

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Aboubakr Belkaïd – Tlemcen –

Faculté de Technologie

Département de Génie Electrique et Electronique

Filière **Génie Industriel**

Spécialité : **Ingénierie de production**



Projet de Fin d'Etudes de Master

Intitulé :

**Conception d'une chaîne logistique pour la distribution
alimentaire à la ville de Tlemcen**

Présenté par :

➤ **REDJAIMIA Mohammed Seyfeddine**

➤ **METRITER Mohammed**

Soutenue, le 07/07/2021, devant le jury composé de :

Mme KHEDIM Amaria	MCB	Université de Tlemcen	Présidente
M. BENSMAN Yassir	MCB	Université de Tlemcen	Examineur
M. MEKAMCHA Khalid	MCB	Université de Tlemcen	Examineur
M. BELKAID Fayçal	MCA	Université de Tlemcen	Encadrant
M. BENNEKROUF Mohammed	MCB	ESSAT Tlemcen	Co-encadrant

Année Universitaire : 2020/2021

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

اِنَّا فَخَّرْنَا مُبِيبًا

Remerciement :

En guise de reconnaissance, nous tenons à témoigner nos sincères remerciements tout d'abord à Allah le tout puissant et miséricordieux, qui nous a donné de la force et de la patience pour accomplir ce modeste travail, ensuite à toutes les personnes qui ont contribué de près ou de loin au bon déroulement de notre projet de fin d'étude et à l'élaboration de ce modeste travail et spécialement à notre encadrant Mr BELKAID.F et notre Co-encadrant Mr BENNEKROUF.M.

Nous tenons à remercier aussi l'ensemble du personnel des départements Génie Industriel et leurs conseils pleins de sens et pour le suivi et l'intérêt qu'ils ont portés à nos travaux. Nos remerciements s'étendent également à tous les enseignants et toutes les enseignantes durant les années de nos études, en particulier tous les membres du Laboratoire MELT, sans oublier les membres de Jury pour avoir accepté de juger notre travail.

Dans l'impossibilité de citer tous les noms, nos sincères remerciements vont à tous ceux et celles, qui de près ou de loin, ont permis par leurs conseils et leurs compétences la réalisation de ce projet de fin d'étude.

Enfin, nous n'oserons oublier de remercier tous nos camarades nos enseignants ainsi que nos familles pour leurs soutiens durant toute cette merveilleuse aventure.

Dédicace :

Je commence par rendre grâce à « Allah » de nous aider à accomplir ce travail

"الحمد لله حمدا كثيرا طيبا مباركا فيه ملئ السماوات والارض ملئ ما شاء ربي"

Avec pleins de joie et un énorme plaisir que je dédie ce modeste travail comme preuve de gratitude tout d'abord à ma famille, mes yeux dans la vie mes chères mère et père, mes frangins Zakaria et Idriss et frangines N et B, ma meilleure grande mère, à toute ma famille oncles et tantes, mon grand frère Ali. DELLOUL qui m'ont beaucoup soutenu et encouragé pendant tout mon cursus, je les remercie un par un du fond du cœur pour leurs sacrifices et soutiens envers moi. Ce modeste travail est dédié aussi à ma deuxième famille en génie industriel mes amis(e), mes chères collègues, et enseignants que j'ai passé avec eux de très bons moments à jamais oublier, sans oublier mon frère et mon bras droit mon binôme METRIER, Mohammed.

REDJAIMIA Med Seyfeddine



Dédicaces :

La vie n'est qu'un éclair,

Et un jour de réussite est un jour très cher.

Je dédie ce mémoire

À mon cher père

Et ma très chère mère

ET ma deuxième mère (ma tantes)

Pour l'éducation et le grand amour dont ils m'ont entouré depuis ma naissance.

Et pour leurs patiences et leurs sacrifices.

À mon cher frère: BACHIR;

À ma chère sœur : W;

À tous mes proches : mes tantes, mes cousines Abdelkrim, Hocine, Mustapha et ... Et surtout Mr : BELKAID.F qui nous a aidé pendant la phase de ce travail, mes voisins et toute la famille METRITER,

À tous ceux qui m'aiment ;

À tous mes ami (e) s;

À tous ceux que j'aime ...;

À tous mes collègues de promo Master II en

Génie industriel (2020/2021);

METRITER Mohamed



TABLE DES MATIÈRES:

LISTE DES FIGURES:	VII
LISTE DES TABLEAUX:	IX
LISTE DES SIGLES:	X
INTRODUCTION GENERALES:	1
CHAPITRE I: GENERALITES ET DEFINITIONS.	3
I.1 APERÇU HISTORIQUE SUR LA CHAÎNE LOGISTIQUE :	5
I.2 INTRODUCTION :	6
I.3 DEFINITION DE LA LOGISTIQUE :	6
I.3.1 <i>La stratégie de chaîne logistique</i> :	6
I.4 DEFINITION DE LA CHAÎNE LOGISTIQUE :	6
I.4.1 <i>Définition de la chaîne logistique</i> : « <i>supply chain</i> »	6
I.4.2 <i>Chaîne logistique inverse</i> :	8
I.4.2 <i>Chaîne logistique verte</i> :	8
I.4.3 <i>Finalité de la logistique</i> :	9
I.4.3.1 <i>Sur le court terme</i> :	9
I.4.3.2 <i>Sur le moyen terme</i> :	9
I.4.3.3 <i>Sur le long terme</i> :	10
I.4.4 <i>Planification d'une chaîne logistique</i>	10
I.4.4.1 <i>Décision d'approvisionnement</i> :	10
I.4.4.2 <i>Décision de production</i> :	10
I.4.4.3 <i>Décisions temporelles</i> :	10
I.5 LES FLUX DE LA CHAÎNE LOGISTIQUE :	11
I.5.1 <i>Les flux physiques</i> :	11
I.5.2 <i>Les flux financiers</i> :	11
I.5.3 <i>Les flux d'informations</i> :	11
I.6 CONCEPTION DES CHAÎNES LOGISTIQUES :	11
I.6.1 <i>Les niveaux de décision dans la chaîne logistique</i> :	12
I.6.1.1 <i>Prise de décision au niveau stratégique</i> :	12
I.6.1.2 <i>Décision au niveau tactique</i> :	13
I.6.1.3 <i>Décision au niveau opérationnel</i> :	14
I.7 CLASSES DE BASE DE CHAÎNE LOGISTIQUE :	14
I.7.1 <i>Les usines</i> :	14
I.7.2 <i>Les centres de distribution</i> :	14
I.7.3 <i>Les fournisseurs</i> :	14
I.7.4 <i>Le transport</i> :	15
I.7.5 <i>Liaison d'information</i> :	15
I.7.6 <i>L'entreprise</i> :	15
I.7.7 <i>Client</i> :	15
I.8 PROBLÉMATIQUES LIÉES À LA CHAÎNE LOGISTIQUE :	16
I.8.1 <i>Impacts et interactions entre les décisions stratégiques, tactiques et opérationnelles</i> :	16

I.8.2	<i>Incertitude tout au long d'une chaîne logistique :</i>	16
I.8.3	<i>Nécessité de passer de l'optimisation monocritère à l'optimisation multicritère :</i>	16
I.8.3.1	<i>Le processus d'approvisionnement :</i>	16
I.8.3.2	<i>Le processus production :</i>	17
I.8.3.3	<i>Le processus distribution :</i>	17
I.8.3.4	<i>Le processus de vente :</i>	17
I.8.4	<i>Etat de l'art des chaînes logistiques :</i>	18
I.9	CONCLUSION :	19
CHAPITRE II: MODELES D'ANALYSE ET RESOLUTION:		20
II.1	INTRODUCTION :	22
II.2	MODELES D'ANALYSE :	22
II.2.1	<i>La nature de la décision :</i>	22
II.2.2	<i>Nombre des étapes dans un réseau logistique :</i>	22
II.2.3	<i>Nombre de périodes considérées :</i>	23
II.2.4	<i>Les objectifs à atteindre :</i>	23
II.2.5	<i>Contraintes prises en compte :</i>	23
II.2.6	<i>Internationalité du réseau logistique :</i>	23
II.2.7	<i>Analyses multicritères :</i>	23
II.3	PROBLEMES DE LOCALISATION/ALLOCATION :	23
II.3.1	<i>Classification de Location-Allocation problem :</i>	24
II.3.1.1	<i>Classification selon la demande des clients :</i>	24
II.3.1.2	<i>Classification selon le nombre d'installations :</i>	24
II.4	LES PROBLEMES USUELS DANS LA LOGISTIQUE :	24
II.4.1	<i>Problèmes de recouvrement :</i>	24
II.4.2	<i>Problème de recouvrement maximale :</i>	25
II.4.3	<i>Problèmes P-médianes :</i>	26
II.4.4	<i>Le problème du centre (center problem) :</i>	27
II.4.5	<i>Le problème de localisation des sites à deux niveaux :</i>	28
II.5	LES CARACTERISTIQUES DE BASE D'UNE PRISE DE DECISION :	30
II.5.1	<i>Les méthodes interactives d'aide à la décision :</i>	31
II.5.2	<i>Procédure de choix d'une méthode d'aide à la décision multicritère :</i>	31
II.5.3	<i>Goal programming (programmation par objectif) :</i>	31
II.5.4	<i>Avantages et inconvénients du GP :</i>	32
II.5.4.1	<i>Avantages :</i>	32
II.5.4.2	<i>Inconvénients :</i>	32
II.6	LE GOAL PROGRAMMING INTERACTIVE :	32
II.6.1	<i>Le GP pondéré :</i>	33
II.6.1.1	<i>Le GP minimax :</i>	34
II.6.1.2	<i>Le GP flou :</i>	34
II.6.1.3	<i>Le GP avec intervalle :</i>	35
II.6.1.4	<i>Comparaison entre le modèle linéaire et le modèle GP :</i>	36
II.7	CONCLUSION :	37
CHAPITRE III : CAS D'ETUDES :		36
III.1	INTRODUCTION :	39
III.2	SYSTEMES D'INFORMATIONS GEOGRAPHIQUES (SIG) :	39

III.2.1	<i>Définition et présentation</i>	39
III.2.2	<i>Les composants d'un SIG</i> :.....	39
a)	Matériel :	39
a)	Les logiciels :	40
b)	Les données :.....	40
c)	Les utilisateurs :	40
d)	Méthodes :.....	40
III.2.3	<i>Types des données d'un SIG</i> :.....	40
III.2.4	<i>Fonctionnalité d'un SIG</i> :	42
III.3	LES PLATEFORMES LOGISTIQUES :	43
III.3.1	<i>Définition des plateformes</i> :	43
III.3.2	<i>Objectif d'établissement des plateformes logistiques</i> :.....	44
III.4	LES CARACTERISTIQUES D'UNE PLATEFORME :	44
III.5	DEFINITION DE LA PROBLEMATIQUE :.....	45
III.5.1	<i>Présentation de l'entreprise Giplait</i> :	45
III.5.2	<i>Processus de production du lait de vache pasteurisé</i> :.....	46
III.5.3	<i>Cas d'étude</i> :	46
III.6	PROBLEMATIQUE 01 MODELE (P-MEDIAN) :	48
III.6.1	<i>Problème de capacité</i> :.....	54
III.6.2	<i>Problème de distance maximale</i> :	56
III.6.3	<i>Problème des points de vente</i> :.....	58
III.7	PROBLEMATIQUE 02 :.....	61
III.7.1	<i>Problème des points de vente</i> :.....	70
III.8	CONCLUSION :.....	71
	CHAPITRE IV: METHODE GOAL PROGRAMMING.....	70
IV.1	INTRODUCTION :	73
IV.2	DESCRIPTION DE LA PROBLEMATIQUE :.....	73
IV.3	EXEMPLES ILLUSTRATIFS :	73
IV.3.1	<i>Exemple 1</i> :.....	73
IV.3.1.1	Cas où les deux méthodes fournissent les mêmes résultats :	73
Interprétation 01	:	75
Interprétation 02	:	77
IV.3.1.2	Cas où les deux modèles fournissent des résultats différents :	78
IV.3.2	<i>Exemple 2 problème de minimisation</i> :.....	80
Scénario 0 1	:.....	81
Scénario 03	:.....	83
IV.3.3	<i>Attribution des poids (priorités) aux objectifs</i> :	85
IV.4	APPLICATION DE LA METHODE GOAL PROGRAMMING AU CAS D'ETUDE :.....	88
IV.4.1	<i>Problématique 01 : Modèle p-médian</i> :	88
IV.4.2	<i>Problème de non faisabilité à cause de capacité insuffisante</i> :	88
IV.4.2.1	Interprétation :	90
IV.4.3	<i>Problème de non faisabilité à cause de distance</i> :	90
IV.4.3.1	Interprétation :	92
IV.4.4	<i>Problème de non faisabilité à cause de nombre de point de vente</i> :.....	92
IV.4.4.1	Interprétation :	94
IV.5	PROBLEMATIQUE 02 : MODELE MAX-COVERING :	94

<i>IV.5.1 Problème de non faisabilité à cause de la capacité des points de vente :</i>	94
IV.5.1.1 Interprétation :	95
IV.5.1.2 Interprétation :	97
<i>IV.5.2 Problème de non faisabilité à cause de point de vente :</i>	97
IV.5.2.1 Interprétation :	98
IV.6PROBLEMATIQUE 03 :	99
IV.6.1 Interprétation :	100
IV.7PROBLEMATIQUE 04 :	100
IV.7.1.1 Interprétation :	102
IV.7.1.2 Interprétation :	103
IV.8SYNTHESE :	103
CONCLUSION GENERALE :	101
BIOBLIOGRAPHIE:	102
WEBOGRAPHIE:	106
Résumé/Abstract/ملخص:	106

LISTE DES FIGURES :

Figure 1. 1 : Représentation d'une chaine logistique [R]-----	7
Figure 1. 2 : Représentation d'une chaine logistique inverse [S] -----	8
Figure 1. 3 : Représentation d'une chaine logistique verte [T]-----	9
Figure 1. 4: Pyramide des niveaux de décisions [U] -----	12
Figure 2. 1 : Situation typique de la décision	30
Figure 2. 2: Schéma montre les caractéristiques d'une prise de décision	31
Figure 2. 3: La fonction d'appartenance Δ_i (forme triangulaire) [AYADI, 2010]	35
Figure 2. 4 : Fonction de pénalité à 5 segments [AYADI, 2010]	36
Figure 3. 1: Données géographiques et styles [QGIS]	40
Figure 3. 2: Données descriptives (attributaires) [QHIS]	41
Figure 3. 3: Les formes des données vectorielles [QGIS].....	41
Figure 3. 4: Type de données pixels [V]	42
Figure 3. 5: Exemple des Types de Raster [QGSi]	42
Figure 3. 6: Des couches de données en superposition [QGIS]	43
Figure 3. 7: Représentation d'une PFL [W].....	44
Figure 3. 8: Processus de fabrication du lait pasteurisé	46
Figure 3. 9: Représentation des zones avant nettoyage sur QGIS	47
Figure 3. 10: Carte clé QGIS.....	48
Figure 3. 11: Représentation des zones dans le logiciel QGIS	49
Figure 3. 12: Représentation des points de vente dans le logiciel QGIS	50
Figure 3. 13: Résultats du modèle Lingo P-médian	52
Figure 3. 14: Représentation des résultats sur QGIS	53
Figure 3. 15: Résultats du modèle Lingo P-médian problème de capacité	55
Figure 3. 16: : Résultats du modèle Lingo P-médian problème de capacité	56
Figure 3. 17: Résultats du modèle Lingo P-médian problème de distance	57
Figure 3. 18: Résultats du modèle Lingo P-médian problème de nombre des points de ventes	59
Figure 3. 19: Résultats du modèle Lingo P-médian problème de nombre de points de ventes	60
Figure 3. 20: Résultats du modèle Lingo P-médian problème de nombre de points de ventes	61
Figure 3. 21: Schéma explicatif.....	63
Figure 3. 22: Résultats du modèle Lingo max-covering standards	64
Figure 3. 23: Résultats du modèle Lingo max-covering standard.....	65
Figure 3. 24: Résultats du modèle Lingo max-covering standard.....	66
Figure 3. 25: Résultats du modèle Lingo max-covering extension.....	67
Figure 3. 26: : Résultats du modèle Lingo max-covering extension.....	68
Figure 3. 27: Représentation du résultat sur le QGIS	69

LISTE DES FIGURES

Figure 3. 28: Résultats du modèle Lingo max-covering extension problème de distance.....	70
Figure 3. 29: Résultats du modèle Lingo max-covering extension problème de capacité.....	71
Figure 4. 1: Modèle linéaire non faisable -----	74
Figure 4. 2: Résultats du modèle Lingo LP-1- -----	75
Figure 4. 3: Résultats du modèle Goal programming-1 -----	77
Figure 4. 4: Résultats du modèle LP-2- -----	79
Figure 4. 5: Résultats du modèle Goal programming GP-2- -----	80
Figure 4. 6: Résultats du modèle Linéaire LP-3.1- -----	82
Figure 4. 7: Résultats du modèle Linéaire LP-3.2- -----	83
Figure 4. 8: Résultats du modèle Linéaire LP -3.3-----	84
Figure 4. 9: Résultats du modèle GP-3- -----	85
Figure 4. 10: Résultats du modèle GP sans poids-----	87
Figure 4. 11: Résultats du modèle GP avec poids -----	88
Figure 4. 12: Résultats du modèle GP P-médian problème de capacité -----	90
Figure 4. 13: Résultats du modèle p-médian GP problème de distance -----	92
Figure 4. 14: Résultats du modèle GP problème P-médian de point de vente -----	93
Figure 4. 15: Résultats du modèle GP max covering problème de capacité -----	95
Figure 4. 16: Résultats du modèle GP max-covering problème de distance -----	97
Figure 4. 17: Résultats de modèle GP max-covering problème de point de vente -----	98
Figure 4. 18: Résultats du modèle lingo problème multi objectifs -----	100
Figure 4. 19: Résultats du modèle multi objectifs sans poids d'importance-----	101
Figure 4. 20: Résultats du modèle Gp multi objectifs avec poids d'importance -----	103

LISTE DES TABLEAUX :

Tableau 2. 1: Comparaison entre le modèle linéaire et le modèle GP [1] ----- 36

Tableau 3. 1: État journalier de la distribution du lait "LPC" sur Tlemcen -----47

Tableau 3. 2: Les distances entre les zones et les points de vente -----49

Tableau 3. 3: La demande estimée pour chaque zone -----53

Tableau 4. 1: Exemple-1- Goal programming -----73

Tableau 4. 2 : Exemple-1.1- Goal programming -----78

Tableau 4. 3: Exemple -2- Goal programming -----80

Tableau 4. 4: Exemple-3- Goal programming avec poids -----86

Tableau 4. 5: Les poids affectés pour chaque objectif -----87

LISTE DES SIGLES:

FSCM: Food supply chain management

CPLP: Capacitated plant location problem

UCFLP: Uncapacitated fixed charge facility location problem

SIG : Système d'information géographique

DCs : Distribution centers

PFL : Plateforme logistique

GP : goal programming

LP : linear programming

PFE : projet de fin d'études

Introduction Générale :

La crise économique apparue les dernières années a influencé beaucoup sur les Pays du tiers-monde surtout ceux qui leurs revenus économiques est reposé essentiellement sur le pétrole qui a marqué une atroce des prix dans les dernières années. L'Algérie fait partie de cette communauté vue qu'elle prend sa force économique du pétrole qui représente plus de 95 pourcents de son revenu annuel. L'Algérie n'a pas pu traverser la crise économique mondiale avec succès du coup elle était obligée d'appliquer politique d'austérité durant cette période de crise.

Pratiquement c'est le bon moment de chercher des nouvelles solutions pour améliorer le secteur économique et pour fonder un état qui se repose sur une rigide infrastructure et trouver alternatives qui vont remplacer la place de pétrole.

Ce travail s'inscrit dans cette courbure, nous sommes donc dans ce travail à propos de concevoir une chaine logistique pour la distribution alimentaire sur la ville de Tlemcen, particulièrement nous sommes basés dans cette étude sur le lait pasteurisé représente un état de crise sur presque tout le territoire de l'état.

Le but de notre travail est d'implanter des plateformes logistiques (points de vente) dédiés spécialement à la distribution du lait pasteurisé de l'usine directement vers le client sans avoir passer par la partie intermédiaire pour satisfaire les demandes du client comme un premier objectif et par la suite économiser le cout global de la chaine par la minimisation des distances de déplacement pour chaque client de tels sorte il sera affecté au point de vente le plus proche.

Nous allons donc voir dans le premier chapitre des définitions de la chaine logistique, ses différents types, les niveaux de décision organisationnels, ses flux ainsi que ses classes de base, ensuite nous allons conclure par un état de l'art des travaux connexes à ce domaine.

Dans le deuxième chapitre nous parlerons des modèles d'analyse et de résolution des problèmes, nous allons présenter ensuite les modèles usuels dans la logistique tels que (p-médian), (max-covering), (problème de recouvrement) et (problème du centre). Nous allons par la suite abordons une méthode de décision multicritère qui est goal programming, nous allons parler de ses avantages et ses inconvénients, ses différents types...

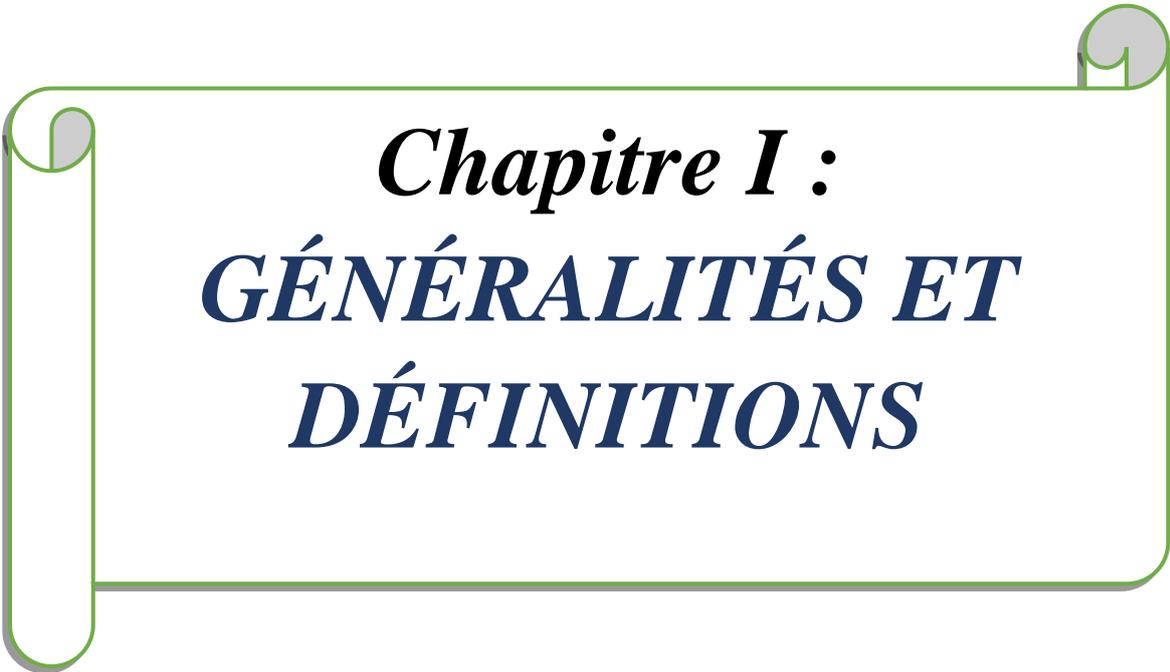
Le troisième chapitre est consacré au cas d'études ou nous allons appliquer les modèles p-médian et max-covering notre réseau conçu pour permettre couvrir le maximum des clients à moindre cout de déplacement c'est-à-dire que chaque client doit être affecté au point de vente le plus proche et qui peut satisfaire sa demande bien sûr en utilisant le solver lingo 18.0 et le logiciel QGIS pour concevoir notre carte géographique

Le quatrième chapitre est dédié spécialement à l'application de l'approche goal programming sur notre cas d'étude, ce chapitre est partagé en quatre sous chapitres, le premier est consacré à des exemples illustratifs pour permettre de comprendre le principe de la méthode.

La deuxième partie ça concerne la résolution des modèles non faisables du modèle p-médian rencontrés dans le troisième chapitre à l'aide du goal programming.

La troisième partie ça concerne la résolution des modèles non faisables du modèle max-covering rencontrés que nous avons rencontré le troisième chapitre à l'aide du goal programming.

La dernière partie est pour but de présenter la notion multi objectifs et la priorisation des objectifs à l'aide du goal programming en l'appliquant sur notre cas d'étude.



Chapitre I :
GÉNÉRALITÉS ET
DÉFINITIONS

I.1 Aperçue Historique sur la chaine logistique :

L'aspect économique à l'époque des années 1970 était basé essentiellement sur l'offre de tel sorte que les entreprises suivent une stratégie de production de masse (sur stock) pour satisfaire les besoins du client à l'heure. Un moment donné les gens ont découvert que cette démarche avait plus d'inconvénients tels que :

- La lenteur de développement et lancement des nouveaux produits.
- Le besoin des stocks ce qui implique une immobilisation financière.

[BOUDAHRI,2013]

Le concept de la chaine logistique comprenant le mot « Logistique » qui est d'origines grecques signifiant « L'art de raisonnement et de calcul »

[MARTINEZ-PONS,1996]

La logistique est apparue pour la première fois dans un environnement militaire qui implique tous qui est nécessaire (physiquement) pour appliquer des décisions stratégiques et tactiques dans ce domaine. Certains y retournent jusqu'à l'époque d'Alexander le grand (Engles,1978) ils ont tous souligné sa signification de gérer la chaine logistique de son armée. On peut donc dire qu'Alexander était le précurseur de cette approche (chaine logistique), après que la logistique militaire est arrivée au domaine industriel surtout dans les activités de support à la production.

Les années 1980 ont constaté des bouleversements des marchés à cause de la concurrence les exigences de performance et le développement technologique du coup les entreprises ont été obligé d'offrir des produits de bonnes qualités avec des prix réduits dans le but d'améliorer les performances de rendement et réduire les temps de cycle face aux concurrents. Pour cela les entreprises ont décidé d'utiliser des méthodes de management tels que : le juste à temps (JIT : Just in time) permettant de réduire les stocks de composants et ordonnancer et planifier l'approvisionnement avec les fournisseurs.

[FRANCOIS, 2007]

I.2 Introduction :

Les dernières années témoignaient une grande évolution dans le secteur économique à cause du développement technologique et surtout informatique que nous a permis de planifier et ordonnancer les tâches facilement, estimer les gains ainsi que les dépenses probables. La chaîne logistique l'une des démarches qui ont contribué beaucoup à l'amélioration de l'économie et l'industrie dans le monde entier. Elle consiste à résoudre les problèmes à l'aide des algorithmes de recherche générant de très bonnes solutions.

Nous allons entamer ce premier chapitre par la définition de la chaîne logistique, les trois types de la chaîne logistique : traditionnelle, inverse et la chaîne logistique verte, ensuite nous allons citer les finalités des chaînes logistiques ainsi que les différents flux de la chaîne logistique : physiques, financier et flux d'informations, la planification d'une chaîne logistique enfin nous citerons quelques fonctions de la chaîne logistique et ses classes de base.

I.3 Définition de la logistique :

« Le terme logistique vient du mot grec « logistike » qui porte le sens de l'art du raisonnement et du calcul. Initialement, la logistique vient du domaine militaire où elle précise l'ensemble des techniques mises en œuvre pour assurer l'approvisionnement, et le maintien en conditions opérationnelles des troupes armées en temps de guerre ».

[MERZOUK, 2007].

I.3.1 La stratégie de chaîne logistique :

La chaîne d'approvisionnement joue un rôle stratégique très important puisqu'elle permet d'obtenir un avantage concurrentiel, le fait de maintenir un prix de vente bas par rapport aux concurrents, en plus de ça une bonne chaîne d'approvisionnement pour fidéliser la clientèle nécessite d'avoir la fiabilité, la rapidité de livraison, l'innovation et un système de traçabilité en cas des problèmes ou des crises.

[RAHMOUNI et SOUICI, 2019]

ASLOG définit la logistique comme « l'art et la manière de fournir un produit donné au bon moment, au bon endroit, au moindre coût et avec la meilleure qualité [possible] ».

[Faq logistique, 2002]

I.4 Définition de la chaîne logistique :

« Supply chain » la traduction anglaise signifie « chaîne de l'offre ».

[HAMMAMI, 2003]

Ce concept concerne plus précisément l'approvisionnement. Nous intéressons beaucoup plus de la définition professionnelle et scientifique de la « chaîne logistique » :

I.4.1 Définition de la chaîne logistique : « supply chain »

« La chaîne logistique est un réseau d'installations qui assure les fonctions d'approvisionnement en matières premières, la transformation de ces matières premières en

composants puis en produits finis, et [finalement] la distribution du produit fini vers le client ultime. ».[Lee et al., 1993], [HAMMAMI,2003]

Une installation peut représenter :

- Un fournisseur
- Une unité de production
- Une unité de stockage
- Une unité de manutention (transport)
- Une unité de distribution (entrepôt)
- Un client

« Une chaîne logistique est un ensemble de deux ou plusieurs entreprises liées par des flux de marchandises (physiques), d'informations et financiers. »

[TSAYet al., 1999], [HAMMAMI2003]

« 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 »

[ROTA-FRANZet al., 2001]

La première définition indique que la chaîne logistique inclut le cycle de transformation des matières premières jusqu'à l'obtention du produit fini.

La deuxième définition montre qu'au moins la chaîne logistique fait intervenir deux unités organisationnelles.

La troisième définition précise que les unités organisationnelles qui font parties d'une chaîne logistique sont liés par trois types de flux qui sont : flux physique, Flux financier et flux d'informations.

La figure (1.1) montre une représentation d'une chaîne logistique.

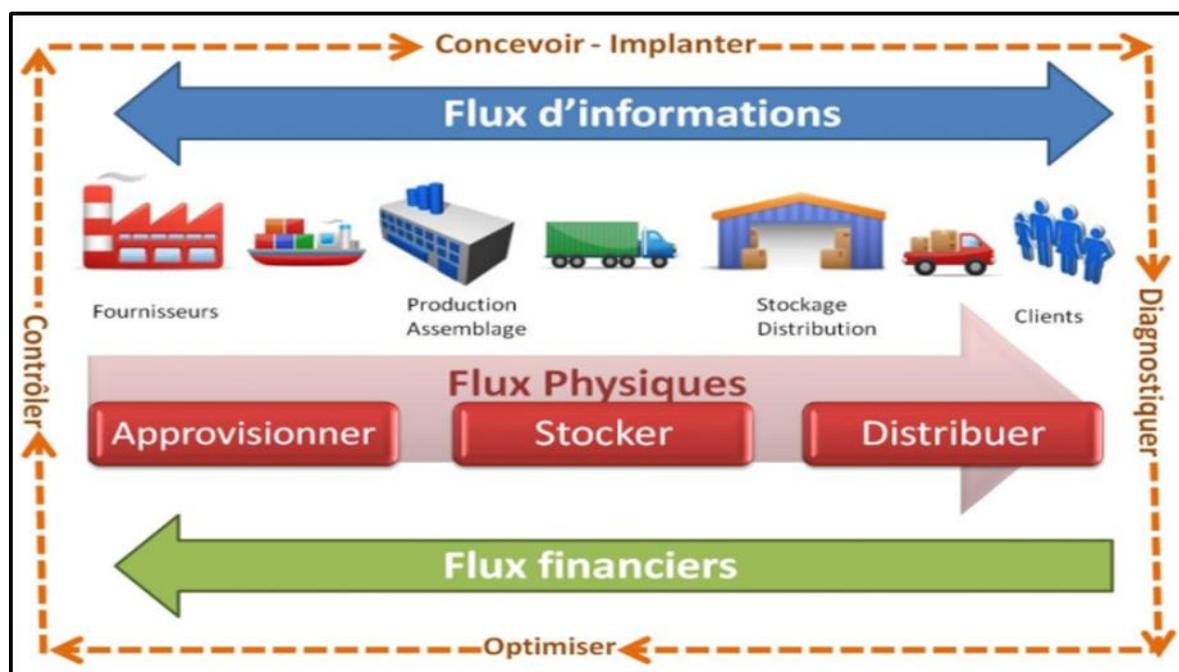


Figure 1.1 : Représentation d'une chaîne logistique [R]

I.1.2 Chaîne logistique inverse :

C'est par opposition à la chaîne logistique traditionnelle, la logistique inverse peut être définie comme le mouvement dans le sens inverse c'est-à-dire du client vers le producteur pour objectif de gérer et réduire les déchets qui viennent des activités industrielles. Elle consiste donc à retirer les produits après les avoir utilisés et de les traiter, les transformer ou les recycler.

[BENNEKROUF, 2013]

La logistique inverse c'est une démarche qui vise à planifier, mettre en place, et contrôler le niveau de performance de :

- L'utilisation des matières premières.
- Production, stocks, produits finis
- La gestion des flux d'informations allant du client vers le fournisseur pour récupérer ou créer de la valeur pour les produits vendus ainsi que pour les emballages utilisés, tout en minimisant l'influence de ces procédures sur l'environnement et minimiser l'utilisation des ressources nécessaires.

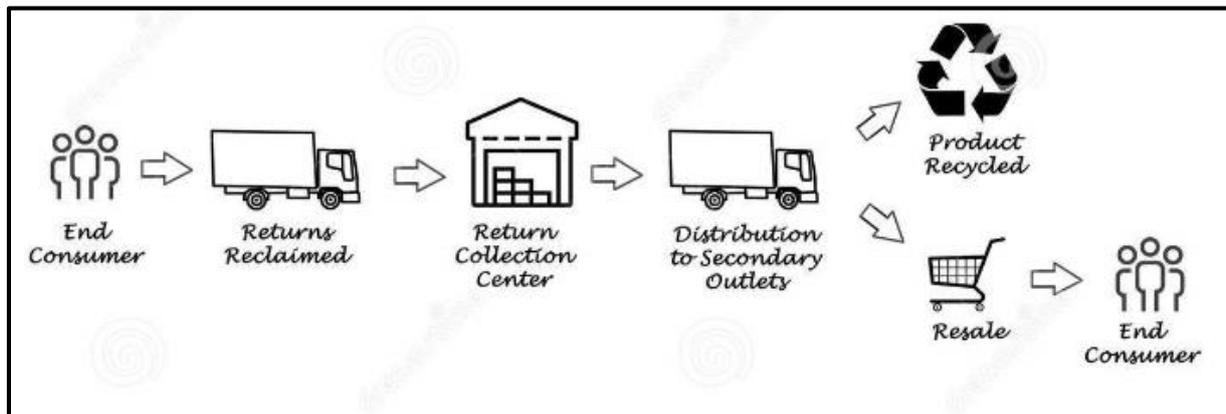


Figure 1. 2 : Représentation d'une chaîne logistique inverse[S]

I.4.2 Chaîne logistique verte :

La logistique verte est définie comme étant une démarche visant à réduire l'impact de l'industrie sur l'environnement prenant en considération le cycle de vie d'un produit dès la phase de conception jusqu'à la mise en arrêt. Cette approche repose essentiellement sur les facteurs suivants :

- La conformité aux réglementations du gouvernement et respecter et prendre en considération toujours l'environnement.
- L'instabilité des logistiques (augmentation).
- Améliorer la réputation et l'image de l'entreprise pour avoir un avantage concurrentiel face aux concurrents.
- Le rétrécissement du marché dans le cas où l'entreprise ne prend pas en compte le facteur de l'environnement.

Selon [RODRIGUE et al., 2001] La logistique verte (Green logistics) est un système de distribution et de transport qui aide à préserver et protéger l'environnement.

D'après (wu et Dunn ,1995) la logistique verte est plus bénéfique que la logistique inverse pour l'environnement parce qu'elle vise à minimiser le maximum possible des ressources, supprimer les déchets ainsi qu'améliorer la productivité donc on peut dire que la logistique verte a plus intérêts pour l'environnement.

[KUMAR, 2015]

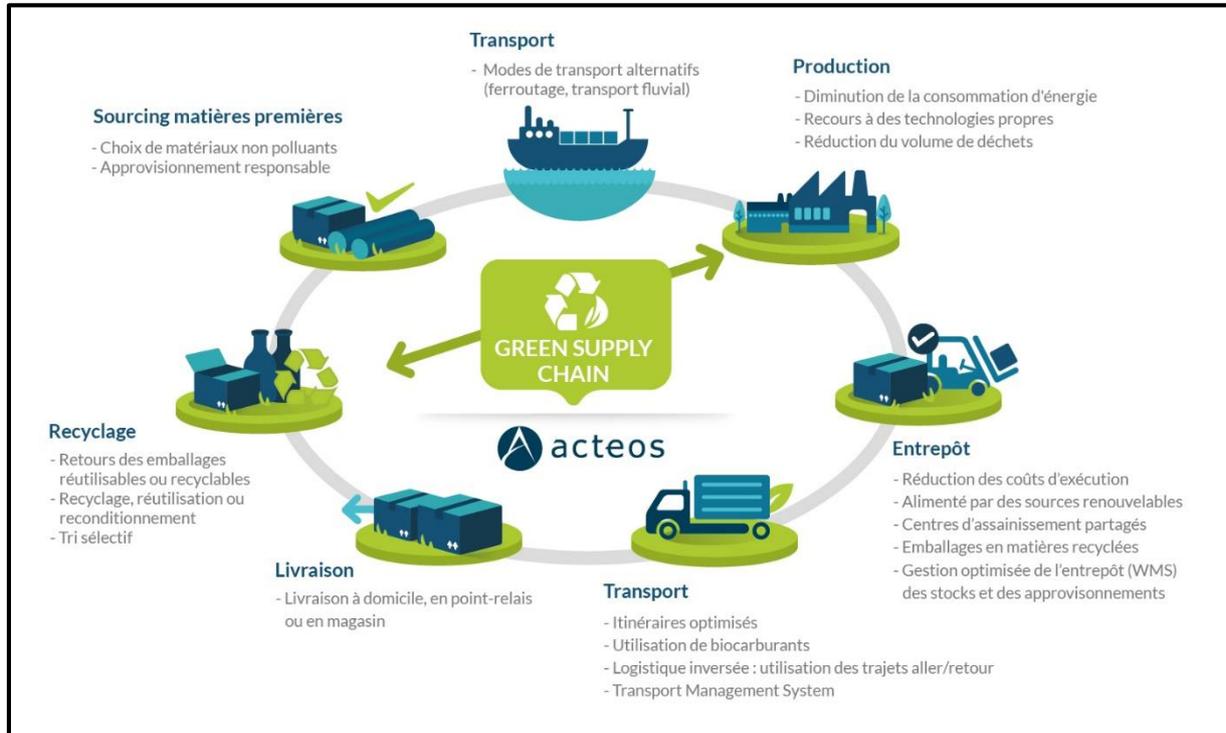


Figure 1. 3 : Représentation d'une chaîne logistique verte [T]

I.4.3 Finalité de la logistique :

Pour une chaîne d'approvisionnement des finalités peuvent être fixées à court, moyen ou à long terme :

I.4.3.1 Sur le court terme :

L'optimisation des flux physiques de l'amont vers l'aval pour il faut donc :

- Définir des programmes de production et d'approvisionnement.
- Ordonnancement des livraisons.
- Fournir des services après-vente.
- Suivre des prévisions commerciales à court terme.

I.4.3.2 Sur le moyen terme :

À partir des budgets et les plans d'actions établis la logistique a pour but de :

- Minimiser les dépenses et optimiser les frais d'investissement.
- Éliminer les tâches sans valeur ajoutée c'est-à-dire les tâches qui ne servent qu'à augmenter les coûts.
- Établir des plans d'actions pour contrôler les dépenses de l'entreprise.

I.4.3.3 Sur le long terme :

Sur le long terme l'objectif de la logistique est d'aider l'entreprise à garantir sa pérennité et garder sa place et sa bonne réputation dans le marché par la maîtrise des délais et les incertitudes.

