Problème combiné de localisation, routage et gestion des stocks

Seydou coulibaly

Mai 2016

Abstract

Resumé

Mot-clés : problème d'inventaire, probléme de localisation, problème de routage, heuristique, methodes exactes, CPLEX.

1 Introduction

Très longuement dans la litterature, des problèmes de Localisations, de Routages, ou d'Inventaires ont fut l'objet des thematiques de recherches dans plusieurs laboratoires et entreprise à travers le monde. Divers résultats ont été obtenus pour ces problèmes d'optimisation combinatoire notament l'idée de l'association de plusieurs de ces problèmes (Localisation-Routage, Routage-Inventaire, Localisation-Inventaire) afin d'élaborer toujours de meilleurs décisions strategiques, tactiques et operationnelles pour les entreprises. De la même idée (faire de meilleurs décisions), quelques experts ont eu l'idée de se pencher sur la combinaison des trois problèmes ci-dessus formant ainsi le LIRP (Location, Invetory and Routing Problem), un problème très present dans le domaine de la logistique pour les entreprises industrielles.

Hypothèse I

Dans tout ce qui va suivre, on suppose qu'on dispose d'une usine, d'un ensemble de depôts dispersés dans une region donnée et de plusieurs clients. Les produits sont fabriqués puis acheminer de l'usine aux depôts où ils peuvent être stocker. L'usine peut ne pas être proche des depôts et ces derniers assurent la livraison des produits au clients en fonction de leur demande et de leur état de stocks.

Problème de Localisation

Dans ce problème, on s'interesse à la localisation des depôts à ouvrir parmi un nombre fini de depots, chacun ayant un côut d'installation et un espace (superficie) dans lequel il peut livrer ou fournir des clients. L'objectif est de minimiser le coût total d'installation des sites à ouvrir tout en s'assurant de satisfaire tous les clients sans exception. Il existe plusieurs variantes de ce problème dans la litterature parmi lesquelles des contraintes de capacités sur les depôts à ouvrir.

Problème de Routage

Le problème de routage est le problème dans lequel on cherche à trouver la route ayant la distance la plus minimale afin de l'emprunter. Chaque route passe par un ensemble de points qui peuvent être des depôts ou des clients. Plusieurs variantes en existe comme le TSP (où un visiteur fait le tour d'un ensenble de ville en passant qu'une et une seule fois par une ville donnée), le VRP (Vehicule Routing Problem), CVRP (Capacited Vehicule Routing Problem) , etc. Chaque variante ajoute sa touche de contraintes.

Problème d'Inventaire

Quant au problème d'inventaire, il cherche à decider de la quantité de commande à éffectuer à un temps t pour satisfaire la demande des clients. Cette quantité est contraint par une capacité de stocks du site dans lequel il sera posé . Un coût est associé à chaque quantité de stockage ainsi qu'à chaque commande éffectuée, l'objectif est de trouver cette quantité de commande au temps t minimisant le coût total sur un horizon fini. Plusieurs variantes en existe également.

