Vehicle Routing Problem in XYZ company

Postgraduate thesis of
Kolyvakis Stavros

Supervisor: Diamantidis
Alexandros

November 2019
Abstract

In the middle of the Greek economic crisis, distribution companies try to minimize their logistics and distribution expenses. One mean to do this, is by examining the Vehicle Routing Problem (VRP), which is an expansion of the Travelling Salesman Problem (TSP). The scientific field of this thesis is the application of the operational research in logistics and distribution. In general, scientists, researchers and employees in the logistics sector occupy themselves a lot with the VRP and it’s solutions because it is considered an applicable problem with a lot of money and time saving applications in companies and organizations.

In this thesis we examine the application of Vehicle Routing Problem, in a logistics company. The data of the problem were entered in matrixes in excel and were processed with the help of Visual Basic for Applications (VBA). The algorithm which was executed with the help of VBA code was the Clark and Wright algorithm, both parallel and sequential version. The results of this algorithm were the optimal routes which can be used by the company in order to decrease its distribution expenses. The program, which was written in VBA and was executed in excel, uses permanent data like the location of the company’s customers, which of course were altered in this thesis, and changing data like the customer’s daily demand.

In the 1st chapter we have an introduction of the thesis and the VRP. Then, in the part number 1, the theory of this thesis is presented. To be more specific, the second chapter has to do with previous researches in this subject. In the third chapter we have a general presentation of the supply chain management. In the fourth chapter we examine some theory regarding the Travelling Salesman Problem which is considered a predecessor of the Vehicle Routing Problem. Moreover in the fifth chapter we examine the factors and the necessity of the VRP, the course of the VRP through time. Moreover in the fifth chapter we examine some variations of the VRP like VRP with backhauls, VRP with time windows and VRP with pick up and delivery.

In the 6th chapter we have a brief theoretical presentation of the exact algorithms, like the branch and bound algorithm and branch and cut algorithm. In the seventh chapter, there is a theoretical review of heuristic algorithms, more specifically, three subjects we examine are: constructive heuristics, two-phase algorithms and improvement heuristics. Moreover, in the 8th chapter, we introduce Clark and Wright algorithm in this thesis, as well as, the algorithm’s variations, the sequential and parallel version. In the 10th chapter, we present some information of the company in which the Clark and Wright algorithm will be applied. We present some historical data of the company, the systems which are applied in the warehouse and, briefly, some information about the company’s ERP and WMS. Finally, in the 10th chapter and the second part of the thesis, we have the theoretical, implementation of Clark and Wright algorithm in the company’s distribution schedule with the use of VBA code and its results.
Keywords: Vehicle Routing Problem, Logistics, Clark and Wright algorithm, parallel version, sequential version

Περιεχόμενα

Abstract ................................................................................................................................. 1

1. Introduction .................................................................................................................... 4

Part 1: Theoretical Review .............................................................................................. 5

2. Literature review .......................................................................................................... 5

3. Supply Chain Management ......................................................................................... 6

   3.1 Difficulties in supply chain ...................................................................................... 9

4 Traveling salesman problem .......................................................................................... 13

   4.1 Nearest neighbor algorithm .................................................................................... 15

5. Vehicle routing problem ............................................................................................... 18

   5.1 Historical review of the Vehicle Routing Problem ................................................. 20

   5.2 Factors affecting the problem .................................................................................. 22

   5.3 The need for routing optimization .......................................................................... 23

   5.4 The vehicle routing problem with backhauls ......................................................... 25

   5.5 The vehicle routing problem with pickup and delivery ......................................... 27

   5.6 The vehicle routing problem with time windows .................................................... 29

6. Exact algorithms .......................................................................................................... 32

   6.1 Branch and bound algorithm .................................................................................. 32

   6.2 Branch and Cut algorithm ...................................................................................... 33

7. Heuristic algorithms .................................................................................................... 36

   7.1 Constructive Heuristics .......................................................................................... 36

   7.2 Two-phase algorithms ............................................................................................. 38

   7.3 Improvement heuristics .......................................................................................... 39

8. Clarke and Wright algorithm ......................................................................................... 40

   8.1 Parallel version ........................................................................................................ 43
8.2 Sequential version .........................................................................................44

