
RAPPORT DE PROJET OS11

SUJET : Tournées de Véhicules

Réalisé par :

David MORA MEZA

Yann Audric NOUBENE SIDJE

1. Modélisation du modèle mathématique

À travers les étapes suivantes nous avons proposé un modèle mathématique pour la résolution de ce problème particulier de TSP.

- Définition des données :

n : nombre de clients

P_i : profit engendré par la visite d'un client i

Cx_i : abscisse x d'un nœud ou point

Cy_i : ordonnées y d'un nœud

W : distance maximale

D_{ij} : distance d'un point i à j

M : numéro très grand pour les contraintes de sous tours

- Définition des variables

X_{ij} : variable binaire qui est égale à 1 si on quitte d'un point i à un point j , sinon est égale à 0

t : variable de sous tours qui représente la distance (à l'entrée du nœud)

- Définition de la fonction objective

La fonction objective maximise le profit lors de la visite des différents clients est la suivante

$$\text{Maximise } Z = \sum_{i=0}^n \sum_{j=1, j \neq i}^n P_j X_{ij} \quad (1)$$

- Définition des contraintes

Les différentes contraintes auxquelles est soumise cette fonction objective sont :

Le véhicule n'utilise qu'un arc pour partir d'un point i ou point j pour un autre point

$$\forall_j \in \{0..n\} \sum_{i=0}^n X_{ij} \leq 1 \quad (2)$$

$$\forall_j \in \{0..n\} \sum_{i=0}^n X_{ij} = \sum_{i=0}^n X_{ji} \quad (3)$$

La contrainte d'initialisation

$$\forall_i \in \{1..n\} D_{0i} X_{0i} \leq t_i \quad (4)$$

La contrainte de sous tours

$$\forall_i \in \{1..n\}, \forall_{j \neq i} \in \{0..n\} D_{ij} + t_i \leq t_j + M(1 - X_{ij}) \quad (5)$$

Déclaration des variables.

$$X_{ij} \in \{0,1\}, 0 \leq t_i \leq W$$

Par la suite à partir du logiciel python nous avons construit de manière aléatoire 5 petits cas d'une dizaine de clients que nous avons résolue de manière optimale et les résultats obtenues sont récapitulés dans le tableau suivant :

Scenario	Longueur totale de la tournée	Profit total	Nombre de clients visités
Scenario 1	93.3445	202	8
Scenario 2	98.2169	105	3
Scenario 3	99.6238	155	7
Scenario 4	94.8441	145	5
Scenario 5	99.8016	182	7

Tableau 1: Solution des scénarios avec la méthode exacte

Ce tableau recense la longueur totale, le nombre de clients visités et le profit total des tournées obtenues.

2. Définition des différentes heuristiques

En effet dans la suite de notre projet nous avons défini trois différentes heuristiques comme heuristique gloutonne à savoir :

L'heuristique biggest profit qui elle a pour principal objectifs la maximisation du profit ici on cherchera à aller vers les clients qui nous génère le plus de profit tout en respectant la limite de distance $L=100$ pour nos petits cas.

L'heuristique plus proche voisin PPV qui comme son nom l'indique on se déplacera de vers les clients les plus proches les uns des autres et bien évidemment de la base.

L'heuristique profit/distance qui elle nous fait visiter les clients qui ont le meilleur ratio.

Dans toutes les heuristiques, la distance maximale est vérifiée à chaque itération, et dans le cas où elle n'est pas atteinte, elle continue avec le meilleur client suivant qui l'atteint.

Après avoir défini ces différentes heuristiques et implémenté via Spyder nous les avons appliqués sur nos 5 petits cas d'une dizaine (10) de clients nous avons trouvé la meilleur solution (maximiser le profit) de chaque cas parmi les 3 heuristiques définis et nous les avons comparés avec les solutions obtenues avec gusek.

Les résultats décrits ci-dessous peuvent être visualisés graphiquement dans la *figure 5*, avec les modifications « non ».

- Scenario 1

La meilleure solution est issue ici de l'heuristique **plus proche voisin** avec 8 clients visités un profit de 202 et une distance de 99,84 tandis qu'avec gusek elle nous donne 8 clients visités un profit de 202 et une distance de 93,34.