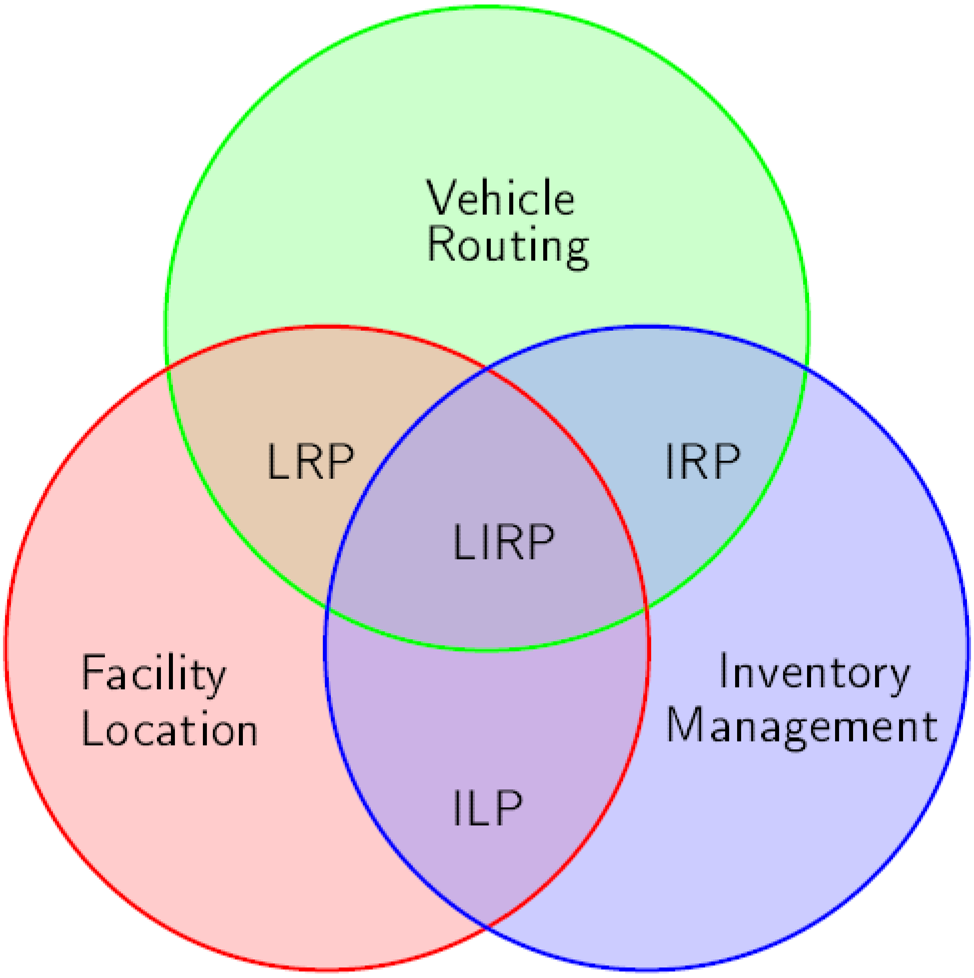

Figure 1: **Location + Inventory + Routing = LIRP []**

2 Modèle LIRP

Le premier modèle mathématiques connu du problème LIRP a été réalisé par *Liu and Lee* (2003) [] puis en 2005 *Liu and Lin* [] ont proposé une metaheuristique hybribe (recherche tabou et recuit simulé) pour le resoudre. Des années évoluant, d'autres chercheurs s'y sont intéressé et ont presenté des solutions nouvelles comme *Ambrosino Scutella* (2005) [], *Ahmadi-Javid and Seddighi* (2012) [], *Guerrero et al* (2013) [] .

2.1 Topologie du modèle

Plusieurs variantes existant pour chacun des sous-problèmes formant le LIRP, un choix s'impose sur les types de contraintes à adopter et le modèle à elaborer. Le schema 2 ci-dessous illustre quelques variantes possible du modèle.

