Master's degree thesis

LOG950 Logistics

Industry 4.0 in civil engineering: delivery route optimization with smart roads

Mariia Shamych

Number of pages including this page: 55

Molde, 01.09.2020

Molde University College
Specialized University in Logistics

Mandatory statement

Each student is responsible for complying with rules and regulations that relate to examinations and to academic work in general. The purpose of the mandatory statement is to make students aware of their responsibility and the consequences of cheating. Failure to complete the statement does not excuse students from their responsibility.

*Please complete the mandatory statement by placing a mark **in each box** for statements 1-6 below.*		
1.	**I/we hereby declare that my/our paper/assignment is my/our own work, and that I/we have not used other sources or received other help than mentioned in the paper/assignment.**	☒
2.	**I/we hereby declare that this paper** 1. Has not been used in any other exam at another department/university/university college 2. Is not referring to the work of others without acknowledgement 3. Is not referring to my/our previous work without acknowledgement 4. Has acknowledged all sources of literature in the text and in the list of references 5. Is not a copy, duplicate or transcript of other work	Mark each box: 1. ☒ 2. ☒ 3. ☒ 4. ☒ 5. ☒
3.	I am/we are aware **that any breach of the above will be considered as cheating, and may result in annulment of the examination and exclusion from all universities and university colleges in Norway for up to one year, according to the** Act relating to Norwegian Universities and University Colleges, section 4-7 and 4-8 **and** Examination regulations section 14 and 15.	☒
4.	**I am/we are aware that all papers/assignments may be checked for plagiarism by a software assisted plagiarism check**	☒
5.	**I am/we are aware that Molde University College will handle all cases of suspected cheating according to prevailing guidelines.**	☒
6.	**I/we are aware of the University College's** rules and regulation for using sources	☒

Personal protection

Personal Data Act

Research projects that processes personal data according to Personal Data Act, should be notified to Data Protection Services (NSD) for consideration.

Have the research project been considered by NSD? ☐yes ☒no

- If yes:

Reference number:

- If no:

I/we hereby declare that the thesis does not contain personal data according to Personal Data Act.: ☒

Act on Medical and Health Research

If the research project is effected by the regulations decided in Act on Medical and Health Research (the Health Research Act), it must be approved in advance by the Regional Committee for Medical and Health Research Ethic (REK) in your region.

Has the research project been considered by REK? ☐yes ☒no

- If yes:

Reference number:

Publication agreement

ECTS credits: 30

Supervisor: Yury Redutskiy

Agreement on electronic publication of master thesis

Author(s) have copyright to the thesis, including the exclusive right to publish the document (The Copyright Act §2).

All theses fulfilling the requirements will be registered and published in Brage HiM, with the approval of the author(s).

Theses with a confidentiality agreement will not be published.

I/we hereby give Molde University College the right to, free of charge, make the thesis available for electronic publication: ☒**yes** ☐**no**

Is there an agreement of confidentiality? ☐**yes** ☒**no**

(A supplementary confidentiality agreement must be filled in)

- If yes:

Can the thesis be online published when the period of confidentiality is expired? ☐**yes** ☐**no**

Date: 01.09.2020

Preface

This thesis is a final work as partial fulfillment for the degree of Master of Science in Logistics Analytics in Molde University College. The thesis is researched and written from December 2019 to August 2020. The work on this thesis has been interesting and challenging at the same time.

Summary

This thesis focuses on analyzing and developing a solution to the extended Two-Echelon Capacitated Vehicle Routing Problem that considers the additional benefits provided by smart roads, such as the environmental and historical traffic data. In particular, this thesis shows how different factors can be carried out with a regression model (on an example of OLS) and be integrated into the extended 2E-CVRP. In the practical part of this work, we apply and validate the developed method to the real data on the example of the post offices of Nova Poshta company in the city of Kyiv. The results of this research show the benefit of using the additional factors in the routing problems and encourage further research in this direction.

Contents

1.0 Introduction

1.1 Background of the study and research problem

The logistics area of research has grown tremendously in the past few decades. The road delivery sector by itself has shown an annual growth rate of 4.6% just between 2014 and 2018. Such a trend can be explained by two factors: globalization and urbanization.

On one side, the globalization of the world economy has led to growing interconnectivity of businesses around the globe. Nowadays, a company in Brazil can easily form a partnership in Belgium, a company established in Spain can move its production facilities to China at will, and trading has never been simpler than it is nowadays. Moreover, the global presence of the companies has contributed to an exponentially growing number of international purchases. In addition, the rise of the Internet has enabled people to place orders on domestic markets regardless of the distance to be traveled to deliver the goods.

On the other side, the worldwide tendency is for the inhabitants of rural areas to further urbanize the surrounding areas of the metropolitan areas. Thus, the cities and their suburbs continue to expand, and so do the roads connecting the city.

Unlike the previous century, nowadays the logistics companies have to deliver much larger amounts of parcels. The growing attention to the service quality on the consumer’s side is influencing the way the delivery problems are solved by transportation companies. One such important issue is the time of the delivery. The consumer is no longer willing to wait for his parcel for more than a few days. Moreover, the client-oriented business model, adopted by the companies has introduced a new type of service - the exact-time-window delivery.

All the mentioned reasons have contributed to the relevance of the chosen project topic. Logistics despite being one of the driving forces of the national economies, remains to be one of the fields with least data used. As stated in the report published by PwC in 2016, the lack of digital training and culture is still the biggest challenge that the logistics companies have to solve. One of the reasons for it is that a company mostly utilizes the roads, road signs and other tools, which they have no direct control over. Many cities have initiatives aimed at digitizing the municipal road infrastructure, and these improvements could not only help the citizens, but also the service companies. In particular, one of the most popular initiatives of the municipal and national authorities is the implementation of smart roads.

1.2 Research setting

In the scope of the outlined goals of the project, a couple of research tasks can be pointed out. The major research task is to develop a model, which will utilize the additional benefits that the smart roads provide. Generally, such benefits can include taking into account weather conditions, exchange of information between the road objects, charging capacities, alert systems, etc.

Provided the scope of the research and the primary purpose of the project, the most important data points that will be vital to the project, is the historical traffic data and all the environmental data available (e.g. weather). Secondly, the availability of some sort of data on the approximate distribution of vehicles by type could also play an important role in the development of an applicable solution.

The data from multiple sources will have to be consistently extracted, properly transformed and loaded in a stable fashion (extract-transform-load or ETL framework). The data required to assess the developed approaches will not be readily available, so another task of generating properly distributed data will also be carried out.

2.0 Research models

2.1 Research problem

The major research question to be studied within the project is the implementation of the smart roads within the setting in logistics. A set of data points relevant to finding the solution to the transportation problem will be determined and collected. Ultimately, the project is aimed at building a traffic management system, equipped with sensors necessary for the computer to process and produce the optimal output, whether it is the best possible route predicted, delivery time prediction, etc.

Within the proposed project, the study will focus on solving the vehicle routing problem for the last-mile delivery companies. The data on delivery locations and environment will be gathered and generated in the city of Kyiv, Ukraine. The data on the road conditions, weather, etc. will be obtained using the Google API. The data on the delivery points will be generated as a uniform distribution of points around the existing locations of the postal offices within the city limits.

In addition, the data from smart-roads sensors will be generated as a normal distribution from the city-wide data available (such as sensors, road quality, traffic, accidents etc.).

2.2 Delivery route optimization

The main purpose of the study is to develop a model which will help the companies performing the last-mile delivery to choose the optimal route for delivery. As outlined by Holland et al. relieving the drivers of such a hassle as optimizing the route can contribute to the efficiency of operations of the delivery department in two ways. Firstly, the drivers will not get distracted from their primary duties. Secondly, the drivers may not necessarily make the best possible decisions on the route to take for delivery.

2.3 Route optimality criteria

To better assess the efficiency of the developed algorithm, an additional set of metrics will have to be developed. Moreover, within the scope of this task, additional measures might be required in order to properly assess the decisions made by the model. For instance, choosing a road with poor conditions might turn out to be beneficial in the short-term, but will incur higher costs combined in the long-run. Thus, multifactor analysis will also have to be performed.

3.0 Background

3.1 The Two-Echelon Capacitated Vehicle Routing Problem

As it was mentioned earlier, within the proposed project, the study will focus on solving the vehicle routing problem for the last-mile delivery companies. The Two-Echelon Capacitated Vehicle Routing Problem (2E-CVRP) was chosen as an optimal method to solve the above task. 2E-CVRP is an expansion of the classical CVRP where the delivery depot-customers runs through transitional depots (called satellites). As in CVRP, the goal is to supply cargo to clients with known demands, minimizing the total delivery cost in the respect of vehicle capacity constraints. In the following sections, information about this method and its advantages will be highlighted in more details.

3.1.1 Introduction to the Problem

Freight transportation includes two major distribution strategies: straight shipping and multi-echelon distribution. Direct shipping means, that vehicles start the distribution from a depot and bring indicated cargo directly to the destination. Whilst the multi-echelon systems provide freight delivery from the depot to the customers through intermediate stations. The growth in freight traffic, as well as the necessity of taking into account such important factors as environmental impacts and road congestion, have focused research on multi-echelon distribution systems. In two-echelon distribution systems, cargo is delivered to an intermediate station, and from there to clients.

Multi-echelon systems represented in the literature usually view the routing problem at the final stage of the transportation system, whilst a simplified routing problem is considered at upper levels. This, the routing costs in the upper levels are frequently undervalued and decision-makers cannot directly utilize the solutions received from the models. In addition, in the past decade multi-echelon systems have been represented by practitioners in various fields:

- Logistics concerns and express supply service crews. These operators usually are in a two-echelon system. They use own offices as intermediate states for organizing the delivering of cargo and compose vehicles that will be used in order to transport the cargo to another transitional point (airport, regional center, etc.) or to the final appointment.

- Multimodal cargo transportation. The amount of intermodal logistics centers in the countries of central and south-west Europe significantly increased in the past decade. This is a good instance of two or more echelons of cargo distribution. In a classical road-train multimodal allocation the cargo goes from the manufacturer to a logistic center by road and then loaded on a train which goes to other logistics center. The train is discharged and cargo goes by road to its final appointment.
- Grocery and supermarkets products distribution. Big firms use hypermarkets as transitional storage points to serve smaller shops and supermarkets of the same mark in city zones.
- Spare parts distribution in the automotive market. Some firms use couriers and other actors to supply their spare parts. For example, the case of General motors and FIAT, which spare parts TNT distribute from the companys' plants to the garages. In a like manner, Bridgeston uses distribution scheme organization in zones and sub-zones in order to reduce the transportation times and cut the size of the storage spaces.
- E-commerce and homeward delivery services. This is modern opportunities presented by the growth of e-commerce and the homeward delivery services proposed by some supermarkets and shops like SEARS, in some large cities implies the existence of transitional depots in order to optimize the delivery operations.
- City logistics. Recently, researchers started to explore city areas as a single frame, stopping to consider each company, cargo, and car apart. Rather, one should consider that all movements and stakeholders are elements of an integrated logistics frame. This means the coordination of providers, carriers, and transfers as well as the consolidation of loads of several clients and carriers into the same eco cars and the adopted distribution system is characteristically a two-echelon scheme.

The key goal of this section is to introduce the Two-Echelon Capacitated Vehicle Routing Problem (2E-CVRP), one of the simplest types of Multi-Echelon Vehicle Routing Problem. In 2E-CVRP, the cargo delivery from the depot to the clients is managed by transportation cargo through transitional depots. Thus, the transportation system is decomposed into two steps: on the 1st step the depot is connecting to transitional depots and on the 2nd one connecting the transitional depots to the clients. The goal is to minimize the total transportation cost of the movements on both steps. It is necessary to consider the constraints on the maximum capacity of the automobiles and the transitional depots, while the timing of the deliveries can be disregarded.

3.1.2 Literature review

There are various distribution strategies in cargo distribution. The most developed strategy include direct transportation: cargo arrives to customers directly from the depot. This strategy is not the best one and applying of a two-echelon distribution scheme can help to optimize few features as the amount of automobiles, the transportation costs and loading rate.

In the literature the multi-echelon system, and the two-echelon system notably, cite mostly to inventory and supply chain problems (Sand, 1981; Svoronos, 1988; Verrijdt and de Kok, 1995). These problems more focusing on the manufacture and supply chain management questions and do not using an explicit routing approach for the various stages.

The first practical application of a two-tier allocation network optimizing the worldwide shipping expenses is due to Crainic, Ricciardi and Storchi and is connected to the city logistics zone (Crainic et al., 2004). A two-tier cargo distribution scheme was developed for overpopulated urban zones, using little transitional platforms, called satellites, as intermediate states for the cargo allocation. This system was developed for a concrete case research and is not generalize all such systems. Generally, the generalization of such a system has not been formulated yet.

To the knowledge, VRP variations are considered in their single-level variant, even if VRP contains a big class of combinatorial problems. In the following, the main results in this area is presented.

The Capacitated VRP (CVRP) is the case where park of same vehicles are considered. The CVRP objective is to minimize transportation expenses below the capacity constraints on the high cargo amount what can be loaded on each vehicle. Distance Constrained VRP (DVRP) is a version of the problem, where ones add supplementary constraint on the top route length that every vehicle can cover. These two categories of constraints can be joint in the Distance Constrained Capacitated VRP (DCVRP). This version of VRP is the most usually considered, and last studies have developed nice heuristic algorithms. Accurate methods can solve comparatively small samples and their computational times are extremely unstable (Cordeau et al., 2004). Accurate algorithms are mostly utilized to define optimal solutions of the test samples, whilst heuristic methods usually used for realistic applications. The savings algorithm of Clarke and Wright (Clarke and Wright, 1964) is one of the first and majority used heuristics. Tabu Search (Cordeau et al., 1997, 2001), which has been improved in last studies with the help of hybrid algorithms, is one of the best methods to solve CVRP. This method is usually combining Tabu Search with Genetic Algorithms (Perboli et al., 2006; Prins, 2004; Mester and Br¨aysy, 2005).

Due to the fact that transportation means a high rate of the cost added to shipment in some market segments, VRP has become a main problem in the area of logistics and cargo transportation. Thus, using of computerized methods for transportation constantly results in substantial economy from 5% to 20% in summary, as reported at Toth and Vigo, 2002. Commonly, in practicable applications, issue can be different to the one introduced here. Lot versions of VRP have been developed in order to present these applications. The most famous options are multi-depot VRP (MDVRP), VRP with time windows (VRP-TW) and VRP with pickups and deliveries (VRP-PD) (for a survey, see Toth and Vigo, 2002). Let's mark only one option of VRP where satellites facilities are explicitly considered, the VRP with Satellites facilities (VRPSF). In this option, facilities that are used to supplement vehicles during a path are introduced. Satellite replenishment permit drivers to continue delivering of cargo without obligatorily returning to the main depot when it is possible. This situation arises first of all in the spreading of petrol and others trade items and one do not use satellites as depots to decrease the transportation expenses (Angelelli and Speranza, 2002; Bard et al., 1998).

3.1.3 2E-CVRP as part of The Multi-Echelon Vehicle Routing Problems

Intelligent Transportation Systems technologies and operations research-based methodologies allow to optimize the frame, planning, management, and procedure of City Logistics systems (Crainic and Gendreau, forthcoming; Taniguchi et al., 2001).
Consolidation actions taking place at so-called Distribution Centers (DCs). When sizes of DCs are smaller than the depot and the cargo can be stored for a short time, the DCs are also called satellite platforms, or just satellites. Longhaul transportation cars go to Satellite and unload freight. Afterthat loads are consolidated into smaller cars that afterwards deliver cargo to the final appointment. Alike system can be determined for opposite movements.
Multi-Echelon Vehicle Routing Problems solve the cargo delivering task from the depot to the customers via rerouting and consolidating cargo across various transitional satellites. Common purpose of the operations is to guarantee an efficient and inexpensive procedure, whilst the demand is supplies on schedule and the total cost of transportation is minimized. Generally, capacity constraints on the vehicles are considered.