- Scenario 2

La meilleure solution est issue ici de l'heuristique **profit/distance** avec 4 clients visités un profit de 102 et une distance de 99,23 tandis qu'avec gusek elle nous donne 3 clients visités un profit de 105 et une distance de 98,22.

- Scenario 3

La meilleure solution est issue ici de l'heuristique **profit/distance** et **plus proche voisin** les deux avec 7 clients visités un profit de 155 et une distance de 94,68 et 94,625 tandis qu'avec gusek elle nous donne 7 clients visités un profit de 155 et une distance de 99,624.

- Scenario 4

La meilleure solution est issue ici de l'heuristique **profit/distance** avec 3 clients visités un profit de 115 et une distance de 93,87 tandis qu'avec gusek elle nous donne 5 clients visités un profit de 145 et une distance de 94.84.

- Scenario 5

La meilleure solution est issue ici de l'heuristique **profit/distance** avec 4 clients visités un profit de 158 et une distance de 77,22. tandis qu'avec gusek elle nous donne 7 clients visités un profit de 182 et une distance de 99,80.

3. Calcul de l'écart moyen

Nous avons calculé l'erreur moyenne de tous les scenarios et son écart type entre chaque heuristique et la solution optimale de gusek à travers la formule : $\frac{\text{profits optimales} - \text{profits de l'heuristique}}{\text{profits optimales}}$, et nous avons obtenues les valeurs représentées par la

Figure 1.

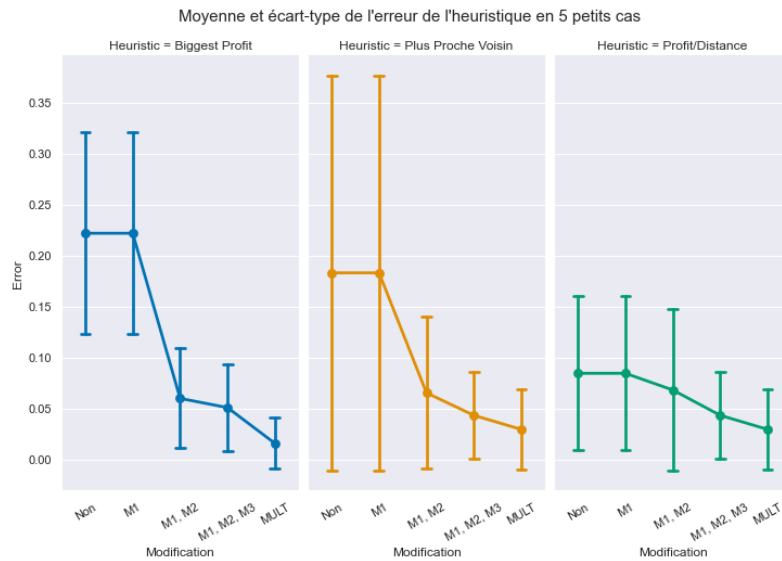


Figure 1: Moyenne et écart-type de l'erreur de l'heuristique en 5 petits cas

Ici nous ne nous intéressons à chaque première partie des graphes (celle qui a pour abscisse **Non**) et nous savons que plus l'erreur est petite plus la solution de l'heuristique se rapproche de la solution optimale de gusek ce qui nous permet donc de voir que **l'heuristique profit/distance** se rapproche beaucoup plus de la solution optimale avec l'écart qui varie entre 0,01 et 0,16.

Scenario	Heuristic	Modification	Distance	Profit	Profit Gusek	Clients Visited	Error	Nombre Clients Visited
1	Biggest Profit	Non	98,80901	164	202	[9, 4, 6, 5, 8]	0,188119	5
1	Plus Proche Voisin	Non	99,83868	202	202	[4, 9, 8, 1, 2, 10, 6, 5]	0	8
1	Profit/Distance	Non	87,93019	191	202	[4, 9, 6, 5, 1, 2, 8]	0,054455	7
2	Biggest Profit	Non	97,40043	96	105	[10, 3]	0,085714	2
2	Plus Proche Voisin	Non	88,96183	91	105	[3, 9, 2]	0,133333	3
2	Profit/Distance	Non	99,23496	102	105	[3, 2, 9, 5]	0,028571	4
3	Biggest Profit	Non	98,2411	95	155	[5, 10, 6, 9]	0,387097	4
3	Plus Proche Voisin	Non	99,62389	155	155	[9, 5, 3, 1, 7, 4, 6]	0	7
3	Profit/Distance	Non	94,68456	155	155	[5, 3, 1, 4, 7, 6, 9]	0	7