I.4.4 Planification d'une chaîne logistique

La planification de la chaîne logistique comprend l'optimisation des activités d'approvisionnement, Produire et distribuer des produits selon la demande Prévision du client. Le but principal de la planification est de minimiser tous les niveaux d'inventaire pour répondre aux besoins des clients à temps afin d'optimiser l'utilisation des ressources. La planification des besoins de la chaîne logistique Prend une série de décisions à différents des niveaux différents. Par conséquent, les décisions de planification de la chaîne d'approvisionnement sont classées selon la fonction de réseau logistique conçu, et selon le calendrier (classification des fonctions et temps déterminé).

[HADJ-HAMOU, 2002]

I.4.4.1 Décision d'approvisionnement :

Les décisions d'achat définissent la structure du réseau d'approvisionnement. La décision principale concerne le nombre et le choix de fournisseurs, répartition du fournisseur à la base de production et définition du plan d'approvisionnement, il faut donc affecter chaque client à un fournisseur en respectant et prenant en considération des critères bien précis et déterminés dès le début.

I.4.4.2 Décision de production :

Les décisions sur la fonction de production sont principalement :

- L'emplacement de l'usine (le nombre d'usines, ou se situe chacune d'elles), distribution des matériaux aux sites de production, les problèmes les plus connus dans la logistique (location/allocation).
- Sélection des lignes de production (Articles à fabriquer sur chaque site de production).
- Prévoir les capacités des usines qui sont allouées aux projets et aux décisions inventaire.
- Les articles à stocker dans chaque centre de stockage, les niveaux de stock et le plan de production.

I.4.4.3 Décisions temporelles :

Ce type de décisions est généralement relatif aux fonctions de planification et la gestion de la chaîne logistique sur le court, moyen et long terme et ils incluent les différents horizons de la prise de décision pour la chaîne.

I.5 Les flux de la chaîne logistique :

I.5.1 Les flux physiques :

Ces flux incluent et couvrent toutes les entités qui passent par la chaîne d'approvisionnement, en particulier le flux de matières premières, les encours, et les produits finis. Ces flux généralement traversent la chaîne logistique de l'amont (la source) vers l'aval (client ultime), alors qu'ils existent d'autres flux physiques (annexes) qui peuvent traverser la chaîne logistique dans les deux directions de l'amont vers l'aval et de l'aval vers l'amont tels que les conteneurs, les outils de transport et manutention, et les produits retournés.

[BABAI, 2005]

I.5.2 Les flux financiers :

Ce sont des flux monétaires liées aux flux physiques. Ces flux cross-chain principalement de l'aval vers l'amont. Certains flux peuvent également d'aller d'amont en aval, tels que les paiements et les remboursements. L'aspect financier malgré qu'il soit très important dans l'étude logistique mais il reste difficile à estimer et il dépasse toujours notre champ d'action puisqu'il est affecté de plusieurs facteurs que nous ne pouvons jamais les contrôler étroitement.

[BABAI, 2005]

I.5.3 Les flux d'informations :

Il s'agit des échanges d'informations entre les divers acteurs qui appartiennent à la chaîne logistique. Ces informations peuvent être liées à l'état du système, aux niveaux des inventaires, niveaux des encours et aussi à la demande du client. Les flux d'information peuvent se produire dans les deux directions d'amont vers l'aval ou bien d'aval vers l'amont

[BABAI,2005]

I.6 Conception des chaînes logistiques :

La conception de la chaîne logistique comprend généralement plusieurs étapes, de l'approvisionnement à la distribution. Pour les activités d'achat, les décideurs doivent identifier des fournisseurs potentiels pour la sélection afin de fournir des matières premières, des pièces et des produits semi-finis à différentes usines. Pour les activités de production, on doit déterminer le meilleur emplacement pour ses usines afin d'assurer la performance et la rentabilité des activités de production. Afin d'avoir une chaîne de distribution efficace, nous devons d'abord déterminer le nombre et l'emplacement de ses différents centres de distribution. Une fois que nous avons sélectionné différents fournisseurs, usines et centres de distribution, nous devons trouver la meilleure structure de connexion pour relier ces sites. Par conséquent, les décideurs ont identifié différentes méthodes de contact et moyens de transport pour assurer la connectivité entre les sites. Lors de la conception, nous devons tenir compte des contraintes économiques, sociales et environnementales, tout en minimisant les coûts et en maximisant la satisfaction des clients.

[DING, 2004]

Lors de la conception de la chaîne logistique, l'objectif principal est d'optimiser l'investissement dans différents lieux économiques, la société et l'environnement pour différents sites, de minimiser les coûts d'exploitation de toutes les activités de la chaîne et de maximiser la satisfaction des clients finaux. La conception de la chaîne d'approvisionnement nécessite une série de décisions à prendre dans des délais différents (court terme, moyen terme et long terme). Ces décisions peuvent être divisées en trois niveaux : les décisions stratégiques, les décisions tactiques et les décisions opérationnelles. (Voir la figure 1.4). [MERZOUK, 2007]

[New, 1997]

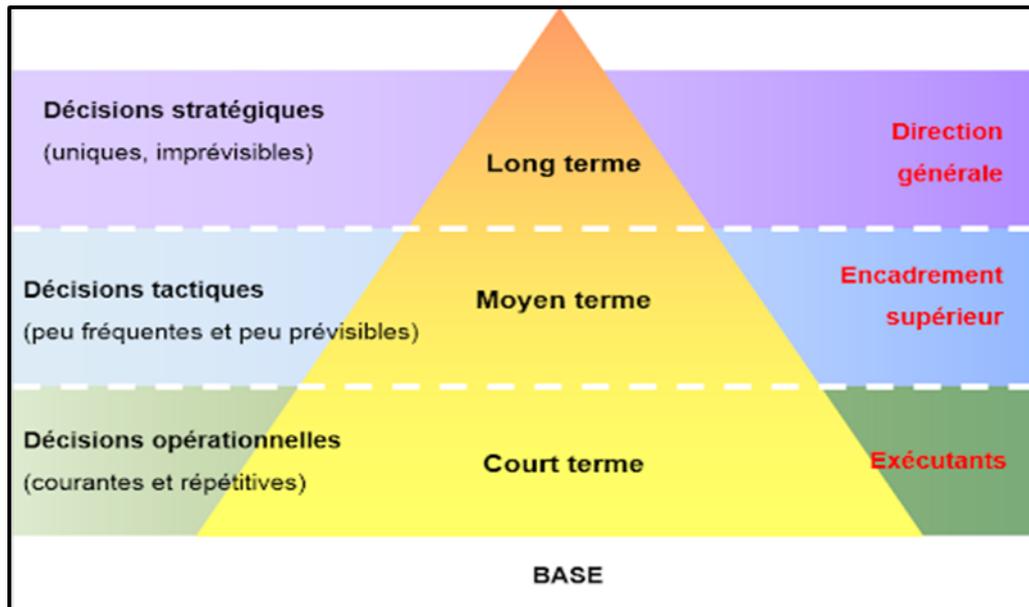


Figure 1. 4: Pyramide des niveaux de décisions[U]

I.6.1 Les niveaux de décision dans la chaîne logistique :

Les décisions dans la chaîne logistique sont divisées en trois niveaux :

I.6.1.1 Prise de décision au niveau stratégique :

Ce niveau est appelé management stratégique ou planification stratégique, qui regroupe toutes les décisions stratégiques de l'entreprise. Ces décisions sont prises par les directeurs généraux et sont des orientations à long terme (de 6 mois à plusieurs années), par exemple le choix des fournisseurs et des détaillants, la décision d'ouvrir ou de fermer certaines bases de production ou leur délocalisation, allocation, etc. Une nouvelle zone de marché vers un centre de distribution (entrepôt), à la recherche de nouveaux partenaires industriels, mais aussi au développement de nouveaux produits, de la configuration d'usine, de ses modes de fonctionnement et des objectifs financiers à atteindre.

Le management stratégique fait partie de la conception de la chaîne, impliquant le choix des emplacements des sites, la définition des rôles et des tâches, les modes de transport, les choix technologiques et la distribution des produits sur ces sites. La direction vise à prendre des décisions stratégiques de première importance [BOUCHRIHA, 2002] :

- La sélection et choix des fournisseurs.
- Nombre, emplacement et capacité des sites de fabrication.

- Distribution des produits aux installations sélectionnées (ouvertes) de la chaîne.
- Ouvrir/fermer certaines installations dans les chaînes de magasins.
- Les moyens de transport à utiliser.
- La quantité des matériaux sur chaque site (fournisseur, usine, inventaire, client, etc.) dans la chaîne.[BOUCHRIHA, 2002]

I.6.1.2 Décision au niveau tactique :

Les décisions de ce niveau sont prises par les cadres de la production et les chefs d'atelier. Et qu'ils sont à moyen et long terme (de quelques semaines à quelques mois) qui devront être mises en application pour déployer la stratégie décidée par l'entreprise.

Les décisions tactiques, il s'agit de produire au moindre coût pour satisfaire les demandes prévisibles en s'inscrivant dans le cadre fixé par le plan stratégique de l'entreprise (ressources matérielles et humaines connues). [DING, 2004]

Le niveau tactique implique également la coordination des opérations entre les installations (clients / fournisseurs, production / distribution, inventaire / distribution) et la gestion des stocks dans la chaîne. Trois types de coordination ont été identifiés :

A/La coordination dans le secteur d'activité client / fournisseur implique principalement : [BOUCHRIHA, 2002] :

- Connaître la taille du lot d'approvisionnement.
- Comprendre les points de commande associés à chaque fournisseur.
- Définir les délais de livraison.
- Définir les niveaux de stock.

B/La coordination dans l'interface production / distribution fait principalement référence à [BOUCHRIHA, 2002] :

- La définition de la taille des lots de production.
- Définition du délai de production.
- Définir la date limite d'inventaire des produits semi-finis.

C/La coordination au niveau de l'interface inventaire / distribution implique principalement : [BOUCHRIHA, 2002] :

- La meilleure taille pour le lot d'expédition.
- Déterminer la meilleure stratégie d'allocation (soit directement au client, soit via le centre de distribution) [COHEN et al, 1988] ont développé quatre sous-modèles optimisés pour la gestion et le contrôle :
- D'approvisionnement des matières premières.
- Des processus de production.
- Du processus d'inventaire(stocks) des produits finis.
- Du Processus de distribution.

Des heuristiques sont utilisées pour permettre des liens entre différents modèles pour réaliser l'optimisation de l'ensemble de la chaîne d'approvisionnement.

[COHEN et al, 1988]

I.6.1.3 Décision au niveau opérationnel :

Prendre des décisions opérationnelles pour assurer le fonctionnement quotidien de la chaîne. Ces décisions comprennent la gestion des stocks, la gestion du travail, la gestion des équipements, la planification de la production, etc. Ces trois types de décisions de gestion de la chaîne d'approvisionnement sont différents dans au moins trois éléments

[BOUCHRIHA, 2002] :

- Dans le délai considéré. Les décisions opérationnelles sont prises à l'heure. Les décisions tactiques impliquent une planification à moyen terme (moins de 18 mois). Les décisions stratégiques concernent la planification sur le long terme.
- Par le niveau d'agrégation : les décisions opérationnelles sont prises au niveau de l'atelier, les décisions tactiques sont prises au niveau de l'usine et les décisions stratégiques au niveau de l'ensemble de l'entreprise.
- Par le niveau de responsabilité : les décisions opérationnelles sont prises par les agents de maîtrise, les décisions tactiques par les cadres et les décisions stratégiques par la direction générale de l'entreprise

[DING, 2004]

I.7 Classes de base de chaîne logistique :

I.7.1 Les usines :

Une usine fait référence à un bâtiment ou ensemble des bâtiments utilisés pour la production industrielle. Elle convertit généralement des matières premières ou des produits semi-finis en produits finis, mais aussi en énergie. L'usine est l'une des entités clés de la chaîne logistique (d'approvisionnement).

I.7.2 Les centres de distribution :

Le rôle de ce site dans la chaîne logistique peut se résumer dans 3 étapes : la réception, le stockage et le transport de différents types de produits (matières premières, composants, produits finis, etc.).

On peut donc affecter des attributs au centre de distribution tels que :

- La capacité d'accueil.
- La capacité de stockage.
- Fiabilité de réseaux de transport.
- Cout unitaire de produit.
- Les prix de chargement et déchargement.

[DING, 2004]

I.7.3 Les fournisseurs :

Les fournisseurs sont des particuliers ou des entreprises qui fabriquent, traitent, conditionnent ou installent des produits contrôlés, ou importent ou vendent ces produits. Dans la chaîne d'approvisionnement, le fournisseur est généralement la source du flux de matières. Les fournisseurs peuvent proposer plusieurs produits avec des caractéristiques différentes : matières premières, composants, produits semi-finis et même produits finis. Le prix de vente a toujours été et reste un attribut clé dans le choix d'un fournisseur. Cependant, aujourd'hui, le

prix d'achat lui-même n'est pas le facteur déterminant. Le rapport qualité / prix est bien plus que le prix d'achat. En outre, le délai de livraison et la qualité du service sont des éléments clés qui seront utilisés dans la stratégie de chaîne d'approvisionnement qui sera adoptée dans notre évaluation.

Les principaux critères pour le choix d'un fournisseur sont donc :

[DING, 2004]

- Achat à un prix unitaire HT (hors taxe).
- Taux du tarif douanier.
- Taux de produits qui ne répondent pas aux spécifications (non conformes).
- Délai de livraison.
- La quantité minimale acceptée par commande.

I.7.4 Le transport :

Dans la chaîne logistique, les principales entités (fournisseurs, usines, centres de distribution et clients finaux) sont reliées entre elles par des liens impliquant différents modes de transport.

Principaux attributs : [DING, 2004]

- Capacité de charge maximale
- Quantité minimale requise pour commencer le transport
- Temps de transport moyen
- Écart type du temps de transit
- Prix unitaire de transport
- Frais de transport par départ ou expédition
- Taux de consommation de carburant

I.7.5 Liaison d'information :

Les liens d'information permettent de relier les sites entre eux, principalement en termes de commande (achat, approvisionnement et de production ...).

Les principaux facteurs sont : [DING, 2004]

- Expéditeur d'ordres
- Récepteur d'ordres

I.7.6 L'entreprise :

Une entreprise est une organisation ou une unité institutionnelle dont les projets sont transformés en stratégies, politiques et plans d'action. Son objectif est de produire et de fournir des marchandises ou des services à un groupe de clients ou d'utilisateurs.

[DING, 2004]

I.7.7 Client :

Dans un cadre économique, un client se rapporte à une personne physique ou morale qui décide d'acheter des biens ou des services à un fournisseur occasionnellement ou habituellement.

Les principaux critères pour un client sont :

- La demande moyenne.
- L'écart type de la quantité demandée.
- Fréquence de demande (constante ou aléatoire).
- Le comportement du client (patient ou impatient).
- Le temps de réception requis.
- Classement des services selon la priorité.

I.8 Problématiques liées à la chaîne logistique :

I.8.1 Impacts et interactions entre les décisions stratégiques, tactiques et opérationnelles :

Lors de la planification de la « supply chain », on s'intéresse souvent à l'efficacité des choix stratégiques à des niveaux de décision complètement différents (tactique, opérationnel). Ces trois unités de sélection sont traitées dans l'ordre : stratégie, puis tactique, puis action. En raison de la complexité du problème, peu de solutions considéreront à la fois stratégie, tactique et options opérationnelles, et en raison des changements et dynamiques de la chaîne (fluctuations de la demande client, délai de livraison, transport incertain, etc....)

[DING,2004]

I.8.2 Incertitude tout au long d'une chaîne logistique :

L'existence de l'incertitude dans la chaîne logistique est relative à l'instabilité de l'environnement. Et cette incertitude peut se présenter sous forme :

- Des demandes clients
- Processus de fabrication
- Réseaux de transport
- Processus d'approvisionnement (c'est un facteur important parce qu'il se change constamment et de manière imprévisible).[DING, 2004]

I.8.3 Nécessité de passer de l'optimisation monocritère a l'optimisation multicritère :

La conception de la chaîne d'approvisionnement est un problème de prise de décision multicritères, y compris des normes quantitatives et qualitatives. Dans la littérature existante sur la conception de la chaîne logistique, le coût total (y compris toutes les activités de la chaîne d'approvisionnement) est généralement utilisé comme seul critère de décision à optimiser. Dans une économie dominée par des clients aux enjeux sociaux et environnementaux de plus en plus intenses, l'optimisation des standards économiques ne suffit plus.[DING,2004]

Principaux processus d'une entreprise :

I.8.3.1 Le processus d'approvisionnement :

Le processus d'approvisionnement est utilisé pour fournir des composants de fabrication et se compose de deux étapes :

- Étape de sélection des fournisseurs : la sélection des fournisseurs est basée sur le prix, le temps, la quantité de transport, la qualité des produits transportés et l'acceptation par le fournisseur de l'évolution des demandes. Vous pouvez fournir plusieurs fournisseurs pour un produit (quantité partagée), tout comme il peut y avoir un fournisseur pour chaque produit, ce qui minimise le risque d'interruptions de livraison.
- Étape de configuration de la commande :
Y compris la passation des commandes à ces fournisseurs en fonction de la production à réaliser, et la vérification du comportement de livraison des composants. En d'autres termes, la livraison doit inclure [FRANCOIS, 2007] :
 - La qualité voulue.
 - La quantité voulue.
 - Le moment de livraison voulu.

I.8.3.2 Le processus production :

Selon Francois [FRANCOIS, 2007] : Le processus de Production concerne l'ensemble des transformations qui peuvent subir les éléments pour fournir le produit fini de l'entreprise. La production est la composante essentielle de la chaîne d'approvisionnement, elle consiste à transformer la matière première pour la création des produits et/ou services en combinant des facteurs de production. Les stratégies mises en œuvre dans le mode de production (planification, ordonnancement, décision de l'échelle des lots de production, des lots de transfert, etc.) sont unies pour améliorer le flux d'assemblage.

I.8.3.3 Le processus distribution :

D'après Jihéne [TOUNSI, 2009], La distribution comprend toutes les activités qui prennent en charge les commandes des clients et leur livraison. Elle comprend la gestion des commandes (saisie et traitement des commandes), la gestion du transport et la livraison aux clients.

D'après François [FRANCOIS, 2007], Le processus de distribution passe par la livraison des marchandises finies aux clients, et passe par l'amélioration du réseau de distribution : l'organisation et la sélection des choix celle du transport, la sélection des couches. (Ou une agence intermédiaire), le placement du réseau de distribution et des entrepôts et leurs modes de gestion, afin de rechercher l'équilibre le plus efficace entre qualité de service et prix économiques, la marchandise peut être transportée en grande quantité par train et triée au cours un entrepôt, puis transportés par camion jusqu'aux clients d'une même zone géographique.

I.8.3.4 Le processus de vente :

Comme pour un processus de production, le terme de processus de vente fait référence à le fait qu'un bien ou un service détenu par un vendeur est cédé à un acheteur avec un prix de vente.

D'après François [FRANCOIS, 2007] Le processus de vente imposé par la direction commerciale établit des relations avec les clients (négociation des coûts et des délais, enregistrement des commandes, etc.) pour chasser une compréhension plus solide du marché. Cette méthode commerciale est également à blâmer pour la prédiction de la demande de

processus et les aspects commerciaux de déségrégation tels que la durée de vie du produit pour prévoir les tendances des ventes

D'après Mouloua [MOULOUA, 2007], La fonction de vente est la fonction ultime de la chaîne d'approvisionnement, et sa puissance dépend de la performance des fonctions en amont. Si nous avons tendance à optimiser dans les étapes précédentes, nous allons promouvoir le travail des vendeurs, car ils seront prêts à fournir beaucoup de coûts compétitifs par rapport aux concurrents, sinon le profit sera terriblement mince et donc le profit ne sera pas géant. Les étapes et les opérations de la méthode de vente sont généralement partagées par l'équipe de promotion en amont et donc l'équipe de vente en aval, et son efficacité dépend de la performance de la performance en amont.

Les étapes et les opérations du processus de vente sont généralement partagées entre l'équipe de marketing en amont et l'équipe commerciale en aval.

1. Comment renforcer son exploration ?
2. Premier contact et qualification.
3. L'approche.
4. L'évaluation des besoins.
5. La présentation.
6. Fermeture.
7. Suivi de la mise en œuvre

I.8.4 État de l'art des chaînes logistiques :

L'étude logistique est un sujet d'actualité qui ne cesse d'avoir des nouvelles problématiques posés sur le terrain économique et qui oblige les chercheurs de trouver des solutions à propos. Le développement économique passe directement par le développement des chaînes d'approvisionnement qui représentent une infrastructure rigide pour s'épanouir dans les secteurs industriels et économiques. Au cours des années beaucoup d'études et travaux ont été élaborés dans le cadre de développement des chaînes d'approvisionnement.

Pour pouvoir concrétiser le concept de la chaîne logistique nous avons établi une littérature liée aux travaux et études accomplis dans ce stade parmi lesquels nous citerons :

- [TANONKOU, 2007] : L'étude est basée essentiellement sur des demandes et des délais d'approvisionnement aléatoires dans le but de l'intégration des décisions de la localisation des sites et choix des fournisseurs pour un réseau de distribution. L'auteur a supposé que les détaillants aillent effectuer des demandes aléatoires pour un seul type de produits et que les délais de livraison entre fournisseurs et chaque centre de distribution (DCs) sont constants. Le problème consiste à localiser des centres de distribution, choisir des fournisseurs et les affecter aux (DCs) en utilisant des méthodes de relaxation lagrangienne.
- [MOULOUA, 2007] touchait les problèmes d'ordonnements coopératifs pour les chaînes logistiques, dans le but de maîtriser les délais de livraison et optimiser les transports entre les divers acteurs de la chaîne logistique l'auteur a proposé un modèle d'ordonnement juste à temps qui doit être résolu à l'aide des algorithmes génétiques.

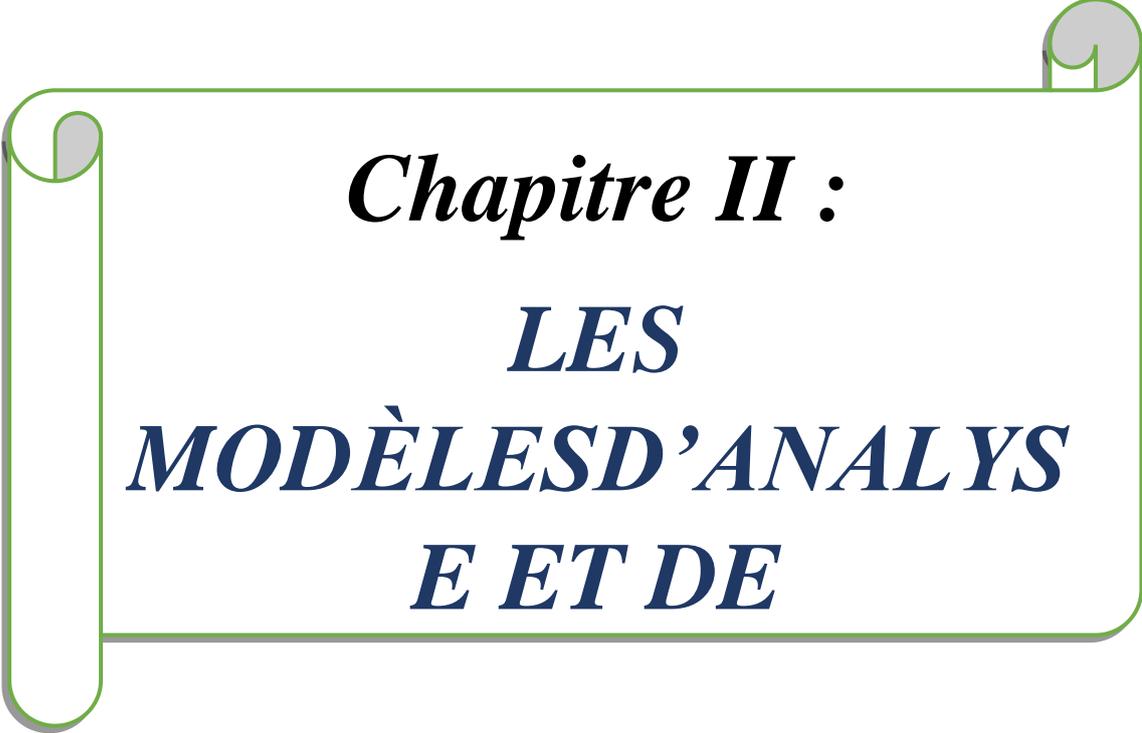
- [ROLLAND et al., 1996] les auteurs de ce travail ont présente une solution pour les problèmes p-médian par l'application d'une heuristique, puis ils ont comparé les résultats obtenus avec d'autres résultats des autres heuristiques.
- [CANOS et al., 2001] les auteurs de ce travail ont défini le problème classique de localisation p-médian comme étant localiser des p installations pour but de satisfaire toute la demande tout en minimisant les frais de transport, ils ont supposé que les couts de transport soient relativement liés à la quantité demandée et la distance parcourue.
- [BOUDAHRI et al.,2011] ont établi une étude pour la rénovation de la distribution des viandes de poulet à partir des abattoirs vers les détaillants sur la ville de Tlemcen. Cette étude vise à minimiser les couts globaux de la chaine par la localisation et allocation des abattoirs (minimiser les couts de localisation et de transport) tout en respectant les capacités des abattoirs et des véhicules.
- [HADJILA et MANSOURI,2016] : ont travaillé sur la distribution des produits pharmaceutiques dans la ville de Tlemcen. Ce cas d'étude consiste à localiser l'ensemble des pharmacies (amas) en utilisant le modèle CCCP (capacitated centered clustering problem) de tels sorte les regrouper selon la distance pour minimiser les distances entre le centre de gravité des amas et les pharmacies, ensuite le traitement du problème de localisation/allocation et finalement la planification des tournées de véhicules en utilisant le modèle TSP (Traveling Sales man Problem) partant des différents entrepôts pour visiter l'ensemble des pharmacies et revenir au point de départ.
- [RAHMOUNI et SOUICI,2019] ont travaillé sur l'amélioration du système de traçabilité dans une chaine logistique agroalimentaire pour rendre le transfert des données plus facile et pratique spécialement dans la gestion des stocks en se basant sur des nouvelles technologies puissantes tels que l'internet des objets (IOT), Blockchain pour le stockage et transfert des données dans le cadre d'améliorer la gestion de la chaine logistique et améliorer ses performances.
- Les applications des RFID dans la gestion de chaîne logistique Agroalimentaire (Viande Halal) Présenté par : BRIKI Sahar Imane (Chaine Logistique), KEDDOURI Thabet Nacer Eddine.

En s'inspirant de l'ensemble de ces travaux et autres, dans cette étude nous sommes à propos d'analyser la chaine de distribution des produits alimentaires, particulièrement nous intéressons à la distribution du lait pasteurisé afin de trouver les failles pour pouvoir concevoir un réseau de distribution qui satisfait les besoins des clients et assure la disponibilité du lait au cas des crises par la localisation des points de vente (plateformes) dans les régions de Tlemcen en appliquant les modèles p-médian et max-covering, et introduire l'approche goal programming par la suite à l'aide du solver lingo 18.0 et le logiciel d'information géographiques QGIS.

I.9 Conclusion :

Nous avons vu dans ce chapitre l'histoire de développement de la chaine logistique, définition de la chaine logistique selon plusieurs auteurs et plusieurs références, ainsi que ses trois types (chaine logistique simple, chaine logistique verte, et chaine logistique inverse).Nous

avons parlé aussi des types de décision stratégiques qui sont exercés entre les acteurs d'une chaîne d'approvisionnement, nous avons aussi cité les classes de base et quelques problématiques liées à la conception des chaînes logistiques, ensuite nous avons fermé ce chapitre par une revue de la littérature sur le domaine des chaînes logistiques.



Chapitre II :
LES
MODÈLES D'ANALYS
E ET DE

II.1 Introduction :

Dans ce chapitre nous examinerons des méthodes de prise de décision monocritère et, citer quelques problèmes usuels dans les chaînes d'approvisionnement tels que la p-médiane, p-centre, problème de recouvrement... Ensuite nous parlerons des méthodes de d'aide à la décision multicritères, et particulièrement nous allons parler de l'approche goal programming, ses avantages et inconvénients, ses différents types avec leurs modèles mathématiques et à la fin nous allons conclure par une comparaison entre l'approche goal programming et la programmation linéaire.

II.2 Modèles d'analyse :

La majorité des modèles d'analyse sont des modèles de programmation linéaire mixte de type de décision "faire ou ne pas faire", par exemple l'algorithme simplexe, procédure de séparation et d'évaluation, méthodes heuristiques (telles que les méthodes basées sur la relaxation) Lagrangienne et même les métaheuristiques. En fait le problème de la reconfiguration du réseau logistique est un problème de décision Complexe, impliquant de nombreux éléments. Une étude préliminaire de la littérature nous permet de déterminer certains facteurs couramment utilisés, prises en considération et influencent l'ampleur du problème à résoudre. Ces différents facteurs permettent également de caractériser le modèle à étudier.

[DURRIEU et VALETTE-FLORENCE, 2005]

Parmi ces facteurs nous citons :

II.2.1 La nature de la décision :

Lorsqu'une entreprise a identifié une série de projets en fonction de sa de son réseau logistique et/ou sa stratégie de redéploiement, elle doit encore décider quel projet peut être entrepris qui permet d'avoir un avantage concurrentiel. Par conséquent, l'entreprise fait face le problème de la prise de décision appelé « problème de reconfiguration du réseau logistique ». Il existe quatre types de décisions basées sur la nature du projet potentiel lié à cette question :

- La décision du centre de production et du point de Distribution.
- Décisions liées aux flux de matières entre ces sites.
- Décisions liées aux modes de fonctionnement du réseau.
- Décisions liées aux décisions entre réseaux, investissement en main-d'œuvre et en équipement dans chaque site.

[SIMCHI-LEVI, 2008]

II.2.2 Nombre des étapes dans un réseau logistique :

Un paramètre très important dans la résolution des problèmes qui est la taille du problème Par exemple, il peut y avoir un problème avec une partie du réseau, un autre problème qui couvre un sous-réseau ou plusieurs sous-réseaux. D'habitude, nous pouvons prendre en compte deux niveaux dans les différents types des problèmes. Les sites de niveau 1 (centres de production) fournissent des produits aux sites de niveau 2 (centres de distribution) pour qu'ils seront livrés aux clients ultimes par la suite.

[DURRIEU et VALETTE-FLORENCE, 2005]

II.2.3 Nombre de périodes considérées :

Pour un problème quelconque, nous pouvons prendre en considération une seule ou plusieurs périodes. Le fait de développer un plan stratégique sur plusieurs périodes de temps rendre compliquer de plus en plus le modèle.

[BOUDAHRI, 2013]

II.2.4 Les objectifs à atteindre :

L'objectif principal c'est de minimiser le cout global de la chaine. Nous pouvons structurer les données sous forme linéaire et les regrouper en une composante fixe et une composante variable. Réduire les couts va mener à des économies très importants, des gains entre parenthèses. En général, les entreprises ne ciblent pas une seule norme pour optimiser les coûts, mais s'intéressent également à la maximisation d'autres normes financières.

[BOUDAHRI, 2013]

II.2.5 Contraintes prises en compte :

Nous pouvons prendre en considération plusieurs contraintes simultanément dans un même problème. Habituellement, avec Services (satisfaction des besoins) et contraintes liées aux capacités et performances des installations de la chaîne logistique.

II.2.6 Internationalité du réseau logistique :

Il très important lorsque le réseau logistique opère sur le marché international Compte tenu de nombreux facteurs qui affectent sur le bénéfice, Tels que les frais de douane, les taxes, les taux de change et les réglementations locales. Nous faisons la première différence entre les modèles développés pour le réseau logistique actuel dans un pays prend en compte ce caractère international.

[DURRIEU et VALETTE-FLORENCE, 2005]

II.2.7 Analyses multicritères :

Les décisions multicritères désignent l'évaluation et le suivi du développement d'un ensemble de problèmes à partir d'un ensemble des contraintes prédéfinies. Pour un problème de localisation par exemple les critères peuvent être la distance, la durée, cout de transport... etc.

[DASKIN, et al, 2005]

II.3 Problèmes de localisation/allocation :

Le problème de localisation (location problem) fait partie de la recherche opérationnelle consiste à localiser un ensemble d'installations (des usines, entrepôts, restaurants, stations d'essence...) pour but de satisfaire les clients tout en minimisant les couts de déplacement (transport) prenant en considération d'autres contraintes relatives aux performances du centre de distribution et/ou aux clients.

[DASKIN et al, 2005]

II.3.1 Classification de Location-Allocation problem :

Ce genre de problème est caractérisé par 4 paramètres selon (Revelle et Eiselt,2005) qui sont : les clients, les installations, la zone qui couvre chacun des clients et installations, la distance et durée de temps qui séparent les clients des installations. On se basant sur les études de Revelle et Eiselt,(2005), Revelle et al (2008), Azarmand et Neishabouri (2009), Beaumont (1981), Amour et Juel (1982), nous avons pu classer les problèmes de location-allocation selon les catégories suivantes :

II.3.1.1 Classification selon la demande des clients :

On se basant sur la certitude des informations relatives à la demande des clients et leurs exigences on peut différencier un modèle stochastique d'un modèle déterministe ; Si le nombre des clients, leurs emplacements et les demandes sont certainement connues, le modèle sera appelé déterministe, par contre si les demandes des clients sont établies avec des distributions de probabilité dans ce cas le modèle sera appelé stochastique.

[HADJILAet MANSOURI, 2016]

II.3.1.2 Classification selon le nombre d'installations :

On peut gérer le modèle localisation-allocation selon le nombre d'installations (une seule ou plusieurs installations), dans le but toujours c'est de trouver le minimum d'installations qui couvrent tous les points de demande prenant en considération la contrainte de distance prédéfinie (toregas et al.1971), si les capacités à fournir par l'ensemble des installations aux clients sont limitées le modèle donc est appelé avec capacité (capacitated), autrement on dit sans capacité (uncapacitated).Il est aussi légitime de faire le classement selon le service fourni par les installation (mono-service ou multi-services).

[DASKIN et al, 2005]

II.4 Les problèmes usuels dans la logistique :

II.4.1 Problèmes de recouvrement :

Dans ce genre de problème nous devons fournir des dépôts pour tous les clients, (généralement les clients les plus proches). L'affectation des clients à l'entrepôt ne fait pas partie de la décision à prendre. Nous sommes donc à propos de traiter un problème de couverture fixe (SCP). L'objectif est de trouver des sites à ouvrir pour toucher les clients au moindre coût. Ce problème peut être formulé comme suivant :

[AIKENS, 1985]

$$\text{Min } Z = \sum_{j=1}^J (F c_j * X_j) \quad (1.1)$$

Sous les contraintes :

$$\sum_{i=1}^I (Y_{ij} * X_{ij}) \geq 1 \quad \forall j \in J \quad (1.2)$$

$$X_j, Y_{ij} \in \{0, 1\} \quad \forall i \in I, j \in J \quad (1.3)$$

Où :

i : indice des points de demande (clients).

j : indice des sites potentiels d'offre (usines).

Fc_j : Coût d'ouverture d'une usine dans la région j .

Variables de décisions :

$$X_j = \begin{cases} 1 & \text{si on localise l'usine } j \\ 0 & \text{si non} \end{cases}$$

$$Y_{ij} = \begin{cases} 1 & \text{si le site } j \text{ peut couvrir le client } i \\ 0 & \text{sinon} \end{cases}$$

- L'équation (1.1) représente la fonction-objective.
- La contrainte (1.2) assure que tous les clients sont servis au moins une fois.
- La contrainte (1.3) détermine la nature binaire des variables de décision.

II.4.2 Problème de recouvrement maximale :

Parmi les problèmes qui s'imposent dans le problème de recouvrement c'est que le nombre des sites à localiser pour couvrir tous les clients peut excéder le nombre des sites possibles à ouvrir en réalité. En outre il est préférable de couvrir un client avec une grande demande par rapport à plusieurs clients qui ont une faible demande dans l'ensemble.

Ces deux notions ont envisagé les chercheurs de fixer le nombre des sites possibles à implanter et maximiser l'ensemble des demandes couvertes. [CHURCH et REVELLE, 1974] a proposé un modèle qui permet de maximiser le couvreurment « maximum covering location problem », ce modèle est représenté comme suivant :

[DASKIN, 2011], [Notre traduction]

Variable de décision :

$$z_i = \begin{cases} 1 & \text{si le client } i \in I \text{ est couvert} \\ 0 & \text{sinon} \end{cases}$$

h_i : représente la demande du client i .

$$\max = \sum_{i \in I} h_i * z_i \quad (2.1)$$

Sous les contraintes suivantes :

$$z_i \leq \sum_{j \in J} a_{ij} X_j \quad \forall i \in I \quad (2.2)$$

$$\sum_{j \in J} X_j \leq P \quad (2.3)$$

$$X_j \in \{0, 1\} \quad \forall j \in J \quad (2.4)$$

$$Z_i \in \{0, 1\} \quad \forall i \in I \quad (2.5)$$

- La fonction objective (2.1) maximise le nombre des demandes couvertes.
- La contrainte (2.2) impose que chaque demande ne peut être couverte si seulement si un des sites qui la couvre sera ouvert.
- La contrainte (2.3) indique le nombre des sites qu'on peut ouvrir donc on ne peut pas ouvrir plus de p sites.
- Les deux contraintes (2.4) et (2.5) montre la nature binaire des variables de décision z_i et X_j .

Dans ce problème nous avons exigé le nombre des sites à ouvrir à ne pas dépasser les p sites du coup si le nombre des sites candidats est égal au nombre des points de demande, la solution optimale nous pouvons avoir un nombre des sites à ouvrir inférieur aux nombres des points de demande.

II.4.3 Problèmes P-médianes :

Le problème p-médianes (p-centres) consiste à déterminer la configuration géographique de l'unité d'alimentation afin de minimiser la distance totale parcourue sous une série de contraintes énoncées par l'utilisateur. Ce modèle garantit une couverture médiatique efficace. L'avantage principal de la p-médiane est qu'elle est flexible et s'adapte facilement aux problèmes spécifiques posés par les utilisateurs et peut être résolu par des méthodes efficaces, rapides et flexibles.

[VERTERet DINCER, 1992]

Le modèle non seulement se concentre sur la forme et l'emplacement optimal de la zone, mais aussi il peut suggérer le nombre idéal de services et simuler des solutions sous diverses contraintes et hypothèses. Le problème réside donc dans le choix de la configuration géographique de l'unité d'approvisionnement. Afin de minimiser la somme pondérée de la distance parcourue par l'utilisateur. Le modèle mathématique de ce problème est présenté comme suivant :

[GEOFFRION, 1974]

$$\text{Min } Z = \sum_{i=1}^I \sum_{j=1}^J (d_i * c_{i,j} * y_{i,j}) \quad (3.1)$$

Sous les contraintes :

$$\sum_{i=1}^I \sum_{j=1}^J Y_{i,j} = 1 \quad i \in I \quad (3.2)$$

$$\sum_{j=1}^J (X_j) = P \quad (3.3)$$

$$Y_{i,j} \leq X_j \quad \forall i \in I, \forall j \in J \quad (3.4)$$

$$X_j, Y_{i,j} \in \{0, 1\} \quad \forall i \in I, \forall j \in J \quad (3.5)$$

Où :

i : indice des points de demande.

j : indice des sites potentiels d'offre.

d_i : Demande totale du client i .

$c_{i,j}$: Coût de transport entre i et j .

P : nombre de sites à localiser.

Variable de décisions :

$$X_j = \begin{cases} 1 & \text{si on doit localiser l'usine } j; \\ 0 & \text{si non.} \end{cases}$$

$$Y_{i,j} = \begin{cases} 1 & \text{si le client } i \text{ sera affecté à l'usine } j; \\ 0 & \text{sinon.} \end{cases}$$

- L'équation (3.1) représente la fonction-objective.
- La contrainte (3.2) impose que chaque client soit affecté à un seul site.
- La contrainte (3.3) détermine le nombre de sites.
- La contrainte (3.5) montre la nature binaire des variables de décision.

