



Mémoire de fin d'étude

Pour l'obtention du diplôme d'ingénieur

Filière : Génie industriel  
Spécialité : Management industriel et logistique

Présenté par : ILLOUL Youcef

Thème

**Optimisation des coûts de collecte du  
lait pour l'entreprise Danone**

Soutenu publiquement, le 13 / 09 / 2020 , devant le jury composé de :

M. SOUIER Mehdi	M.C.A	ESSA.Tlemcen	Président
M. BRAHAMI Mustapha Anwar	M.A.A	ESSA.Tlemcen	Directeur de mémoire
M. MALIKI Fouad	M.C.B	ESSA.Tlemcen	Co- Directeur de mémoire
M. BENNEKROUF Mohammed	M.C.B	ESSA.Tlemcen	Examineur 1
M. MEKAMCHA Khalid	M.C.B	UAB. Tlemcen	Examineur 2

*Je dédie ce modeste travail :*

*A mes chers parents*

*A qui je dois ce que je suis*

*A ceux qui m'ont tout donné sans rien en retour*

*A toute ma famille*

*Que dieu leur préserve longue vie*

*A tous mes amis ainsi qu'à tous ceux qui me sont chers*

*Que ce travail soit le témoignage sincère et affectueux*

*De ma profonde reconnaissance pour tout ce que vous avez Fait pour moi.*

*Youcef*

## **REMERCIEMENT**

*Avant toute personne, je remercie le bon Dieu de nous avoir prêté vie, santé et volonté pour achever ce modeste travail.*

*Je tiens à remercier mes encadrants M. BRAHAMI Mustapha Anwar et M. MALIKI Fouad pour tout le temps qu'ils m'ont consacré, pour leur conseils précieux, pour toute leur aide et leur appui durant la réalisation de ce travail.*

*Je tiens à remercier, mes chers parents pour leur encouragement et soutien.*

*J'adresse également un grand merci à M. Sahnoun M'hammed Maître de Conférences à LINEACT de Rouen, pour son aide et ses conseils précieux tout au long de ce projet.*

*Je tiens à remercier chacun des membres du jury pour nous avoir fait l'honneur d'examiner et d'évaluer notre travail.*

# TABLES DES MATIERES

<b>INTRODUCTION GENERALE .....</b>	<b>7</b>
<b>CHAPITRE I .....</b>	<b>10</b>
<b>1. Introduction .....</b>	<b>11</b>
<b>2. Fiche technique de l'entreprise .....</b>	<b>12</b>
<b>3. Présentation de l'entreprise Danone Djurdjura Algérie .....</b>	<b>13</b>
3.1. Historique.....	13
3.2. Laiterie Danone Djurdjura .....	13
3.3. Situation géographique .....	14
3.4. La mission de l'entreprise DDA .....	15
3.5. Les objectifs de l'entreprise DDA.....	15
<b>4. L'organigramme de l'entreprise DDA.....</b>	<b>15</b>
4.1. Direction générale :.....	17
4.2. Direction Marketing :.....	17
4.3. Direction Supply Chain :.....	17
4.4. Direction Opération : .....	18
4.5. Direction des achats .....	18
4.6. Service Collecte .....	18
4.6.1. Unité Qualité .....	18
4.6.2. Unité Recherche en fourrage et distribution .....	18
4.6.3. Unité logistique .....	19
4.6.4. Missions principale de l'unité logistique.....	19
<b>5. Conclusion.....</b>	<b>21</b>
<b>CHAPITRE II .....</b>	<b>22</b>
<b>1. Introduction .....</b>	<b>23</b>
<b>2. La logistique.....</b>	<b>23</b>

<b>3. La chaîne logistique .....</b>	<b>24</b>
3.1. Gestion de la chaîne logistique .....	27
3.2. Structure générale d'une chaîne logistique .....	27
3.3. Niveaux de décisions dans une chaîne logistique .....	28
<b>4. Le transport et la distribution dans la chaîne logistique .....</b>	<b>29</b>
4.1. Le problème du voyageur de commerce TSP .....	30
4.2. Le problème de tournées de véhicules .....	30
4.3. Modélisation mathématique .....	31
4.3.1. Paramètres .....	31
4.3.2. Variables de décision .....	31
4.3.3. Fonction objectif : .....	31
4.3.4. Contraintes .....	31
4.4. Quelques variantes du problème de tournées de véhicules .....	32
<b>5. Les méthodes de résolution du problème VRP : .....</b>	<b>36</b>
5.1. Les méthodes approchées .....	37
5.1.1. Les heuristiques .....	37
5.1.2. Les métaheuristiques .....	38
5.1.3. Les métaheuristiques à solution unique .....	41
5.1.3.1 Recuit simulé .....	42
5.1.3.2 Recherche tabou .....	43
5.1.4. Les métaheuristiques à population de solutions .....	44
5.1.4.1 Les algorithmes génétiques .....	45
<b>6. Conclusion .....</b>	<b>46</b>
 <b>CHAPITRE III .....</b>	 <b>47</b>
<b>1. Introduction .....</b>	<b>48</b>
<b>2. Description du problème .....</b>	<b>48</b>
<b>3. Modèle mathématique .....</b>	<b>49</b>
3.1. Hypothèses .....	49
3.2. Paramètres .....	50
3.3. Variables de décision .....	50
3.4. Fonction objectif .....	50
3.5. Contraintes .....	51
3.6. Complexité du problème VRPTWDC .....	52

<b>4. Résolution du problème VRPTWDC</b> .....	<b>52</b>
4.1. Solution initiale.....	52
4.1.1. Exemple de déroulement de l'heuristique gloutonne.....	54
4.2. Approche de résolution par Recuit simulé (RS).....	54
4.3. Approche de résolution par recherche tabou.....	57
4.3.1. Liste tabou adoptée.....	58
4.3.2. Principe de fonctionnement de la matrice tabou.....	58
4.4. Approche de résolution par algorithme génétique.....	60
4.4.1. Codage.....	60
4.4.2. Opérateurs génétiques adoptés.....	61
4.4.3. Opérateur de sélection.....	62
4.4.4. Opérateurs de croisement.....	62
4.4.5. Opérateurs de mutation.....	62
4.4.6. Population initiale.....	63
4.4.7. Les étapes de l'AG adopté.....	63
4.5. Amélioration des performances de l'AG.....	64
4.6. Méthode d'amélioration basée sur la recherche locale (AG).....	64
4.7. Méthode d'algorithme génétique hybridé avec la recherche tabou (AG + RT).....	65
4.7.1. Paramètres de l'AG adopté.....	66
<b>5. Expérimentation</b> .....	<b>67</b>
5.1. Jeux de donnée.....	68
5.2. Réglage des paramètres des métaheuristiques utilisées.....	69
5.3. Résultats obtenus.....	69
5.3.1. La validation des résultats par la méthode exacte Branch & Bound :.....	72
5.3.2. Résultats pour les moyennes et grandes instances.....	72
5.3.3. Passage d'une solution gloutonne vers une solution d'une métaheuristique :.....	76
<b>6. Conclusion</b> :.....	<b>76</b>
 <b>CHAPITRE IV</b> .....	 <b>78</b>
<b>1. Introduction</b> .....	<b>79</b>
<b>2. Description du cas d'application étudié</b> .....	<b>79</b>
<b>3. Les données réelles du cas d'application étudié</b> .....	<b>80</b>
3.1. Centres de Collectes (désignation des différents points de collecte).....	80
3.2. Productions moyennes journalières respectivement.....	80

3.3.	Fenêtres de temps des centres de collecte .....	81
3.4.	Temps de chargement moyen .....	81
3.5.	Temps maximale d'une tournée .....	81
3.6.	Matrice des distances .....	82
3.7.	Détails sur les camions et leurs tournées actuelles chez DDA .....	85
<b>4.</b>	<b>Présentation des résultats .....</b>	<b>87</b>
4.1.	La tournée actuelle adoptée par DDA.....	87
4.2.	Les tournées obtenues par chaque métaheuristique .....	87
<b>5.</b>	<b>Synthèse des résultats.....</b>	<b>96</b>
5.1.	Interprétation des résultats .....	97
5.2.	Affichage de la tournée de TS jours paire sur Google Maps .....	98
<b>6.</b>	<b>Conclusion.....</b>	<b>100</b>
	<b>CONCLUSION GENERALE ET PERSPECTIVES.....</b>	<b>101</b>

## TABLES DES FIGURES

Figure 1.1: Situation géographique de Danona Djurdjura Algérie .....	14
Figure 1.2: Organigramme de l'entreprise DANONE DJURDJURA ALGERIE .....	16
Figure 2.1 : Chaîne logistique d'une entreprise de production laitière .....	26
Figure 2.2 : Structure générale d'une chaîne logistique.....	28
Figure 2.3 : Classification de méthodes d'optimisation.....	36
Figure 2.4 : Exemples de solutions voisines dans un problème VRP [NGUEVEU. 2009] .....	40
Figure 2.5 : Algorithme du recuit simulé .....	42
Figure 2.6 : Algorithme de recherche tabou.....	44
Figure 2.7 : Exemples de croisement .....	46
Figure 3.1 : Exemple de solution gloutonne .....	54
Figure 3.2 : : Exemple de solution voisine 1 .....	56
Figure 3.3 : Exemple de solution voisine 2.....	56
Figure 3.4 : Exemple de sauvegarde d'une solution voisine dans la Matrice Tabou.....	59
Figure 3.5 : : Exemple de chromosome.....	61
Figure 3.6 : Solutions instance N°13 .....	73
Figure 3.7 : Solutions instance N°19.....	74
Figure 3.8 Solutions instance N°25.....	75
Figure 3.9 : Passage d'une solution gloutonne vers une solution d'une métaheuristique instance N°1 .....	76
Figure 3.10 : Passage d'une solution gloutonne vers une solution d'une métaheuristique instance N°17 .....	76
Figure 4.1 : Circuits des camions .....	98



## TABLES DES TABLEAUX

Tableau 1.1 : Capacité et couts de transport par camion et par sous-traitant.....	21
Tableau 3.1 : Paramètres de l'AG adopté .....	67
Tableau 3.2 : résultats obtenus pour les différentes instances.....	71
Tableau 4.1 : Productions moyennes journalières des CC de DDA.....	80
Tableau 4.2 : Temps de service au plus tôt des CC de DDA .....	81
Tableau 4.3 : Matrice des distances entres les CC de DDA ( partie 1).....	83
Tableau 4.4 : Matrice des distances entres les CC de DDA ( partie 2).....	84
Tableau 4.5 : Détails sur les camions de DDA et de leurs circuits pour les jours paires .....	85
Tableau 4.6 : Détails sur les camions de DDA et de leurs circuits pour les jours impaires.....	86
Tableau 4.7 : Résultats des 4 métaheuristiques et les améliorations apportées .....	96

## **Introduction générale**

Dans un contexte économique instable et de plus en plus concurrentiel, les entreprises ne se contentent plus d'optimiser leurs systèmes de production, mais visent, désormais, l'optimisation de toute la chaîne logistique, tout en développant une relation privilégiée avec l'ensemble des partenaires. Or, les fournisseurs de matières premières figurent parmi les partenaires clés permettant d'assurer une production de qualité supérieure tout en minimisant les coûts logistiques. Pour ce faire, la conception d'un réseau d'approvisionnement et/ou de distribution efficace, avec une gestion adéquate et à moindre coût, devient nécessaire. Toutefois, cette tâche s'avère complexe en raison des nombreux paramètres et contraintes qui interviennent tout le long de ce processus et que l'entreprise doit prendre en considération avant d'entamer la conception de son réseau logistique.

Cette problématique touche différents secteurs d'activité, notamment le secteur agro-alimentaire en raison de la particularité de ses caractéristiques qui le distinguent des autres industries. Aujourd'hui en Algérie, le secteur agroalimentaire joue un rôle important au sein du tissu industriel, il est le deuxième secteur industriel après la filière énergétique sur les plans de la production et de la valeur ajoutée.

Acteur principal dans le secteur agro-alimentaire en Algérie, Danone Djurdjura Algérie (DDA) filiale du groupe Danone qui fait partie des leaders mondiaux de l'industrie agroalimentaire : numéro 1 mondial des produits laitiers frais, numéro 2 mondial des eaux en bouteille et de la nutrition infantile, et numéro 1 européen de la nutrition médicale. En Algérie l'entreprise DDA est réputée dans la production des produits laitiers et principalement le Yaourt. Ce dernier est fabriqué à base de lait (entier, demi-écrémé ou écrémé), généralement de vache, fermenté par le

développement des seules bactéries lactiques thermophiles qui doivent êtreensemencées simultanément et se trouver vivantes dans le produit fini, il est donc pasteurisé, et souvent enrichi avec du lait en poudre pour avoir ainsi une teneur en protéines et en calcium plus importante que du lait normal.

A travers son investissement en Algérie, la laiterie DDA occupe une place de choix dans le marché algérien de produits laitiers. En plus de la qualité du Yaourt produite, elle offre une très bonne qualité de service à ses clients, et entretient des relations solides avec ses partenaires, notamment avec ses fournisseurs tels que les éleveurs de vaches qui s'avèrent les premiers et seules fournisseurs de lait frais essentiel à la production du Yaourt.

Afin de parvenir à satisfaire la clientèle avec une production continu, le département Collecte de l'entreprise DDA a comme mission de faire la collecte du lait frais en envoyant quotidiennement des camions citerne (Tanker) vers les éleveurs de vaches ou les centres de collecte répartis sur l'ensemble du territoire national. Ces camions sont sous-traités auprès de prestataire de transport logistique. Il est clair que cette politique de sous-traitance en Algérie offre des avantages de réduction des coûts, de rémunération, d'exécution des travaux et d'assurance de la qualité des produits et des services. Cependant, chacun de ces avantages est facturé selon les ressources et les efforts fournis par l'entreprise sous-traitante, ce qui fait qu'un manque d'optimisation des opérations sous-traitées engendre une facturation élevée et un rendement faible.

Malgré les efforts du département collecte, il reste beaucoup à faire en matière d'optimisation des coûts logistiques notamment en ce qui concerne l'élaboration des tournées de collecte de lait frais. A travers le stage effectué au niveau de l'entreprise DDA, nous avons constaté que l'élaboration de cette tournée est réalisée d'une manière intuitive basée uniquement sur l'expérience du personnel.

Le présent travail consiste donc à mener une étude stratégique visant l'amélioration et l'optimisation des tournées de collecte de de lait frais de l'entreprise Danone Djurjura Algérie. Notre démarche est divisée en deux étapes. Dans la première étape, nous

proposons une modélisation rigoureuse du problème de collecte de lait frais. Le modèle proposé constitue une extension du célèbre modèle VRP (Vehicle Routing Problem) auquel nous introduisons deux contraintes temporelles. La première contrainte concerne les fenêtres de temps à l'intérieur desquelles chaque client peut être visité. La deuxième contrainte impose une durée maximale pour chaque route (ou circuit). Pour la résolution du modèle proposé, nous développons dans la deuxième étape du travail quatre (04) approches de résolution basées sur les métaheuristiques suivantes : Recuit simulé, Recherche tabou, Algorithme génétique et une dernière approche hybridant un algorithme génétique avec la recherche tabou.

Ainsi, le premier chapitre traitera la présentation de l'entreprise Danone Djurdjura Algérie (DDA) en exposant notamment son historique, son évolution, son organigramme et ses missions. Un audit logistique ainsi qu'un diagnostic mené au sein du département Collecte de DDA seront également présentés dans ce premier chapitre.

Le deuxième chapitre sera consacré à la présentation du problème VRP et ses principales variantes ainsi qu'une synthèse des méthodes de résolution des problèmes de type VRP. Une attention particulière sera accordée aux méthodes de résolution basées sur les métaheuristiques.

Dans le troisième chapitre, nous décrivons le modèle de tournée de véhicules proposé ainsi que les quatre approches développées pour sa résolution. Nous terminons ce chapitre par la présentation d'un panel d'expériences numériques et analyses, où nous comparons les résultats des différentes métaheuristiques.

Une étude d'une application industrielle, issue de la collecte de lait frais de l'entreprise DDA fera l'objet du quatrième et dernier chapitre. Les résultats obtenus sont présentés afin de montrer la validité et l'efficacité des approches proposées. Ces résultats fournissent des tournées bien meilleures que les tournées actuelles adoptées par DDA.

Enfin, nous terminons ce mémoire par une conclusion générale.

## **Chapitre I**

### **Présentation de la laiterie Danone Djurdjura Algérie (DDA)**

## **1. Introduction**

Durant notre stage de fin d'études au sein de l'entreprise Danone Djurdjura Algérie (située Taharacht commune d'Akbou wilaya de Bejaia) nous avons été affectés au département « Collecte ».

Plus largement, ce stage a été l'opportunité pour nous d'appréhender la logistique et les moyens, outils et politiques utilisés afin de satisfaire l'usine en terme de matière première. Dans le cas de DDA, le lait frais représente la matière première essentielle pour toute production laitière.

Au-delà d'enrichir nos connaissances en Management industriel et logistique, ce stage nous a permis de comprendre comment appliquer nos connaissances acquises au cours de notre formation dans l'organisation logistique d'une entreprise de production laitière.

L'entreprise Danone Djurdjura Algérie (DDA) est spécialisée dans la production des produits laitiers et principalement le Yaourt. Ce dernier est fabriqué à partir du lait (entier, demi-écrémé ou écrémé), généralement de vache, fermenté par le développement des bactéries lactiques thermophiles, il est donc pasteurisé, et le lait collecter est souvent enrichi avec du lait en poudre pour avoir ainsi une teneur en protéines et en calcium plus importante.

A travers son investissement en Algérie, la multinationale Danone avec son partenaire Djurdjura occupe une place de choix dans le marché algérien de produits laitiers. En plus de la qualité de ses produit, l'entreprise DDA offre une très bonne qualité de service à ses clients, et entretient des relations solides avec ses partenaires, notamment avec ses sous-traitants tels que les éleveurs de vaches qui représentent les premiers fournisseurs de lait cru.

## **2. Fiche technique de l'entreprise**

**Raison sociale :** Danone Djurdjura Algérie.

**Siege social :** Tour Abc 8ème étage, Pins Maritimes, Mohammadia, Alger Algérie.

**Zone d'activité :** Zone Industrielle Taharacht, 06001 Akbou, Algérie.

Statue juridique : SPA.

**Capital :** 2 700 000 000DA

**Tél :** +213796 77 73 48

**Fax :** +213 34 358971

**Site Web :** <http://www.danone.com>

### **3. Présentation de l'entreprise Danone Djurdjura Algérie (DDA)**

#### **3.1. Historique**

Danone est une entreprise multinationale qui a fait un partenariat avec Djurdjura en Octobre 2001.

L'origine du groupe Danone remonte à 1966, lors de la fusion de deux sociétés verrières française, glaces de Boussois et verrières Sonchoir Newrsel. Cette fusion a donné naissance à la société Boussois Sonchoir Newrsel (BSN). En 1967, le groupe BSN a réalisé un chiffre d'affaire d'environ 150 million d'euros dans le verre plat et le verre d'emballage pour affirmer son statut de groupe international de l'agroalimentaire et des boissons. Pour renforcer sa notoriété, le groupe BSN a décidé en 1994, de se rebaptiser groupe Danone. En 1997, le groupe a engagé un important programme de recentrage sur trois métiers prioritaires à vocation mondiale (produit laitiers frais, boisson et biscuits, snacks céréaliers) qui représentent 77% du chiffre d'affaire, le groupe DANONE est le premier producteur mondial de produit frais, et d'eau conditionnée.

#### **3.2. Laiterie Danone Djurdjura**

C'est en 1948, que mûrit dans l'esprit du groupe Batouche, l'idée de création d'une petite unité de fabrication de Yaourt avec des moyens très limitée, l'unité n'a démarré qu'avec une remplisseuse de pots préforme d'une capacité de 1000 pots/heure. En 1983, la laiterie algérienne Djurdjura fut créée par la Famille Batouche, cette laiterie a connu une croissance soutenue pendant 18 ans. En octobre 2001, le leader Mondial des produits laitiers frais Groupe Danone a conclu un accord de partenariat avec la laiterie Djurdjura, leader du marché Algérien des produits laitiers frais (PLF) en 2001 dont le capital est estimé à 2.700.000.000 DA. Ce qui a donné naissance à une nouvelle société « Danone Djurdjura Algérie SPA ». L'entreprise poursuit son expansion avec l'acquisition d'une nouvelle usine à Constantine en 2017.



Danone Djurdjura Algérie (DDA) est donc le fruit d'une belle rencontre entre un leader mondial des produits laitiers frais Danone et un acteur important dans le marché algérien des produits laitiers qui est Djurdjura.

### 3.3. Situation géographique

Ci-dessous la carte géographique de la wilaya de Bejaia, la flèche rouge indique l'endroit exact où est située l'entreprise DDA.

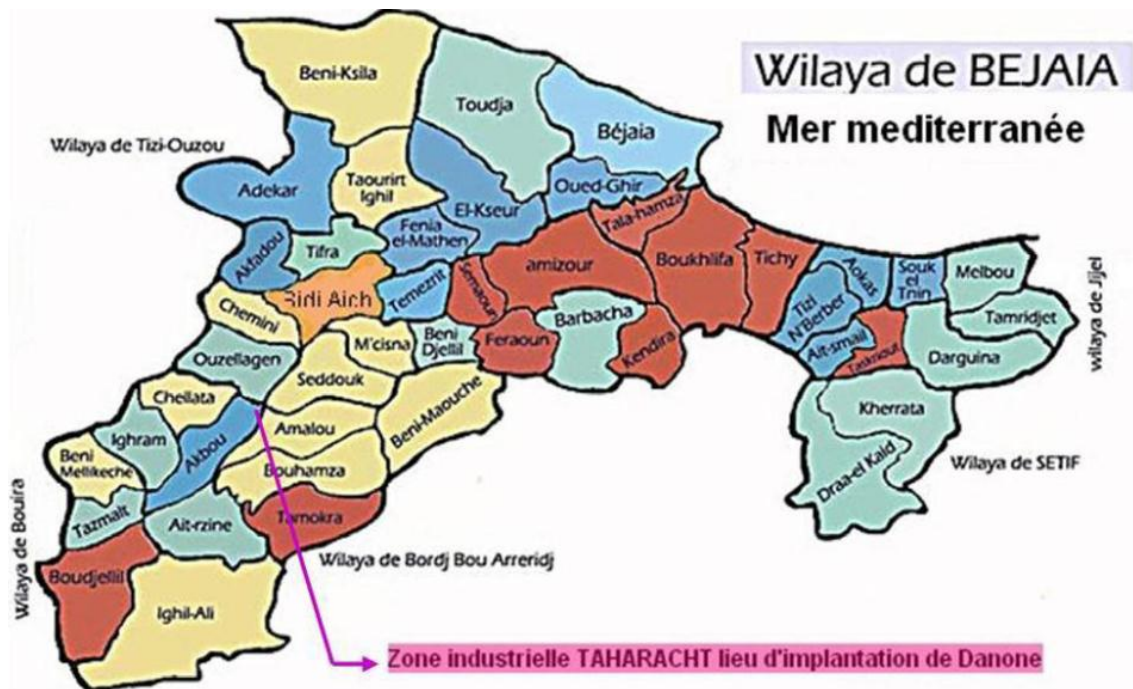


Figure 1.1: Situation géographique de Danona Djurdjura Algérie

Danone Djurdjura Algérie est implantée dans la zone industrielle de « Taharacht », véritable carrefour économique de la wilaya de Bejaia. En effet, cette localisation géographique offre à l'entreprise une panoplie d'avantages dont les principaux sont :

- A deux (02) Km d'une grande agglomération (Akbou).
- A quelques dizaines de mètres de la voie ferrée.
- A 60 Km de la ville de Bejaia, chef-lieu de la région et pôle économique important en Algérie dotée d'un port à fort trafic et un aéroport international reliant divers destinations (Paris, Marseille, Lyon, Saint-Etienne, Charleroi....etc.).

- A 170 Km à l'est de la capitale Alger.
- Présence de plusieurs acteurs économiques importants pouvant constituer de bons partenaires, tels que : All Plast, Soummam, IFRI, etc.

### **3.4. La mission de l'entreprise DDA**

La mission principale de l'entreprise DDA est de satisfaire les besoins du marché en matière de produits laitiers frais. Pour cela, elle s'est positionnée sur l'aspect de qualité et de sécurité alimentaire du produit. Les activités principales de l'entreprise sont principalement la transformation des matières premières en produits laitiers finis, prêt à être utilisé par le consommateur final. Le système de management de sécurité alimentaire assure l'adéquation du produit avec les exigences réglementaires et les attentes du consommateur.

### **3.5. Les objectifs de l'entreprise DDA**

En sa qualité de leader dans son domaine, l'entreprise DDA a comme ultime objectif de maintenir son statut de leadership, à cela s'ajoute d'autres objectifs complémentaires dont :

- Accroître ses parts de marché en volume et en valeur.
- Satisfaire les besoins et attentes des clients en vue de les fidéliser.
- Lancer de nouveaux produits sur le marché.
- Etablir d'autres contrats d'exclusivités avec de nouveaux clients.

## **4. L'organigramme de l'entreprise DDA**

L'organigramme renseigne sur la répartition des tâches et des activités, sur les liaisons hiérarchiques fonctionnelles et de conseil existant. Il n'est qu'une image de la structure à un moment précis de la vie de l'entreprise.

L'organigramme de l'entreprise DDA est composé de sept (07) directions, son organisation est représentée dans le schéma ci-après (Figure 1.2) avec mise en évidence du département « Collecte » dans lequel nous avons effectué notre stage.

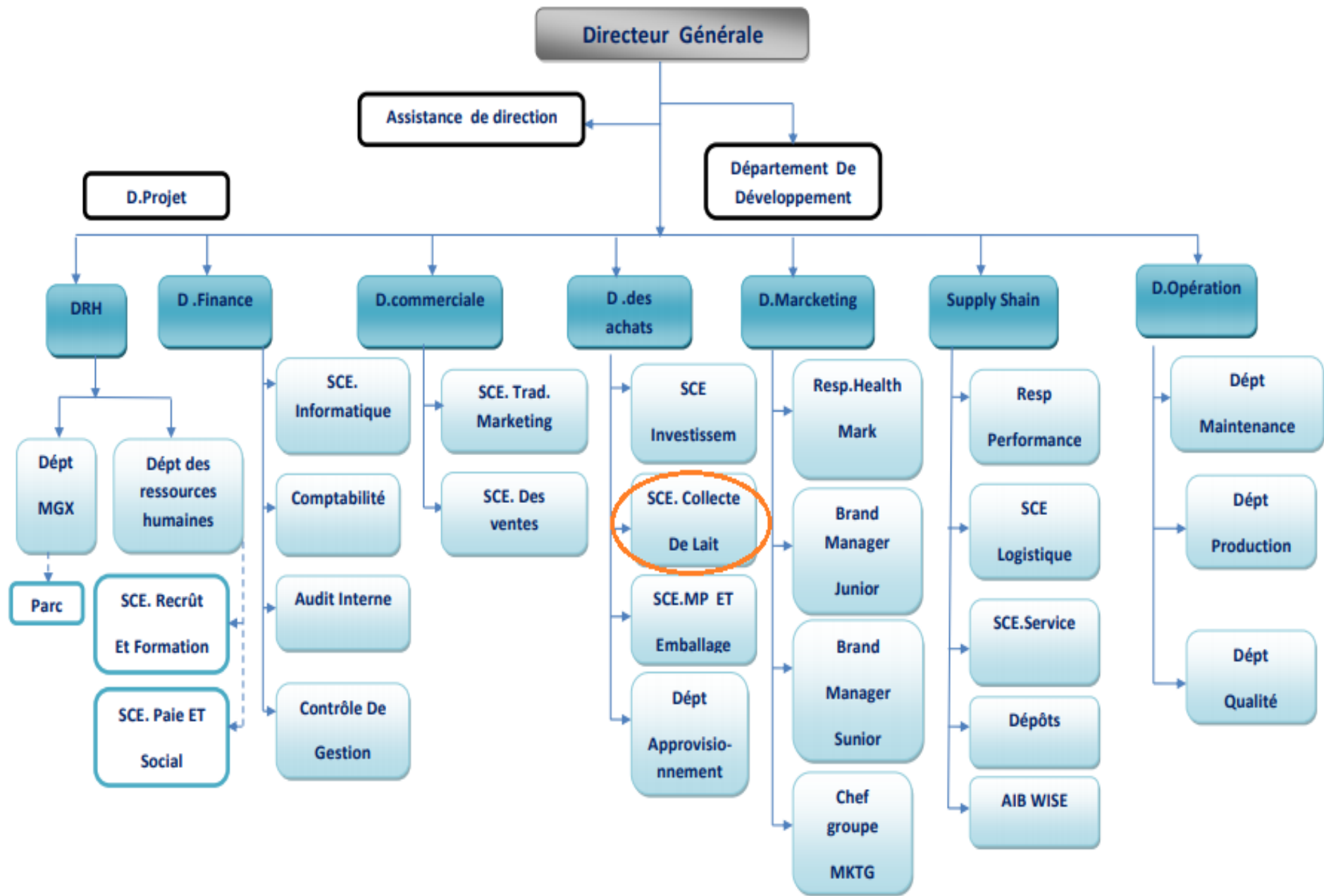


Figure 1.2: Organigramme de l'entreprise DANONE DJURDJURA ALGERIE

Comme nous le constatons dans l'organigramme ci-dessus l'entreprise DDA est composée de la direction générale, d'une assistance de direction, d'autres directions. Nous présentons dans ce qui suit le rôle de quelques directions :

#### **4.1. Direction générale :**

La direction générale assure le rôle de pilotage qui consiste à tracer les grandes orientations et objectifs à atteindre.