4	Biggest Profit	Non	94,94976	108	145	[6, 8, 4]	0,255172	3
4	Plus Proche Voisin	Non	76,66901	70	145	[4, 2, 3, 9]	0,517241	4
4	Profit/Distance	Non	93,87101	115	145	[3, 4, 2, 8, 9]	0,206897	5
5	Biggest Profit	Non	81,41647	147	182	[3, 8, 7, 2]	0,192308	4
5	Plus Proche Voisin	Non	79,56973	134	182	[8, 3, 7, 5, 6]	0,263736	5
5	Profit/Distance	Non	77,21948	158	182	[8, 3, 7, 5, 2]	0,131868	5

Tableau 2: Récapitulatif des différents résultats obtenus à l'issu des heuristiques et gusek

A partir de ces données, il est clair qu'en général, l'heuristique Profit/Distance est la plus performante, et qu'il est donc préférable de la mettre en application pour les grands problèmes où il n'est pas possible d'appliquer une méthode exacte.

4. Réalisation des 5 grands cas

Ici nous avons appliqué les différents heuristiques sur des grands cas qui comportent 100 clients avec une limite de distance $L=250$. Nous avons obtenu les résultats présentés par le tableau suivant :

Scenario	Heuristic	Modification	Distance	Profit	Nombre Clients Visited
6	Biggest Profit	Non	248,2913	538	12
6	Plus Proche Voisin	Non	249,2396	1701	60
6	Profit/Distance	Non	249,626	1827	53
7	Biggest Profit	Non	249,197	476	11
7	Plus Proche Voisin	Non	246,756	1525	62
7	Profit/Distance	Non	249,9983	1487	45
8	Biggest Profit	Non	249,5317	351	8
8	Plus Proche Voisin	Non	243,7164	1390	55
8	Profit/Distance	Non	249,6017	1574	47
9	Biggest Profit	Non	248,9931	465	10
9	Plus Proche Voisin	Non	248,5569	1477	53
9	Profit/Distance	Non	246,3835	1783	52
10	Biggest Profit	Non	248,8279	558	13
10	Plus Proche Voisin	Non	249,1889	1582	62

10	Profit/Distance	Non	246,7072	1564	49
----	-----------------	-----	----------	------	----

Tableau 3: Récapitulatif des résultats de grands cas obtenus à l'issue des heuristiques

A travers ce tableau nous constatons qu'au niveau des scénarios 6, 8 et 9 la meilleure solution est obtenue par l'heuristique profit/distance tandis qu'au niveau des scénarios 7 et 10 elle est obtenue par l'heuristique plus proche voisin.