II.4.4 Le problème du centre (center problem) :

Le problème central est l'extension du problème P-médian. Prenons un exemple où l'objectif consistait à minimiser la distance entre l'entrepôt et le client. Hakimi [HAKIMI, 1964], Mineaka [MINIEKA, 1970] et Tansel et al. [TANSEL et al, 1982]) ont proposé plusieurs versions pour ce problème. Les recherches de [KARIV et HAKIMI, 1979] montrent que le problème réside dans le NP. Les problèmes de recouvrement, de P -médian et du centre, constituent une base rigide des modèles déterministes rencontrés dans la théorie des problèmes de localisation.

[BOUKLI-HACENE, 2020]

La modélisation mathématique du problème du p-centre est la suivante :

$$\text{Min}Z = \sum_{i=1}^I \sum_{j=1}^J (d_i * c_{i,j} * y_{i,j}) \quad (4.1)$$

Sous les contraintes :

$$\sum_{i=1}^I \sum_{j=1}^J Y_{i,j} = 1 \quad i \in I \quad (4.2)$$

$$\sum_{j=1}^J (X_j) = P \quad (4.3)$$

$$Y_{i,j} \leq X_j \quad \forall i \in I \quad \forall j \in J \quad (4.4)$$

$$Y_{i,j} \in \{0, 1\} \quad \forall i \in I, \forall j \in J \quad (4.5)$$

Où :

i : indice des points de demande.

j : indice des sites potentiels d'offre.

d_i : Demande totale du client i .

$c_{i,j}$: Cout de transport entre i et j .

P : nombre de sites à localiser.

- L'équation (4.1) représente la fonction objective.
- La contrainte (4.2) impose que chaque client doit être affecté à un seul site.
- La contrainte (4.3) exige que le nombre des sites ouverts doive être égal à P .
- La contrainte (4.4) impose que les clients affectés doivent être inférieure ou égal aux sites d'offre
- La contrainte (4.5) montre la nature binaire de la variable de décision Y .

II.4.5 Le problème de localisation des sites à deux niveaux :

Le problème de localisation des sites à deux niveaux généralement rencontrés est le « Fixed Charge Facility Location Problem ». Pour ce genre de problème, toutes les données sont connues de manière déterministe. L'objectif est donc de trouver le meilleur emplacement du site (**usine / centre de distribution**) et le mode de transport utilisé pour satisfaire les différentes zones de demande, tout en minimisant le coût fixe de localisation et de transport ainsi que répondre à toutes les exigences.

[BOUKLI-HACENE,2020]

Pour ce type de problèmes Il existe deux versions :

- **Le problème ou la capacité des sites est illimitée (UCFLP) :**

« Uncapacitated fixed charge facility location problem » [DASKIN, 2011].

- **Le problème ou les sites ont une capacité finie (CPLP) :**

« Capacitated plant location problem ». Cette problématique est à la base de nombreuses études liées à la conception de la chaîne d'approvisionnement. Plusieurs méthodes et algorithmes analytiques ont été proposés pour à propos de ce type de problèmes. Geoffrion et al. [GEOFFRION, 1974] ont montré qu'en utilisant des méthodes de séparation et d'évaluation, la méthode de relaxation lagrangienne est très efficace pour trouver la solution optimale. Dans le même article, les auteurs examinaient la fiabilité des activités de transport entre le fournisseur et le centre de distribution. Dans Daskin et Galvao et al [GALVAO, 1993], ces derniers utilisent des méthodes basées sur la relaxation lagrangienne pour résoudre le problème. Al-Sultan et al [AL- SULTAN etAL- FAWZAN, 1999] proposent une méthode de recherche Tabou pour résoudre le problème. La méthode a été testée avec succès mais sur des petits problèmes (des problèmes de petites tailles).

Pour la catégorie de problème (CPLP), des hypothèses restrictives indiquent que la capacité des usines et des centres de distribution quoi que ce soit reste limitée. La formulation du problème est la même que celle de la question (UCFPL), sauf qu'on doit ajouter des contraintes liées à la capacité du site. Dans la plupart des travaux, les chercheurs supposent que chaque client est servi par un et un seul centre de production (usine) ou de distribution. Klinecicz et al [KLINCEWICZ et LUSS. (1986)], Pirkul, [PIRKUL, 1987] et Sridharan [Sridharan, Ramaswami., 1995] donnent plus de détails sur les modèles et algorithmes utilisés. Efrayson et Ray [EFROYMSON et Ray, 1966] et Spielberg [SPIELBERG, 1969] ont proposé des méthodes heuristiques et des algorithmes précis basés sur la méthode Branch & Bound. Sridharan [SRIDHARAN, 1995] a proposé d'autres méthodes. Les problèmes et méthodes de décision :

La plupart des méthodes traitées dans cette étude visent à aider ceux qui sont concernés par la prise de décision que ce soit dans l'économie, management ... Logiquement plus que ces méthodes correspondent à des décisions pratiques ils auront plus de chance d'acceptance mais ce n'est pas la seule façon à juger une méthode. Il existe pas mal d'outils pour but d'aider les décideurs à la prise de décision.

Pour pouvoir bénéficier des caractéristiques de ces méthodes et les adapter aux besoins de la pratique nous avons essayé d'établir un schéma simple qui montre le passage de la situation d'un (decision maker) qui veut prendre une décision s'appuyant sur une analyse dépendante d'une situation quelconque. Pour un processus de prise de décision on indique seulement les aspects organisationnels (**liens de subordination, les niveaux de décision hiérarchiques, les interactions et les communications humaines...**). Nous avons supposé que le décideur (le decision maker) fasse partie du système qui produit et offre des biens ou des services. Un flux d'actions apparaît sous l'influence et la recommandation du décideur et d'autres acteurs, ces actions permettent d'obtenir un ensemble de résultats par exemple (des bénéfices, des

salaires...). Ces résultats seront évalués par le (decision maker) et les autres acteurs (des assistants par exemple). La figure (2.1) montre le processus de prise de décision

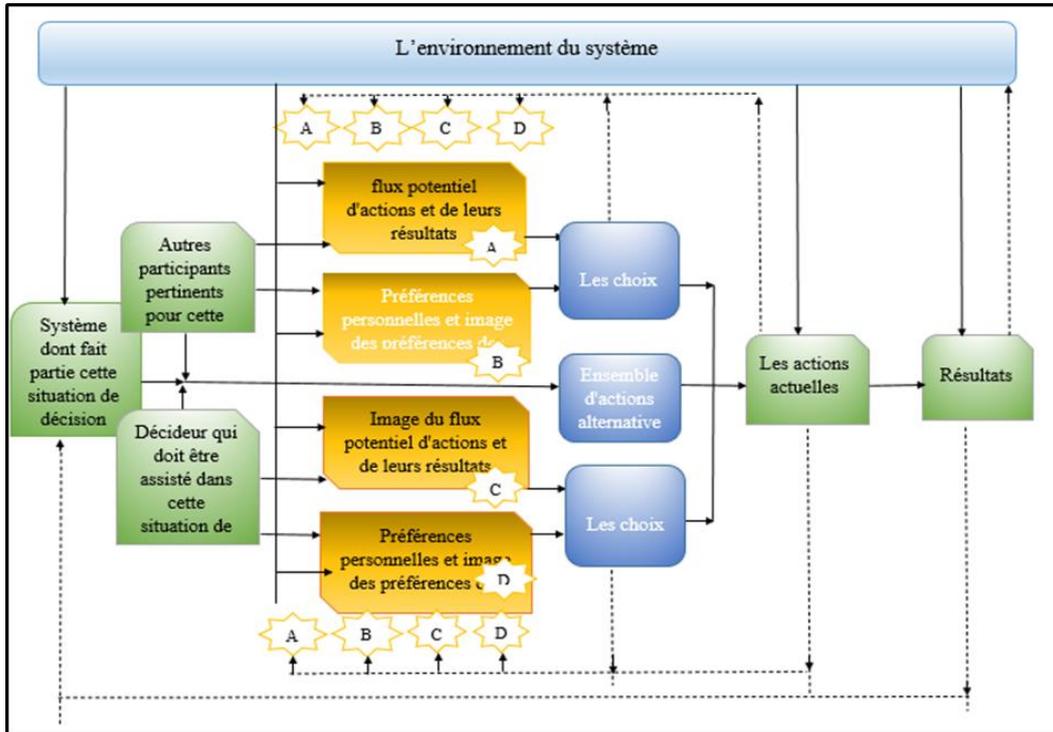


Figure 2. 1 : Situation typique de la décision

II.5 Les caractéristiques de base d'une prise de décision :

On peut classer les divers problèmes de décision selon un certain nombre des caractéristiques pour lesquelles il est possible d'identifier des méthodes de décision orientées spécialement pour ce type de problème. Pour cela il est nécessaire de définir et indiquer dans quelle(s) situation(s) il est légitime d'appliquer chaque méthode. Les méthodes de décision à critères multiples ne sont pas assez détaillées, Jusqu'à maintenant il n'existe pas un aperçu détaillé et accepté pour lequel les méthodes de décision à critères multiples peuvent être appliquées. Il est totalement clair que chaque situation de décision différente nécessite une méthode spécifique alors qu'il existe certaines méthodes peuvent être appliquées dans plusieurs situations. Des études ont été lancées dans ce sens par Despontin et Spronk [1979] et Scarola [1979]. D'après ces études On peut subdiviser les classes des caractéristiques en trois sous-ensembles comme il est montré dans le schéma suivant:

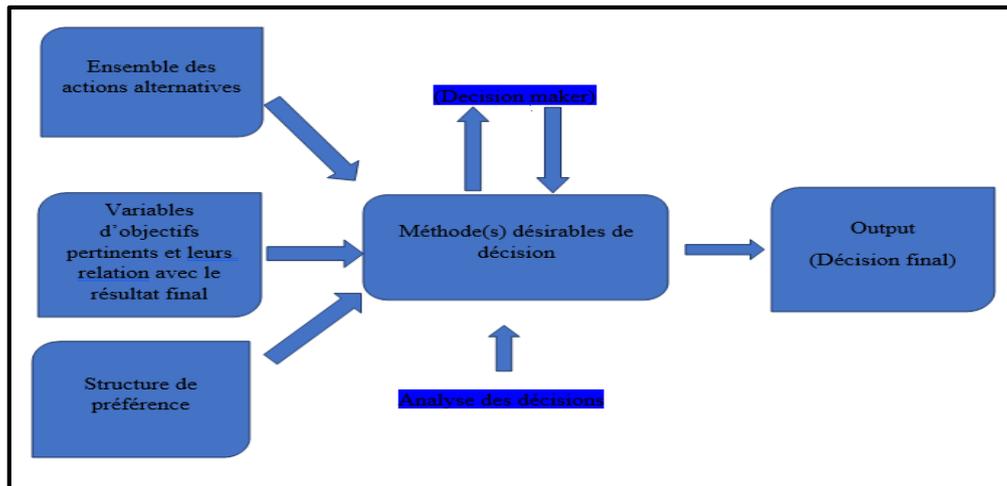


Figure 2. 2: Schéma montre les caractéristiques d'une prise de décision

II.5.1 Les méthodes interactives d'aide à la décision :

Les méthodes de décision interactives sont des méthodes dont lesquelles le décideur contribue à construire la solution non seulement par la définition du problème mais aussi par l'intervention dans la méthode de résolution. Il existe plusieurs méthodes interactives dans la littérature le truc qui les diffère par rapport aux autres méthodes c'est que le décideur contribue directement à la résolution du problème non seulement par la formulation du problème mais aussi par l'intervention dans la méthode. [VINCKE, 1976]

[AYADI, 2010]

II.5.2 Procédure de choix d'une méthode d'aide à la décision multicritère :

Les études des méthodes d'aide à la décision multicritère montrent que le point de différence entre ces méthodes est la façon de traduire et modéliser les préférences du décideur, c'est-à-dire les méthodes d'aide à la décision multicritère se caractérisent par les flux d'information utilisés à l'entrée (inputs) et le flux sortant des résultats (outputs) ce qui permet de choisir une bonne méthode d'aide à la décision multicritère.

[AYADI, 2010]

II.5.3 Goal programming (programmation par objectif) :

La programmation par objectif est une approche développée par

(CHARNES A. et al., 1961) considérée parmi les premières techniques de prise de décision à critères multiples. C'est une technique puissante, pratique et très adaptable, très utilisée dans les domaines de la budgétisation, la planification financière et l'industrie son utilisation dans la pratique consiste à résoudre un problème de prise de décision avec plusieurs objectifs simultanément, ce qui est impossible de les atteindre tous par la fois.

Goal programming (GP) donc consiste à trouver un compromis (une balance) entre ces objectifs afin de se rapprocher le maximum possible pour chacun d'eux, avec un facteur de pénalisation au cas de déviation pour chaque objectif

$d^+(i)$: déviation positive de l'objectif $g(i)$ (dépassement) supérieur ou égal à zéro.

$d^-(i)$: déviation négative de l'objectif $g(i)$ (diminution) inférieur ou égal à zéro.

Il faut faire attention pour les déviations $d^+(i)$ et $d^-(i)$, ils ne peuvent jamais être supérieurs à zéro les deux à la fois, c'est-à-dire qu'on ne peut jamais être en dessous (d^-) et en dessus (d^+) d'un objectif fixé $g(i)$ simultanément.

[AYADI, 2010]

L'utilité opérationnelle de la programmation par objectifs a été montrée dans la pratique comme il est indiqué dans l'exemple de l'analyse de rentabilité [jiri, 1965] et d'autres applications apportées dans la littérature.

[NIJKAMP et SPRONK, 1980]

II.5.4 Avantages et inconvénients du GP :

II.5.4.1 Avantages :

- Le principal avantage de la programmation par objectifs c'est qu'elle toujours fournit des résultats même si aucun objectif n'est réalisable à condition que la partie réalisable ne soit pas infinie, grâce à la personification des variables qui montrent si les objectifs sont atteints, et ensuite calculer l'écart entre l'objectif souhaité et l'objectif atteint.
- Un deuxième avantage de plus de la programmation par objectifs c'est qu'elle ne nécessite pas des démarches de résolution très compliquées, par contre la majorité des problèmes peuvent être résolus par des méthodes de programmation linéaire simple et facile à appliquer.

[NIJKAMP et SPRONK, 1980]

- Goal programming est très utile pour les problèmes pour lesquels on ne peut pas avoir des solutions optimales.
- Goal programming permet de se rapprocher le maximum possible aux objectifs voulus.
- Goal programming permet d'avoir une solution aux problèmes qui sont infaisables.

II.5.4.2 Inconvénients :

- La formule standard du goal programming peut générer des solutions non fiables dans le cas où les objectifs et les critères sont fixés de façon pessimiste.

II.6 Le Goal Programming interactive :

Cette approche est développée par [BENAYOUN et al., 1971] consiste à l'utilisation interactive du goal programming de telle sorte le décideur aura plus d'implication dans le processus de décision il aura la possibilité de modéliser ses préférences

Selon [ZELENY, 1982] il fallait prendre en considération l'évolution des préférences du décideur qui sont relatives à une situation de prise de décision et les renforcer par l'échange d'informations. Par contre [YANG et al., 1996] voient que la méthode goal programming interactive est basée sur un processus d'apprentissage.

La formulation standard du GP est la suivante :

$$\begin{aligned} & \mathbf{Min}_{\underline{x} \in A} \sum_{i=1}^p (\delta_i^+ + \delta_i^-) \\ \text{subject a: } & c_1(\underline{x}) \leq 0, l = 1, 2, \dots, L \\ & g_i(\underline{x})\delta_i^+ + \delta_i^- = b_i, i = 1, 2, \dots, n \\ & \delta_i^+, \delta_i^- \geq 0 \end{aligned}$$

La fonction objective sera toujours pour but de minimiser les déviations non favorables.

Avec :

δ_i^- : L'écart négatif de \underline{x} par rapport au but, (d^-) comme il est indiqué précédemment.

δ_i^+ : L'écart positif de \underline{x} par rapport au but, (d^+) comme il est indiqué précédemment.

$c_1(\underline{x})$: L'ensemble des contraintes relatives au système.

$g_i(\underline{x})$: L'évaluation de la solution par rapport au critère i .

Goal programming était développée par la suite du coup plusieurs types sont apparus tels que : Le GP pondérée, GP minimax, GP lexicographie, GP flou, GP avec intervalles.

[KETTANI et al, 2004]

II.6.1 Le GP pondéré :

Développé par (CHARNES A. et al.,1977), Le truc qui se diffère dans ce type par rapport au modèle standard c'est l'attribution des coefficients d'importance donc le GP pondéré sera présenté comme suivant :

$$\begin{aligned} & \mathbf{Min}_{\underline{x} \in A} \sum_{i=1}^p (w_i^+ \delta_i^+ + w_i^- \delta_i^-) \\ \text{subject a: } & f_i(x) + \delta_i^- - \delta_i^+ = g_i, i = 1, 2, \dots, p \\ & x \in X \\ & \delta_i^+, \delta_i^- \geq 0 \end{aligned}$$

w_i^+ Et w_i^- représentent les coefficients d'importance attribués aux déviations positives et négatives respectivement. L'attribution de ces coefficients sert à valoriser chaque objectif pour avoir un compromis qui montre l'ordre de préférence des objectifs voulu par le décideur.

[AYADI, 2010]

II.6.1.1 Le GP minimax :

La méthode est développée par [FALVELL et SMITH, 1976], cette variante consiste à minimiser la déviation la plus importante qui ne marche pas avec les aspirants du décideur, ce modèle est représenté comme suivant : [ROMERO, 1991]

$$\min_{\underline{x} \in A} \sum_{i=1}^p (w_i^+ \delta_i^+ + w_i^- \delta_i^-)$$

$$\text{sujet a: } f_i(x) + \delta_i^- - \delta_i^+ = g_i, i = 1, 2, \dots, p$$

$$w_i^- \delta_i^- + w_i^+ \delta_i^+ \leq Z, i = 1, 2, \dots, p$$

$$x \in X$$

$$\delta_i^+, \delta_i^- \geq 0$$

Ou Z : représente la déviation maximale.

Le goal programming minimax est classée parmi les approches a priori des préférences puisqu'elle utilise des coefficients de pondération $w(i)$ pour préciser les préférences du décideur. Généralement ce n'est pas possible d'avoir une solution idéale, au cas d'existence c'est très rare pour des cas bien précis et pour cela nous pouvons plus les généraliser. [OGRYCZAK, 2001] et [GANJAVI et al., 2002].

II.6.1.2 Le GP flou :

Ce modèle a été développé par [ZIMMERMANN, 1978] permet de modéliser des problèmes pour lesquels le décideur ne peut attribuer que des valeurs approximatives pour les objectifs voulus [SELIM et al., 2008].

Cette variante est modélisée par le modèle mathématique suivant :

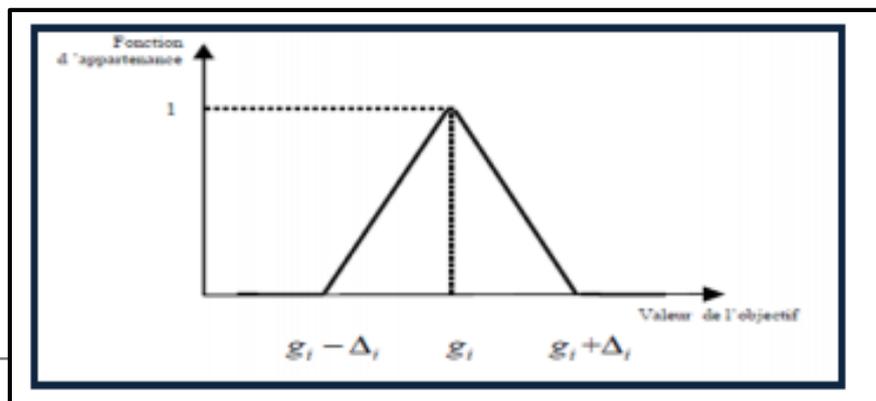
$$\text{Max } \alpha$$

$$\text{sujet a: } \frac{f_i(x)}{\Delta_i} + \delta_i^- - \delta_i^+ = \frac{g_i}{\Delta_i}, i = 1, 2, \dots, p$$

$$\alpha + \delta_i^- + \delta_i^+ \leq 1, i = 1, 2, \dots, p$$

$$\delta_i^+, \delta_i^- \geq 0, x \in X, i = 1, 2, \dots, 4$$

$\Delta(i)$ c'est la de par rapport $g(i)$ fixés décideur.



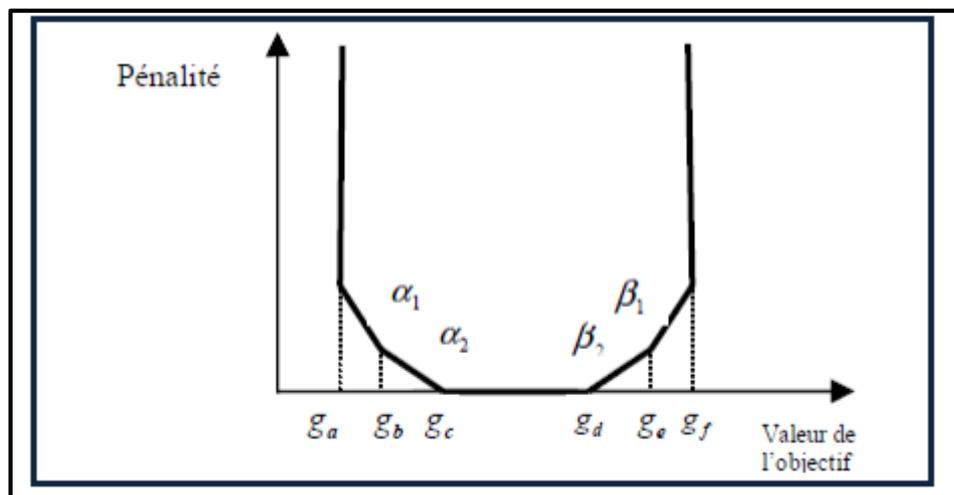
constante déviation aux buts par le

Figure 2. 3: La fonction d'appartenance Δ_i (forme triangulaire)[AYADI, 2010]

II.6.1.3 Le GP avec intervalle :

Ce modèle développé par (CHARNES A. et al., 1976) repose essentiellement sur l'emplacement des objectifs dans des intervalles on peut les appeler des intervalles de tolérance de tels sorte le décideur accepte et sera satisfait le fait d'avoir une valeur appartenant à cet intervalle, avec une pénalité symbolique attribuée pour chaque intervalle selon sa proximité de la cible qui représente l'objectif voulu par le décideur.

La figure suivante illustre la fonction de pénalité à cinq segment, L'intervalle central, représenté par $[g_c, g_d]$, est l'intervalle «cible», où aucune pénalité n'est appliquée, le décideur sera satisfait des résultats appartenant à cet intervalle.



[AYADI, 2010]

Figure 2. 4 : Fonction de pénalité à 5 segments [AYADI, 2010]

Le modèle mathématique du GP avec intervalle est représenté comme suit :

$$\min_{x \in A} \alpha_1 \delta_{11}^- + \alpha_2 \delta_{12}^- + \alpha_3 \delta_{13}^- + \alpha_4 \delta_{14}^-$$

$$\text{sujet } a: f_1(x) \leq g_f$$

$$f_1(x) + \delta_{14}^- - \delta_{14}^+ = g_e$$

$$f_1(x) + \delta_{13}^- - \delta_{13}^+ - \delta_{14}^+ = g_d$$

$$f_1(x) + \delta_{11}^- + \delta_{12}^- - \delta_{12}^+ = g_c$$

$$f_1(x) + \delta_{11}^- - \delta_{11}^+ = g_b$$

$$f_1(x) \geq g_a$$

$$\delta_{11}^+, \delta_{1j}^- \geq 0, x \in X, j = 1, 2, \dots, 4$$

On peut également classer le modèle GP avec intervalle avec des fonctions de pénalité dans la catégorie des méthodes avec intégration a priori de l'information fournie par le décideur, tant que ce dernier est sensé de fixer ses objectifs, les informations relatives aux décisions de décideur doivent être prises en compte par les diverses inclinaisons qui définissent les pénalités.

[AYADI, 2010]

II.6.1.4 Comparaison entre le modèle linéaire et le modèle GP :

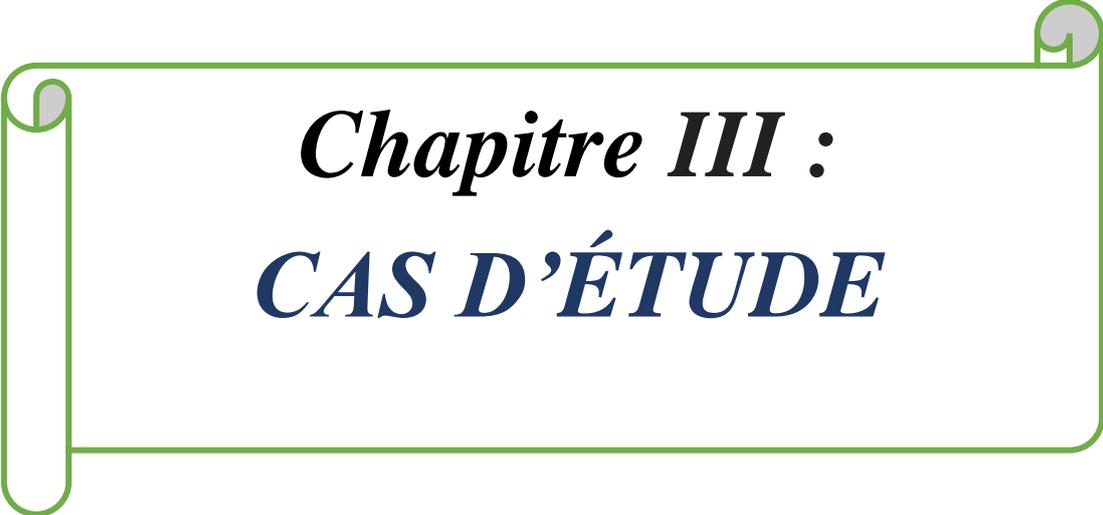
Tableau 2. 1: Comparaison entre le modèle linéaire et le modèle GP [1]

Modèle linéaire	Modèle GP
✚ Un seul objectif	✚ Multi objectifs
✚ Maximiser (profits, ventes, revenus...)	✚ Minimiser les déviations des objectifs voulus.
✚ Minimiser (les dépenses, temps de	

séjour...)	
<ul style="list-style-type: none"> ✚ La fonction objective peut être maximiser ou minimiser 	<ul style="list-style-type: none"> ✚ La fonction objective est toujours une minimisation.
<ul style="list-style-type: none"> ✚ Utilisé pour des systèmes relativement simples 	<ul style="list-style-type: none"> ✚ Utilisé pour des systèmes relativement compliqués (plus réalistes)
<ul style="list-style-type: none"> ✚ Déterminer et écrire la Fonction objective puis les contraintes. 	<ul style="list-style-type: none"> ✚ Déterminer les contraintes puis la Fonction objective.
<ul style="list-style-type: none"> ✚ Avec un modèle LP on obtient la meilleure solution possible (solution idéale) pour chaque problème. 	<ul style="list-style-type: none"> ✚ À L'aide du Goal programming on obtient une solution satisfaisante et acceptable qui rapproche de la solution idéale à cause des conflits entre les objectifs.
<ul style="list-style-type: none"> ✚ Nous voulons trouver les valeurs des variables de décision qui maximisent ou minimisent la FO. ✚ Les signes des contraintes peuvent être « <= », « >= » ou « = ». 	<ul style="list-style-type: none"> ✚ Nous visons à trouver les valeurs des variables de décision pour lesquelles nous obtenons des résultats plus proches à nos objectifs. ✚ Les signes des contraintes toujours « = »

II.7 Conclusion :

Dans ce chapitre nous avons présenté les diverses méthodes d'analyses, nous avons aussi présenté les problème célèbres dans les chaines logistiques tels que le problème p-médian problème de recouvrement, problème de recouvrement maximale, problème p-centre, ensuite nous avons parlé des méthodes d'aide à la décision multicritère ces avantages pour la résolution des problèmes logistiques. Parmi les méthodes d'aide multicritère nous avons présenté la méthode goal programming, ses différents types, ses avantages ainsi que ses inconvénients et enfin nous avons conclu par un tableau qui résume la différence entre un modèle goal programming et un modèle linéaire.



Chapitre III :
CAS D'ÉTUDE

III.1 Introduction :

La gestion d'une chaîne logistique nécessite l'intégration des systèmes d'information géométriques afin de permettre localiser les points de vente, les réseaux de distribution, les partenaires et la dispersion de la population selon les zones.

Ce chapitre est consacré au cas d'étude, nous allons dans un premier temps parler des plateformes logistiques, puis nous parlerons des systèmes d'information géométriques.

Ensuite nous allons entamer notre cas d'étude concernant la distribution alimentaire sur la ville de Tlemcen de telle sorte nous allons concevoir un réseau de distribution du lait pasteurisé sur des zones selon la répartition de la population pour but de gérer les crises de la non disponibilité du lait pasteurisé. Nous allons donc choisir une carte géographique comporte 3 communes de la ville de Tlemcen (Tlemcen-centre-ville, Chetouane et Mansourah) à l'aide de logiciel QGIS, ensuite nous allons installer des points de vente sur la carte pour fournir du lait nous allons faire un maillage pour cette carte puis estimer la population dans chaque maillon par la suite nous allons estimer les demandes selon la population dans chaque maillon et nous allons appliquer les modèles p-médian et max-covering pour concevoir notre réseau de distribution.

III.2 Systèmes d'informations géographiques (SIG) :

III.2.1 Définition et présentation

Un système d'information géographique est un système d'information capable de fournir des informations, organiser et présenter des données alphanumériques spatialement référencées à l'aide des satellites, ainsi qu'il nous permet de produire des cartes et des plans géographiques adaptés selon nos besoins.

Le terme « système d'informations géographiques » se rapporte aux outils logiciels, mais l'utilisation de ces logiciels englobe des données, de savoir-faire et beaucoup plus un matériel performant.

Les SIG fournissent plusieurs types d'informations liées à la géographie et l'urbanisation tels que les statistiques, les requêtes... Pour cela les SIG sont utilisés dans tous les domaines collectivités territoriales, secteur public, entreprise, écoles, administrations, et surtout dans le domaine de la recherche et développement puisque n'importe quelle étude est liée directement à la géographie, la croissance démographique, la dispersion de la population. Les SIG sont capables de fournir des informations sur tout ça et en termes d'actualité et rapidité.

De nos jours les SIG occupent un très vaste marché dans le monde entier et offrent des emplois un peu partout dans le monde. [EFROYMSONet Ray, 1966] [Esri France]

III.2.2 Les composants d'un SIG :

Un système d'information géographique est composé de cinq composantes principales qui sont :

a) Matériel :

Les SIG sont disponibles sur toutes les modalités des ordinateurs

a) Les logiciels :

Les logiciels SIG permettent son utilisateur à analyser afficher et stocker des informations relatives à une zone géographique quelconque.

Un logiciel SIG se compose principalement de :

- Un système de gestion de base de données.
- Interface graphique pour l'utilisateur.
- Outils pour la manipulation et saisir les informations géographiques.
- Outils de visualisation, analyse et saisir la requête voulue.

b) Les données :

La partie essentielle d'un SIG sont les données géographiques qui peuvent être mises par défaut de la part du producteur ou bien insérées par l'utilisateur.

c) Les utilisateurs :

Les SIG sont devenus indispensables. Ils sont utilisés dans tous les domaines et tous les secteurs reliés à la géographie en raison de leurs flexibilité et facilité d'utilisation.

d) Méthodes :

L'utilisation d'un SIG exige à l'utilisateur de respecter les réglementations de l'organisation qui est responsables des droits d'auteur.

III.2.3 Types des données d'un SIG :

On distingue quatre composantes principales pour une donnée géographique sont :

- Les données géométriques pour la localisation.
- Les données descriptives qui sont attribuées pour chaque phénomène et/ou chaque objet.
- Les données de styles trouvées à la barre des tâches et paramètres d'affichage.
- Les données associées (date, Nom, méthode...).

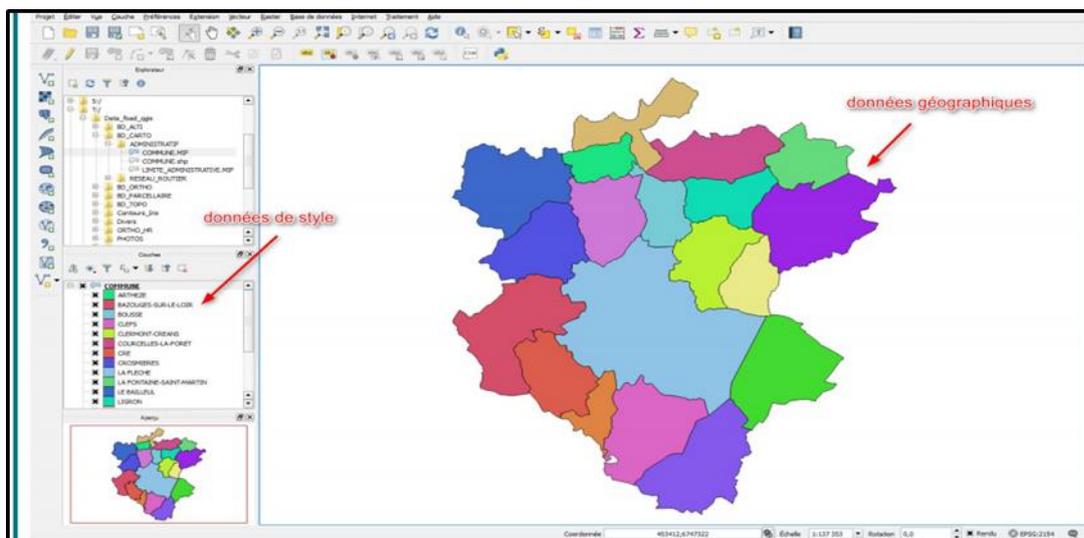


Figure 3. 1: Données géographiques et styles [QGIS]

Table attributaire - COMMUNE : Total des entités: 19, filtrées: 19, sélectionnées: 0 [1,7] [2,7] [3,7]

ID_BDCARTO	NOM_COMM	INSEE_COMM	STATUT	X_COMMUNE	Y_COMMUNE	SUPERFICIE	POPULATION	INSEE_CANT	INSEE_ARR	NOM_DEPT
720000282	SAINT-JEAN-DE-L...	72291	Commune simple	478935	6744018	3203	900	26	1	SARTHE
720000009	ARTHEZE	72009	Commune simple	466877	6748256	865	400	17	1	SARTHE
490000363	VAULANDRY	49380	Commune simple	472055	6726373	2765	300	04	3	MAINE-ET-LOIRE
490000100	CLEFS	49101	Commune simple	470066	6730106	2592	900	04	3	MAINE-ET-LOIRE
720000180	MAREIL-SUR-LOIR	72185	Commune simple	475371	6739051	1183	600	14	1	SARTHE
720000042	BOUSSE	72044	Commune simple	470515	6745247	1202	400	17	1	SARTHE
720000021	LE BAILLEUL	72022	Commune simple	462145	6746131	2746	1200	17	1	SARTHE
720000081	CLERMONT-CRE...	72084	Commune simple	473148	6741278	1782	1200	14	1	SARTHE
720000174	MALICORNE-SUR...	72179	Chef-lieu de cant...	469673	6750652	1513	2000	17	1	SARTHE
720000348	THOREE-LES-PINS	72357	Commune simple	477876	6733984	2818	700	16	1	SARTHE
720000131	LA FONTAINE-SA...	72135	Commune simple	479050	6747256	1372	600	26	1	SARTHE
720000149	LA FLECHE	72154	Sous-préfecture	470872	6737445	7421	15400	14	1	SARTHE
720000366	VILLAINES-SOUS...	72377	Commune simple	467557	6744178	1916	1000	17	1	SARTHE
720000104	CRE	72108	Commune simple	464444	6733839	1719	800	14	1	SARTHE

Montrer toutes les entités

Figure 3. 2: Données descriptives (attributaires) [QHIS]

Les données géographiques d'un SIG sont réparties en deux types différents sont :

- **Des données vectorielles :**

Sont généralement représentées par des points, des lignes, des polygones ou des polygones à trous.

La figure suivante montre les différentes formes des données vectorielles :

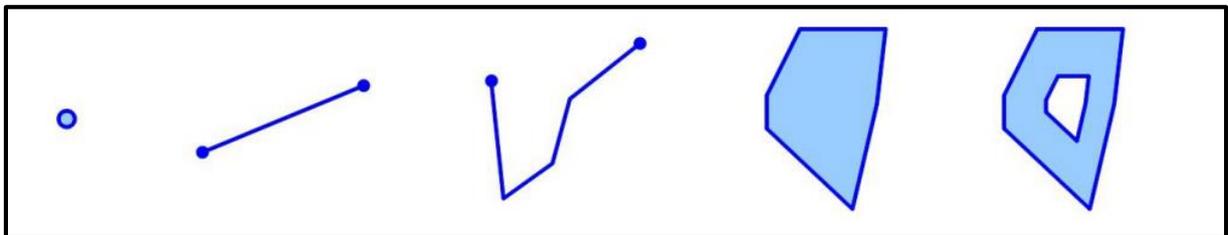


Figure 3. 3: Les formes des données vectorielles [QGIS]

- **Des données Raster :**

Les données Raster sont particulièrement des images composées des points colorés qui s'appellent des « pixels » la figure (4.4) montre une donnée Raster (pixel) :



Figure 3. 4: Type de données pixels [V]



Modèle numérique Ortho photo Scan 25

Figure 3. 5: Exemple des Types de Raster [QGIS]

III.2.4 Fonctionnalité d'un SIG :

- Affichage ou création des cartes.
- Analyser des données pour but d'avoir des nouvelles informations.
- L'obtention des données.
- Représentation des données et la possibilité de les archiver.

[QGIS]

Les données dans les SIG peuvent être représentées sous forme de couches superposées l'une sur l'autre.



Figure 3. 6: Des couches de données en superposition [QGIS]

QGIS est un logiciel « open source » développé par la CCIG (commission de coordination de l'information géographique) en 2011. QGIS est l'un parmi les SIG les plus performants et les plus utilisés dans le monde, il utilise le SCR (système de coordonnées de référence) pour positionner et localiser les objets les uns par rapport aux autres.

[QGIS]

III.3 Les plateformes logistiques :

Les plateformes logistiques jouent un rôle très important dans la gestion de la chaîne d'approvisionnement

III.3.1 Définition des plateformes :

Une plateforme logistique c'est un endroit dont lequel on reçoit des marchandises afin de les réexpédier sur une période de temps très courte.

Selon MOCELLIN[MOCELLIN, 2003] « une plateforme logistique c'est un endroit où nous pouvons recevoir, stocker des marchandises et les réexpédier dans un délai très court. Par opposition aux entrepôts qui peuvent stocker la marchandise pour une longue période pour but de regrouper, fractionner ou conditionner la marchandise. Les entrepôts globalement sont des zones de stockage des marchandises leurs gestions est soumis principalement à la fluctuation des prix. »

[MOCELLIN, 2003]

Il est rarement possible d'utiliser les plateformes pour le stockage et conditionnement des marchandises, par ce que l'objectif principal de localiser et installer ces plateformes c'est d'accélérer le processus d'expédition des marchandises vers une autre direction probablement cette direction c'est un client.

Les plateformes logistiques sont considérées comme un moyen d'optimisation des chaînes d'approvisionnement.

Les plateformes logistiques peuvent aussi être des points de vente orienté du centre de production vers le consommateur directement sans faire passer par la partie intermédiaire (fournisseurs, grossistes, détaillants, ...) et par conséquent le client bénéficie du produit qu'il le plait avec un moindre cout.

III.3.2 Objectif d'établissement des plateformes logistiques :

L'étude d'installation des plateformes logistiques a pour but stratégique et économique à la fois, le fait que les plateformes servent à gérer les situations de crise de la non disponibilité des produits nécessaires pour les citoyens d'une part c'est une bonne stratégie pour satisfaire les besoins de la clientèle, d'autre part elle contribue fortement au développement du secteur économique dans le pays.