In more detail, in the Multi-Echelon Vehicle Routing Problems common transportation network can be decay into $k \geq 2$ levels:

- the 1st level connects depots and the 1st-level satellites;
- $k - 2$ medium levels connecting satellites;
- and the last level, where cargo is delivered to the final clients from the satellites.

Every transportation step has its proper vehicle park to control the delivery process. Vehicles that are set to a one level cannot be reassigned to another.
Two-Echelon Vehicle Routing Problem is the most spread variant of Multi-Echelon Vehicle Routing Problem that is used in practice. Given method consider just two levels of delivery. From a physical point of view, a Two-Echelon Capacitated Vehicle Routing system works as follows:

- cargo come to external area, the depot, where it is consolidated into the 1st-level vehicles, unless it is already carried into a fully-loaded 1st-level truck;
- each 1st-level vehicle goes to a subset of satellites which will be determined by the model and then it will go back to the depot;
- cargo is moved from 1st-level vehicles to 2nd-level vehicles at a satellite;
- each 2nd-level vehicle performs a path to serve marked customers, and then goes to a satellite for its following series of operations. The 2nd-level vehicles comeback to their original satellite.

At each level of 2E-CVRP all vehicles allocable to that level have idem fixed capacity. The aim is to serve clients through minimizing total transportation cost and satisfying vehicles capacity constraints.

Single depot and a fixed number of capacitated satellites exist. All clients demand is fixed and known in advance. In addition, zero time windows are defined for the deliveries and satellite operations. All client demands must be satisfied. For the 2nd level, the demand of every client is smaller than each vehicle capacity and demands cannot be split into numerous paths of the same level.

For the 1st level two extra distribution strategies can be considered. First one is that satellite is served by only one 1st-level vehicle and the demand which pass via the satellite cannot be split into various 1st-level vehicles. Second strategy says that satellite can be served by more of one 1st-level vehicle, thus every demand of the satellite can be split.

In order to formalize 2E-CVRP and describe a mathematical statement able to solve small and medium-sized examples, the depot v_0, the set of transitional depots called satellites V_s and the set of clients V_c should be defined.

Let n_s denote the number of satellites and n_c the number of clients. The depot is the launching point of the cargo. The satellites may be capacitated. The clients are the final destinations of the cargo and each client i has assosiated a demand d_i, i.e. the number of cargo which has to be supplied to that client. The demand of every client cannot be split between various vehicles, neither at the 1st nor at the 2nd level.

Cargo distribution cannot be managed by direct shipping from the depot to the client, but cargo must be consolidated from the depot to a satellite and further, from the satellite, is delivered to the final client. This unreservedly determines a two-level transportation system: the 1st level connect depot and satellites and the 2nd level connect satellites with final customers.

Example of 2E-CVRP transportation network is shown in Figure 1.

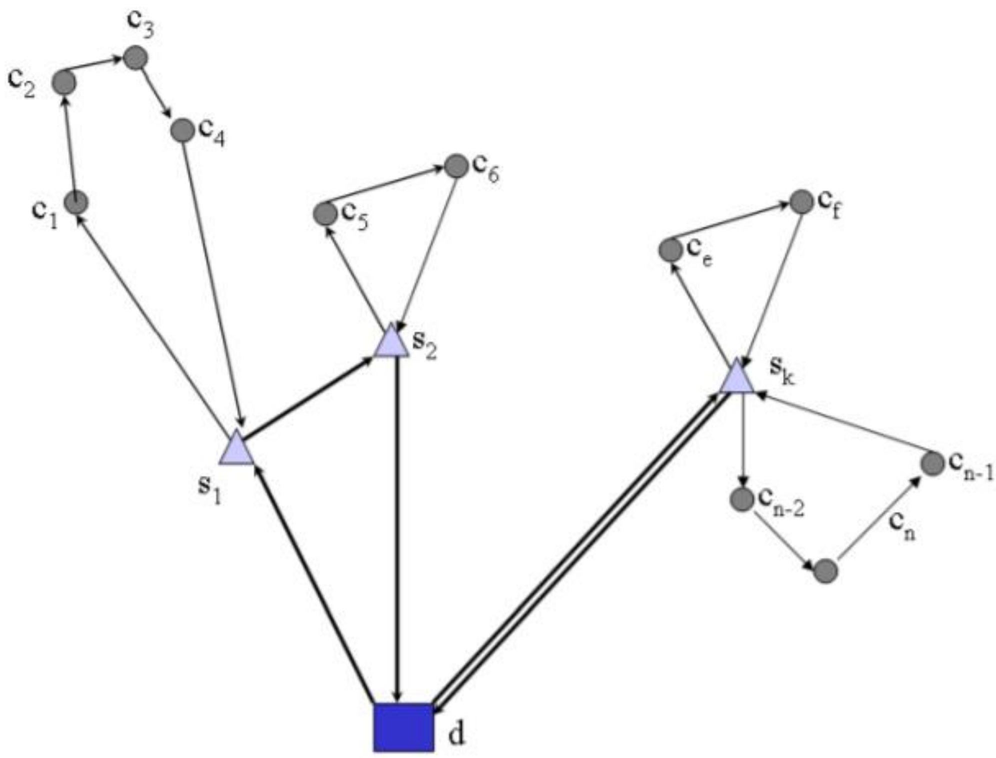


Figure 1. Example of 2E-CVRP transportation network

Let's define the arc *(i, j)* as direct path which connects node i and node j. If both of the nodes are satellites or one is the satellite and the other one is depot, let's define the arc as belonging to the 1st-level network, whilst if both nodes are clients or one is a client and other is satellite, the arc belongs to the 2nd-level network.

Let's consider that exist only one type of cargo, i.e. the capacity of cargo which belongs to various clients can be stored together and loaded in the same vehicle, for both the 1st and the 2nd-level vehicles. In addition, the vehicles which belongs to the same level should have the same capacity.

A 1st-level path is defined as a path made by a 1st-level vehicle which begins from the depot, attend one or more satellites and ends in the depot. A 2nd-level path is a path made by a 2nd-level vehicle and it starts from a satellite, attend one or more clients and ends in the same satellite.

3.1.4 A flow-based model for 2E-CVRP

According to the determination of 2E-CVRP, if the assignments among clients and satellites are defined, the problem decreases to 1 + n_s CVRP (1 for the 1st-level and n_s for the 2nd-level).

Let's determine five sets of variables, that can be separated in three groups:

- The first group introduce the arc usage variables. One determine two sets of such variables, one for every level. The variable x_{ij} is an integer variable of the 1st-level routing and is equal to the number of 1st-level vehicles using arc *(i, j)*. The variable y_{ij}^k is a binary variable of the 2nd-level routing and is equal to 1 if a 2nd-level vehicle makes a route starting from satellite *k* and goes from node *i* to node *j* and 0 otherwise.

Table 1. Definitions and notations for 2E-CVRP model

$V_0 = \{v_0\}$	Depot
$V_s = \{v_{s1}, v_{s2}, \dots, v_{s_{n_s}}\}$	Set of satellites
$V_c = \{v_{c1}, v_{c2}, \dots, v_{c_{n_c}}\}$	Set of customers
n_s	Number of satellites
n_c	Number of customers
m_1	Number of the 1st-level vehicles

m_2	Number of the 2nd-level vehicles
K^1	Capacity of the vehicles for the 1st level
K^2	Capacity of the vehicles for the 2nd level
d_i	Demand required by customer *i*
c_{ij}	Cost of the arc *(i, j)*
S_k	Cost for loading/unloading operations of a unit of freight in satellite *k*

- The second group of variables assign every client to one satellite and are used for connection the two transportation levels. More accurately, one determine z_{kj} as a binary variable that is equal to 1 if cargo to be delivered to client *j* is consolidated in satellite *k* and 0 otherwise.
- The third group of variables which is splitted in two subsets (each one represents separate level), introduce the cargo flow passing via each arc. One determine the cargo stream as a variable Q^1_{ij} for the 1st-level and Q^2_{ijk} for the 2nd level, where k performs the satellite where the cargo transits across. Both variables are continuous.

With an eye to lighten the model formulation, one determine the subsidiary quantity

$$D_k = \sum_{j \in V_c} d_j z_{kj} \text{ , } \forall \ k \ \epsilon V_s \quad (1)$$

which performs the cargo passing across every satellite *k*.

The model to minimize the total cost of the system may be formulated as follows:

$$\min \sum_{i,j \in V_0 \cup V_s} c_{ij} x_{ij} + \sum_{k \in V_s} \sum_{i,j \in V_s \cup V_c} c_{ij} y^k_{ij} + \sum_{k \in V_s} S_k D_k$$

Subject to

$$\sum_{i \in V_s} x_{oi} \leq m_1 \quad (2)$$

$$\sum_{j \in V_s \cup V_0, j \neq k} x_{jk} = \sum_{i \in V_s \cup V_0, i \neq k} x_{ki} \ \ \forall k \in V_s \cup V_o \quad (3)$$

$$\sum_{k \in V_s} \sum_{j \in V_c} y^k_{kj} \leq m_2 \quad (4)$$

$$\sum_{i\in V_s, j\in V_c} y_{ij}^k = \sum_{i\in V_s, j\in V_c} y_{ji}^k \,\forall k \in V_s \tag{5}$$

$$\sum_{i\in V_s, i\neq j} Q_{ij}^1 - \sum_{i\in V_s, i\neq j} Q_{ji}^1 = \begin{cases} D_j \; j \; is \; not \; the \; depot \\ \sum_{i\in V_c} -d_i \quad otherwise \end{cases} \;\forall j \in V_s \cup V_o \tag{6}$$

$$Q_{ij}^1 \leq K^1 x_{ij} \forall i, j \in V_s \cup V_0, i \neq j \tag{7}$$

$$\sum_{i\in V_c, i\neq j} Q_{ijk}^2 - \sum_{i\in V_c, i\neq j} Q_{jik}^2 = \begin{cases} z_{kj} d_j \; j \; is \; not \; the \; satellite \\ -D_j \qquad\qquad otherwise \end{cases} \;\forall j \in V_c \cup V_s, \forall k \in V_s \tag{8}$$

$$Q_{ijk}^2 \leq K^2 y_{ij}^k \;\; \forall i, j \in V_s \cup V_c, i \neq j, \forall k \in V_s \tag{9}$$

$$\sum_{i\in V_s} Q_{iv_0}^1 = 0 \tag{10}$$

$$\sum_{j\in V_c} Q_{jkk}^2 = 0 \;\forall k \in V_s \tag{11}$$

$$y_{ij}^k \leq z_{kj} \;\; \forall i \in V_s \cup V_c, \forall j \in V_c, \forall k \in V_s \tag{12}$$

$$y_{ji}^k \leq z_{kj} \;\; \forall i \in V_s, \forall j \in V_c, \forall k \in V_s \tag{13}$$

$$\sum_{i\in V_s \cup V_c} y_{ij}^k = z_{kj} \;\; \forall k \in V_s, \forall j \in V_c, i \neq k \tag{14}$$

$$\sum_{i\in V_s} y_{ji}^k = z_{kj} \;\; \forall k \in V_s, \forall j \in V_c, i \neq k \tag{15}$$

$$\sum_{i\in V_s} z_{ij} = 1 \;\; \forall j \in V_c \tag{16}$$

$$y_{ij}^k \leq \sum_{l\in V_s \cup V_o} x_{kl} \;\;\; \forall k \in V_s, \forall i, j \in V_c \tag{17}$$

$$y_{ij}^k \in \{0,1\}, z_{kj} \in \{0,1\}, \forall k \in V_s \cup V_o, \forall i, j \in V_c \tag{18}$$

$$x_{kj} \in Z^+, \forall k, j \in V_s \cup V_o \tag{19}$$

$$Q_{ij}^1 \geq 0, \forall i, j \in V_s \cup V_0, Q_{ijk}^2 \geq 0, \forall i, j \in V_s \cup V_c, \forall k \in V_s \tag{20}$$

The objective function minimizes the sum of the traveling and handling operations costs. Constraints (3) show, for $k = v_0$, that every 1st-level path starts and ends at the depot, whilst when k is a satellite, impose the balance of vehicles which are entering and leaving that satellite. Constraints (5) make each 2nd-level path to staer and end to one satellite and the balance of vehicles entering and leaving every cliebt. The amount of the pathes on every level must not exceed the amount of vehicles for that level, as inflicted by constraints (2) and (4).

Constraints (6) and (8) specify that the streams balance on every node is peer to the demand of this node, besides the depot, where the exit stream is peer to the summary demand of the clients, and for the satellites at the 2nd-level, where the stream is peer to the demand (unknown) assigned to the satellites. Desides, constraints (6) and (8) disallow the existence of subtours which are not containing the depot or a satellite, respectively. In fact, every node get an amount of stream which is equal to its demand, averting the existence of subtours. Consider, for example, that a subtour is exist between the nodes *i, j* and *k* at the 1st level. It is easy to verify that, in such a occasion, does not exist any value for the variables Q^1_{ij} , Q^1_{jk} and Q^1_{ki} satisfying the constraints (6) and (8). The capacity constraints are formulated in (7) and (9), for the 1st-level and the 2nd-level, accordingly.

Constraints (10) and (11) do not permit residual streams in the routes, making the revert stream of every route to the depot (1st-level) and to every satellite (2nd-level) equal to 0.

Constraints (12) and (13) specify that a client *j* is served by a satellite *k* (z_{kj} = 1) only if it obtains cargo from that satellite (y^k_{ij} = 1). Constraint (16) assigns every client to one and only one satellite, whilst constraints (14) and (15) specify that there is only one 2nd-level route passing via every client and unite the two levels. Constraints (17) permit to start a 2nd-level route from a satellite *k* only if a 1st-level route has served it.

3.2 Simulated annealing

To solve the VRP problem in this paper, it was decided to use Simulated annealing (SA) probabilistic technique, which is designed for approximating the global optimum of a given function.

The state of some physical systems, and the function *E(s)* to be minimized, is analogous to the internal energy of the system in that state. The goal is to bring the system, from an arbitrary initial state, to a state with the minimum possible energy.

3.2.1 The basic iteration

The simulated annealing heuristic considers some adjacent state s* of the actual state s at each step, and probabilistically decides whether it is necessary to move the system to state s* or it is better stay in-state s. These probabilities finally conduct the system to go to the states of lower energy. Characteristically this step is repeated before the system arrives a state which is quite good for the application, or before a given calculation stock has been depleted.

3.2.2 The neighbour of a state

Solution optimization includes estimating the neighbours of a state of the problem, which are new states produced via conservatively shifting a present state. For instance, in the TSP every state is characteristically determined as a transposition of the cities to be visited, thus the neighbors of every state are the kit of transpositions produced by changing any two of these cities. A well-defined method in which states change to create neighboring states is called "move," and different moves produce different sets of neighboring states. These moves commonly result in least changes of the last state, in an effort to enhance the solution via iteratively improving its parts (such as the city connections in the TSP).

Such heuristics as hill climbing, that move by discovering better neighbour after better neighbour and ends when they have achieved a result which has no neighbours which are better solutions, cannot ensure to bring any of the present better solutions – their result may lightly be simly a local optimum, whilst the valid best solution would be a global optimum and it could differs. Metaheuristics utilize the neighbours as a variant to research space of the solution, and whilst they choose better neighbours, they likewise take worse neighbours with an eye to get away getting stuck in local optima; they can detect the global optimum if run for a quite long amount of time.