9. Case study ........................................................................................................45

9.1 Presentation of the company ........................................................................45

9.2 Historical review of the company .................................................................45

9.3 Picking systems inside the warehouse .........................................................45

9.4 XYZ company’s WMS ..................................................................................47

9.4 The application of pick-by-light systems in XYZ company ......................53

9.5 XYZ company’s ERP ......................................................................................55

9.6 XYZ company’s mobile ERP ........................................................................56

Part B: The application of the algorithm ..........................................................56

10. The application of Clark and Wright algorithm in XYZ company ...56

10.1 Clark and Wright algorithm (Parallel Version) ........................................62

10.2 Clark and Wright algorithm (Sequential Version) ..................................76

Conclusions .........................................................................................................85

Future research .................................................................................................86

References ...........................................................................................................86
1. Introduction

Nowadays, the economy has been globalized. The reason behind this globalization is partially because of the relocation of companies. For example many companies moved their production lines at countries in east Asia like China, Taiwan or Bangladesh. But still these companies want to sell their products in western countries where they will acquire more margins and higher selling prices. Consequently there is an increasing need for mass transportation of products in a cheap and efficient manner. Furthermore there is a rapid increase in the introduction of new products in the market. The negative result of this situation is the increasing costs of transportation, for instance the fuel costs, environmental pollution and road accidents (Alexiou & Katsavounis, 2015). As a result the development of the scientific field of supply chain management is more essential than ever. Nowadays with the contribution of mathematics and informatics, supply chains aim to minimize transportation and operational costs as well as minimize mistakes and malfunctions across the supply chain. Regarding the aforementioned contribution of maths and algorithms in supply chain management, in this paper we are going to analyze the application of the Vehicle Routing Problem in a logistics company. Let’s say for example that we have a logistics company which transports products from a central depot to the company’s customer’s retail shops, which customers have a certain demand. The Vehicle Routing Problem aims to find the optimal routes so that the demand of the retail shops to be satisfied with the minimum feasible cost and with the minimum number of trucks (Alexiou & Katsavounis, 2015).
Part 1: Theoretical Review

2. Literature review

Rodríguez-Martín et. al. (2019) conducted a research about a variant of the Periodic Vehicle Routing Problem which is called Periodic Vehicle Routing Problem with Driver Consistency. In this case the optimal routes are calculated for several days forward and each customer can set some rules about the hours when the products can be delivered to them (Rodríguez-Martín, Salazar-González, & Yaman, 2016). Stavropoulou et. al. (2019) solved and modeled the Consistent Vehicle Routing Problem with Profits using AdeptiveTabu Search. The researchers used computational experiments in order to demonstrate the effectiveness of the algorithm. Alexiou and Katsavounis (2015) made a research about a variation of VRP incorporating factors of transportation costs in conjunction with adverse situations. They developed a branch and bound tree search algorithm in order to find out the optimal routes and they used graph theory in order to find the optimal locations of the depots. Marinakis et. al. (2006) formulated the VRP as a problem of two decision levels. At the first stage of their project they used a genetic algorithm in order to assign each customer to a specific route. In the second level the Travelling Salesman Problem was solved independently for each member of the population and for each assignment to vehicles. Ibarra-Rojas et. al. (2017) examined a Vehicle Routing Problem to optimize approachability based on six indexes:

1) The amount of areas with access to opportunities with defined mobility,
2) The amount of areas covered by the itinerary
3) The cost of travel
4) The distance to the nearest opportunity
5) The amount of opportunities
6) The spatial disaggregation

They used a mixed-integer linear formulation for the examined problem. Also they designed an iterated local search algorithm and analyzed its efficiency pursuant to a benchmark of randomly generated instances. Torki et. al. (1997) applied a self organization Neural Network algorithm in a category of Vehicle Routing Problems. The researchers noticed that neural networks can have a outstanding performance in the Travelling Salesman Problem, so they
developed the algorithm with the intention of extending its applicability into more complex problems. Their first application of the aforementioned algorithm is on the Multiple Traveling Salesman Problem. Then they included into the algorithm some additions in order to satisfy some additional constraints.