L'implantation des plateformes logistiques du producteur (l'usine) directement vers le consommateur permet aussi d'éviter les coulisses de la marché noire et l'accaparement des marchandises par les monopoles, et assure la réduction des prix de vente par l'élimination de la partie intermédiaire entre l'usine et le consommateur.

L'objectif principal du concept des PFL c'est de rendre disponibles les produits quand les consommateurs ont besoin.

Une PFL peut jouer le rôle d'une zone de stockage, entreposage, ou bien un point de vente.

[GUYON et al., 2010]



Figure 3. 7: Représentation d'une PFL [W]

III.4 Les caractéristiques d'une plateforme :

Une PFL doit avoir un ensemble des caractéristiques tels que : [Catram Consultants, 2010]

- **L'accessibilité** : c'est-à-dire que l'emplacement de la plateforme doit être dans un endroit accessible, sécurisé et proche du client.
- **Tailles et services disponibles** :
La PFL peut prendre plusieurs tailles selon l'opportunité et le projet et aussi selon le service (stockage, entreposage, point de vente...).
- **L'intégration urbaine** :
Une PFL représente un équipement à grande échelle c'est pour ça qu'il est nécessaire de l'intégrer dans l'environnement urbain. Par exemple la circulation aux prés des grandes surfaces et les centres commerciaux, il faut donc les intégrer dans les plans urbains de circulation.

III.5 Définition de la problématique :

Grâce à la richesse de sa composition et la variété de ses constituants, le lait donne naissance par transformation à une très vaste famille de produits. Parmi tous les aliments, le lait est le plus complet. Il contient principalement : (87%) de l'eau, des lipides, des protéines (caséine, lactalbumine), des vitamines A, B, D, des glucides (lactose), des éléments minéraux (calcium 125mg/L, sodium, potassium...). Pour cela le lait est une nécessité quotidienne indispensable pour le citoyen. Les dernières années nous avons constaté une grande incapacité de fournir du lait pasteurisé sur tout le territoire algérien presque en raison de la perturbation dans la production, la disponibilité et la distribution du produit. Pour ce faire nous avons essayé de faire une étude en collaboration avec l'entreprise « Giplait » Tlemcen concernant la distribution du lait pasteurisé sur la ville de Tlemcen, afin de localiser les failles du réseau de distribution les problèmes qui s'imposent pour pouvoir par la suite proposer une conception d'un réseau de distribution accordé par des plateformes destiné à la distribution directe du lait de l'usine vers le client ultime directement sans passer par la partie intermédiaire (les fournisseurs) pour assurer la disponibilité du produit et satisfaire les besoins du client.

III.5.1 Présentation de l'entreprise Giplait :

Créée en 1969 par l'office national du lait à cette époque y avait trois usines nationales qui sont :

Coletal à Alger, Clo à Oran et Clc à Constantine. En 1982 la restructuration de l'office national en trois offices régionaux :

- Orlac (centre)
- Orolait (ouest)
- Orelait (est)

En Janvier 1998 les deux offices de l'est et l'ouest ont été absorbé par l'office du centre, puis dans le premier trimestre de 1998 l'apparition du Giplait

Les principales activités de l'entreprise sont : a collecte du lait cru, traitement, conditionnement, distribution et commercialisation du lait pasteurisé et d'autres dérivés de laitiers.

La filiale de Giplait « Mansourah » est hiérarchisé comme suivant :

- La direction générale.
- La sous-direction finance et comptabilité.
- La sous-direction administration générale.
- La sous-direction approvisionnement.
- La sous-direction de maintenance.
- La sous-direction agro élevage.
- La sous-direction de production.
- La sous-direction laboratoire.
- La sous-direction contrôle de gestion.
- La sous-direction sécurité.

III.5.2 Processus de production du lait de vache pasteurisé :

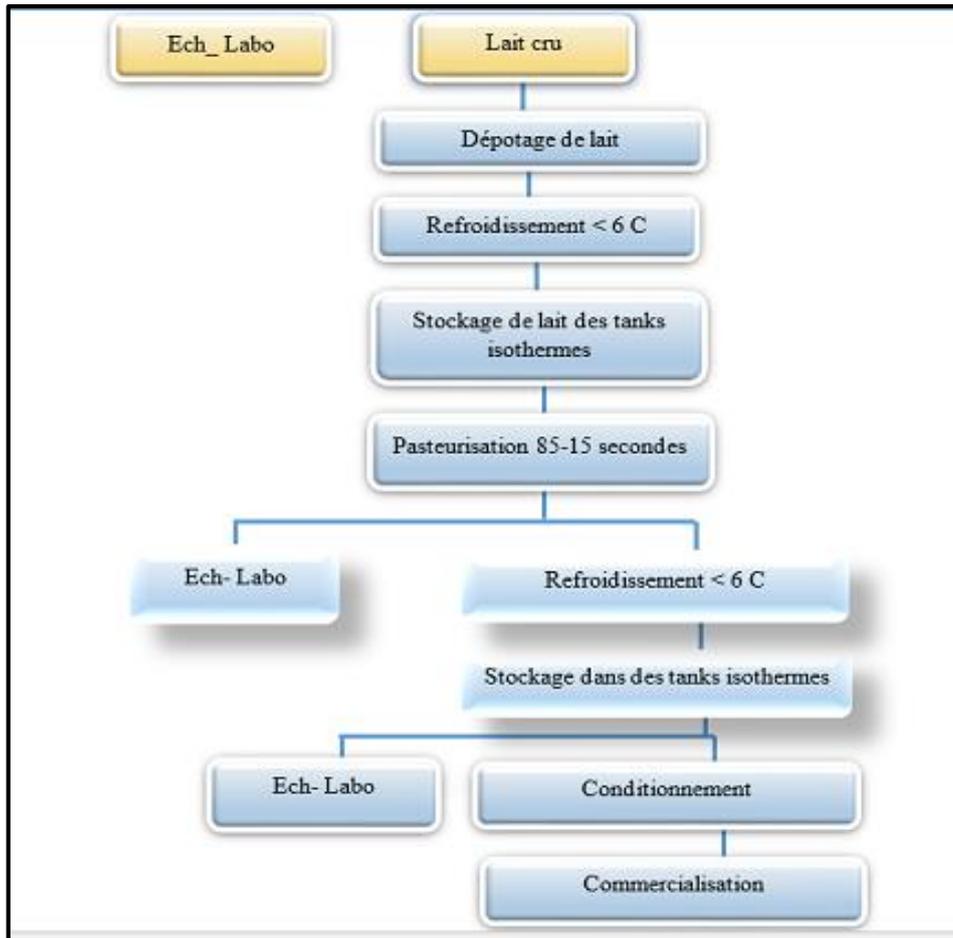


Figure 3. 8: Processus de fabrication du lait pasteurisé

III.5.3 Cas d'étude :

Nous avons pris dans nos études les trois communes de **Tlemcen** centre-ville, **Mansourah**, une partie de **Chetouane et (Safsaf)** dans une carte géographique en utilisant le logiciel QGIS nous avons fait un maillage de 1 km carré nous avons obtenu par la suite 68 maillons(zones). Ensuite nous avons essayé d'estimer la densité de population dans chaque commune. À partir de cette estimation nous avons fait une estimation de distribution de la population sur les maillons obtenus dans la carte. Bien évidemment cette distribution régie par la conjugaison des facteurs physiques, climatiques et socio-économique, nous avons fixé le nombre moyen des membres dans chaque famille est cinq personnes, puis dans chaque zone nous avons estimé un nombre de bâtiments, des villas et appartements, chaque bâtiment comporte 10 appartements de cinq personnes ce qui nous donne 50 personnes par bâtiment multiplié fois le nombre des bâtiments dans chaque maillon de même manière pour les villas et les appartements cinq personnes multiplié fois le nombre des appartements et villas dans chaque maillon et ensuite on calcule la population totale de chaque maillon. On trouve donc des zones dont la densité est très élevée. D'autre part on trouve des zones moins denses ou presque vides tels que les zones industrielles, ces zones sont à éliminer dans notre cas d'étude.

Après avoir analysé la distribution de la population nous avons éliminé 47 parmi les 63 zones donc il nous reste 16 zones.

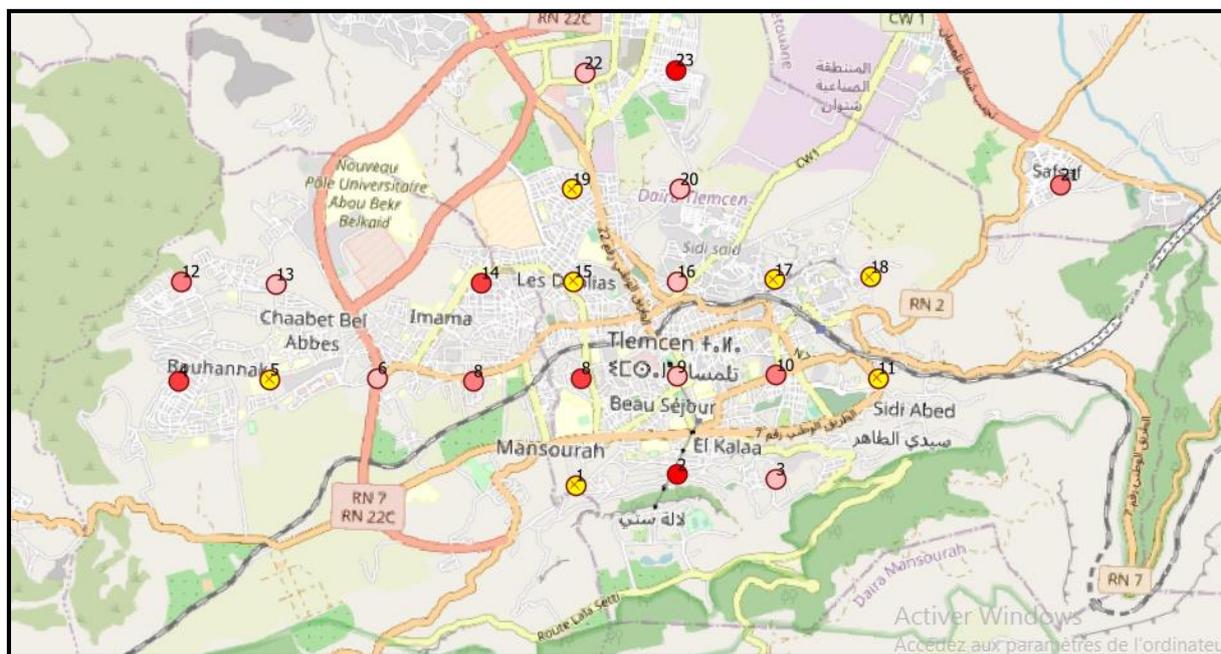


Figure 3. 9: Représentation des zones avant nettoyage sur QGIS

L'office national interprofessionnel du lait détermine les quantités à livrer par jour dans chaque région et chaque commune selon la densité de population qui existe là-bas. Pour la grande ville de Tlemcen les quantités par commune sont comme suivant :

Tableau 3. 1:État journalier de la distribution du lait "LPC" sur Tlemcen

Dairas	Communes	Quantité programmée par l'ONIL (Litres) par jour
TLEMCEN	TLEMCEN	18678
REMCHI	REMCHI	2418
SABRA	SABRA	1382
SEBDOU	SEBDOU	6144
MAGHNIA	MAGHNIA	2304
ZENATA	ZENATA	922
MANSOURAH	MANSOURAH	307
CHETOUANE	CHETOUANE	384
HENNAYA	HENNAYA	2534
AMIEUR	AMIEUR	768
BENI SENOUS	BENI SENOUS	154
GHAZAOUET	GHAZAOUET	922
NEDROMA	NEDROMA	691
AIN FEZZA	AIN FEZZA	1152
AIN YOUSSEF	AIN YOUSSEF	0
HONAINE	HONAINE	2534

BENI-OUARSSOUS	BENI-OUARSSOUS	1435
OULED MIMOUN	OULED MIMOUN	6374
TOTAL	TLEMCCEN	49103

La capacité journalière de production du lait pasteurisé pour l'entreprise « Giplait » est environ 85000 litres par jour distribué sur les trois villes de Tlemcen, Ain Temouchent et Oran.

La ville de Tlemcen s'occupe de la plus grande part par rapport à les deux autres villes.

III.6 Problématique 01 modèle (P-médian) :

La quantité à livrer dans les zones prises dans notre étude est égal à 18678 pour le centre-ville de Tlemcen, 307 litres pour Mansourah la commune et 384 litres pour Chetouane ce qui nous fait un total de 19369 litres par jour. Le but de notre étude est d'implanter des points de vente du lait pasteurisé qui passe directement de l'usine vers le consommateur sans avoir passer par la partie intermédiaire donc la communauté ciblée dans cette étude est le client ultime de la chaîne (le consommateur), pour ce faire nous n'allons pas inclure les cafétérias et les détaillants on s'intéresse uniquement par le consommateur.

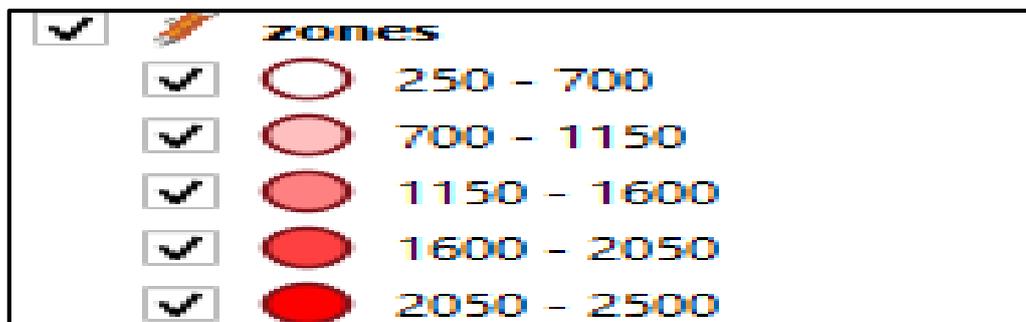


Figure 3. 10: Carte clé QGIS

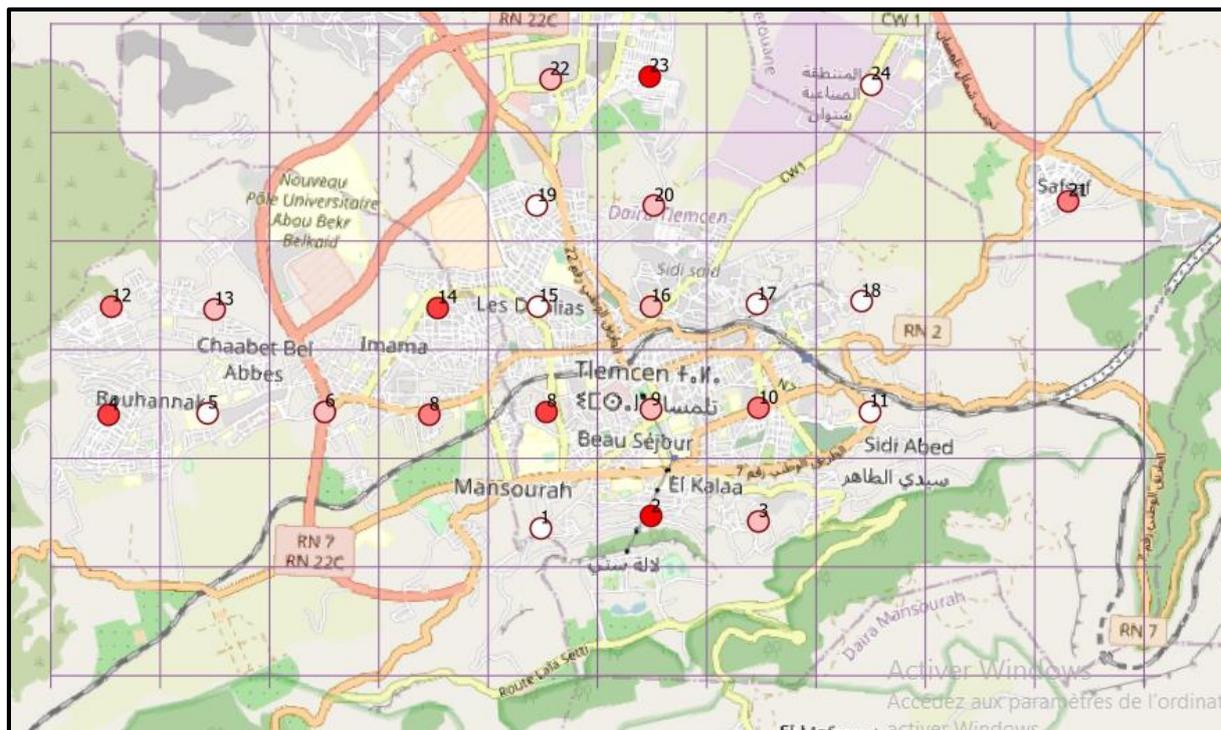


Figure 3. 11: Représentation des zones dans le logiciel QGIS

Ensuite nous avons calculé la distance entre chaque point de vente et le centre de gravité de chaque zone parmi les 16 zones restantes, les résultats sont indiqués dans le tableau suivant :

Tableau 3. 2: Les distances entre les zones et les points de vente

Distance (Km)	Zone 01	Zone02	Zone03	Zone04	Zone05	Zone06	Zone07	Zone08
Point de vente 01	4.26	5.30	0.80	1.21	2.14	3.20	4.15	5.20
Point de vente 02	2.63	3.51	3.01	1.16	0.68	1.28	2.14	3.13
Point de vente 03	1.94	1.70	5.94	3.96	3.04	2.02	1.14	0.63
Point de vente 04	4.87	4.13	8.88	7.00	6.07	5.01	4.23	3.38
Point de vente 05	3.77	3.29	4.85	3.32	2.87	2.61	2.76	3.20
Distance (Km)	Zone 09	Zone 10	Zone 11	Zone 12	Zone 13	Zone 14	Zone 15	Zone 16
Point de vente 01	1.20	0.94	2.37	4.29	4.60	8.31	4.36	5.10
Point de vente 02	2.93	1.96	2.94	2.08	2.42	6.10	2.60	3.90
Point de vente 03	5.91	4.93	2.95	1.00	1.64	3.76	3.01	2.60

Point de vente 04	8.70	7.71	5.76	3.86	3.65	2.70	4.67	3.82
Point de vente 05	4.37	3.48	1.62	0.64	1.09	4.83	4.27	0.95

Pour ce faire comme une première problématique nous avons supposé que chaque famille consomme au moyen **deux litres** par jour selon des statistiques en 2017. [Horizons, 2019].

En se basant de ces statistiques nous avons supposé que chaque famille consomme deux litres du lait par jour multiplié fois le nombre totale des familles estimé sur les zones choisies nous avons trouvé une consommation journalière égale à 10100 litres par jour ce qui est inférieure à la quantité produite chaque jour donc on peut satisfaire la demande journalière, il faut donc rapprocher les points de vente du client de tel sorte chaque zone sera affectée au point de vente le plus proche pour ce faire nous avons établi un modèle p-médian consistant à affecter chaque zone au point de vente le plus proche.

Nous avons pris cinq points de vente candidats qui sont installés en réalité.

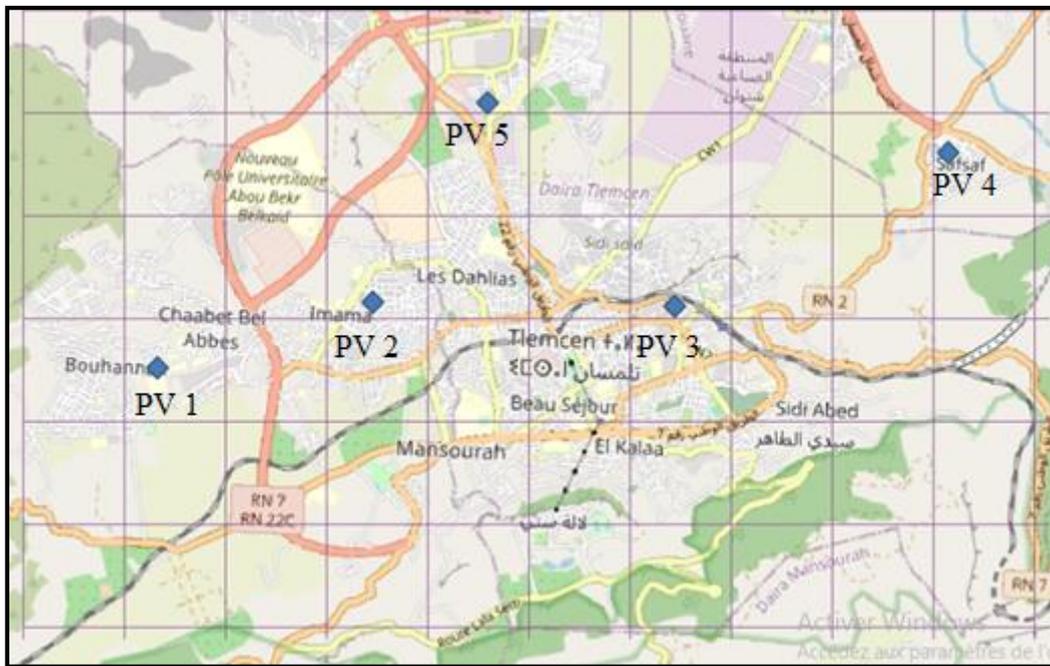


Figure 3. 12: Représentation des points de vente dans le logiciel QGIS

Le modèle mathématique du problème P-médian est comme suivant :

$$\text{Min } Z = \sum_{i=1}^I \sum_{j=1}^J (d_i * c_{ij} * Y_{ij}) \quad (3.1.1)$$

Sous les contraintes :

$$\sum_{i=1}^I \sum_{j=1}^J (Y_{ij}) = 1 \quad \forall i \in I \quad (3.1.2)$$

$$\sum_{j=1}^J (X_j) = P \quad (3.1.3)$$

$$Y_{ij} \leq X_j \quad \forall i \in I \quad \forall j \in J \quad (3.1.4)$$

$$Y_{ij}, X_j \in \{0, 1\} \quad \forall i \in I \quad \forall j \in J \quad (3.1.5)$$

Où : i : indice des points de demande.

j : indice des sites potentiels d'offre.

d_i : demande totale du client i .

c_{ij} : coût de transport entre i et j .

P : nombre de sites à localiser.

Variable de décisions :

$$X_j = \begin{cases} 1 & \text{si on doit localiser l'usine } j; \\ 0 & \text{si non.} \end{cases}$$

$$Y_{ij} = \begin{cases} 1 & \text{si le client } i \text{ sera affecté à l'usine } j; \\ 0 & \text{sinon.} \end{cases}$$

- L'équation (3.1.1) représente la fonction-objective.
- La contrainte (3.1.2) impose que chaque client soit affecté à un seul site.
- La contrainte (3.1.3) détermine le nombre de sites.
- La contrainte (3.1.4) indique que le nombre des sites affectés doit être inférieur ou égal aux sites ouverts.
- La contrainte (3.1.5) montre la nature binaire des variables de décision.

En supposant dans un premier temps que les demandes sont identiques, les capacités des points de vente sont infinies. Le but est de minimiser la distance parcourue multiplié par la demande et le cout de de déplacement qui est une unité monétaire par kilomètre pour chaque client vers le point de vente.

Le modèle Lingo référent à notre problématique contient cinq sites candidats et 16 zones est représentées par la figure suivante :

Modèle Lingo P-médian

```
sets :
pointv/1..5/:y;
zone/1..16/:dem;
pointzone(pointv, zone):dist,x;
endsets
data:
```

```

costKm = 1;
dem= 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10 10;

dist=
4. 5. 0. 1. 2. 3. 4. 5. 1. 0. 2. 4. 4. 8. 4. 5.1
26 30 80 21 14 20 15 20 20 94 37 29 60 31 36 0
2. 3. 3. 1. 0. 1. 2. 3. 2. 1. 2. 2. 2. 6. 2. 3.9
63 51 01 16 68 28 14 13 93 96 94 08 42 10 60
1. 1. 5. 3. 3. 2. 1. 0. 5. 4. 2. 1. 1. 3. 3. 2.6
94 70 94 96 04 02 14 63 91 93 95 00 64 76 01 0
4. 4. 8. 7. 6. 5. 4. 3. 8. 7. 5. 3. 3. 2. 4. 3.8
87 13 88 00 07 01 23 38 70 71 76 86 65 70 67 2
3. 3. 4. 3. 2. 2. 3. 4. 3. 1. 0. 1. 4. 4. 0.9
77 29 85 32 87 61 76 20 37 48 62 64 09 83 27 5;

enddata
min=@sum(pointv(i):@sum(zone(j):dem(j)*dist(i,j)*costKm*x(i,j)));
!(1.1); @sum(pointv(i):y(i))=5;
!(1.2); @for(zone(j):@sum(pointv(i):x(i,j))=1);
!(1.3); @for(pointv(i):@for(zone(j):x(i,j)<=y(i)));
!(1.4); @for(pointv(i):@sum(zone(j):x(i,j))>=y(i));
@for(pointv:@bin(y));
@for(pointv(i):@for(zone(j):@bin(x(i,j))));

```

(1.1) :Nombre de point de vente à ouvrir.

(1.2) :Une zone doit être affectée à un seul point de vente.

(1.3) :Un client doit être servi par un point de vente ouvert.

(1.4) :Un point de vente s'il est ouvert, il doit servir au moins une zone.

La solution optimale obtenue pour ce modèle est la suivante :

Solver Status		Variables		Variable	Value	Reduced Cost
Model Class:	PIIP	Total:	85	COSTKM	1.000000	0.000000
State:	Global Opt	Nonlinear:	0	Y(1)	1.000000	0.000000
Objective:	210.7	Integers:	85	Y(2)	1.000000	0.000000
Infeasibility:	0	Constraints		Y(3)	1.000000	0.000000
Iterations:	0	Total:	103	Y(4)	1.000000	0.000000
Extended Solver Status		Nonlinear:	0	Y(5)	1.000000	0.000000
Solver Type:	B-and-B	Nonzeros		DEM(1)	10.00000	0.000000
Best Obj:	210.7	Total:	410	DEM(2)	10.00000	0.000000
Obj Bound:	210.7	Nonlinear:	0	DEM(3)	10.00000	0.000000
Steps:	0	Generator Memory Used (K)		DEM(4)	10.00000	0.000000
Active:	0	65		DEM(5)	10.00000	0.000000
		Elapsed Runtime (hh:mm:ss)		DEM(6)	10.00000	0.000000
		00:00:00		DEM(7)	10.00000	0.000000
				DEM(8)	10.00000	0.000000
				DEM(9)	10.00000	0.000000
				DEM(10)	10.00000	0.000000
				DEM(11)	10.00000	0.000000
				DEM(12)	10.00000	0.000000

Figure 3. 13: Résultats du modèle Lingo P-médian

Interprétation des résultats :

Les résultats des affectations sont comme suivant :

Les cinq sites sont ouverts du coup les affectations selon la solution optimale sont les suivantes :

- Les zones 03,09,10 sont affectées **au point de vente 01**.

- Les zones 04,05,06 et 15 sont affectées **au point de vente 02**.
- Les zones 01,02 ,07,08 sont affectées **au point de vente 03**.
- La zone 14 est affectée **au point de vente 04**.
- Les zones 11,12,13 et 16 sont affectées **au point de vente 05**.

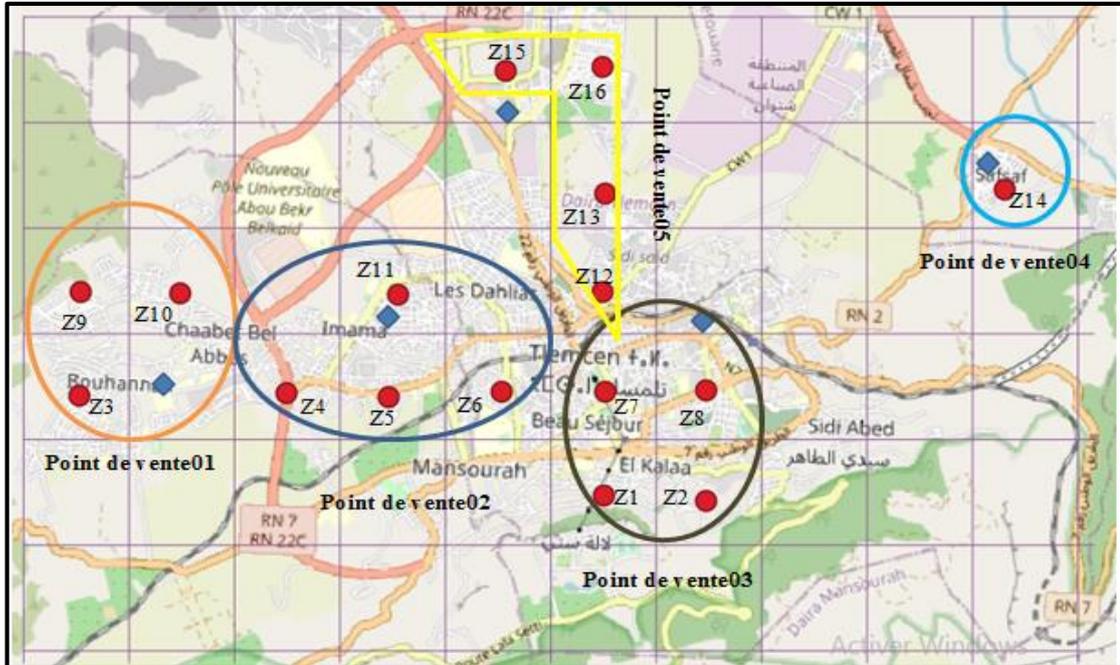


Figure 3. 14: Représentation des résultats sur QGIS

Si on analyse bien les résultats on constate que chaque zone a été affectée au point de vente le plus proche ce qui permet de minimiser la distance parcourue et le cout de déplacement pour chaque client.

Dans un deuxième temps nous avons pris des capacités limitées pour les points de vente et nous avons introduit les demandes réelles estimées pour chaque zone et nous avons ajouté une contrainte disant que la somme des demandes affectées pour chaque point de vente ne doit pas dépasser sa capacité.

Les demandes estimées sont représentées dans par le tableau suivant :

Tableau 3. 3: La demande estimée pour chaque zone

Zones	Zone01	Zone02	Zone03	Zone04	Zone05	Zone06	Zone07	Zone08
Demande	1000	400	1060	400	740	800	400	800
Zones	Zone09	Zone10	Zone11	Zone12	Zone13	Zone14	Zone15	Zone16
Demande	600	400	500	400	400	400	1000	800

III.6.1 Problème de capacité :

Nous avons pris une capacité de 3000 litres pour chaque point de vente se qui dépasse les demandes affectées pour chaque point de vente :

Le modèle Lingo suivant est dans le cas où les capacités sont supérieures ou égal à la demande :

Modèle Lingo P-médian problème de capacité

```
sets:
pointv/1..5/:y,cap;
zone/1..16/:dem;
pointzone(pointv,zone):dist,x;
endsets
data:
costKm = 1;
dem= 1000 400 1060 400 740 800 400 800 600 400 500 400 400 400 1000 800;
cap = 3000 3000 3000 3000 3000;
dist=
  4.  5.  0.  1.  2.  3.  4.  5.  1.  0.  2.  4.  4.  8.  4.  5.1
  26  30  80  21  14  20  15  20  20  94  37  29  60  31  36  0
  2.  3.  3.  1.  0.  1.  2.  3.  2.  1.  2.  2.  2.  6.  2.  3.9
  63  51  01  16  68  28  14  13  93  96  94  08  42  10  60
  1.  1.  5.  3.  3.  2.  1.  0.  5.  4.  2.  1.  1.  3.  3.  2.6
  94  70  94  96  04  02  14  63  91  93  95  00  64  76  01  0
  4.  4.  8.  7.  6.  5.  4.  3.  8.  7.  5.  3.  3.  2.  6.  3.8
  87  13  88  00  07  01  23  38  70  71  76  86  65  70  67  2
  3.  3.  4.  3.  2.  2.  2.  3.  4.  3.  1.  0.  1.  4.  4.  0.9
  77  29  85  32  87  61  76  20  37  48  62  64  09  83  27  5;

enddata

min=@sum(pointv(i):@sum(zone(j):dem(j)*dist(i,j)*costKm*x(i,j)));
```

```
!(1.1); @sum(pointv(i):y(i))=5;
!(1.2); @for(zone(j):@sum(pointv(i):x(i,j))=1);
!(1.3); @for(pointv(i):@for(zone(j):x(i,j)<=y(i)));
!(1.4); @for(pointv(i):@sum(zone(j):x(i,j))>=y(i));
!(1.5);@for(pointv(i@:(sum)zone(j):dem(j)*x(i,j))<=cap)) ;
@for(pointv:@bin(y));
@for(pointv(i):@for(zone(j):@bin(x(i,j))));
```

- (1.1)** :Nombre de point de vente à ouvrir.
- (1.2)** :Une zone doit être affectée à un seul point de vente.
- (1.3)** :Un point de vente s'il est ouvert, il doit servir au moins une zone.
- (1.4)** :Un point de vente s'il est ouvert, il doit servir au moins une zone.
- (1.5)** :la somme des demandes ne doit pas dépasser la capacitédu point de vente.

Les résultats pour ce modèle sont comme suivants :

Solver Status Model Class: PIIIP State: Global Opt Objective: 13457.2 Infeasibility: 0 Iterations: 16		Variables Total: 85 Nonlinear: 0 Integers: 85		Variable Value COSTKM 1.000000 Y (1) 1.000000 Y (2) 1.000000 Y (3) 1.000000 Y (4) 1.000000 Y (5) 1.000000 CAP (1) 3000.000 CAP (2) 3000.000 CAP (3) 3000.000 CAP (4) 3000.000 CAP (5) 3000.000 DEM (1) 1000.000 DEM (2) 400.0000 DEM (3) 1060.000 DEM (4) 400.0000 DEM (5) 740.0000 DEM (6) 800.0000 DEM (7) 400.0000 DEM (8) 800.0000	
Extended Solver Status Solver Type: B-and-B Best Obj: 13457.2 Obj Bound: 13457.2 Steps: 0 Active: 0		Constraints Total: 108 Nonlinear: 0			
		Nonzeros Total: 490 Nonlinear: 0			
		Generator Memory Used (K) 52			
		Elapsed Runtime (hh:mm:ss) 00:00:00			

Figure 3. 15: Résultats du modèle Lingo P-médian problème de capacité

Interprétation :

Les capacités des points de vente sont satisfaisantes pour les demandes des clients donc le modèle dans ce cas donne un résultat optimal qui est **13457.2** et garde les mêmes affectations du premier modèle puisque les demandes sont satisfaites.

Le troisième cas représente le cas où les capacités sont inférieures à la demande nous allons donc diminuer les capacités jusqu'elles seront inférieures à la demande globale qui est égale à 10000 litres par jour.

Nous avons pris une capacité globale de 9000 litres de tous les points de vente :

Modèle Lingo P-médian problème de capacité

```

sets:
pointv/1..5/:y,cap;
zone/1..16/:dem;
pointzone(pointv,zone):dist,x;
endsets
data:
costKm = 1;
dem= 1000 400 1060 400 740 800 400 800 600 400 500 400 400 400 1000 800;
cap= 1000 2000 2000 2000 2000;

dist=
4. 5. 0. 1. 2. 3. 4. 5. 1. 0. 2. 4. 4. 8. 4. 5.1
26 30 80 21 14 20 15 20 20 94 37 29 60 31 36 0
2. 3. 3. 1. 0. 1. 2. 3. 2. 1. 2. 2. 2. 6. 2. 3.9
63 51 01 16 68 28 14 13 93 96 94 08 42 10 60
1. 1. 5. 3. 3. 2. 1. 0. 5. 4. 2. 1. 1. 3. 3. 2.6
94 70 94 96 04 02 14 63 91 93 95 00 64 76 01 0
4. 4. 8. 7. 6. 5. 4. 3. 8. 7. 5. 3. 3. 2. 6. 3.8
87 13 88 00 07 01 23 38 70 71 76 86 65 70 67 2
3. 3. 4. 3. 2. 2. 2. 3. 4. 3. 1. 0. 1. 4. 4. 0.9
77 29 85 32 87 61 76 20 37 48 62 64 09 83 27 5;
Enddata

```

```

min=@sum(pointv(i):@sum(zone(j):dem(j)*dist(i,j)*costKm*x(i,j)));
!(1.1); @sum(pointv(i):y(i))=5;
!(1.2); @for(zone(j):@sum(pointv(i):x(i,j))=1);
!(1.3); @for(pointv(i):@for(zone(j):x(i,j)<=y(i)));
!(1.4); @for(pointv(i):@sum(zone(j):x(i,j))>=y(i));
!(1.5);@for(pointv(i@:(sum)zone(j):dem(j)*x(i,j))<=cap)) ;
@for(pointv:@bin(y));
@for(pointv(i):@for(zone(j):@bin(x(i,j))));

```

- (1.1) :Nombre de point de vente à ouvrir.
(1.2) : Une zone doit être affectée à un seul point de vente.
(1.3) : Un point de vente s'il est ouvert, il doit servir au moins une zone.
(1.4) :Un point de vente s'il est ouvert, il doit servir au moins une zone.
(1.5) :la somme des demandes ne doit pas dépasser la capacitédu point de vente.

Les résultats pour ce modèle sont comme suivants :

Solver Status		Variables		Error Code:	
Model Class:	PILP	Total:	85	81	
State:	Infeasible	Nonlinear:	0		
Objective:	0	Integers:	85		
Infeasibility:	...	Constraints		Error Text:	
Iterations:	30	Total:	108	No feasible solution found.	
		Nonlinear:	0		
Extended Solver Status		Nonzeros			
		Total:	490		

Figure 3. 16: Résultats du modèle Lingo P-médian problème de capacité

Interprétation :

Dans le cas où les capacités des points de vente sont supérieures ou égal à l'ensemble des demandes nous avons eu des résultats de tel sorte chaque zone a été affectée pour le point de vente le plus proche et qui peut satisfaire sa demande.

Dans le cas où les capacités sont inférieures à la demande c'était impossible d'affecter toutes les zones en raison de manque de capacité, et puisque nous avons imposé que chaque zone doit être affecté à un point de vente ces deux dernières contraintes ne peuvent pas être vérifiées du coup le solver nous a donné une solution infaisable.

III.6.2 Problème de distance maximale :

Dans ce cas nous allons ajouter une contrainte par rapport à la distance :

Modèle Lingo P-médian problème de distance

```

sets :
pointv/1..5/:y,cap;
zone/1..16/:dem;
pointzone(pointv,zone):dist,x;
endsets
data:

```

```

costKm = 1;
dem= 1000 400 1060 400 740 800 400 800 600 400 500 400 400 400 1000 800;
cap= 3000 3000 3000 3000 3000;
dist=
  4.  5.  0.  1.  2.  3.  4.  5.  1.  0.  2.  4.  4.  8.  4.  5.1
 26  30  80  21  14  20  15  20  20  94  37  29  60  31  36  0
  2.  3.  3.  1.  0.  1.  2.  3.  2.  1.  2.  2.  2.  6.  2.  3.9
 63  51  01  16  68  28  14  13  93  96  94  08  42  10  60
  1.  1.  5.  3.  3.  2.  1.  0.  5.  4.  2.  1.  1.  3.  3.  2.6
 94  70  94  96  04  02  14  63  91  93  95  00  64  76  01  0
  4.  4.  8.  7.  6.  5.  4.  3.  8.  7.  5.  3.  3.  2.  6.  3.8
 87  13  88  00  07  01  23  38  70  71  76  86  65  70  67  2
  3.  3.  4.  3.  2.  2.  2.  3.  4.  3.  1.  0.  1.  4.  4.  0.9
 77  29  85  32  87  61  76  20  37  48  62  64  09  83  27  5;

enddata

min=@sum(pointv(i):@sum(zone(j):dem(j)*dist(i,j)*costKm*x(i,j)));
@sum(pointv(i):@sum(zone(j):dist(i,j)*x(i,j)))=20;
!(1.1); @sum(pointv(i):y(i))=5;
!(1.2); @for(zone(j):@sum(pointv(i):x(i,j))=1);
!(1.3); @for(pointv(i):@for(zone(j):x(i,j)<=y(i)));
!(1.4); @for(pointv(i):@sum(zone(j):x(i,j))>=y(i));
!(1.5);@for(pointv(i@:(sum)zone(j):dem(j)*x(i,j))<=cap)) ;
@for(pointv:@bin(y));
@for(pointv(i):@for(zone(j):@bin(x(i,j))));

```

- (1.1) :Nombre de point de vente à ouvrir.
(1.2) : Une zone doit être affectée à un seul point de vente.
(1.3) :Un point de vente s'il est ouvert, il doit servir au moins une zone.
(1.4) :Un point de vente s'il est ouvert, il doit servir au moins une zone.
(1.5) :la somme des demandes ne doit pas dépasser la capacité du point de vente.