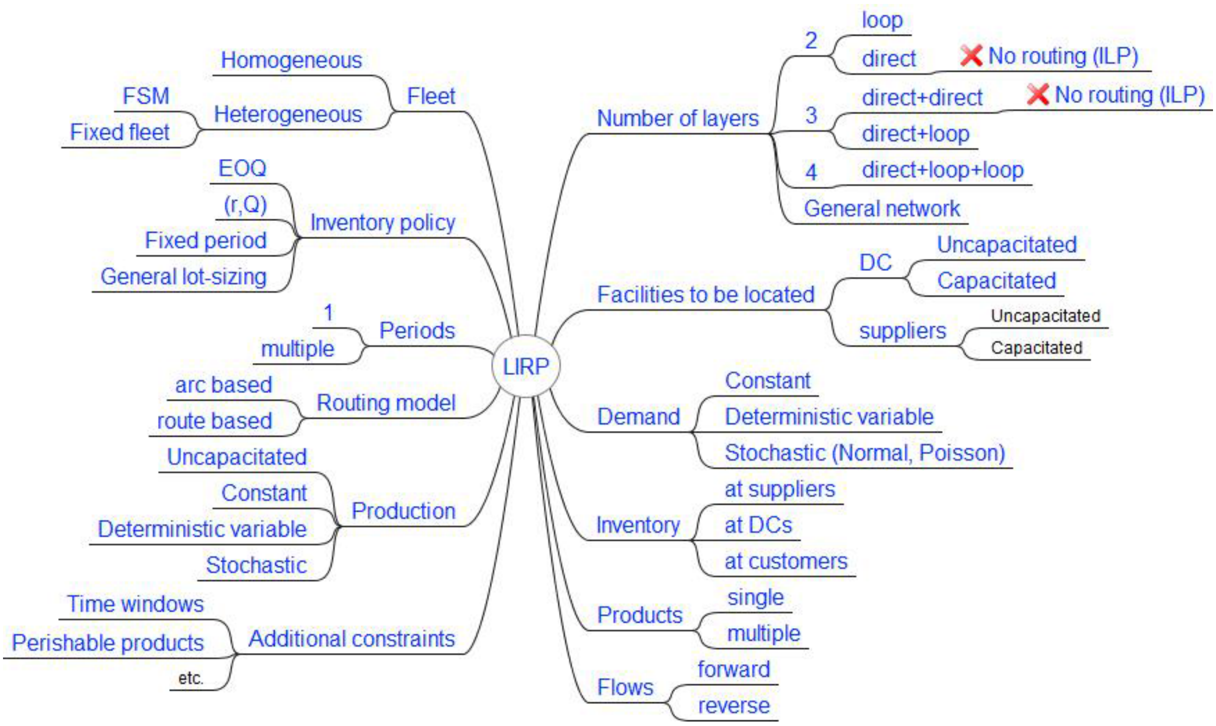

Figure 2: **Topologie du modèle LIRP []**

2.1.1 Hypothèse II

A partir de l'Hypothèse I de l'introduction , on s'itéresse à un LIRP dont le nombre de couches est 3 (Usines, Depôts et Clients). On associe une capacité de stocks à chaque Depôt et client, des demandes variables dans le temps (plusieurs periodes) pour les Clients. Un seul produit transporté par des vehicules homogènes et on suppose également que l'emplacement ou coordonnées des 3 types de sites sont connues.

Graphiquement un exemple de solution de model LIRP est illustré dans le schema ci-dessous 3 à un temps t quelconque où on a une usine (en rouge), quelques depôts ouverts (en rouge), et des clients (en bleu). Les routes utilisées sont representées par des pointillés et tous les clients sont servis par exactement une route et un depôt.

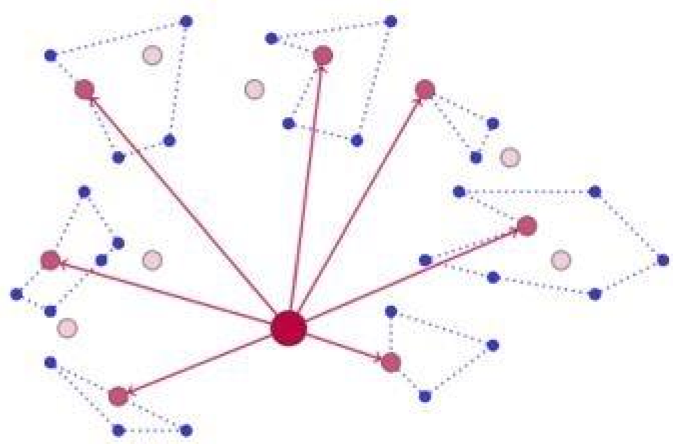

Figure 3: **Exemple de solution possible de LIRP []**

2.2 Formulation mathématiques

Cette formulation est issue du document de Olivier Peton [] introduisant le Problème LIRP.

2.2.1 Notations

Ensemble	Définition
I	ensemble de clients
J	ensemble de centres de distributions ou depôts
P	ensemble d'usines (1 usine ici)
$T = \{0, ..., \|T\|\}$	ensemble de periodes (days)
K	nombre de vehicule
V	ensemble de sites $V = P \cup I \cup J$
V^*	ensemble de depôts et de clients $V^* = I \cup J$
R	ensemble de routes possible

Données	Definition
f_j	Coût fixe d'ouverture du depôt $j \in J$
Q	Capacité du Vehicule (homogènes, qui sont de même taille)
d_i^t	Demande du client $i \in I$ au temps $t \in \{1, ..., \lvert T \rvert\}$
h_i^t	Cout de stockage d'une unité du site $i \in V^*$ au temps $t \in T$
I_{i0}	Inventaire initial $i \in V^*$
c_j	Coût de livraison de depôt $j \in J$
c_r	Coût de livraison de la route $r \in R$
α_{ir}	1 si la route $r \in R$ visite le site $i \in V^*$. 0 sinon
I_i^{max}	Inventaire maximal du site $i \in V^*$

