

Projet Mathématiques - Informatique

Gabriel VALDES-ALONZO
Sami SOGHAIER

Licence 3 - Mathématiques-Informatique
Maria Raquel URENA PEREZ

Juin 2025



Introduction

Le problème du voyageur de commerce [1, 2] (Traveling Salesman Problem en anglais) est un problème classique d'optimisation combinatoire : le problème consiste à trouver le plus court chemin passant une seule fois par une liste de n villes définies, et revenant au point initial.

Le TSP a diverses applications dans plusieurs domaines comme la logistique, séquençage d'ADN, manufacture, et autres. Chaque problème composé d'un ensemble que doit être rempli dans un ordre particulier pourrait être pensé comme le problème TSP, où on cherche un chemin moins coûteux. Le problème peut être résolu exactement si tous les chemins entre villes sont analysés et comparés, mais la quantité de chemins à vérifier augmente rapidement, ce qu'est connu comme un problème NP-Difficile.

Dû à son coût, des algorithmes sont proposés pour trouver des solutions approximatives qui peuvent être optimales ou pas, mais qui résolvent le problème rapidement. Dans ce travail, seulement la version symétrique de l'algorithme sera analysée. Cela veut dire que les distances considérées entre villes sont égales dans les deux sens. Les villes sont aussi considérées toutes interconnectées, donc il existera toujours une connexion directe entre une ville et l'autre.

Nous nous sommes répartis le travail de la façon suivante : après concertation, nous nous sommes mis d'accord sur le fait de prendre deux heuristiques chacun. L'heuristique du plus proche voisin ainsi que 2-opt sont une continuité, comme l'heuristique d'Insertion qui fonctionne avec le recuit simulé. En ce qui concerne le rapport, nous l'écrivons chacun ensemble en développant nos parties respectives.

On va donc vous présenter l'implémentation de ces quatre heuristiques ainsi que les différents avantages et inconvénients de celles-ci. On procédera ensuite à l'analyse des résultats puis on finira par conclure en comparant nos résultats.

Heuristiques implementes

Dans ce contexte qu'on va mettre en pratique des heuristiques afin d'apporter des solutions qui soient le plus correctes possible en minimisant le temps de calcul. On va donc se pencher sur les quatre heuristiques suivantes :

- Nearest Neighbor :
- Heuristique d'Insertion : consiste à assembler un chemin en insérant à chaque étape la ville qui perturbe le moins possible le parcours déjà mis en place.
- 2-opt :
- Recuit Simulé : consiste à utiliser une solution déjà proposée et à l'améliorer en utilisant des solutions temporaires qui ne sont pas optimales pour éviter les minima-locaux.

On a donc décider de travailler sur les quatre heuristiques pré-citées afin d'avoir une solide base de données pour comparer nos résultats.

Résultats

Voisins plus proches

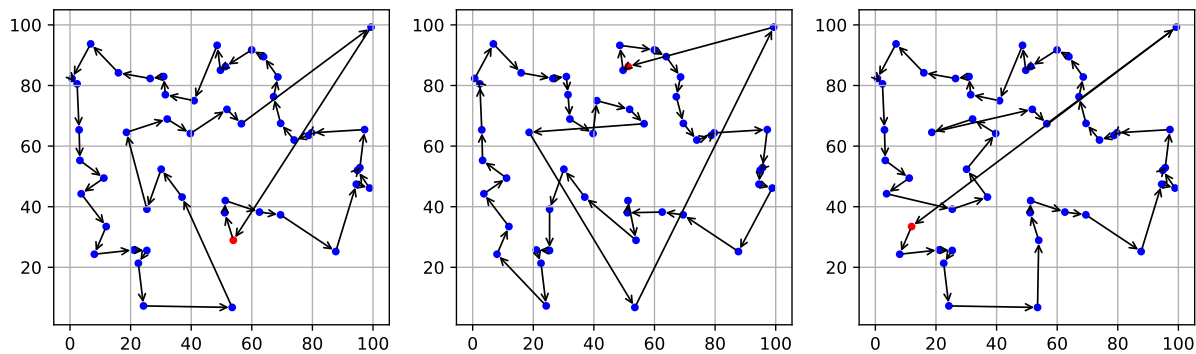


FIGURE 1 – Résultat pour l'algorithme des Voisins plus proches avec trois départs différents (50 villes).