Les résultats pour ce modèle sont comme suivants :

Solver Status		Variables		Error Code:	
Model Class:	PILP	Total:	85	81	
State:	Infeasible	Nonlinear:	0		
Objective:	0	Integers:	85		
Infeasibility:	. . .	Constraints		Error Text:	
Iterations:	30	Total:	108	No feasible solution found.	
		Nonlinear:	0		
Extended Solver Status		Nonzeros			
		Total:	490		

Figure 3. 17: Résultats du modèle Lingo P-médian problème de distance

Interprétation :

Nous avons imposé que la distance globale de la chaîne de distribution doit être égale à **20** km le solveur n'a pas pu trouver une solution du coup il a indiqué que le modèle est non faisable.

III.6.3 Problème des points de vente :

Cette fois-ci nous allons jouer sur le nombre des points de vente candidats nous allons changer le nombre plusieurs fois et on constate les résultats :

Le modèle Lingo suivant est dans le cas où le nombre des points candidats est égal à cinq :

Modèle Lingo P-médian problème de nombre des points de ventes

```
sets :
pointv/1..5/:y, cap;
zone/1..16/:dem;
pointzone(pointv, zone):dist, x;
endsets
data:
costKm = 1;
dem= 1000 400 1060 400 740 800 400 800 600 400 500 400 400 400 1000 800;
cap= 3000 3000 3000 3000 3000;

dist=
4. 5. 0. 1. 2. 3. 4. 5. 1. 0. 2. 4. 4. 8. 4. 5.1
26 30 80 21 14 20 15 20 20 94 37 29 60 31 36 0
2. 3. 3. 1. 0. 1. 2. 3. 2. 1. 2. 2. 2. 6. 2. 3.9
63 51 01 16 68 28 14 13 93 96 94 08 42 10 60
1. 1. 5. 3. 3. 2. 1. 0. 5. 4. 2. 1. 1. 3. 3. 2.6
94 70 94 96 04 02 14 63 91 93 95 00 64 76 01 0
4. 4. 8. 7. 6. 5. 4. 3. 8. 7. 5. 3. 3. 2. 6. 3.8
87 13 88 00 07 01 23 38 70 71 76 86 65 70 67 2
3. 3. 4. 3. 2. 2. 2. 3. 4. 3. 1. 0. 1. 4. 4. 0.9
77 29 85 32 87 61 76 20 37 48 62 64 09 83 27 5;
enddata

min=@sum(pointv(i):@sum(zone(j):dem(j)*dist(i,j)*costKm*x(i,j)));

!(1.1); @sum(pointv(i):y(i))<=5;
!(1.2); @for(zone(j):@sum(pointv(i):x(i,j))=1);
!(1.3); @for(pointv(i):@for(zone(j):x(i,j)<=y(i)));
!(1.4); @for(pointv(i):@sum(zone(j):x(i,j))>=y(i));
!(1.5);@for(pointv(i@:(sum) zone(j):dem(j)*x(i,j))<=cap)) ;
@for(pointv:@bin(y));
@for(pointv(i):@for(zone(j):@bin(x(i,j))));
```

- (1.1)** :Nombre de point de vente à ouvrir.
- (1.2)** :Une zone doit être affectée à un seul point de vente.
- (1.3)** :Un point de vente s'il est ouvert, il doit servir au moins une zone.
- (1.4)** :Un point de vente s'il est ouvert, il doit servir au moins une zone.
- (1.5)** :la somme des demandes ne doit pas dépasser la capacité du point de vente.

Les résultats pour ce modèle sont comme suivants :

Solver Status Model Class: FILP State: Global Opt Objective: 13457.2 Infeasibility: 0 Iterations: 16		Variables: Total: 85 Nonlinear: 0 Integers: 85		<table border="1"> <thead> <tr> <th>Variable</th> <th>Value</th> <th>Reduced Cost</th> </tr> </thead> <tbody> <tr><td>COSTKM</td><td>1.000000</td><td>0.000000</td></tr> <tr><td>Y(1)</td><td>1.000000</td><td>0.000000</td></tr> <tr><td>Y(2)</td><td>1.000000</td><td>0.000000</td></tr> <tr><td>Y(3)</td><td>1.000000</td><td>0.000000</td></tr> <tr><td>Y(4)</td><td>1.000000</td><td>0.000000</td></tr> <tr><td>Y(5)</td><td>1.000000</td><td>0.000000</td></tr> <tr><td>CAP(1)</td><td>3000.000</td><td>0.000000</td></tr> <tr><td>CAP(2)</td><td>3000.000</td><td>0.000000</td></tr> <tr><td>CAP(3)</td><td>3000.000</td><td>0.000000</td></tr> <tr><td>CAP(4)</td><td>3000.000</td><td>0.000000</td></tr> <tr><td>CAP(5)</td><td>3000.000</td><td>0.000000</td></tr> <tr><td>DEM(1)</td><td>1000.000</td><td>0.000000</td></tr> <tr><td>DEM(2)</td><td>400.0000</td><td>0.000000</td></tr> <tr><td>DEM(3)</td><td>1060.000</td><td>0.000000</td></tr> <tr><td>DEM(4)</td><td>400.0000</td><td>0.000000</td></tr> <tr><td>DEM(5)</td><td>740.0000</td><td>0.000000</td></tr> <tr><td>DEM(6)</td><td>800.0000</td><td>0.000000</td></tr> <tr><td>DEM(7)</td><td>400.0000</td><td>0.000000</td></tr> <tr><td>DEM(8)</td><td>800.0000</td><td>0.000000</td></tr> </tbody> </table>			Variable	Value	Reduced Cost	COSTKM	1.000000	0.000000	Y(1)	1.000000	0.000000	Y(2)	1.000000	0.000000	Y(3)	1.000000	0.000000	Y(4)	1.000000	0.000000	Y(5)	1.000000	0.000000	CAP(1)	3000.000	0.000000	CAP(2)	3000.000	0.000000	CAP(3)	3000.000	0.000000	CAP(4)	3000.000	0.000000	CAP(5)	3000.000	0.000000	DEM(1)	1000.000	0.000000	DEM(2)	400.0000	0.000000	DEM(3)	1060.000	0.000000	DEM(4)	400.0000	0.000000	DEM(5)	740.0000	0.000000	DEM(6)	800.0000	0.000000	DEM(7)	400.0000	0.000000	DEM(8)	800.0000	0.000000
Variable	Value	Reduced Cost																																																																
COSTKM	1.000000	0.000000																																																																
Y(1)	1.000000	0.000000																																																																
Y(2)	1.000000	0.000000																																																																
Y(3)	1.000000	0.000000																																																																
Y(4)	1.000000	0.000000																																																																
Y(5)	1.000000	0.000000																																																																
CAP(1)	3000.000	0.000000																																																																
CAP(2)	3000.000	0.000000																																																																
CAP(3)	3000.000	0.000000																																																																
CAP(4)	3000.000	0.000000																																																																
CAP(5)	3000.000	0.000000																																																																
DEM(1)	1000.000	0.000000																																																																
DEM(2)	400.0000	0.000000																																																																
DEM(3)	1060.000	0.000000																																																																
DEM(4)	400.0000	0.000000																																																																
DEM(5)	740.0000	0.000000																																																																
DEM(6)	800.0000	0.000000																																																																
DEM(7)	400.0000	0.000000																																																																
DEM(8)	800.0000	0.000000																																																																
Extended Solver Status Solver Type: B-and-B Best Obj: 13457.2 Obj Bound: 13457.2 Steps: 0 Active: 0		Constraints: Total: 108 Nonlinear: 0																																																																
		Nonzeros Total: 490 Nonlinear: 0																																																																
		Generator Memory Used (K) 52																																																																
		Elapsed Runtime (hh:mm:ss) 00:00:01																																																																

Figure 3. 18: Résultats du modèle Lingo P-médian problème de nombre des points de ventes

Interprétation :

Dans ce cas Nous avons constaté que le résultat optimal est **13457.2** et que tous les points de ventes sont tous ouverts, car le modèle vise à maximiser le profit.

Le modèle Lingo suivant est dans le cas où nous avons changé le nombre des sites à quatre :

Modèle Lingo P-médian problème de nombre des points de ventes

```

sets :
pointv/1..5/:y,cap;
zone/1..16/:dem;
pointzone(pointv,zone):dist,x;
endsets
data:
costKm = 1;
dem= 1000 400 1060 400 740 800 400 800 600 400 500 400 400 400 1000 800;
cap= 3000 3000 3000 3000 3000;

dist=
4. 5. 0. 1. 2. 3. 4. 5. 1. 0. 2. 4. 4. 8. 4. 5.1
26 30 80 21 14 20 15 20 20 94 37 29 60 31 36 0
2. 3. 3. 1. 0. 1. 2. 3. 2. 1. 2. 2. 2. 6. 2. 3.9
63 51 01 16 68 28 14 13 93 96 94 08 42 10 60
1. 1. 5. 3. 3. 2. 1. 0. 5. 4. 2. 1. 1. 3. 3. 2.6
94 70 94 96 04 02 14 63 91 93 95 00 64 76 01 0
4. 4. 8. 7. 6. 5. 4. 3. 8. 7. 5. 3. 3. 2. 6. 3.8
87 13 88 00 07 01 23 38 70 71 76 86 65 70 67 2
3. 3. 4. 3. 2. 2. 2. 3. 4. 3. 1. 0. 1. 4. 4. 0.9
77 29 85 32 87 61 76 20 37 48 62 64 09 83 27 5;
Enddata

min=@sum(pointv(i):@sum(zone(j):dem(j)*dist(i,j)*costKm*x(i,j)));
!(1.1); @sum(pointv(i):y(i))<=4;
!(1.2); @for(zone(j):@sum(pointv(i):x(i,j))=1);
!(1.3); @for(pointv(i):@for(zone(j):x(i,j)<=y(i)));
!(1.4); @for(pointv(i):@sum(zone(j):x(i,j))>=y(i));
!(1.5);@for(pointv(i):@sum(zone(j):dem(j)*x(i,j))<=cap(i));
@for(pointv:@bin(y));
@for(pointv(i):@for(zone(j):@bin(x(i,j))));

```

- (1.1) : Nombre de point de vente à ouvrir.
 (1.2) : Une zone doit être affectée à un seul point de vente.
 (1.3) : Un point de vente s'il est ouvert, il doit servir au moins une zone.
 (1.4) : Un point de vente s'il est ouvert, il doit servir au moins une zone.
 (1.5) : la somme des demandes ne doit pas dépasser la capacité du point de vente.

Les résultats pour ce modèle sont comme suivants :

Solver Status Model Class: PILP State: Global Opt Objective: 13881.2 Infeasibility: 0 Iterations: 38		Variables Total: 85 Nonlinear: 0 Integers: 85		Variable Value COSTKM 1.000000 Y (1) 1.000000 Y (2) 1.000000 Y (3) 1.000000 Y (4) 0.000000 Y (5) 1.000000 CAP (1) 3000.000 CAP (2) 3000.000 CAP (3) 3000.000 CAP (4) 3000.000 CAP (5) 3000.000 DEM (1) 1000.000 DEM (2) 400.0000 DEM (3) 1060.000 DEM (4) 400.0000 DEM (5) 740.0000 DEM (6) 800.0000 DEM (7) 400.0000 DEM (8) 800.0000	
Extended Solver Status Solver Type: B-and-B Best Obj: 13881.2 Obj Bound: 13881.2 Steps: 0 Active: 0		Constraints Total: 108 Nonlinear: 0			
		Nonzeros Total: 490 Nonlinear: 0			
		Generator Memory Used (K) 52			
		Elapsed Runtime (hh:mm:ss) 00:00:00			

Figure 3. 19: Résultats du modèle Lingo P-médian problème de nombre de points de ventes

Interprétation :

Dans ce cas nous avons constaté que la solution optimale **13881.2** est supérieure à la solution précédente **13457.2** en raison de diminution des points de vente par conséquent la distance parcourue a été augmentée.

Ce modèle représente le cas où nous avons imposé que le nombre des points de vente doit être inférieure à quatre :

Modèle Lingo P-médian problème de nombre des points de ventes

```
sets :
pointv/1..5/:y, cap;
zone/1..16/:dem;
pointzone(pointv, zone):dist, x;
endsets
data:
costKm = 1;
dem= 1000 400 1060 400 740 800 400 800 600 400 500 400 400 400 1000 800;
cap= 3000 3000 3000 3000 3000;
dist=
4.  5.  0.  1.  2.  3.  4.  5.  1.  0.  2.  4.  4.  8.  4.  5.1
26 30  80 21  14 20 15 20 20 94 37 29 60 31 36  0
2.  3.  3.  1.  0.  1.  2.  3.  2.  1.  2.  2.  2.  6.  2.  3.9
63 51  01 16  68 28 14 13 93 96 94 08 42 10 60
1.  1.  5.  3.  3.  2.  1.  0.  5.  4.  2.  1.  1.  3.  3.  2.6
94 70  94 96  04 02 14 63 91 93 95 00 64 76 01  0
4.  4.  8.  7.  6.  5.  4.  3.  8.  7.  5.  3.  3.  2.  6.  3.8
87 13  88 00  07 01 23 38 70 71 76 86 65 70 67  2
```

```

3. 3. 4. 3. 2. 2. 2. 3. 4. 3. 1. 0. 1. 4. 4. 0.9
77 29 85 32 87 61 76 20 37 48 62 64 09 83 27 5;
Enddata

```

```

min=@sum(pointv(i):@sum(zone(j):dem(j)*dist(i,j)*costKm*x(i,j)));
@sum(pointv(i):@sum(zone(j):dem(j)*dist(i,j)*costKm*x(i,j)))<=13500;

!(1.1); @sum(pointv(i):y(i))<=4;

!(1.2); @for(zone(j):@sum(pointv(i):x(i,j))=1);
!(1.3); @for(pointv(i):@for(zone(j):x(i,j)<=y(i)));
!(1.4); @for(pointv(i):@sum(zone(j):x(i,j))>=y(i));
!(1.5);@for(pointv(i):@sum(zone(j):dem(j)*x(i,j))<=cap);
@for(pointv(i):@bin(y));
@for(pointv(i):@for(zone(j):@bin(x(i,j))));

```

- (1.1) :Nombre de point de vente à ouvrir.
(1.2) :Une zone doit être affectée à un seul point de vente.
(1.3) :Un point de vente s'il est ouvert, il doit servir au moins une zone.
(1.4) :Un point de vente s'il est ouvert, il doit servir au moins une zone.
(1.5) :la somme des demandes ne doit pas dépasser la capacité du point de vente.

Les résultats pour ce modèle sont comme suivants :

Solver Status		Variables		Error Code:	
Model Class:	PILP	Total:	85	81	
State:	Infeasible	Nonlinear:	0		
Objective:	0	Integers:	85	Error Text:	
Infeasibility:	. . .	Constraints		No feasible solution found.	
Iterations:	30	Total:	108		
Extended Solver Status:		Nonlinear:	0		
		Nonzeros			
		Total:	490		

Figure 3. 20: Résultats du modèle Lingo P-médian problème de nombre de points de ventes

Interprétation :

Quand nous avons ajouté une contrainte qui impose que la fonction objective ne doit pas dépasser la valeur **13500**. Sachant que nous connaissons la solution pour quatre points de vente et pour cinq aussi, et puis nous avons imposé que le nombre des points de vente doit être inférieur ou égal à quatre, du coup le modèle sera non faisable.

III.7 Problématique 02 :

Dans un deuxième temps nous avons supposé que chaque famille consomme quatre litres par jour. Selon les mêmes itérations précédentes nous avons trouvé une consommation journalière égale à 20760 litres par jour ce qui est supérieure à la quantité produite chaque jour par l'usine donc il existe un manque et la demande ne sera pas satisfaite. Pour ce faire nous avons pensé à introduire le modèle **MAX-COVERING** pour maximiser les zones couvertes tout en essayant

d'affecter chaque zone au point de vente le plus proche pour minimiser les coûts de déplacement.

Le modèle mathématique du problème p-médian est comme suivant :

Les entrées :

$$a_{ij} = \begin{cases} 1 & \text{si le site candidat } j \in J \text{ peut couvrir les demandes au niveau du noeud } i \in I \\ 0 & \text{sinon} \end{cases}$$

Variable de décision :

$$z_i = \begin{cases} 1 & \text{si le client } i \in I \text{ est couvert} \\ 0 & \text{sinon} \end{cases}$$

h_i : représente la demande du client i .

$$\max = \sum_{i \in I} h_i * z_i \quad (2.1)$$

Sous les contraintes suivantes :

$$z_i \leq \sum_{j \in J} a_{ij} X_j \quad \forall i \in I \quad (2.2)$$

$$\sum_{j \in J} X_j \leq P \quad (2.3)$$

$$X_j \in \{0,1\} \quad \forall j \in J \quad (2.4)$$

$$z_i \in \{0,1\} \quad \forall i \in I \quad (2.5)$$

- La fonction objective (2.1) maximise le nombre des demandes couvertes.
- La contrainte (2.2) impose que chaque demande ne peut être couverte si seulement si un des sites qui la couvre était ouvert.
- La contrainte (2.3) indique le nombre des sites qu'on peut ouvrir donc on ne peut pas ouvrir plus de p sites.
- Les deux contraintes (2.4) et (2.5) montre la nature binaire des variables de décision z_i et X_i .

En supposant au début qu'on a neuf zones et trois points candidats, demande identique. Le but est de maximiser les couvertures pour avoir le maximum possible des zones couvertes.

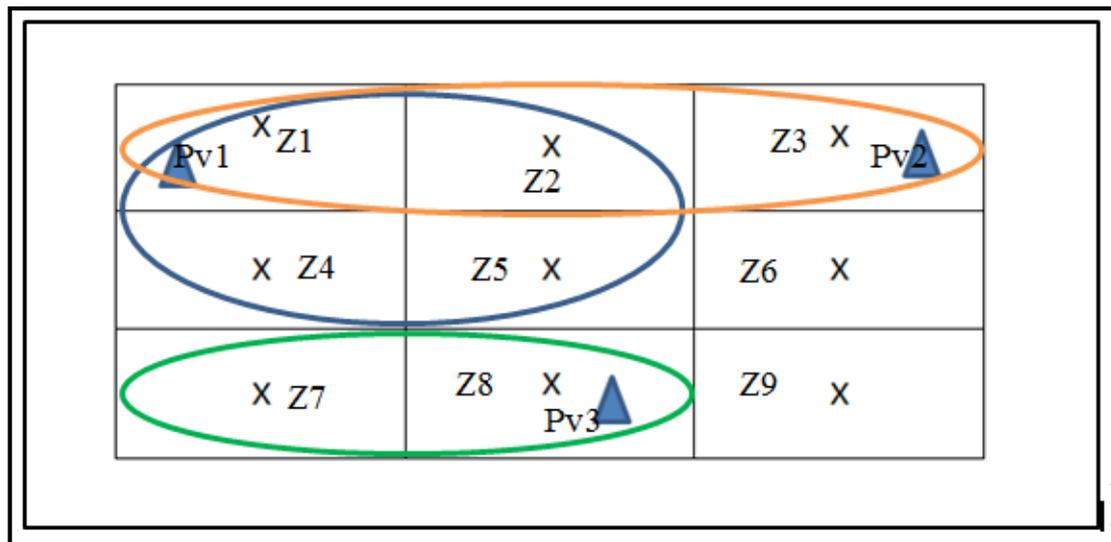


Figure 3. 21: Schéma explicatif

Explication :

Dans un cas où toutes les demandes sont identiques et nous pouvons ouvrir que deux points de vente, le but est de maximiser les zones couvertes ou d'une autre façon les demandes, le solver va ouvrir soit A et B ou A et C puisqu'ils couvrent le maximum des demandes sachant qu'une zone peut être couverte deux fois par contre n'est pas complètement couverte.

Dans un autre cas on suppose que la demande de la zone 03 est égal à 50 dans ce cas-là le solver va ouvrir A et B puisqu'ils couvrent le maximum des demandes

Ce modèle représente le modèle max-covering de base ou les affectations des clients se fait par rapport à la matrice « a ».

Modèle max-covering standard

```

sets :
pointv/1..3/:y, cap;
zone/1..9/:dem, z;
pointzone(pointv, zone):dist, a, x;
endsets
data:
dem= 10  10  10  10  10  10  10  10  10;

a=  1  1  0  1  1  0  0  0  0
1  1  1  0  0  0  0  0  0
0  0  0  0  0  0  1  1  0;
enddata
max=@sum(zone(j):dem(j)*z(j));
@for(zone(j):@sum(pointv(i):a(i,j)*y(i))>=z(j));
@sum(pointv(i):y(i))<=2;
@for(zone(j):@bin(z));
@for(pointv:@bin(y));

```

Les résultats pour ce modèle sont comme suivants :

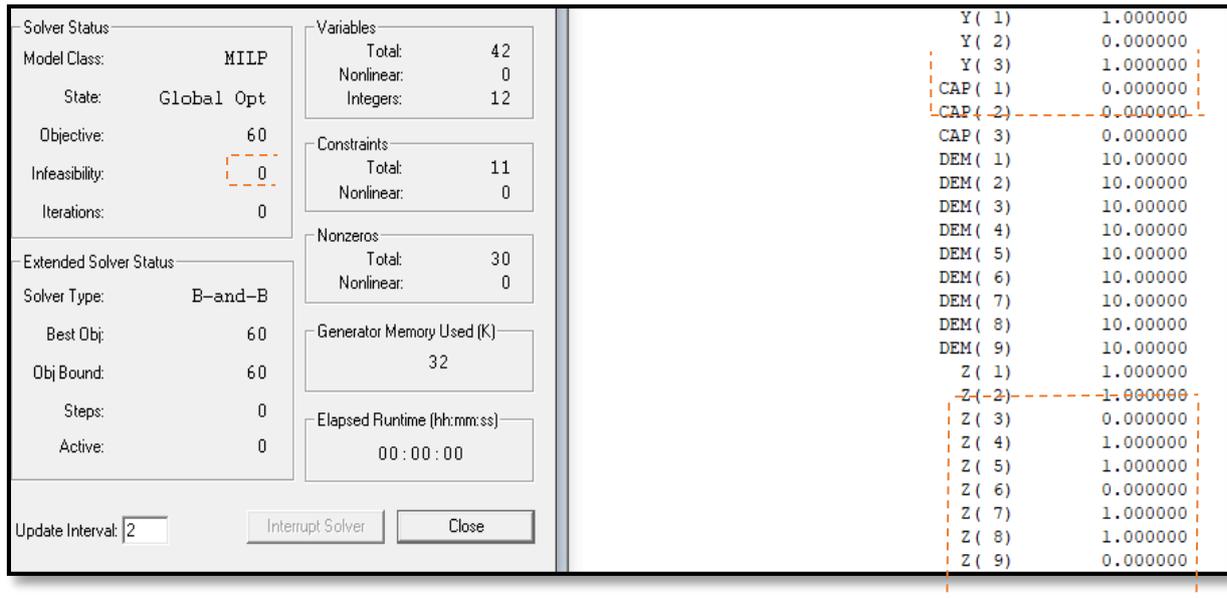


Figure 3. 22: Résultats du modèle Lingo max-covering standards

Nous allons maintenant faire un changement sur les demandes et on constate les résultats :

Modèle max-covering standards

```
sets:
pointv/1..3/:y;
zone/1..9/ :z;
pointzone(pointv,zone):dist,a,x;
endsets
data:
dem= 10 10 10 50 10 10 10 10 10;

a=
    1 1 0 1 1 0 0 0 0
    1 1 1 0 0 0 0 0 0
    0 0 0 0 0 0 1 1 0;

enddata
max=@sum(zone(j):dem(j)*z(j));
@for(zone(j):@sum(pointv(i):a(i,j)*y(i))>=z(j));
@sum(pointv(i):y(i))<=2;
@for(zone(j):@bin(z));
@for(pointv:@bin(y));
```

Les résultats pour ce modèle sont comme suivants :

Solver Status Model Class: MILP State: Global Opt Objective: 90 Infeasibility: 0 Iterations: 0		Variables Total: 42 Nonlinear: 0 Integers: 12		Y(1) 1.000000 0.000000 Y(2) 1.000000 0.000000 Y(3) 0.000000 0.000000 CAP(1) 0.000000 0.000000 CAP(2) 0.000000 0.000000 CAP(3) 0.000000 0.000000 DEM(1) 10.000000 0.000000 DEM(2) 10.000000 0.000000 DEM(3) 50.000000 0.000000 DEM(4) 10.000000 0.000000 DEM(5) 10.000000 0.000000 DEM(6) 10.000000 0.000000 DEM(7) 10.000000 0.000000 DEM(8) 10.000000 0.000000 DEM(9) 10.000000 0.000000 Z(1) 1.000000 -10.000000 Z(2) 1.000000 -10.000000 Z(3) 1.000000 -50.000000 Z(4) 1.000000 -10.000000 Z(5) 1.000000 -10.000000 Z(6) 0.000000 -10.000000 Z(7) 0.000000 -10.000000 Z(8) 0.000000 -10.000000 Z(9) 0.000000 -10.000000		
Extended Solver Status Solver Type: B-and-B Best Obj: 90 Obj Bound: 90 Steps: 0 Active: 0		Constraints Total: 11 Nonlinear: 0				
		Nonzeros Total: 30 Nonlinear: 0				
		Generator Memory Used (K) 32				
		Elapsed Runtime (hh:mm:ss) 00:00:00				
Update Interval: 2 Interrupt Solver Close						

Figure 3. 23: Résultats du modèle Lingo max-covering standard

Interprétation :

On constate que les points de vente ont été changé le fait de changer la demande puisque on vise à la base à maximiser la demande donc le solver va choisir les points de vente qui ont plus de demande à ouvrir. Mais le point aveugle de ce modèle c'est qu'on peut affecter une zone plusieurs fois tandis qu'une n'est pas complètement affecté. En plus, ce n'est pas pratique que à chaque fois nous modifions ou ajoutons une zone a la matrice « a » nous calculons depuis le début.

Donc nous avons remplacé cette matrice par une matrice de distance dont le solver visera à affecter les zones au point candidat le plus proche par rapport à la distance.

Le modèle suivant représente le max-covering avec la matrice de distance :

Modèle max-covering standard

```

sets :
pointv/1..3/:y,cap;
zone/1..9/:dem,z;
pointzone(pointv,zone):dist,a;
endsets
data:
dem= 10 10 10 10 10 10 10 10 10;

dist=4.26 5.30 0.80 1.21 2.14 3.20 4.15 5.20 1.20
      2.63 3.51 3.01 1.16 0.68 1.28 2.14 3.13 2.93
      1.94 1.70 5.94 3.96 3.04 2.02 1.14 0.63 5.91;
enddata
max=@sum(zone(j):dem(j)*z(j));
@for(zone(j):@sum(pointv(i):dist(i,j)*y(i))>=z(j));
@sum(pointv(i):y(i))<=2;
@for(zone(j):@bin(z));
@for(pointv:@bin(y));
@for(pointv(i):@for(zone(j):@bin(a(i,j))));

```

Les résultats pour ce modèle sont comme suivants :

Solver Status Model Class: MILP State: Global Opt Objective: 60 Infeasibility: 0 Iterations: 0		Variables Total: 42 Nonlinear: 0 Integers: 12		Y (1) 1.000000 Y (2) 0.000000 Y (3) 1.000000 CAP (1) 0.000000 CAP (2) 0.000000 CAP (3) 0.000000 DEM (1) 10.000000 DEM (2) 10.000000 DEM (3) 10.000000 DEM (4) 10.000000 DEM (5) 10.000000 DEM (6) 10.000000 DEM (7) 10.000000 DEM (8) 10.000000 DEM (9) 10.000000 Z (1) 1.000000 Z (2) 1.000000 Z (3) 0.000000 Z (4) 1.000000 Z (5) 1.000000 Z (6) 0.000000 Z (7) 1.000000 Z (8) 1.000000 Z (9) 0.000000	
Extended Solver Status Solver Type: B-and-B Best Obj: 60 Obj Bound: 60 Steps: 0 Active: 0		Constraints Total: 11 Nonlinear: 0			
Update Interval: 2		Nonzeros Total: 30 Nonlinear: 0			
Interrupt Solver		Generator Memory Used (K) 32			
Close		Elapsed Runtime (hh:mm:ss) 00:00:00			

Figure 3. 24: Résultats du modèle Lingo max-covering standard

Interprétation :

Nous avons remarqué que le résultat reste le même non plus si on utilise la matrice « a » ou la matrice de distance. En plus nous allons utiliser ce modèle dans notre cas d'étude donc c'est plus fonctionnel et bien mieux d'utiliser la matrice de distance.

Par contre dans le modèle standard on ne peut pas savoir pour quel point de vente une zone a été affectée, pour ce faire nous avons fait une extension (un modèle adapté à nos besoins).

Le modèle ci-dessous représente l'extension du modèle max-covering standard :

Modèle max-covering extension

```

sets:
pointv/1..5/:y,cap;
zone/1..16/:dem;
pointzone(pointv,zone):dist,x;
endsets
data:
dem= 2000 800 2120 800 1480 1600 800 1600 1200 800 1000 800 800 800 2000
800;
costKm=1;
PrixVente=1;
cap= 3874 3874 3874 3874 3874;
dist=
4. 5. 0. 1. 2. 3. 4. 5. 1. 0. 2. 4. 4. 8. 4. 5.1
26 30 80 21 14 20 15 20 20 94 37 29 60 31 36 0
2. 3. 3. 1. 0. 1. 2. 3. 2. 1. 2. 2. 2. 6. 2. 3.9
63 51 01 16 68 28 14 13 93 96 94 08 42 10 60
1. 1. 5. 3. 3. 2. 1. 0. 5. 4. 2. 1. 1. 3. 3. 2.6
94 70 94 96 04 02 14 63 91 93 95 00 64 76 01 0
4. 4. 8. 7. 6. 5. 4. 3. 8. 7. 5. 3. 3. 2. 6. 3.8
87 13 88 00 07 01 23 38 70 71 76 86 65 70 67 2
3. 3. 4. 3. 2. 2. 2. 3. 4. 3. 1. 0. 1. 4. 4. 0.9
77 29 85 32 87 61 76 20 37 48 62 64 09 83 27 5;
Enddata
max=@sum(pointv(i):@sum(zone(j):x(i,j)*dem(j)*PrixVente -
dist(i,j)*x(i,j)*costKm));

```

```

!(1.1); @sum(pointv(i):y(i))<=4;
!(1.2); @for(zone(j):@sum(pointv(i):x(i,j))=1);
!(1.3); @for(pointv(i):@for(zone(j):x(i,j)<=y(i)));
!(1.4); @for(pointv(i):@sum(zone(j):x(i,j))>=y(i));
!(1.5); @for(pointv(i):@sum(zone(j):dem(j)*x(i,j))<=cap(i));
!(1.6); @for(pointv(i):@for(zone(j):dist(i,j)*x(i,j)<=4));
@for(pointv:@bin(y));
@for(pointv(i):@for(zone(j):@bin(x(i,j))));

```

- (1.1) :Nombre de point de vente à ouvrir.
(1.2) :Une zone doit être affectée à un seul point de vente.
(1.3) :Un client doit être servi par un point de vente ouvert.
(1.4) :Un point de vente s'il est ouvert, il doit servir au moins une zone.
(1.5) :la somme des demandes ne doit pas dépasser la capacité du point de vente.
(1.6) :la distance maximale à ne pas dépasser entre un point de vente et une zone affectée.

Les résultats pour ce modèle sont comme suivants :

Solver Status		Variables		Error Code:	
Model Class:	PILP	Total:	85	81	
State:	Infeasible	Nonlinear:	0		
Objective:	0	Integers:	85	Error Text:	
Infeasibility:	. . .	Constraints		No feasible solution found.	
Iterations:	30	Total:	108		
Extended Solver Status:		Nonlinear:	0		
		Nonzeros			
		Total:	490		

Figure 3. 25: Résultats du modèle Lingo max-covering extension

Interprétation :

Dans ce cas, puisque les demandes ont augmenté l'usine ne pouvait pas satisfaire les clients. Nous visons donc à couvrir ce manque en fournissant le reste de la quantité chez une autre usine (laiterie alnadjah).

Ensuite, nous avons augmenté les capacités des points de ventes à 4000 :

Modèle max-covering extension

```

sets:
pointv/1..5/:y, cap;
zone/1..16/:dem;
pointzone(pointv, zone):dist, x;
endsets
data:
dem= 2000800 2120 800 1480 1600 800 1600 1200 800 1000 800 800 800 2000
800;
costKm=1;
PrixVente= 1;
cap= 4000 4000 4000 4000 4000;
dist=
4. 5. 0. 1. 2. 3. 4. 5. 1. 0. 2. 4. 4. 8. 4. 5.1
26 30 80 21 14 20 15 20 20 94 37 29 60 31 36 0

```

2.	3.	3.	1.	0.	1.	2.	3.	2.	1.	2.	2.	2.	6.	2.	3.9
63	51	01	16	68	28	14	13	93	96	94	08	42	10	60	
1.	1.	5.	3.	3.	2.	1.	0.	5.	4.	2.	1.	1.	3.	3.	2.6
94	70	94	96	04	02	14	63	91	93	95	00	64	76	01	0
4.	4.	8.	7.	6.	5.	4.	3.	8.	7.	5.	3.	3.	2.	6.	3.8
87	13	88	00	07	01	23	38	70	71	76	86	65	70	67	2
3.	3.	4.	3.	2.	2.	2.	3.	4.	3.	1.	0.	1.	4.	4.	0.9
77	29	85	32	87	61	76	20	37	48	62	64	09	83	27	5;

Enddata

```

max=@sum(pointv(i):@sum(zone(j):x(i,j)*dem(j)*PrixVente -
dist(i,j)*x(i,j)*costKm));
!min=@sum(pointv(i):@sum(zone(j): dist(i,j)*x(i,j)*costKm));
!(1.1); @sum(pointv(i):y(i))<=4;
!(1.2); @for(zone(j):@sum(pointv(i):x(i,j))=1);
!(1.3); @for(pointv(i):@for(zone(j):x(i,j)<=y(i)));
!(1.4); @for(pointv(i):@sum(zone(j):x(i,j))>=y(i));
!(1.5); @for(pointv(i):@sum(zone(j):dem(j)*x(i,j))<=cap(i));
!(1.6); @for(pointv(i):@for(zone(j):dist(i,j)*x(i,j)<=4));
@for(pointv:@bin(y));
@for(pointv(i):@for(zone(j):@bin(x(i,j))));

```

- (1.1) :Nombre de point de vente à ouvrir.
(1.2) :Une zone doit être affectée à un seul point de vente.
(1.3) :Un client doit être servi par un point de vente ouvert.
(1.4) :Un point de vente s'il est ouvert, il doit servir au moins une zone.
(1.5) :la somme des demandes ne doit pas dépasser la capacité du point de vente.
(1.6) :la distance maximale à ne pas dépasser entre un point de vente et une zone affectée.

Les résultats pour ce modèle sont comme suivant :

Solver Status Model Class: PILP State: Global Opt Objective: 19364.8 Infeasibility: 0 Iterations: 2421		Variables Total: 85 Nonlinear: 0 Integers: 85		Constraints Total: 188 Nonlinear: 0		Variable Value COSTKM 1.000000 PRIXVENTE 1.000000 Y (1) 1.000000 Y (2) 1.000000 Y (3) 1.000000 Y (4) 1.000000 Y (5) 1.000000 CAP (1) 4000.000 CAP (2) 4000.000 CAP (3) 4000.000 CAP (4) 4000.000 CAP (5) 4000.000 DEM (1) 2000.000 DEM (2) 800.0000 DEM (3) 2120.000 DEM (4) 800.0000 DEM (5) 1480.000 DEM (6) 1600.000	
Extended Solver Status Solver Type: B-and-B Best Obj: 19364.8 Obj Bound: 19364.7 Steps: 50 Active: 0		Nonzeros Total: 570 Nonlinear: 0		Generator Memory Used (K) 60		Elapsed Runtime (hh:mm:ss) 00:00:00	

Figure 3. 26: : Résultats du modèle Lingo max-covering extension

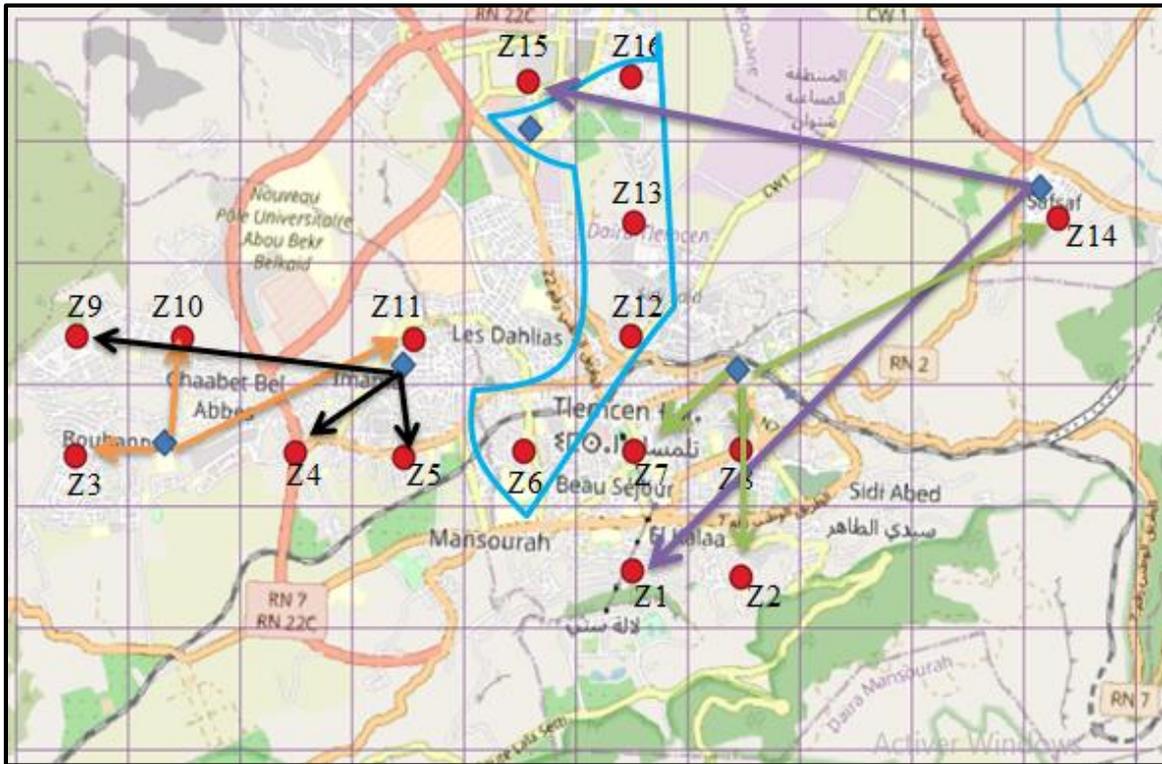


Figure 3. 27: Représentation du résultat sur le QGIS

Modèle max-covering extensiondistance

```

sets:
pointv/1..5/:y, cap;
zone/1..16/:dem;
pointzone(pointv, zone) :dist, x;
endsets
data:
costKm=1;
PrixVente=1;
dem= 2000 800 2120 800 1480 1600 800 1600 1200 800 1000 800 800 800 2000
800;
cap= 4000 4000 4000 4000 4000;
dist=
4. 5. 0. 1. 2. 3. 4. 5. 1. 0. 2. 4. 4. 8. 4. 5.1
26 30 80 21 14 20 15 20 20 94 37 29 60 31 36 0
2. 3. 3. 1. 0. 1. 2. 3. 2. 1. 2. 2. 2. 6. 2. 3.9
63 51 01 16 68 28 14 13 93 96 94 08 42 10 60
1. 1. 5. 3. 3. 2. 1. 0. 5. 4. 2. 1. 1. 3. 3. 2.6
94 70 94 96 04 02 14 63 91 93 95 00 64 76 01 0
4. 4. 8. 7. 6. 5. 4. 3. 8. 7. 5. 3. 3. 2. 6. 3.8
87 13 88 00 07 01 23 38 70 71 76 86 65 70 67 2
3. 3. 4. 3. 2. 2. 2. 3. 4. 3. 1. 0. 1. 4. 4. 0.9
77 29 85 32 87 61 76 20 37 48 62 64 09 83 27 5;
enddata
max=@sum(pointv(i):@sum(zone(j):x(i,j)*dem(j)*PrixVente -
dist(i,j)*x(i,j)*costKm));
@sum(pointv(i):@sum(zone(j):dist(i,j)*x(i,j)))=100; !distance globale de
réseau à ne pas dépasser
!(1.1); @sum(pointv(i):y(i))<=4;
!(1.2); @for(zone(j):@sum(pointv(i):x(i,j))=1);
!(1.3); @for(pointv(i):@for(zone(j):x(i,j)<=y(i)));
!(1.4); @for(pointv(i):@sum(zone(j):x(i,j))>=y(i));
!(1.5); @for(pointv(i):@sum(zone(j):dem(j)*x(i,j))<=cap(i));

```

```
!(1.6); @for(pointv(i):@for(zone(j):dist(i,j)*x(i,j)<=4));
@for(pointv:@bin(y));
@for(pointv(i):@for(zone(j):@bin(x(i,j))));
```

- (1.1) :Nombre de point de vente à ouvrir.
 (1.2) :Une zone doit être affectée à un seul point de vente.
 (1.3) :Un client doit être servi par un point de vente ouvert.
 (1.4) :Un point de vente s'il est ouvert, il doit servir au moins une zone.
 (1.5) :la somme des demandes ne doit pas dépasser la capacité du point de vente.
 (1.6) :la distance maximale à ne pas dépasser entre un point de vente et une zone affectée.