2.2.2 Variables

Variables Binares	
y_j	$\rightarrow$ 1 si le depôt j is selectionné. 0 sinon
z_r^t	$\rightarrow$ 1 si route $r \in R$ est selectionnée au temps $t \in T$. 0 sinon
x_j^t	$\rightarrow$ 1 si le depôt $j \in J$ est selectionné au temps t. 0 sinon
variables Continues	
q_j^t	$\rightarrow$ Quantité delivrée au depôt $j \in J$ au temps $t \in T$
u_{ir}^t	$\rightarrow$ Quantité delivrée par la route $r \in R$ au client $i \in I$ dans le temps $t \in T$
I_i^t	$\rightarrow$ Inventaire du site $i \in I \cup J$ au temps $t \in T$

2.2.3 MIP defintion

$$\min \sum_{j \in J} f_j y_j + \sum_{t \in T} \left(\sum_{j \in J} c_j x_j^t + \sum_{r \in R} c_r z_r^t + \sum_{t \in T} \sum_{i \in V^*} h_i^t I_i^t \right) \tag{1}$$

$$
\begin{aligned}
\text{s.t.} \quad & \sum_{r\in R} \alpha_{ir} z_r^t \leq 1 && \forall i \in I, \forall t \in T && (2)\\
& q_j^t \leq Q x_j^t && \forall j \in J, \forall t \in T && (3)\\
& x_j^t \leq y_j && \forall j \in J, \forall t \in T && (4)\\
& \sum_{i\in I} u_{ir}^t \leq Q z_r^t && \forall r \in R, t \in T && (5)\\
& \sum_{r\in R} z_r^t \leq |K| && \forall t \in T && (6)\\
& z_r^t \leq \sum_{j\in J} \alpha_{jr} y_j && \forall r \in R, \forall t \in T && (7)\\
& I_j^{t-1} + q_j^t = I_j^t + \sum_{r\in R} \alpha_{jr} \left(\sum_{i\in I} u_{ir}^t \right) && \forall j \in J, \forall t \in T && (8)\\
& I_i^t = I_i^{t-1} + \sum_{r\in R} u_{ir}^t - d_i^t && \forall i \in I, \forall t \in T && (9)\\
& I_i^t \leq \min(I_i^{max}, \sum_{t'>t}^{t'<T} d_i^{t'}) && \forall i \in I, \forall t \in T && (10)\\
& I_j^t \leq I_j^{max} y_j && \forall j \in J, \forall t \in T && (11)\\
& u_{ir}^t \leq Q\alpha_{ir} && \forall i \in I, \forall r \in R, \forall t \in T && (12)\\
& q_j^t \geq 0 && \forall j \in J, \forall t \in T && (13)\\
& u_{ir}^t \geq 0 && \forall i \in I, \forall r \in R, \forall t \in T && (14)\\
& y_j \in \{0,1\}, && \forall j \in J && (15)\\
& z_r^t \in \{0,1\}, && \forall r \in R, \forall t \in T && (16)\\
& x_{ij}^t \in \{0,1\}, && \forall i \in I, \forall j \in J, \forall t \in T && (17)\\
& I_i^t \geq 0 && \forall i \in V^*, \forall t \in T && (18)\\
& && && (19)
\end{aligned}
$$

La fonction objective (1) minimise le coût total : d'ouverture de sites, de la gestion de stocks et du coût de routage. La contrainte (2) fait état à ce que chaque client soit visité par au moins une route à un temps T. La contrainte (3) est une contrainte de capacité sur la quantité de distribution de depôt qui depasse pas la quantité maximale d'un vehicule. La contrainte (4) impose qu'un depôt est distribué si et seulement s'il est ouvert. La contrainte (5) est une contrainte de capacité sur la quantité distribuée par une route qui ne depasse pas celle d'un vehicule. La contrainte (6) impose une borne sur le nombre de route à utiliser à un instant T en fonction du nombre de vehicule. La contrainte (7) assure qu'une route passe par un et un seul depot ouvert si elle est selectionnée au temps T. La contrainte (8) est une conservation de flots d'inventaire au niveau des depôts. La contrainte (9) est une conservation de flots d'inventaire au niveau des clients. La contrainte (10) impose une borne superieure sur la quantité de stocks des clients au temps T. La contrainte (11) impose une borne superieure sur la quantité de stocks des depôts au temps T. La contrainte (12) est une contrainte de capacité sur la quantité delivrée à un client connaissant la route et l'instant de livraison.