Letchford et. al. (2018) used old and new relaxations that could be solved in pseudo-polynomial time, which were based on column generation, with the purpose to solve the Capacitated Vehicle Routing Problem (CVRP) which is NP-hard. They also examined the consequences of inserting some known inequalities. Bruglieri et. al. proposed a two-phase solution approach for the Green Vehicle Routing Problem which has to do with the routing of vehicles that use alternative fuels, the base of these vehicles is at a single depot. The goal of their project was to minimize the total distance and the same time satisfy all the customers. These type of vehicles have a particular peculiarity, they have limited fuel capacity, consequently it may be necessary for them to stop at Alternative Fuel Stations (AFSs) during their trip. The researchers suggested a solution in which many paths compose a route. The goal of the solution is to eliminate the stops at AFSs. In the first phase, dominance rules were used in order to limit the number of generated feasible paths. In the second phase, the researchers used Mixed Integer Linear Programming in with the intention select and combine the paths in order to generate the routes.

Arnold and Sörensen (2019), in their paper, used local search in order to solve the Vehicle Routing Problem. Many state-of-the-art heuristics include local search in their methods. They synthesized three powerful local search methods, and tried to make their implementation as efficient as possible so that the computational effort would be minimum. In their paper they managed to show that a local search can be used in order to provide high-quality solutions in short time. The researchers aimed to show if perturbation and pruning can be used together with local search, and made a series of experiments (Arnold & Sörensen, 2019).

3. Supply Chain Management

The goal of the supply chain is to move a product from supplier to customer. Inside supply chain there is a system of organizations, people, activities, information and resources. The final goal of the supply chain is to take all the
raw materials, natural resources and components and with the appropriate processes and transportations to deliver them as a finished product to the final customer. Supply Chain Management (SCM) deals with the planning and management of all actions, processes and operations which are involved in sourcing, procurement and conversion. Another specifying of supply chain management is that it combines supply and demand management within and across companies. Also supply chain management can be considered as a bond between business functions and business processes inside a specific company or even across different companies. Moreover supply chain management must be coordinated with another sectors of management like processes and activities, marketing, sales, product design, finance and information technology.

In general, supply chain management is the management of the conversion of raw materials into final products, this includes the flow of goods, the flow of services and also all the necessary processes. The activities of supply chains interact with production, product development and information systems. Through SCM suppliers try to develop efficient and economical supply chains. Consequently suppliers aim to maximize customer value and make their companies competitive in the marketplace. Recently the companies started to recognize the importance of SCM and how it can add value to their processes. The companies try to manage efficiently internal inventories, internal production, distribution, sales, and the inventories control of the company’s suppliers. The goal of supply chain is to minimize or eliminate unnecessary costs and limit the lead time of the order. Finally, the objective of SCM is to centrally control or connect the manufacturing, transportation and dispensation of an item (Hayes, 2019).

The responsibility of SCM is to supervise each touch point where two or more companies cooperate in order to deliver the product or the service to the final customer. If the number of touch points across the supply chain is large then there are many events, processes and operations which can add value through efficiency or increase the total costs if they are designed poorly. Consequently the appropriate management of the supply chain can increase profits, reduce expenses and make the whole operation of the companies more efficient (Hayes, 2019).

For example Walgreens Boots Alliance Inc. is the second largest pharmacy chain in the United States. For a company of this type it is important to have efficiency in its supply chain in order to have a competitive advantage against its competitors. Consequently this company tries constantly to develop and enhance its supply chain. Walgreens has invested, among others, in the data analysis regarding the supply chain. Walgreens has developed a system where, based on data of the past and analytics, forecasts demand and customer behavior. For instance, the company predicts, based on patterns, the anticipated cases of flu and then estimates the needed inventory for the flu remedies, the
goal of this project is to minimize the stock. As a result the company can minimize all the unnecessary costs, such as the cost of inventory keeping and transportation (Hayes, 2019).

The demands of modern society from the commercial world call on those responsible to pay a lot of resources in the field of management. The problems to be solved mainly consist of the construction of a modern free market, combined with the evolution of the Internet and new marketing methods (Christopher, 2016). In this context, managers need to be able to recognize the market situation, which is more modest than the one that was in place a decade ago. Now people are able to meet all their needs over the internet, while the largest share of the market is owned by companies, which can meet the high demands for speed and quality. Managers now understand that they cannot lead their companies to success if they do not properly coordinate between departments in order to increase the quality of the services provided to customers (Christopher, 2016).

The term "supply chain management" first emerged in 1982 when companies first introduced the process of producing products, not the effort of a unit. On the contrary, it is a collective process that consists of the base, which is the raw material production units, the middle, namely the process of designing, assembling and producing the products, and finally the customers, namely the distribution process and the comments about with the quality of the product (Hugos, 2018).