Les résultats pour ce modèle sont comme suivants :

Solver Status		Variables		Error Code:	
Model Class:	PILP	Total:	85	81	
State:	Infeasible	Nonlinear:	0	Error Text:	
Objective:	0	Integers:	85	No feasible solution found.	
Infeasibility:	...	Constraints			
Iterations:	30	Total:	108		
Extended Solver Status:		Nonlinear:	0		
		Nonzeros			
		Total:	490		

Figure 3. 28: Résultats du modèle Lingo max-covering extension problème de distance

Interprétation :

Pour le troisième cas nous avons ajouté une contrainte qui impose que la somme globale des distances parcourues dans le réseau de distribution doit être égale à 100 km par conséquent le modèle ce n'était pas faisable.

III.7.1 Problème des points de vente :

Modèle max-covering extension nombre de points de vente

```
sets:
pointv/1..5/:y,cap;
zone/1..16/:dem;
pointzone(pointv,zone):dist,x;
endsets
data:
costKm=1;
PrixVente=1;
dem= 2000 800 2120 800 1480 1600 800 1600 1200 800 1000 800 800 800 2000
800;
cap= 4000 4000 4000 4000 4000;
dist=
4. 5. 0. 1. 2. 3. 4. 5. 1. 0. 2. 4. 4. 8. 4. 5.1
26 30 80 21 14 20 15 20 20 94 37 29 60 31 36 0
2. 3. 3. 1. 0. 1. 2. 3. 2. 1. 2. 2. 2. 6. 2. 3.9
63 51 01 16 68 28 14 13 93 96 94 08 42 10 60
1. 1. 5. 3. 3. 2. 1. 0. 5. 4. 2. 1. 1. 3. 3. 2.6
94 70 94 96 04 02 14 63 91 93 95 00 64 76 01 0
4. 4. 8. 7. 6. 5. 4. 3. 8. 7. 5. 3. 3. 2. 6. 3.8
87 13 88 00 07 01 23 38 70 71 76 86 65 70 67 2
```

```

3. 3. 4. 3. 2. 2. 2. 3. 4. 3. 1. 0. 1. 4. 4. 0.9
77 29 85 32 87 61 76 20 37 48 62 64 09 83 27 5;

```

```

Enddata

```

```

max=@sum(pointv(i):@sum(zone(j):x(i,j)*dem(j)*PrixVente -
dist(i,j)*x(i,j)*costKm));

```

```

!(1.1); @sum(pointv(i):y(i))<=4;
!(1.2); @for(zone(j):@sum(pointv(i):x(i,j))=1);
!(1.3); @for(pointv(i):@for(zone(j):x(i,j)<=y(i)));
!(1.4); @for(pointv(i):@sum(zone(j):x(i,j))>=y(i));
!(1.5); @for(pointv(i):@sum(zone(j):dem(j)*x(i,j))<=cap(i));
!(1.6); @for(pointv(i):@for(zone(j):dist(i,j)*x(i,j)<=4));
@for(pointv:@bin(y));
@for(pointv(i):@for(zone(j):@bin(x(i,j))));

```

- (1.1) :Nombre de point de vente à ouvrir.
- (1.2) :Une zone doit être affectée à un seul point de vente.
- (1.3) : Un client doit être servi par un point de vente ouvert.
- (1.4) :Un point de vente s'il est ouvert, il doit servir au moins une zone.
- (1.5) :la somme des demandes ne doit pas dépasser la capacité du point de vente.
- (1.6) :la distance maximale à ne pas dépasser entre un point de vente et une zone affectée.

Les résultats pour ce modèle sont comme suivants :

Solver Status		Variables		Error Code:	
Model Class:	PILP	Total:	85	81	
State:	Infeasible	Nonlinear:	0		
Objective:	0	Integers:	85	Error Text:	
Infeasibility:	. . .	Constraints			
Iterations:	30	Total:	108		
Extended Solver Status		Nonlinear:	0		
		Nonzeros			
		Total:	490		

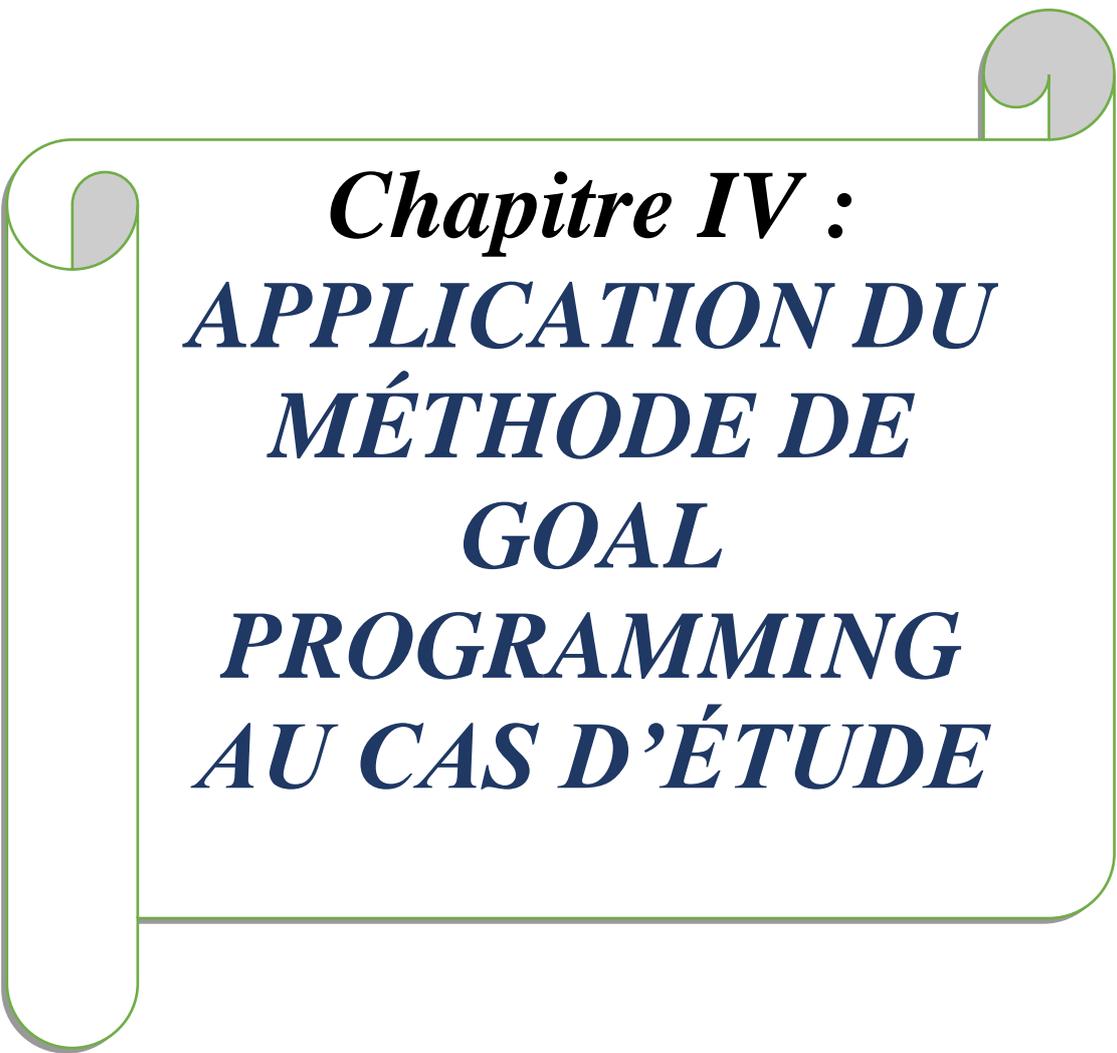
Figure 3. 29: Résultats du modèle Lingo max-covering extension problème de capacité

Interprétation :

Le fait d'exiger trois points de vente à ouvrir ce n'était pas possible de couvrir toutes les demandes et affecter toutes les zones pour cela le modèle n'était pas faisable.

III.8 Conclusion :

Nous avons vu dans ce chapitre les systèmes d'informations géographiques, ainsi que les plateformes logistiques, par la suite nous avons entamé notre cas d'étude par l'application des modèles p-médian et max-covering standard et puis des modèles étendus adaptés à nos besoins et on constate à chaque fois les changements des résultats d'affectation en introduisant à chaque fois des nouvelles contraintes relatives à la capacité, distance maximale et les points de vente respectivement.



Chapitre IV :
***APPLICATION DU
MÉTHODE DE
GOAL
PROGRAMMING
AU CAS D'ÉTUDE***

IV.1 Introduction :

Dans ce chapitre nous sommes à propos de valider et vérifier la fiabilité de la méthode goal programming, pour ce faire nous allons prendre des problèmes illustratifs et les formuler une fois sous forme d'un modèle linéaire et une autre fois sous forme d'un modèle goal programming pour localiser les points de différence. L'analyse des résultats se fait à l'aide du solver Lingo 18 afin de sortir avec une synthèse finale concernant l'efficacité de la méthode goal programming. Ensuite nous allons appliquer les notions de la méthode sur des différents scénarios de notre étude.

IV.2 Description de la problématique :

Nous allons prendre deux problèmes de base un premier qui est pour but de maximiser le profit et le deuxième pour but de minimiser les dépenses. Pour chaque problème nous allons établir un modèle mathématique linéaire et un modèle goal programming, puis nous allons voir un cas pour lequel les deux modèles fournissent un résultat identique et un deuxième cas pour lequel les deux modèles fournissent des résultats différents. À la fin et à partir des résultats obtenus nous allons sortir avec une synthèse concernant l'approche goal programming. La résolution et l'analyse des problèmes se fait à l'aide du solver Lingo 18.0.

IV.3 Exemples illustratifs :

IV.3.1 Exemple1 :

IV.3.1.1 Cas ou les deux méthodes fournissent les mêmes résultats :

Énoncé :

Une entreprise fabrique trois types de produit, elle veut améliorer ses performances afin d'avoir le meilleur profit possible avec les ressources disponibles et en respectant les temps opératoires, les données sont établies dans le tableau suivant :

Tableau 4. 1: Exemple-1- Goal programming

Type de produit	Produit 1	Produit 2	Produit 3	Goals
Temps de traitement(unit)	40	30	200	<=100
Ressources	2	4	3	=10
Profit(Mda)	5	8	4	>=30 M

Nous avons essayé de traduire les données sous forme d'un modèle linéaire comme suivant :

$$Max Z = \sum_{i=1}^3 (X_i * profit_i)$$

$$\sum_{i=1}^3 (X_i * profit_i) \geq 30 \quad \forall i$$

$$\sum_{i=1}^3 (X_i * \text{temps de traitement}_i) \leq 100 \quad \forall i$$

$$\sum_{i=1}^3 (X_i * \text{ressource}_i) \leq 10 \quad \forall i$$

$X_i \geq 0$: variable de décision montre la quantité produite du produit(i)

Si nous exécutons le modèle pour une contrainte du profit qui doit être supérieur ou égal à **30 M** nous allons obtenir une solution infaisable puisqu'avec les deux contraintes de temps et de ressources exigées nous ne pouvons jamais atteindre un profit de **30 M** ou plus.

Modèle linéaire non faisable

```
sets:
product:time,ressources,profit,x;
endsets
data:
time          = 40   30   20;
ressources    =  2    4    3;
profit        =  5    8    4;
enddata
max=@sum(product(i):profit(i)*x(i);
@sum(product(i):profit(i)*x(i))>=30;
@sum(product(i):ressources(i)*x(i))<=10;
@sum(product(i):time(i)*x(i))<=100;
@for(product(i):@gin(x(i)));
```

Le résultat de ce modèle est :

Solver Status Model Class: P1LP State: Infeasible Objective: 0 Infeasibility: . . . Iterations: 0		Variables Total: 3 Nonlinear: 0 Integers: 3
Extended Solver Status Solver Type: B-and-B Best Obj: . . . Obj Bound: 1e+030 Steps: 0 Active: 0		Constraints Total: 4 Nonlinear: 0 Nonzeros Total: 12 Nonlinear: 0 Generator Memory Used (K) 35 Elapsed Runtime (hh:mm:ss) 00:00:20

Figure 4. 1: Modèle linéaire non faisable

Pour cela nous allons changer la contrainte de profit comme suivant :

$$\sum_{i=1}^3 (X_i * \text{profit}_i) \leq 30 \quad \forall i$$

Modèle Lingo LP-1-

```

sets:
product:time,ressources,profit,x;
endsets
data:
time      = 40   30   20 ;
ressources = 2    4    3 ;
profit    = 5    8    4 ;
enddata
max=@sum(product(i):profit(i)*x(i));
@sum(product(i):profit(i)*x(i))<=30;
@sum(product(i):ressources(i)*x(i))<=10;
@sum(product(i):time(i)*x(i))<=100;
@for(product(i):@gin(x(i)));

```

Les résultats obtenus sont les suivants :

Solver Status		Variables		Extended solver steps:		
Model Class:	PILP	Total:	3	Total solver iterations: 0		
State:	Global Opt	Nonlinear:	0			
Objective:	21	Integers:	3			
Infeasibility:	0	Constraints		Variable	Value	Reduced Cost
Iterations:	0	Total:	3	TIME (1)	40.00000	0.000000
Extended Solver Status		Nonlinear:	0	TIME (2)	30.00000	0.000000
Solver Type	B-and-B	Nonzeros		TIME (3)	20.00000	0.000000
Best Obj:	21	Total:	9	RESSOURCES (1)	2.000000	0.000000
Obj Bound:	21	Nonlinear:	0	RESSOURCES (2)	4.000000	0.000000
Steps:	0	Generator Memory Used (K)		RESSOURCES (3)	3.000000	0.000000
Active:	0	18		PROFIT (1)	5.000000	0.000000
Elapsed Runtime (hh:mm:ss)		00:00:00		PROFIT (2)	8.000000	0.000000
				PROFIT (3)	4.000000	0.000000
Update Interval: 2		Interrupt Solver		X (1)	1.000000	-5.000000
		Close		X (2)	2.000000	-8.000000
				X (3)	0.000000	-4.000000
				Row	Slack or Surplus	Dual Price
				1	21.00000	1.000000
				2	0.000000	0.000000
				3	0.000000	0.000000

Figure 4. 2: Résultats du modèle Lingo LP-1-

Interprétation 01 :

Le modèle LP nous a donné une solution optimale qu'elle s'agit de produire **un produit de type (1)** et **deux produits de type (2)** ce qui nous permet d'avoir un profit de **21 Millions** da, sans dépasser les 100 unités de temps et utiliser exactement les 10 ressources disponibles.

Ensuite nous avons établi le même problème sous forme d'un modèle goal programming et pour cela nous avons déterminé les objectifs suivants afin de les atteindre :

$$\sum_{i=1}^3 (X_i * profit_i) \leq 30$$

$$\sum_{i=1}^3 (X_i * ressource_i) = 10$$

$$\sum_{i=1}^3 (X_i * temps\ de\ traitement_i) \leq 100$$

Le fait d'avoir plusieurs objectifs à la fois à traiter c'est un peu compliqué puisque on ne peut pas les atteindre à 100% simultanément et ça peut y causer d'avoir des déviations pour chaque objectif, nous allons dans un premier temps appliquer le modèle standard du goal programming les objectifs seront formulés comme suit :

$$\sum_{i=1}^3 (X_i * profit_i) + a = 30$$

$$\sum_{i=1}^3 (X_i * ressource_i) - b + c = 10$$

$$\sum_{i=1}^3 (X_i * temps\ de\ traitement_i) - d = 100$$

La fonction objective sera donc :

$$\text{Min } Z = \sum_{i=1}^3 (a - b + c - d)$$

Tels que : $a \geq 0, b \geq 0, c \geq 0, d \geq 0$.

a : représente la déviation négative de l'objectif g (1) (diminution).

b : représente la déviation positive de l'objectif g (2) (dépassement).

c : représente la déviation négative de l'objectif g (2) (diminution).

d : représente la déviation positive de l'objectif g (3) (dépassement).

Modèle Lingo Goal programming GP-1-

```
sets:
product:time,ressources,profit,x;
endsets
data:
time          = 40   30   20;
ressources    = 2    4    3;
profit        = 5    8    4;
enddata
min=a+b+c+d;
@sum(product(i):profit(i)*x(i))+d=30;
@sum(product(i):ressources(i)*x(i))-b+c=10;
@sum(product(i):time(i)*x(i))-a=100;
@for(product(i):@gin(x(i)));
@gin(a);
@gin(b);
@gin(c);
@gin(d);
```

Les résultats de ce modèle sont dans la figure suivante :

Solver Status Model Class: PIIIP State: Global Opt Objective: 9 Infeasibility: 0 Iterations: 9		Variables Total: 7 Nonlinear: 0 Integers: 7		Global optimal solution found. Objective value: 9.000000 Extended solver steps: 0 Total solver iterations: 9																																																				
Extended Solver Status Solver Type: B-and-B Best Obj: 9 Obj Bound: 9 Steps: 0 Active: 0		Constraints Total: 4 Nonlinear: 0		<table border="1"> <thead> <tr> <th>Variable</th> <th>Value</th> <th>Reduced Cost</th> </tr> </thead> <tbody> <tr><td>A</td><td>9.000000</td><td>1.000000</td></tr> <tr><td>B</td><td>0.000000</td><td>1.000000</td></tr> <tr><td>C</td><td>0.000000</td><td>1.000000</td></tr> <tr><td>D</td><td>0.000000</td><td>1.000000</td></tr> <tr><td>TIME (1)</td><td>40.000000</td><td>0.000000</td></tr> <tr><td>TIME (2)</td><td>30.000000</td><td>0.000000</td></tr> <tr><td>TIME (3)</td><td>20.000000</td><td>0.000000</td></tr> <tr><td>RESSOURCES (1)</td><td>2.000000</td><td>0.000000</td></tr> <tr><td>RESSOURCES (2)</td><td>4.000000</td><td>0.000000</td></tr> <tr><td>RESSOURCES (3)</td><td>3.000000</td><td>0.000000</td></tr> <tr><td>PROFIT (1)</td><td>5.000000</td><td>0.000000</td></tr> <tr><td>PROFIT (2)</td><td>8.000000</td><td>0.000000</td></tr> <tr><td>PROFIT (3)</td><td>4.000000</td><td>0.000000</td></tr> <tr><td>X (1)</td><td>1.000000</td><td>0.000000</td></tr> <tr><td>X (2)</td><td>2.000000</td><td>0.000000</td></tr> <tr><td>X (3)</td><td>0.000000</td><td>0.000000</td></tr> </tbody> </table>		Variable	Value	Reduced Cost	A	9.000000	1.000000	B	0.000000	1.000000	C	0.000000	1.000000	D	0.000000	1.000000	TIME (1)	40.000000	0.000000	TIME (2)	30.000000	0.000000	TIME (3)	20.000000	0.000000	RESSOURCES (1)	2.000000	0.000000	RESSOURCES (2)	4.000000	0.000000	RESSOURCES (3)	3.000000	0.000000	PROFIT (1)	5.000000	0.000000	PROFIT (2)	8.000000	0.000000	PROFIT (3)	4.000000	0.000000	X (1)	1.000000	0.000000	X (2)	2.000000	0.000000	X (3)	0.000000	0.000000
Variable	Value	Reduced Cost																																																						
A	9.000000	1.000000																																																						
B	0.000000	1.000000																																																						
C	0.000000	1.000000																																																						
D	0.000000	1.000000																																																						
TIME (1)	40.000000	0.000000																																																						
TIME (2)	30.000000	0.000000																																																						
TIME (3)	20.000000	0.000000																																																						
RESSOURCES (1)	2.000000	0.000000																																																						
RESSOURCES (2)	4.000000	0.000000																																																						
RESSOURCES (3)	3.000000	0.000000																																																						
PROFIT (1)	5.000000	0.000000																																																						
PROFIT (2)	8.000000	0.000000																																																						
PROFIT (3)	4.000000	0.000000																																																						
X (1)	1.000000	0.000000																																																						
X (2)	2.000000	0.000000																																																						
X (3)	0.000000	0.000000																																																						
Nonzeros Total: 17 Nonlinear: 0		Generator Memory Used (K) 21																																																						
Elapsed Runtime (hh:mm:ss) 00:00:00																																																								
Update Interval: 2		Interrupt Solver		Close																																																				

Figure 4. 3: Résultats du modèle Goal programming-1-

Nous pouvons toujours vérifier les résultats trouvés comme suivant :

$$\sum_{i=1}^3 (X_i * profit_i)$$

$$\sum_{i=1}^3 (X_i * ressource_i)$$

$$\sum_{i=1}^3 (X_i * temps\ de\ traitement_i)$$

Interprétation 02 :

Nous constatons que les deux problèmes nous ont fournis la même solution qui est produire **un produit de type (1) et deux produits de type (2)**, ceci nous donnera un profit optimal de **21 millions** da en consommant les 10 ressources disponibles dans exactement les 100 unités de temps à ne pas dépasser.

Le truc qui se diffères entre les deux modèles c'est que le goal programming va traiter plusieurs objectifs simultanément et elle va trouver un compromis qui permet de générer la meilleure solution pour les trois objectifs (une balance entre les trois objectifs) afin de minimiser le maximum possible l'écart pour chaque objectif.

Si on analyse bien les résultats précédents nous allons voir que la fonction objective par rapport au goal programming nous a donné **A=9** qui est relative au profit, bien évidemment nous avons obtenu **21** comme un profit optimal qui est moins de l'objectif voulu de neuf da, donc l'objectif g (1) est achevé à 70%.

B =0 et **C=0** : c'est-à-dire que nous avons consommé les 10 ressources disponibles ni plus ni moins donc nous avons atteint l'objectif g (2) à 100% aussi.

$D=0$ c'est à dire qu'on n'a pas dépassé les 100 unités de temps c'est-à-dire que l'objectif g (3) est achevé à 100%.

IV.3.1.2 Cas où les deux modèles fournissent des résultats différents :

Dans ce cas nous allons établir le même exemple sauf que nous allons changer le temps opératoire du produit (2) et nous analysons les résultats obtenus :

Une entreprise fabrique trois types de produit, elle veut améliorer ses performances afin d'avoir le meilleur profit possible avec les ressources disponibles et en respectant les temps opératoires, les données sont établies dans le tableau suivant :

Tableau 4. 2 : Exemple-1.1- Goal programming

Type de produit	Produit 1	Produit 2	Produit 3	Goals
Temps de traitement(unit)	40	50	200	≤ 100
Ressources	2	4	3	$= 10$
Profit(Mda)	5	8	4	≤ 30 M

Nous avons essayé de traduire les données sous forme d'un modèle linéaire comme suivant :

$$\text{Max } Z = \sum_{i=1}^3 (X_i * \text{profit}_i)$$

$$\sum_{i=1}^3 (X_i * \text{ressource}_i) \leq 30 \quad \forall i$$

$$\sum_{i=1}^3 (X_i * \text{temps de traitement}_i) \leq 100 \quad \forall i$$

$$\sum_{i=1}^3 (X_i * \text{ressource}_i) \leq 10 \quad \forall i$$

Modèle Lingo linéaire LP-2-

```
sets:
product:time, ressources, profit, x;
endsets
data:
time          = 40   50   20;
ressources    =  2    4    3;
profit        =  5    8    4;
enddata
max=@sum(product(i):profit(i)*x(i));
@sum(product(i):profit(i)*x(i))<=30;
@sum(product(i):ressources(i)*x(i))<=10;
@sum(product(i):time(i)*x(i))<=100;
@for(product(i):@gin(x(i)));
```

Les résultats obtenus sont indiqués dans la figure suivante :

Solver Status Model Class: PIIIP State: Global Opt Objective: 16 Infeasibility: 0 Iterations: 0		Variables Total: 3 Nonlinear: 0 Integers: 3		Variable Value TIME (1) 40.00000 TIME (2) 50.00000 TIME (3) 20.00000 RESSOURCES (1) 2.000000 RESSOURCES (2) 4.000000 RESSOURCES (3) 3.000000 PROFIT (1) 5.000000 PROFIT (2) 8.000000 PROFIT (3) 4.000000 X (1) 0.000000 X (2) 2.000000 X (3) 0.000000	
Extended Solver Status Solver Type: B-and-B Best Obj: 16 Obj Bound: 16 Steps: 0 Active: 0		Constraints Total: 4 Nonlinear: 0		Nonzeros Total: 12 Nonlinear: 0	
Generator Memory Used (K): 20 Elapsed Runtime (hh:mm:ss): 00:00:00		Row Slack or Surplus 1 16.00000 2 14.00000 3 2.000000 4 0.000000			

Figure 4. 4: Résultats du modèle LP-2-

Interprétation 03 :

Dans ce cas nous avons obtenu un profit optimal égal à **16 da** par la production de **deux produits de type (2)** ce qui va engendrer la consommation de **8/10 ressources disponibles** dans 100 unités de temps exactement.

Nous avons refait le même problème sous forme d'un modèle goal programming :

$$\sum_{i=1}^3 (X_i * profit_i) + a = 30$$

$$\sum_{i=1}^3 (X_i * ressource_i) - b + c = 10$$

$$\sum_{i=1}^3 (X_i * temps\ de\ traitement_i) - d = 100$$

$$\text{Min } Z = \sum_{i=1}^3 (a - b + c - d)$$

Tels que : $a \geq 0, b \geq 0, c \geq 0, d \geq 0$.

Modèle Goal programming GP-2-

```
sets:
product:time,ressources,profit,x;
endsets
data:
time      = 40   50   20;
ressources = 2    4    3;
profit    = 5    8    4;
enddata
min=a+b+c+d;
@sum(product(i):profit(i)*x(i))+d=30;
@sum(product(i):ressources(i)*x(i))-b+c=10;
@sum(product(i):time(i)*x(i))-a=100;
@for(product(i):@gin(x(i)));
```

```
@gin (a) ;
@gin (b) ;
@gin (c) ;
@gin (d) ;
```

La figure suivante montre les résultats obtenus :

Solver Status		Variables		Total solver iterations: 23		
Model Class:	PILP	Total:	7	Variable	Value	Reduced Cost
State:	Global Opt	Nonlinear:	0	A	13.000000	1.000000
Objective:	14	Integers:	7	B	1.000000	1.000000
Infeasibility:	0	Constraints		C	0.000000	1.000000
Iterations:	23	Total:	4	D	0.000000	1.000000
		Nonlinear:	0	TIME (1)	40.000000	0.000000
Extended Solver Status		Nonzeros		TIME (2)	50.000000	0.000000
Solver Type	B-and-B	Total:	17	TIME (3)	20.000000	0.000000
Best Obj:	14	Nonlinear:	0	RESSOURCES (1)	2.000000	0.000000
Obj Bound:	14	Generator Memory Used (K)		RESSOURCES (2)	4.000000	0.000000
Steps:	0	21		RESSOURCES (3)	3.000000	0.000000
Active:	0	Elapsed Runtime (hh:mm:ss)		PROFIT (1)	5.000000	0.000000
		00 : 00 : 00		PROFIT (2)	8.000000	0.000000
				PROFIT (3)	4.000000	0.000000
				X (1)	1.000000	0.000000
				X (2)	0.000000	0.000000
				X (3)	3.000000	0.000000

Figure 4. 5: Résultats du modèle Goal programming GP-2-

Interprétation 04 :

Dans ce cas nous avons obtenus une solution optimale égale à **14** représente la somme des déviations optimales obtenus par la production **d'un seul produit de type (1) et 3 produits de type (3)** ce qui permet d'avoir un profit égal à **17** moins de **A=13** unités de déviation de l'objectif voulu qui est un profit de **30**. L'obtention de ce profit implique la consommation de 11 ressources ce qui dépassent les ressources disponibles par une ressource **B=1** dans exactement les **100 unités de temps D=0**.

IV.3.2 Exemple 2 problème de minimisation :

Dans cet exemple nous allons garder les mêmes données sauf qu'on va considérer les profits comme des coûts de production et nous allons déterminer la fonction objective suivante :

$$\text{Min } Z = \sum_{i=1}^3 (X_i * \text{cost}_i)$$

Une entreprise fabrique trois types de produit, elle veut améliorer ses performances afin minimiser les coûts de production avec les ressources disponibles et en respectant les temps opératoires, les données sont établies dans le tableau suivant :

Tableau 4. 3: Exemple -2- Goal programming

Type de produit	Produit 1	Produit 2	Produit 3	Goals
Temps de traitement(unit)	40	30	200	<=100
Ressources	2	4	3	=10
Cost (M. da)	5	8	4	<=30 M

Nous avons essayé de traduire les données sous forme d'un modèle linéaire comme suit :

$$\text{Min } Z = \sum_{i=1}^3 (X_i * \text{cost}_i)$$

$$\sum_{i=1}^3 (X_i * \text{temps de traitement}_i) \leq 100 \quad \forall i$$

$$\sum_{i=1}^3 (X_i * \text{ressource}_i) \leq 10 \quad \forall i$$

$X_i \geq 0$: variable de décision indique la quantité produite du produit(i)

Dans un premier temps nous allons traduire ce problème en un modèle linéaire comme suivant :

Modèle linéaire LP-3-

```
sets:
product:time,ressources,cost,x;
endsets
data:
time      = 40   30   20;
ressources = 2    4    3;
cost      = 5    8    4;
enddata
max=@sum(product(i):cost(i)*x(i));
@sum(product(i):cost(i)*x(i))<=30;
@sum(product(i):ressources(i)*x(i))<=10;
@sum(product(i):time(i)*x(i))<=100;
@for(product(i):@gin(x(i)));
```

Scénario 0 1 :

Si on exécute ce modèle exactement tel qu'il est nous allons avoir zéro comme valeur optimale puisque le solver va prendre zéro de chaque produit du coup les dépenses, la consommation des ressources, et les temps opératoires seront nuls et c'est tout affait juste puisque nous voulons à minimiser les dépenses. Pour rendre le problème plus réaliste il fallait faire des petits changements sur les contraintes comme suivant :

Modèle Linéaire LP-3.1-

```
sets:
product:time,ressources,cost,x;
endsets
data:
time      = 40   30   20;
ressources= 2    4    3;
cost      = 5    8    4;
enddata
min=@sum(product(i):cost(i)*x(i));
@sum(product(i):cost(i)*x(i))<=30;
@sum(product(i):ressources(i)*x(i))=10;
```

```
@sum(product(i):time(i)*x(i))=100;
@for(product(i):@gin(x(i)));
```

Les résultats obtenus de ce modèle sont :

Solver Status		Variables		Variable		Value	
Model Class:	PIIP	Total:	3	TIME (1)			40.00000
State:	Global Opt	Nonlinear:	0	TIME (2)			30.00000
Objective:	21	Integers:	3	TIME (3)			20.00000
Infeasibility:	0	Constraints		RESSOURCES (1)			2.000000
Iterations:	0	Total:	4	RESSOURCES (2)			4.000000
Extended Solver Status		Nonlinear:	0	RESSOURCES (3)			3.000000
Solver Type	B-and-B	Nonzeros		COST (1)			5.000000
Best Obj:	21	Total:	12	COST (2)			8.000000
Obj Bound:	21	Nonlinear:	0	COST (3)			4.000000
Steps:	0	Generator Memory Used (K)		X (1)			1.000000
Active:	0	20		X (2)			2.000000
		Elapsed Runtime (hh:mm:ss)		X (3)			0.000000
		00:00:00		Row		Slack or Surplus	
				1		21.00000	
				2		9.000000	
				3		0.000000	
				4		0.000000	

Figure 4. 6: Résultats du modèle Linéaire LP-3.1-

Interprétation 05 :

La solution optimale donc pour un cout global qui ne doit pas dépasser 30 M est égale : **21** prenant **un produit de type (1)** et **deux produits de type (2)** et c'est exactement la même solution pour le problème de maximisation de profit. En principe nous devons avoir moins sauf que nous avons imposé dans les contraintes que la somme des temps opératoires doit être égale à **100** et la somme des ressources utilisées doit être égale à **10**, c'est-à-dire quoi le solver va trouver le cout minimal qui permet de consommer 10 ressources dans exactement 100 unités de temps.

Scénario 02 :

Cette fois-ci nous avons changé la contrainte des temps, les résultats devenus comme suit :

Modèle Linéaire LP-3.2-

```
sets:
product:time,ressources,cost,x;
endsets
data:
time          = 40  30  20;
ressources=  2   4   3;
cost          = 5   8   4;
enddata
min=@sum(product(i):cost(i)*x(i));
@sum(product(i):cost(i)*x(i))<=30;
@sum(product(i):ressources(i)*x(i))=10;
@sum(product(i):time(i)*x(i))<=100;
@for(product(i):@gin(x(i)));
```

La figure suivante montre les résultats obtenus :

Solver Status Model Class: PILP State: Global Opt Objective: 16 Infeasibility: 0 Iterations: 0		Variables Total: 3 Nonlinear: 0 Integers: 3		Constraints Total: 4 Nonlinear: 0																																					
Extended Solver Status Solver Type: B-and-B Best Obj: 16 Obj Bound: 16 Steps: 0 Active: 0		Nonzeros Total: 12 Nonlinear: 0		Generator Memory Used (K) 20																																					
Elapsed Runtime (hh:mm:ss) 00:00:00		<table border="1"> <thead> <tr> <th>Variable</th> <th>Value</th> </tr> </thead> <tbody> <tr><td>TIME (1)</td><td>40.00000</td></tr> <tr><td>TIME (2)</td><td>30.00000</td></tr> <tr><td>TIME (3)</td><td>20.00000</td></tr> <tr><td>RESSOURCES (1)</td><td>2.000000</td></tr> <tr><td>RESSOURCES (2)</td><td>4.000000</td></tr> <tr><td>RESSOURCES (3)</td><td>3.000000</td></tr> <tr><td>COST (1)</td><td>5.000000</td></tr> <tr><td>COST (2)</td><td>8.000000</td></tr> <tr><td>COST (3)</td><td>4.000000</td></tr> <tr><td>X (1)</td><td>0.000000</td></tr> <tr><td>X (2)</td><td>1.000000</td></tr> <tr><td>X (3)</td><td>2.000000</td></tr> </tbody> </table> <table border="1"> <thead> <tr> <th>Row</th> <th>Slack or Surplus</th> </tr> </thead> <tbody> <tr><td>1</td><td>16.00000</td></tr> <tr><td>2</td><td>14.00000</td></tr> <tr><td>3</td><td>0.000000</td></tr> <tr><td>4</td><td>30.00000</td></tr> </tbody> </table>				Variable	Value	TIME (1)	40.00000	TIME (2)	30.00000	TIME (3)	20.00000	RESSOURCES (1)	2.000000	RESSOURCES (2)	4.000000	RESSOURCES (3)	3.000000	COST (1)	5.000000	COST (2)	8.000000	COST (3)	4.000000	X (1)	0.000000	X (2)	1.000000	X (3)	2.000000	Row	Slack or Surplus	1	16.00000	2	14.00000	3	0.000000	4	30.00000
Variable	Value																																								
TIME (1)	40.00000																																								
TIME (2)	30.00000																																								
TIME (3)	20.00000																																								
RESSOURCES (1)	2.000000																																								
RESSOURCES (2)	4.000000																																								
RESSOURCES (3)	3.000000																																								
COST (1)	5.000000																																								
COST (2)	8.000000																																								
COST (3)	4.000000																																								
X (1)	0.000000																																								
X (2)	1.000000																																								
X (3)	2.000000																																								
Row	Slack or Surplus																																								
1	16.00000																																								
2	14.00000																																								
3	0.000000																																								
4	30.00000																																								

Figure 4. 7: Résultats du modèle Linéaire LP-3.2-

Interprétation 06 :

Nous avons pris dans ce cas pris la somme des temps opératoires **inférieur ou égal à 100** : donc Lingo a fourni une solution qui consiste à prendre **un produit de type (2) et un produit de type (3)** ce qui nous donnera **10** ressources consommées exactement dans **70 unités de temps** en mobilisant un cout global de **16 M da**.

Scénario 03 :

Le même problème principal sauf que nous allons changer la contrainte des ressources cette fois ci :

Modèle Linéaire LP -3.3-

```
sets:
product:time, ressources, cost, x;
endsets
data:
time          = 40  30  20;
ressources=  2    4    3;
cost          =  5    8    4;
enddata
min=@sum(product(i):cost(i)*x(i));
@sum(product(i):cost(i)*x(i))<=30;
@sum(product(i):ressources(i)*x(i))<=10;
@sum(product(i):time(i)*x(i))=100;
@for(product(i):@gin(x(i)));
```

Les résultats sont comme suivant :

Solver Status Model Class: PILP State: Global Opt Objective: 14 Infeasibility: 0 Iterations: 0		Variables Total: 3 Nonlinear: 0 Integers: 3		Constraints Total: 4 Nonlinear: 0		Nonzeros Total: 12 Nonlinear: 0		Variable Value TIME (1) 40.00000 TIME (2) 30.00000 TIME (3) 20.00000 RESSOURCES (1) 2.000000 RESSOURCES (2) 4.000000 RESSOURCES (3) 3.000000 COST (1) 5.000000 COST (2) 8.000000 COST (3) 4.000000 X (1) 2.000000 X (2) 0.000000 X (3) 1.000000	
Extended Solver Status Solver Type B-and-B Best Obj: 14 Obj Bound: 14 Steps: 0 Active: 0		Generator Memory Used (K) 20		Elapsed Runtime (hh:mm:ss) 00:00:00		Row Slack or Surplus 1 14.00000 2 16.00000 3 3.000000 4 0.000000			

Figure 4. 8: Résultats du modèle Linéaire LP -3.3-

Interprétation 07 :

Comme il est mentionné dans la figure dans ce cas-là vaut mieux produire **deux produits de type (1) et un produit de type (3)** ce qui permet de consommer **7 ressources dans 100 unités** de temps et qui va engendrer un cout global de **14 M da**.

