.

ملخص:

تعد مشكلة توجيه المركبة واحدة من اشهر مشاكل التحسين التجميعي وأكثرها صعوبة فهي تتضمن تحديد المسارات المثلى لمجموعة من المركبات من اجل خدمة مجموعة معينة من العملاء .

يستخدم الاستدلال التجريبي لحل مشكلات التحسين الصعبة التي لا يمكن لأي طريقة حلها بالضبط في وقت معقول.

من بين خوارزميات الاستدلال التجريبي استخدمنا خوارزمية مستعمرة النمل لحلها (ACO). في هذا العمل نحن مهتمون بالحل العددي لمشكلة تجول المركبات ذات السعة.

وفي هذا الصدد اقترحنا استدلال تجريبي الذي يقوم على التحسين من قبل مستعمرة النمل بفضل بساطتها وكفاءتها التي تقوم على تحسين المركبات ذات السعة المحدودة .

تتضمن هذه المذكرة على اجراء تجارب رقمية على عدد قليل من الأمثلة العلمية.

الكلمات المفتاحية : الاستدلال التجريبي، مستعمرة النمل، جولة مركبة بسعة .