#### **4.2. Direction Marketing :**

Elle est composée d'assistants chefs produits. Elle s'occupe du client, sa démarche consiste à collecter les informations et à analyser les besoins des consommateurs afin de les satisfaire tout en respectant les objectifs de l'entreprise. Elle essaye aussi par les moyens publicitaires de faire connaître le produit le plus possible jusqu'à son intégration dans la culture de la population.

#### **4.3. Direction Supply Chain :**

Ses principales fonctions sont la prévision et la planification et la performance :

- a) La prévision : Il s'agit de prévoir les changements selon les événements futurs (promotion, publicité) en coordination avec la communication et marketing.
- b) La planification : Elle consiste à gérer les stocks des matières premières et les achats à l'importation et le dédouanement. Ses principales missions sont :
  - Alimentation des dépôts selon leurs besoins et ventes aux clients directs.
  - Gestion des camions, des palettes, de l'emballage et des transferts.
- c) La performance : Elle évalue la performance logistique suivant des indicateurs de rapprochement entre tous les rapports de la Supply Chain (Taux de vente, disponibilité produit dans les dépôts, mesurer les pertes) et la coordination entre la production et le commercial.

#### **4.4. Direction Opération :**

La laiterie DDA a une capacité de production estimée de 350 à 400 tonnes/jour. Ses différents produits sont :

- Yaourt ferme (Yaoumi)
- Activia
- Mini Prix
- Cremix
- Danino
- Crème dessert (DANETTE)
- Yaourt à boire (Dan'up)
- Jus (Danao)

#### **4.5. Direction des achats**

Elle est composée de : service Matière Première, département Approvisionnement, service investissement et service *Collecte de lait* . Elle assure la protection du marché et gère tous les achats d'entreprise en matière, matière consommable, emballage, pièces de recharge et la collecte du lait.

#### **4.6. Service Collecte**

Nous trouvons dans le service Collecte les unités suivantes :

##### **4.6.1. Unité Qualité**

Responsable de la qualité du lait cru. Lorsque le lait est arrivé à l'usine, il doit être examiné avant le dépotage (mesure de densité et test antibiotique), si tout est dans les normes, on donne l'ordre de dépoter le lait aux tanks.

##### **4.6.2. Unité Recherche en fourrage et distribution**

Cette unité est chargée de faire des recherches en ce qui concerne l'alimentation des vaches et la gestion de la distribution du fourrage aux éleveurs. Cette unité chapote le projet « **H'Lib Dzair** » qui a pour but d'augmenter et de développer les fermes algériennes en taille, volume et espèces. C'est un projet très ambitieux dans lequel Danone a investi une somme de 4.9 M€.

#### **4.6.3. Quatre unités de négociations et supervisions des centres de collectes (CC) :**

*Unité Zone Est 1* : Prend en charge le CC situés dans le Nord-Est algérien.

*Unité Zone Est 2* : Prend en charge les CC et les fermes situés dans le Sud-Est algérien.

*Unité Zone Centre* : Prend en charge les CC situés au centre du pays.

*Unité Zone Ouest* : Prend en charge les CC et les fermes situés dans l'Ouest algérien.

#### **4.6.4. Unité logistique**

C'est l'unité dans laquelle nous avons effectué notre stage. Elle est l'unité la plus importante du département. Outre sa mission de collecte de lait frais à travers les 48 wilayas du territoire national, l'unité « Collecte de lait » a pour objectif :

- Assurer la bonne qualité de lait en référence aux normes exigées par Danone.
- Faire des accords avec de nouveaux éleveurs de vaches et chercher à augmenter le réseau et la capacité des centres de collecte et des maisons d'éleveurs.
- Superviser et surveiller les fournisseurs (centres et maisons d'éleveurs) en dépit de la quantité de lait collectée et s'il y a un besoin en fourrage ou de maintenance de cuves et/ou des groupes électrogènes.
- La communication permanente avec les sous-traitants de la logistique (TMF, Fleche Bleu et MOUSSA) et de maintenance (EchoClim) sur les coûts de transport et la maintenance des cuves et des groupes électrogène situées dans les centres de collecte (CC).

#### **4.6.5. Missions principales de l'unité logistique**

##### *1) La collecte du lait auprès des centres de collecte (CC) et les maisons d'éleveurs (MDE) :*

La collecte se fait chaque jour en envoyant des camions aux centres de collecte (CC) et aux maisons d'éleveurs (MDE). Nous précisons que les centres de collecte sont des centres intermédiaires qui regroupent les productions des éleveurs proches du centre. Tandis que les MDE sont des grands éleveurs de vaches qui produisent une grande quantité de lait cru.

Pour s'assurer que le lait soit de bonne qualité, l'unité a donné des instructions aux chauffeurs afin de faire des tests (Alcool, densité) avant de remplir les citernes des camions de collecte. S'il y a un doute où un problème au niveau des CC, MDE, camion ou personnel, l'unité sera informé et tentera de

trouver une solution dans les plus brefs délais, soit en contactant le chauffeur de camion ou le CC/MDE.

Il est à noter que l'élaboration des tournées de camions pour la collecte de lait est organisée d'une manière intuitive suivant des circuits permettant d'atteindre le remplissage maximal des camions.

Notre objectif principal à travers ce projet de fin d'étude est justement d'améliorer et optimiser ces tournées de camions afin de diminuer la distance totale parcourue et ainsi contribuer à réduire les coûts d'approvisionnement en lait frais de l'entreprise DDA. Cette optimisation sera réalisée moyennant une modélisation rigoureuse et une résolution assez fine du problème en utilisant les métaheuristiques qui présentent actuellement des alternatives intéressantes pour la résolution des problèmes d'optimisation difficile comme les problèmes de tournée de véhicules.

### 2) *Négociation et facturation avec les sous-traitants (TMF, FB et MOUSSA) :*

Le coût de transport est le facteur le plus important dans toutes les entreprises, ainsi, le coût du litre de lait joue un rôle important dans l'unité, le minimiser au maximum à chaque reprise est la mission principale, donc, on calcule la facturation du transport en fonction du lait collecté presque hebdomadairement.

### 3) Livraison des cuves et des groupes électrogènes :

Si une ferme ou bien un éleveur accepte de travailler avec Danone, l'unité se charge de lui offrir comme service des cuves (sa capacité en dépend de la quantité de lait produite quotidiennement) et un groupe électrogène (en cas de coupure d'électricité) pour assurer le bon conditionnement et la conservation de la qualité du lait.

### 4) La gestion de camions :

Chaque jour, 10 camions quittent le parking de l'entreprise DDA et se dirigent vers les CC pour la collecte, chaque chauffeur prend la fiche de route du camion qui sert à la traçabilité du lait collecté et à la bonne vérification et surveillance des CC. Quotidiennement, un programme de collecte est établi pour chaque camion en fonction de sa capacité et de la quantité du lait disponible dans les CC (voir Tableau 1.1).

SOUS-TRAITANT	Coût de transport	NUMERO	CAPACITE
TMF	48 DA/Km	C24	24 000
		C14	24 000
		C18	18 750
		C21	22 000
		C23	22 000
FB	42,14 DA/Km	C12	22 000
		C15	22 000
		C26	25 000
		C27	25 000
		C28	25 000
		C13	22 000
MOUSSA	42,14 DA/Km	C30	18 000

*Tableau 1.1 : Capacité et coûts de transport par camion et par sous-traitant*

## 5. Conclusion

Dans ce premier chapitre, nous avons présenté l'entreprise Danone Djurdjura Algérie (DDA) où notre stage de fin d'études a été effectué. Après avoir présenté le groupe Danone et le contexte dans lequel s'est effectuée la fusion avec Djurdjura, nous avons exposé les missions et les objectifs de l'entreprise DDA. Nous avons ensuite détaillé l'organigramme de l'entreprise en s'intéressant particulièrement à l'unité « Collecte de lait » dans laquelle s'est déroulé notre stage.

La mission de l'unité « Collecte de lait » consiste principalement à élaborer pour chaque camion les tournées de collecte du lait frais auprès des centres de collecte ou éleveurs. Pour établir ces programmes de collecte, l'unité « Collecte de lait » utilise une démarche intuitive basée uniquement sur l'expérience du personnel. Nous avons constaté que cette démarche présente beaucoup de limites notamment en ce qui concerne les coûts de transport importants qui sont engendrés par les tournées. C'est dans ce sens que s'oriente notre objectif à travers ce stage de fin d'études, qui est de proposer une nouvelle approche optimisant la tournée des camions de collecte. L'approche proposée consiste à modéliser la tournée de collecte du lait comme une variante du problème d'optimisation VRP (Vehicle Routing Problem). Afin d'obtenir des solutions efficaces à ce problème considéré comme NP-difficile, nous utilisons des méthodes de résolution basées sur les métaheuristiques..



## **Chapitre II**

### **Généralités sur les problèmes de tournées de véhicules**

## 1. Introduction

Dans ce chapitre, nous rappelons l'ensemble des définitions et des concepts de base liés à la logistique et à la chaîne logistique. Nous nous intéressons particulièrement à une fonction importante de la gestion de la chaîne logistique qui est l'élaboration des tournées de véhicules et qui constitue le thème central de ce projet de fin d'étude. Après avoir présenté le problème de tournée de véhicules VRP, son modèle mathématique de base ainsi que ses principales variantes, nous présentons une synthèse des méthodes de résolution des problèmes de type VRP. Une attention particulière sera accordée aux méthodes de résolution basées sur les métaheuristiques. Ces dernières seront utilisées pour résoudre efficacement un cas pratique d'élaboration de tournée de véhicules qui concerne la collecte de lait frais de l'entreprise DDA.

## 2. La logistique

La logistique est l'activité qui a pour objet de gérer les flux physiques, et les données (informatives, douanières et financières) s'y rapportant, dans le but de mettre à disposition les ressources correspondant à des besoins (plus ou moins) déterminés en respectant les conditions économiques et légales prévues, le degré de qualité de service attendu, les conditions de sécurité et de sûreté réputées satisfaisantes.

Le mot "logistique" a pour origine un mot grec qui indique l'art du raisonnement et du calcul, la notion de logistique a été utilisée depuis l'époque de la guerre mondiale, et on constate, après toutes les définitions attribuées par les spécialistes à ce mot, que la logistique signifie un ensemble de maillons reliés entre eux et qui collaborent pour accomplir un travail bien déterminé. [P5 Thèse Doctorat Oualid GUEMRI]

Pour [L'Association for Supply Chain Management (ASCM)] la logistique est définie comme :

- 1) Dans un contexte industriel, l'art et la science d'obtenir, produire et distribuer les composants et produits au bon endroit et dans les quantités requises.

- 2) Dans un contexte militaire (qui est l'usage le plus fréquent), cela peut aussi inclure les mouvements de personnel.

Dans la définition de la norme AFNOR (norme X 50-600), la logistique est "une fonction dont la finalité est la satisfaction des besoins exprimés ou latents, aux meilleures conditions économiques pour l'entreprise et pour un niveau de service déterminé. Les besoins sont de nature interne (approvisionnement de biens et de services pour assurer le fonctionnement de l'entreprise) ou externe (satisfaction des clients). La logistique fait appel à plusieurs métiers et savoir-faire qui concourent à la gestion et à la maîtrise des flux physiques et d'informations ainsi que des moyens".

### **3. La chaîne logistique**

De nombreux processus de l'entreprise impliquent donc des facettes logistiques, en particulier la Chaîne Logistique (CL) qui va des fournisseurs aux clients. Davantage encore que d'autres activités, la Logistique est caractérisée par un maillage complexe d'actions entre des acteurs faisant souvent des métiers différents, qu'il faut planifier, piloter ou coordonner dans l'espace et dans le temps. Considérée au départ surtout en termes de stocks, la logistique raisonne de plus en plus en termes de flux : flux de planification et ordonnancement des tâches, ordonnancement et circulation (workflow) des activités, objets et documents nécessaires à la chaîne logistique.

[Lee et al., 1993] a défini la chaîne logistique comme étant un " réseau d'installations qui assure les fonctions d'approvisionnement en matières premières, de transformation de ces matières premières en composants puis en produits finis, et de distribution du produit fini vers le client."

Pour [Tsay et al., 1999] la CL se définit comme "un ensemble de deux ou plusieurs entreprises liées par des flux de marchandises, d'informations et financiers."

[Rota- Franz et al., 2001] reconnaît que "La chaîne logistique d'un produit fini se définit comme l'ensemble des entreprises qui interviennent dans les processus d'approvisionnement en composants, de fabrication, de distribution et de vente du produit, du premier des fournisseurs au client ultime".

Bien que ces définitions paraissent similaires, elles mettent en exergue les points essentiels [Buzon, 2006] :

- une chaîne logistique se définit par rapport à un produit fini ou à un service offert,
- une chaîne logistique fait intervenir au moins trois organisations,
- les organisations industrielles sont liées par au moins 3 types de flux : des flux de produits, des flux d'informations et des flux financiers,
- les organisations assurent les fonctions d'approvisionnement, de transformation, de distribution et de vente,
- une organisation peut être impliquée dans plusieurs chaînes logistiques,
- une chaîne logistique est dynamique.

A partir de ces définitions nous pouvons constater que la CL est un réseau de distribution qui assure les fonctions d'acquisition de matériaux, la transformation de ces matériaux en produits intermédiaires et finis, et la distribution de ces produits finis aux clients.

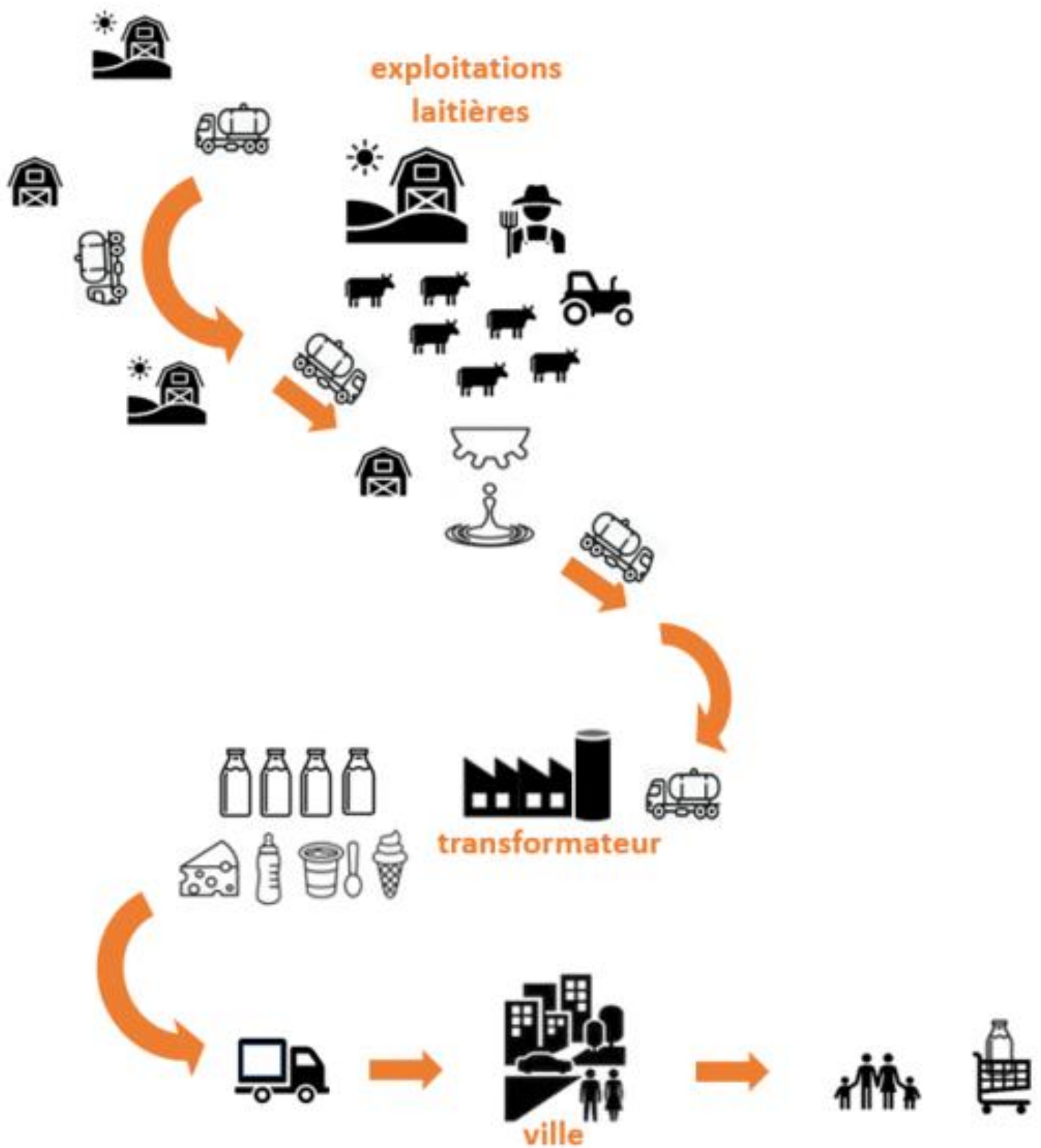


Figure 2.1 : Chaîne logistique d'une entreprise de production laitière

### **3.1. Gestion de la chaîne logistique**

Étant donné la complexité de la chaîne logistique, la gestion de cette dernière (Supply Chain Management (SCM) en anglais) est une tâche qui peut s'avérer difficile et complexe. Le [Council of Supply Chain Management Professionals - CSCMP (2010, p.180)] explique que « *le SCM englobe la planification et la gestion de toutes les activités impliquées dans le sourcing et l'approvisionnement, la conversion, et toutes les activités de gestion logistique. Surtout, elle comprend également la coordination et la collaboration avec les partenaires du canal, qui peuvent être les fournisseurs, les intermédiaires, les tiers fournisseurs de services et les clients. Principalement, le SCM intègre la gestion de l'offre et de la demande au sein et entre les entreprises. Le SCM est une fonction d'intégration avec comme responsabilité principale relier des fonctions et des processus d'affaires importants au sein et entre les entreprises dans un modèle d'affaires cohérent et hautement performant. Il comprend toutes les activités de gestion logistique notées ci-dessus, ainsi que les opérations de fabrication, et il entraîne la coordination des processus et des activités avec et à travers le marketing, les ventes, la conception des produits, les finances et les technologies de l'information* ». Aussi Lambert (2006) explique que « *le Supply Chain management réussi nécessite l'intégration inter fonctionnelle des processus clé au sein de l'entreprise et à travers le réseau des entreprises qui constituent la Supply Chain* ».

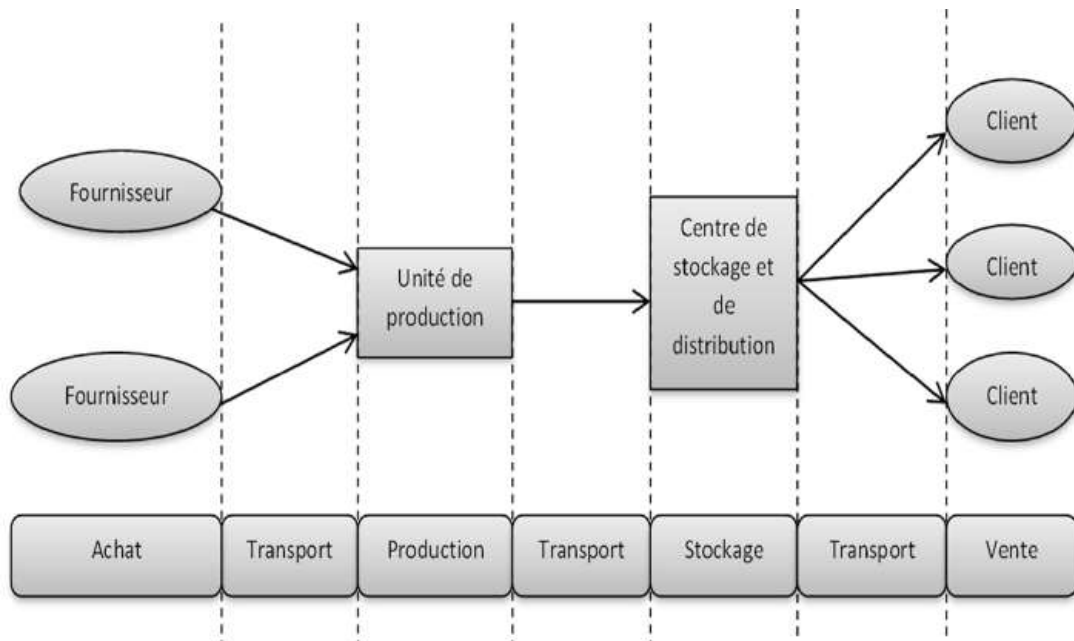
En outre, Mentzer (2004, p.22) définit le SCM comme étant « *la coordination systémique et stratégique des fonctions traditionnelles de gestion au sein d'une entreprise en particulier et à travers les entreprises au sein de la SC, afin d'améliorer la performance à long terme des entreprises individuelles et la SC dans son ensemble.*»

Ainsi, le Supply Chain management est la gestion globale de toutes les activités et processus impliqués dans l'écoulement du produit ou service allant de l'amont jusqu'à l'aval de la chaîne logistique. Visant la maximisation de la valeur globale de cette dernière, il implique principalement la coordination et l'alignement des objectifs et des décisions inter et intra organisationnelles.

### **3.2. Structure générale d'une chaîne logistique**

En prenant la définition de [R. Ganeshan and T. P Harrison 1995], la chaîne logistique a un ensemble de fonctions principales, qui consistent en l'achat des matières premières, la production, le

stockage, la distribution et la vente des produits finis. Dans la *Figure 2.2* nous présentons un exemple d'une chaîne logistique en indiquant l'emplacement de chaque fonction dans la chaîne [F. Galasso. 2007.].



*Figure 2.2 : Structure générale d'une chaîne logistique*

### 3.3. Niveaux de décisions dans une chaîne logistique

La gestion d'une chaîne logistique nécessite de prendre un ensemble de décisions importantes. Dans les chaînes logistiques, on distingue généralement trois niveaux de décisions selon la portée temporelle de la décision: des décisions opérationnelles, des décisions tactiques et des décisions stratégiques [K. Bahloul 2011. ; O. Guemri, and all 2016. ; P. Vallin 2003.]. Nous donnons, ci-après, un bref aperçu sur chaque niveau avec des exemples :

- **Décisions stratégiques** : les décisions stratégiques définissent la politique de l'entreprise sur le long terme, une durée s'étalant souvent sur plusieurs années (deux ans et plus), ce sont les décisions les plus importantes qui permettent de dessiner la structure générale de la chaîne logistique : la localisation des infrastructures (l'ouverture et la fermeture des usines, l'emplacement des dépôts etc.), le choix des partenaires et des fournisseurs et enfin, le choix de mode de transport et des différentes technologies.

- **Décisions tactiques** : leurs influences portent sur un horizon à moyen terme de plusieurs semaines jusqu'à plusieurs mois. A titre d'exemples pour les décisions de ce niveau : l'allocation des clients aux entrepôts, l'affectation des produits aux sites de production et la planification des processus de production.
- **Décisions opérationnelles** : leurs influences portent sur un horizon d'un jour jusqu'à une semaine : l'élaboration des tournée de véhicules pour livrer les produits aux clients ou s'approvisionner en terme de matières premières, l'affectation des ressources aux tâches au sein des unités de production etc.

Dans ce travail nous nous intéressons à un type particulier de décision opérationnelle qui concerne l'élaboration de la tournée de véhicule pour la collecte de lait frais pour l'entreprise Danone Djurdjura Algérie (DDA).

Dans la section suivante nous montrons l'importance des activités de transport et de distribution dans la chaîne logistique en insistant notamment sur la fonction de l'élaboration des tournées de véhicules qui constitue le thème central de notre travail de PFE.

#### **4. Le transport et la distribution dans la chaîne logistique**

L'optimisation des systèmes de transport et de distribution est un défi majeur pour les entreprises car les études ont montré que les coûts de transport et de distribution constituent le tiers des coûts opérationnels d'une chaîne logistique [Journal of business logistics]. Le système de distribution est composé d'un ensemble de fonctions principales, comprenant la localisation des centres d'entreposage et de distribution, la gestion de stock et enfin l'élaboration des tournées de véhicules pour l'approvisionnement en matière premières et la livraison des produits finis aux clients. Donc, l'optimisation des systèmes de distribution consiste à résoudre l'ensemble des problèmes liés à ces fonctions. Dans ce qui suit, nous allons nous intéresser uniquement à la fonction l'élaboration des tournées de véhicules qui constitue le thème principal de ce projet de fin d'étude.



#### **4.1. Le problème du voyageur de commerce TSP**

Le TSP ou Traveling Salesman Problem, est un problème d'optimisation combinatoire qui consiste à trouver le plus court chemin à parcourir pour un seul véhicule, depuis un point de départ, passant par un ensemble de points où chaque point est visité une et une seule fois et retournant au même point de départ.

Le terme problème du voyageur de commerce, vient de la traduction de l'anglais américain Traveling salesman problem, qui est apparu dans les années 1930 ou 40, sans doute à l'université de Princeton où plusieurs chercheurs s'y intéressaient (The traveling Salesman Problem : A Computational Study, Princeton University Press, 2006.).

#### **4.2. Le problème de tournées de véhicules**

Le problème de tournées de véhicules (appelé VRP en anglais pour Vehicle Routing Problem) est une classe de problèmes de recherche opérationnelle et d'optimisation combinatoire, il a été introduit pour la première fois par Dantzig et al. [1954] sous le nom de « Truck Dispatching Problem ». Le but est de minimiser le coût de livraison des biens. Vu son importance et ses multiples applications dans l'industrie et dans la vie quotidienne, le VRP sollicite l'attention de plusieurs chercheurs et fait l'objet de nombreux travaux de recherche.

