Rapport de Projet — Vehicle Routing Problem (VRP)

# Polytech Lyon – PeiP2 – 2024-2025

# Encadrant :

BONNEVAY Stéphane

# Réalisé par :

SHABANI Eron

GUEZOULI Selmene

DHIF Mohamed-Azize

# Objectif du projet

Ce projet s'inscrit dans le cadre de l'enseignement de recherche opérationnelle et d’optimisation combinatoire. L’objectif est de résoudre le problème de tournées de véhicules (VRP - Vehicle Routing Problem), qui consiste à planifier les itinéraires de plusieurs véhicules devant desservir un ensemble de clients à partir d’un ou plusieurs dépôts, tout en minimisant la distance totale parcourue.  
  
Le projet est implémenté en Python et inclut :  
- une construction initiale de tournées,  
- une optimisation par recuit simulé,  
- une visualisation des itinéraires sur une carte.

# Approche initiale — version fonctionnelle

Dans un premier temps, une version fonctionnelle sans classes a été développée. Cette version permettait de :  
- charger les données du problème (dépôts, clients, demandes),  
- construire les graphes des distances entre clients,  
- générer des tournées de livraison de manière naïve (approche gloutonne),  
- calculer les coûts de parcours.  
  
Cette version nous a permis de bien comprendre la structure des données et les contraintes du problème. Cependant, elle devenait rapidement difficile à maintenir et à étendre (notamment pour ajouter des méthodes d’optimisation ou gérer plusieurs véhicules de manière propre).

# Version finale — refonte orientée objet

Nous avons ensuite développé une version complète orientée objet, beaucoup plus modulaire et évolutive. Elle est composée principalement des classes suivantes :  
- Vehicle : représente un véhicule de livraison, avec son identifiant, sa capacité, sa vitesse, sa couleur, etc.  
- MultiVehicleOptimizer : classe centrale qui gère :  
 - la lecture et l’interprétation des fichiers de données,  
 - la construction du graphe de distances (NetworkX),  
 - la création des tournées initiales (Nearest Neighbor),  
 - l’optimisation par recuit simulé,  
 - la visualisation des itinéraires.

# Optimisation des tournées

L’algorithme d’optimisation repose sur la méthode du recuit simulé (Simulated Annealing), avec :  
- un refroidissement progressif de la température,  
- des déplacements intra- et inter-tournées (échanges, relocalisations),  
- une probabilité d’acceptation décroissante pour les solutions sous-optimales,  
- un mode "super" plus intensif pour de meilleures performances.  
  
Des mécanismes robustes assurent le respect des capacités véhicules et le recalcul cohérent des coûts.

# Visualisation

Chaque tournée générée (initiale et optimisée) est affichée graphiquement :  
- clients en bleu,  
- dépôt en rouge,  
- chaque véhicule a une couleur distincte,  
- les itinéraires sont tracés avec flèches et légende des coûts.  
  
Cela permet une compréhension visuelle et rapide des solutions obtenues.

# Conclusion

Ce projet nous a permis de :  
- modéliser un problème classique de logistique,  
- développer une solution complète en Python,  
- appliquer des techniques d’optimisation avancées,  
- exploiter des bibliothèques puissantes comme NetworkX, Matplotlib et Numpy.  
  
L’évolution du code depuis une version fonctionnelle simple vers une architecture orientée objet nous a offert une base solide pour intégrer des algorithmes avancés comme le recuit simulé.