3.2.3 Simulated annealing algorithm

The basic SA algorithm for a generic optimization problem can be outlined as follows. Let S be the set of all possible feasible solutions, and $f : S \rightarrow R$ be the objective function to be minimized. An optimal solution s∗ is a solution in S such that $f(s*) \leq f(s)$ holds for all $s \in S$.

SA is an iterative technique that design a path of solutions $s^{(0)}$, ... , $s^{(k)}$ in *S*. At each iteration, SA considers moving from the actual feasible solution $s^{(i)}$ to a candidate *new* feasible solution s_{new}. Let $\Delta(s^{(i)},\ s_{new}) = f(s_{new}) - f(s^{(i)})$ be the objective function *worsening* when moving from $s^{(i)}$, to s_{new} – positive if s_{new} is strictly worse than $s^{(i)}$. The hallmark of SA is that worsening moves are not forbidden but accepted with a certain acceptance probability $p(s^{(i)}, s_{new}, T)$ that depends on the amount of worsening $\Delta(s^{(i)}, s_{new})$ and on a parameter $T > 0$ called temperature. A typical way to calculate the acceptance probability is via Metropolis' formula:

$$p\left(s^{(i)}, s_{new}, T\right) = \begin{cases} e^{-\Delta\left(s^{(i)},\, s_{new}\right)/T} & if\ \Delta\left(s^{(i)},\ s_{new}\right) \geq 0 \\ 1 & if\ \Delta\left(s^{(i)},\ s_{new}\right) \leq 0 \end{cases}$$

Thus, the probability of accepting a worsening move is large if the temperature T is large and the amount of worsening $\Delta(s^{(i)}, s')$ > 0 is small. Note that the probability is 1 when $\Delta(s^{(i)}, s') \leq 0$, meaning that improving moves are always accepted by the SA algorhitm.

Temperature T is a crucial parameter: it is initialized to a specific value T_0, and iteratively decreased within the SA performance so as to make worsening moves less and less likely in the conclusive iterations. A ordinary update formula for T is based on a cooling factor $\alpha \in (0, 1)$ and reads $T = \alpha \cdot T$; typical ranges for α are 0.95–0.99 (if cooling is applied at each SA iteration) or 0.7–0.8 (if cooling is only applied at the end of a "computational epoch", i.e., after some SA iterations with an invariable temperature).

The basic SA scheme is outlined in Figure 2; more advanced implementations are possible, e.g., the temperature can be restored multiple times to the initial value.

Input: function f to be minimized, initial temperature $T_0 > 0$, cooling factor $\alpha \in (0, 1)$, number of iterations $nIter$
Output: the very last solution $s^{(nIter)}$

1: Compute an initial solution $s^{(0)}$ and initialize $T = T_0$
2: **for** $i = 0, \ldots, nIter - 1$ **do**
3:     Pick a new tentative solution s_{new} in a convenient neighborhood $\mathcal{N}(s^{(i)})$ of $s^{(i)}$
4:     $worsening = f(s_{new}) - f(s^{(i)})$
5:     $prob = e^{-worsening/T}$
6:     **if** $random(0, 1) < prob$ **then**
7:         $s^{(i+1)} = s_{new}$
8:     **else**
9:         $s^{(i+1)} = s^{(i)}$
10:     **end if**
11:     $T = \alpha \cdot T$
12: **end for**

Figure 2. The basic SA scheme

At Step 6, *random(0, 1)* is a pseudo-random value uniformly distributed in [0,1]. Mark that, at Step 5, the acceptance probability *prob* becomes larger than 1 in case *worsening* < 0, meaning that improving moves are anytime accepted (as required).

4.0 Model development

4.1 Introduction

This chapter shows the application of the researched solution of the Two-Echelon VRP in the city of Kyiv for delivery to the post offices of the Nova Poshta company.

All the data is initialized, computed, and retrieved from scratch using Google Maps API and other open-source solutions. All the code described in this work is reproducible using the present "data" folder.

4.2 Initial setup and data collection using Google Maps API

In order to reasonably limit the size of the urban area for which the study is conducted let's define the borders of the researched area of interest that will cover the Kyiv city. This area is defined using the following coordinates of latitude and longitude, where `bl` stands for bottom left coordinate and `tr` - top right one. (Figure 3)

```
bl = (50.381347, 30.442942)
tl = (50.533611, 30.442942)
tr = (50.533611, 30.65)
br = (50.381347, 30.65)
```

Figure 3. Coordinates of the borders of the researched rectangular area

The diagonals of the region of interest are approximately 22.4 km long.

In order to get the most results using Google Maps API, we need to specify circles halfway to the center with a diameter of half the diagonal (equal to 5.6 km). This will result in some overlap, but the whole area of interest will be covered (Figure 4).

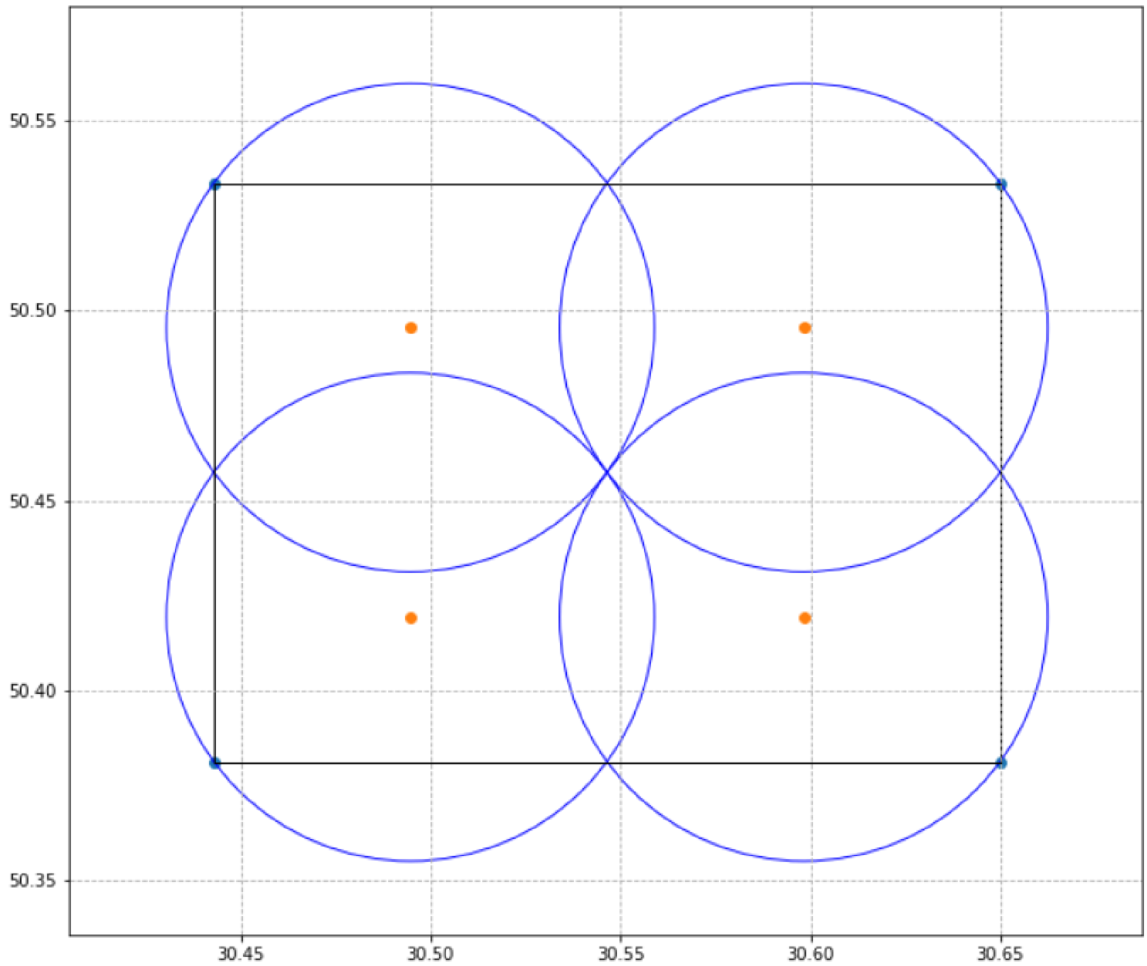


Figure 4. Circles halfway to the center with the diameter of half the diagonal

A total of 135 Nova Poshta post offices were found within the 4 circles using the code represented in Figure 5. And a total of 118 post offices were found within the considered rectangle defined in Figure 3.

```
try:
    df = pd.read_csv('parsed_np.csv')
except:
    gmaps = googlemaps.Client(key = api_key)
    del df
    for center in centers.keys():
        print(centers[center])
        has_next_page = True
        first_page = True
        while has_next_page:
            if first_page:
                response = gmaps.places(query = "nova poshta", location = centers[center], radius = radius)
                first_page = False
            else:
                response = gmaps.places(query = "nova poshta", location = centers[center], radius = radius, page_token = next_pa
ge_token)
            try:
                next_page_token = response["next_page_token"]
                print(next_page_token)
            except:
                has_next_page = False
            try:
                df = pd.concat([df, pd.DataFrame(response['results'])])
            except:
                df = pd.DataFrame(response['results'])
            print(df.shape)
            sleep(5)
    df.drop_duplicates(subset = 'formatted_address', inplace = True)
df.head()
```

Figure 5. Process of generating points with post offices within the specified circles

The final dataset of 116 offices is plotted in Figure 6.

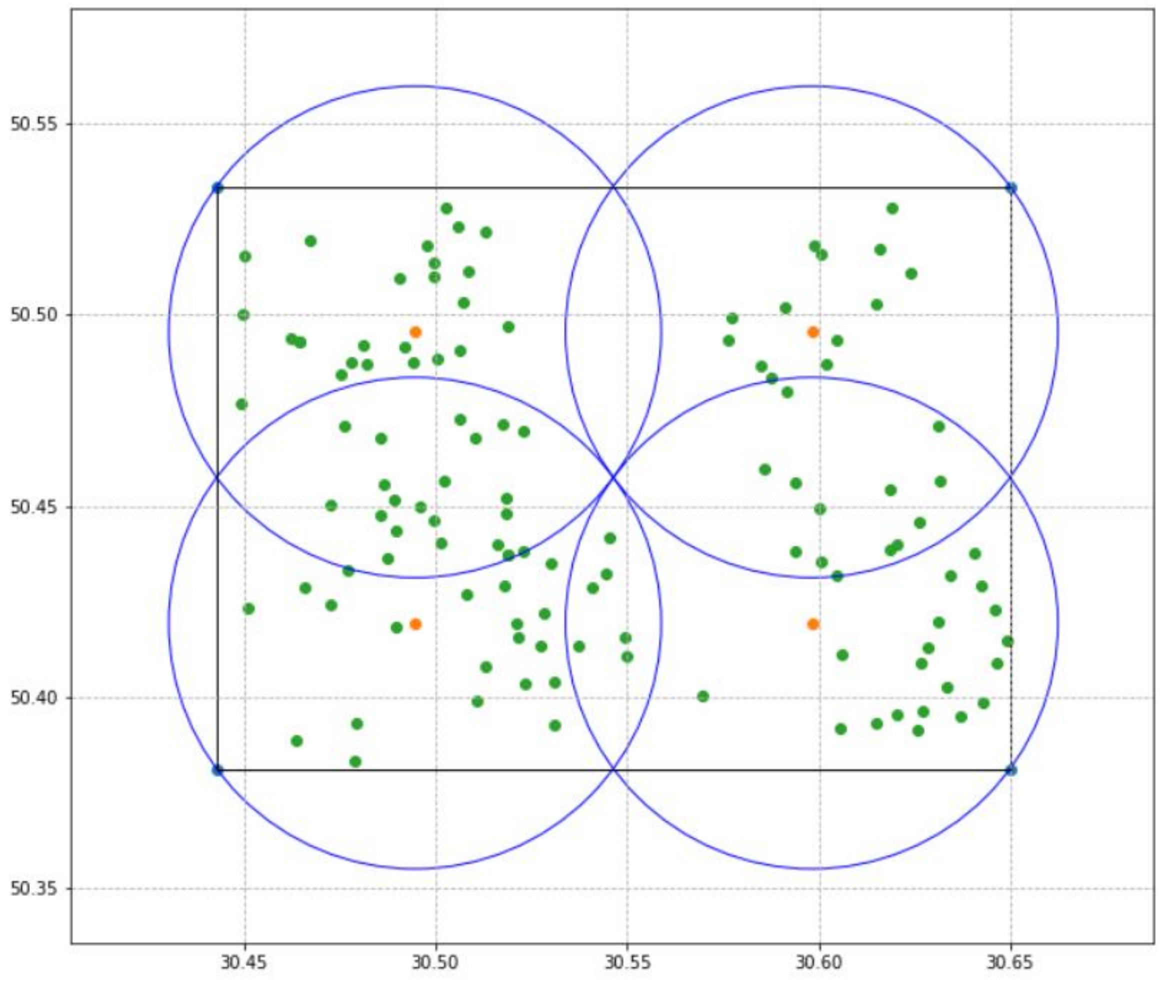


Figure 6. Post offices location points for further research

4.3 Clustering. Part 1

The next necessary step to solve the Two-Echelon Capacitated Vehicle Routing Problem should be the determination of the optimal satellite locations that will meet customer demand in the two-tier operating scheme. This can be done by applying cluster analysis for customer locations – branches of Nova Poshta.

In accordance with the general practice of urban freight transportation, it was decided to use the K-means clustering algorithm to determine the resulting centroids of clusters as the location of satellites. As the K-means algorithm uses the Euclidean metric for the pairwise distance, thus there will most likely be non-optimal cluster assignments.

Let's analyze what number of clusters is the best fit for the given data points.

4.3.1 Silhouette analysis for K-means (Euclidian distances)

Let's perform the Silhouette analysis with cluster numbers from 2 to 6. In the silhouette analysis, 1 is trivial and everything above 6 is not viable for the considered task and not optimal. In Figures 7, 8, 9, 10, 11 Silhouette analysis for K-means clustering with cluster numbers from 2 to 6 is performed.

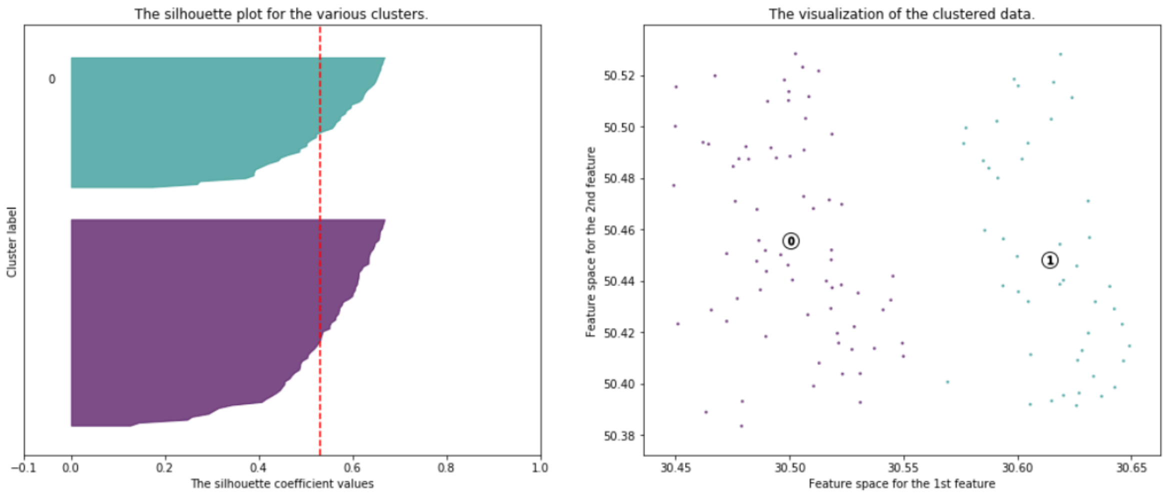


Figure 7. Silhouette analysis for K-means clustering on sample data with 2 clusters

Silhouette analysis for the two clusters determined the average value of the silhouette equal to 0.5297815679982419.