Le problème VRP est une extension du problème du voyageur du commerce (TSP) [Dhaenens et al., 2002]. Dans sa version la plus basique dite Capacitated VRP (CVRP) ou VRP avec contraintes de capacité, une flotte de véhicules de capacité finie, basée dans un dépôt, doit assurer des tournées entre plusieurs clients (ou villes) ayant demandé chacun une certaine quantité de marchandises. L'ensemble des clients visités par un véhicule désigne la tournée de celui-ci. Chaque client doit être desservi une et une seule fois et chaque tournée commence et se termine au dépôt.

L'objectif du VRP est de minimiser le coût 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 ou collectées sur une tournée ne doit pas dépasser la capacité du véhicule qui l'assure.

### 4.3. Modélisation mathématique

Nous présentons ici une formulation mathématique pour le problème VRP. Dans cette formulation l'objectif est de minimiser la distance parcourue par les véhicules. Au départ, nous définissons les paramètres et les variables utilisées :

#### 4.3.1. Paramètres

- $I = \{0, \dots, N\}$  : ensemble des points de collecte (clients) (dont le dépôt est le client 0)
- $K = \{1, \dots, V\}$  : ensemble des véhicules
- $h_{ij}, \forall (i, j) \in I \times I$  : coût relatif au déplacement du client  $i$  vers le client  $j$
- $d_i, i \in I$  : demande du client  $i$
- $C_k, k \in K$  : capacité du véhicule  $v$

#### 4.3.2. Variables de décision

$$x_{ijk} = \begin{cases} 1 & \text{si le véhicule } k \text{ va de } i \text{ vers } j \\ 0 & \text{sinon} \end{cases} \quad \forall (i, j) \in I \times I, \forall k \in K$$

$$y_{ik} = \begin{cases} 1 & \text{si le véhicule } k \text{ visite le client } i \\ 0 & \text{sinon} \end{cases} \quad \forall i \in I, \forall k \in K$$

#### 4.3.3. Fonction objectif :

En utilisant les notations et les hypothèses ci-dessus, la formulation mathématique du modèle VRP est présentée comme suit:

$$(VRP) \quad \text{Minimiser} \quad \sum_{i \in I} \sum_{j \in I} \left( h_{ij} \times \sum_{k \in K} x_{ijk} \right) \quad (1)$$

#### 4.3.4. Contraintes

$$\sum_{k \in K} y_{ik} = 1, i = 1, \dots, N \quad (2)$$

$$\sum_{j \in I} x_{ijk} = y_{ik}, \forall i \in I, \forall k \in K \quad (3)$$

$$\sum_{i \in I} x_{ijk} = y_{jk}, \forall j \in I, \forall k \in K \quad (4)$$

$$\sum_{i \in I} d_i y_{ik} \leq C_k, \forall k \in K \quad (5)$$

$$\sum_{i \in S} \sum_{j \in S} x_{ijk} \leq \text{Card}(S) - 1, \forall S \subset I, 1 \leq \text{Card}(S) \leq \left\lfloor \frac{N}{2} \right\rfloor, \forall k \in K \quad (6)$$

$$\begin{aligned} x_{ijk} &\in \{0,1\}, \quad \forall i, j \in I, \forall k \in K \\ y_{ik} &\in \{0,1\}, \quad \forall i \in I, \forall k \in K \end{aligned} \quad (7)$$

Dans le modèle proposé, la fonction objectif (1) à minimiser représente la somme des distances parcourues par la flotte de véhicules.

La contrainte (2) indique que chaque client (hors dépôt) est visité par un et un seul véhicule. Les contraintes (3) et (4) expriment que si un véhicule visite un client, il arrive à partir d'un autre client et repart vers un autre client. La contrainte (5) garantit le non dépassement des capacités des véhicules. La contrainte (6) permet d'éliminer les sous-cycles ou les sous-circuits.

Enfin, la contrainte (11) exprime les contraintes d'intégrités des variables de décisions.

Le VRP peut être formulé comme un problème de programmation linéaire en nombre entier. Il a des applications directes dans le transport, la logistique, etc. Par exemple trouver le chemin le plus court pour les bus de ramassage scolaire, pour les camions de ramassage des ordures et aussi pour les camions de collecte de lait. Le VRP, qui est une extension du TSP, présente plusieurs caractéristiques communes aux problèmes d'optimisation combinatoire et fournit ainsi une plateforme idéale pour étudier les méthodes générales applicables à un grand nombre de ces problèmes.

#### 4.4. Quelques variantes du problème de tournées de véhicules

Durant les cinq dernières décennies, de nombreuses variantes de VRP ont été proposées [G. Laporte 2009.]. En effet, l'apparition d'une nouvelle variante est due essentiellement à l'ajout d'une contrainte, une caractéristique ou un nouvel élément au problème (l'ajout, la modification ou l'élimination d'une

ou de plusieurs contraintes du VRP peut nous mener vers d'autres variantes). Dans cette partie nous essayons de présenter les variantes les plus connus du VRP [GUEMRI, 2017] :

1. Problème de tournées de véhicules avec contrainte de capacité (Capacitated Vehicle Routing Problem (CVRP)):

Le CVRP est un VRP où la contrainte de capacité de véhicules est imposée, c'est-à-dire que la somme des demandes des clients servis par chaque véhicule ne doit pas dépasser sa capacité. Dans le CVRP la flotte de véhicules est supposée homogène.

2. Problème de tournées de véhicules dynamique (Dynamic Vehicle Routing Problem (DVRP)) :

Le DVRP est une variante du VRP où les données nécessaires pour résoudre le problème dépendent du temps, c-à-dire, soit elles ne sont pas connues à l'avance (manque d'information avant la résolution de problème) ou bien elles peuvent être changées après le commencement des tournées comme par exemple l'ajout d'une nouvelle demande pour un nouveau client ou l'annulation d'une demande d'un client.

3. Problème de tournées de véhicules avec fenêtres de temps (Vehicle Routing Problem with Time Windows (VRPTW)) :

Le VRPTW est un VRP où une contrainte temporelle est imposée. En effet, dans le VRP un véhicule peut visiter un client à n'importe quel moment. Cependant, dans le VRPTW un véhicule ne peut visiter un client que dans un intervalle de temps bien précis, c'est ce que l'on appelle une fenêtre de temps.

4. Problème de tournées de véhicules multi-périodes (Periodic Vehicle Routing Problem (PVRP)) :

Dans le PVRP, nous avons un ensemble de périodes  $P = \{1, \dots, M\}$  qui représente l'horizon de planification, et pour chaque client, nous avons un nombre de visites  $k$  ( $1 \leq k \leq M$ ) qui doivent être effectuées. Donc, pour résoudre le PVRP nous devons, pour chaque période, déterminer les clients à visiter et construire les tournées de véhicules pour satisfaire ces clients.

5. Problème de tournées de véhicules multi-dépôts (Multi-Depots Vehicle Routing Problem (MDVRP)) :

Dans le VRP, la tournée de chaque véhicule commence et se termine à partir d'un dépôt central. Cependant, dans le MDVRP nous disposons de plusieurs dépôts et dans chaque dépôt nous disposons d'un ensemble de véhicules. Comme dans le VRP, la tournée de chaque véhicule commence et se termine à partir de son dépôt et chaque client doit être servi exactement une seule fois.

6. Problème de tournées de véhicules ouvert (Open Vehicle Routing Problem (OVRP)) :

Contrairement au VRP où la tournée de chaque véhicule doit se terminer au dépôt central, dans le problème OVRP les véhicules ne sont pas obligés de retourner au dépôt central. En effet, cette caractéristique est importante parce qu'elle change complètement la structure de VRP.

7. Problème de tournées de véhicules avec multi-tournées pour chaque véhicule (Le Vehicle Routing Problem with Multiple Trips (VRPMT)) :

La différence entre le VRP et le VRPMT réside dans le fait que dans le VRPMT un véhicule peut faire plusieurs tournées, c'est-à-dire, dans un horizon de planification avec plusieurs périodes et lorsque nous disposons d'une flotte de véhicules limitée, un véhicule peut faire, dans une même période, plusieurs tournées en tenant compte d'une durée maximale d'utilisation par période.

8. Problème de tournées de véhicules stochastique (Stochastic Vehicle Routing Problem (SVRP)) :

Un VRP est stochastique si un des éléments du problème est aléatoire, comme par exemple : la représentation des demandes des clients, du temps de transport sur un arc ou bien du coût d'un arc par une variable aléatoire.

9. Problème de tournées de véhicules multi-compartiments (Multi-Compartment Vehicle Routing Problem (MCVRP)) :

Le MCVRP est utilisé pour modéliser les cas où les demandes des clients concernent plusieurs produits incompatibles. Ces produits doivent être transportés sur un même véhicule mais dans des compartiments indépendants.

*10. Problème de tournées de véhicules avec une flotte hétérogène (Heterogeneous Fleet Vehicle Routing Problem (HVRP)) :*

Dans le VRP nous disposons d'une flotte de véhicules homogènes. Cependant dans le HVRP, nous disposons d'une flotte de véhicules ayant des caractéristiques différentes. Les véhicules peuvent être différenciés par : leurs capacités, leurs coûts de routage, leurs coûts d'utilisation, etc.

*11. Problème de tournées de véhicules avec sélection (Selective Vehicle Routing Problem (SVRP)) :*

On rencontre le SVRP lorsque nous disposons d'une flotte de véhicules avec une capacité qui n'est pas suffisante pour satisfaire toutes les demandes des clients. Donc, nous ne pouvons servir qu'un sous ensemble de clients sélectionnés à partir de l'ensemble des clients.

*12. Problème de tournées de véhicules avec livraison et ramassage (Vehicle Routing Problem with Pickup and Delivery (VRPPD)) :*

Comme pour le VRP, le problème VRPPD consiste à élaborer un ensemble de tournées pour satisfaire les demandes de clients. Cependant dans le VRPPD, nous distinguons deux types de demande : la livraison d'un produit et le ramassage d'un produit.

*13. Problème de tournées de véhicules avec retours (Vehicle Routing Problem with backhauls (VRPB)) :*

Comme pour le VRPPD, dans cette variante nous distinguons deux types de clients : des livreurs où le véhicule doit faire un ramassage et des receveurs où le véhicule doit faire une livraison. Cependant la différence entre les deux variantes est que dans le VRPB nous visitons tous les receveurs avant de visiter les livreurs.

*14. Problème de tournées de véhicules avec distribution partagée (Split Delivery Vehicle Routing Problem (SDVRP)) :*

On rencontre le SDVRP lorsqu'on a des demandes de clients qui sont supérieures à la capacité du véhicule. Dans ce cas, chaque client peut être servi une ou plusieurs fois selon le besoin, c'est-à-dire, le client peut être affecté à une ou plusieurs tournées.

## 5. Les méthodes de résolution du problème VRP :

Les méthodes de résolution des problèmes d'optimisation NP-difficiles tels que les problèmes de tournées de véhicules sont classés en deux catégories : les méthodes exactes et les méthodes heuristiques. Les méthodes exactes sont conçues pour trouver un optimum du problème. Toutefois, leurs durées de calcul tendent à augmenter exponentiellement avec la taille des instances des problèmes NP-difficiles qu'elles essaient de résoudre. Elles doivent donc être arrêtées prématurément lors de la résolution d'instances de grandes tailles, et ne produisent alors que des minorants appelés bornes inférieures ou des majorants appelés bornes supérieures du coût optimal.

A l'inverse, les méthodes heuristiques sont des algorithmes conçus pour produire une solution réalisable du problème traité, de bonne qualité mais pas forcément optimale, sans nécessiter des temps de calcul importants. Elles comprennent les heuristiques constructives, qui construisent une solution, les recherches locales, qui génèrent une suite de solutions de coûts décroissants jusqu'à l'optimum local, et les métaheuristiques, qui contiennent divers mécanismes pour se sortir des optima locaux. Les méthodes exactes et les méthodes heuristiques sont complémentaires, car leurs résultats permettent de s'évaluer mutuellement : un faible écart entre les meilleures bornes inférieures et bornes supérieures signifie que l'optimum est proche, un écart nul prouve que les valeurs sont déjà optimales. Enfin, il est courant d'utiliser les résultats d'une de ces approches pour améliorer la performance de l'autre. [NGUEVEU, 2009].

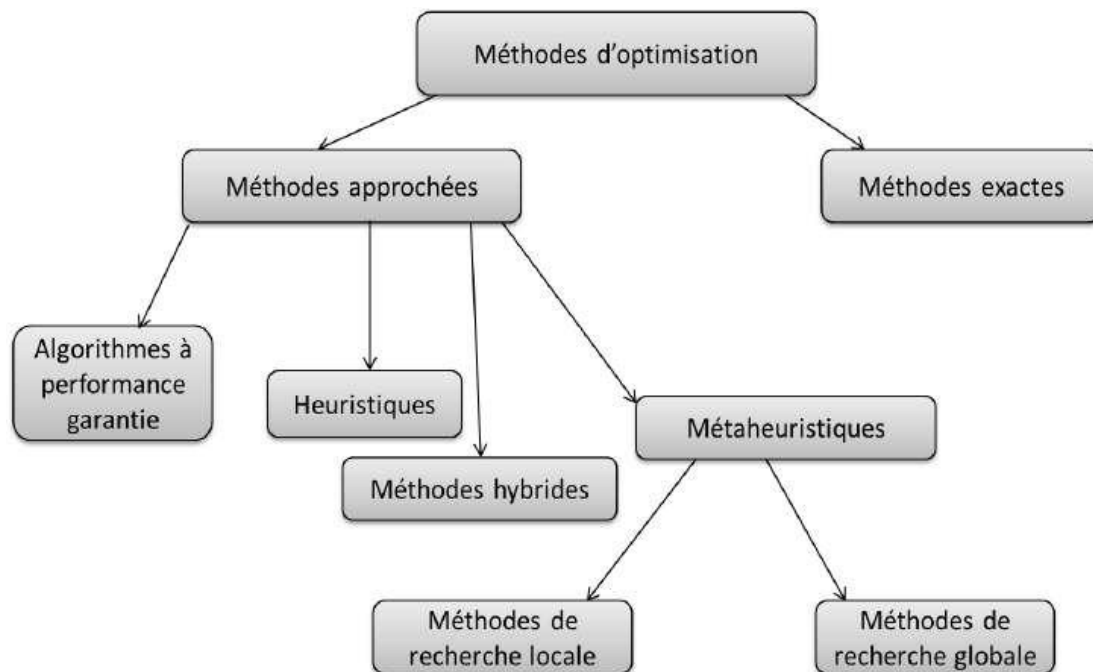


Figure 2.3 : Classification de méthodes d'optimisation

## 5.1. Les méthodes exactes

Les méthodes exactes sont des méthodes qui cherchent la solution optimale d'un problème en explorant exhaustivement toutes les solutions possibles dans l'espace de recherche. Cependant, l'inconvénient majeur de ces méthodes est le temps d'exécution, car toutes les solutions sont évaluées une à une et le temps d'exécution augmente exponentiellement avec la taille du problème considéré. Donc, ces techniques restent inappropriées aux problèmes combinatoires de grandes tailles [Gherboudj, 2013]. Comme exemple de ces méthodes, nous pouvons citer : la programmation dynamique, la génération de colonne, l'algorithme A\*, les algorithmes de type : branch & bound, branch & cut, branch & price, etc. [Guemri, 2017]

## 5.2. Les méthodes approchées

Les problèmes d'optimisation du monde industriel sont généralement de grandes tailles et disposent de nombreuses contraintes, donc les méthodes exactes ne sont pas utilisables. Dans ce cas, on doit chercher une bonne solution en un temps raisonnable au lieu d'attendre l'obtention d'une solution optimale après des années de calcul ! [Gherboudj, 2013]. Contrairement aux méthodes exactes, les méthodes approchées (ou méthodes heuristiques) ne garantissent pas l'optimalité de la solution mais, elles permettent de trouver des solutions de très bonne qualité en un temps d'exécution raisonnable, c-à-dire, elles cherchent un bon compromis entre la qualité de la solution et le temps de calcul. Dans la littérature, de nombreuses méthodes approchées ont été proposées. Nous pouvons diviser ces méthodes en 4 catégories: des heuristiques, des métaheuristiques, des algorithmes à performance garantie et des algorithmes hybrides. Par la suite nous présentons chaque catégorie.

### 5.2.1. Les heuristiques

Le mot heuristique vient du grec eurisko εὐρίσκω qui signifie « je trouve » d'où le célèbre Eureka d'Archimède. Dans la littérature, il existe plusieurs définitions d'une heuristique, ici nous présentons celle de [Feigenbaum et Feldman, 1963] : "*Une heuristique (règle heuristique, méthode heuristique) est une règle d'estimation, une stratégie, une astuce, une simplification, ou tout autre type de dispositif qui limite considérablement la recherche de solutions dans des espaces problématiques importants. Les heuristiques ne garantissent pas des solutions optimales. En fait, elles ne garantissent pas une*



*solution du tout. Tout ce qui peut être dit d'une heuristique utile, c'est qu'elle propose des solutions qui sont assez bonnes la plupart du temps*". En outre, dans le domaine de l'optimisation combinatoire on peut dire qu'une heuristique est une méthode approchée développée pour résoudre un problème particulier et elle nécessite des connaissances approfondies sur le problème traité.

Une heuristique, ou méthode approximative, est un algorithme qui fournit rapidement (en temps polynomial) une solution réalisable, pas nécessairement optimale, pour un problème d'optimisation NP-difficile. On oppose les méthodes approchées aux méthodes exactes, qui trouvent toujours l'optimum si on leur en laisse le temps (énumération complète, méthodes arborescentes, programmation dynamique). Les approches heuristiques, contournent le problème de l'explosion combinatoire en faisant délibérément des impasses et n'explorent qu'une partie de l'espace des combinaisons. Une méthode heuristique est généralement conçue pour un problème particulier, en s'appuyant sur sa structure propre. On parle de métaheuristique pour les méthodes approximatives générales, pouvant s'appliquer à différents problèmes.

### 5.2.2. Les métaheuristiques

Le mot métaheuristique est dérivé de la composition de deux mots grecs : méta, du grec  $\mu\epsilon\tau\alpha$  signifiant « au-delà » (ou « à un plus haut niveau ») et heuristique. Les métaheuristiques sont en général des problèmes aux données incomplètes, incertaines, bruitées ou confrontés à une capacité de calcul limitée **mais c'est des algorithmes d'optimisation visant à résoudre des problèmes d'optimisation difficile (souvent issus des domaines de la recherche opérationnelle, de l'ingénierie ou de l'intelligence artificielle) pour lesquels on ne connaît pas de méthode classique plus efficace ou plus robuste**. Ces méthodes sont, pour la plupart, inspirées de la physique (recuit simulé), de la biologie (algorithmes évolutionnaires) ou de l'éthologie (essais particuliers, colonies de fourmis). Le but d'une métaheuristique est de résoudre un problème d'optimisation donné : elle cherche un objet mathématique (une permutation, un vecteur, etc.) minimisant (ou maximisant) une fonction objectif, qui décrit la qualité d'une solution au problème.

L'ensemble des solutions possibles forme l'espace de recherche. L'espace de recherche est au minimum borné, mais peut être également limité par un ensemble de contraintes.

Dans certains cas, le but recherché est explicitement de trouver un ensemble d'optima "satisfaisants". L'algorithme doit alors trouver l'ensemble des solutions de bonne qualité, sans nécessairement se limiter au seul optimum : on parle de méthodes multimodales.

Les métaheuristiques manipulent une ou plusieurs solutions, à la recherche de l'optimum, la meilleure solution au problème. Les itérations successives doivent permettre de passer d'une solution de mauvaise qualité à la solution optimale. L'algorithme s'arrête après avoir atteint un critère d'arrêt, consistant généralement en l'atteinte du temps d'exécution imparti ou en une précision demandée.

D'une manière générale, les métaheuristiques s'articulent autour de ces notions :

- a) Voisinage : Le voisinage d'une solution est un sous-ensemble de solutions qu'il est possible d'atteindre par une série de transformations données. Par extension on désigne parfois par le terme « voisinage » l'ensemble des transformations considérées. Un voisinage simple pour le problème du voyageur de commerce sera, par exemple, l'ensemble des solutions qu'il est possible de construire en permutant deux villes dans une solution donnée (Figure 2.4).
- b) Diversification : La diversification (ou exploration) désigne les processus visant à récolter de l'information sur le problème optimisé.
- c) Intensification : L'intensification (ou exploitation) vise à utiliser l'information déjà récoltée pour définir et parcourir les zones intéressantes de l'espace de recherche.
- d) Apprentissage : La mémoire est le support de l'apprentissage, qui permet à l'algorithme de ne tenir compte que des zones où l'optimum global est susceptible de se trouver, évitant ainsi les optima locaux.

Une solution ou un ensemble de solutions est parfois appelé un état, que la métaheuristique fait évoluer via des transitions ou des mouvements en progressent de façon itérative et alternant des phases d'intensification, de diversification et d'apprentissage, ou en mêlant ces notions de façon plus étroite. Si une nouvelle solution est construite à partir d'une solution existante, elle est sa voisine. Le choix du voisinage et de la structure de donnée le représentant peut être crucial. L'état de départ est souvent choisi aléatoirement, l'algorithme se déroulant ensuite jusqu'à ce qu'un critère d'arrêt soit atteint.

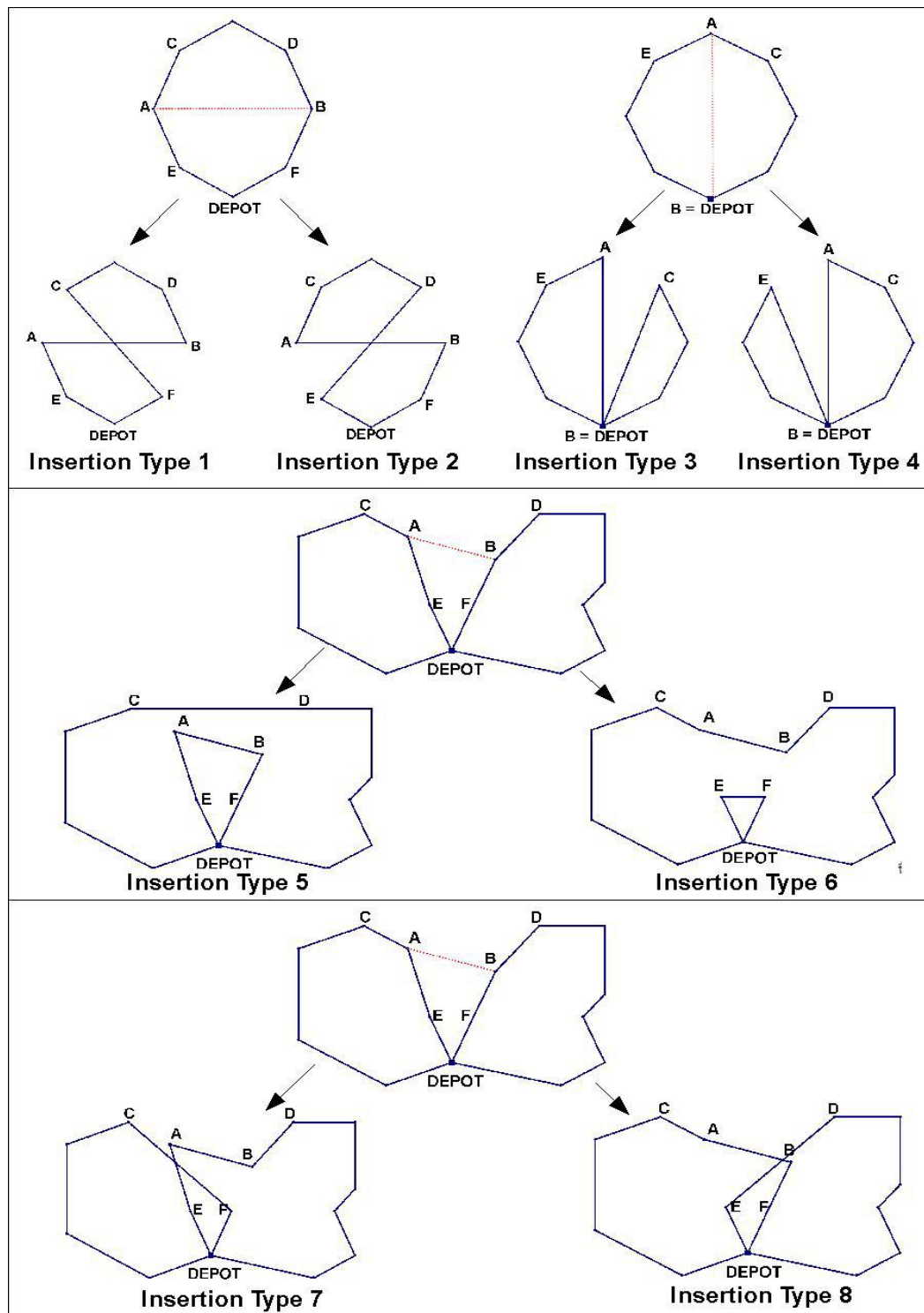


Figure 2.4 : Exemples de solutions voisines dans un problème VRP [Ngueveu. 2009]

Lorsqu'une solution est associée à une seule valeur, on parle de problème mono-objectif, lorsqu'elle est associée à plusieurs valeurs, on parle de problème multi-objectifs (ou multi-critères).

Dans un numéro spécial de la revue scientifique *European Journal of Operational Research*, consacré aux applications des métaheuristiques, les éditeurs ont constaté que la majorité des 20 articles publiés le furent dans deux domaines : les problèmes d'ordonnancement et de logistique. Les méthodes les plus utilisées appartiennent à la famille des algorithmes évolutionnaires, souvent hybridés avec des méthodes de recherche locale.

Quelques exemples de problèmes concrets, optimisés via des métaheuristiques :

- Problèmes de tournée de véhicules
- Optimisation de réseaux mobiles UMTS
- Gestion du trafic aérien
- Optimisation des plans de chargement des cœurs de réacteurs nucléaires

Dans le cadre de ce projet de fin d'étude, nous proposons quatre métaheuristiques pour la résolution d'une nouvelle variante du problème VRP baptisé « Vehicle Routing Problem with Time Windows and Duration Constraints (VRPTWDC) ». De plus, ces quatre métaheuristiques proposées seront appliquées sur un cas pratique issu de l'industrie agro-alimentaire qui concerne de la collecte de lait frais pour l'entreprise DDA. Ceci fera l'objet des chapitres 3 et 4.

### **5.2.3. Les métaheuristiques à solution unique**

Les méthodes à solution unique, plus connues sous le nom de méthodes de recherche locale ou encore méthodes de trajectoire, sont basées sur l'évolution d'une seule solution dans l'espace de recherche. Typiquement, les méthodes de recherche locale démarrent d'une solution unique, puis à chaque itération, la solution courante est déplacée dans un voisinage local en espérant améliorer la fonction objectif. Les méthodes à solution unique englobent principalement la méthode de descente, la recherche locale itérée, la recherche à voisinage variable, la méthode du recuit simulé et la recherche tabou. Ces méthodes diffèrent essentiellement dans leur manière d'exploiter le voisinage local, i.e, choix des candidats et critères de déplacement.

Dans ce qui suit, nous présentons uniquement les métaheuristiques à solution unique utilisées pour la résolution de notre problème VRPTWDC, à savoir, le recuit simulé et la recherche tabou.

### 5.1.3.1 Recuit simulé

Le recuit simulé est une méthode de programmation empirique (métaheuristique) inspirée d'un processus utilisé en métallurgie. On alterne dans cette dernière des cycles de refroidissement lent et de réchauffage (recuit) qui ont pour effet de minimiser l'énergie du matériau. Cette méthode est transposée en optimisation pour trouver les extrema d'une fonction.

Elle a été mise au point par trois chercheurs de la société IBM, S. Kirkpatrick, C.D. Gelatt et M.P. Vecchi en 1983, et indépendamment par V. Černý en 1985.