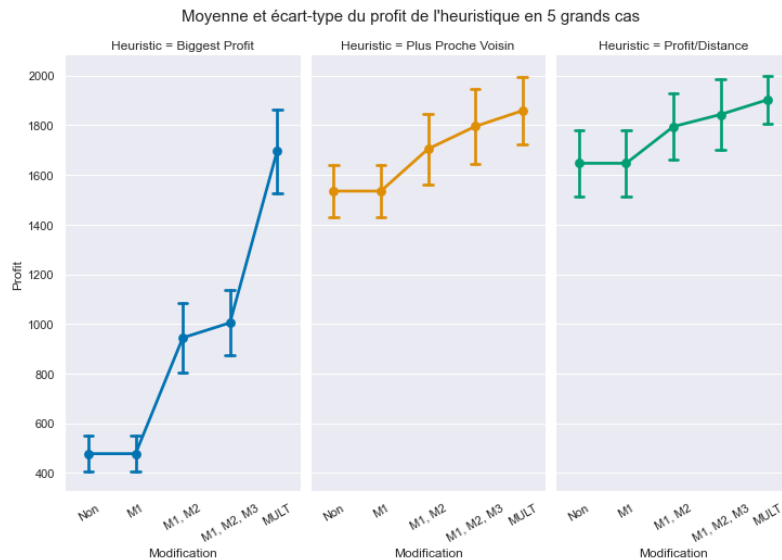


Figure 2: Moyenne et écart-type du profit de l'heuristique en 5 grands cas

Si nous examinons l'application des heuristiques sans modifications dans la *Figure 2*, nous pouvons constater qu'en général, l'heuristique Profit/Distance est plus performante que les 2 autres, et le Biggest Profit étant le moins performant. Ces résultats sont cohérents avec ce que nous avons initialement trouvé avec les 5 petits cas.

5. Méthode de recherche locale

Dans cette partie du projet nous avons effectué une méthode de recherche locale sur les différentes tournées obtenues par les heuristiques afin d'améliorer progressivement chaque tournée obtenue. Cette méthode consistait à tester succinctement trois différentes modifications et de les itérer tant que possible.

Dans les figures 3 et 4, nous pouvons voir la progression dans 2 cas différents, la *Figure 3* pour le petit cas et la *Figure 4* pour le grand cas. Dans ceux-ci, nous pouvons voir visuellement ce qui change dans chacune des modifications locales proposées ci-dessous.