Donc pour améliorer la solution il faut jouer sur les autres contraintes

Ensuite nous allons traduire ce problème en goal programming :

Pratiquement le modèle reste presque le même que le modèle goal programming de maximisation des profits dans le premier exemple sauf que nous allons remplacer le « **profit** » par « **cost** » :

Modèle Goal programming GP-3-

```
sets:
product:time, ressources, cost, x;
endsets
data:
time      = 40  30  20;
ressources= 2   4   3;
cost      = 5   8   4;
enddata
min=a+b+c+d;
@sum(product(i):cost(i)*x(i))+a=30;
@sum(product(i):ressources(i)*x(i))-b+c=10;
@sum(product(i):time(i)*x(i))-d=100;
@for(product(i):@gin(x(i)));
@gin(a);
@gin(b);
@gin(c);
@gin(d);
```

Le modèle établi va chercher La valeur minimale pour laquelle nous allons rapprocher le maximum possible vers :

- 30 M da comme un gout global maximal.
- 100 unités de temps opératoires.

- 10 ressources à consommer.

La figure suivante montre les résultats obtenus :

Solver Status Model Class: PILP State: Global Opt Objective: 9 Infeasibility: 0 Iterations: 9		Variables Total: 7 Nonlinear: 0 Integers: 7	<table border="1"> <thead> <tr> <th>Variable</th> <th>Value</th> </tr> </thead> <tbody> <tr><td>A</td><td>9.000000</td></tr> <tr><td>B</td><td>0.000000</td></tr> <tr><td>C</td><td>0.000000</td></tr> <tr><td>D</td><td>0.000000</td></tr> <tr><td>TIME (1)</td><td>40.000000</td></tr> <tr><td>TIME (2)</td><td>30.000000</td></tr> <tr><td>TIME (3)</td><td>20.000000</td></tr> <tr><td>RESSOURCES (1)</td><td>2.000000</td></tr> <tr><td>RESSOURCES (2)</td><td>4.000000</td></tr> <tr><td>RESSOURCES (3)</td><td>3.000000</td></tr> <tr><td>COST (1)</td><td>5.000000</td></tr> <tr><td>COST (2)</td><td>8.000000</td></tr> <tr><td>COST (3)</td><td>4.000000</td></tr> <tr><td>X (1)</td><td>1.000000</td></tr> <tr><td>X (2)</td><td>2.000000</td></tr> <tr><td>X (3)</td><td>0.000000</td></tr> </tbody> </table>	Variable	Value	A	9.000000	B	0.000000	C	0.000000	D	0.000000	TIME (1)	40.000000	TIME (2)	30.000000	TIME (3)	20.000000	RESSOURCES (1)	2.000000	RESSOURCES (2)	4.000000	RESSOURCES (3)	3.000000	COST (1)	5.000000	COST (2)	8.000000	COST (3)	4.000000	X (1)	1.000000	X (2)	2.000000	X (3)	0.000000
Variable	Value																																				
A	9.000000																																				
B	0.000000																																				
C	0.000000																																				
D	0.000000																																				
TIME (1)	40.000000																																				
TIME (2)	30.000000																																				
TIME (3)	20.000000																																				
RESSOURCES (1)	2.000000																																				
RESSOURCES (2)	4.000000																																				
RESSOURCES (3)	3.000000																																				
COST (1)	5.000000																																				
COST (2)	8.000000																																				
COST (3)	4.000000																																				
X (1)	1.000000																																				
X (2)	2.000000																																				
X (3)	0.000000																																				
Extended Solver Status Solver Type: B-and-B Best Obj: 9 Obj Bound: 9 Steps: 0 Active: 0		Constraints Total: 4 Nonlinear: 0																																			
		Nonzeros Total: 17 Nonlinear: 0																																			
		Generator Memory Used (K): 21																																			
		Elapsed Runtime (hh:mm:ss): 00:00:01																																			

Figure 4. 9: Résultats du modèle GP-3-

Interprétation 08 :

Par la prise de d'un produit de type (1) et deux de type (2) nous avons obtenu un cout global égal à **21** écartés de l'objectif g (1) par (**A= 9**) par contre nous allons consommer exactement **10 ressources(B=0, c=0)** dans exactement **100 unités de temps (D=0)**.

Dans cet exemple on peut clairement voir la différence entre le modèle linéaire et le modèle Goal programming :

Dans les 3 scénarios de modèle linéaire le modèle établi pour chaque scénario consiste à trouver un cout minimal qui respecte les contraintes citées.

Par contre le modèle goal programming a trouvé une solution indépendamment de notre objectif de tels sorte va trouver un point d'équilibre permettant d'avoir la meilleure solution pour chaque objectif et qui se rapproche le maximum possible de l'objectif voulu.

Le truc qui se diffère entre les deux modèles c'est que le goal programming va traiter plusieurs objectifs simultanément et elle va trouver un compromis qui permet de générer la meilleure solution pour les trois objectifs (une balance entre les trois objectifs) afin de minimiser le maximum possible l'écart pour chaque objectif.

IV.3.3 Attribution des poids (priorités) aux objectifs :

Dans ce cas nous allons prendre un exemple ou nous attribuons des poids aux objectifs et voir comment la méthode goal programming va traiter ce problème.

Nous allons prendre le même exemple de maximisation de profit dans un premier temps sans affectation des poids puis on rajoute les poids et on compare les résultats obtenus :

Tableau 4. 4: Exemple-3- Goal programming avec poids

Type de produit	Produit 1	Produit 2	Produit 3	Goals
Temps de traitement(unit)	40	50	200	<=100
Ressources	2	4	3	=10
Profit(Mda)	5	8	4	<=30 M

$$\sum_{i=1}^3 (X_i * profit_i) + a = 30$$

$$\sum_{i=1}^3 (X_i * ressource_i) - b + c = 10$$

$$\sum_{i=1}^3 (X_i * temps\ de\ traitement_i) - d = 100$$

La fonction objective sera donc :

$$Min Z = \sum_{i=1}^3 (a - b + c - d)$$

Tels que : $a \geq 0, b \geq 0, c \geq 0, d \geq 0$.

Le modèle Lingo de ce problème est comme suivant :

Modèle GP sans poids

```
sets:
product:time,ressources,profit,x;
endsets
data:
time          = 40   50   20;
ressources    = 2    4    3;
profit        = 5    8    4;
enddata
min=a+b+c+d;
@sum(product(i):profit(i)*x(i))+a=30;
@sum(product(i):ressources(i)*x(i))-b+c=10;
@sum(product(i):time(i)*x(i)) -d=100;
@for(product(i):@gin(x(i)));
@gin(a);
@gin(b);
@gin(c);
@gin(d);
```

Les résultats obtenus pour ce modèle sont :

Solver Status Model Class: PIIIP State: Global Opt Objective: 14 Infeasibility: 0 Iterations: 23		Variables Total: 7 Nonlinear: 0 Integers: 7		<table border="1"> <thead> <tr> <th>Variable</th> <th>Value</th> </tr> </thead> <tbody> <tr><td>A</td><td>13.000000</td></tr> <tr><td>B</td><td>1.000000</td></tr> <tr><td>C</td><td>0.000000</td></tr> <tr><td>D</td><td>0.000000</td></tr> <tr><td>TIME (1)</td><td>40.000000</td></tr> <tr><td>TIME (2)</td><td>50.000000</td></tr> <tr><td>TIME (3)</td><td>20.000000</td></tr> <tr><td>RESSOURCES (1)</td><td>2.000000</td></tr> <tr><td>RESSOURCES (2)</td><td>4.000000</td></tr> <tr><td>RESSOURCES (3)</td><td>3.000000</td></tr> <tr><td>PROFIT (1)</td><td>5.000000</td></tr> <tr><td>PROFIT (2)</td><td>8.000000</td></tr> <tr><td>PROFIT (3)</td><td>4.000000</td></tr> <tr><td>X (1)</td><td>1.000000</td></tr> <tr><td>X (2)</td><td>0.000000</td></tr> <tr><td>X (3)</td><td>3.000000</td></tr> </tbody> </table>	Variable	Value	A	13.000000	B	1.000000	C	0.000000	D	0.000000	TIME (1)	40.000000	TIME (2)	50.000000	TIME (3)	20.000000	RESSOURCES (1)	2.000000	RESSOURCES (2)	4.000000	RESSOURCES (3)	3.000000	PROFIT (1)	5.000000	PROFIT (2)	8.000000	PROFIT (3)	4.000000	X (1)	1.000000	X (2)	0.000000	X (3)	3.000000
Variable	Value																																					
A	13.000000																																					
B	1.000000																																					
C	0.000000																																					
D	0.000000																																					
TIME (1)	40.000000																																					
TIME (2)	50.000000																																					
TIME (3)	20.000000																																					
RESSOURCES (1)	2.000000																																					
RESSOURCES (2)	4.000000																																					
RESSOURCES (3)	3.000000																																					
PROFIT (1)	5.000000																																					
PROFIT (2)	8.000000																																					
PROFIT (3)	4.000000																																					
X (1)	1.000000																																					
X (2)	0.000000																																					
X (3)	3.000000																																					
Extended Solver Status Solver Type: B-and-B Best Obj: 14 Obj Bound: 14 Steps: 0 Active: 0		Constraints Total: 4 Nonlinear: 0																																				
		Nonzeros Total: 17 Nonlinear: 0																																				
		Generator Memory Used (K) 21																																				
		Elapsed Runtime (hh:mm:ss) 00 : 00 : 00																																				

Figure 4. 10: Résultats du modèle GP sans poids

Cette fois-ci nous allons affecter à chaque objectif un poids selon son importance :

Tableau 4. 5: Les poids affectés pour chaque objectif

Objectifs	Profit	Ressources	Temps
Poids	2	1	1

Le modèle sera comme suivant :

Modèle GP avec poids

```

sets:
product:time,ressources,profit,x;
endsets
data:
time      = 40  50  20;
ressources = 2   4   3;
profit    = 5   8   4;
enddata
min=2*a+b+c+d;
@sum(product(i):profit(i)*x(i))+a=30;
@sum(product(i):ressources(i)*x(i))-b+c=10;
@sum(product(i):time(i)*x(i))-d=100;
@for(product(i):@gin(x(i)));
@gin(a);
@gin(b);
@gin(c);
@gin(d);

```

La figure suivante montre les résultats obtenus :

Solver Status Model Class: PILP State: Global Opt Objective: 25 Infeasibility: 0 Iterations: 2		Variables Total: 7 Nonlinear: 0 Integers: 7		A 10.00000 B 5.000000 C 0.000000 D 0.000000	
Extended Solver Status Solver Type: B-and-B Best Obj: 25 Obj Bound: 25 Steps: 0 Active: 0		Constraints Total: 4 Nonlinear: 0		TIME (1) 40.00000 TIME (2) 50.00000 TIME (3) 20.00000 RESSOURCES (1) 2.000000 RESSOURCES (2) 4.000000 RESSOURCES (3) 3.000000 PROFIT (1) 5.000000 PROFIT (2) 8.000000 PROFIT (3) 4.000000	
		Nonzeros Total: 17 Nonlinear: 0		X (1) 0.000000 X (2) 0.000000 X (3) 5.000000	
		Generator Memory Used (K) 21		Row Slack or Surplus 1 25.00000 2 0.000000	
		Elapsed Runtime (hh:mm:ss) 00 : 00 : 00			

Figure 4. 11: Résultats du modèle GP avec poids

Interprétation 09 :

Nous constatons un changement dans les résultats, tels que le global optimal est devenu **25** au lieu de **14** ce qui représente la somme des déviations.

Si on compare bien les résultats des deux modèles : dans le modèle sans poids nous avons obtenu **A=13** le paramètre représentant la déviation par rapport à l'objectif de profit et **B=1** qui représente la déviation positive par rapport à l'objectif des ressources, par contre dans le modèle ou on a attribué **un poids=2 pour le paramètre A** c'est-à-dire que l'objectif de profit est deux fois plus important par rapport à les deux autres objectifs nous avons obtenu **B=5** mais la déviation des profits a été diminué **A=10**.

IV.4 Application de la méthode goal programming au cas d'étude :

Dans le chapitre précédent nous avons fait une projection du problème p-médian et le problème max covering et nous avons vu des situations ou le solver n'a pas pu trouver une solution faisable.

L'une des propriétés de l'approche goal programming c'est de trouver une solution pour les modèles non faisables de manière on fixe un objectif à atteindre, le goal programming va rapprocher le maximum possible à cet objectif par l'obtention de minimum des déviations possibles.

IV.4.1 Problématique 01 : Modèle p-médian :

IV.4.2 Problème de non faisabilité à cause de capacité insuffisante :

Nous allons donc dans un premier temps appliquer la méthode GP sur le premier modèle non faisable qui concerne la capacité :

Le problème donc est apparu quand nous avons mis la capacité des points de vente inférieure à la demande globale du coup le fait d'avoir une capacité insuffisante et une contrainte impose d'affecter toutes les zones d'autre part c'était impossible de trouver une solution pour ce faire nous avons fait appelle à la méthode goal programming :

Nous avons donc ajouté une contrainte pour la fonction objective qui doit dépasser : **13000** et nous avons ajouté une contrainte qui impose que la somme globale des demandes ne doit pas dépasser la capacité totale qui est égale à **9000**, puisque nous avons imposé d'affecter toutes les contraintes et nous savions que la demande globale dépasse les 9000 litres, c'était clair que le modèle va être non faisable.

Le modèle Lingo pour ce raisonnement est comme suivant :

Modèle Lingo GP P-médian problème de capacité

```
sets:
pointv/1..5/:y,cap;
zone/1..16/:dem;
pointzone(pointv,zone):dist,x;
endsets
data:
costKm = 1;
dem= 1000 400 1060 400 740 800 400 800 600 400 500 400 400 400 1000 800;
cap= 1000 2000 2000 2000 2000;

dist=
4.  5.  0.  1.  2.  3.  4.  5.  1.  0.  2.  4.  4.  8.  4.  5.1
26  30  80  21  14  20  15  20  20  94  37  29  60  31  36  0
2.  3.  3.  1.  0.  1.  2.  3.  2.  1.  2.  2.  2.  6.  2.  3.9
63  51  01  16  68  28  14  13  93  96  94  08  42  10  60
1.  1.  5.  3.  3.  2.  1.  0.  5.  4.  2.  1.  1.  3.  3.  2.6
94  70  94  96  04  02  14  63  91  93  95  00  64  76  01  0
4.  4.  8.  7.  6.  5.  4.  3.  8.  7.  5.  3.  3.  2.  6.  3.8
87  13  88  00  07  01  23  38  70  71  76  86  65  70  67  2
3.  3.  4.  3.  2.  2.  2.  3.  4.  3.  1.  0.  1.  4.  4.  0.9
77  29  85  32  87  61  76  20  37  48  62  64  09  83  27  5;
enddata
min=a+b;
@sum(pointv(i):@sum(zone(j):dem(j)*dist(i,j)*costKm*x(i,j)))>=13000;

!(1.1);@sum(zone(j):dem(j))-a+b=9000;
!(1.2);@sum(pointv(i):y(i))<=5;
!(1.3);@for(zone(j):@sum(pointv(i):x(i,j))=1);
!(1.4);@for(pointv(i):@for(zone(j):x(i,j)<=y(i)));
!(1.5);@for(pointv(i):@sum(zone(j):x(i,j))>=y(i));
@for(pointv:@bin(y));
@for(pointv(i):@for(zone(j):@bin(x(i,j))));
```

(1.1) : la somme des demandes ne doit pas dépasser la capacité du point de vente.

(1.2) : Nombre de point de vente à ouvrir.

(1.3) : Une zone doit être affectée à un seul point de vente.

(1.4) : Un client doit être servi par un point de vente ouvert.

(1.5) : Un point de vente s'il est ouvert, il doit servir au moins une zone.

Les résultats pour ce modèle donc sont représentés par la figure suivante :

Solver Status Model Class: MILP State: Global Opt Objective: 1100 Infeasibility: 0 Iterations: 0		Variables Total: 87 Nonlinear: 0 Integers: 85		COSTEM A 1100.000 B 0.000000 Y (1) 1.000000 Y (2) 1.000000 Y (3) 1.000000 Y (4) 1.000000 Y (5) 1.000000	
Extended Solver Status Solver Type: B-and-B Best Obj: 1100 Obj Bound: 1100 Steps: 0 Active: 0		Constraints Total: 105 Nonlinear: 0		CAP (1) 1000.000 CAP (2) 2000.000 CAP (3) 2000.000 CAP (4) 2000.000 CAP (5) 2000.000 DEM (1) 1000.000 DEM (2) 400.0000 DEM (3) 1060.000 DEM (4) 400.0000 DEM (5) 740.0000 DEM (6) 800.0000 DEM (7) 400.0000 DEM (8) 800.0000	
		Nonzeros Total: 414 Nonlinear: 0			
		Generator Memory Used (K) 83			
		Elapsed Runtime (hh:mm:ss) 00:00:00			

Figure 4. 12: Résultats du modèle GP P-médian problème de capacité

IV.4.2.1 Interprétation :

Comme nous avons déjà dit le problème de non faisabilité c'était lié à la non suffisante capacité par rapport aux demandes, dans le modèle goal programming nous avons autorisé à dévier la capacité selon le besoin que ce soit déviation positive représentée dans la contrainte par (-a) c'est-à-dire qu'il existe un manque de capacité et nous avons consommé plus ou bien une déviation négative représentée par (+b) c'est-à-dire que nous avons consommé moins de la valeur nominale.

Les résultats apparus montrent que la fonction objective est égale à **1100** qui représente la valeur manquante de capacité, par la suite nous avons obtenus **A=1100 et B=0** c'est-à-dire qu'il faut rajouter 1100 litres pour couvrir le manque de capacité, si on calcule la consommation globale par rapport aux demandes nous allons obtenir 10100 ce qui valide nos résultats qu'il manque 1100 litres dans la capacité global des points de vente.

Nous tenons à indiquer aussi qu'il ne faut pas avoir les deux déviations positives et négatives simultanément, il suffit d'apparaître l'une l'autre va être automatiquement nulle puisqu'on ne peut jamais avoir un manque et un dépassement au même temps pour le même objectif.

IV.4.3 Problème de non faisabilité à cause de distance :

Nous allons dans ce cas résoudre le problème de non faisabilité qui concerne la contrainte de distance vu dans le chapitre précédent là où nous avons imposé que la distance globale de la chaîne de distribution doit être égale à **20 km** le solveur n'a pas pu trouver une solution du coup il a indiqué que le modèle est non faisable.

En réalité on ne sait pas si la distance globale du réseau conçu dépasse les 20 km ou pas pour cela nous avons rajouté à la contrainte de distance une variable (-a) sert à calculer l'écart dans le cas de dépassement et une variable (b) pour calculer l'écart dans le cas inverse.

En utilisant l'approche goal programming nous allons autoriser la déviation de la contrainte de distance et on constatera l'écart obtenu par la fonction objective qui va être la minimisation des déviations.

Le modèle Lingo sera donc :

Modèle Lingo GP P-médian problème de distance

```
sets:
pointv/1..5/:y,cap;
zone/1..16/:dem;
pointzone(pointv,zone):dist,x;
endsets
data:
costKm = 1;
dem= 1000 400 1060 400 740 800 400 800 600 400 500 400 400 400 1000 800;
cap= 3000 3000 3000 3000 3000;
dist =
  4.  5.  0.  1.  2.  3.  4.  5.  1.  0.  2.  4.  4.  8.  4.  5.1
  26  30  80  21  14  20  15  20  20  94  37  29  60  31  36  0
  2.  3.  3.  1.  0.  1.  2.  3.  2.  1.  2.  2.  2.  6.  2.  3.9
  63  51  01  16  68  28  14  13  93  96  94  08  42  10  60
  1.  1.  5.  3.  3.  2.  1.  0.  5.  4.  2.  1.  1.  3.  3.  2.6
  94  70  94  96  04  02  14  63  91  93  95  00  64  76  01  0
  4.  4.  8.  7.  6.  5.  4.  3.  8.  7.  5.  3.  3.  2.  6.  3.8
  87  13  88  00  07  01  23  38  70  71  76  86  65  70  67  2
  3.  3.  4.  3.  2.  2.  2.  3.  4.  3.  1.  0.  1.  4.  4.  0.9
  77  29  85  32  87  61  76  20  37  48  62  64  09  83  27  5;

enddata
min=a+b;
@sum(pointv(i):@sum(zone(j):dem(j)*dist(i,j)*costKm*x(i,j)))>1300;
@sum(pointv(i):@sum(zone(j):dist(i,j)*x(i,j)))-a+b=20;
!(1.1); @for(pointv(i):@sum(zone(j):dem(j)*x(i,j))<=cap(i));
!(1.2); @sum(pointv(i):y(i))<=5;
!(1.3); @for(zone(j):@sum(pointv(i):x(i,j))=1);
!(1.4); @for(pointv(i):@for(zone(j):x(i,j)<=y(i)));
!(1.5); @for(pointv(i):@sum(zone(j):x(i,j))>=y(i));
@for(pointv:@bin(y));
@for(pointv(i):@for(zone(j):@bin(x(i,j))));
```

- (1.1)** : la somme des demandes ne doit pas dépasser la capacité du point de vente.
(1.2) : Nombre de point de vente à ouvrir.
(1.3) : Une zone doit être affectée à un seul point de vente.
(1.4) : Un client doit être servi par un point de vente ouvert.
(1.5) : Un point de vente s'il est ouvert, il doit servir au moins une zone.

Les résultats pour ce modèle sont les suivants :

Solver Status:		Variables			
Model Class:	MILP	Total:	87	COSTKM	1.000000
State:	Global Opt	Nonlinear:	0	A	1.070000
Objective:	1.07	Integers:	85	B	0.000000
Infeasibility:	1.33227e-015	Constraints		Y (1)	1.000000
Iterations:	16	Total:	110	Y (2)	1.000000
		Nonlinear:	0	Y (3)	1.000000
		Nonzeros		Y (4)	1.000000
Extended Solver Status:		Total:	574	Y (5)	1.000000
Solver Type:	B-and-B	Nonlinear:	0	CAP (1)	3000.000
Best Obj:	1.07	Generator Memory Used (K)		CAP (2)	3000.000
Obj Bound:	1.07	93		CAP (3)	3000.000
Steps:	0	Elapsed Runtime (hh:mm:ss)		CAP (4)	3000.000
Active:	0	00:00:00		CAP (5)	3000.000
				DEM (1)	1000.000
				DEM (2)	400.0000
				DEM (3)	1060.000
				DEM (4)	400.0000
				DEM (5)	740.0000
				DEM (6)	800.0000

Figure 4. 13: Résultats du modèle p-médian GP problème de distance

IV.4.3.1 Interprétation :

Nous avons précédemment imposé que la distance globale du réseau de distribution doit être égale à 20 km ce qui était impossible à l'ombre des autres contraintes par conséquent le modèle était non faisable. Nous avons donc introduit la notion du goal programming sur cette contrainte de telle sorte on autorise la déviation positive et négative et on constate le résultat obtenu.

Nous avons par la suite obtenu $A=1.07$ c'est-à-dire que la distance globale parcourue dans le réseau est supérieure à la distance exigée, donc nous avons dépassé la distance exigée, bien sûr que $B=0$ puisqu'on ne peut pas avoir les deux déviations au même temps.

IV.4.4 Problème de non faisabilité à cause de nombre de point de vente :

Dans ce cas nous allons résoudre le problème de la non faisabilité qui concerne le nombre de point de vente, là où nous avons imposé d'ouvrir que quatre points de vente parmi les cinq disponibles, du coup ce n'était pas possible d'affecter toutes les zones contradiction avec la contrainte qui impose d'affecter toutes les zones du coup ce modèle est devenu infaisable.

Nous allons donc autoriser la déviation d'augmenter le nombre de point de vente et nous allons constater la solution obtenue :

Dans le cas où nous avons installé quatre points de vente la solution optimale était **13881.2** si on augmente les points de vente à cinq la solution optimale devient : **13457.2**

À partir d'ici nous avons ajouté une contrainte qui impose que la fonction objective ne doit pas dépasser la valeur **13500**. Sachant que nous connaissons la solution pour quatre points de vente et pour cinq aussi, et puis nous avons imposé que le nombre des points de vente doit être inférieure ou égal à quatre, du coup le modèle sera non faisable.

Pour résoudre le problème il fallait d'ajouter des déviations dans la contrainte des points de vente pour voir comment le solver va réagir avec ce cas, le modèle goal programming pour ce cas est le suivant :

Modèle GP problème P-médian de points de vente

```

sets:
pointv/1..5/:y,cap;
zone/1..16/:dem;
pointzone(pointv,zone):dist,x;
endsets
data:
costKm = 1;
dem= 1000 400 1060 400 740 800 400 800 600 400 500 400 400400 1000 800;
cap= 3000 3000 3000 3000 3000;
dist=
  4.  5.  0.  1.  2.  3.  4.  5.  1.  0.  2.  4.  4.  8.  4.  5.1
  26  30  80  21  14  20  15  20  20  94  37  29  60  31  36  0
  2.  3.  3.  1.  0.  1.  2.  3.  2.  1.  2.  2.  2.  6.  2.  3.9
  63  51  01  16  68  28  14  13  93  96  94  08  42  10  60
  1.  1.  5.  3.  3.  2.  1.  0.  5.  4.  2.  1.  1.  3.  3.  2.6
  94  70  94  96  04  02  14  63  91  93  95  00  64  76  01  0
  4.  4.  8.  7.  6.  5.  4.  3.  8.  7.  5.  3.  3.  2.  6.  3.8
  87  13  88  00  07  01  23  38  70  71  76  86  65  70  67  2
  3.  3.  4.  3.  2.  2.  2.  3.  4.  3.  1.  0.  1.  4.  4.  0.9
  77  29  85  32  87  61  76  20  37  48  62  64  09  83  27  5;
enddata
min=a+b;
@sum(pointv(i):@sum(zone(j):dem(j)*dist(i,j)*costKm*x(i,j)))<=13500;
!(1.1); @for(pointv(i):@sum(zone(j):dem(j)*x(i,j))<=cap(i));
!(1.2); @sum(pointv(i):y(i))-a+b=4;
!(1.3); @for(zone(j):@sum(pointv(i):x(i,j))=1);
!(1.4); @for(pointv(i):@for(zone(j):x(i,j)<=y(i)));
!(1.5); @for(pointv(i):@sum(zone(j):x(i,j))>=y(i));
@for(pointv:@bin(y));
@for(pointv(i):@for(zone(j):@bin(x(i,j))));

```

- (1.1) : la somme des demandes ne doit pas dépasser la capacité du point de vente.
(1.2) : Nombre de point de vente à ouvrir.
(1.3) : Une zone doit être affectée à un seul point de vente.
(1.4) : Un client doit être servi par un point de vente ouvert.
(1.5) : Un point de vente s'il est ouvert, il doit servir au moins une zone.

Les résultats obtenus pour ce modèle donc sont :

Solver Status		Variables		COSTKM		1.000000	
Model Class:	MILP	Total:	87	A	1.000000		
State:	Global Opt	Nonlinear:	0	B	0.000000		
Objective:	1	Integers:	85	Y(1)	1.000000		
Infeasibility:	0	Constraints		Y(2)	1.000000		
Iterations:	0	Total:	109	Y(3)	1.000000		
Extended Solver Status		Nonlinear:	0	Y(4)	1.000000		
Solver Type:	B-and-B	Nonzeros		Y(5)	1.000000		
Best Obj:	1	Total:	494	CAP(1)	3000.000		
Obj Bound:	1	Nonlinear:	0	CAP(2)	3000.000		
Steps:	0	Generator Memory Used (K)		CAP(3)	3000.000		
Active:	0	63		CAP(4)	3000.000		
		Elapsed Runtime (hh:mm:ss)		CAP(5)	3000.000		
		00:00:00		DEM(1)	1000.000		
				DEM(2)	400.0000		
				DEM(3)	1060.000		
				DEM(4)	400.0000		
				DEM(5)	740.0000		
				DEM(6)	800.0000		

Figure 4. 14: Résultats du modèle GP problème P-médian de point de vente

IV.4.4.1 Interprétation :

Le fait d'imposer une valeur que c'était impossible de l'atteindre par ouvrir quatre points de vente ou moins le solver a indiqué que le modèle est non faisable pour cela nous avons ajouté des variables qui permettent de dévier la contrainte pour avoir une solution, effectivement nous avons obtenu $A=1c$ est-à-dire qu'il fallait ajouter un point de vente pour pouvoir atteindre la valeur imposée **13500**

Si on constate bien les $Y(i)$ qui désignent l'ouverture du point de vente(i) on trouve que les **cinq points de vente sont ouverts** et ce qui valide nos résultats.

IV.5 Problématique 02 : Modèle max-covering :

Nous allons maintenant résoudre les problèmes de non faisabilité apparue dans le modèle max covering en troisième chapitre.

Nous avons dans le chapitre précédent appliqué le modèle max covering dans le cas des saisons de haute consommation tels que le mois de Ramadan e la saison d'été les crèmeries et tous donc nous avons estimé que chaque famille consomme quatre litres par jour donc nous avons rencontré un manque dans la capacité globale pour ce faire nous avons fait appel au modèle max covering pour voir avec la capacité disponible combien de zone nous pouvons couvrir nous avons donc éliminé la contrainte d'affectation de toutes les zones pour voir avec la quantité disponible combien de zone nous pouvons couvrir.

Nous avons partagé la quantité produite par jour sur les cinq points de vente $193670/5=$
3874 pour chaque point de vente, puis nous avons multiplié les demandes par quatre.

IV.5.1 Problème de non faisabilité à cause de la capacité des points de vente :

Dans un autre cas nous avons imposé que toutes les zones doivent être affectées nous avons rencontré un problème de non faisabilité du modèle en raison de la capacité insuffisante, puisque la demande totale dépasse la capacité de production.

Nous avons donc introduit la notion du goal programming pour résoudre ce problème de tels sorte nous avons ajouté une contrainte

Le modèle Lingo pour ce problème est comme suivant :

Modèle GP max-covering problème de capacité

```
sets:
pointv/1..5/:y, cap;
zone/1..16/:dem;
pointzone(pointv, zone):dist, x;
endsets
data:
costKm=1;
PrixVente=1;
dem= 2000 800 2120 800 1480 1600 800 1600 1200 800 1000 800 800 800 2000
800;
cap= 3874 3874 3874 3874 3874;
```

```

dist=
4.  5.  0.  1.  2.  3.  4.  5.  1.  0.  2.  4.  4.  8.  4.  5.1
26 30  80 21  14 20  15 20 20 94 37 29 60 31 36  0
2.  3.  3.  1.  0.  1.  2.  3.  2.  1.  2.  2.  2.  6.  2.  3.9
63 51  01 16  68 28  14 13 93 96 94 08 42 10 60
1.  1.  5.  3.  3.  2.  1.  0.  5.  4.  2.  1.  1.  3.  3.  2.6
94 70  94 96  04 02  14 63 91 93 95 00 64 76 01  0
4.  4.  8.  7.  6.  5.  4.  3.  8.  7.  5.  3.  3.  2.  6.  3.8
87 13  88 00  07 01  23 38 70 71 76 86 65 70 67  2
3.  3.  4.  3.  2.  2.  2.  3.  4.  3.  1.  0.  1.  4.  4.  0.9
77 29  85 32  87 61  76 20 37 48 62 64 09 83 27  5;

enddata
min=a+b;
!(1.1); @sum(pointv(i):y(i))<=5;
!(1.2); @for(zone(j):@sum(pointv(i):x(i,j))=1);
!(1.3); @for(pointv(i):@for(zone(j):x(i,j)<=y(i)));
!(1.4); @for(pointv(i):@sum(zone(j):x(i,j))>=y(i));
@for(pointv:@bin(y));
@for(pointv(i):@for(zone(j):@bin(x(i,j))));
!(1.5); @sum(pointv(i):@sum(zone(j):dem(j)*x(i,j)))-a+b=19370;

```

(1.1) : la somme des demandes ne doit pas dépasser la capacité du point de vente.

(1.2) : Nombre de point de vente à ouvrir.

(1.3) : Une zone doit être affectée à un seul point de vente.

(1.4) : Un client doit être servi par un point de vente ouvert.

(1.5) : Un point de vente s'il est ouvert, il doit servir au moins une zone.

Les résultats de ce modèle sont comme suivant :

Solver Status Model Class: MILP State: Global Opt Objective: 30 Infeasibility: 0 Iterations: 0		Variables Total: 87 Nonlinear: 0 Integers: 85		Constraints Total: 104 Nonlinear: 0		Nonzeros Total: 414 Nonlinear: 0		Generator Memory Used (K) 100		Elapsed Runtime (hh:mm:ss) 00:00:00		Variable Value	
Extended Solver Status Solver Type: B-and-B Best Obj: 30 Obj Bound: 30 Steps: 0 Active: 0										COSTRM 1.000000 PRIXVENTE 1.000000 A 30.000000 B 0.000000 Y(1) 1.000000 Y(2) 0.000000 Y(3) 0.000000 Y(4) 0.000000 Y(5) 0.000000 CAP(1) 3874.000 CAP(2) 3874.000 CAP(3) 3874.000 CAP(4) 3874.000 CAP(5) 3874.000 DEM(1) 2000.000 DEM(2) 800.0000 DEM(3) 2120.000 DEM(4) 800.0000 DEM(5) 1480.000			

Figure 4. 15: Résultats du modèle GP max covering problème de capacité

IV.5.1.1 Interprétation :

Dans ce cas la capacité était moins que la demande du coup ce n'était pas possible de satisfaire toutes les demandes, pour cela le modèle était non faisable.

À l'aide du goal programming nous avons pu résoudre le problème de non faisabilité et entre temps savoir la quantité qui manques **A=30** ;

Dans ce cas nous allons résoudre le problème de non faisabilité pour le modèle max-covering lié à la distance maximale exigé. Nous avons dans le chapitre précédent exigé que la somme globale des distances doit être égale à **100** km par conséquent le modèle ce n'était pas faisable.

Nous avons introduit la notion des déviations de la méthode goal programming pour savoir si nous avons dépassé la distance exigée nous n'avons pas l'atteindre, le modèle Lingo de ce raisonnement est le suivant :

Modèle GP max-covering problème de distance

```
sets:
pointv/1..5/:y,cap;
zone/1..16/:dem;
pointzone(pointv,zone):dist,x;
endsets
data:
costKm=1;
PrixVente=1;
dem= 2000 800 2120 800 1480 1600 800 1600 1200 800 1000 800 800 800 2000
800;
cap= 4000 4000 4000 4000 4000;
dist=
  4.  5.  0.  1.  2.  3.  4.  5.  1.  0.  2.  4.  4.  8.  4.  5.1
 26  30  80  21  14  20  15  20  20  94  37  29  60  31  36  0
  2.  3.  3.  1.  0.  1.  2.  3.  2.  1.  2.  2.  2.  6.  2.  3.9
 63  51  01  16  68  28  14  13  93  96  94  08  42  10  60
  1.  1.  5.  3.  3.  2.  1.  0.  5.  4.  2.  1.  1.  3.  3.  2.6
 94  70  94  96  04  02  14  63  91  93  95  00  64  76  01  0
  4.  4.  8.  7.  6.  5.  4.  3.  8.  7.  5.  3.  3.  2.  6.  3.8
 87  13  88  00  07  01  23  38  70  71  76  86  65  70  67  2
  3.  3.  4.  3.  2.  2.  2.  3.  4.  3.  1.  0.  1.  4.  4.  0.9
 77  29  85  32  87  61  76  20  37  48  62  64  09  83  27  5;
enddata
min=a+b;
!@sum(pointv(i):@sum(zone(j):x(i,j)*dem(j)*PrixVente -
dist(i,j)*x(i,j)*costKm));
!(1.1); @for(pointv(i):@sum(zone(j):dem(j)*x(i,j))<=cap(i));
!(1.2); @sum(pointv(i):y(i))<=5;
!(1.3); @for(zone(j):@sum(pointv(i):x(i,j))=1);
!(1.4); @for(pointv(i):@for(zone(j):x(i,j)<=y(i)));
!(1.5); @for(pointv(i):@sum(zone(j):x(i,j))>=y(i));
@for(pointv:@bin(y));
@for(pointv(i):@for(zone(j):@bin(x(i,j))));
!(1.6); @sum(pointv(i):@sum(zone(j):dist(i,j)*x(i,j)))-a+b=100
```

- (1.1) : la somme des demandes ne doit pas dépasser la capacité du point de vente.
(1.2) : Nombre de point de vente à ouvrir.
(1.3) : Une zone doit être affectée à un seul point de vente.
(1.4) : Un client doit être servi par un point de vente ouvert.
(1.5) : Un point de vente s'il est ouvert, il doit servir au moins une zone.
(1.6) : distance globale de réseau à ne pas dépasser.

Les résultats de ce modèle sont comme suivant :

Solver Status Model Class: MILP State: Global Opt Objective: 18.85 Infeasibility: 5.32907e-015 Iterations: 1684		Variables Total: 87 Nonlinear: 0 Integers: 85		COSTKM 1.000000 PRIXVENTE 1.000000 A 0.000000 B 18.850000 Y (1) 1.000000 Y (2) 1.000000 Y (3) 1.000000 Y (4) 1.000000 Y (5) 1.000000	
Extended Solver Status Solver Type: B-and-B Best Obj: 18.85 Obj Bound: 18.85 Steps: 0 Active: 0		Constraints Total: 109 Nonlinear: 0		CAP (1) 4000.000 CAP (2) 4000.000 CAP (3) 4000.000 CAP (4) 4000.000 CAP (5) 4000.000 DEM (1) 2000.000 DEM (2) 800.00000 DEM (3) 2120.000 DEM (4) 800.00000 DEM (5) 1480.000 DEM (6) 1600.000	
		Nonzeros Total: 494 Nonlinear: 0			
		Generator Memory Used (K) 101			
		Elapsed Runtime (hh:mm:ss) 00:00:00			

Figure 4. 16: Résultats du modèle GP max-covering problème de distance

IV.5.1.2 Interprétation :

La notion de déviation du goal programming nous a donné **B=18.85** c'est-à-dire que la distance maximale parcourue du réseau n'a pas dépassée la valeur exigée et c'est tout affect bénéfique pour notre conception puisque nous avons pu affecter toutes les zones en minimisant la distance parcourue et par conséquent optimiser le cout global de la chaîne et c'est exactement le 'objectif majeur d'une étude logistique.

IV.5.2 Problème de non faisabilité à cause de point de vente :

Dans ce cas nous allons résoudre le cas de non faisabilité lié à l'insuffisance des points de vente rencontré dans le chapitre précédent ou nous avons supposé que nous avons un budget d'ouverture de trois points de vente du coup ce n'était pas possible de couvrir toutes les zones avec trois points de vente par conséquent le modèle était non faisable.

Goal programming nous a permis de savoir combien de point de vente nous avons besoin d'ouvrir afin de satisfaire toutes les demandes et couvrir toutes les zones.

Le modèle Lingo qui représente ce raisonnement est le suivant :

Modèle Lingo GP max-covering problème de point de vente

```

sets:
pointv/1..5/:y,cap;
zone/1..16/:dem;
pointzone(pointv,zone):dist,x;
endsets
data:
costKm=1;
PrixVente=1;
dem= 2000 800 2120 800 1480 1600 800 1600 1200 800 1000 800 800 800 2000
800;
cap= 4000 4000 4000 4000 4000;
dist =
  4.  5.  0.  1.  2.  3.  4.  5.  1.  0.  2.  4.  4.  8.  4.  5.1
  26  30  80  21  14  20  15  20  20  94  37  29  60  31  36  0
  2.  3.  3.  1.  0.  1.  2.  3.  2.  1.  2.  2.  2.  6.  2.  3.9
  63  51  01  16  68  28  14  13  93  96  94  08  42  10  60
  1.  1.  5.  3.  3.  2.  1.  0.  5.  4.  2.  1.  1.  3.  3.  2.6

```

94	70	94	96	04	02	14	63	91	93	95	00	64	76	01	0
4.	4.	8.	7.	6.	5.	4.	3.	8.	7.	5.	3.	3.	2.	6.	3.8
87	13	88	00	07	01	23	38	70	71	76	86	65	70	67	2
3.	3.	4.	3.	2.	2.	2.	3.	4.	3.	1.	0.	1.	4.	4.	0.9
77	29	85	32	87	61	76	20	37	48	62	64	09	83	27	5;

```

Enddata
min=a+b;
!@sum(pointv(i):@sum(zone(j):x(i,j)*dem(j)*PrixVente -
dist(i,j)*x(i,j)*costKm));
!(1.1); @for(pointv(i):@sum(zone(j):dem(j)*x(i,j))<=cap(i));
!(1.2); @sum(pointv(i):y(i))-a+b=3;
!(1.3); @for(zone(j):@sum(pointv(i):x(i,j))=1);
!(1.4); @for(pointv(i):@for(zone(j):x(i,j)<=y(i)));
!(1.5); @for(pointv(i):@sum(zone(j):x(i,j))>=y(i));
@for(pointv:@bin(y));
@for(pointv(i):@for(zone(j):@bin(x(i,j))));

```

- (1.1) : la somme des demandes ne doit pas dépasser la capacité du point de vente.
(1.2) : Nombre de point de vente à ouvrir.
(1.3) : Une zone doit être affectée à un seul point de vente.
(1.4) : Un client doit être servi par un point de vente ouvert.
(1.5) : Un point de vente s'il est ouvert, il doit servir au moins une zone.