2-opt

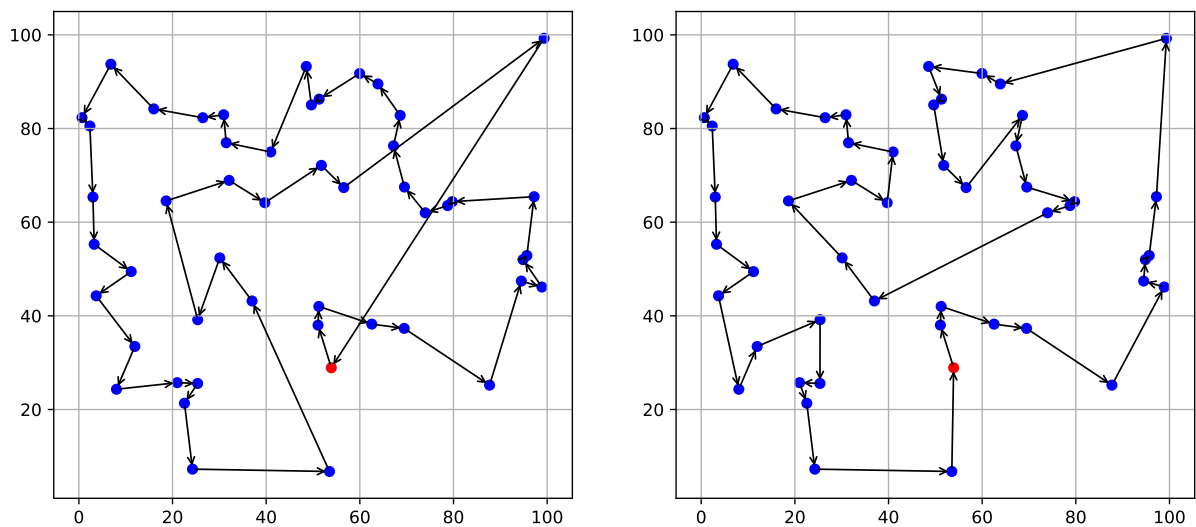


FIGURE 2 – Résultat pour l'algorithme 2-opt et sa comparaison avec le résultat des Voisins plus proches (50 villes).

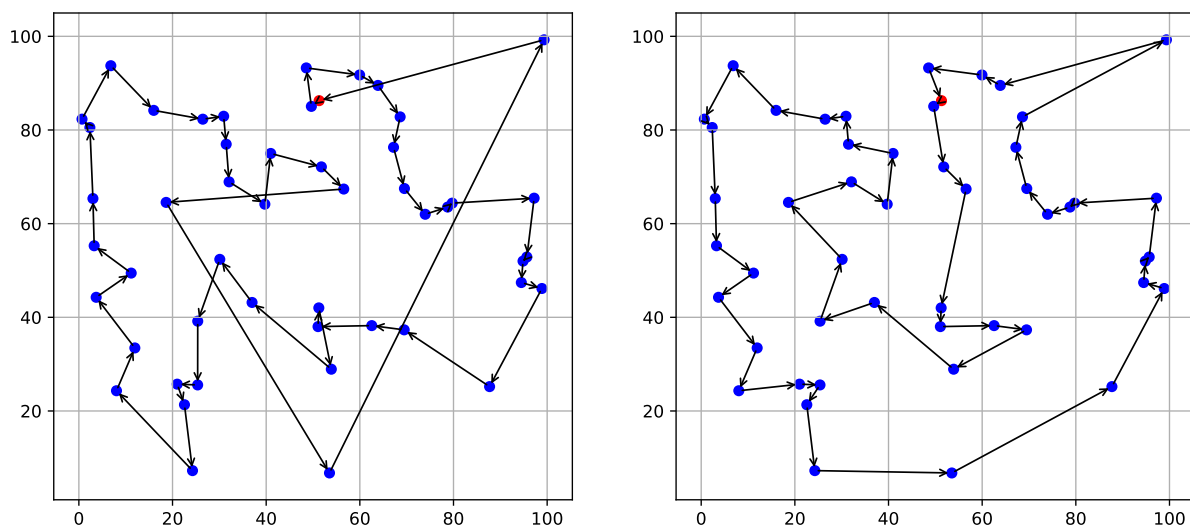


FIGURE 3 – Résultat de 2-opt et sa comparaison avec le résultat des Voisins plus proches pour un départ au milieu.