Le recuit simulé s'appuie sur l'algorithme de Metropolis-Hastings, qui permet de décrire l'évolution d'un système thermodynamique. Par analogie avec le processus physique, la fonction objectif à optimiser est assimilée à l'énergie du matériau  $E$ . On introduit également un paramètre fictif, la température  $T$  du système. La figure 2.5 détaille l'algorithme du recuit simulé.

**Nécessite :** La fonction objectif  $f$  et la taille de la liste taboue

- 1: Générer une solution aléatoire  $S$
- 2: Calculer la *fitness*  $f(S)$  associée à la solution initiale  $S$
- 3: Initialiser la solution optimale :  $S_{opt} \leftarrow S$
- 4: **Tant que** la condition d'arrêt n'est pas vérifiée **Faire**
- 5:     Générer la liste des candidats non tabous par opération de voisinage
- 6:     Trouver la meilleure solution  $S'$  parmi les candidats
- 7:     **Si**  $f(S') < f(S)$  **Alors**
- 8:          $S \leftarrow S'$
- 9:          $S_{opt} \leftarrow S$
- 10:     **Fin Si**
- 11:     Mettre à jour la liste taboue
- 12: **Fin tant que**
- 13: **Retourner :** La solution optimale  $S_{opt}$

*Figure 2.5 : Algorithme du recuit simulé*

### 5.1.3.2 Recherche tabou

La recherche tabou est une métaheuristique d'optimisation présentée par Fred W. Glover en 1986. Cette méthode est une métaheuristique itérative qualifiée de recherche locale au sens large.

L'idée de la recherche tabou consiste, à partir d'une position donnée, à en explorer le voisinage et à choisir la position dans ce voisinage qui minimise la fonction objectif.

Il est essentiel de noter que cette opération peut conduire à augmenter la valeur de la fonction (dans un problème de minimisation) : c'est le cas lorsque tous les points du voisinage ont une valeur plus élevée. C'est à partir de ce mécanisme que l'on sort d'un minimum local.

Le risque cependant est qu'à l'étape suivante, on retombe dans le minimum local auquel on vient d'échapper. C'est pourquoi il faut que l'heuristique ait de la mémoire : le mécanisme consiste à interdire (d'où le nom de tabou) de revenir sur les dernières positions explorées.

Les positions déjà explorées sont conservées dans une file FIFO (appelée souvent liste tabou) d'une taille donnée, qui est un paramètre ajustable de l'heuristique. Cette file doit conserver des positions complètes, ce qui dans certains types de problèmes, peut nécessiter l'archivage d'une grande quantité d'informations. Cette difficulté peut être contournée en ne gardant en mémoire que les mouvements précédents, associés à la valeur de la fonction à minimiser.

**Nécessite :** La fonction objectif  $f$ , la température maximale  $T_{max}$ , la température minimale  $T_{min}$  et la fonction de diminution de la température  $abaisser(T)$ .

- 1: Initialiser la température :  $T \leftarrow T_{max}$
- 2: Générer une solution aléatoire  $S$
- 3: Calculer la *fitness*  $f(S)$  associée à la solution initiale  $S$
- 4: Initialiser la solution optimale :  $S_{opt} \leftarrow S$
- 5: **Tant que**  $T > T_{min}$  **Faire**
- 6:     **Tant que** l'équilibre thermodynamique n'est pas atteint **Faire**
- 7:         Tirer une nouvelle solution  $S'$  dans le voisinage de  $S$
- 8:         Calculer la variation d'énergie :  $\Delta f \leftarrow f(S') - f(S)$
- 9:         **Si**  $\Delta f \leq 0$  **Alors**
- 10:             Accepter la nouvelle solution :  $S \leftarrow S'$
- 11:         **Sinon**
- 12:             **Si**  $\exp\left(-\frac{\Delta f}{T}\right) > x$  aléatoire  $\in [0, 1]$  **Alors**
- 13:                 Accepter la nouvelle solution :  $S \leftarrow S'$
- 14:             **Fin Si**
- 15:         **Fin Si**
- 16:         **Si**  $f(S') < f(S)$  **Alors**
- 17:             Mettre à jour la solution optimale :  $S_{opt} \leftarrow S'$
- 18:         **Fin Si**
- 19:     Abaisser la température  $T$  :  $T \leftarrow abaisser(T)$
- 20:     **Fin tant que**
- 21: **Fin tant que**
- 22: **Retourner :** La solution optimale  $S_{opt}$

*Figure 2.6 : Algorithme de recherche tabou*

#### 5.2.4. Les métaheuristiques à population de solutions

Contrairement aux algorithmes partant d'une solution singulière, les métaheuristiques à population de solutions améliorent, au fur et à mesure des itérations, une population de solutions.

On distingue dans cette catégorie, les algorithmes évolutionnaires, qui sont une famille d'algorithmes issus de la théorie de l'évolution par la sélection naturelle, énoncée par Charles Darwin [Darwin, 1859] et les algorithmes d'intelligence en essaim qui, de la même manière que les algorithmes évolutionnaires, proviennent d'analogies avec des phénomènes biologiques naturels.

Nous citons quelques algorithmes évolutionnistes :

- les stratégies d'évolution,
- les algorithmes génétiques,
- les algorithmes à évolution différentielle,
- les algorithmes à estimation de distribution,
- les systèmes immunitaires artificiels,
- la recomposition de chemin (Path relinking en anglais)
- Shuffled Complex Evolution (Duan et al. 1992)

Dans la prochaine section, nous présentons les algorithmes génétiques qui seront utilisés comme métaheuristique pour la résolution du problème VRPTWDC étudié dans le cadre de projet de fin d'étude.

#### 5.1.4.1 Les algorithmes génétiques

Les algorithmes génétiques (GA : Genetic Algorithms en anglais) sont, sans conteste, la technique la plus populaire et la plus largement utilisée des algorithmes évolutionnaires. Les origines de ces algorithmes remontent au début des années 1970, avec les travaux de John Holland et ses élèves à l'Université du Michigan sur les systèmes adaptatifs [Holland, 1975]. L'ouvrage de référence de David E. Goldberg [Goldberg, 1989] a fortement participé à leur essor. Ces algorithmes se détachent en grande partie par la représentation des données du génotype, initialement sous forme d'un vecteur binaire et plus généralement sous forme d'une chaîne de caractères.

Chaque étape de l'algorithme génétique (AG) est associée à un opérateur décrivant la façon de manipuler les individus :

- *Sélection* : Pour déterminer quels individus sont plus enclins à se reproduire, une sélection est opérée. Il existe plusieurs techniques de sélection, les principales utilisées sont la sélection par tirage à la roulette (roulette-wheel selection), la sélection par tournoi (tournament selection), la sélection par rang (ranking selection), etc. [Goldberg & Deb, 1991; Blicke & Thiele, 1995].

- *Croisement* : L'opérateur de croisement combine les caractéristiques d'un ensemble d'individus parents (généralement deux) préalablement sélectionnés, et génère de nouveaux individus enfants. Là encore, il existe de nombreux opérateurs de croisement, par exemple le croisement en un point, le croisement en n-points ( $n \geq 2$ ) et le croisement uniforme (voir figure 2.7).

- *Mutation* : Les descendants sont mutés, c'est-à-dire que l'on modifie aléatoirement une partie de leur génotype, selon l'opérateur de mutation.

- *Remplacement* : Le remplacement (ou sélection des survivants), comme son nom l'indique, remplace certains des parents par certains des descendants. Le plus simple est de prendre les meilleurs



individus de la population, en fonction de leurs performances respectives, afin de former une nouvelle population (typiquement de la même taille qu'au début de l'itération).

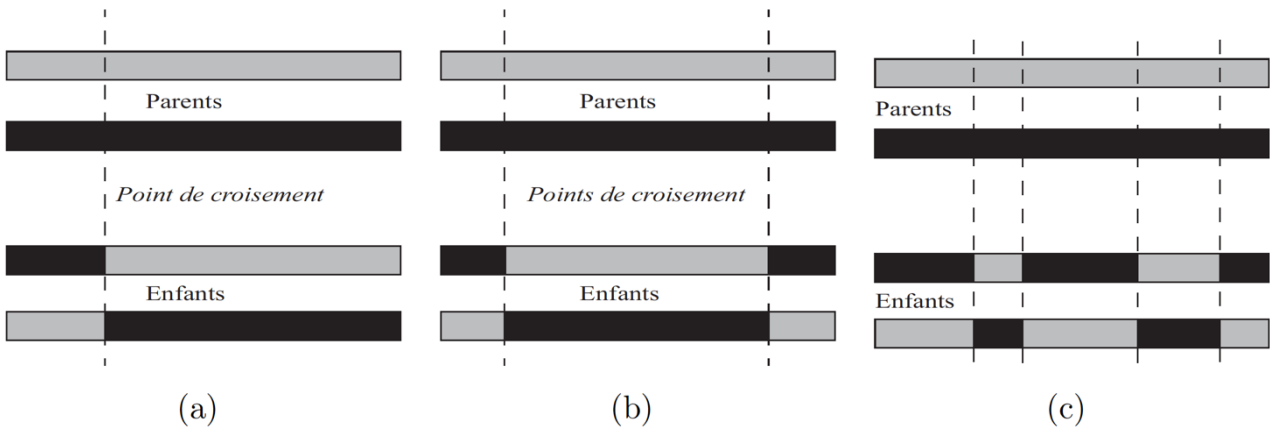


Figure 2.7 : Exemples de croisement : (a) croisement simple en un point, (b) croisement en deux points, (c) croisement uniforme [Boussaid, 2013]

## 6. Conclusion

À travers ce deuxième chapitre nous avons exposé les concepts de la logistique, la chaîne logistique et le Supply Chain Management. Une attention particulière a été accordée à l'élaboration des tournées de véhicules qui constitue l'une des fonctions les plus importantes de la gestion de la chaîne logistique. Nous avons énoncé par la suite la définition du problème de tournée de véhicule VRP et présenté son modèle de base ainsi que ses principales variantes. Nous avons dressé une synthèse des méthodes de résolution des problèmes de type VRP en s'intéressant particulièrement aux métaheuristiques. Enfin, nous avons détaillé le fonctionnement de trois métaheuristiques, à savoir, le recuit simulé, la recherche tabou et les algorithmes génétiques, ces métaheuristiques seront utilisées dans la suite de ce mémoire.

Dans le chapitre qui suit, nous développons un nouveau modèle de tournée de véhicules intégrant deux contraintes temporelles et proposons quatre métaheuristiques pour sa résolution.

## **Chapitre III**

### **Résolution du problème VRPTWDC**

## 1. Introduction

Dans ce chapitre, nous proposons un nouveau modèle de tournée de véhicules baptisé « *Vehicle Routing Problem with Time Windows and Duration Constraints (VRPTWDC)* ». Le modèle VRPTWDC développé constitue une extension et une généralisation du modèle VRP dans la mesure où nous introduisons en plus deux contraintes temporelles. La première contrainte concerne les fenêtres de temps à l'intérieur desquelles chaque client peut être visité. La deuxième contrainte impose une durée maximale pour chaque route (ou circuit). Cette deuxième contrainte est motivée principalement par les nouvelles règles complexes imposées par les différentes législations des pays concernant les transports de longue distance. Ceci permet de limiter le nombre d'heures que les chauffeurs peuvent consacrer à la conduite. Ainsi, la durée d'une route (tournée) comprenant le temps de déplacement entre les différents clients et le temps de service au niveau chaque client, ne peut dépasser une limite supérieure.

En dépit du caractère fortement restrictif de sa définition, le problème VRPTWDC permet de modéliser un panel étendu d'applications réelles : allant de l'élaboration des tournées pour les livraisons de produits et de colis dans diverses industries et commerces aux livraisons d'argent en espèces pour les banques et les distributeurs automatique de billets (DAB), le ramassage scolaire, etc.

Pour la résolution du modèle VRPTWDC, nous proposons quatre (04) approches d'optimisation basée sur les métaheuristiques suivantes : Recuit simulé, Recherche tabou, Algorithme génétique et une dernière approche hybridant un algorithme génétique avec la recherche tabou. Ces différentes approches sont détaillées et expliquées. Enfin, pour illustrer l'applicabilité des approches d'optimisation proposées, nous présentons un panel d'expériences numériques et analyses, où nous comparons les résultats des différentes métaheuristiques.

## 2. Description du problème

Le problème VRPTWDC étudié dans ce chapitre est une extension du problème classique de tournée de véhicules VRP. Il modélise bien un problème de transport très répandu qui est celui de la collecte de produits auprès d'un ensemble de points de collecte (clients) répartis géographiquement.

Ainsi, le problème VRPTWDC consiste à affecter chaque client à une tournée (ou 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 livré par chaque client. En plus des contraintes de capacités, le problème VRPTWDC impose deux contraintes temporelles : (i) fenêtres de temps pour chaque client, où un véhicule ne peut visiter un client que dans un intervalle de temps bien précis et (ii) une contrainte sur la durée maximum de chaque route. 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.

### **3. Modèle mathématique VRPTWDC**

#### **3.1. Hypothèses**

Le problème VRPTWDC étudié dans ce chapitre est basé sur les hypothèses suivantes :

- a) Chaque véhicule doit effectuer une seule tournée par période. Son parcours commence et se termine au niveau du dépôt (point de départ et lui-même le point d'arrivée final).
- b) La charge totale d'un véhicule lors d'un circuit ne doit pas dépasser sa capacité.
- c) Les demandes des clients peuvent être différentes.
- d) Le coût de déplacement qui doit être minimisé est relatif à la distance parcouru par les véhicules de chaque tournée.
- e) Tous les clients doivent être visités.
- f) Chaque client est visité une et une seule fois sauf le dépôt.
- g) Le véhicule ne peut visiter un client qu'après que le temps de disponibilité de ce dernier soit arrivé (borne inférieure de la fenêtre de temps).
- h) Si un véhicule arrive à un client avant la date de début de sa fenêtre, il attend.
- i) Un véhicule reste à l'arrêt au niveau d'un client le temps nécessaire pour le chargement des produits (temps de service du client).
- j) La durée de chaque tournée ne peut dépasser une durée maximale.

### 3.2. Paramètres

- $I = \{0, \dots, N\}$  : ensemble des points de collecte (clients) (dont le dépôt est le client 0)
- $K = \{1, \dots, V\}$  : ensemble des véhicules
- $h_{ij}, \forall (i, j) \in I \times I$  : coût relatif au déplacement du client  $i$  vers le client  $j$
- $d_i, i \in I$  : demande du client  $i$
- $C_k, k \in K$  : capacité du véhicule  $v$
- $t_{ijk}$ : temps mis par le véhicule  $k$  pour aller du client  $i$  au client  $j$
- $[e_i, l_i]$  : fenêtre de temps du client  $i$
- $s_i$  : le temps de service/chargement au client  $i$  ( $s_0 = 0$ , temps de service nul pour le dépôt)
- $T$  : durée maximale de chaque tournée

### 3.3. Variables de décision

$$x_{ijk} = \begin{cases} 1 & \text{si le véhicule } k \text{ va de } i \text{ vers } j \\ 0 & \text{sinon} \end{cases} \quad \forall (i, j) \in I \times I, \forall k \in K$$

$$y_{ik} = \begin{cases} 1 & \text{si le véhicule } k \text{ visite le client } i \\ 0 & \text{sinon} \end{cases} \quad \forall i \in I, \forall k \in K$$

$A_i$  : temps d'arrivée au client  $i$

$D_i$  : temps de départ du client  $i$

### 3.4. Fonction objectif

En utilisant les notations et les hypothèses ci-dessus, la formulation mathématique du modèle VRPTWDC est présentée comme suit:

$$(VRPTW) \quad \text{Minimiser} \quad \sum_{i \in I} \sum_{j \in I} \left( h_{ij} \times \sum_{k \in K} x_{ijk} \right) \quad (1)$$

### 3.5. Contraintes

$$\sum_{k \in K} y_{ik} = 1, i = 1, \dots, N \quad (2)$$

$$\sum_{j \in I} x_{ijk} = y_{ik}, \forall i \in I, \forall k \in K \quad (3)$$

$$\sum_{i \in I} x_{ijk} = y_{jk}, \forall j \in I, \forall k \in K \quad (4)$$

$$\sum_{i \in I} d_i y_{ik} \leq C_k, \forall k \in K \quad (5)$$

$$\sum_{i \in S} \sum_{j \in S} x_{ijk} \leq \text{Card}(S) - 1, \forall S \subset I, 1 \leq \text{Card}(S) \leq \left\lfloor \frac{N}{2} \right\rfloor, \forall k \in K \quad (6)$$

$$x_{ijk} = 1 \Rightarrow e_i \leq A_i \leq l_i, \forall i, j \in I, \forall k \in K \quad (7)$$

$$x_{ijk} = 1 \Rightarrow e_i \leq A_i + s_i \leq l_i, \forall i, j \in I, \forall k \in K \quad (8)$$

$$x_{ijk} = 1 \Rightarrow D_i + t_{ijk} \leq (l_i - s_i) \leq l_i, \forall i, j \in I, \forall k \in K \quad (9)$$

$$\sum_{i \in I} \sum_{j \in I} (t_{ijk} + s_i) x_{ijk} \leq T, \forall k \in K \quad (10)$$

$$\begin{aligned} x_{ijk} &\in \{0,1\}, \quad \forall i, j \in I, \forall k \in K \\ y_{ik} &\in \{0,1\}, \quad \forall i \in I, \forall k \in K \\ D_i &\geq A_i \geq 0, \quad \forall i \in I \end{aligned} \quad (11)$$

Dans le modèle proposé, la fonction objectif (1) à minimiser représente la somme des distances parcourues par la flotte de véhicules.

La contrainte (2) indique que chaque client (hors dépôt) est visité par un et un seul véhicule. Les contraintes (3) et (4) expriment que si un véhicule visite un client, il arrive à partir d'un autre client et repart vers un autre client. La contrainte (5) garantit le non dépassement des capacités des véhicules. La contrainte (6) permet d'éliminer les sous-cycles ou les sous-circuits.

Les contraintes (7), (8) et (9) assurent le respect des fenêtres de temps. La contrainte (10) limite la durée maximale de chaque route. La somme des temps de service aux clients ainsi que des temps de transports d'une même route est alors limitée à T.

Enfin, la contrainte (11) exprime les contraintes d'intégrités des variables de décisions.

### **3.6. Complexité du problème VRPTWDC**

Bien que simple en apparence, ce problème est notablement très difficile à résoudre. En fait, il a été montré que le problème VRP classique était NP-difficile, alors le problème VRPTWDC qui est plus général est certainement NP-difficile. Ainsi, s'il est tout à fait possible de déterminer une solution optimale pour des instances de petite taille, cela devient rapidement irréalisable pour des instances de moyenne ou de grande taille. Or, la plupart des problèmes de tournées de véhicules de la vie réelle se situent justement dans cette seconde catégorie.

Par conséquent, des méthodes approchées sont nécessaires pour résoudre le problème VRPTWDC.

Dans ce travail, nous proposons quatre approches basées sur les métaheuristiques pour la résolution efficace du problème VRPTWDC. La section suivante décrit ces quatre approches de résolution.

## **4. Résolution du problème VRPTWDC**

Après avoir modélisé le problème, il nous reste qu'à proposer une solution au problème. Dans la présente étude, nous proposons l'utilisation de quatre approches de résolution basées sur les métaheuristiques. Ces dernières présentent actuellement des alternatives intéressantes pour la résolution des problèmes d'optimisation difficile (comme le problème VRPTWDC) pour lesquels on ne connaît pas d'algorithmes classiques plus efficaces.

Les méthodes de résolution que nous avons proposées pour la résolution du problème VRPTWDC sont basées sur les métaheuristiques suivantes : le recuit simulé, la recherche tabou, l'algorithme génétique (AG) ainsi qu'une dernière méthode basée sur une hybridation d'un AG avec recherche tabou.

### **4.1. Solution initiale**

Toutes les méthodes d'optimisation commencent par la génération d'une solution initiale. En général, cette solution initiale est générée de façon aléatoire. Cependant, il est possible de générer la solution initiale en utilisant une heuristique permettant d'accélérer la convergence de la méthode utilisée. Dans notre cas, nous avons préféré générer la solution initiale par un algorithme glouton (Greedy).

Le principe de fonctionnement de l’algorithme glouton est décrit à travers les étapes suivantes :

- Pour chaque véhicule, l’algorithme glouton choisit un nœud parmi tous les nœuds candidats qui respectent les contraintes du modèle VRPTWDC, à savoir : la capacité maximale du véhicule, la fenêtre de temps du nœud candidat et la durée maximale de la route du véhicule traité.
- Le nœud choisi est le nœud qui présente le coût (distance) de déplacement minimal par rapport au dernier nœud ajouté au véhicule traité. Le nœud choisi est ajouté à la route du véhicule traité.
- Si l’algorithme ne trouve pas de nœud candidat il passe au véhicule suivant, s’il ne reste plus de véhicules à traiter l’algorithme s’arrête sans trouver de solution.
- L’algorithme glouton se termine si tous les nœuds ont été affectés, il retourne le coût total des tournées de tous les véhicules.

L’Algorithme ci-dessous décrit un pseudo code de l’heuristique gloutonne.

---

```

Coût_Total = 0 ; /* Coût de la solution gloutonne, initialisée à 0 au début de l’algorithme */
Indice_Veh = 0 ; /* Indice du véhicule traité, au début initialisé à 0 */
Tant que (il existe des nœuds qui ne sont pas affecté)
    Coût_Min = Valeur infinie ;
    Pour (i = 1 ; i <= N ; i++)
        Si (Nœud[i] n’est pas affecté) Alors
            Si (Nœud[i] peut être affecté au Vehicule[Indice_Veh] tout en respectant les
                contraintes de capacité maximale du Vehicule[Indice_Veh], de temps d’arrivé au
                plus tôt du Nœud[i] et de la durée maximale de la tournée) Alors
                Coût = Distance entre le Nœud[i] et le dernier nœud affecté à
                    Vehicule[Indice_Veh];
                Si (Coût < Coût_Min) Alors
                    Coût_Min = Coût ;
                    Nœud_Candidat = Nœud[i] ;
                FinSi
            FinSi
        FinSi
    FinPour
    Si (Nœud_Candidat == NULL) Alors /* Aucun nœud candidat qui satisfait les contraintes */
        Si (Indice_Veh < K-1) Alors /* S’il reste encore des véhicules à traiter */
            Indice_Veh++ ;
        FinSi
        Sinon /* Il ne reste plus de véhicules à traiter, donc pas de solution possible */
            Afficher(“Pas de solution possible”) ;
        FinSinon
    FinSi
    Sinon
        Ajouter le Nœud_Candidat au véhicule Vehicule[Indice_Veh] ;
        Coût_Total = Coût_Total + Coût_Min ;
    FinSinon
FinTantQue
Retourner Coût_Total ;

```

---

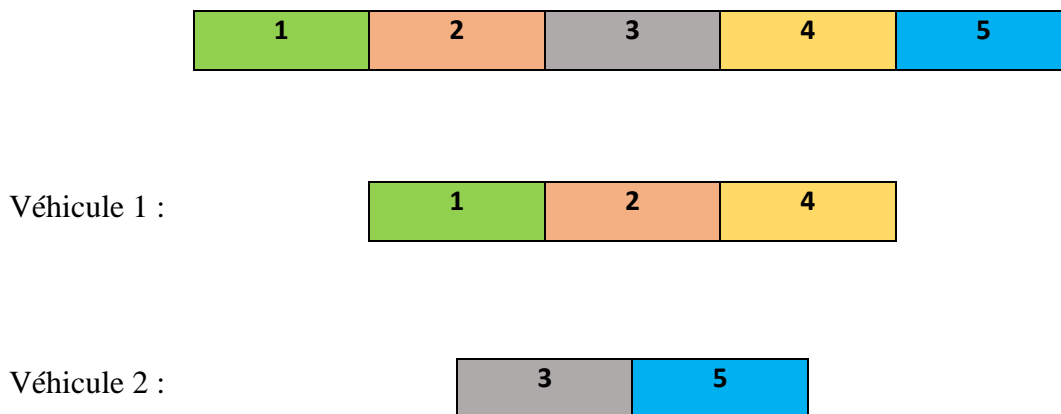
Algorithme 3.1 : Pseudo code de l’heuristique gloutonne



#### 4.1.1. Exemple de déroulement de l'heuristique gloutonne

Exemple d'une solution gloutonne : Cinq clients (1 à 5) ayant des demandes 10, 40, 50, 20, 35 respectivement, et 2 véhicules de capacité  $C_1 = C_2 = 90$  chacun (*Figure 3.1*).

L'algorithme glouton commence par le premier véhicule qui lui sera affecter la 1<sup>er</sup> client et puis le 2<sup>ème</sup> client, la somme des demandes est de 50, ensuite il passe au 3<sup>ème</sup> client mais il ne peut pas l'ajouter car ça va dépasser la capacité du véhicule ( $100 > C_1$ ) donc il saute vers le 4<sup>ème</sup> client et il l'ajoute, la demande totale étant de 70 est supportée par le véhicule 1. Comme il ne peut pas ajouter le 5<sup>ème</sup> client car ( $105 > C_1$ ), il passe vers le véhicule suivant (Deuxième véhicule). Enfin, puisque les clients 3 et 5 respectent les contraintes de capacité ( $85 < C_2$ ) du véhicule 2, ces deux clients seront affectés au véhicule 2.



*Figure 3.1 : Exemple de solution gloutonne*

Le routage obtenu de la solution gloutonne est comme suit (le nœud 0 étant le dépôt) :

Véhicule 1 :  $0 \rightarrow 2 \rightarrow 4 \rightarrow 1 \rightarrow 0$

Véhicule 2 :  $0 \rightarrow 5 \rightarrow 3 \rightarrow 0$

#### 4.2. Approche de résolution par Recuit simulé (RS)

Dans ce qui suit, nous présentons les différentes étapes du déroulement de l'algorithme RS proposé :

##### Étape 1 Initialisation :

Au début de la recherche, la température  $T$  est élevée, autorisant ainsi des dégradations importantes de la fonction objectif.

Après avoir générer la solution initiale  $S_0$  avec l'heuristique gloutonne, le résultat obtenu dans cette tournée est traduit par la distance totale parcourue appelée « fitness ».

## Étape 2      **Passage de la solution actuelle vers une nouvelle solution voisine :**

A partir de cette solution initiale, une solution voisine aléatoire  $S'$  est générée par un mouvement élémentaire. Dans notre cas, un mouvement élémentaire représente un **déplacement d'un nœud** qui possède 4 paramètres à déterminer :

*Vehicule\_Source* : Le véhicule source (route source) à partir duquel le nœud choisi va être déplacé.

*Vehicule\_Destination* : Le véhicule destination (route destination) vers lequel le nœud choisi sera déplacé.

*Position\_Source* : La position source dans la route source qui indique le nœud à déplacer

*Position\_Destination* : La position destination dans la route destination qui indique où le nœud sera déplacé.

Ce mouvement va engendrer une solution voisine, cela se fait comme suit :

En premier lieu, un véhicule source (route source) et un autre véhicule destination (route destination) sont choisis aléatoirement. Nous déterminons ensuite d'une manière aléatoire la position source au niveau de la route source qui représente le nœud à déplacer. Pareil pour la position de destination qui sera déterminée aléatoirement au niveau de la route de destination. Enfin, il ne reste plus qu'à déplacer le nœud à déplacer vers la position de destination au niveau de la route de destination.

Les schémas suivants (*Figures 3.2 et 3.3*) illustrent deux exemples de déplacement pour deux cas différents : le premier cas concerne le déplacement d'un nœud dans le même véhicule (Intra-Véhicule), tandis que le deuxième cas concerne le déplacement d'un nœud d'un véhicule vers un autre véhicule (Inter-Véhicule).