Les résultats sont comme suivant :

Solver Status: Model Class: MILP State: Global Opt Objective: 2 Infeasibility: 0 Iterations: 502		Variables: Total: 87 Nonlinear: 0 Integers: 85		Constraints: Total: 108 Nonlinear: 0		Nonzeros: Total: 414 Nonlinear: 0		Generator Memory Used (K): 101		Elapsed Runtime (hh:mm:ss): 00:00:00		<table border="1"> <thead> <tr> <th></th> <th>PRIXVENTE</th> <th>1.000000</th> <th>0.000000</th> </tr> </thead> <tbody> <tr><td>A</td><td>2.000000</td><td>0.000000</td><td>0.000000</td></tr> <tr><td>B</td><td>0.000000</td><td>2.000000</td><td>0.000000</td></tr> <tr><td>Y (1)</td><td>1.000000</td><td>1.000000</td><td>1.000000</td></tr> <tr><td>Y (2)</td><td>1.000000</td><td>1.000000</td><td>1.000000</td></tr> <tr><td>Y (3)</td><td>1.000000</td><td>1.000000</td><td>1.000000</td></tr> <tr><td>Y (4)</td><td>1.000000</td><td>1.000000</td><td>1.000000</td></tr> <tr><td>Y (5)</td><td>1.000000</td><td>1.000000</td><td>1.000000</td></tr> <tr><td>CAP (1)</td><td>4000.000</td><td>0.000000</td><td>0.000000</td></tr> <tr><td>CAP (2)</td><td>4000.000</td><td>0.000000</td><td>0.000000</td></tr> <tr><td>CAP (3)</td><td>4000.000</td><td>0.000000</td><td>0.000000</td></tr> <tr><td>CAP (4)</td><td>4000.000</td><td>0.000000</td><td>0.000000</td></tr> <tr><td>CAP (5)</td><td>4000.000</td><td>0.000000</td><td>0.000000</td></tr> <tr><td>DEM (1)</td><td>2000.000</td><td>0.000000</td><td>0.000000</td></tr> <tr><td>DEM (2)</td><td>800.0000</td><td>0.000000</td><td>0.000000</td></tr> <tr><td>DEM (3)</td><td>2120.000</td><td>0.000000</td><td>0.000000</td></tr> <tr><td>DEM (4)</td><td>800.0000</td><td>0.000000</td><td>0.000000</td></tr> <tr><td>DEM (5)</td><td>1480.000</td><td>0.000000</td><td>0.000000</td></tr> <tr><td>DEM (6)</td><td>1600.000</td><td>0.000000</td><td>0.000000</td></tr> <tr><td>DEM (7)</td><td>800.0000</td><td>0.000000</td><td>0.000000</td></tr> </tbody> </table>					PRIXVENTE	1.000000	0.000000	A	2.000000	0.000000	0.000000	B	0.000000	2.000000	0.000000	Y (1)	1.000000	1.000000	1.000000	Y (2)	1.000000	1.000000	1.000000	Y (3)	1.000000	1.000000	1.000000	Y (4)	1.000000	1.000000	1.000000	Y (5)	1.000000	1.000000	1.000000	CAP (1)	4000.000	0.000000	0.000000	CAP (2)	4000.000	0.000000	0.000000	CAP (3)	4000.000	0.000000	0.000000	CAP (4)	4000.000	0.000000	0.000000	CAP (5)	4000.000	0.000000	0.000000	DEM (1)	2000.000	0.000000	0.000000	DEM (2)	800.0000	0.000000	0.000000	DEM (3)	2120.000	0.000000	0.000000	DEM (4)	800.0000	0.000000	0.000000	DEM (5)	1480.000	0.000000	0.000000	DEM (6)	1600.000	0.000000	0.000000	DEM (7)	800.0000	0.000000	0.000000
	PRIXVENTE	1.000000	0.000000																																																																																												
A	2.000000	0.000000	0.000000																																																																																												
B	0.000000	2.000000	0.000000																																																																																												
Y (1)	1.000000	1.000000	1.000000																																																																																												
Y (2)	1.000000	1.000000	1.000000																																																																																												
Y (3)	1.000000	1.000000	1.000000																																																																																												
Y (4)	1.000000	1.000000	1.000000																																																																																												
Y (5)	1.000000	1.000000	1.000000																																																																																												
CAP (1)	4000.000	0.000000	0.000000																																																																																												
CAP (2)	4000.000	0.000000	0.000000																																																																																												
CAP (3)	4000.000	0.000000	0.000000																																																																																												
CAP (4)	4000.000	0.000000	0.000000																																																																																												
CAP (5)	4000.000	0.000000	0.000000																																																																																												
DEM (1)	2000.000	0.000000	0.000000																																																																																												
DEM (2)	800.0000	0.000000	0.000000																																																																																												
DEM (3)	2120.000	0.000000	0.000000																																																																																												
DEM (4)	800.0000	0.000000	0.000000																																																																																												
DEM (5)	1480.000	0.000000	0.000000																																																																																												
DEM (6)	1600.000	0.000000	0.000000																																																																																												
DEM (7)	800.0000	0.000000	0.000000																																																																																												

Figure 4. 17: Résultats de modèle GP max-covering problème de point de vente

IV.5.2.1 Interprétation :

Le fait d'exiger trois points de vente à ouvrir ce n'était pas possible de couvrir toutes les demandes et affecter toutes les zones pour cela le modèle dans un premier temps était non faisable, goal programming nous a donné A=2 c'est-à-dire qu'il faut ouvrir deux points de vente de plus pour couvrir toutes les zones si on constate bien les résultats au-dessus on trouve que les cinq points de vente sont ouverts pour pouvoir satisfaire les toutes demandes et cela valide le résultat obtenu.

IV.6 Problématique 03 :

Nous allons maintenant exploiter une autre caractéristique de la méthode goal programming, comme nous avons déjà défini goal programming est une méthode de décision multicritère c'est-à-dire qu'on peut traiter plusieurs objectifs simultanément à l'aide de cette approche, ce qui est important on ne peut pas certainement atteindre à 100% tous les objectifs voulus simultanément pour ce faire le goal programming nous permet d'avoir un résultat satisfaisant qui donne la meilleure combinaison entre les objectifs fixés tout en minimisant les écarts(déviations) pour chaque objectif.

Pour ce faire nous allons fusionner entre les deux modèles déjà vu dans ce cas d'étude qui sont le problème P-médian qui est un problème de minimisation avec le problème Max-covering qui est un problème de maximisation.

Nous avons exigé que la fonction objective du modèle max-covering doit être égale à 26000, et la fonction objective relative au modèle p-médian doit être égale 20000 et nous avons gardé les mêmes contraintes en commun entre les deux modèles, le modèle Lingo est le suivant :

Modèle Lingo problème multi objectifs

```

pointv/1..5/:y,cap;
zone/1..16/:dem;
pointzone(pointv,zone):dist,x;
endsets
data:
costKm=1;
PrixVente=1;
dem= 2000 800 2120 800 1480 1600 800 1600 1200 800 1000 800 800 800 2000
800;
cap= 4000 4000 4000 4000 4000;
dist=
4. 5. 0. 1. 2. 3. 4. 5. 1. 0. 2. 4. 4. 8. 4. 5.1
26 30 80 21 14 20 15 20 20 94 37 29 60 31 36 0
2. 3. 3. 1. 0. 1. 2. 3. 2. 1. 2. 2. 2. 6. 2. 3.9
63 51 01 16 68 28 14 13 93 96 94 08 42 10 60
1. 1. 5. 3. 3. 2. 1. 0. 5. 4. 2. 1. 1. 3. 3. 2.6
94 70 94 96 04 02 14 63 91 93 95 00 64 76 01 0
4. 4. 8. 7. 6. 5. 4. 3. 8. 7. 5. 3. 3. 2. 6. 3.8
87 13 88 00 07 01 23 38 70 71 76 86 65 70 67 2
3. 3. 4. 3. 2. 2. 2. 3. 4. 3. 1. 0. 1. 4. 4. 0.9
77 29 85 32 87 61 76 20 37 48 62 64 09 83 27 5;
Enddata
min=a+b+c+d;
@sum(pointv(i):@sum(zone(j):x(i,j)*dem(j)*PrixVente -
dist(i,j)*x(i,j)*costKm))+a=26000;
@sum(pointv(i):@sum(zone(j):dem(j)*dist(i,j)*costKm*x(i,j)))-b=20000;
!(1.1); @for(pointv(i):@sum(zone(j):dem(j)*x(i,j))<=cap(i));
!(1.2); @sum(pointv(i):y(i))<=5;
!(1.3); @for(zone(j):@sum(pointv(i):x(i,j))=1);
!(1.4); @for(pointv(i):@for(zone(j):x(i,j)<=y(i)));
!(1.5); @for(pointv(i):@sum(zone(j):x(i,j))>=y(i));
@for(pointv:@bin(y));
@for(pointv(i):@for(zone(j):@bin(x(i,j))));

```

(1.1) : la somme des demandes ne doit pas dépasser la capacité du point de vente.

(1.2) : Nombre de point de vente à ouvrir.

- (1.3) : Une zone doit être affectée à un seul point de vente.
 (1.4) : Un client doit être servi par un point de vente ouvert.
 (1.5) : Un point de vente s'il est ouvert, il doit servir au moins une zone.

Les résultats de ce modèle sont comme suivant :

Solver Status	Variables	COSTKM	1.000000	0.000000
Model Class: MILP	Total: 89	PRIXVENTE	1.000000	0.000000
State: Global Opt	Nonlinear: 0	A	0.000000	2.000000
Objective: 26643.8	Integers: 85	B	6633.440	0.000000
Infeasibility: 7.95808e-013	Constraints	C	20010.40	0.000000
Iterations: 1166	Total: 110	D	0.000000	2.000000
Extended Solver Status	Nonlinear: 0	Y(1)	1.000000	0.000000
Solver Type: B-and-B	Nonzeros	Y(2)	1.000000	0.000000
Best Obj: 26643.8	Total: 578	Y(3)	1.000000	0.000000
Obj Bound: 26643.8	Nonlinear: 0	Y(4)	1.000000	0.000000
Steps: 6	Generator Memory Used (K)	Y(5)	1.000000	0.000000
Active: 0	101	CAP(1)	4000.000	0.000000
	Elapsed Runtime (hh:mm:ss)	CAP(2)	4000.000	0.000000
	00:00:00	CAP(3)	4000.000	0.000000
		CAP(4)	4000.000	0.000000
		CAP(5)	4000.000	0.000000
		DEM(1)	2000.000	0.000000
		DEM(2)	800.0000	0.000000
		DEM(3)	2120.000	0.000000
		DEM(4)	800.0000	0.000000

Figure 4. 18: Résultats du modèle lingo problème multi objectifs

IV.6.1 Interprétation :

La solution optimale obtenue est égale à 26643.6 représente la somme totale des déviations, cette valeur est partagée entre les deux objectifs de tel sorte **A=6633.44** représente la déviation par rapport au premier objectif implique que nous n'avons pas atteint l'objectif voulu et l'écart est représenté par la valeur A.

B=20010.4 exprime la valeur de dépassement du deuxième objectif.

IV.7 Problématique 04 :

Nous avons ajouté des changements sur notre cahier des charges cette fois nous avons gardé la fonction objective du modèle max-covering=**26000**, nous avons la fusionner par une contrainte qui impose que la somme des distances parcourues dans le réseau de distribution doit être égale à **100 km**

Le modèle est comme suivant :

Modèle GP multi objectifs sans poids d'importance

```
sets:
pointv/1..5/:y,cap;
zone/1..16/:dem;
pointzone(pointv,zone):dist,x;
endsets
data:
costKm=1;
PrixVente=1;
dem= 2000 800 2120 800 1480 1600 800 1600 1200 800 1000 800 800 800 2000
```

```

800;
cap= 4000 4000 4000 4000 4000;
dist=
  4.  5.  0.  1.  2.  3.  4.  5.  1.  0.  2.  4.  4.  8.  4.  5.1
 26  30  80  21  14  20  15  20  20  94  37  29  60  31  36  0
  2.  3.  3.  1.  0.  1.  2.  3.  2.  1.  2.  2.  2.  6.  2.  3.9
 63  51  01  16  68  28  14  13  93  96  94  08  42  10  60
  1.  1.  5.  3.  3.  2.  1.  0.  5.  4.  2.  1.  1.  3.  3.  2.6
 94  70  94  96  04  02  14  63  91  93  95  00  64  76  01  0
  4.  4.  8.  7.  6.  5.  4.  3.  8.  7.  5.  3.  3.  2.  6.  3.8
 87  13  88  00  07  01  23  38  70  71  76  86  65  70  67  2
  3.  3.  4.  3.  2.  2.  2.  3.  4.  3.  1.  0.  1.  4.  4.  0.9
 77  29  85  32  87  61  76  20  37  48  62  64  09  83  27  5;
enddata
min=a+b+c+d;
@sum(pointv(i):@sum(zone(j):x(i,j)*dem(j)*PrixVente -
dist(i,j)*x(i,j)*costKm))-a+b=35000;
@sum(pointv(i):@sum(zone(j):x(i,j)*dist(i,j)))-c+d=100;
!(1.1); @for(pointv(i):@sum(zone(j):dem(j)*x(i,j))<=cap(i));
!(1.2); @sum(pointv(i):y(i))<=5;
!(1.3); @for(zone(j):@sum(pointv(i):x(i,j))=1);
!(1.4); @for(pointv(i):@for(zone(j):x(i,j)<=y(i)));
!(1.5); @for(pointv(i):@sum(zone(j):x(i,j))>=y(i));
@for(pointv:@bin(y));
@for(pointv(i):@for(zone(j):@bin(x(i,j))));

```

(1.1) : la somme des demandes ne doit pas dépasser la capacité du point de vente.

(1.2) : Nombre de point de vente à ouvrir.

(1.3) : Une zone doit être affectée à un seul point de vente.

(1.4) : Un client doit être servi par un point de vente ouvert.

(1.5) : Un point de vente s'il est ouvert, il doit servir au moins une zone.

Les résultats obtenus sont :

Solver Status: Model Class: MILP State: Global Opt Objective: 6700 Infeasibility: 1.36424e-012 Iterations: 24		Variables: Total: 89 Nonlinear: 0 Integers: 85		PRIXVENTE: 1.000000	
Extended Solver Status: Solver Type: B-and-B Best Obj: 6700 Obj Bound: 6700 Steps: 0 Active: 0		Constraints: Total: 110 Nonlinear: 0		A: 0.000000 B: 6655.560 C: 0.000000 D: 44.44000	
Nonzeros: Total: 578 Nonlinear: 0		Generator Memory Used (K): 102		Y (1): 1.000000 Y (2): 1.000000 Y (3): 1.000000 Y (4): 1.000000 Y (5): 1.000000	
Elapsed Runtime (hh:mm:ss): 00:00:00		CAP (1): 4000.000 CAP (2): 4000.000 CAP (3): 4000.000 CAP (4): 4000.000 CAP (5): 4000.000		DEM (1): 2000.000 DEM (2): 800.0000 DEM (3): 2120.000 DEM (4): 800.0000 DEM (5): 1480.000	

Figure 4. 19: Résultats du modèle multi objectifs sans poids d'importance

IV.7.1.1 Interprétation :

Nous avons obtenu une solution optimale égale à **6700**, **B=6655.56** qui représente la valeur à rajouter pour atteindre exactement le premier objectif, tandis que **D=44.44** qui représente l'écart à rajouter pour atteindre exactement le deuxième objectif.

Nous avons constaté que l'écart par rapport au premier objectif est plus grand et nous voulions le diminuer :

Une autre utilité du goal programming c'est qu'elle permet de classer les objectifs voulus par ordre de priorité par l'attribution des poids d'importance pour chaque objectif, nous avons investi sur cette caractéristique pour réduire l'écart par rapport au premier objectif. Le modèle qui représente ce raisonnement est comme suivant :

Modèle GP multi objectifs avec poids d'importance

```
sets:
pointv/1..5/:y, cap;
zone/1..16/:dem;
pointzone(pointv, zone):dist, x;
endsets
data:
costKm=1;
PrixVente=1;
dem= 2000 800 2120 800 1480 1600 800 1600 1200 800 1000 800 800 800 2000
800;
cap= 4000 4000 4000 4000 4000;
dist=
  4.  5.  0.  1.  2.  3.  4.  5.  1.  0.  2.  4.  4.  8.  4.  5.1
26 30 80 21 14 20 15 20 20 94 37 29 60 31 36 0
  2.  3.  3.  1.  0.  1.  2.  3.  2.  1.  2.  2.  2.  6.  2.  3.9
63 51 01 16 68 28 14 13 93 96 94 08 42 10 60
  1.  1.  5.  3.  3.  2.  1.  0.  5.  4.  2.  1.  1.  3.  3.  2.6
94 70 94 96 04 02 14 63 91 93 95 00 64 76 01 0
  4.  4.  8.  7.  6.  5.  4.  3.  8.  7.  5.  3.  3.  2.  6.  3.8
87 13 88 00 07 01 23 38 70 71 76 86 65 70 67 2
  3.  3.  4.  3.  2.  2.  2.  3.  4.  3.  1.  0.  1.  4.  4.  0.9
77 29 85 32 87 61 76 20 37 48 62 64 09 83 27 5;
Enddata
min=2*a+2*b+c+d;
@sum(pointv(i):@sum(zone(j):x(i,j)*dem(j)*PrixVente -
dist(i,j)*x(i,j)*costKm))-a+b=26000;
@sum(pointv(i):@sum(zone(j):x(i,j)*dist(i,j)))-c+d=100;
!(1.1); @for(pointv(i):@sum(zone(j):dem(j)*x(i,j))<=cap(i));
!(1.2); @sum(pointv(i):y(i))<=5;
!(1.3); @for(zone(j):@sum(pointv(i):x(i,j))=1);
!(1.4); @for(pointv(i):@for(zone(j):x(i,j)<=y(i)));
!(1.5); @for(pointv(i):@sum(zone(j):x(i,j))>=y(i));
@for(pointv:@bin(y));
@for(pointv(i):@for(zone(j):@bin(x(i,j))));
```

- (1.1) : la somme des demandes ne doit pas dépasser la capacité du point de vente.
(1.2) : Nombre de point de vente à ouvrir.
(1.3) : Une zone doit être affectée à un seul point de vente.
(1.4) : Un client doit être servi par un point de vente ouvert.
(1.5) : Un point de vente s'il est ouvert, il doit servir au moins une zone.

Les résultats obtenus sont représentés par la figure suivante :

Solver Status: MILP	Variables: Total: 89	COSTRM: 1.000000	0.000000
State: Global Opt	Nonlinear: 0	PRIXVENTE: 1.000000	0.000000
Objective: 13330.9	Integers: 85	A: 0.000000	4.000000
Infeasibility: 3.41061e-013	Constraints: Total: 110	B: 6630.940	0.000000
Iterations: 1189	Nonlinear: 0	C: 0.000000	2.000000
Extended Solver Status: B-and-B	Nonzeros: Total: 578	D: 69.06000	0.000000
Solver Type: B-and-B	Nonlinear: 0	Y (1): 1.000000	0.000000
Best Obj: 13330.9	Generator Memory Used (K): 103	Y (2): 1.000000	0.000000
Obj Bound: 13330.9	Elapsed Runtime (hh:mm:ss): 00:00:00	Y (3): 1.000000	0.000000
Steps: 10		Y (4): 1.000000	0.000000
Active: 0		Y (5): 1.000000	0.000000
		CAP (1): 4000.000	0.000000
		CAP (2): 4000.000	0.000000
		CAP (3): 4000.000	0.000000
		CAP (4): 4000.000	0.000000
		CAP (5): 4000.000	0.000000
		DEM (1): 2000.000	0.000000
		DEM (2): 800.0000	0.000000
		DEM (3): 2120.000	0.000000
		DEM (4): 800.0000	0.000000

Figure 4. 20: Résultats du modèle Gp multi objectifs avec poids d'importance

IV.7.1.2 Interprétation :

On constate que la solution optimale a été augmentée c'est tout affect normal puisque nous avons multiplié les variables de déviation du premier objectif par deux, si on analyse bien les résultats la valeur de **B=6630.94** a été diminuée par rapport au premier cas ou les deux objectifs avaient le même poids d'importance tandis que la valeur de **D=69.06** a été augmentée par rapport au premier cas.

IV.8 Synthèse :

L'approche goal programming est une méthode de résolution des problèmes multi objectifs elle consiste à traiter le problème selon l'ordre d'importance des objectifs voulu et trouver un compromis qui permet d'ajuster et trouver un point d'équilibre pour lequel nous obtenons la meilleure approximation en minimisant les déviations pour chaque objectif.

Ces déviations peuvent être positives au cas de dépassement ou négatives au cas opposé. Le décideur peut également attribuer des importances relatives pour chaque un des objectifs voulus afin de les classer par ordre de priorité.

Goal programming ne fournit pas forcément une solution optimale pour chaque objectif mais elle donne une solution satisfaisante qui contient la meilleure combinaison entre les objectifs voulus.

IV.9 Conclusion :

Nous avons vu dans ce chapitre la différence entre la modélisation linéaire et la modélisation goal programming, les points d'avantage d'utiliser le goal programming aussi nous avons vu aussi plusieurs scénarios pour un seul problème pour nous permettre d'analyser les divers résultats et localiser les points de différence et les meilleures solutions possibles en utilisant le solveur Lingo 18.0, par la suite nous avons projeté les notions de goal programming sur notre cas d'études en plusieurs scénarios.

Conclusion Générale :

Les études logistiques est un domaine très fréquent mais il reste toujours un sujet d'actualité même si les études se déroulent de manière presque similaire, puisque l'environnement se change à chaque fois du coup les conditions vont changer ce qui permet d'avoir des nouveaux critères et nouvelles contraintes permettant d'avoir des nouveaux systèmes.

Les méthodes appliquées dans notre PFE sont des outils très efficace dans le domaine logistique, le modèle p-médian qui nous a permis de concevoir un réseau de distribution qui minimise la distance entre chaque point de vente et chaque client, d'autre part le modèle max-covering nous a donné une autre configuration qui se base sur la maximisation de couverture des zones pour chaque point de vente.

La méthode goal programming de son était un point fort dans notre travail vu ses caractéristiques qui nous ont permis de résoudre pleins de problèmes de non faisabilité ainsi qu'améliorer les solutions obtenues à chaque fois avec la notion de déviation et la fusion de plusieurs objectifs dans un modèle unique.

L'impact de notre étude ne se limite pas aux parties prenantes vendeur-client il s'étend de tel minimiser le cout global de la chaine, assurer la disponibilité du lait, optimiser les distances parcourues et par conséquent réduire la consommation des carburants pour but de préserver l'environnement.

Nous avons utilisé dans ce PFE le solver Lingo 18.0 pour modéliser nos problématiques et pour la résolution séquentielle de ce problème, nous avons aussi utilisé le logiciel QGIS pour concevoir notre carte de réseau de distribution en localisant les points de vente candidats et les zones des clients.

Nous essayerons dans les futures occasions d'élargir l'études sur toute la ville de Tlemcen et pourquoi pas sur tous l'West algérien, nous pouvons aussi d'agrandir le travail pour plus de produits alimentaires, en appliquant plus d'autres modèles tels que le modèle p-centre, le modèle set-covering et même les heuristiques et le métaheuristiques.

Nous avons dans ce travail analysé le réseau de distribution qui existe déjà en réalité, nous pouvons par la suite concevoir un nouveau réseau qui permet de localiser des points de vente selon des critères bien étudiés tels que la sécurité, l'accessibilité, la densité de population dans cette région...

Les résultats obtenus dans les deux premières problématiques P-médian et max-covering ne sont pas très satisfaisants, par contre l'approche goal programming nous a permis d'améliorer ces résultats et même d'améliorer notre cahier des charges dans les futures occasions pour avoir des résultats plus satisfaisants et pour pouvoir améliorer et expandre notre réseau.

Bibliography:

- AIKENS, Charles H. Facility location models for distribution planning. *European journal of operational research*, 1985, vol. 22, no 3, p. 263-279.
- AL- SULTAN, Khaled S. et AL- FAWZAN, Mohammad Abdulrahman. A tabu search approach to the uncapacitated facility location problem. *Annals of Operations Research*, 1999, vol. 86, p. 91-103.
- AYADI, Dorra. *Optimisation multicritère de la fiabilité: application du modèle de goal programming avec les fonctions de satisfactions dans l'industrie de traitement de gaz*. 2010. Thèse de doctorat. Université d'Angers.
- BABAI, Mohamed Zied. *Politiques de pilotage de flux dans les chaînes logistiques: impact de l'utilisation des prévisions sur la gestion de stocks*. 2005. Thèse de doctorat. Ecole Centrale Paris.
- BENAYOUN, Raphael, DE MONTGOLFIER, J., TERGNY, Jo, *et al.* Linear programming with multiple objective functions: Step method (STEM). *Mathematical programming*, 1971, vol. 1, no 1, p. 366-375.
- BENNEKROUF, Mohammed. *Modélisation et simulation d'une chaîne logistique inverse en tenant compte de la robustesse*. 2013. Thèse de doctorat, Université de Tlemcen.
- BOUCHRIHA, Hanen. *Faire ou faire-faire dans la conception d'une chaîne logistique: un outil d'aide à la décision*. 2002. Thèse de doctorat. Institut National Polytechnique de Grenoble-INPG.
- BOUDAHRI, Fethi, SARI, Zaki, MALIKI, Fouad, *et al.* Design and optimization of the supply chain of agri-foods: Application distribution network of chicken meat. In: *2011 International conference on communications, computing and control applications (CCCA)*. IEEE, 2011. p. 1-6.
- BOUDAHRI, Fethi. *Conception et Pilotage d'une Chaîne Logistique Agroalimentaire. Application : produits de volaille dans la ville de Tlemcen*. 2013. Université Abou-Bekr Belkaïd–Tlemcen, Thèse de doctorat.
- BOUKLI-HACENE, Rabab. *Conception et optimisations d'une chaîne logistique destinée aux produits pharmaceutiques*. 2020. Thèse de doctorat. Université Abou-Bekr Belkaïd–Tlemcen.
- CANÓS, María José, IVORRA, Carlos, et LIERN, Vicente. The fuzzy p-median problem: A global analysis of the solutions. *European Journal of Operational Research*, 2001, vol. 130, no 2, p. 430-436.
- Catram Consultants, Team International et Team Maroc, Un réseau euro-méditerranéen de plates-formes logistiques, Rapport de Synthèse, FEMIP, 2010.
- CHARNES, Abraham et COOPER, William Wager. Goal programming and multiple objective optimizations: Part 1. *European journal of operational research*, 1977, vol. 1, no 1, p. 39-54.
- CHARNES, Abraham, FROME, Edward L., et YU, Po-Lung. The equivalence of generalized least squares and maximum likelihood estimates in the exponential

- family. *Journal of the American Statistical Association*, 1976, vol. 71, no 353, p. 169-171.
- CHURCH, Richard et REVELLE, Charles. The maximal covering location problem. In : *Papers of the regional science association*. Springer-Verlag, 1974. p. 101-118.
 - COHEN, Morris A. et LEE, Hau L. Strategic analysis of integrated production-distribution systems: models and methods. *Operations research*, 1988, vol. 36, no 2, p. 216-228.
 - DASKIN, Mark S. *Network and discrete location: models, algorithms, and applications*. John Wiley & Sons, 2011.
 - DASKIN, Mark S., SNYDER, Lawrence V., et BERGER, Rosemary T. Facility location in supply chain design. In: *Logistics systems: Design and optimization*. Springer, Boston, MA, 2005. p. 39-65.
 - DING, Hongwei. *une approche d'optimisation basée sur la simulation pour la conception des chaînes logistiques: Applications dans les industries automobile et textile*. 2004. Thèse de doctorat. Université Paul Verlaine-Metz.
 - DURRIEU, François et VALETTE-FLORENCE, Pierre. L'analyse typologique: De l'exploratoire au confirmatoire. *Management des ressources humaines: Méthodes de recherche en sciences humaines et sociales*, 2005, p. 379-429.
 - EFROYMSON, M_A et RAY, T. L. A branch-bound algorithm for plant location. *Operations Research*, 1966, vol. 14, no 3, p. 361-368.
 - FLAVELL, Richard B. et SMITH, Derek B. Nucleotide sequence organisation in the wheat genome. *Heredity*, 1976, vol. 37, no 2, p. 231-252.
 - FRANCOIS, Julien. *Planification des chaînes logistiques: Modélisation du système décisionnel et performance*. 2007. Thèse de doctorat. Université Sciences et Technologies-Bordeaux I.
 - GALVAO, ROBERTO DIEGUEZ. The use of Lagrangean relaxation in the solution of uncapacitated facility location problems. *Computers & Operations Research*, 1993.
 - GANJAVI, M. R. Coordinated comprehensive design of distribution systems at medium and low voltage levels. In : *16th International Conference and Exhibition on Electricity Distribution, 2001. Part 1: Contributions. CIRED.(IEE Conf. Publ No. 482)*. IET, 2001. p. 6 pp. vol. 5.
 - GEOFFRION, Arthur M. Lagrangean relaxation for integer programming. In : *Approaches to integer programming*. Springer, Berlin, Heidelberg, 1974. p. 82-114.
 - GUYON, Olivier, ABSI, Nabil, BOUDOUIN, Daniel, et al. Plats-formes en centre ville pour la Logistique Urbaine: étude sur la ville de Marseille. In : *2ème Journée de Recherche " Mobilité, Transport et Logistique » (MTL 2010)*. 2010.
 - HADJ-HAMOU, Khaled. *Contribution à la conception de produits à forte diversité et de leur chaîne logistique : une approche par contraintes*. 2002. Thèse de doctorat. Institut National Polytechnique de Toulouse-INPT.
 - HADJILA Zakariya, MANSOURI, Ismail. La conception d'une chaîne logistique pour la distribution des produits pharmaceutique Application sur la ville de Tlemcen. 2016
 - HAKIMI, S. Louis. Optimum locations of switching centers and the absolute centers and medians of a graph. *Operations research*, 1964, vol. 12, no 3, p. 450-459.

- HAMMAMI, Abdelkader. *Modélisation technico-économique d'une chaîne logistique dans une entreprise réseau*. 2003. Thèse de doctorat. Ecole Nationale Supérieure des Mines de Saint-Etienne; Université Laval.
- KARIV, Oded et HAKIMI, S. Louis. An algorithmic approach to network location problems. I: The p-centers. *SIAM Journal on Applied Mathematics*, 1979, vol. 37, no 3, p. 513-538.
- KETTANI, Ossama, AOUNI, Belaïd, et MARTEL, Jean-Marc. The double role of the weight factor in the goal programming model. *Computers & Operations Research*, 2004, vol. 31, no 11, p. 1833-1845.
- KLINCEWICZ, John G. et LUSS, Hanan. A Lagrangian relaxation heuristic for capacitated facility location with single-source constraints. *Journal of the Operational Research Society*, 1986, vol. 37, no 5, p. 495-500.
- KUMAR, Anil. Green Logistics for sustainable development: an analytical review. *IOSRD International Journal of Business*, 2015, vol. 1, no 1, p. 7-13.
- LEE, Hau L., BILLINGTON, Corey, et CARTER, Brent. Hewlett-Packard gains control of inventory and service through design for localization. *Interfaces*, 1993, vol. 23, no 4, p. 1-11.
- MAREÏ, Nora, AGUILÉRA, Anne, BELTON-CHEVALLIER, Leslie, *et al.* Pratiques et lieux du e-commerce alimentaire. Nouvelles logistiques, nouvelles mobilités?. *Netcom. Réseaux, communication et territoires*, 2016, no 30-1/2, p. 119-138.
- MARTINEZ-PONS, Manuel. Test of a model of parental inducement of academic self-regulation. *The Journal of Experimental Education*, 1996, vol. 64, no 3, p. 213-227.
- MERZOUK, Salah Eddine. *Problème de dimensionnement de lots et de livraisons: application au cas de la chaîne logistique*. 2007. Thèse de doctorat. Besançon.
- MINIEKA, Edward. The m-center problem. *Siam Review*, 1970, vol. 12, no 1, p. 138-139.
- MOCELLIN, Fabrice. *Gestion des entrepôts et plates-formes: assurez la performance de votre supply chain par la maîtrise des zones de stockage*. Dunod, 2003.
- MOUFAD, Imane et JAWAB, Fouad. Etude d'impact des plateformes logistiques sur la logistique urbaine au Maroc. In : *Xème Conférence Internationale: Conception et Production Intégrées*. 2015.
- MOULOUA, Zerouk. *Ordonnements coopératifs pour les chaînes logistiques*. 2007. Thèse de doctorat. Institut National Polytechnique de Lorraine.
- NEW, Stephen J. The scope of supply chain management research. *Supply Chain Management: An International Journal*, 1997.
- NIJKAMP, Peter et SPRONK, Jaap. Interactive multiple goal programming: an evaluation and some results. In: *Multiple Criteria Decision Making Theory and Application*. Springer, Berlin, Heidelberg, 1980. p. 278-293.

- OGRYCZAK, Włodzimierz et RUSZCZYŃSKI, Andrzej. On consistency of stochastic dominance and mean–semideviation models. *Mathematical Programming*, 2001, vol. 89, no 2, p. 217-232.
- OULMOUDNE, Aaziz et BENMOUSSA, Mohamed. Contribution à la compréhension de la confiance et de l'engagement en contexte marocain: cas de la chaîne logistique d'une enseigne de grande distribution alimentaire/Contribution to the understanding of trust and commitment in Moroccan context: the case of a supply chain of food retail brand. *International Journal of Innovation and Applied Studies*, 2017, vol. 19, no 1, p. 140.
- PIRKUL, Hasan. A heuristic solution procedure for the multiconstraint zero- one knapsack problem. *Naval Research Logistics (NRL)*, 1987, vol. 34, no 2, p. 161-172.
- Planification stratégique des réseaux logistiques des entreprises multi-sites, thèse de doctorat, (2005).
- RAHMOUNI, ImadEddine, SOUICI, Khoudir. Amélioration d'une chaîne logistique agro-alimentaire, par l'intégration et la réalisation d'un système de traçabilité basé sur l'IOT et une plateforme web. 2019.
- RODRIGUE, Jean-Paul, SLACK, Brian, et COMTOIS, Claude. GREEN LOGISTICS. IN: HANDBOOK OF LOGISTICS AND SUPPLY-CHAIN MANAGEMENT. 2001.
- ROLLAND, Erik, PIRKUL, Hasan, et GLOVER, Fred. Tabu search for graph partitioning. *Annals of operations research*, 1996, vol. 63, no 2, p. 209-232.
- ROTA-FRANZ, K., THIERRY, C., et BEL, G. Gestion des flux dans les chaînes logistiques. *P. Burlat and J.-P. Campagne, Performance Industrielle et Gestion Des Flux. Paris: Lavoisier*, 2001, p. 153-86.
- SELIM, Hasan, ARAZ, Ceyhun, et OZKARAHAN, Irem. Collaborative production–distribution planning in supply chain: a fuzzy goal programming approach. *Transportation Research Part E: Logistics and Transportation Review*, 2008, vol. 44, no 3, p. 396-419.
- SIMCHI-LEVI, David, KAMINSKY, Philip, SIMCHI-LEVI, Edith, et al. *Designing and managing the supply chain: concepts, strategies and case studies*. Tata McGraw-Hill Education, 2008.
- SPIELBERG, Kurt. Algorithms for the simple plant-location problem with some side conditions. *Operations Research*, 1969, vol. 17, no 1, p. 85-111.
- SRIDHARAN, Ramaswami. The capacitated plant location problem. *European Journal of Operational Research*, 1995, vol. 87, no 2, p. 203-213.
- TANONKOU, Guy Aimé. *Une approche par relaxation lagrangienne pour l'optimisation d'un réseau de distribution : modèles stochastiques et fiables*. 2007. Thèse de doctorat. Université Paul Verlaine-Metz.
- TANSEL, Barbaros C., FRANCIS, Richard L., et LOWE, Timothy J. A biobjective multifacility minimax location problem on a tree network. *Transportation Science*, 1982, vol. 16, no 4, p. 407-429.

- TOUNSI, Jihène. *Modélisation pour la simulation de la chaîne logistique globale dans un environnement de production PME mécatroniques*. 2009. Thèse de doctorat. Université de Savoie.
- TSAY, Andy A., NAHMIAS, Steven, et AGRAWAL, Narendra. Modeling supply chain contracts: A review. *Quantitative models for supply chain management*, 1999, p. 299-336.
- VERTER, Vedat et DINCER, M. Cemal. An integrated evaluation of facility location, capacity acquisition, and technology selection for designing global manufacturing strategies. *European Journal of Operational Research*, 1992, vol. 60, no 1, p. 1-18.
- VINCKE, Philippe. Une méthode interactive en programmation linéaire à plusieurs fonctions économiques. *Revue française d'automatique, informatique, recherche opérationnelle. Recherche opérationnelle*, 1976, vol. 10, no V2, p. 5-20.
- YANG, Jian-Bo et SEN, Pratyush. Preference modelling by estimating local utility functions for multiobjective optimization. *European Journal of Operational Research*, 1996, vol. 95, no 1, p. 115-138.
- ZELENY, M. High technology management. *Human Systems Management*, 1982, vol. 3, no 2, p. 57-59.
- ZIMMERMANN, Martin H. Hydraulic architecture of some diffuse-porous trees. *Canadian Journal of Botany*, 1978, vol. 56, no 18, p. 2286-2295.

Webography

- Esri France. (1999). [En ligne].
- Faq logistique (2002). [En ligne]. *La logistique : définition, périmètre et enjeux* [En ligne Consulté 13/06/2021]. Disponible à l'adresse : <https://www.faq-logistique.com>.
- Horizone. 2019. *Une situation difficile pour l'Algérie*. Disponible à l'adresse : https://hoogwegt.com/media/2596/horizons_april-2019.pdf.
- QGIS. [En ligne]. *Système d'Information Géographique Libre et Open Source*. [En ligne Consulté 12/06/2021]. Disponible à l'adresse : <https://www.qgis.org/fr/site/>.
- [R] <https://abas-erp.com/sites/default/files/Captura%20de%20pantalla%202018-04-25%20a%20las%209.29.06.png>.
- [S] <https://www.sprint-project.com/wp-content/uploads/2018/09/REVERSIO.jpg>.
- <https://www.acteos.fr/wp-content/uploads/2015/03/green-supply-chain.jpg/>.
- [T] <https://youtu.be/gsvOj1rLZGQ>. [En ligne]. Consulté le 28/05/2021.
- [U] https://lh3.googleusercontent.com/proxy/Qha7n4_359gwLlvZ9Family_LoDVn14eGBVImIQjHI0_Yak3B_1WJ1XUQtQuhu5Yf1kNeU8W_Q7graG8DJec705v-IHaW0XuLLxoyEIM82l7adHKbibqU8KEs.
- [V] https://st2.depositphotos.com/4316461/10551/i/950/depositphotos_105512902-stock-photo-seamless-color-pixels-background.jpg
- [W] <https://dpme.ca/wp-content/uploads/2019/04/optimiser-logistique.jpg>