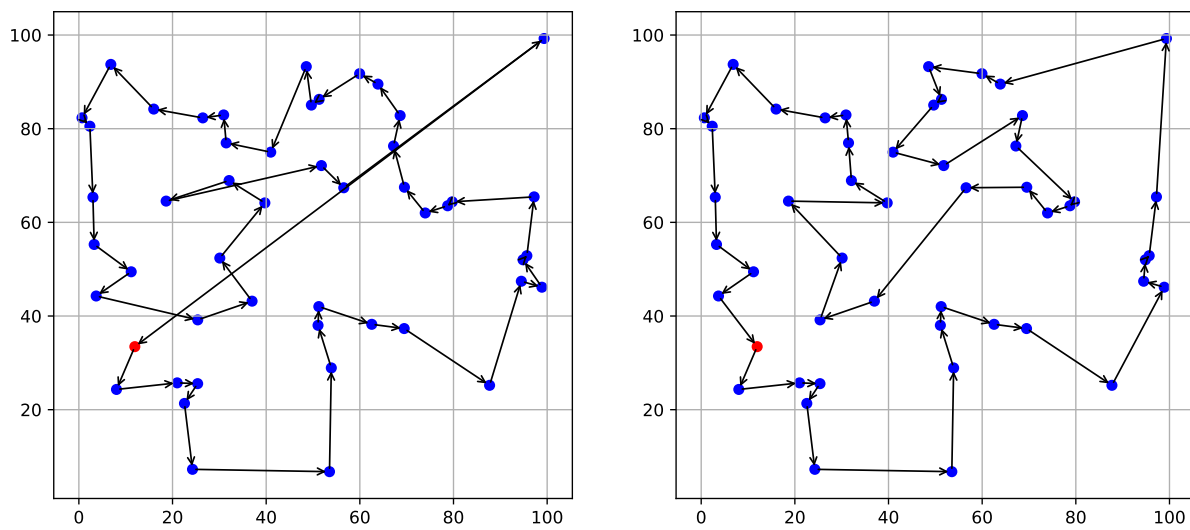


FIGURE 4 – Résultat de 2-opt pour un départ différent. On vérifie que le résultat de 2-opt est dépendant du résultat initial des voisins plus proches, donc le résultat n'est pas fixé.

Heuristique d'insertion

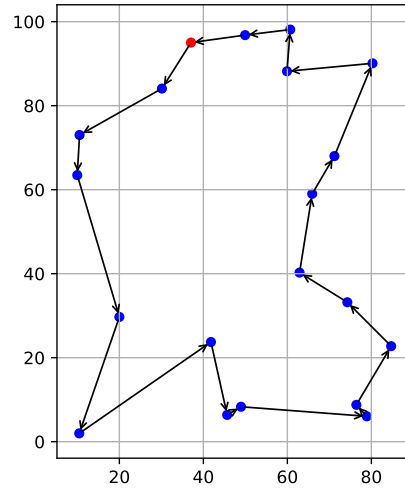


FIGURE 5 – Résultat pour l'heuristique d'insertion avec 20 villes.