**1<sup>er</sup> cas : Déplacement d'un nœud dans le même véhicule (Déplacement Intra-Véhicule)**

Paramètres : Véhicule\_Source = Véhicule\_Destination = 1 ; Position\_Source = 1 ; Position\_Destination = 3

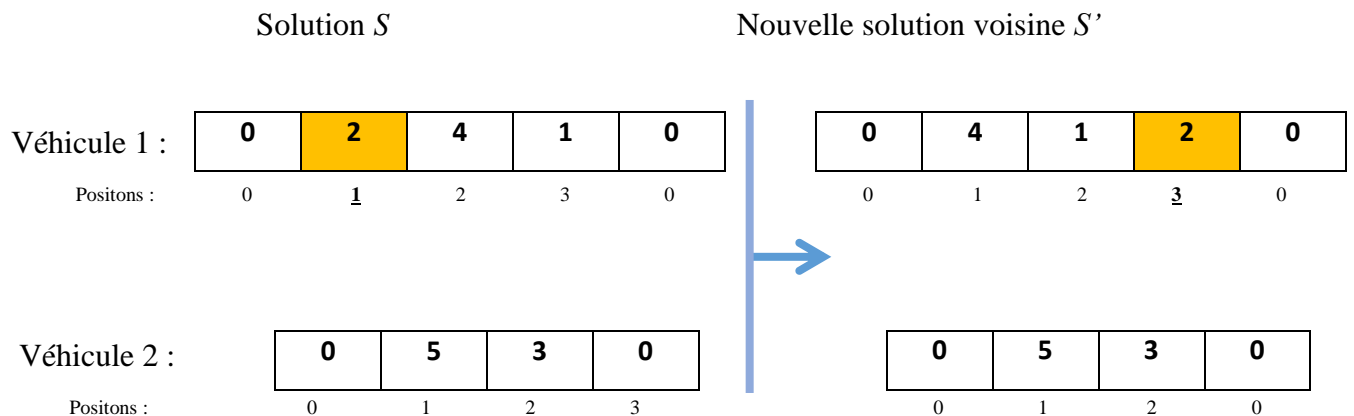


Figure 3.2 : : Exemple de solution voisine 1

La solution  $S'$  obtenue est comme suit (le nœud 0 étant le dépôt) :

Véhicule 1 :  $0 \rightarrow 4 \rightarrow 1 \rightarrow 2 \rightarrow 0$

Véhicule 2 :  $0 \rightarrow 5 \rightarrow 3 \rightarrow 0$

**2<sup>ème</sup> cas : Déplacement d'un nœud entre deux véhicules différents (Déplacement Inter-Véhicule)**

Paramètres : Véhicule\_Source = 1 ; Véhicule\_Destination = 2 ; Position\_Source = 2 ; Position\_Destination = 1

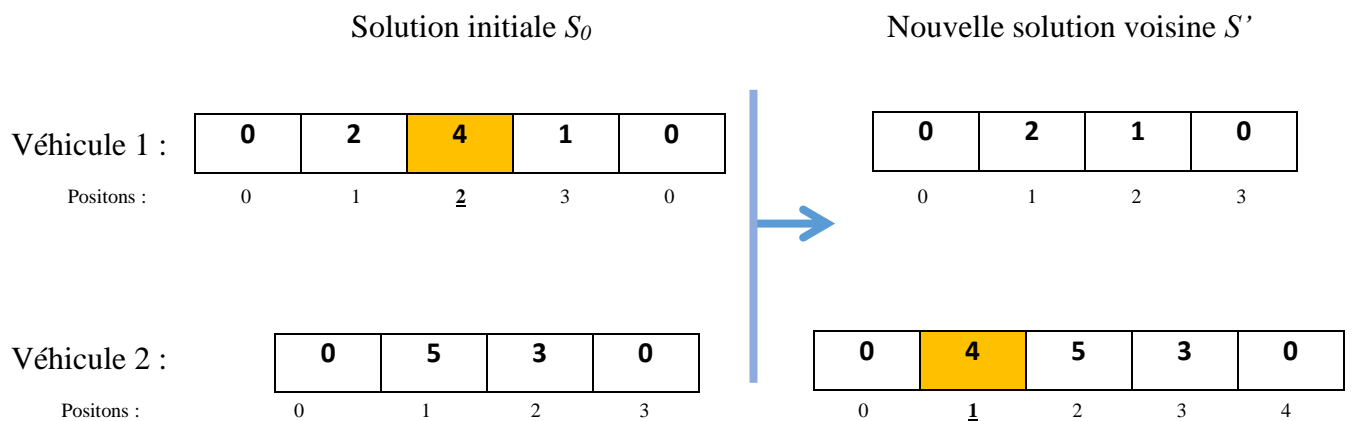


Figure 3.3 : Exemple de solution voisine 2

La solution  $S'$  obtenue est comme suit (le nœud 0 étant le dépôt) :

Véhicule 1 :  $0 \rightarrow 1 \rightarrow 2 \rightarrow 0$

Véhicule 2 :  $0 \rightarrow 4 \rightarrow 5 \rightarrow 3 \rightarrow 0$

### **Étape 3 Comparaison :**

En générant ainsi la nouvelle solution  $S'$ , si cette nouvelle solution est meilleure en fitness que la précédente, elle est acceptée comme solution initiale de la prochaine itération. Dans le cas contraire, la nouvelle solution n'est pas forcément rejetée, et peut être acceptée avec la probabilité de Boltzmann exprimée par la fonction exponentielle de  $(d_1-d_0)/T$ , où  $T$  est la température actuelle. Si la nouvelle solution voisine est rejetée (ne respecte pas la probabilité), la solution initiale  $S_0$  sera conservée pour la prochaine itération.

Cette étape est réitérée tant que l'équilibre thermodynamique n'est pas atteint.

### **Étape 4 Abaissement de la température :**

Une fois l'équilibre atteint, la température est abaissée d'un certain nombre de degrés, déterminés par le coefficient de refroidissement « $\mu$ ». La température initiale de la prochaine itération est remplacée par la nouvelle température  $T$  qui est égale à  $(T_{\text{initiale}} * \mu)$ .

Au fil des itérations, la diminution de la température réduit la probabilité d'acceptation des dégradations de la fitness. Quand la température tend vers 0, seules les améliorations sont acceptées. (Mekamcha. 2020)

### **Étape 5 Critère d'arrêt :**

Ainsi de suite, jusqu'à l'atteinte d'un critère d'arrêt qui dépend de la température actuelle  $T$ ; si  $T$  n'atteint pas la température finale  $T_{\text{finale}}$ , nous revenons à l'étape 2, sinon l'exécution est arrêtée lorsque  $T = T_{\text{finale}}$ , ou nombre d'itérations maximal atteint ou bien dans le cas de "système figé".

## **4.3. Approche de résolution par recherche tabou**

A l'instar de l'algorithme du recuit simulé (RS) présenté précédemment, la méthode basée sur la recherche tabou (RT) que nous proposons utilise le même mouvement élémentaire pour générer les solutions voisines. Nous rappelons que ce mouvement consiste à déplacer un nœud vers une autre position au niveau de la même route ou une autre route différente (voir section 4.2 (Étape 2)).

Avant de détailler les différentes étapes de la méthode RT, il est judicieux de présenter d'abord le principe de fonctionnement de la liste tabou utilisée par l'algorithme RT proposé.

#### 4.3.1. Liste tabou adoptée

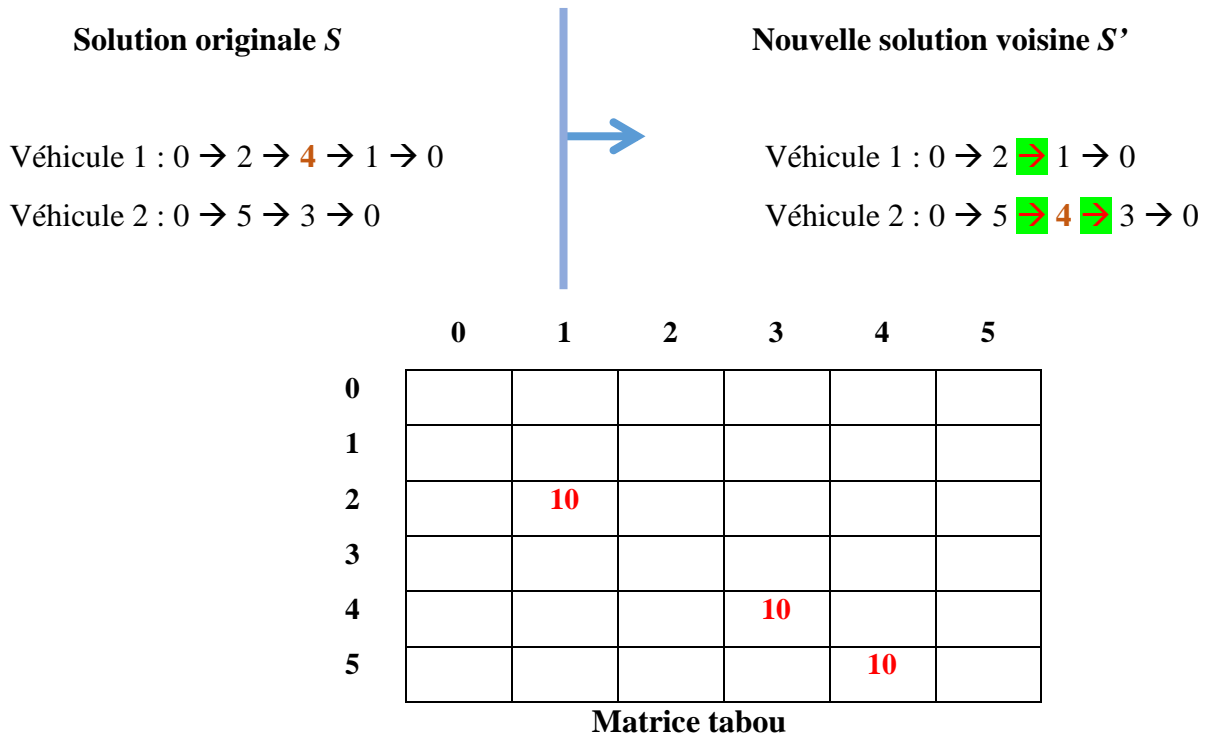
Dans la méthode RT adoptée, nous avons représenté la liste tabou sous forme d'une matrice. Cette dernière appelée « matrice tabou » permet de sauvegarder les dernières solutions explorées permettant ainsi d'empêcher l'algorithme RT tout déplacement vers les dernières solutions visitées.

#### 4.3.2. Principe de fonctionnement de la matrice tabou

- La matrice tabou adoptée est une matrice carrée de taille  $[N] \times [N]$  ( $N$  : nombre de nœuds).
- A chaque itération de l'algorithme RT, la meilleure solution voisine (obtenue après un déplacement d'un nœud) est mémorisée dans la matrice tabou.
- Pour mémoriser une solution voisine, nous conservons dans la matrice tabou uniquement les modifications apportées à la configuration de la solution originale qui ont permis l'obtention de la solution voisine.
- Ces modifications concernent les nouveaux liens (arcs) créés après le déplacement d'un nœud :
  - o Un nouveau lien est créé entre le nœud qui se trouve dans la *Position\_Source - 1* et le nœud qui se trouve dans la *Position\_Source + 1*
  - o Un nouveau lien est créé entre le nœud qui se trouve dans la *Position\_Destination* et le nœud qui se trouve dans la *Position\_Source*
  - o Un nouveau lien est créé entre le nœud qui se trouve dans la *Position\_Source* et le nœud qui se trouve dans la *Position\_Destination + 1*
- Dans la matrice tabou, une valeur *Tabou\_Horizon* est affectée aux cellules de la matrice correspondantes aux liens créés. Notons que la valeur *Tabou\_Horizon* indique le nombre d'itérations où une solution mémorisée dans la matrice tabou sera interdite d'utilisation. Notre matrice tabou peut être considérée alors comme une file d'attente de taille *Tabou\_Horizon*.
- Avant d'accepter une nouvelle solution voisine, l'algorithme vérifie d'abord si les cellules correspondant aux nouveaux liens créés possèdent des valeurs nulles (=0). Si ce n'est pas le cas, la nouvelle solution voisine sera rejetée.
- A chaque itération de l'algorithme RT, les valeurs de toutes les cellules (qui ont une valeur  $> 0$ ) sont décrémentés de 1. Ceci permettra de lever l'interdiction d'utilisation des meilleures solutions voisines enregistrées dans la matrice tabou après un nombre d'itérations égale à *Tabou\_Horizon*.

La *Figure 3.4* illustre un exemple montrant le fonctionnement de la matrice tabou. L'exemple concerne la mémorisation dans la matrice tabou d'une solution voisine issue d'une solution originale contenant 6 nœuds (y compris le nœud dépôt) et 2 véhicules. Dans l'exemple, nous supposons que le paramètre *Tabou\_Horizon* = 10.

*Paramètres : Véhicule\_Source = 1 ; Véhicule\_Destination = 2 ; Position\_Source = 2 ; Position\_Destination = 1*



*Figure 3.4 : Exemple de sauvegarde d'une solution voisine dans la Matrice Tabou*

Après avoir présenté le fonctionnement de la liste tabou adoptée dans notre méthode de recherche tabou, nous détaillons dans ce qui suit les différentes étapes du déroulement de l'algorithme RT proposé :

### Étape 1 Initialisation :

L'algorithme RT commence par calculer une solution initiale gloutonne  $S_0$ . En outre, la procédure d'initialisation crée une matrice tabou vide de taille  $[N] \times [N]$  ( $N$  étant le nombre de nœuds) et initialise les paramètres suivants : Nombre d'itérations, le nombre de voisins et le *Tabou\_Horizon*.

## **Étape 2      Passage de la solution actuelle vers une nouvelle solution voisine :**

A la différence de la méthode (RS) où une seule solution voisine est générée, dans la méthode (RT) nous générons un ensemble de solutions voisines de la solution courante en un seul mouvement élémentaire comme celui utilisé dans la méthode RS, pour ensuite en choisir la meilleure ( $S'$ ) après une comparaison de la fitness de chacune des configurations construites. Ce choix est adopté même si la solution voisine  $S'$  est moins bonne que la solution actuelle, c'est le cas lorsque tous les points du voisinage ont une fitness plus élevée. Nous rappelons que c'est grâce à ce mécanisme d'acceptation de la détérioration de la fonction objectif que la méthode tabou permet d'éviter des minimums locaux.

Notons qu'avant d'accepter une nouvelle solution voisine, l'algorithme vérifie d'abord si cette solution ne figure pas dans la matrice tabou (voir section 4.3.2). Si ce n'est pas le cas, la nouvelle solution voisine sera rejetée et une autre solution devra être générée jusqu'à l'obtention d'une solution non tabou.

## **Étape 4      Mis à jour de liste tabou :**

Cette nouvelle solution voisine  $S'$  sera enregistrée dans la matrice tabou. L'enregistrement de la solution  $S'$  dans la matrice tabou se fait en affectant une valeur *Tabou\_Horizon* aux cellules de la matrice correspondantes aux liens nouvellement créés (voir section 4.3.2).

Par ailleurs, les valeurs de toutes les cellules (qui ont une valeur  $> 0$ ) sont décrémentés de 1. Ceci permettra de lever l'interdiction d'utilisation des meilleures solutions voisines enregistrées dans la matrice tabou après un nombre d'itérations égale à *Tabou\_Horizon*.

## **Étape 5      Critère d'arrêt**

L'algorithme itère jusqu'à atteindre le nombre d'itérations maximal fixé dans l'étape d'initialisation.

## **4.4.      Approche de résolution par algorithme génétique**

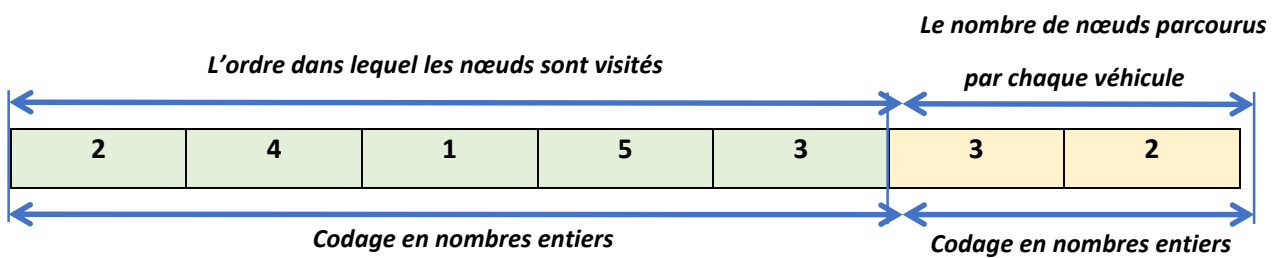
### **4.4.1.      Codage**

La première question abordée par tout utilisateur des algorithmes génétiques est la représentation ou le codage des variables de décisions à optimiser. Après avoir identifié le problème et les variables de

décision, nous devons représenter ces variables ainsi que les relations entre elles sous une forme propre aux algorithmes génétiques et appelée « chromosome ».

Dans notre algorithme génétique, un chromosome représente les tournées de chaque véhicule. En effet, chaque chromosome représente une solution candidate et est constitué de deux parties distinctes codées en nombre entiers. La première partie représente l'ordre dans lequel les nœuds sont visités et la seconde partie représente le nombre de nœuds parcourus par chaque véhicule. Du moment où chaque véhicule commence et termine sa tournée par le nœud dépôt (par convention, le nœud 0 désigne le dépôt), ce nœud n'est pas représenté dans le chromosome.

Un exemple de chromosome correspondant à un problème avec 5 nœuds et 2 véhicules est illustré à travers la *Figure 3.5*.



*Figure 3.5 : Exemple de chromosome*

La solution obtenue après décodage du chromosome donné en exemple est comme suit :

Véhicule 1 : 0 → 2 → 4 → 1 → 0

Véhicule 2 : 0 → 5 → 3 → 0

#### 4.4.2. Opérateurs génétiques adoptés

Les opérateurs de sélection, croisement et mutation sont utilisés pour faire évoluer l'algorithme génétique et ainsi améliorer la qualité des solutions au cours des générations. Puisque sur le même chromosome, il existe deux parties distinctes: ordre de visite des nœuds et nombre de nœuds parcourus par chaque véhicule, il convient alors de spécifier pour chacune d'elle les opérateurs génétiques de croisement et de mutation permettant une meilleure exploration de l'espace de recherche.



#### 4.4.3. Opérateur de sélection

Avant de lancer les opérations de croisement et de mutation, il faut d'abord sélectionner les individus pour la reproduction, parmi ceux de la population actuelle. La méthode de sélection adoptée dans notre AG est la « sélection binaire par tournoi ». Elle consiste à tirer au hasard deux solutions de la population P puis sélectionner la solution avec la fitness (performance) la plus élevée. Dans l'AG proposé, la fitness d'une solution représente le coût total de déplacement des véhicules (distance totale parcourue) exprimé par la fonction objectif de l'équation (1) (section 3.4).

#### 4.4.4. Opérateurs de croisement

En ce qui concerne les opérateurs de croisement, chaque paire de chromosomes parents sélectionnée subira une opération de croisement avec une probabilité ( $p_c$ ) et elle sera remplacée par ses descendants dans la prochaine génération.

Pour la première partie du chromosome qui exprime l'ordre de visite des nœuds, nous utilisons un croisement deux points qui est une extension du croisement un point, avec le choix aléatoire de deux points de coupure, ainsi, les segments entre les deux points sélectionnés sont permutés entre les deux parents pour créer deux enfants. Notons que les points possèdent la même probabilité d'être sélectionnés.

Dans la deuxième partie du chromosome représentant le nombre de nœuds visités par chaque véhicule, nous utilisons un croisement un point qui consiste à choisir aléatoirement un point de coupure identique pour les deux parents et d'échanger les deux fragments situés à droite du point choisit, ce qui produit deux solutions enfants.

#### 4.4.5. Opérateurs de mutation

La mutation joue le rôle de bruit et empêche l'évolution de se figer, elle est primordiale puisqu'elle permet de maintenir la diversité de la population et par conséquent une bonne exploration de l'espace de recherche. Le rôle de l'opérateur de mutation est de modifier aléatoirement la valeur d'un gène dans un chromosome avec une probabilité fixée ( $p_m$ ).

Dans notre AG et dans la première partie du chromosome, le gène à muter est remplacé par un nombre aléatoire entre 1 et  $N$  ( $N$  étant le nombre de nœuds). Dans la deuxième partie du chromosome, nous remplaçons le gène à muter par un aléatoire entre 1 et  $K$  ( $K$  étant le nombre de véhicules).

Après l'étape de mutation, une procédure de modification ou de réparation est exécutée pour chaque chromosome ne respectant pas les contraintes du modèle VRPTWDC.

#### 4.4.6. Population initiale

Le mécanisme de génération de la population initiale permet de produire une population d'individus qui servira de base pour les futures générations. Le choix de la population initiale est important car il peut rendre plus ou moins rapide la convergence vers l'optimum global. En général, cette population initiale est créée de façon aléatoire. Cependant, il est possible de générer la population initiale avec des solutions structurées en utilisant des méthodes heuristiques simples pour obtenir des individus de bonne qualité ([Zhu 2000], [Baker et al. 2003]).

Dans notre cas, nous avons généré une population initiale mixte contenant à la fois des solutions aléatoires et des solutions structurées obtenues à partir de l'heuristique gloutonne (voir section 4.1).

#### 4.4.7. Les étapes de l'AG adopté

Les différentes étapes de l'algorithme génétique adopté se présentent comme suit :

- **Étape 1.** Générer la population initiale  $P$  de taille  $N$  aléatoirement.
- **Étape 2.** Calculer les fitness de toutes les solutions dans  $P$ .
- **Étape 3.** Sélectionner deux parents en utilisant la « sélection binaire par tournoi ».
- **Étape 4.** Générer deux solutions enfants par le croisement des deux solutions parents avec une probabilité  $p_c$ .
- **Étape 5.** Exécuter l'opérateur de mutation avec une probabilité  $p_m$  pour chaque solution enfant.
- **Étape 6.** Exécuter une fonction de correction pour les solutions enfants infaisables qui ne respectent pas les contraintes du modèle VRPTWDC proposé.

- **Étape 7.** Appliquer une stratégie (heuristique ou métaheuristique) pour améliorer la qualité des solutions enfants.
- **Étape 8.** Ajouter les deux solutions enfants dans la population suivante  $G$ .
- **Étape 9.** Répéter les étapes de 3 à 8 pour obtenir  $N$  solutions enfants dans  $G$ .
- **Étape 10.** Répéter les étapes de 2 à 9 jusqu'à la satisfaction des conditions d'arrêt (le nombre maximal d'itérations est atteint).

Pour améliorer les performances de l'AG adopté, nous appliquons une méthode d'amélioration pour augmenter la qualité de chaque chromosome enfant obtenu (Étape 7). Deux méthodes d'amélioration sont utilisées, ceci fera l'objet de la prochaine section.

#### 4.5. Amélioration des performances de l'AG

Pour améliorer les performances de l'AG adopté et décrit ci-dessus, nous avons incorporé deux méthodes permettant d'augmenter la qualité des solutions de chaque chromosome à travers les différentes générations. La première méthode consiste en l'application d'une heuristique d'amélioration par recherche locale (RL). La deuxième méthode utilise une métaheuristique basée sur la recherche taboue (RT) permettant ainsi d'obtenir une méthode de résolution hybride combinant deux métaheuristicues, à savoir, AG avec la recherche tabou.

Notons que ces deux méthodes sont basées sur le même AG décrit précédemment.

Dans ce qui suit, nous présentons les deux méthodes utilisées pour améliorer les performances de l'AG.

#### 4.6. Méthode d'amélioration de l'AG basée sur la recherche locale (AG+RL)

Cette méthode est décomposée en deux étapes. Dans la première étape, l'AG détermine (à travers les chromosomes de chaque génération) l'affectation des véhicules aux différents nœuds et l'ordre de visite de ces nœuds. Dans la seconde étape, une heuristique d'amélioration par recherche locale est

utilisée pour améliorer l'ordre de visite des nœuds affectés à chaque véhicule permettant ainsi d'augmenter la qualité des solutions obtenues par l'AG.

Dans cette méthode, à partir d'une solution de départ  $S_0$  obtenue à partir du décodage d'un chromosome de l'AG, une heuristique de recherche locale explore un voisinage relativement restreint de solutions issues de mouvements basiques, afin d'y trouver une solution  $S'$  améliorante.  $S'$  remplace alors  $S$  pour une nouvelle itération de l'algorithme. La recherche locale s'arrête quand la meilleure solution trouvée par l'algorithme n'a pas été améliorée depuis un nombre donné d'itérations ( $R$ ).

Un mouvement basique est utilisé pour passer d'une solution donnée à une solution voisine. Ce mouvement consiste tout d'abord à choisir aléatoirement une route (un véhicule), cette route sera considérée comme une solution TSP unique. Ensuite, un nœud de cette route est choisi aléatoirement, pour enfin déplacer ce nœud vers une autre position déterminée d'une manière aléatoire dans la même route.

L'Algorithme 3.2 décrit un pseudo code de l'heuristique d'amélioration par recherche locale.

---

**Input** :  $s_0$  /\* Solution initiale obtenue à partir du décodage d'un chromosome de l'AG \*/  
 $s = s_0$  ;  
 $Nb\_tentatives = 0$  ; /\* Nombre de tentatives pour trouver une solution améliorante \*/  
**Répéter**  
    A partir de  $s$ , générer un voisin aléatoire  $s'$  issu d'un mouvement basique;  
    **Si** ( $s'$  est meilleure que  $s$ ) **Alors**  
         $s = s'$  ;  
         $Nb\_tentatives = 0$  ;  
    **FinSi**  
    **Sinon**  
         $Nb\_tentatives = Nb\_tentatives + 1$  ;  
    **FinSinon**  
**Jusqu'à** ( $Nb\_tentatives > R$ )  
**Output** :  $s$

---

*Algorithme 3.2 : Pseudo code de l'heuristique basée sur la recherche locale*

*Dans la suite de ce document, nous employons la simple appellation « AG » pour désigner cette première méthode d'amélioration de l'AG basée sur la recherche locale.*

## 4.7. Méthode d'algorithme génétique hybridé avec la recherche tabou (AG+RT)

Dans cette deuxième méthode, nous employons une métaheuristique basée sur la recherche tabou comme celle utilisée dans la section 4.3 afin d'améliorer les performances de l'AG adopté. Cette hybridation entre AG et recherche tabou permet de combiner les avantages des deux métaheuristicques. L'hybridation entre métaheuristicques a fait l'objet de plusieurs travaux traitant les problèmes VRP [Haiyan, 2005][Guemri, 2017].

Notons que l'algorithme de la recherche tabou utilisé dans cette méthode hybride est identique à l'algorithme présenté dans la section 4.3. L'algorithme de recherche tabou est exécuté pour chaque solution issue du décodage de chaque chromosome à travers les différentes générations.

Le principe de la méthode hybride (AG+RT) que nous avons implémenté est présenté à travers les étapes suivantes :

- Pour toutes les générations, l'algorithme génétique détermine au niveau de chaque chromosome l'affectation des véhicules aux différents nœuds et l'ordre de visite de ces nœuds.
- Les solutions sont obtenues après décodage de chaque chromosome.
- Ces solutions sont utilisées comme solutions initiales à l'algorithme de recherche tabou.
- L'exécution de recherche tabou permet d'améliorer la qualité des solutions de chaque chromosome.