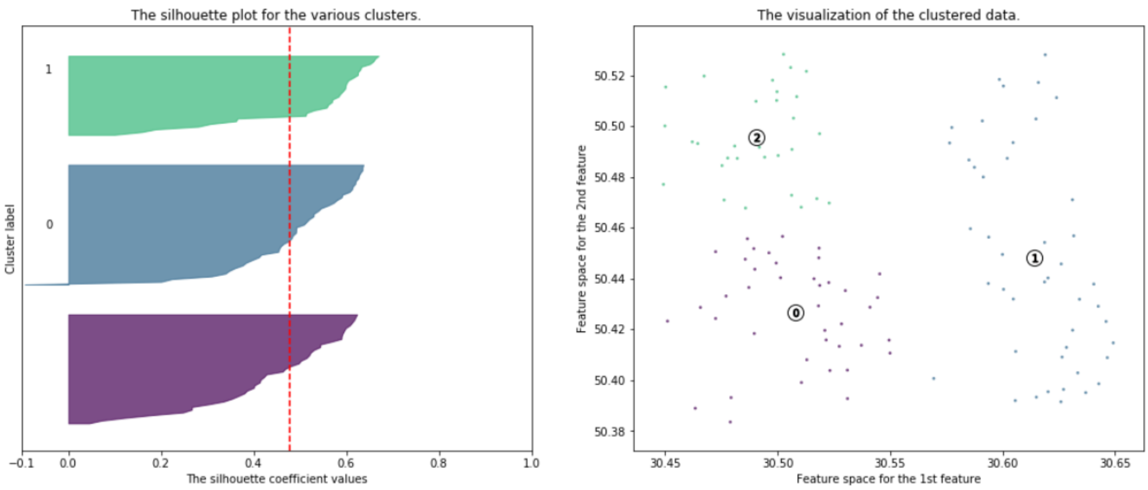


Figure 8. Silhouette analysis for K-means clustering on sample data with 3 clusters

Silhouette analysis for three clusters determined the average value of the silhouette equal to 0.47723816246385137.

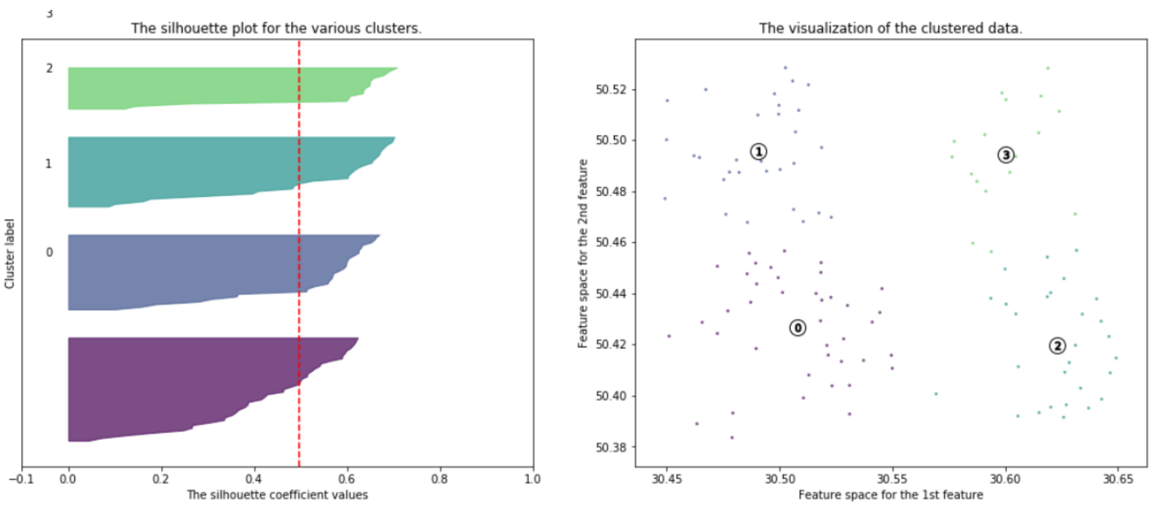


Figure 9. Silhouette analysis for K-means clustering on sample data with 4 clusters

Silhouette analysis for four clusters determined the average value of the silhouette equal to 0.4962916615629401.

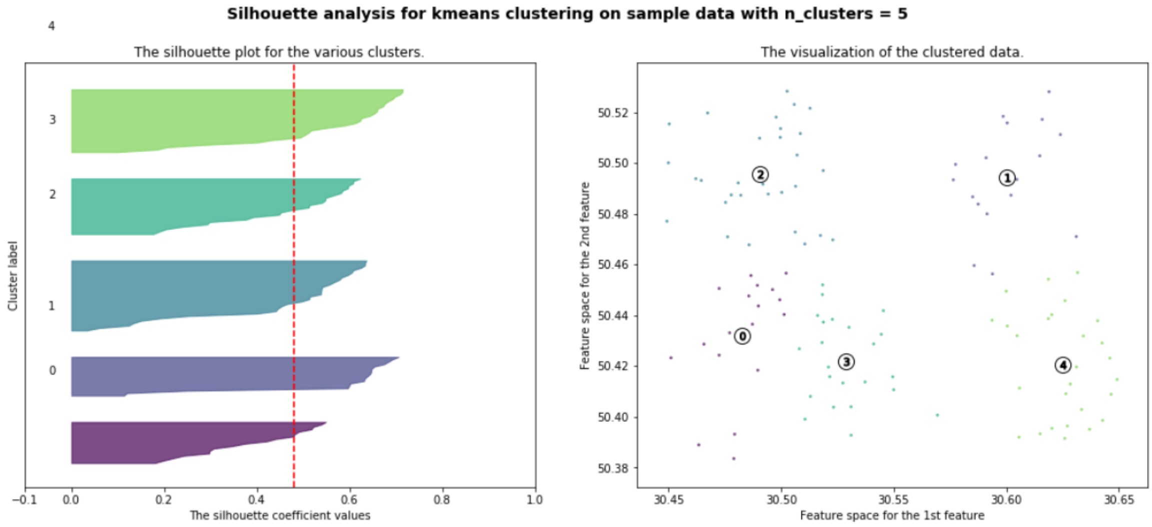


Figure 10. Silhouette analysis for K-means clustering on sample data with 5 clusters

Silhouette analysis for five clusters determined the average value of the silhouette equal to 0.47978788533205874.

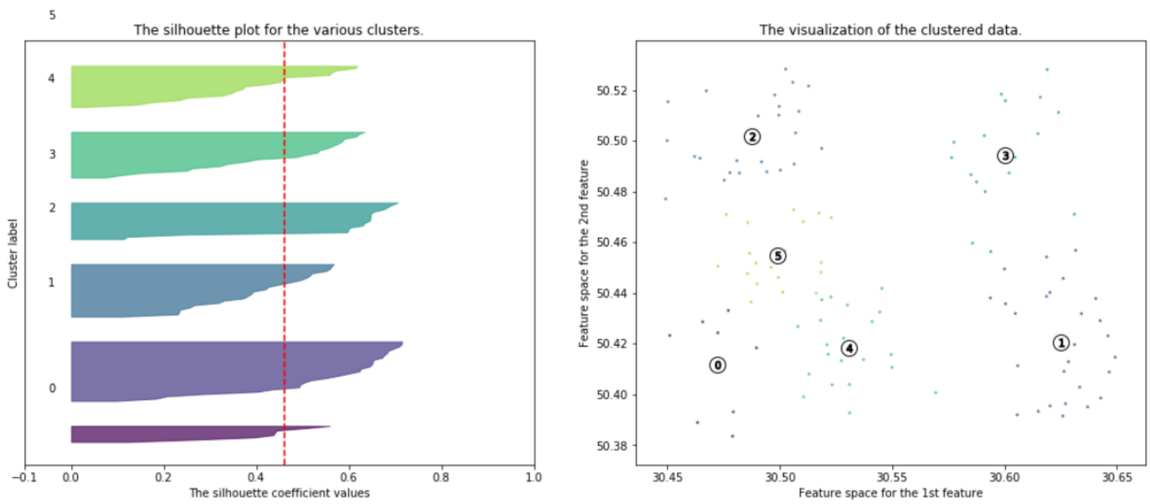


Figure 11. Silhouette analysis for K-means clustering on sample data with 6 clusters

Silhouette analysis for six clusters determined the average value of the silhouette equal to 0.4604912961970378.

As one can see from Silhouette analysis, the optimal number of clusters should be equal to 2, and this can be explained by the natural split of Kyiv on two sides with the Dnipro river. Concerning the domain knowledge of the data and for sake of non-oversimplification of the problem it was decided to proceed with four clusters as the second-best option with the silhouette value equal to 0.49.

4.3.2 K-means with 4 clusters

As it can be seen in the Figure 12 as a result of four cluster K-means with the Euclidean distance, one obtained two centroids on each side of Kyiv city; they represented as pink points in the Figure. The data points (location of Nova Poshta's offices) are represented in four colors, depending on the belonging to their corresponding clusters. The larger dot, the higher the weighted rating of the office is.

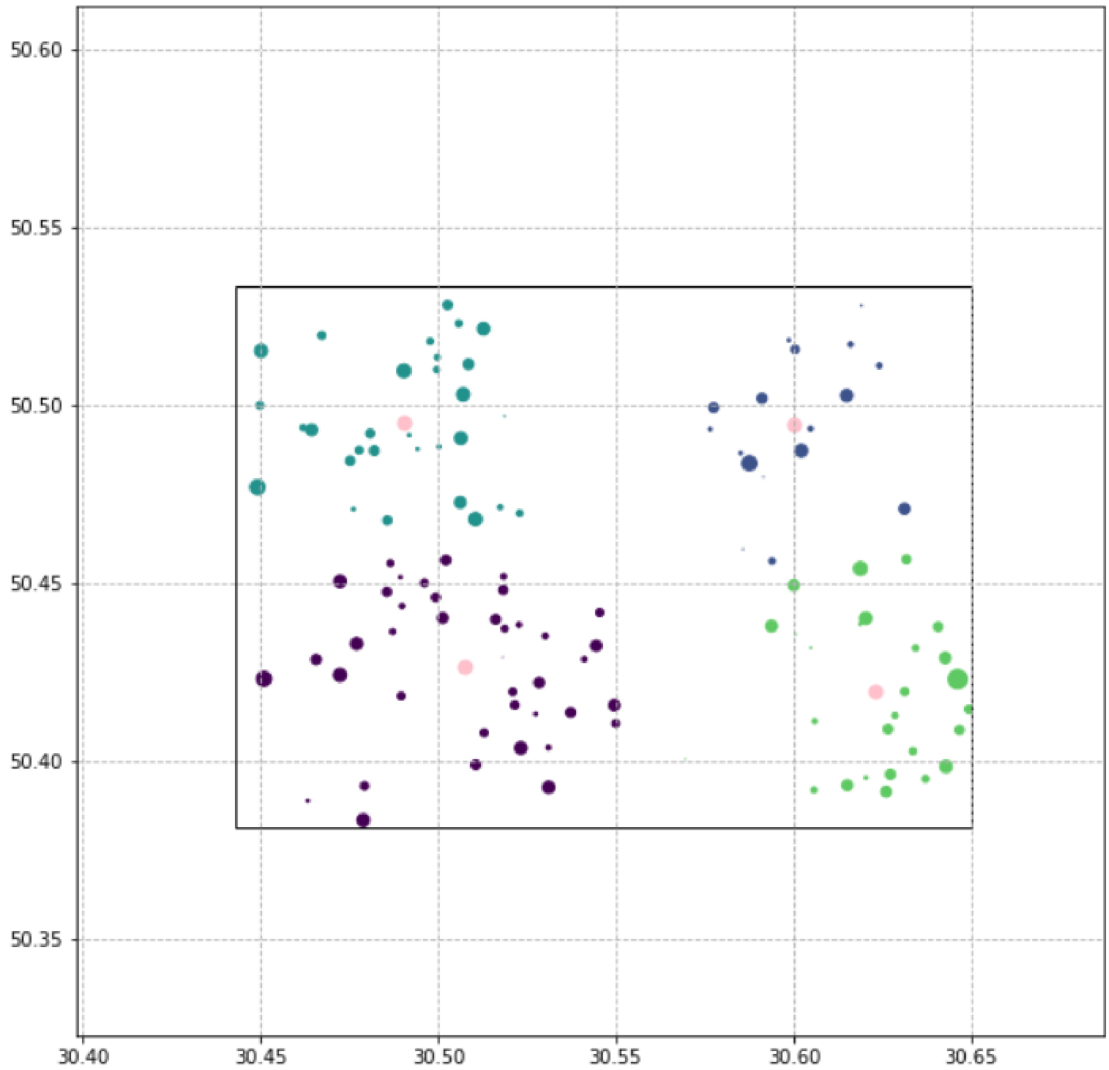


Figure 12. Weighted ratings of the post offices colored in the colors of one of the 4 assigned

The weighed rating is computed using the rule in Figure 13.

```
clusters_df['user_ratings_total_scaled'] = clusters_df['user_ratings_total'] / clusters_df['user_ratings_total'].max()
clusters_df['weighted_ratings'] = clusters_df['rating'] * clusters_df['user_ratings_total'] / (clusters_df['rating'] *
clusters_df['user_ratings_total']).max()
```