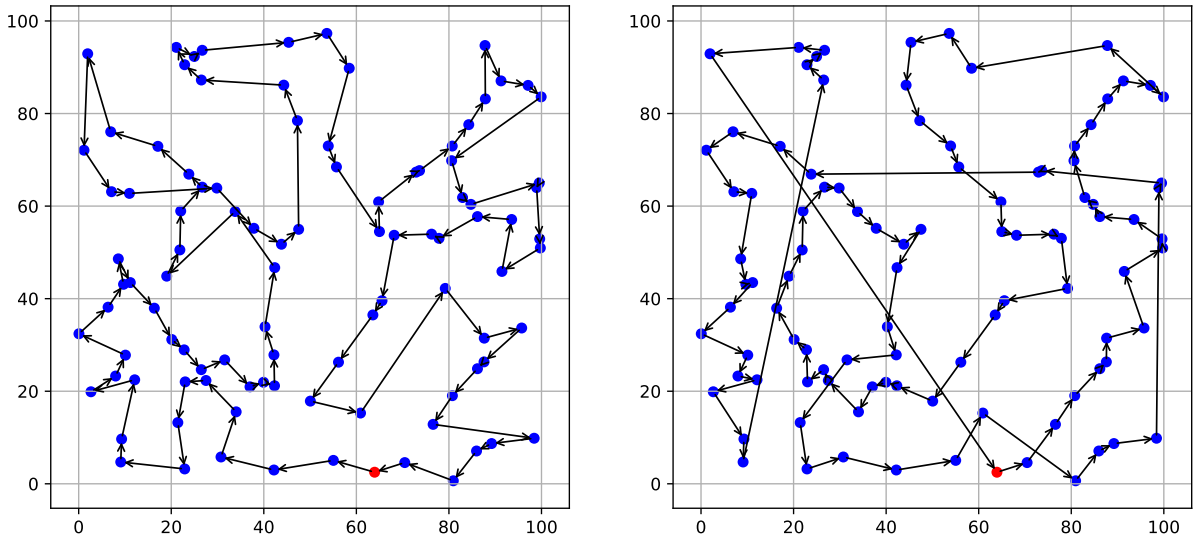


FIGURE 6 – Résultat pour l'heuristique d'insertion avec 100 villes et une comparaison avec le résultat des Voisins plus proches pour le même réseau.

Recuit simulé

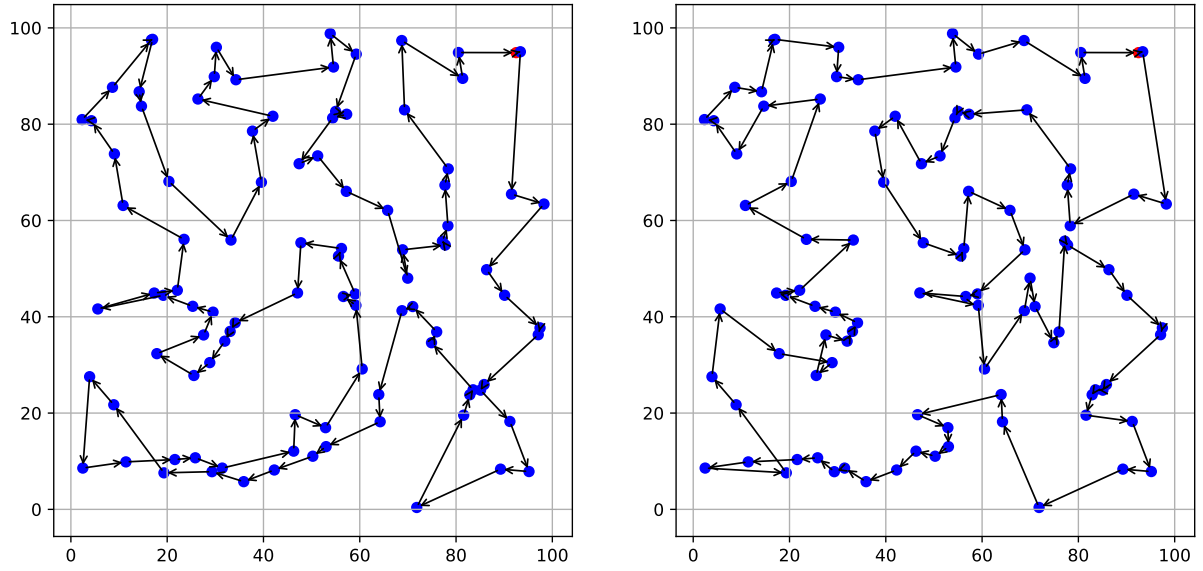


FIGURE 7 – Résultat pour l’heuristique d’insertion (à gauche) et sa amélioration par l’algorithme de Recuit simulé (à droite) avec 100 villes.

Comparaison et complexité

TABLE 1 – Distance obtenu avec les différents algorithmes pour une quantité N de villes.

N villes	VPP	2-opt	Heuristique d’insertion	Recuit simulé
10	286.2587	266.1043	299.9325	266.1043
50	588.5159	549.2188	615.9452	565.7999
100	888.5622	-	853.6044	827.6085
500	2099.8537	-	1936.8251	1936.8251
1000	2916.9891	-	2716.844	2716.8443

FIGURE 8 – Nombre de villes vs. le temps moyenne d'exécution des algorithmes.

Conclusion

Références

- [1] Donald Davendra. *Traveling Salesman Problem, Theory and Applications*. InTech, Janeza Trdine 9, 51000 Rijeka, Croatia, first edition, 2010.
- [2] Karla Hoffman et al. *Traveling Salesman Problem (TSP)Traveling salesman problem*, pages 849–853. Springer US, New York, NY, 2001.