### 4.7.1. Paramètres de l'AG adopté

Les deux méthodes (AG et AG+RT) utilisent le même algorithme génétique exploitant le codage des données décrites précédemment afin d'optimiser le coût de déplacement total (distance totale parcourue par tous les véhicules). Pour se faire, nous devons choisir soigneusement les valeurs des paramètres régissant l'évolution de la population traitée par cet algorithme génétique : taille de la population, nombre de générations, probabilités de croisement et de mutation. Dans cette étude, après une série de tests, nous avons opté pour les paramètres du *Tableau 3.1* :

	Chromosome	
	L'ordre dans lequel les nœuds sont visités	Le nombre de nœuds parcourus par chaque véhicule
Représentation	Nombres entiers	Nombres entiers
Type de croisement	Deux points	Un point
Probabilité de croisement	0.75	
Type de mutation	Standard	Standard
Probabilité de mutation	0.008	
Taille de la population	30	
Nombre d'itérations	1500	

*Tableau 3.1 : Paramètres de l'AG adopté*

## 5. Expérimentation

Pour évaluer les performances et l'applicabilité de quatre approches de résolution proposées, plusieurs expériences numériques ont été exécutées et les résultats les plus pertinents sont indiqués dans cette section. Les algorithmes des quatre métaheuristiques ainsi que la simulation des différents scénarios sont implémentés en langage JAVA sous l'environnement Eclipse et exécutés sur un MacBook équipé d'un processeur Intel i5 (2.7 GHz), disposant d'une mémoire de 8Go et fonctionnant sous un système d'exploitation MacOS version 10.15.

Afin de vérifier la validité des différentes métaheuristiques utilisées, nous comparons les résultats obtenus par ces dernières avec les résultats obtenus par une méthode exacte. Notre choix s'est porté sur la comparaison avec la méthode exacte Branch & Bound (B&B) exécutée à travers le solveur Lingo.

Nous analysons le comportement des quatre méthodes en testant leur flexibilité et leur efficacité. Nous entendons par la flexibilité des méthodes, leur capacité à faire face à différentes instances ayant des caractéristiques différentes (en faisant varier le nombre de points de collecte, de véhicules, les fenêtres de temps et la durée maximale des tournées), et nous entendons par l'efficacité, leur capacité à donner des résultats très proches de ceux d'une méthode exacte avec un temps d'exécution acceptable.

## 5.1. Jeux de donnée

Notre procédure de génération des différentes instances consiste tout d'abord à fixer le nombre de nœuds et de véhicules. Ensuite, de générer aléatoirement les coordonnées des nœuds sur une grille de  $100 \times 100$ . Nous calculons enfin les distances euclidiennes entre chaque paire de nœud.

Nous utilisons un jeu de données adapté à notre problème VRPTWDC. Ainsi, pour bien tester les différentes métaheuristiques, nous générons 32 instances. Ces instances varient de 5 clients et 2 véhicules jusqu'à 100 clients et 35 véhicules. Le but est d'augmenter la difficulté à chaque instance et avec lesquelles les performances des métaheuristiques proposées seront clairement montrées.

Les différents paramètres utilisés pour la génération des instances de test sont présentés comme suit :

- **Nombre de nœuds ( $N$ ) et nombre de véhicules ( $K$ )** : nous avons considéré des instances de différentes tailles avec 5, 8, 10, 15, 30, 50, 80 et 100 nœuds et respectivement 2, 3, 4, 6, 10, 15, 30, 35 véhicules ;
- **Demandes des clients** : générées aléatoirement suivant une loi uniforme dans l'intervalle [4, 10] ;
- **La capacité des véhicules ( $C^P$ )**: nous supposons que les capacités sont les mêmes pour tous les véhicules, dans ce cas  $C^P$  représente la capacité de chacun et  $P$  l'indice de la formule respectée pour son calcul. Pour nos tests, nous avons considéré deux capacités de véhicules :  
 $C1 = (\text{Somme totales des demandes} / \text{Nombre de véhicules}) + 10$  ;  
 $C2 = (\text{Somme totales des demandes} / \text{Nombre de véhicules}) + 30$  ;
- **Distance entre les villes** : la distance euclidienne entre les nœuds ayant des coordonnées choisies aléatoirement dans l'espace  $[0, 100] \times [0, 100]$  ;
- **Temps de déplacement entre les clients** : nous supposons que les durées de trajet entre les clients sont égales à la distance entre ces derniers (nous supposons alors que les véhicules ont tous la même vitesse égale à  $1 \text{ u.d/u.t}$  ( $u.d$  : unité de distance /  $u.t$  : unité de temps) ;
- **Fenêtres de temps des clients** : initialisées aléatoirement.
- **Durée maximale des tournées ( $T$ )** : nous avons considéré deux durées maximale des tournées :  
 $T1 = \text{durée infinie (sans contrainte de durée maximale de tournée)}$   
 $T2 = (\text{Durée de trajet maximale}) \times 2$

Par ailleurs, les paramètres des différentes métaheuristiques proposées sont présentés dans le prochain point.

## 5.2. Réglage des paramètres des métaheuristiques utilisées

Après une série de tests pour chaque métaheuristique, nous avons opté pour les paramètres suivants :

- Pour la métaheuristique RS :
  - Température initiale = 10 000 ;
  - Paramètre de refroidissement = 0.98 ;
  - Nombre d'itération à chaque température = 100 ;
- Pour la métaheuristique RT :
  - Nombre d'itérations maximal = 2000
  - Nombre de voisins par itération = 200 ;
  - La taille de la liste tabou (*TABU\_Horizon*) = 10 ;
- Pour la métaheuristique AG : voir *Tableau 3.1*.
  
- Pour la métaheuristique hybride AG+TS : paramètres de de l'AG et de la RT fusionnés.

## 5.3. Résultats obtenus

Les résultats obtenus pour chaque instance de test sont détaillés à travers le *Tableau 3.2*. Ces résultats concernent les distances totales des tournées, le nombre de véhicules utilisés ainsi que le temps d'exécution obtenus pour chaque métaheuristique et par la méthode exacte Branch & Bound exécutée à travers le solveur Lingo. Notons qu'un délai de 1200 secondes (20 minutes) a été imposé sur l'algorithme B&B. Ainsi chaque fois que cette limite a été atteinte, nous interrompons l'exécution au niveau de Lingo et considérons que l'algorithme B&B n'a pas été en mesure de trouver une solution optimale dans un temps raisonnable.

Notons par ailleurs, que nous avons exécuté vingt (20) fois chaque métaheuristique (RS, RT, AG et AG+RT) et avons rapporté dans le *Tableau 3.2* les meilleurs résultats obtenus pour les différentes instances.



Instance	Clients (N)	Véhicules (K)	Capacités (C)	Durée max (T)	Greedy		Solveur Lingo			Recuit Simulé (RS)			Recherche Tabou (RT)			Algorithme Génétique (AG)			Hybridation AG+RT (10 it)		
					Distance obtenue (Km)	Nombre de véhicules	Distance obtenue (Km)	Nombre de véhicules	Temps d'exécution (s)	Distance obtenue (Km)	Nombre de véhicules	Temps d'exécution (s)	Distance obtenue (Km)	Nombre de véhicules	Temps d'exécution (s)	Distance obtenue (Km)	Nombre de véhicules	Temps d'exécution (s)	Distance obtenue (Km)	Nombre de véhicules	Temps d'exécution (s)
1	5	2	C1	T1	203	2	<b>197</b>	2	1s	<b>197</b>	2	0.125	<b>197</b>	2	0.04s	<b>197</b>	2	0.085	<b>197</b>	2	0,444s
2				T2	203	2	<b>197</b>	2	1s	<b>197</b>	2	0.122	<b>197</b>	2	0.05	<b>197</b>	2	0.09	<b>197</b>	2	0,654
3			C2	T1	193	1	<b>189</b>	1	1s	<b>189</b>	1	0.159	<b>189</b>	1	0.059	<b>189</b>	1	0.088	<b>189</b>	1	0.465
4				T2	193	1	<b>189</b>	1	1s	<b>189</b>	1	0.151	<b>189</b>	1	0.038	<b>189</b>	1	0.079	<b>189</b>	1	0.561
5	8	3	C1	T1	285.0	2	<b>278</b>	2	8s	<b>278.0</b>	2	0.128	<b>278.0</b>	2	0.048	<b>278.0</b>	2	0.129	<b>278.0</b>	2	0.819
6				T2	308.0	3	<b>286</b>		10s	<b>286.0</b>	3	0.145	<b>286.0</b>	3	0.04	<b>286.0</b>	3	0.632	<b>286.0</b>	3	0.632
7			C2	T1	286.0	2	<b>243</b>	2	8s	<b>243.0</b>	2	0.146	<b>243.0</b>	2	0.054	<b>243.0</b>	2	0.16	<b>243.0</b>	2	1.016
8				T2	308.0	3	<b>254</b>	2	6s	<b>254.0</b>	2	0.149	<b>254.0</b>	2	0.088	<b>254.0</b>	2	0.154	<b>254.0</b>	2	0.89
9	10	4	C1	T1	403.0	3	<b>318</b>	3	20s	<b>318.0</b>	3	1.451	<b>318.0</b>	3	0.044	<b>318.0</b>	3	0.187	<b>318.0</b>	3	1.152
10				T2	414.0	4	<b>318</b>	3	18s	<b>318.0</b>	3	0.125	<b>318.0</b>	3	0.054	<b>318.0</b>	3	0.181	<b>318.0</b>	3	1.261
11			C2	T1	295.0	2	<b>275</b>	2	22s	<b>275.0</b>	2	0.166	<b>275.0</b>	2	0.056	<b>275.0</b>	2	0.148	<b>275.0</b>	2	1.414
12				T2	383.0	3	<b>292</b>	3	20s	<b>292.0</b>	3	0.203	<b>292.0</b>	3	0.049	<b>292.0</b>	3	0.167	<b>292.0</b>	3	1.342
13	15	6	C1	T1	589.0	4	<b>545</b>	4	< 20min	<b>545.0</b>	4	0.135	<b>545.0</b>	4	0.053	<b>545.0</b>	4	0.234	<b>545.0</b>	4	2.174
14				T2	589.0	4	<b>545</b>	3	< 20min	<b>545.0</b>	3	0.326	<b>545.0</b>	3	0.061	<b>545.0</b>	3	0.287	<b>545.0</b>	4	1.807
15			C2	T1	593.0	3	<b>452</b>	3	< 20min	<b>452.0</b>	3	0.14	<b>452.0</b>	3	0.069	471.0	3	0.292	<b>452.0</b>	3	3.016
16				T2	583.0	4	<b>447</b>	3	< 20min	<u><b>447.0</b></u>	3	0.793	476.0	3	0.354	489.0	3	0.259	476.0	3	2.859
17	30	10	C1	T1	975.0	7	N / A	N / A	N / A	856.0	7	0.174	856.0	7	0.064	817.0	8	0.574	<u><b>776.0</b></u>	8	3.83
18				T2	948.0	8	N / A	N / A	N / A	<u><b>779.0</b></u>	7	0.061	793.0	7	0.164	784.0	8	0.537	781.0	8	4.287
19			C2	T1	766.0	5	N / A	N / A	N / A	623.0	5	0.124	631.0	5	0.076	627.0	5	0.606	<u><b>620.0</b></u>	5	6.068
20				T2	845.0	6	N / A	N / A	N / A	<u><b>629.0</b></u>	5	0.195	637.0	5	0.085	634.0	5	0.703	634.0	5	6.216

21	50	15	C1	T1	1436.0	11	N / A	N / A	N / A	1205.0	11	0.186	1207.0	11	0.065	1162.0	11	0.988	<b><u>1158.0</u></b>	8	7.402
22				T2	1526.0	12	N / A	N / A	N / A	1181.0	11	0.238	1176.0	11	0.061	1162.0	11	1.109	<b><u>1152.0</u></b>	11	7.509
23			C2	T1	1096.0	7	N / A	N / A	N / A	910.0	7	0.131	921.0	7	0.069	940.0	7	1.174	<b><u>907.0</u></b>	7	12.77
24				T2	1177.0	7	N / A	N / A	N / A	910.0	7	0.28	929.0	7	0.067	936.0	8	1.162	<b><u>892.0</u></b>	7	12.02
25	80	30	C1	T1	2447.0	20	N / A	N / A	N / A	2204.0	20	0.191	2192.0	20	0.057	2321.0	20	3.76	<b><u>2099.0</u></b>	21	14.87
26				T2	2511.0	21	N / A	N / A	N / A	2165.0	20	0.232	2114.0	20	0.06	2353.0	20	1.478	<b><u>2064.0</u></b>	21	14.19
27			C2	T1	1699.0	12	N / A	N / A	N / A	<b><u>1459.0</u></b>	12	0.172	1494.0	12	0.077	1616.0	12	5.822	1534.0	12	11.19
28				T2	1761.0	13	N / A	N / A	N / A	1452.0	12	0.42	<b><u>1448.0</u></b>	12	0.085	1677.0	12	6.905	1537.0	13	9.557
29	100	35	C1	T1	2902.0	24	N / A	N / A	N / A	2475.0	24	0.167	2476.0	24	0.062	2667.0	24	6.351	<b><u>2418.0</u></b>	24	13.332
30				T2	2588.0	24	N / A	N / A	N / A	2457.0	24	0.367	<b><u>2387.0</u></b>	24	0.065	2504.0	24	1.583	2402.0	24	3.751
31			C2	T1	2049.0	14	N / A	N / A	N / A	1923.0	14	0.166	1804.0	14	0.08	1907.0	14	8.103	<b><u>1787.0</u></b>	15	20.106
32				T2	2043.0	16	N / A	N / A	N / A	<b><u>1638.0</u></b>	14	0.297	1677.0	15	0.085	1967.0	16	2.839	1705.0	15	18.196

N / A : L'algorithm Branch & Bound n'a pas été en mesure de trouver une solution optimale dans un temps raisonnable.

*Tableau 3.2 : résultats obtenus pour les différentes instances*

\*

Les résultats montrent que le coût (distance totale parcouru) augmente avec le nombre de clients. Comme on pouvait s'y attendre, nous constatons aussi, que pour chaque instance, l'augmentation des capacités des véhicules aura pour conséquence la satisfaction de plus de clients pour chaque circuit. Ceci permettra d'élaborer des tournées plus efficaces et plus optimisées et ainsi de réduire les coûts.

Lorsque nous ajoutons la contrainte de la durée maximale  $T_2$ , nous remarquons une légère amélioration des résultats. Cette limitation de la durée permet de réduire le nombre de points de collecte dans chaque tournée. En négligeant les longs circuits, les métaheuristiques seont plus efficaces et arriveront à la meilleure solution rapidement.

### **5.3.1. La validation des résultats par la méthode exacte Branch & Bound :**

Nous remarquons clairement à travers le *Tableau 3.2* que l'algorithme Branch & Bound a été en mesure de résoudre toutes les instances de test jusqu'à 15 nœuds (clients). De plus, à mesure que l'ampleur des problèmes (instances) augmente, les temps d'exécution de Branch & Bound augmentent considérablement d'une manière exponentielle. A partir des instances moyennes de 30 clients et plus, l'algorithme B&B ne peut même pas fournir de solutions réalisables.

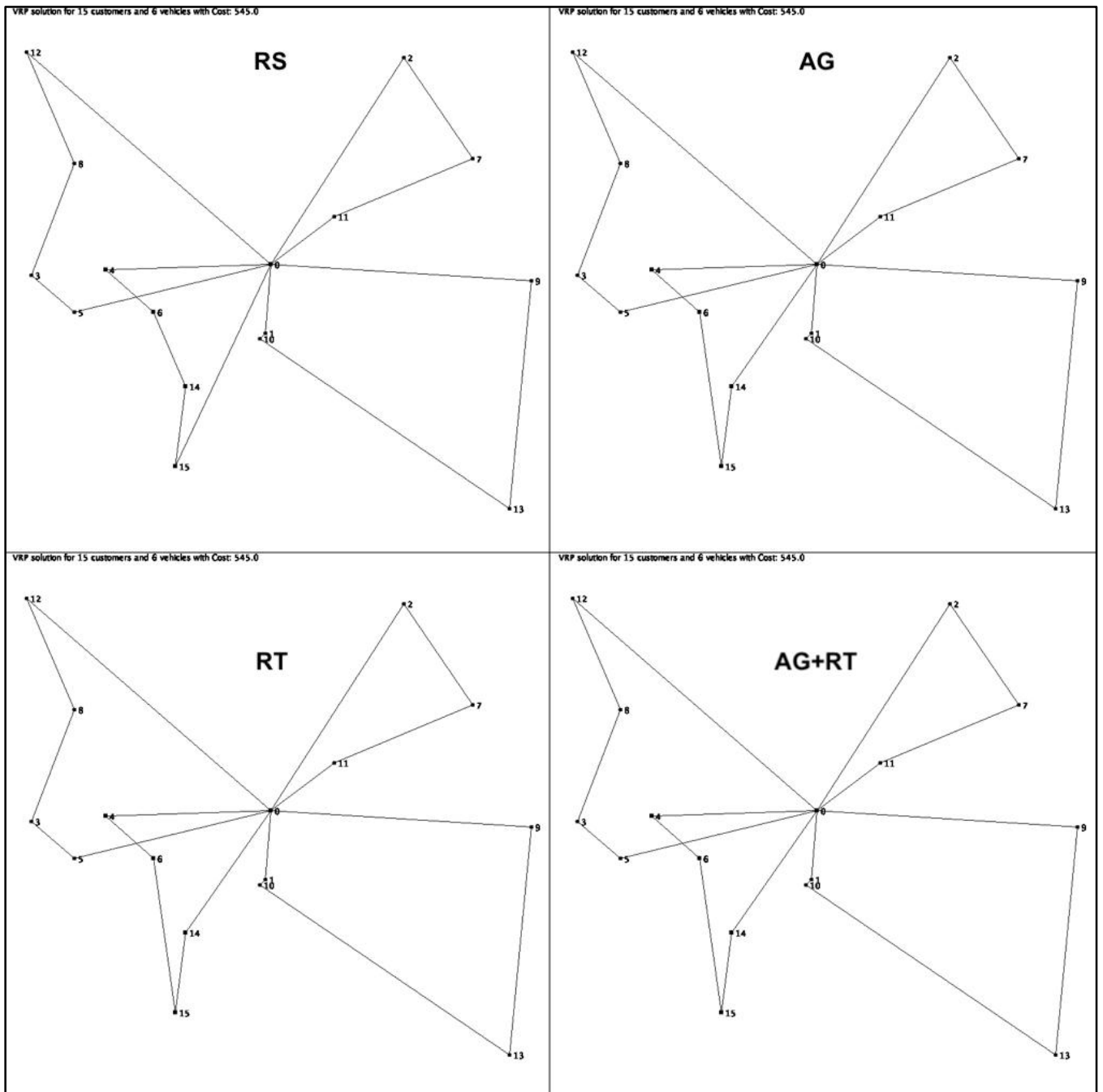
Par ailleurs, nous remarquons pour les petites instances (moins de 30 clients) que les quatre métaheuristiques donnent (à quelques exceptions) des résultats égaux à ceux de la méthode exacte B&B avec évidemment des temps d'exécution largement inférieurs.

La fiabilité et la validité des quatre métaheuristiques ont été prouvées à travers cette comparaison, nous pouvons donc adopter ces métaheuristiques pour les moyennes et grandes instances de notre problème VRPTWDC.

### **5.3.2. Résultats pour les moyennes et grandes instances**

**Dans les moyennes instances ( $30 < N < 80$ ) :** nous constatons que la métaheuristique hybride AG+RT donne les meilleurs résultats par rapport aux autres métaheuristiques pour 6 sur 8 instances, suivi de la métaheuristique RS qui donne des résultats meilleurs pour 2 sur 8 instances.

Les *Figures 3.6 et 3.7*, illustrent les tournées obtenues par chaque métaheuristique pour quelques instances. Nous pouvons remarquer instinctivement à travers ces figures (*3.6 et 3.7*) que les solutions sont semblables à une hélice où les pales représentent les tournées et son axe représente le dépôt. Les meilleures solutions sont souvent celles qui retracent les hélices les mieux dessinées.



*Figure 3.6 : Solutions instance N°13*

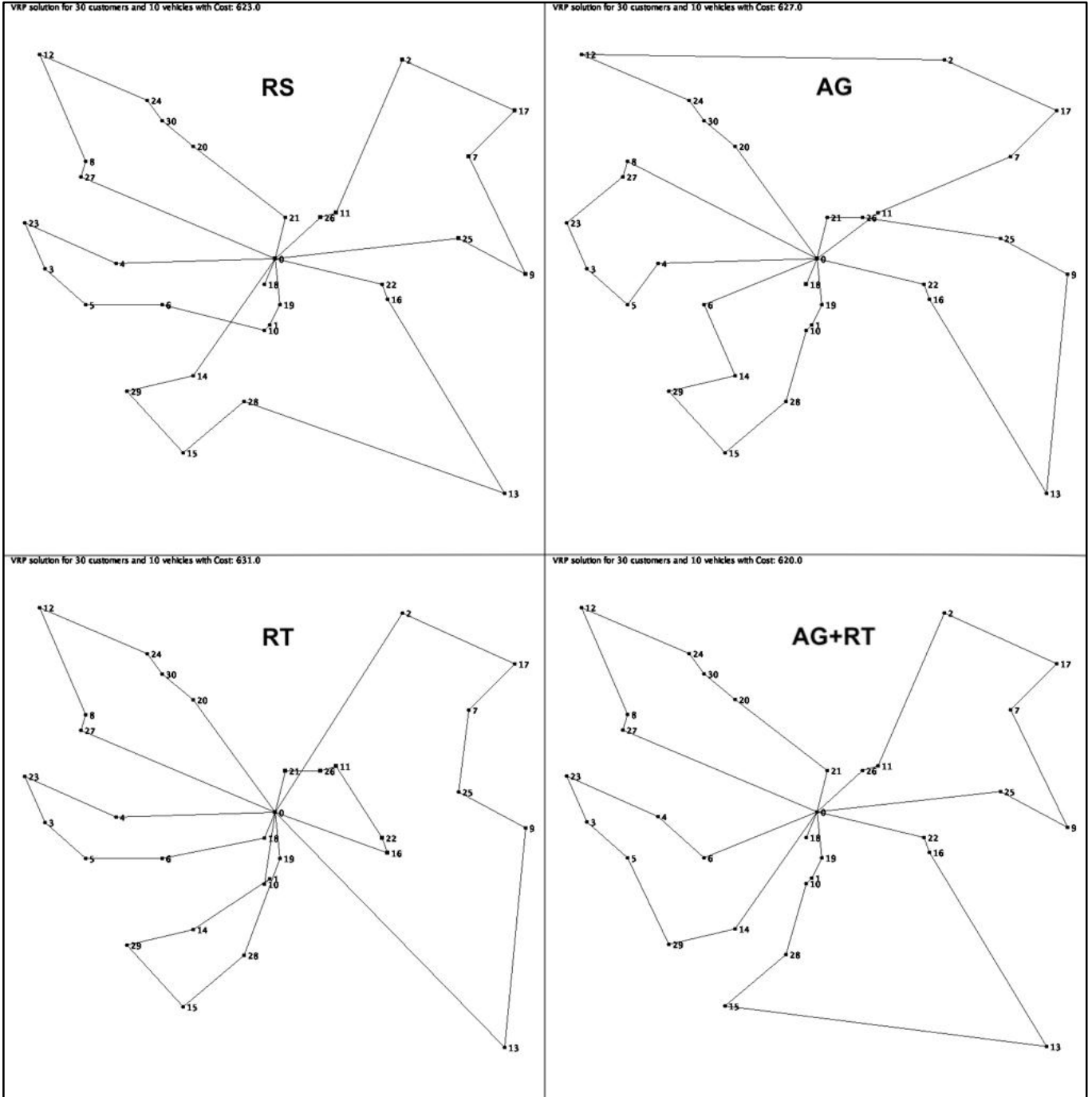


Figure 3.7 : Solutions instance N°19

**Dans les grandes instances (N>80) :** nous remarquons que la métaheuristique hybride AG+RT est plus performante pour 4 sur les 8 instances, donnant les meilleurs résultats malgré un temps d'exécution plus élevé par rapport aux autres métaheuristicues. Les méthodes RT et RS viennent en deuxième position donnant les meilleurs résultats pour 2 sur les 8 instances de grande taille.

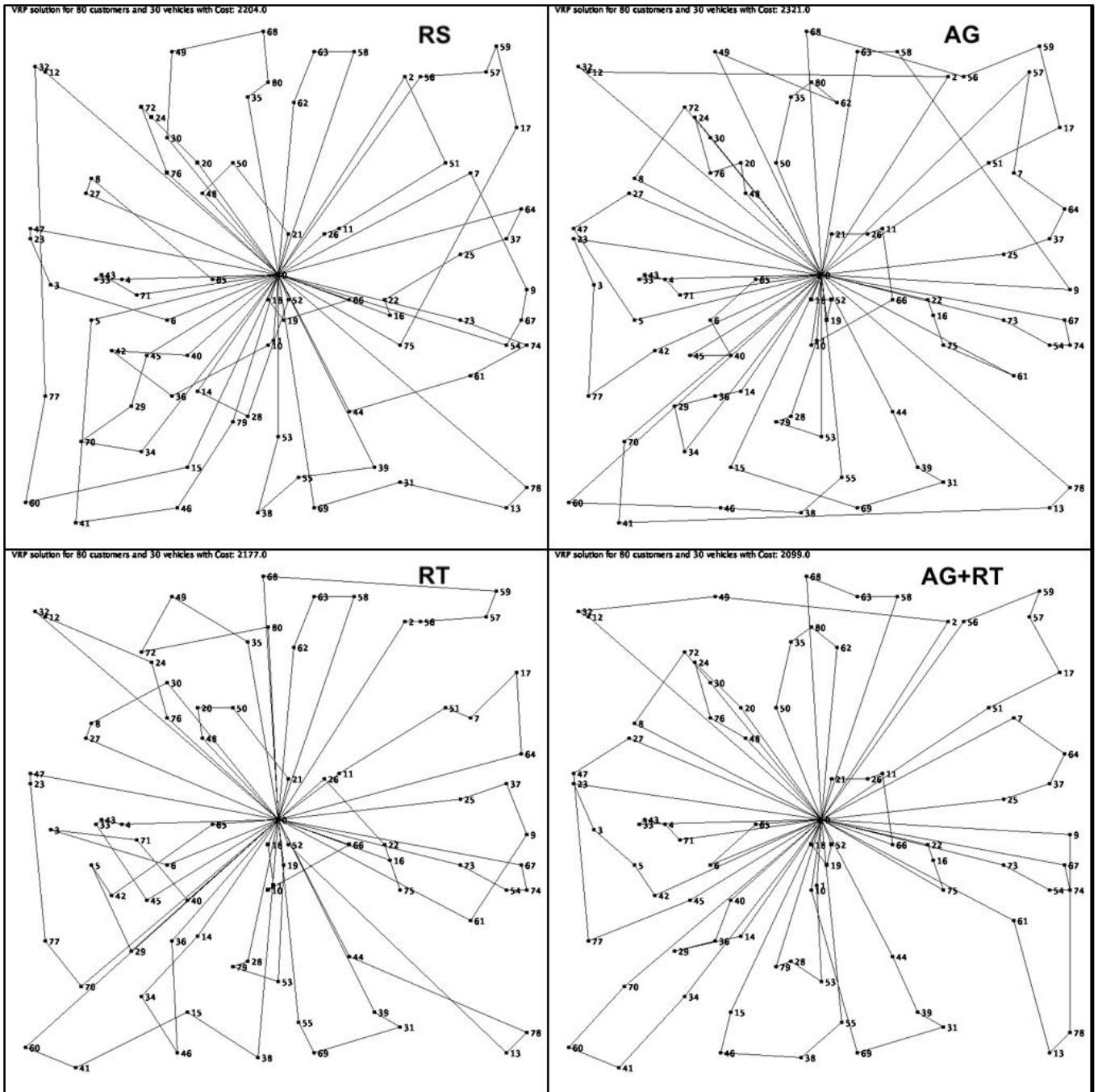


Figure 3.8 Solutions instance N°25

### 5.3.3. Passage d'une solution gloutonne vers une solution d'une métaheuristique :

Les figures ci-dessus montrent clairement les améliorations apportées à la solution initiale gloutonne après exécution des différentes métaheuristicues. Les deux figure (3.9 et 3.10) suivantes concernent la différence entre la solution gloutonne et la métaheuristique pour l'instance N°01 et l'instance N°09.