Scenario 4 développé avec Plus Proche Voisin avec les différentes modifications

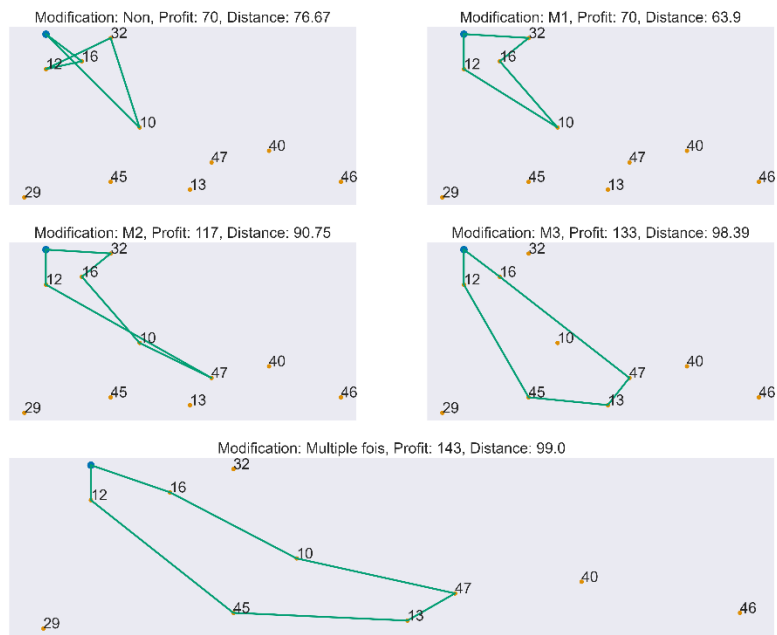


Figure 3: Scenario 4 développé avec Plus Proche Voisin et recherches locales

Scenario 8 développé avec Biggest Profit avec les différentes modifications

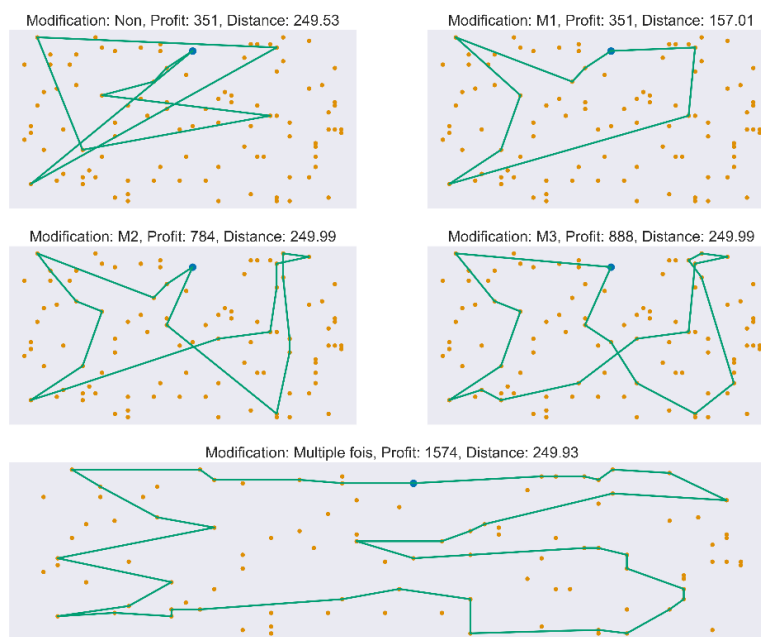


Figure 4: Scenario 8 développé avec Biggest Profit et recherches locales

- Modification M1

Scenario	Heuristic	Modification	Distance
1	Biggest Profit	Non	98,80901
1	Biggest Profit	M1	83,53257

1	Plus Proche Voisin	Non	99,83868
1	Plus Proche Voisin	M1	93,34451
1	Profit/Distance	Non	87,93019
1	Profit/Distance	M1	86,16113

Tableau 4: Comparaison des distances avec l'application du M1

La modification M1 consistait à essayer de réduire de réduire au maximum et tant que possible la longueur de la tournée sans changer les clients visités et donc le profit total. A l'issue de cette première modification sur un de nos différents scenario de petit cas, nous avons obtenu les résultats ci-dessous. Visuellement, on peut voir que le changement consiste à générer un polygone simple avec les points générés. Ceci est cohérent d'un point de vue théorique car aucune jonction de points n'est optimale s'il existe des segments qui se croisent.

Grace à cette première modification appliquée nous constatons bien évidemment à travers le tableau ci-dessus la réduction des distances des tournées sur ce scenario de 10 clients pour toutes les heuristiques.

- Modification M2

Elle s'applique à la suite de la modification M1 lorsqu'on réussit à réduire au max les distances des tournées, M2 nous permet d'insérer un client ou des clients dans les tournées dépendamment de la contrainte de limite de distance définie ce qui nous permettra d'augmenter le profit (objectif du problème). Cette seconde modification sur le même scénario utilisé en M1 nous donne les résultats suivants :

Scenario	Heuristic	Modification	Distance	Profit	Profit Gusek	Nombre Clients Visited
1	Biggest Profit	M1	83,53257	164	202	5
1	Biggest Profit	M1, M2	93,26076	191	202	7
1	Plus Proche Voisin	M1	93,34451	202	202	8
1	Plus Proche Voisin	M1, M2	93,34451	202	202	8
1	Profit/Distance	M1	86,16113	191	202	7
1	Profit/Distance	M1, M2	99,22997	202	202	8

Tableau 5: Comparaison des résultats avec l'application du M2

Nous constatons donc à travers ce tableau de résultat l'amélioration du profit et l'insertion de nouveau clients dans la tournée.

- **Modification M3**

Elle s'applique à la suite de deux précédentes modifications M1 et M2 et consiste à échanger le statut de deux clients, pour chaque client visité, on regarde si on peut l'enlever de la tournée et le remplacer par un client non visité de profit supérieur tout en respectant la limite de distance définie. Cette troisième modification sur le même scénario utilisé en M1 nous donne les résultats suivant :

Scenario	Heuristic	Modification	Distance	Profit	Profit Gusek	Nombre Clients Visited
1	Biggest Profit	M1, M2	93,26076	191	202	7
1	Biggest Profit	M1, M2, M3	93,26076	191	202	7
1	Plus Proche Voisin	M1, M2	93,34451	202	202	8
1	Plus Proche Voisin	M1, M2, M3	93,34451	202	202	8
1	Profit/Distance	M1, M2	99,22997	202	202	8
1	Profit/Distance	M1, M2, M3	99,22997	202	202	8

Tableau 6: Comparaison des distances avec l'application du M3

Pour ce scenario nous constatons que M3 n'apporte pas vraiment d'amélioration quel que soit l'heuristique. Le profit, la distance et le nombre de client visité restent le pratiquement les mêmes. Toutefois, il est possible de constater que, d'une manière générale, dans les figures 1 et 2, il y a une amélioration par rapport à la modification précédente. Et aussi dans les cas spécifiques présentés dans les figures 3 et 4.