Supply Chain Management is a very important sector that is constantly evolving, as well as methods and means of business management. In particular, to take a business decision, one has to take into account different aspects such as how the system flows, the political landscape, organizational forms and investment choices. As a result, theories in the field also focus on organizational change. Every person who works in the supply chain is part of the business and vice versa (Akkermans, Bogerd, Yücesan, & Van Wassenhove, 2003).

In order to develop a business, analytical techniques must be applied that recognize the importance of total quality management that focuses on factors
such as customer and employee satisfaction and security at every stage of production. In addition, a strong emphasis is placed on linking the market with the company and the national economic targets. This trend is observable worldwide and affects the way that businesses operate at a global level (Kotter, 2009).

Nowadays, supply chain management or SCM also involves the executive responsibilities of businesses. Thus, based on what has been discussed above, it emerges that supply chain issues include customer management, marketing and product promotion. The cost factor and the time factor are decisive (Chkanikova & Mont, 2015). Specialization creates a relationship of interdependence between production units. Incorrectly many managers believe that fulfilling the goals of the secondary production company will lead to the achievement of primary unit targets. However, the reality is that even within the supply chain, different companies need to be able to fulfill their goals autonomously in order to benefit from co-operation (Monczka, Handfield, Giunipero, & Patterson, 2015).

3.1 Difficulties in supply chain

Nowadays the supply chains are more complex than ever. Companies try to monitor their supply chains which is very difficult. In general supply chain has many risks and there are many things which might go wrong. There are three key issues that must be taken seriously by logistics companies:

1) Supply chain visibility
2) Cyber risk
3) Natural disasters

If just one of the aforementioned go wrong then there is going to be a massive negative impact in the operation of the supply chain (Womack, 2017). Due to the rapid changes and challenges in the supply chains, the use of new technologies like Internet of Things, Big Data, geocoding etc. has become more essential than ever. Consequently supply chain managers must constantly educate and train themselves in all these new technologies and processes in order to be able to adjust in all these fast-paced changes. Also another challenge of the supply chain professionals is to monitor all these changes and make them beneficial for the company’s operation (Womack, 2017).

All this technology entering the operation of supply chains forces them continuously to change systems, practices, functions and operations. As we aforementioned, technology can become a useful tool in the hands of supply
chain companies, but like every tool it can be harmful too. For example there is a big chance of cyber attacks and viruses. An email, an attachment or even a public Wi-Fi can be considered as menaces. It is very important for the employees to understand the nature and importance of these dangers in order to keep the company’s data safe (Womack, 2017). Another unexpected challenge for logistics companies is the sudden changes of weather and natural disasters. Usually these events happen suddenly, and their handling is extremely hard. Consequently, companies should have designed plans and operations in case these extreme weather conditions take place.

Despite the important advantages of an efficient supply chain, achieving often encounters significant barriers that may lead to a reduction in performance. Such problems are encountered by all companies in the modern business world. Many of these are caused by topical geopolitical conditions (e.g., Iran and the US), or by technological advances that are moving at a rapid pace (Matthews, Power, Touboulic, & Marques, 2016). The key challenge in the supply chain is that the management can produce in a short time and without harm in a context that requires good coordination and organization of resources. More generally, because all businesses are organized for profitability and because competition is more global, flexibility and security are necessary (Chkanikova & Mont, 2015).

In summary, the main difficulties are five (Giannakis & Papadopoulos, 2016):

- Customer service
- Determination of costs
- Planning risk
- Inter-company relations
- Specialization

In terms of customer service, the difficulty concerns publicly viewing comments. Maintaining a satisfied customer base is particularly difficult due to the very high requirements and the many alternatives available to people. At the same time, all clients have the opportunity both to publicize and criticize the company that satisfied or did not satisfy them, publicly (Simpson, Meredith, Boyer, Dilts, Ellram, & Leong, 2015). This creates significant complications in the modern, often changing market, as the majority of customers who are able to buy a product also have a range of alternatives. The
result of the above is that companies need to invest a large part of their resources in customer service. Large companies are able to do so because of the large capital available, the introduction of the brand and the large available human resources (Simpson, Meredith, Boyer, Dilts, Ellram, & Leong, 2015).

Small companies, on the contrary, are trying to manage their available resources in the best possible way, so that they can maintain a satisfactory percentage of satisfied customers. Many times, however, they make the mistake of doing so at the expense of product quality, which always leads to negative results (Reimann & Ketchen Jr, 2017).