Figure 13. Rule for computation a weighted ratings

4.3.3 Plotting K-means clustering on Google Maps

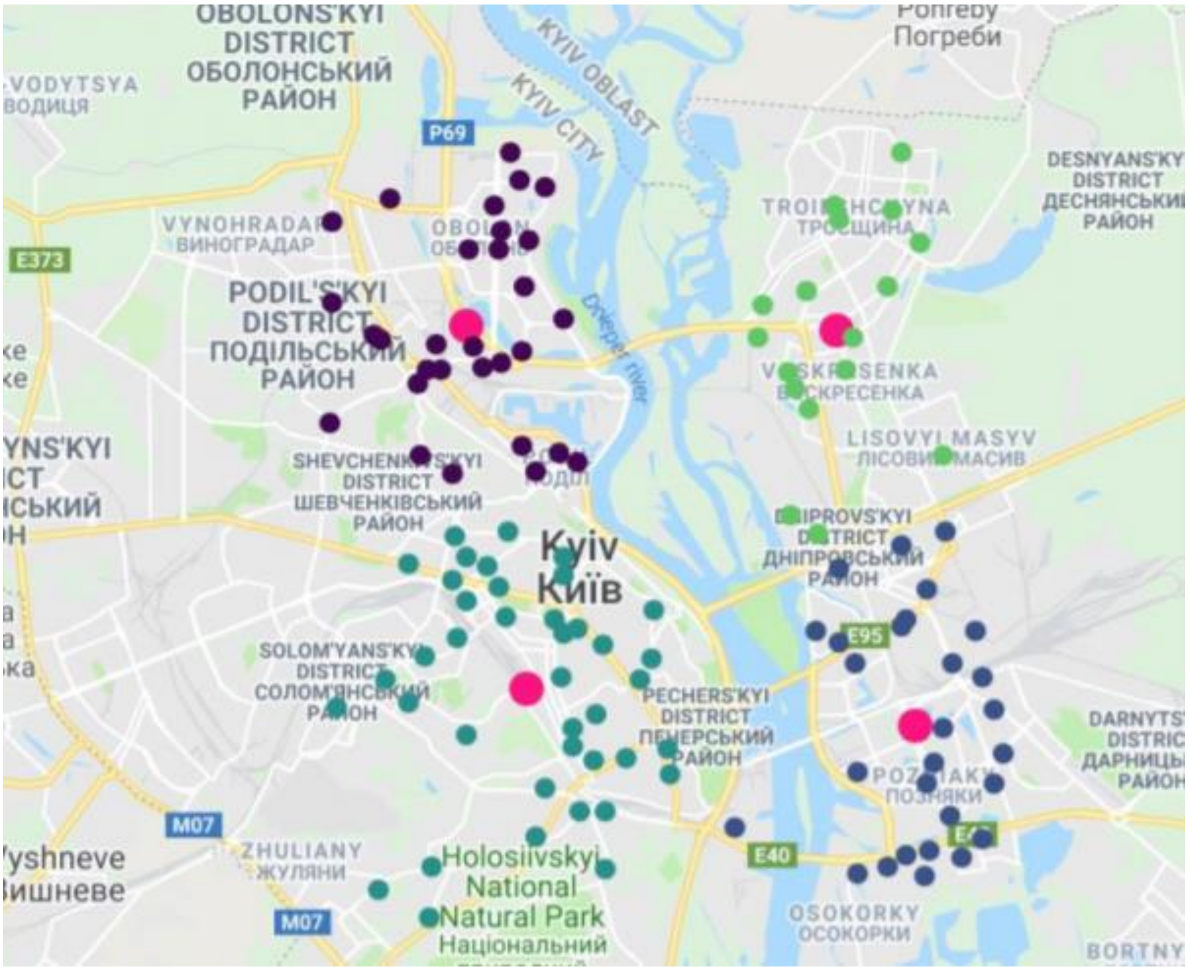


Figure 14. Plotting K-means clustering on Google Maps

Figure 14 visualized Nova Poshta's offices with color coded belonging to the corresponding clusters on Google Maps. Important and interesting note that two out of four centroids correspond to the real location of Nova Poshta distributive centers.

4.4 Changing Euclidian distance to true distance matrices using Google Distance Matrix API

The main disadvantage of the previous analysis was the use of Euclidean distance instead of the real distances that a courier has to cover for getting from one location to another. To address this issue we use Google Distance Matrix API for getting both walking and driving distances that will be used in the following analysis.

4.5 Clustering. Part 2

Now that the true distance matrices for driving and walking instead of Euclidean distance metric have obtained, one can compute a more accurate clustering distribution.

However, *sklearn* does not have K-means clustering for custom distance matrices, and thus one ought to seek some other clustering technique.

4.5.1 Agglomerative clustering

Agglomerative Clustering algorithm requires the number of clusters to be provided explicitly. Thus, one resort to Silhouette analysis once again to determine the optimal number of clusters. In the Figures 15, 16, 17, 18, 19 Silhouette analysis for Agglomerative clustering with cluster numbers from 2 to 6 is performed.

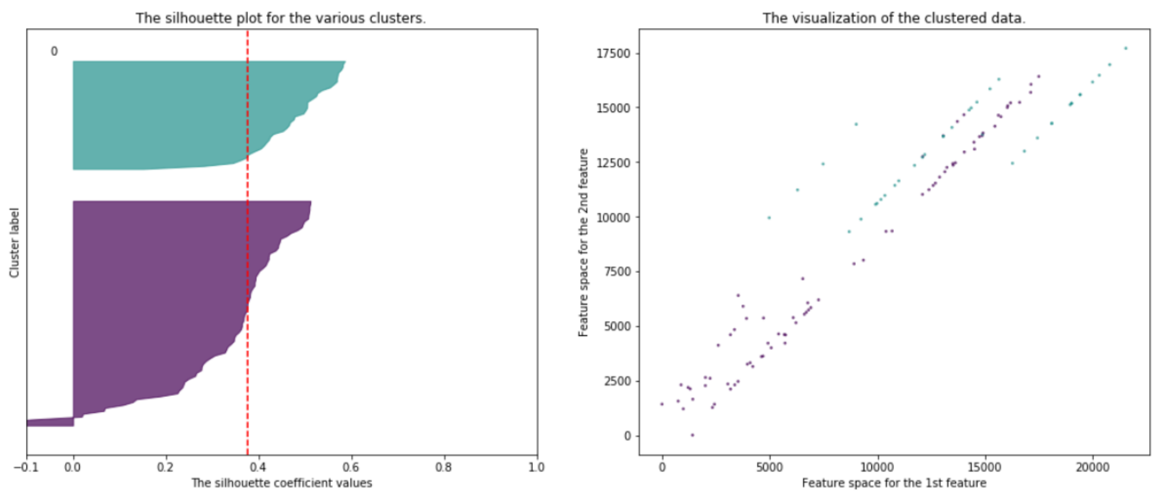


Figure 15. Silhouette analysis for Agglomerative clustering on sample data with 2 clusters

Silhouette analysis for the two clusters determined the average value of the silhouette equal to 0.37734037284811134.

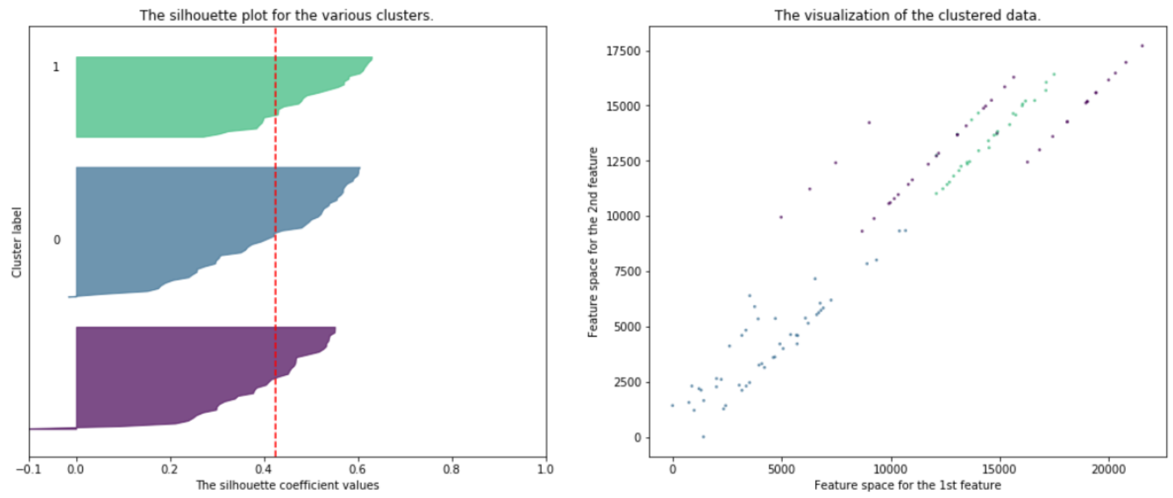


Figure 16. Silhouette analysis for Agglomerative clustering on sample data with 3 clusters

Silhouette analysis for the two clusters determined the average value of the silhouette equal to 0.4252732444890752.

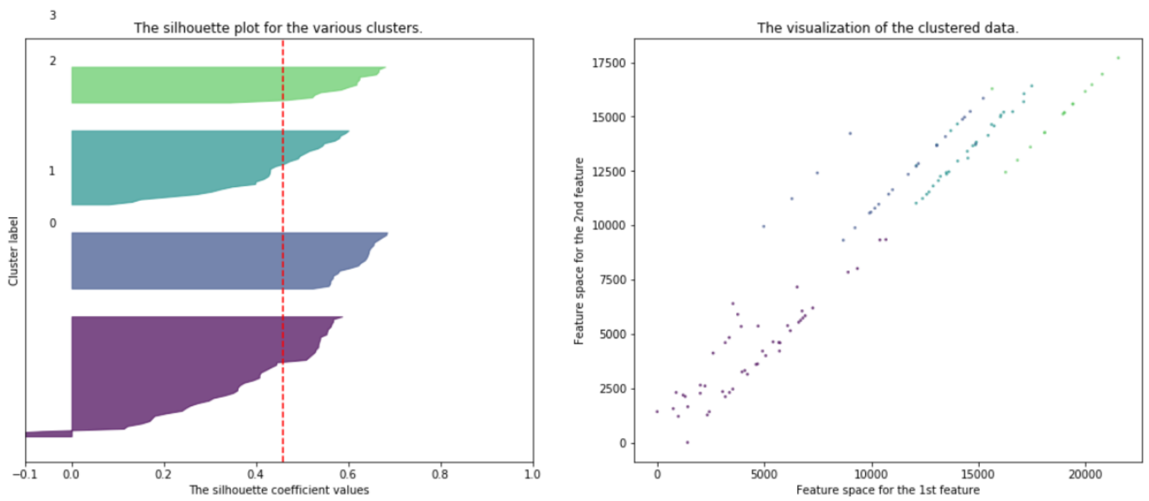


Figure 17. Silhouette analysis for Agglomerative clustering on sample data with 4 clusters

Silhouette analysis for the two clusters determined the average value of the silhouette equal to 0.45801498953532294.

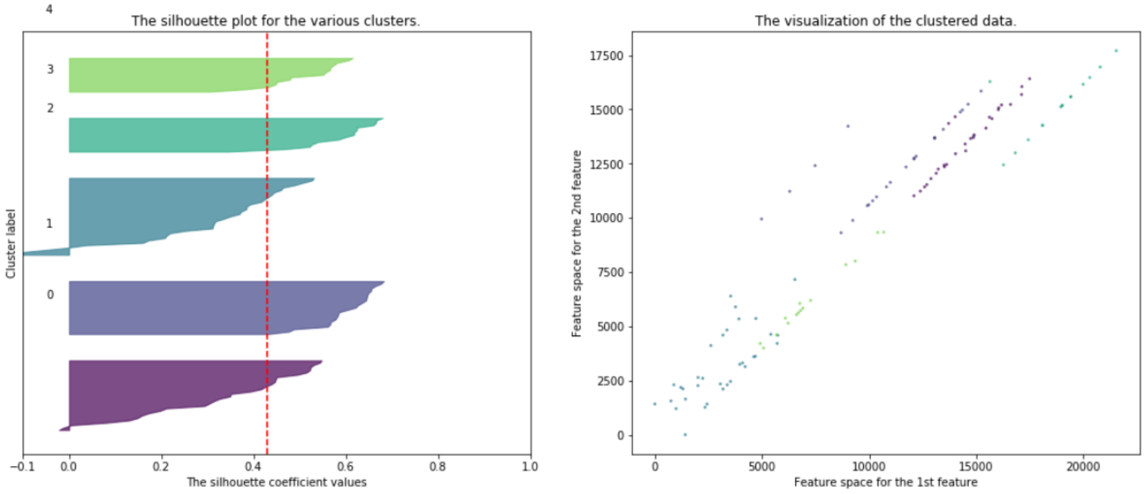


Figure 18. Silhouette analysis for Agglomerative clustering on sample data with 5 clusters

Silhouette analysis for the two clusters determined the average value of the silhouette equal to 0.4289619702791857.

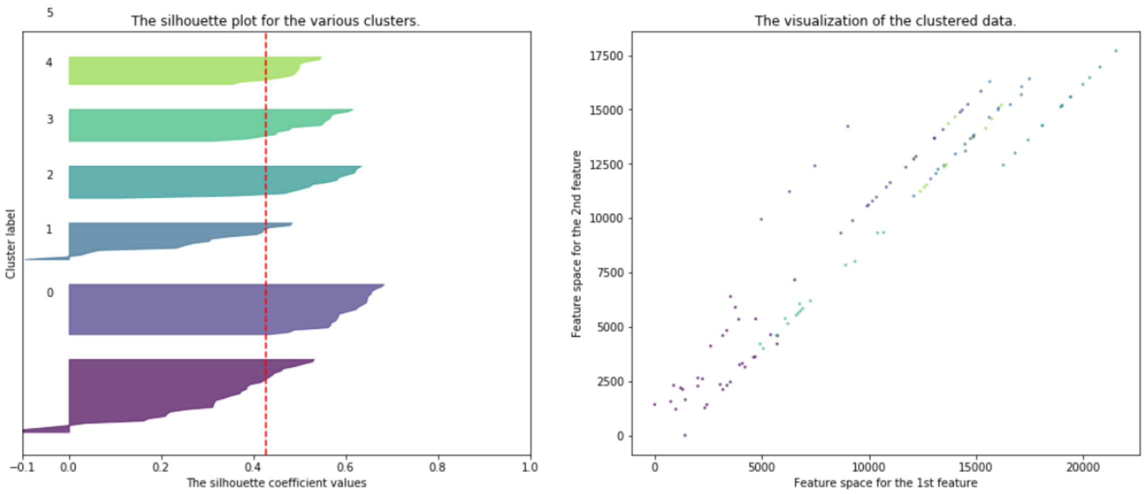


Figure 19. Silhouette analysis for Agglomerative clustering on sample data with 6 clusters

Silhouette analysis for the two clusters determined the average value of the silhouette equal to 0.4276927423984137.

The Silhouette analysis of Agglomerative Clustering for both driving and walking distances hints that the optimal number of clusters is 4.

4.5.2 Agglomerative clustering for driving distances

Result of Agglomerative clustering algorithm using the driving distances from Google Distance Matrix API can be found in Figure 20.