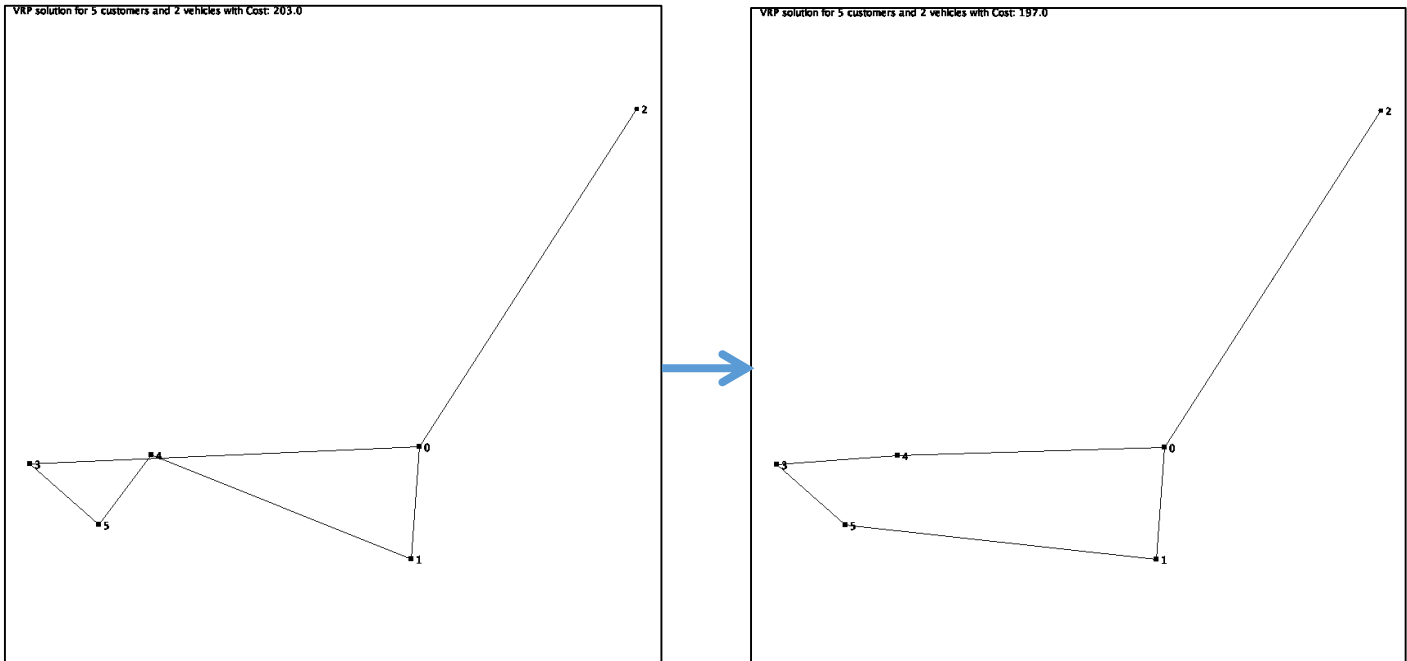


Figure 3.9 : Passage d'une solution gloutonne vers une solution d'une métaheuristique instance N°1

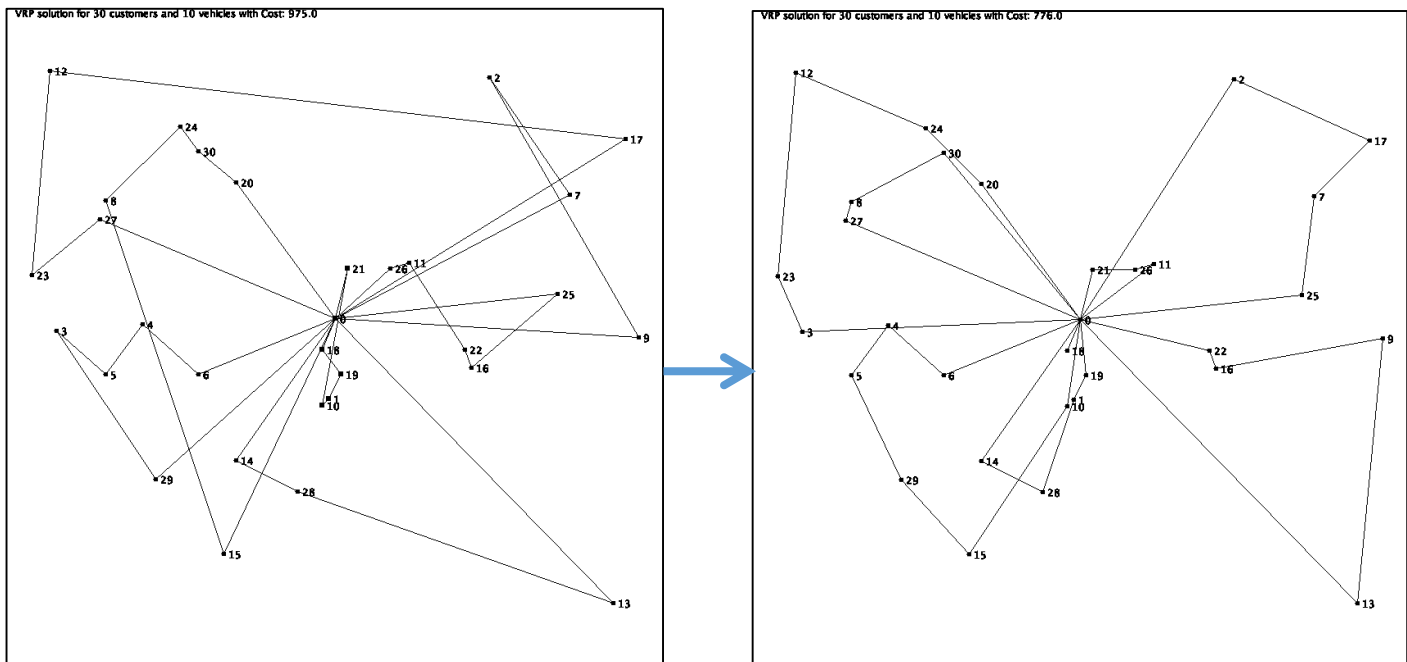


Figure 3.10 : Passage d'une solution gloutonne vers une solution d'une métaheuristique instance N°17

## 6. Conclusion :

Dans ce chapitre nous avons proposé un nouveau modèle baptisé VRPTWDC qui représente une variante du problème VRP intégrant simultanément les contraintes de fenêtres de temps pour les visites des clients et les contraintes de durée maximale pour les tournées. Ce modèle a plusieurs applications importantes notamment dans la livraison de colis, la distribution des produits périssables, la collecte des productions agricoles (lait, les produits céréaliers, les produits maraîchers ...), le ramassage scolaire et le service bancaire.

Pour la résolution du problème VRPTWDC qui est considéré comme NP-difficile, nous avons proposé quatre approches de résolution basées sur les métaheuristiques suivantes : Recuit simulé (RS), Recherche Tabou (RT), Algorithme génétique (AG) et une méthode hybridant un algorithme génétique avec la recherche tabou (AG+RT). Les différentes approches de résolution proposées ont été détaillées et expliquées dans la deuxième partie du chapitre.

Des expériences numériques ont été réalisées et analysées sur des instances de différentes tailles (petites, moyennes et grandes). Pour les petites instances, nous avons constaté que les métaheuristiques donnent des résultats quasi identiques à la méthode exacte Branch & Bound. Ceci atteste de la fiabilité des métaheuristiques proposées pour la résolution du problème VRPTW. Pour les moyennes et grandes instances, les résultats obtenus montrent que la métaheuristique hybride AG+RT est la plus performante suivi des métaheuristiques RS, RT et AG respectivement.

Afin de valider et concrétiser l'application du modèle VRPTWDC et les approches de résolution proposés, une étude d'une application industrielle, issue de la collecte de lait frais pour l'entreprise DDA sera présentée et les expériences numériques analysées dans le chapitre qui suit.



## **Chapitre IV**

### **Cas d'application : Amélioration de la tournée de collecte de lait frais pour l'entreprise DDA**

# 1. Introduction

Afin de valider le modèle VRPTWDC et les approches de résolution proposées, nous présentons dans ce chapitre un cas d'application réel qui concerne l'amélioration des tournées de collecte de lait frais pour la laiterie DDA. Pour cela, nous présentons une description détaillée du cas d'application étudié. Les données réelles sont exposées dans la deuxième partie du chapitre. Enfin, nous présentons et interprétons les résultats numériques en montrant les améliorations obtenus par chaque métaheuristique.

## 2. Description du cas d'application étudié

Notre cas d'application concerne l'amélioration de la tournée de collecte de lait de l'entreprise de produits laitiers DDA.

La collecte de lait frais au niveau de la laiterie DDA se fait quotidiennement en envoyant des camions aux centres de collecte (CC). Les centres de collecte sont au nombre de 43 et sont dispersés sur l'ensemble du territoire national, ils peuvent être de deux types :

- (1) grands éleveurs de vaches qui produisent quotidiennement une grande quantité de lait cru.
- (2) centres intermédiaires qui regroupent les productions des petits éleveurs proches du centre de collecte. Les petits éleveurs avec qui travail DDA sont environs 1500 éleveurs au niveau national.

La collecte se fait au niveau des centres de collecte par le biais de camions (équipés de citerne de lait). Ces camions sont sous-traités auprès de deux prestataires de transport logistique : TMF Logistics (Transport de Marchandises et Frigorifiques) et FB (Flèche Bleu). Le coût de transport au kilomètre est propre à chaque prestataire logistique.

La production des éleveurs varie d'une saison à une autre, ainsi le nombre de CC concerné par la collecte change à travers les saisons. Dans notre étude, nous nous sommes intéressés à la collecte de lait durant la saison d'été 2020, qui concerne en tout 43 CC. Nous précisons que la collecte est organisée sur deux jours : (1) la tournée de collecte des jours impaires du mois (c.-à-d. 1<sup>er</sup>, 3, 5, 7...*etc*) qui implique 35 CC, et (2) la tournée des jours paires (c.-à-d. 2, 4, 6,

8...etc) qui implique 36 CC. Notons que les deux tournées ont des CC en commun c.-à-d. des CC qui sont collectés chaque jour.

### 3. Les données réelles du cas d'application étudié

Dans cette partie nous exposons les données réelles du cas d'application étudié :

#### 3.1. Centres de Collectes (désignation des différents points de collecte)

*Ain arnat, Ain beida, EL MAADER, Temouchent, Annaba, Barika, Beida bordj, YEKHLEF, BABA ALI, CHOUKRANE, FERME DDA, BOUSSOUF, Djelfa, TAFRAOUI, D.MAOUCHI, FESDIS, Freha, Ghardaia, El Kennar, BAGIA, Khenchela, Laghouat, Makouda, Chlef, Medea, Oued el Djemaa, DJDIOUIIA, Ras El Oued, Seriana, Skikda, FERME TIDJANI, Souk Ahras, , CAZEL, Oued Seguen, TLEMCEN, TLILAT, Maghnia, Blida 1, Hadji blida, , Tiaret, Saida, DIDOUCHE, DOUAOUDA.*

#### 3.2. Productions moyennes journalières respectivement

AKBOU (dépôt)	Ain arnat	Ain beida	EL MAADER	Temouchent	Annaba	Barika	Beida bordj
0	2000	3000	1000	20000	2000	4000	4000
D.MAOUCHI	FESDIS	Freha	Ghardaia	El Kennar	BAGIA	Khenchela	Laghouat
3000	3000	8000	10000	5000	3000	4000	2000
Skikda	FERME TIDJANI	Souk Ahras	CAZEL	Oued Seguen	TLEMCEN	TLILAT	Maghnia
6000	1500	4000	1200	2000	4000	5000	3000
YEKHLEF	BABA ALI	CHOUKRANE	FERME DDA	BOUSSOUF	Djelfa	TAFRAOUI	Makouda
2000	1000	3000	1500	12000	2000	2000	9000
Chlef	Medea	Oued el Djemaa	DJDIOUIIA	Ras El Oued	Seriana	Blida 1	Hadji blida
600	2000	5000	4000	2000	5000	2000	1500
Tiaret	Saida	Didouche	Douaouda				
2000	4000	6000	2000				

Tableau 4.1 : Productions moyennes journalières des CC de DDA

### 3.3. Fenêtres de temps des centres de collecte

La laiterie DDA considère uniquement le temps de début de service au plus tôt (borne inférieure des fenêtres de temps):

AKBOU	Ain arnat	Ain beida	EL MAADER	Temouchent	Annaba	Barika	Beida bordj
o	12h	11h	10h	14h	11h	12h 30min	11h
D.MAOUCHI	FESDIS	Freha	Ghardaia	El Kennar	BAGIA	Khenchela	Laghouat
10h	11h	14h	11h	10h	11h	11h	13h
Skikda	FERME TIDJANI	Souk Ahras 2	CAZEL	Oued Seguen	TLEMCEN	TLILAT	Maghnia
11h	14h	13h	11h	11h	12h	11h	14h
YEKHLEF	BABA ALI	CHOUKRANE	FERME DDA	BOUSSOUF	Djelfa	TAFRAOUI	Makouda
11h	12h	12h	12h	13h	13h	14h	11h
Chlef	Medea	Oued el Djemaa	DJDIOUIIA	Ras El Oued	Seriana	Blida 1	Hadji blida
11h	12h	13h	11	11h	11h	11h	11h
Tiaret	Saida	Didouche	Douaouda				
11h	11h	11h	10h				

Tableau 4.2 : Temps de service au plus tôt des CC de DDA

### 3.4. Temps de chargement moyen

- Temps de stationnement : 15 min
- Test de qualité du lait : 20 min
- Vitesse de chargement du lait au camion: 220 litres/min

### 3.5. Temps maximale d'une tournée

Il est de 24 heures et notons que tous les paramètres de temps sont convertis en minute dans notre application d'optimisation des problèmes de tournée de véhicules.

### 3.6. Matrice des distances

La matrice de distance représente les distances entre chaque paire de CC. Nous avons utilisé l'API gratuit de *OpenStreetMap* pour générer les 45x45 distances. Nous avons considéré par ailleurs que les distances sont symétriques c.-à-d. la distance entre deux centres 'A' et 'B' est égale à la distance entre 'B' et 'A'. De ce fait, nous obtiendrons une matrice de distance carrée et symétrique.

Par soucis de visibilité, nous avons découpé la matrice en deux parties (*Tableaux 4.3 et 4.4*) :





### 3.7. Détails sur les camions et leurs tournées actuelles chez DDA

Les deux *Tableaux 4.5 et 4.6*, illustrent les tournées actuelles des journées paires et impaires telles qu'elles sont programmées par DDA ainsi que les détails (numéro et capacité) sur les camions impliqués dans ces tournées ainsi que le cout de transport imposé par les chaque prestataire.

Journée paires								
SOUS-TRAITANT	Coût de transport	NUMERO	CAPACITE	CIRCUIT				
TMF	48 DA/Km	C24	24 000	TIMOUCHEM				
		C14	24 000	GHARDAIA	LAGHOUATE	DJELFA		
		C18	18 750	YEKHEF	AIN BEIDA	FERME TIDJANI	SOUK AHRAS	
		C21	22 000	ANNABA	SIKDA	DIDOUCHE	OUED SEGUEN	
		C23	22 000					
FB	42,14 DA/Km	C12	22 000	EL MAADER	DOUAOUA	BARIKA	REO	AIN ARNAT
		C15	22 000	EL KENNAR	CAZEL	D.MAOUCHI		
		C26	25 000	BOUSSOUF	FERME DDA	CHOUKRANE		
		C27	25 000	B.BORDJ	BAGIA	KHENCHELA	FESDIS	SARIANA
		C28	25 000	TLEMCEN	TAFRAOUI	TLILAT	OUED DJEMAA	DJDIOUIA
		C13	22 000					
MOUSSA	42,14 DA/Km	C30	18 000	MAKOUDA	FREHA			

*Tableau 4.5 : Détails sur les camions de DDA et de leurs circuits pour les jours paires*



Journée impaires									
SOUS-TRAITANT	Coût de transport	NUMERO	CAPACITE	CIRCUIT					
TMF	4,8 DA/Km	C24	24 000	TAFRAOUI	MAGHNIA	TLEMCEN	TLILAT		
		C14	24 000	ANNABA	SKIKDA	DIDOUCHE	OUED SEGUEN		
		C18	18 750	EL MAADER	DOUAOUDA	BARIKA	REO	AIN ARNAT	
		C21	22 000						
		C23	22 000						
FB	42,14 DA/Km	C12	22 000	BLIDA	BABA ALI	HADJI	MEDEA	CHLEF	
		C15	22 000	TIARET	SAIDA	OUED DJEMAA	DJDIOUIA		
		C13	22 000	EL KENNAR	CAZEL	D.MAOUCHI			
		C26	25 000	BOUSSOUF	FERME DDA	CHOUKRANE			
		C27	25 000	B.BORDJ	FESDIS	KHENCHELA			
		C28	25 000	TIMOUCHEUTE					
MOUSSA	42,14 DA/Km	C30	18 000	MAKOUDA	FREHA				

Tableau 4.6 : Détails sur les camions de DDA et de leurs circuits pour les jours impaires

## 4. Présentation des résultats

Dans cette section nous détaillons et interprétons les résultats obtenus. Après avoir intégré toutes les données nécessaires pour notre étude dans notre application et exécuté les quatre approches de résolution à base des métaheuristiques : RS, RT, AG, AG+RT, nous avons obtenu les solutions de chaque métaheuristique qui seront comparées avec les tournées actuelles adoptées par le département Collecte de l'entreprise DDA.

Ainsi, nous allons présenter les itinéraires que devra suivre chaque camion au niveau de chaque tournée permettant d'améliorer la distance et le coût logistique par rapport aux tournées actuelles adoptées par DDA.

### 4.1. La tournée actuelle adoptée par DDA

Ci-dessous la distance totale parcourue et le coût logistique (coût de transport) des tournées actuelles adoptées par DDA :

#### **Journée Paire :**

- Solution Distance (distance totale parcourue) 8923 KM
- Solution Cost (coût logistique) 403768,16 DA

#### **Journée impaire :**

- Solution Distance 8603 KM
- Solution Cost 344350,68 DA

### 4.2. Les tournées obtenues par chaque métaheuristique

Les tournées obtenues par notre application pour chaque métaheuristique sont présentées comme suit :

**Ramarque :** l'affichage des résultats est présenté selon le format suivant :

Vehicle K Cn : **AKBOU** (temps de départ du dépôt) -> (temps d'arrivée réelle au CC)(temps d'arrivée au plus tôt) **Nom du Point de Collecte** (temps de départ du CC après chargement) -> ...etc...-> (heure d'arrivée au dépôt) **AKBOU. Quantité de lait collectée Capacité du camion utilisé**

----- Journée Paire -----

**Solution After Tabu Search**

1. Vehicle 1 C14: (6:0)AKBOU(6:0) -> (14:17)(14:0)Temouchent(16:22) -> (17:17)(11:0)TAFRAOUI(18:1) -> (1:19)AKBOU 22000 24000
2. Vehicle 3 C21: (7:0)AKBOU(7:0) -> (9:20)(10:0)EL MAADER(p)(10:39) -> (13:47)(14:0)Makouda(15:15) -> (15:53)(14:0)Freha(17:4) -> (17:53)AKBOU 18000 22000
3. Vehicle 4 C12: (6:0)AKBOU(6:0) -> (16:1)(13:0)Ghardaia(17:21) -> (22:9)(11:0)Laghouat(22:53) -> (0:17)(11:0)Djelfa(1:1) -> (2:24)(10:0)DOUAOUDA(3:8) -> (5:34)AKBOU 16000 22000
4. Vehicle 5 C15: (7:0)AKBOU(7:0) -> (10:29)(11:0)FESDIS(11:48) -> (13:24)(12:0)Khenchela(14:17) -> (14:40)(11:0)BAGIA(15:28) -> (17:57)(13:0)Seriana(18:54) -> (20:14)(12:30)Barika(21:7) -> (23:34)(12:0)CAZEL(0:14) -> (1:23)AKBOU 20200 22000
5. Vehicle 6 C26: (6:0)AKBOU(6:0) -> (10:47)(11:0)Ain beida(11:48) -> (13:16)(11:0)Souk Ahras 2(14:9) -> (14:24)(11:0)FERME TIDJANI(15:5) -> (16:17)(11:0)Annaba(17:1) -> (18:29)(14:0)Skikda(19:31) -> (21:24)(11:0)El Kennar(22:21) -> (23:40)(13:0)D.MAOUCHI(0:28) -> (1:1)AKBOU 24500 25000
6. Vehicle 7 C27: (6:0)AKBOU(6:0) -> (14:45)(11:0)TLEMENEN(15:38) -> (17:21)(11:0)TLILAT(18:18) -> (19:50)(12:0)Oued el Djemaa(20:47) -> (21:10)(12:0)DJDIIOUIIA(22:3) -> (3:24)AKBOU 18000 25000
7. Vehicle 8 C28: (7:0)AKBOU(7:0) -> (9:23)(10:0)YEKHEF(10:44) -> (11:30)(13:0)Oued Seguen(13:44) -> (14:20)(11:0)DIDOUICHE(15:22) -> (15:44)(10:0)BOUSSOUF(17:13) -> (17:27)(11:0)FERME DDA(18:8) -> (21:36)AKBOU 23500 25000
8. Vehicle 9 C30: (8:0)AKBOU(8:0) -> (10:39)(11:0)Beida bordj(11:53) -> (12:46)(12:0)Ain arnat(13:30) -> (14:11)(13:0)Ras El Oued(14:55) -> (16:23)(14:0)CHOUKRANE(17:11) -> (17:28)AKBOU 11000 18000

**Solution Distance** 7883.999999999996

**Solution Cost** 342955.5599999998

**Opération effectuée en:** 0.19 secondes.

## Solution After Simulated Annealing

1. Vehicle 0 C24: (7:0)AKBOU(7:0) -> (10:28)(11:0)FERME DDA(11:41) -> (11:55)(10:0)BOUSSOUF(13:24) -> (13:46)(11:0)DIDOUCHE(14:48) -> (15:24)(13:0)Oued Seguen(16:8) -> (17:35)(12:0)Ain arnat(18:19) -> (20:9)AKBOU  
23500 24000
2. Vehicle 1 C14: (7:0)AKBOU(7:0) -> (9:20)(10:0)EL MAADER(p)(10:39) -> (17:46)(11:0)TAFRAOUI(18:30) -> (19:25)(14:0)Temouchent(21:30) -> (5:47)AKBOU  
23000 24000
3. Vehicle 4 C12: (6:0)AKBOU(6:0) -> (11:12)(11:0)Laghouat(11:56) -> (16:44)(13:0)Ghardaia(18:4) -> (0:17)(11:0)Djelfa(1:1) -> (2:24)(10:0)DOUAOUDA(3:8) -> (5:34)AKBOU 16000 22000
4. Vehicle 5 C15: (7:0)AKBOU(7:0) -> (9:23)(10:0)YEKHFLEF(10:44) -> (13:40)(11:0)Annaba(14:24) -> (15:36)(11:0)FERME TIDJANI(16:17) -> (16:32)(11:0)Souk Ahras 2(17:25) -> (18:53)(11:0)Ain beida(19:41) -> (22:47)(13:0)Seriana(23:44) -> (1:4)(12:30)Barika(1:57) -> (4:21)AKBOU 21500 22000
5. Vehicle 6 C26: (7:0)AKBOU(7:0) -> (10:29)(11:0)FESDIS(11:48) -> (13:1)(11:0)BAGIA(13:49) -> (14:12)(12:0)Khenchela(15:5) -> (18:17)(14:0)Skikda(19:19) -> (21:12)(11:0)El Kennar(22:9) -> (23:28)(13:0)D.MAOUCHI(0:16) -> (0:49)AKBOU 24000 25000
6. Vehicle 7 C27: (6:0)AKBOU(6:0) -> (13:5)(11:0)TLILAT(14:2) -> (15:45)(11:0)TLEMCEN(16:38) -> (19:50)(12:0)Oued el Djemaa(20:47) -> (21:10)(12:0)DJDIQUIA(22:3) -> (2:58)(14:0)CHOUKRANE(3:46) -> (4:3)AKBOU  
21000 25000
7. Vehicle 8 C28: (8:0)AKBOU(8:0) -> (10:39)(11:0)Beida bordj(11:53) -> (12:46)(13:0)Ras El Oued(13:44) -> (15:22)(12:0)CAZEL(16:2) -> (18:6)(14:0)Makouda(19:21) -> (19:59)(14:0)Freha(21:10) -> (21:59)AKBOU 24200 25000

**Solution Distance** 7825.999999999971

**Solution Cost** 341982.2999999995

**Opération effectuée en:** 0.321 secondes.

## Best Solution After Genetic Algorithm

1. Vehicle 0 C24: (8:0)AKBOU(8:0) -> (9:52)(11:0)El Kennar(11:57) -> (12:40)(12:0)CAZEL(13:20) -> (13:56)(13:0)D.MAOUCHI(14:44) -> (16:14)(14:0)Makouda(17:29) -> (18:52)(14:0)CHOUKRANE(19:40) -> (19:57)AKBOU  
21200 24000
2. Vehicle 1 C14: (7:0)AKBOU(7:0) -> (9:23)(10:0)YEKHFLEF(10:44) -> (11:30)(13:0)Oued Seguen(13:44) -> (15:16)(14:0)Skikda(16:18) -> (17:16)(11:0)DIDOUICHE(18:18) -> (18:25)(11:0)FERME DDA(19:6) -> (20:53)(12:0)Ain arnat(21:37) -> (23:27)AKBOU  
19500 24000
3. Vehicle 2 C18: (7:0)AKBOU(7:0) -> (9:20)(10:0)EL MAADER(p)(10:39) -> (10:49)(10:0)DOUAOUUDA(11:33) -> (14:1)(13:0)Seriana(14:58) -> (16:18)(12:30)Barika(17:11) -> (18:10)(11:0)Beida bordj(19:3) -> (19:56)(13:0)Ras El Oued(20:40) -> (22:9)AKBOU 18000 18750
4. Vehicle 4 C12: (6:0)AKBOU(6:0) -> (9:43)(10:0)BOUSSOUF(11:29) -> (16:0)(14:0)Freha(17:11) -> (18:0)AKBOU 20000 22000
5. Vehicle 5 C15: (7:0)AKBOU(7:0) -> (10:29)(11:0)FESDIS(11:48) -> (13:24)(12:0)Khenchela(14:17) -> (14:40)(11:0)BAGIA(15:28) -> (16:4)(11:0)Ain beida(16:52) -> (18:5)(11:0)FERME TIDJANI(18:46) -> (19:1)(11:0)Souk Ahras 2(19:54) -> (21:21)(11:0)Annaba(22:5) -> (3:25)AKBOU 20500 22000
6. Vehicle 6 C26: (6:0)AKBOU(6:0) -> (11:21)(12:0)DJDIUUIIA(12:53) -> (13:16)(12:0)Oued el Djemaa(14:13) -> (15:45)(11:0)TLILAT(16:42) -> (16:59)(11:0)TAFRAOUI(17:43) -> (1:1)AKBOU 16000 25000
7. Vehicle 7 C27: (6:0)AKBOU(6:0) -> (14:17)(14:0)Temouchent(16:22) -> (17:31)(11:0)TLEMCEN(18:24) -> (3:9)AKBOU 24000 25000
8. Vehicle 8 C28: (6:0)AKBOU(6:0) -> (11:12)(11:0)Laghouat(11:56) -> (16:44)(13:0)Ghardaia(18:4) -> (0:17)(11:0)Djelfa(1:1) -> (4:49)AKBOU 14000 25000