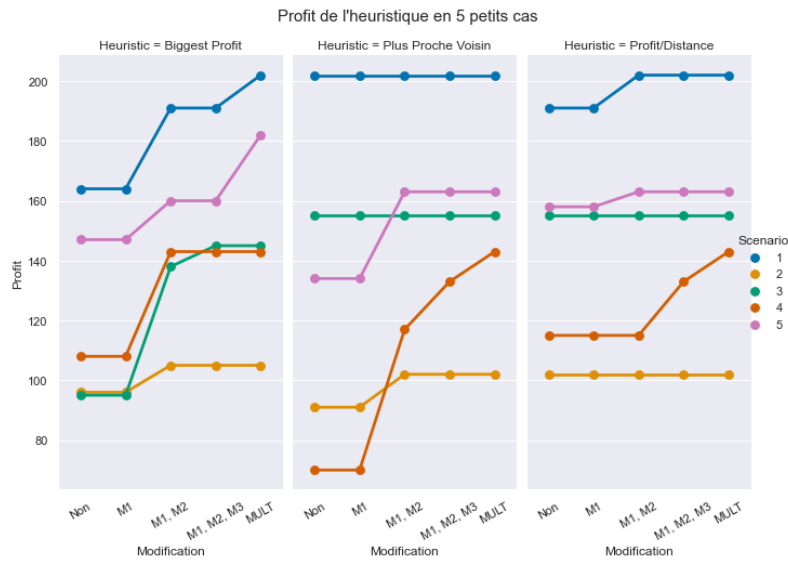


Figure 5: Profit de l'heuristique en 5 petits cas

En outre, dans l'objectif d'une amélioration continue de nos solutions nous avons appliqué la recherche locale plusieurs fois (M1, M2, M3, M1, M2, M3, ...) sur les différents scénarios dans chaque heuristique jusqu'à ce qu'on converge vers des valeurs du profit fixe et optimal. Les figures suivantes nous illustrent les améliorations des profits grâce aux itérations de la recherche locale des différents scénarios (petits cas et grands cas) dans chaque heuristique.

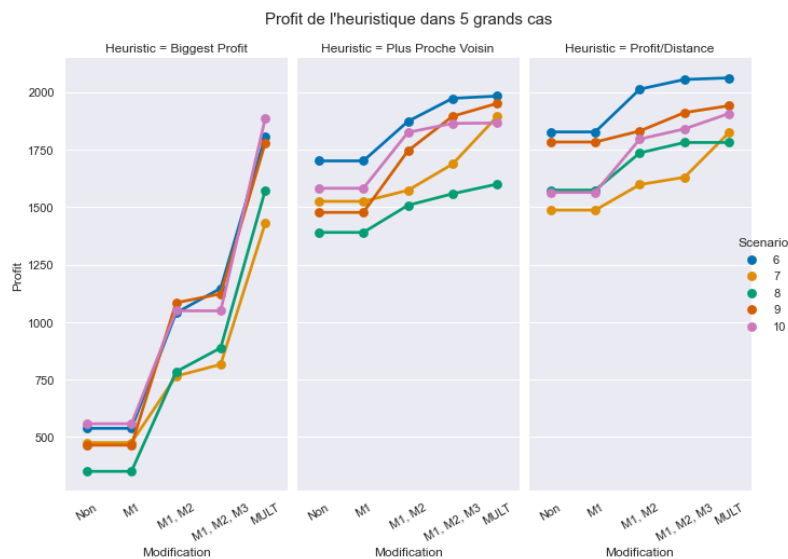


Figure 6: Profit de l'heuristique en 5 grands cas

Après l'analyse du graphe d'amélioration des profits des petits cas nous constatons que quel que soit l'heuristique et dans chaque scénario après une recherche locale multiple les profits évoluent et tendent à converger vers une valeur du profit optimal. Au niveau des grands cas nous constatons aussi une amélioration des profits après la recherche locale dans chaque heuristique mais cependant le profit optimal observe varie d'heuristique selon le scénario.