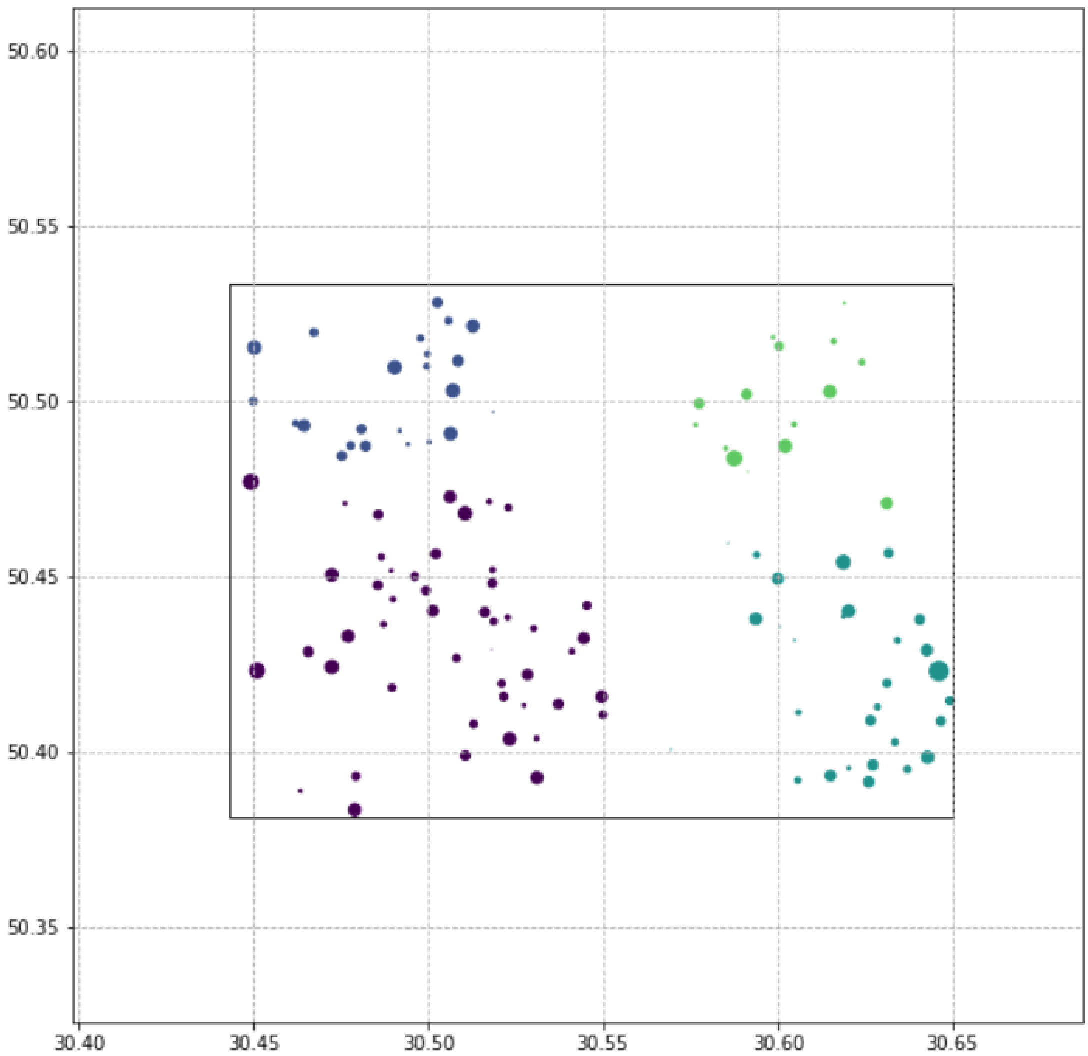


Figure 20. Result of Agglomerative clustering algorithm using the driving distances from GDM API with weighted ratings of the post offices colored in the colors of one of the 4 assigned

4.6 Estimating demand of consumers

Since one do not have any meaningful way of estimating the demand of post office consumers directly, let’s try to obtain it via a relatively relatable metric.

Let’s use the number of ratings as a relative metric of the demand scale. For instance, an office with 50 reviews will have 50 points of demand.

Let’s assume that 80% of the users are users who frequent the considered post office at least once a week and the other 20% use the post office's services at least once a month. So, for an office with 50 points of demand will be considered to have 0.8 * 50 * 4 (weeks) = 160 + 0.2 * 50 = 170 visitors.

Due to these assumptions, the daily demand for post offices was calculated. (Figure 21)

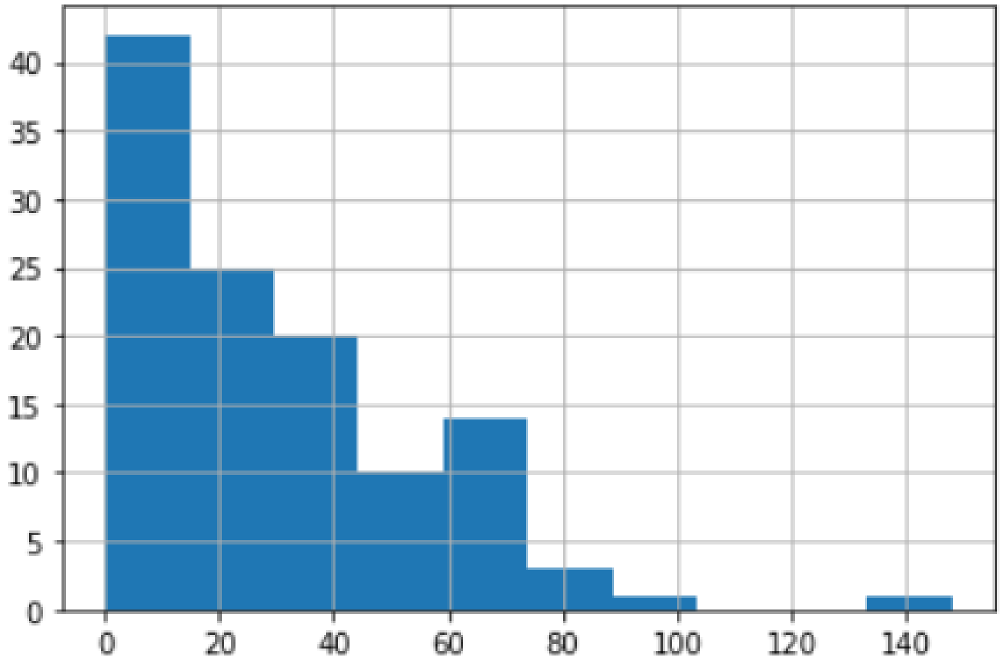
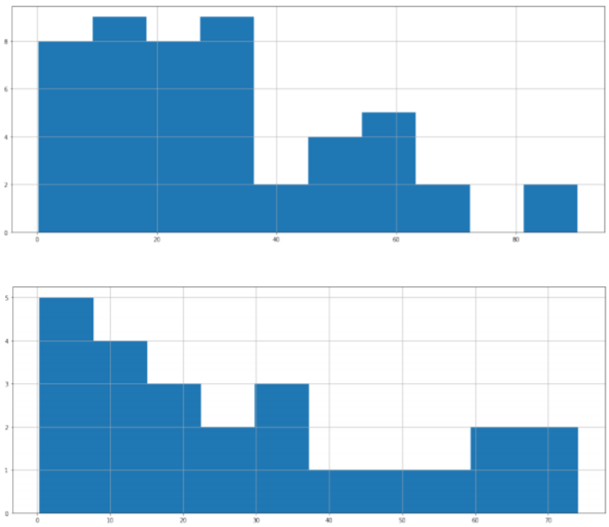


Figure 21. Post offices daily demand

Demand was then distributed among the clusters and presented as histograms. (Figure 22)



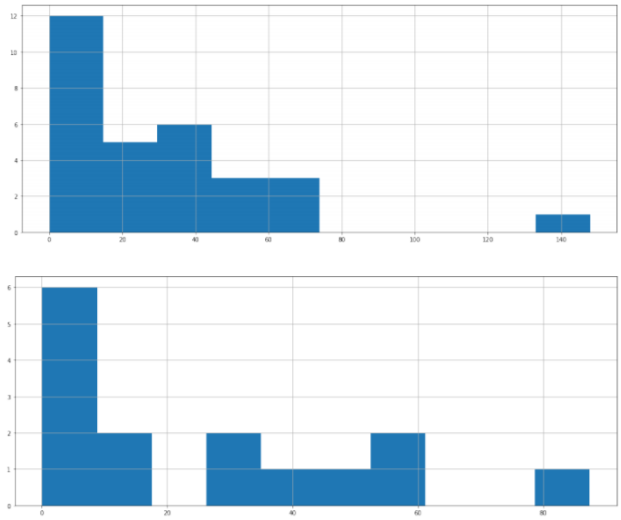


Figure 22. Daily demand of post offices distributed among clusters

If one further assume that the weight of each parcel averages 700 grams, then the total weight of the daily demand cargo for each cluster, and hence for each satellite, is 1049.1, 492.5, 617.3 and 268.2 kilograms, respectively. The total weight of daily demand for all clusters together is 2427,124 kilograms.

4.6.1 Choosing vehicles for transportation

To address such demand, one have to decide on the types of vehicles that will deliver the parcels from the City Distribution Center to the Satellites, and from there to the individual consumers.

The vehicle for delivering between CDC and S needs to be able to carry up to 3,000 kg of cargo.

The vehicles delivering from Satellites need to be capable of delivering up to 1,100 kg of cargo. However, for smaller deliveries, a load capacity of as small as 300 kg, 650 kg, and 500 kg are required.

To ensure CSR and environmental sustainability, let's first consider fully electric vehicles in order to find those that can potentially fulfill the demand.

Mitsubishi Fuso Canter

For deliveries from CDC to Satellites, Mitsubishi Fuso Canter can be considered. The vehicle is capable of carrying up to 3,000 kg of cargo and can cover up to 100 km on a single charge. Although the charge is quite small, the Satellites can be equipped with quick chargers, which will be charging the truck during unloading. The maximum speed of the truck is 90 km/h, which is enough for routes within the city. It can be charged fully in 7 hours from a regular charger, and in just an hour from a quick charger.

Electric tricycles TailG

For deliveries from Satellites to customers, compact and mobile all-electric tricycles seem like a perfect option. Their optimal speed is 35 km/h, and they can cover up to 60 km on a single stock battery, but additional compartments for batteries allow to increases this by a factor of 2 or 3. It charges fully within 5-8 hours. The maximum load of the tricycle is 500 kg, which makes it a perfect option for two of the four Satellites.

Volkswagen e-Caddy

For deliveries at higher load Satellites, a small electric cargo vehicle e-Caddy can be considered. Not only its single charge capacity is enough to cover 250 km, but it also has a payload of 636 kg.

Thus, to cover the needs of our consumers, one would need:

- 1 Fuso Canter for CDC;
- 3 TailG (1 each for Satellites 1 and 3, and 1 more as a backup for Satellite 0);
- 3 e-Caddy (1 for Satellite 2 and 2 for Satellite 0).

4.6.2 Satellite location

In order to estimate the candidate locations for the CDC, let's first obtain the locations of Satellites. In order to do so, one perform a K-means cluster centroid search for each of the clusters obtained by Agglomerative Clustering. (Figure 23)

```python
satellites_centroids = {}
for cluster in clusters_df["cluster"].unique():
    num_clusters = 1
    kmeans = KMeans(n_clusters = num_clusters).fit(clusters_df[['lat', 'lon']][clusters_df["cluster"] == cluster])
    satellites_centroids[cluster] = kmeans.cluster_centers_[0]
```

```python
satellites_centroids
```

```
{0: array([50.43326494, 30.50593291]),
 1: array([50.50294633, 30.48901154]),
 2: array([50.42232394, 30.62077663]),
 3: array([50.49934511, 30.60136474])}
```

```python
fig = gmaps.figure(center = (lat, lon), zoom_level = zoom, layout=figure_layout)
for clust in clusters_df['cluster'].unique():
    cluster_layer = gmaps.symbol_layer(
        clusters_df[['lat', 'lon']][clusters_df['cluster'] == clust],
        fill_color = clusters_df['color'][clusters_df['cluster'] == clust].iloc[0],
        stroke_color = clusters_df['color'][clusters_df['cluster'] == clust].iloc[0],
        scale = 3
    )
    fig.add_layer(cluster_layer)

centers_layer = gmaps.symbol_layer(
        satellites_centroids.values(),
        fill_color = "#ff007f",
        stroke_color = "#ff007f",
        scale = 5)
fig.add_layer(centers_layer)

fig
```

Figure 23. K-means cluster centroid search for each of the clusters obtained by Agglomerative Clustering

Although it might seem a little counter-intuitive, the knowledge of the Kyiv city logistics hints one that CDC should be placed either somewhere near Podil region, or alternatively near the Blockbuster Mall.

Let's define the corresponding points and compute the distances to be covered to the Satellites. (Figure 24)

```python
try:
    podil_driving = np.load(loc_podil_matrix)
except:
    podil_driving = np.empty((len(podil_sat_coords), len(podil_sat_coords)))
    columns = len(podil_sat_coords)
    gmaps_api = googlemaps.Client(key = api_key)
    for row in tqdm(range(len(podil_sat_coords))):
        for column in tqdm(range(columns)):
            origins = (podil_sat_coords[row][0], podil_sat_coords[row][1])
            destination = (podil_sat_coords[column][0], podil_sat_coords[column][1])
    #         print(f"Distance from {origins} to {destination}")
            distance = gmaps_api.distance_matrix(origins, destination, mode='driving')["rows"][0]["elements"][0]["distance"]["va
lue"]
            podil_driving[row, column] = distance
        sleep(5)
    np.save(loc_podil_matrix, podil_driving)
```

```python
try:
    mall_driving = np.load(loc_mall_matrix)
except:
    mall_driving = np.empty((len(mall_sat_coords), len(mall_sat_coords)))
    columns = len(mall_sat_coords)
    gmaps_api = googlemaps.Client(key = api_key)
    for row in tqdm(range(len(mall_sat_coords))):
        for column in tqdm(range(columns)):
            origins = (mall_sat_coords[row][0], mall_sat_coords[row][1])
            destination = (mall_sat_coords[column][0], mall_sat_coords[column][1])
    #         print(f"Distance from {origins} to {destination}")
            distance = gmaps_api.distance_matrix(origins, destination, mode='driving')["rows"][0]["elements"][0]["distance"]["va
lue"]
            mall_driving[row, column] = distance
        sleep(5)
    np.save(loc_mall_matrix, mall_driving)
```

Figure 24. Computation of the distances to be covered to the Satellites

4.7 Finding optimal route for both candidates CDCs

Luckily, the problem is rather small, which means one can solve it by performing a greedy search (i.e. simply go through all options).

```
# Initializing total distances to arbitrarily large numbers to find min via comparison
podil_total_dist = 1e9
mall_total_dist = 1e9
podil_route = ""
mall_route = ""
for route in routes:
    podil_dist = 0
    mall_dist = 0
    for i in range(len(route) - 1):
        podil_dist += podil_driving[route[i], route[i + 1]]
        mall_dist += mall_driving[route[i], route[i + 1]]
    if podil_dist < podil_total_dist:
        podil_total_dist = podil_dist
        podil_route = route
    if mall_dist < mall_total_dist:
        mall_total_dist = mall_dist
        mall_route = route
```

Figure 25. Searching both minimum paths for Podil and Mall CDCs

```
For Podil CDC the optimal route is (4, 1, 3, 2, 0, 4). The distance is 47828.0
For Mall CDC the optimal route is (4, 3, 2, 0, 1, 4). The distance is 47324.0
```

Figure 26. The results of minimum paths for Podil and Mall CDCs computations

As can be deduced from the computed results (Figure 26), the CDCs nearby the Blockbuster Mall has a 500 meter shorter path. For the purposes of further analysis, one assume that it is indeed a better choice for the CDC. Figure 23 shows numbered satellites and CDC nearby the Blockbuster Mall which is marked with number 4.

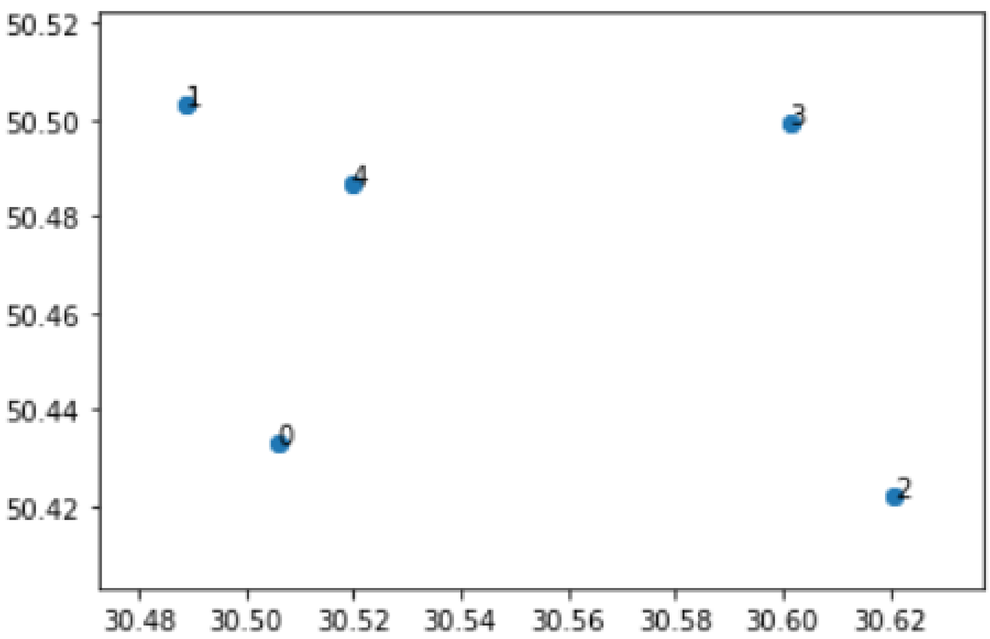


Figure 27. Numbered satellites with CDC in point 4

4.8 Defining transportation costs

To recall, early a decision to promote all-electric parcel delivery was made. Thus one need to estimate the cost of 1 km driven.