**Solution Distance** 7967.0

**Solution Cost** 347414.21999999999

**Opération effectuée en:** 9.01 secondes.

## Best Solution After Hybrid Genetic Algorithm with Tabu Search

1. Vehicle 0 C24: (7:0)AKBOU(7:0) -> (9:23)(10:0)YEKHFLEF(10:44) -> (11:30)(13:0)Oued Seguen(13:44) -> (14:30)(10:0)BOUSSOUF(15:59) -> (16:13)(11:0)FERME DDA(16:54) -> (18:41)(12:0)Ain arnat(19:25) -> (20:25)(12:0)CAZEL(21:5) -> (21:41)(13:0)D.MAOUCHI(22:29) -> (23:2)AKBOU 23700 24000
2. Vehicle 3 C21: (8:0)AKBOU(8:0) -> (10:39)(11:0)Beida bordj(11:53) -> (12:46)(13:0)Ras El Oued(13:44) -> (15:13)AKBOU 6000 22000
3. Vehicle 4 C12: (6:0)AKBOU(6:0) -> (11:12)(11:0)Laghouat(11:56) -> (16:44)(13:0)Ghardaia(18:4) -> (0:17)(11:0)Djelfa(1:1) -> (4:39)(14:0)CHOUKRANE(5:27) -> (5:44)AKBOU 17000 22000
4. Vehicle 5 C15: (7:0)AKBOU(7:0) -> (9:20)(10:0)EL MAADER(p)(10:39) -> (10:49)(10:0)DOUAOUDA(11:33) -> (14:47)(14:0)Makouda(16:2) -> (16:40)(14:0)Freha(17:51) -> (18:40)AKBOU 20000 22000
5. Vehicle 6 C26: (7:0)AKBOU(7:0) -> (10:29)(11:0)FESDIS(11:48) -> (13:23)(11:0)Ain beida(14:11) -> (14:47)(11:0)BAGIA(15:35) -> (15:58)(12:0)Khenchela(16:51) -> (19:43)(13:0)Seriana(20:40) -> (22:0)(12:30)Barika(22:53) -> (1:17)AKBOU 22000 25000
6. Vehicle 7 C27: (6:0)AKBOU(6:0) -> (14:17)(14:0)Temouchent(16:22) -> (17:31)(11:0)TLEMEN(18:24) -> (3:9)AKBOU 24000 25000
7. Vehicle 8 C28: (7:0)AKBOU(7:0) -> (10:32)(11:0)DIDOUICHE(12:2) -> (13:50)(11:0)Annaba(14:34) -> (15:46)(11:0)FERME TIDJANI(16:27) -> (16:42)(11:0)Souk Ahras 2(17:35) -> (19:48)(14:0)Skikda(20:50) -> (22:43)(11:0)El Kennar(23:40) -> (1:32)AKBOU 24500 25000
8. Vehicle 9 C30: (6:0)AKBOU(6:0) -> (11:21)(12:0)DJDIUIIA(12:53) -> (13:16)(12:0)Oued el Djemaa(14:13) -> (15:45)(11:0)TLILAT(16:42) -> (16:59)(11:0)TAFRAOUI(17:43) -> (1:1)AKBOU 16000 18000

**Solution Distance** 7737.0

**Solution Cost** 332072.9799999998

**Opération effectuée en:** 38.2 secondes.

----- Journée impaire -----

**Solution After Tabu Search**

1. Vehicle 3 C12: (8:0)AKBOU(8:0) -> (10:35)(11:0)BABA ALI(11:39) -> (12:19)(11:0)Hadji blida (13:0) -> (13:12)(11:0)Blida 1(13:56) -> (15:57)(14:0)Makouda(17:12) -> (17:53)(14:0)Freha(19:4) -> (19:57)AKBOU 21500 22000
2. Vehicle 4 C15: (6:0)AKBOU(6:0) -> (9:46)(11:0)FESDIS(11:48) -> (13:8)(11:0)BAGIA(13:56) -> (14:21)(12:0)Khenchela(15:14) -> (18:20)(13:0)Seriana(19:17) -> (20:44)(12:30)Barika(21:37) -> (0:47)(13:0)D.MAOUCHI(1:35) -> (2:10)AKBOU 22000 22000
3. Vehicle 5 C13: (6:0)AKBOU(6:0) -> (9:45)(11:0)FERME DDA(11:41) -> (11:49)(11:0)DIDOUCHE(12:51) -> (14:48)(11:0)Annaba(15:32) -> (17:7)(14:0)Skikda(18:9) -> (20:11)(11:0)El Kennar(21:8) -> (21:54)(12:0)CAZEL(22:34) -> (23:49)AKBOU 21700 22000
4. Vehicle 6 C26: (6:0)AKBOU(6:0) -> (10:1)(10:0)BOUSSOUF(11:30) -> (12:20)(13:0)Oued Seguen(13:44) -> (15:17)(11:0)Beida bordj(16:10) -> (17:7)(12:0)Ain arnat(17:51) -> (18:36)(13:0)Ras El Oued(19:20) -> (21:30)(10:0)DOUAOUDA(22:14) -> (22:25)(10:0)EL MAADER(p)(23:4) -> (1:35)AKBOU 25000 25000
5. Vehicle 7 C27: (6:0)AKBOU(6:0) -> (15:58)(11:0)Maghnia(16:46) -> (17:29)(11:0)TLEMCEN(18:22) -> (20:13)(11:0)TLILAT(21:10) -> (2:8)(12:0)Medea(2:52) -> (5:21)AKBOU 14000 25000
6. Vehicle 8 C28: (6:0)AKBOU(6:0) -> (11:11)(11:0)Chlef(11:48) -> (15:42)(14:0)Temouchent(17:47) -> (18:47)(11:0)TAFRAOUI(19:31) -> (3:26)AKBOU 22600 25000
7. Vehicle 9 C30: (6:0)AKBOU(6:0) -> (11:48)(12:0)DJDIUIIA(12:53) -> (13:18)(12:0)Oued el Djemaa(14:15) -> (16:55)(11:0)Saida(17:48) -> (20:23)(11:0)Tiaret(21:7) -> (2:42)(14:0)CHOUKRANE(3:30) -> (3:49)AKBOU 18000 18000 .

**Solution Distance** 7267.999999999999

**Solution Cost** 306273.5199999998

**Opération effectuée en:** 0.233 secondes.

## Solution After Simulated Annealing

1. Vehicle 0 C24: (8:0)AKBOU(8:0) -> (10:1)(11:0)El Kennar(11:57) -> (14:2)(10:0)BOUSSOUF(15:31) -> (15:46)(11:0)FERME DDA(16:27) -> (18:23)(12:0)Ain arnat(19:7) -> (21:7)AKBOU 20500 24000
2. Vehicle 3 C12: (6:0)AKBOU(6:0) -> (15:58)(11:0)Maghnia(16:46) -> (17:29)(11:0)TLEMCEN(18:22) -> (20:13)(11:0)TLILAT(21:10) -> (4:50)AKBOU 12000 22000
3. Vehicle 4 C15: (8:0)AKBOU(8:0) -> (10:35)(11:0)BABA ALI(11:39) -> (12:19)(11:0)Hadji blida (13:0) -> (13:12)(11:0)Blida 1(13:56) -> (15:57)(14:0)Makouda(17:12) -> (17:53)(14:0)Freha(19:4) -> (19:57)AKBOU 21500 22000
4. Vehicle 5 C13: (6:0)AKBOU(6:0) -> (11:11)(11:0)Chlef(11:48) -> (12:31)(12:0)DJDIUIIA(13:24) -> (13:49)(12:0)Oued el Djemaa(14:46) -> (17:26)(11:0)Saida(18:19) -> (20:54)(11:0)Tiaret(21:38) -> (0:32)(12:0)Medea(1:16) -> (3:45)AKBOU 17600 22000
5. Vehicle 6 C26: (7:0)AKBOU(7:0) -> (9:31)(10:0)EL MAADER(p)(10:39) -> (10:50)(10:0)DOUAOUUDA(11:34) -> (14:14)(13:0)Seriana(15:11) -> (17:52)(11:0)BAGIA(18:40) -> (19:5)(12:0)Khenchela(19:58) -> (21:43)(11:0)FESDIS(22:31) -> (0:1)(12:30)Barika(0:54) -> (2:12)(13:0)Ras El Oued(2:56) -> (4:32)AKBOU 24000 25000
6. Vehicle 7 C27: (7:0)AKBOU(7:0) -> (9:52)(11:0)Beida bordj(11:53) -> (13:26)(13:0)Oued Seguen(14:10) -> (14:49)(11:0)DIDOUCHE(15:51) -> (17:48)(11:0)Annaba(18:32) -> (20:7)(14:0)Skikda(21:9) -> (23:57)(12:0)CAZEL(0:37) -> (1:17)(13:0)D.MAOUCHI(2:5) -> (2:40)AKBOU 24200 25000
7. Vehicle 8 C28: (6:0)AKBOU(6:0) -> (14:59)(14:0)Temouchent(17:4) -> (18:4)(11:0)TAFRAOUI(18:48) -> (2:14)(14:0)CHOUKRANE(3:2) -> (3:21)AKBOU 25000 25000

**Solution Distance** 7040.000000000004

**Solution Cost** 300181.59999999666

**Opération effectuée en:** 0.473 secondes.



## Best Solution After Genetic Algorithm

1. Vehicle 0 C13: (7:0)AKBOU(7:0) -> (9:31)(10:0)EL MAADER(p)(10:39) -> (10:50)(10:0)DOUAOUUDA(11:34) -> (14:14)(13:0)Seriana(15:11) -> (16:38)(12:30)Barika(17:31) -> (18:35)(11:0)Beida bordj(19:28) -> (20:26)(13:0)Ras El Oued(21:10) -> (22:46)AKBOU 18000 24000
2. Vehicle 1 C14: (8:0)AKBOU(8:0) -> (10:1)(11:0)El Kennar(11:57) -> (12:43)(12:0)CAZEL(13:23) -> (14:3)(13:0)D.MAOUCHI(14:51) -> (16:29)(14:0)Makouda(17:44) -> (19:45)(11:0)Blida 1(20:29) -> (20:41)(11:0)Hadji blida (21:22) -> (22:2)(11:0)BABA ALI(22:41) -> (1:16)AKBOU 22700 24000
3. Vehicle 3 C12: (9:0)AKBOU(9:0) -> (11:0)(12:0)Ain arnat(12:44) -> (14:56)(10:0)BOUSSOUF(16:25) -> (16:40)(11:0)FERME DDA(17:21) -> (17:29)(11:0)DIDOUCHE(18:31) -> (22:21)AKBOU 21500 22000
4. Vehicle 4 C15: (6:0)AKBOU(6:0) -> (9:46)(11:0)FESDIS(11:48) -> (14:21)(14:0)Skikda(15:23) -> (16:58)(11:0)Annaba(17:42) -> (21:8)(12:0)Khenchela(22:1) -> (22:26)(11:0)BAGIA(23:14) -> (1:8)(13:0)Oued Seguen(1:52) -> (5:15)AKBOU 20000 22000
5. Vehicle 6 C26: (6:0)AKBOU(6:0) -> (11:11)(11:0)Chlef(11:48) -> (12:31)(12:0)DJDIUIIA(13:24) -> (13:49)(12:0)Oued el Djemaa(14:46) -> (17:26)(11:0)Saida(18:19) -> (20:54)(11:0)Tiaret(21:38) -> (3:1)(14:0)Freha(4:12) -> (5:5)AKBOU 23600 25000
6. Vehicle 7 C27: (6:0)AKBOU(6:0) -> (15:58)(11:0)Maghnia(16:46) -> (17:29)(11:0)TLEMCEN(18:22) -> (20:13)(11:0)TLILAT(21:10) -> (2:8)(12:0)Medea(2:52) -> (5:21)AKBOU 14000 25000
7. Vehicle 8 C28: (6:0)AKBOU(6:0) -> (14:59)(14:0)Temouchent(17:4) -> (18:4)(11:0)TAFRAOUI(18:48) -> (2:14)(14:0)CHOUKRANE(3:2) -> (3:21)AKBOU 25000 25000

**Solution Distance** 7365.0

**Solution Cost** 314832,28

**Opération effectuée en:** 10.55 secondes.

## Best Solution After Hybrid Genetic Algorithm with Tabu Search

1. Vehicle 3 C12: (7:0)AKBOU(7:0) -> (9:31)(10:0)EL MAADER(p)(10:39) -> (10:50)(10:0)DOUAOUUDA(11:34) -> (14:14)(13:0)Seriana(15:11) -> (16:38)(12:30)Barika(17:31) -> (18:35)(11:0)Beida bordj(19:28) -> (20:26)(13:0)Ras El Oued(21:10) -> (22:46)AKBOU 18000 22000
2. Vehicle 4 C15: (6:0)AKBOU(6:0) -> (9:46)(11:0)FESDIS(11:48) -> (13:33)(12:0)Khenchela(14:26) -> (14:51)(11:0)BAGIA(15:39) -> (17:48)(11:0)FERME DDA(18:29) -> (18:37)(11:0)DIDOUCHE(19:39) -> (20:18)(13:0)Oued Seguen(21:2) -> (22:37)(12:0)Ain arnat(23:21) -> (1:21)AKBOU 21500 22000
3. Vehicle 5 C13: (6:0)AKBOU(6:0) -> (10:1)(10:0)BOUSSOUF(11:30) -> (13:47)(11:0)Annaba(14:31) -> (16:6)(14:0)Skikda(17:8) -> (19:56)(12:0)CAZEL(20:36) -> (21:51)AKBOU 21200 22000
4. Vehicle 6 C26: (6:0)AKBOU(6:0) -> (13:55)(11:0)TAFRAOUI(14:39) -> (15:39)(14:0)Temouchent(17:44) -> (21:38)(11:0)Chlef(22:15) -> (0:48)(11:0)Hadji blida (1:29) -> (4:42)AKBOU 24100 25000
5. Vehicle 7 C27: (8:0)AKBOU(8:0) -> (10:1)(11:0)El Kennar(11:57) -> (13:22)(13:0)D.MAOUCHI(14:10) -> (15:48)(14:0)Makouda(17:3) -> (17:44)(14:0)Freha(18:55) -> (19:48)AKBOU 25000 25000
6. Vehicle 8 C28: (6:0)AKBOU(6:0) -> (12:2)(11:0)Tiaret(12:46) -> (15:21)(11:0)Saida(16:14) -> (18:54)(12:0)Oued el Djemaa(19:51) -> (20:16)(12:0)DJDIUIIA(21:9) -> (0:5)(11:0)Blida 1(0:49) -> (1:16)(11:0)BABA ALI(1:55) -> (4:3)(14:0)CHOUKRANE(4:51) -> (5:10)AKBOU 21000 25000
7. Vehicle 9 C30: (6:0)AKBOU(6:0) -> (15:58)(11:0)Maghnia(16:46) -> (17:29)(11:0)TLEMCEN(18:22) -> (20:13)(11:0)TLILAT(21:10) -> (2:8)(12:0)Medea(2:52) -> (5:21)AKBOU 14000 18000

**Solution Distance** 7045.0

**Solution Cost** 296876.2999999998

**Opération effectuée en:** 45.60 secondes.

## 5. Synthèse des résultats

Les résultats obtenus par chaque métaheuristique sont résumés dans le *Tableau 4.7*.

		Tournée de la journée paire	Tournée de la journée impaire
<b>Nombre total des points de collecte</b>		35	36
<b>Tournée actuelle de DDA</b>	<b>Nombre de véhicules</b>	10	10
	<b>Distance (Km)</b>	8923	8603
	<b>Coût (DA)</b>	403768,16	344350,68
<b>Recherche Tabou (RT)</b>	<b>Nombre de véhicules</b>	8	7
	<b>Distance (Km)</b>	7884	7268
	<b>Amélioration distance (%)</b>	<b>12%</b>	<b>15,5%</b>
	<b>Coût (DA)</b>	342955,56	306273,52
	<b>Amélioration coût (%)</b>	<b>15,1%</b>	<b>11,1%</b>
<b>Recuit Simulé (RS)</b>	<b>Nombre de véhicules</b>	7	7
	<b>Distance (Km)</b>	7826	7040
	<b>Amélioration distance (%)</b>	<b>12,3%</b>	<b>18%</b>
	<b>Coût (DA)</b>	341982,29	300181,6
	<b>Amélioration coût (%)</b>	<b>15,3%</b>	<b>12,8%</b>
<b>Algorithme génétique (AG)</b>	<b>Nombre de véhicules</b>	8	7
	<b>Distance (Km)</b>	7967	7365
	<b>Amélioration distance (%)</b>	<b>10,7%</b>	<b>14,4%</b>
	<b>Coût (DA)</b>	347414,22	314832,28
	<b>Amélioration coût (%)</b>	<b>14%</b>	<b>8,6%</b>
<b>Méthode hybride (AG + RT)</b>	<b>Nombre de véhicules</b>	8	7
	<b>Distance (Km)</b>	7737	7045
	<b>Amélioration distance (%)</b>	<b>13,3%</b>	<b>18,1%</b>
	<b>Coût (DA)</b>	332072,98	296876,3
	<b>Amélioration coût (%)</b>	<b>17,8%</b>	<b>13,8%</b>

*Tableau 4.7 : Résultats des 4 métaheuristiques et les améliorations apportées*

## 5.1. Interprétation des résultats

Le *Tableau 4.7* résume les résultats obtenus. Les résultats concernent le nombre de camions utilisés, la distance totale et le coût logistique ainsi que les améliorations en termes de distance et de coût apportées par chaque métaheuristique.

Nous constatons clairement d'après les résultats du tableau, que les quatre métaheuristicques fournissent des tournées bien meilleures que les tournées actuelles adoptées par DDA, en apportant des améliorations notables sur plusieurs critères. Ces améliorations concernent d'abord la distance totale parcourue par les camions, où nous avons enregistré des améliorations variant de 10,7% à 13,3% (12% pour RT, 12,3% pour RS, 10,7% pour l'AG et 13,3% pour l'approche hybride AG+RT) pour les jours paires, et de 14% à 18% (15,5% pour RT, 18% pour RS, 14,4% pour l'AG et 18,1% pour l'approche hybride AG+RT) pour les jours impaires. Ces améliorations de distance ont logiquement engendré des gains en termes de coûts logistiques, allant de 14% jusqu'à **17,8%** (15,1% pour RT, 15,3% pour RS, 14% pour l'AG et 17,8% pour l'approche hybride AG+RT) pour les jours paires, et allant de **8,6% jusqu'à 13,8%** (11,1% pour RT, 12,8% pour RS, 8,6% pour l'AG et 13,8% pour l'approche hybride AG+RT) pour les jours impaires. En terme financiers purs, les gains obtenus peuvent aller jusqu'à **71755,18 DA** pour les journées paires et jusqu'à **47474,38 DA** pour les journées impaires, ce qui nous fait un gain total de **1788443,4 DA** par mois, ce qui est assez conséquent comme économie sur les coûts logistiques.

Notons par ailleurs que le taux d'exploitation des capacités des camions a été augmenté permettant d'atteindre des taux d'exploitation allant jusqu'à 90%. Ceci a permis de réduire d'une manière conséquente le nombre de camions utilisés dans les tournées, passant de 10 à 8 camions pour les journées paires et de 10 à 7 camions pour les journées impaires. Cette réduction du nombre de camions va engendrer certainement des coûts en moins pour le département logistique.

Enfin, nous constatons que la métaheuristique hybride AG+RT est la plus performante suivie des métaheuristicques RS, RT et AG respectivement. Ceci confirme la tendance des résultats obtenus dans le chapitre 3.

## 5.2. Affichage des tournée obtenues sur Google Maps

Les figures ci-dessous illustrent sur Google Maps les meilleures tournées obtenues par les métaheuristiques proposées.

### 5.2.1. Affichage sur Google Maps de la tournée obtenue par la métaheuristique hybride (AG+RT) pour la journée paire



Figure 4.1 : Tournée obtenue par la métaheuristique RT pour la journée paire

- 5.2.2. Affichage sur Google Maps de la tournée obtenue par la métaheuristique RT pour la journée paire**
  
- 5.2.3. Affichage sur Google Maps de la tournée obtenue par la métaheuristique hybride (AG+RT) pour la journée impaire**
  
- 5.2.4. Affichage sur Google Maps de la tournée obtenue par la métaheuristique RS pour la journée impaire**

## **6. Conclusion**

Dans ce chapitre, nous avons présenté une application industrielle, issue de la collecte de lait frais pour l'entreprise DDA. Après avoir exposé toutes les données réelles nécessaires à la résolution du problème étudié, nous avons analysé et interprété les résultats obtenus. Nous avons constaté clairement d'après ces résultats, que les quatre métaheuristiques (RS, RT, AG et AG+RT) fournissent des tournées bien meilleures que les tournées actuelles adoptées par DDA, permettant d'apporter des améliorations importantes en terme de : distance totale parcourue, coûts logistiques engendrés, taux d'exploitation des capacités des camions et le nombre de camions utilisés.

## Conclusion générale et perspectives

L'objectif de ce travail d'ingénieur était d'étudier un problème de tournée de véhicules et de proposer des approches à base de métaheuristiques pour la résolution du problème étudié. Etant donné son importance pratique, nous avons développé un modèle pour l'élaboration des tournées de véhicules avec fenêtres de temps et une durée maximale de chaque tournée (VRPTWDC). Ce modèle a plusieurs applications importantes notamment dans la livraison de colis, la distribution des produits périssables, la collecte des productions agricoles (lait, les produits céréaliers, les produits maraîchers ...), le ramassage scolaire et le service bancaire.

Pour la résolution du problème VRPTWDC qui est considéré comme NP-difficile, nous avons proposé quatre approches de résolution basées sur les métaheuristiques suivantes : Recuit simulé (RS), Recherche Tabou (RT), Algorithme génétique (AG) et une méthode hybridant un algorithme génétique avec la recherche tabou (AG+RT). Les différentes approches de résolution proposées ont été détaillées et expliquées dans le troisième chapitre de ce mémoire.

Par ailleurs, des expériences numériques ont été réalisées et analysées sur des instances de différentes tailles (petites, moyennes et grandes). Pour les petites instances, nous avons constaté que les métaheuristiques donnent des résultats quasi identiques à la méthode exacte Branch & Bound. Ceci atteste de la fiabilité des métaheuristiques proposées pour la résolution du problème VRPTWDC. Pour les moyennes et grandes instances, les résultats obtenus montrent que la métaheuristique hybride AG+RT est la plus performante suivi des métaheuristiques RS, RT et AG respectivement.

Enfin, le modèle VRPTWDC et les approches de résolution proposées ont été appliqués à un cas pratique issu de la collecte de lait frais de l'entreprise Danone Djurdjura Algérie (DDA). Après avoir exposé toutes les données réelles nécessaires à la résolution du cas pratique, nous avons analysé et interprété les résultats obtenus. Ces résultats montrent clairement, que les quatre métaheuristiques (RS, RT, AG et AG+RT) fournissent des tournées bien meilleures que les tournées actuelles adoptées par DDA, permettant d'apporter des améliorations importantes en terme de : distance totale parcourue, coûts logistiques engendrés, taux d'exploitation des capacités des camions et le nombre de camions utilisés. Nous avons constaté également que les meilleures améliorations sont obtenues par la



métaheuristique hybride AG+RT suivie des métaheuristicques RS, RT et AG respectivement. Ceci confirme la tendance des résultats du troisième chapitre.

Les perspectives futures de ce travail concernent notamment le développement d'une application offrant une interface graphique ergonomique, permettant d'aider les décideurs et les logisticiens à élaborer les tournées de véhicules de leur entreprise pour les activités d'approvisionnements en matières premières et/ou de distribution des produits finis. Cette application utilisera évidemment le potentiel des quatre métaheuristicques déjà implémentées. Enfin, il est également envisageable dans un avenir à moyen terme, de développer d'autres métaheuristicques hybrides permettant de résoudre efficacement les problèmes de tournées de véhicules.

## Bibliographie

Altman, C. (2017). Optimisation de tournées de véhicules avec contrainte de fragilité (Doctoral dissertation, École Polytechnique de Montréal).

Applegate, D. L., Bixby, R. E., Chvatal, V., & Cook, W. J. (2006). The traveling salesman problem: a computational study. Princeton university press.

B. M. Baker and M.A. Ayechev . « A genetic algorithm for the vehicle routing problem ». *Computers and Operations Research* 30, pages 787–800, 2003.

Benaichouche, A. N. (2014). Conception de métaheuristiques d'optimisation pour la segmentation d'images: application aux images IRM du cerveau et aux images de tomographie par émission de positons (Doctoral dissertation).

Bernard, T. T., & Pauline, F. L. Heuristiques du problème du voyageur de commerce.

Boussaid, I. (2013). Perfectionnement de métaheuristiques pour l'optimisation continue (Doctoral dissertation, Paris Est).

Buzon, L. (2006). *Contribution à la structuration des échanges de connaissances dans le cadre de relations de collaboration des chaînes logistiques* (Doctoral dissertation).

Feigenbaum, E. A. (1963). *Computers and thought: A collection of articles*. J. Feldman (Ed.). New York, NY: McGraw-Hill.

Grimault, A. (2016). Optimisation de tournées de camions complets dans le secteur des travaux publics (Doctoral dissertation, Ecole des Mines de Nantes).

Guerra, L., Murino, T., & Romano, E. (2007). The location-routing problem: an innovative approach. 6th WSEAS Transactions on System Science and Simulation in Engineering, Venice, Italy, 21-23.

Haiyan, Housroum. (2005). *Une approche génétique pour la résolution du problème VRPTW dynamique* (Doctoral dissertation, Université d'Artois).

Ismail, S. B., Legras, F., & Coppin, G. (2011). Synthèse du problème de routage de véhicules.

K. Q. Zhu. « A new genetic algorithm for VRPTW ». International Conference on Artificial Intelligence, Las

Vegas, USA, April 13 2000.

KHELFI, M. M. F. (2017). Proposition de solutions pour l'optimisation des chaînes logistiques (Doctoral dissertation, Université d'Oran).

M. H Hugos 2003. Journal of business logistics, 22(2):1–25, 2001

Mekamcha ,K.(2020). Réalisation d'un support d'aide à la décision pour la gestion du problème stochastique de la collecte des déchets ménagers dans une zone urbaine (Doctoral dissertation, Université de Tlemcen.).

Ngueveu, S. U. (2009). Problèmes de tournées de véhicules avec contraintes particulières pour la maîtrise des risques (Doctoral dissertation, PhD thesis, Université de Technologie de Troyes).

PRODHON, C. (2006). LE PROBLÈME DE LOCALISATION-ROUTAGE.

Rizet, C., & Keita, B. (2005). Chaînes logistiques et consommation d'énergie: cas du yaourt et du jean.

Sevaux, M. (2004). Métaheuristiques: Stratégies pour l'optimisation de la production de biens et de services (Doctoral dissertation).

Van der Vorst, J. G. A. J., Da Silva, C., & Trienekens, J. H. (2011). Gestion de la chaîne d'approvisionnement agro-industrielle: concepts et applications. Document Occasionnel de Gestion, Commercialisation et Finances Agricoles.

Vidal, T., Crainic, T., Gendreau, M., & Prins, C. (2011). Heuristiques pour les problèmes de tournées de véhicules multi-attributs. CIRRELT.

Widmer, M. (2001, April). Les métaheuristiques: des outils performants pour les problèmes industriels. In 3ème Conférence Francophone de Modélisation et Simulation MOSIM (Vol. 1, pp. 25-27).