3 Implémentation

L'implémentation du model ci-dessus est réalisé en C à l'aide du solveur CPLEX. Les instances du problème étudiées lors des tests, sont composées de :

1. Cinq clients, trois depôts en trois periodes
2. Cinq clients, cinq depôts en cinq periodes
3. Cinq clients, cinq depôts en sept periodes
4. Sept clients, cinq depôts en cinq periodes
5. Quinze clients, cinq depôts en cinq periodes

3.1 Les données d'instances

Quant au donnée d'instances, on dispose des elements suivants en entrée :

- Nombre de clients
- Nombre de depôts
- Nombre de periodes
- Nombre de vehicules
- Capacité du vehicule
- Coût par km
- Coût d'installation pour chaque depôt
- Coût de commande pour chaque depôt
- Quantité de stock initial pour chaque depôt et client
- Quantité de stock maximal pour chaque depôt et client
- Demande des clients en fonction du temps
- Coût de stockage d'une unité pour chaque depôt et client en fonction du temps, on suppose que le couts de stockage change en fonction du temps
- Les coordonnées en deux dimensions des clients, depôts et de l'usine.

La dernière étape de la préparation des données est la generation des routes assurant la livraison. Pour s'y faire plusieurs méthodes nous est offert comme la generation de toutes les combinaisons de routes possible, ce qui peut valoir des millions de routes lorsque la taille de l'instances augmente. Par alleurs, on peut utiliser des heuristiques simples et limiter le nombre de routes à generer à un nombre de paramètre definit dans le fichier d'instances. Parmi ces paramètres, on peut citer :

$$\begin{cases} \text{CMAX} & \text{Nombre de clients maximal sur une route} \\ \text{RMAX} & \text{Nombre de route maximale à generer} \\ \text{LMAX} & \text{Coût maximal d'une route} \\ \text{DMAX} & \text{Distance maximal entre un depôt de livraison et son client} \end{cases}$$

Lorqu'une route est generée, on teste si son coût est plus que LMAX et si c'est le cas, la route est abandonée. En outre quand la distance entre un depôt et un client est superieure à DMAX alors il n'existera aucune route reliant ces deux sites.

Toutes les routes

Cette option est interessante uniquement lorsque le paramètre CMAX sur une route n'est pas assez élevé pour pouvoir resoudre le modèle dans un temps raisonnable. Le nombre de routes dans ce contexte vaut m * $(2^{CMAX} - 1)$ dans le pire des cas où m est le nombre de depôts. On genere pour chaque depôt, de 1 client à CMAX clients, une route.

heuristiques

Dans le reste des cas, on genere RMAX routes possible en fonction de CMAX et DMAX. Quatres heuristiques sont élaborer :

- Les routes choisis aleatoirement dans l'ensemble m * $(2^{CMAX} - 1)$
- Les routes les plus courtes de 1 client à CMAX clients en s'arretant lorsque RMAX est atteint
- Les routes les plus longues de CMAX clients à 1 client en s'arretant lorsque RMAX est atteint
- Les routes par paquet : ici, on prend toutes les routes , m * $(2^{CMAX}-1)$ en les fractionnant par paquet de taille RMAX. Les paquets sont construits aleatoirement. On lance le modèle sur chaque paquet de route et l'union des routes selectionnées pour chaque paquet forme “ les routes generées ” sur lesquelles le model est relancé pour avoir de bonnes solutions sinon la solution optimale. On constate que le dernier paquet peut avoir une taille plus petite que RMAX et peut arriver à provoquer la non faisabilité du modèle sur ce dernier.

Remarque : m * $(2^{CMAX} - 1)$ est la taille de toutes les routes possible en fonction de l'instances d'entrée dans le pire des cas possible vu que grâce à DMAX, on élague certains routes de l'ensemble. La valeur de CMAX est toujours plus petit ou egale à n.

3.2 Exécution

4 Conclusion

Conclusion

5 Annexe

Annexe

Rémerciements