The average cost of 1 km travelled by an electric vehicle in Ukraine is estimated at $0.05, which is the lowest price one can get for a cargo vehicle.

Let's will take the cost of 1 km as the only determinant of the variable costs. Since one have the payload capacity of the vehicles as given, the quantity of the total vehicles can be computed. Thus, all other expenses will be considered as either fixed costs or investments (e.g. the purchase of the vehicle).

4.9 Scenario 1. 2-Echelon Single Source Location Model

This scenario assumes that the Satellites act as Pack Stations, i.e. the customers come to them to pick up their parcels.

The problem can then be simplified to a Traveling Salesman Problem, where the City Distribution Center acts as the starting and ending node, and the salesman have to visit each other node on the graph.

This problem was solved earlier when the proper location for the CDC was chosen. The route is 4-3-2-0-1-4 (Figure 28) and total travel distance is 47324 meters.

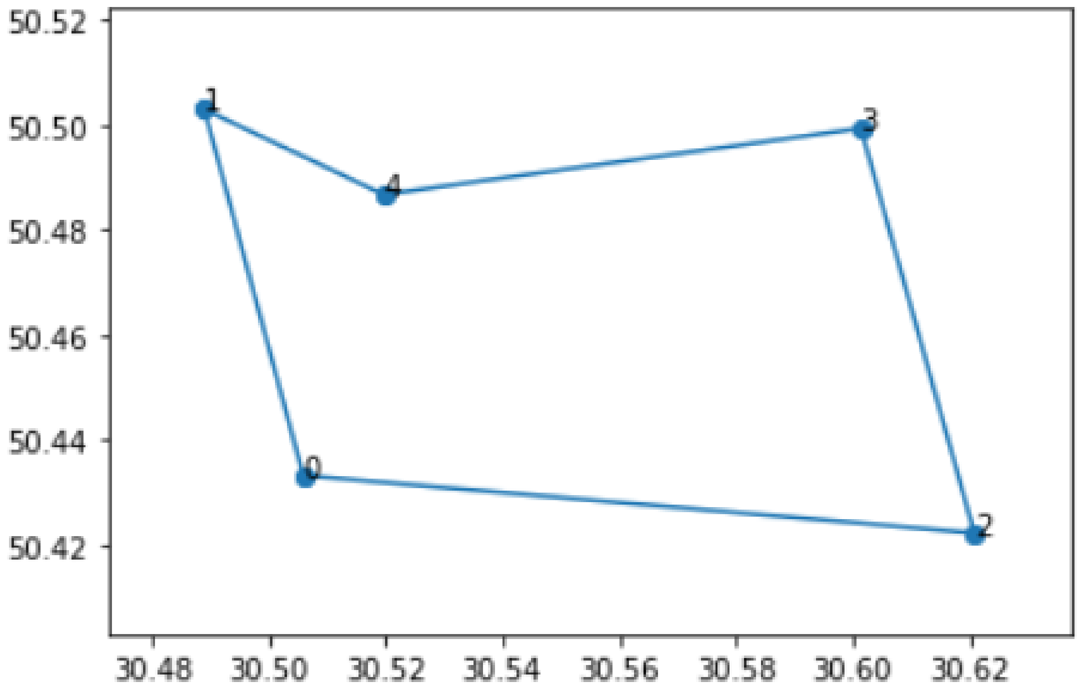


Figure 28. The optimal Route for 2-Echelon Single Source Location Model

4.10 Scenario 2. 2-Echelon Single Source Location Model. Parcel Delivery

From the scenario 1, one take the Pack Stations locations as origin points for parcel delivery to final customers (i.e. post offices locations). Hence, it is needed to solve the traveling salesman problem for each of the pack station points.

The relation of every post office to pack stations has been previously obtained by Agglomerative clustering for driving distances.

Simulated annealing probabilistic technique is using here for approximating the global optimum of a given function and determining approximate solutions to the Travelling Salesman Problem by direct sampling and by simulated annealing.

To compute the distances, one would need to populate the existing driving distance matrix with the distances between every point and the centroids of the clusters.

After applying the algorithm, one will get the results that are displayed in the Figure 29.

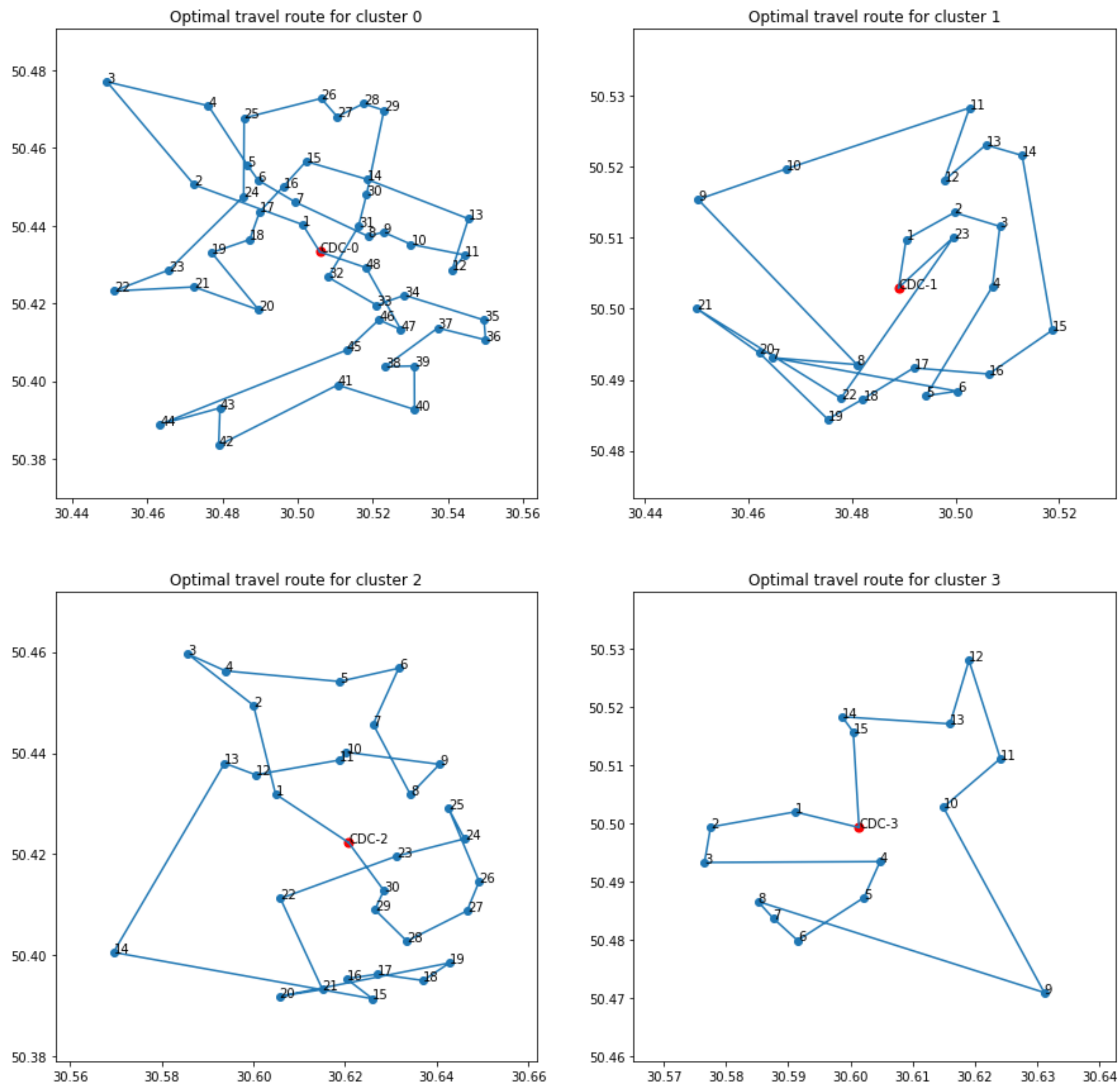


Figure 29. The optimal Routes for 2-Echelon Single Source Location Model with parcels delivery

As a result, the lengths of the routes are:

- 99.5 km for cluster 0;
- 52.04 km for cluster 1;
- 69.778 km for cluster 2;
- 33.119 km for cluster 3.

4.11 Capacitated 2EVRP: Further extensions to the model

One do not consider any time windows and satellite synchronization constraints in this part of research.

To enhance the model and make the model more applicable to the real world, let’s collect and generate the supporting data.

All further work will be conducted to make the input data resemble more the data obtained from smart road conditions. Namely, the following data points will be assumed to be obtainable from every edge of the network:

- Precipitation: intensity (0 if no precipitation)
- Humidity: percentage
- Temperature: float
- Visibility: km of visibility

Since the data one have used previously has already been collected from Google Directions API, every edge of the network already represents the proper distances between any two points of the network. This is also an important disclaimer for further illustrations, as some edge choices might not seem optimal due to the fact, that the plot represents the straight line connections, not the actual driving directions.

As in this paper the task of capacitated vehicle routing problem is exploring, one can translate all weather variables to be linear components of the total distance traveled by the delivery vehicle.

Thus, any unfavorable weather conditions will translate into a penalty for choosing certain route in a form of longer distance travelled.

The described setting can be thought of as a 2-Echelon Vehicle Routing Problem, where the main constraint is not the distance travelled, but the total cost of the trip. Any unfavorable conditions ultimately lead to higher transit times on the same edges of the network, which virtually translates in higher fuel expenditures.

4.12 Building a sustainable model

4.12.1 Obtaining base distance matrix

One will reuse the base distance matrix obtained in the work above. In case new entry points for vertices will appeared, one will use Google Direction Matrix API to obtain the corresponding distance matrix.

4.12.2 Obtaining weather data

One will use DarkSky API to obtain weather data for Kyiv. To extend the obtained data to every edge of the network, let's use normal distribution *N (0,1)*.

To simulate randomness of traffic conditions, let's also introduce deviations of 5σ to 10σ (standard deviations) for visibility and precipitation.

As one will only use a part of the output from the DarkSky API, let’s keep just the variables one need:

- Precipitation: intensity (0 if no precipitation)
- Humidity: percentage
- Temperature: float
- Visibility: km of visibility

Furthermore, since one is currently solving an instantaneous problem, let’s for now omit the variability of the weather conditions in time, taking observations at a single hour (7 A.M).

The values of obtained variables can be seen in the Figure 30.

```
precipIntensity          0
humidity              0.78
temperature           5.78
visibility          16.093
Name: 18, dtype: object
```

Figure 30. Values of obtained variables at a single hour (7 A.M.)

4.12.3 Generating data points for edges

To generate data points, one will use a normal distribution *N (0,1)*.

To further enhance the data, one will randomly diverge some edges by a magnitude of 5 to 10 standard deviations. This will simulate the randomness of the road conditions (e.g. repairs, car crashes, severe smog, etc.).

One has a total of 118 vertices, thus for every variable in the weather dataset, one will have to create a 118×118 matrix of points.

4.12.4 Generating delays and applying to base distance matrix

One will reuse the same coordinates of spiked edges to ensure the linear regression fits fairly well on the generated delay-adjusted data.

To simulate the moderate effect of weather on the road conditions one will introduce delays (i.e. extend driving distances by 10-30%). For the spiked points, one will allow a 50-100% increase in the distance.

4.12.5 Fitting linear regression

Let's use the OLS method of the statsmodels module to fit the model to the data one have generated for further usage.

The endogenous variables are provided in the adjusted distance matrix, whereas the exogenous variables are given in the generated weather dictionary and in the original base distance matrix. The model can be thus formalized as follows:

$$d_{adj} = d_{base} + \beta_{hum}x_{hum} + \beta_{temp}x_{temp} + \beta_{prec}x_{prec} + \beta_{vis}x_{vis} + \epsilon$$

Figure 31 shows Ordinary Least Square Regression Results.

```
                                 OLS Regression Results
=======================================================================================
Dep. Variable:                      y   R-squared (uncentered):                  0.998
Model:                            OLS   Adj. R-squared (uncentered):             0.998
Method:                 Least Squares   F-statistic:                         1.101e+06
Date:                Thu, 07 May 2020   Prob (F-statistic):                       0.00
Time:                        13:38:08   Log-Likelihood:                    -1.0863e+05
No. Observations:               13456   AIC:                                 2.173e+05
Df Residuals:                   13451   BIC:                                 2.173e+05
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
d_base         1.1967      0.001   1054.659      0.000       1.194       1.199
x_hum        115.5749     41.206      2.805      0.005      34.805     196.345
x_temp         4.8436      1.308      3.704      0.000       2.280       7.407
x_prec       -17.4374     11.315     -1.541      0.123     -39.616       4.741
x_vis         -0.9864      1.062     -0.929      0.353      -3.068       1.095
==============================================================================
Omnibus:                     2033.321   Durbin-Watson:                   1.991
Prob(Omnibus):                  0.000   Jarque-Bera (JB):            14513.299
Skew:                           0.530   Prob(JB):                         0.00
Kurtosis:                       7.976   Cond. No.                     8.11e+04
==============================================================================
```

Figure 31. OLS Regression Results

Unsurprisingly, the model R^2 shows very high accuracy. Based on the obtained weights of the model we can see that it aligns with the common sense, the higher humidity and the higher temperature the more difficult this route is, and the higher visibility the more optimal this route is. The negative coefficient of the precipitations factor is explained by a high correlation with humidity factor and jointly makes total sense. Important to note that the coefficients and influence of the selected factors to the final result will most likely differ on the real-world data due to the nature of generating the adjusted distance matrix.

4.12.6 Solving VRP

Since one has only modified the distance matrix thus far, let's try to reuse the previously built tools to solve the VRP with the new distance inputs. (Figure 32)

```
matrix_driving_df.shape
```

```
(116, 116)
```

```
matrix_driving_df = mevrp.distances.add_points_to_distance_matrix(matrix_driving_df, centroids.values())
matrix_driving_df.shape
```

```
(120, 120)
```

```
from ast import literal_eval as make_tuple
loc_distances_with_centroids = "data/driving_w_centroids.csv"
try:
    matrix_driving_df = pd.read_csv(loc_distances_with_centroids, index_col = 0)
    matrix_driving_df.index = [make_tuple(idx) for idx in matrix_driving_df.index]
    matrix_driving_df.columns = [make_tuple(col) for col in matrix_driving_df.columns]
except:
    matrix_driving_df = mevrp.distances.fill_distances(matrix_driving_df, "driving")
    matrix_driving_df.to_csv(loc_distances_with_centroids, header = True, index = True)
```

```
route_points["adjusted_route_order"] = np.nan
for cluster in new_routes.keys():
    cluster_route = partial(get_adjusted_order, cluster = cluster)
    route_points["adjusted_route_order"][route_points["cluster"] == cluster] = route_points["coords"][route_points["cluster"] ==
cluster].apply(cluster_route)
route_points["adjusted_route_order"] = route_points["adjusted_route_order"].astype(int)
```

```
route_points.head()
```

	lat	lon	cluster	coords	route_order	adjusted_route_order
0	50.440276	30.501372	0	(50.44027639999999, 30.5013722)	0	25
1	50.450083	30.496220	0	(50.450083, 30.49622)	15	23
2	50.424245	30.472491	0	(50.4242448, 30.4724908)	20	29
3	50.435211	30.530175	0	(50.4352113, 30.5301745)	9	15
4	50.437261	30.518789	0	(50.4372615, 30.5187893)	7	17

```
adjusted_cluster_routes = {}
for cluster in route_points["cluster"].unique():
    adjusted_cluster_routes[cluster] = [route_points["coords"][route_points["cluster"] == cluster][route_points["adjusted_route_
order"] == x].iloc[0] for x in range(len(route_points[route_points["cluster"] == cluster]))]
```

```
adjusted_cluster_routes[cluster]
```

```
[(50.50202609999999, 30.5910748),
 (50.49942100000001, 30.5775154),
 (50.4933339, 30.5765286),
 (50.4866342, 30.5851212),
 (50.48375799999999, 30.58756199999999),
 (50.4709806, 30.6311434),
 (50.47993349999999, 30.5914967),
 (50.4872866, 30.60214199999999),
 (50.49348209999999, 30.6047476),
 (50.5028083, 30.6149291),
 (50.51118330000001, 30.6240816),
 (50.52806, 30.61904),
 (50.5171471, 30.6159925),
 (50.5183638, 30.5986522),
 (50.5157582, 30.600444)]
```

Figure 32. Reusing the previously built tools for solving the VRP with the new distance inputs

After carrying out the calculations, one will receive new optimal routes for all clusters, which are shown on Figure 33.

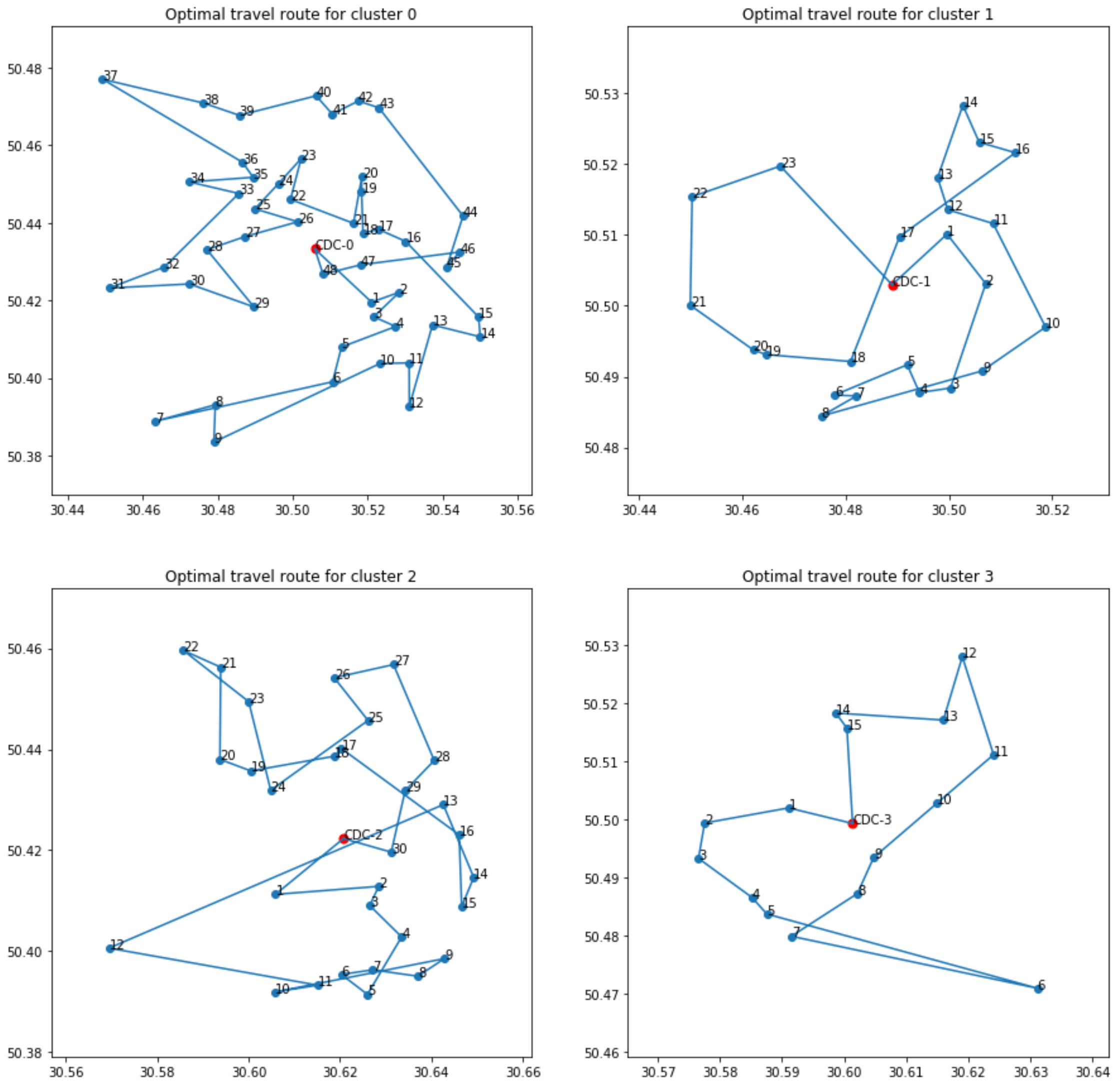


Figure 33. New optimal routes for all clusters

As a result, the lengths of the routes are:

- 91.932 km for cluster 0;
- 50.872 km for cluster 1;
- 72.522 km for cluster 2;
- 32.772 km for cluster 3.

Let's plot the base route (blue line) and the route adjusted (red line) for generated weather data to visually compare the two. (Figure 34)

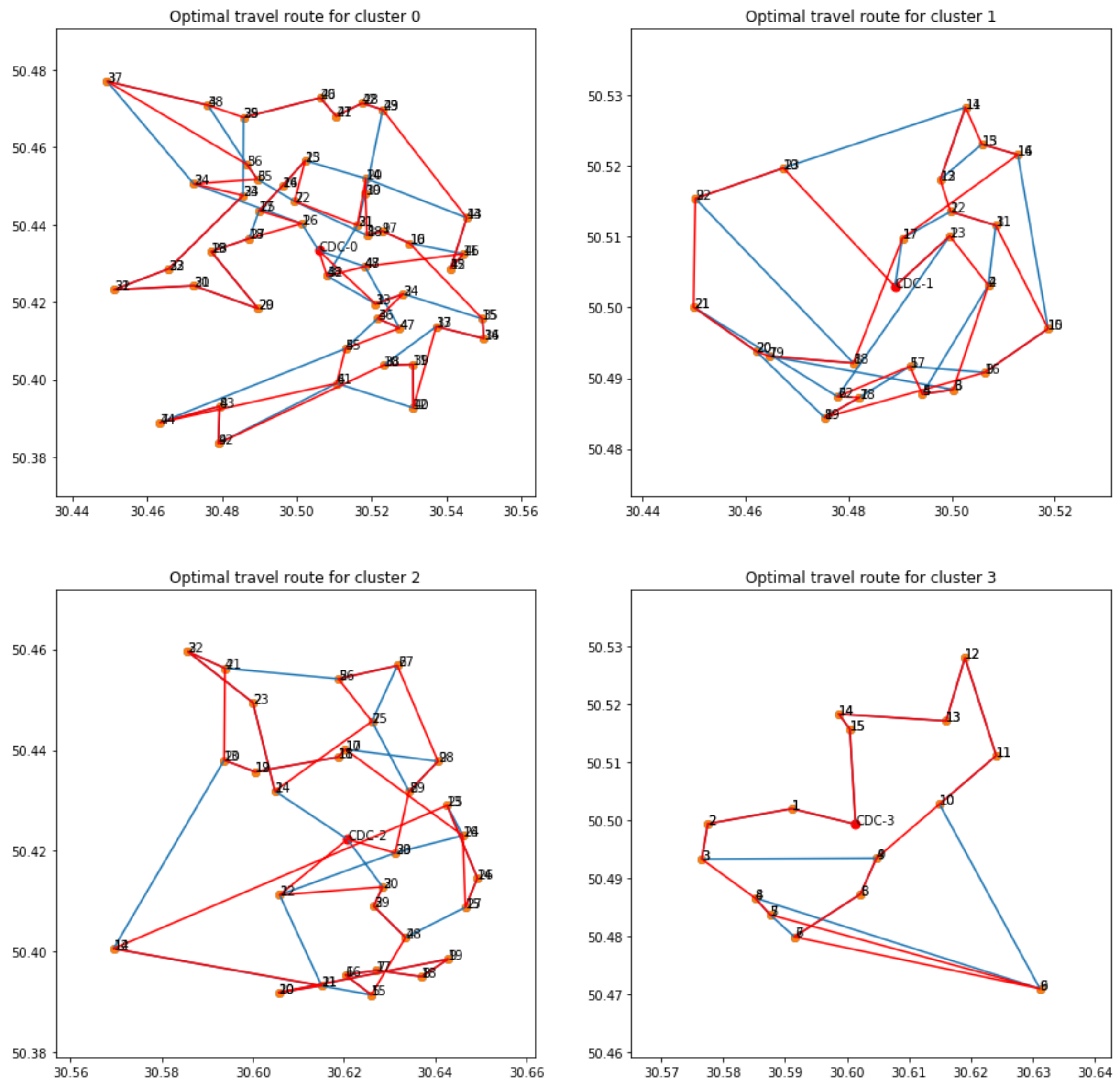


Figure 34. The base route (blue line) and the route adjusted (red line) for generated weather data

4.13 Conclusions

In this chapter, we researched in practice the benefit of utilizing the data from smart road sensors and weather conditions in the Two-Echelon VRP on a real example of Kyiv delivery to the post offices of the Nova Poshta company. The overall joint saved distance is equal to 6,339km even though in one of four clusters we can see the longer resulting route than in the initial model. This is explained by the sample and the approach for generating the additional factors used in the model and large variations in deal distances in that area between the office locations.

Due to the strong distribution assumptions that were applied during the generation of additional factors (humidity, temperature, precipitations, visibility) we've unsurprisingly got good results of our OLS regression model that may differ on the real data that may affect the overall performance of the overall model. In this case, our algorithm and the approach is easy can be extended with more advanced regression techniques that would address the highlighted limitations.

5.0 Concluding remarks and further research

The main goal and the core contribution of this work are to develop an approach and a method of integrating the additional factors that can be obtained from smart roads to the Two-Echelon Capacitated Vehicle Routing Problem. This was done in the "Model development" section on the example of the OLS regression model, where we showed in practice the benefits on the real example of Kyiv delivery to the post offices of the Nova Poshta company.

However, the OLS regression model makes strong assumptions that may not be fulfilled in the practical applications, such as a linear relationship, multivariate normality, homoscedasticity, and no multicollinearity; the last one was violated on the example of humidity and precipitation factors and shows the great potential in researching more advanced regression models that have a weaker list of assumptions. This part of potential research is out of the scope of this work and is proposed for further research.

Reference list

Angelelli, E., Speranza, M., 2002. The periodic vehicle routing problem with intermediate facilities. European Journal of Operational Research 137, 233– 247.

Bard, J., Huang, L., Dror, M., Jaillet, P., 1998. A branch and cut algorithm for the vrp with satellite facilities. Tech. rep.

Christofides, N., Elion, S., 1969. An algorithms for the vehicle dispatching problem. Operational Research Quarterly 20, 309–318

Chuck Holland, Jack Levis, Ranganath Nuggehalli, Bob Santilli, Jeff Winters (2017) UPS Optimizes Delivery Routes. INFORMS Journal on Applied Analytics 47(1):8-23. https://doi.org/10.1287/inte.2016.0875

Clarke, G., Wright, J., 1964. Scheduling of vehicles from a central depot to a number of delivery points. Operations Research 12 (4), 568–581.

Compound annual growth rate (CAGR) of the logistics industry from 2014 to 2018, by segment. https://www.statista.com/statistics/943517/logistics-industry-global-cagr/

Cordeau, J., Gendreau, M., Hertz, A., Laporte, G., Sormany, J., 2004. New heuristics for the vehicle routing problems. Tech. Rep. Les Cahiers du GERARD, G-2004-33

Cordeau, J., Gendreau, M., Mercier, A., 1997. A tabu search heuristic for periodic and multidepot vehicle routing problem. Networks 30, 105–119.

Cordeau, J., Gendreau, M., Mercier, A., 2001. A unified tabu search heuristic for vehicle routing problems with time windows. Journal of the Operational Research Society 52, 928–936.

Cordeau, J., Laporte, G., Savelsbergh, M., Vigo, D., 2007. Vehicle Routing. In: Barnhart, C., Laporte, G. (Eds.), Transportation. North Holland Elsevier, pp. 367–428.

Crainic, T., Gendreau, M., forthcoming. Intelligent freight transportation systems: Assessment and the contribution of operations research. Transportation Research part C.

Crainic, T., Ricciardi, N., Storchi, G., 2004. Advanced freight transportation systems for congested urban areas. Transportation Research part C 12, 119– 137.

Hansen, W. G., 1959. How accessibility shapes land use. Journal of the American Institute Planners 25, 73–76.

How is the World of Logistics Affected by Globalization? (2018) https://www.amcsrouting.com/newsroom/blog/how-is-the-world-of-logistics-affected-byglobalization/

Mester, D., Br¨aysy, O., 2005. Active guided evolution strategies for the large scale vehicle routing problems with time windows. Computers and Operations Research 32 (6), 1593–1614.

Nicholas Constantine Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of state calculation by fast computing machines. Journal of Chemical Physics, 21:1087–1092, 1953.

Perboli, G., Pezzella, F., Tadei, R., 2006. Eve-opt: an hybrid algorithm for the capability vehicle routing problem. DAUIN - Politecnico di Torino - OR/05/06 (2006).

Prins, C., 2004. A simple and effective evolutionary algorithm for the vehicle routing problem. Computers and Operations Research 31 (12), 1985–2002.

Ricciardi, N., Tadei, R., Grosso, A., 2002. Optimal facility location with random throughput costs. Computers and Operations Research 29 (6), 593–607.

Sand, G., 1981. Predicting demand on the secondary echelon: a case study. In: Schwarz, L. (Ed.), Multi-level production inventory control systems: Theory and practice. North-Holland Publishing Co., pp. 209–223.

Scott Kirkpatrick, C. D. Gelatt, and Mario P. Vecchi. Optimization by simulated annealing. Science, 220 4598:671–80, 1983.

Shifting Patterns. The future of the logistics industry (2016) PwC's future in sight series. https://www.pwc.com/sg/en/publications/assets/future-of-the-logistics-industry.pdf

Svoronos, A., 1988. Estimating the performance of multi-level inventory systems. Operations Research 36, 57–72.

Taniguchi, E., Thompson, R., Yamada, T., van Duin, J., 2001. City Logistics: Network Modeling and Intelligent Transport Systems. Pergamon, Amsterdam.

Toth, P., Vigo, D., 2002. An overview of vehicle routing problem. In: Toth, P., Vigo, D. (Eds.), The Vehicle Routing Problem. Vol. 9 of SIAM Monographs on Discrete Mathematics and Applications. pp. 1–51.

Verrijdt, J., de Kok, A., 1995. Distribution planning for a divergent n-echelon network without intermediate stocks under service restrictions. International Journal of Production Economics 38, 225–243.