Next on the list is cost control, it is the determination of cost of sales based on production costs. In this case, geopolitical conditions, as well as transnational trade agreements, play an important role. It is virtually impossible for a company or group to be able to produce all the necessary materials for the construction of a product, and even in this case, the cost is influenced by the cost of the fuel (Quarshie, Salmi, & Leuschner, 2016). This is a real management problem, as large companies operating at an international level are directly dependent on local businesses, agreements and fuel prices. Other factors that directly affect the cost are the legislative labor regulations of each state, the social requirements of the population, as well as the level of technology in each region. All of the above can cause a serious drop in customer satisfaction, as price-quality relationships of the product are one of the key criteria underlying consumer trends (Markman & Krause, 2016).

The above contributes to the risk programming problem. Market elasticity, as well as temporary changes in the state of the production base, whatever the way they may arise, require extensive study at depths of at least 20 years. The study explores the geopolitical conditions, the economy's resilience, consumer trends, as well as the aptitude and planning of the company (Tseng, Lim, & Wong, 2015). This is an important management barrier to the company's course, which can prevent it from achieving its goals. For this reason, major marketing departments are conducting extensive analyzes using advanced analysis and forecasting systems to be able to predict the future development.
of the plan with the least possible margin of error (Chkanikova & Mont, 2015).

Another interference can be inter-company relations between producer and supplier. These are agreements between companies to optimize their logistics chain. However, the two parties must be able to satisfy the agreed conditions, as otherwise significant complications are created in the production process (Tang & Tomlin, 2008).

If, for example, the supplier delivers less material than the agreed one, then there will be serious production problems. Accordingly, if the producer neglects his obligation to pay the price for the materials he has supplied, then the supplier will not be able to fulfill any future orders (Rao & Goldsby, 2009). The last obstacle is also the main one on the list. In the modern age, the economic world is developing significant needs for specialization by its employees. Technologies are evolving at such a rapid rate that educational institutions are unable to adapt them to their curricula, resulting in people with a specialization in a field but only 40% of the technology used (Rao & Goldsby, 2009).

Particularly in the case of the supply chain, the problem is more intense, as it consists of a wide range of parts, each with its own requirements. Supply chain specialization needs are greater due to the fact that the supply chain is not an integral part but a network of cooperating departments and businesses. The need arises from the fact that if one of the segments in the chain grows to a much greater degree than the rest, then one more inequality in the chain operation is caused. This problem can only be solved within businesses, which have to invest more of their resources in training their staff on new technologies (Chopra & Sodhi, 2004)
4 Traveling salesman problem

In the travelling salesman problem (TSP) we have a depot and a set of customers, for example. The goal of TSP is to start from the depot and visit each node precisely one time and then return to the depot. TSP tries to find the optimal solution, more specific, the minimum total travel cost or distance. Generally, scientists and researchers from the scientific disciplines of mathematics and operations research have studied this problem a lot. There are many case studies and articles that deal with this problem. The main reason for this is the fact that TSP has many applications in real life problems, for example in the sector of logistics and more specific in the distribution of products, also in combinatorial optimization TSP has been the prototype problem for theoretical developments. For example some good reference books are Lawler, Lenstra, and Rijnnooy-Khan (1985) and Gutin and Punnen (2002), among many others. Frequently, usual variations happen either in the contexts of scheduling, or sequencing, or routing. In the present, scientists and researchers are able to solve problems with up to several thousand cities. Heuristic methods can find solutions which are, simultaneously, near to the optimal solution and with a reasonable computational effort (Sun, Diaby, & Karwan, 2018).

When more cities or nodes, according to the case, are included into the problem then the worst-case running time for any method-algorithm for the TSP is growing superpolynomially (however not exponentially). The TSP is one of the most popular problems in the sector of operational research and theoretical computer science. It was first appeared in 1930. Moreover TSP is widely applied in sectors such as planning, logistics and in the production of microchips. Also TSP has applications in DNA sequencing. For example in the application of TSP in logistics the concept city is transformed into customers, or in the case of microchips in soldering points. Final in the case of DNA sequencing the concept “distance” equals with similarity measure between DNA fragments (Schrijver, 2017).

The first appearance of the travelling salesman problem was in a handbook from 1832, which contains instance paths in Germany and Switzerland. But this book doesn’t include mathematics. In the 1800s the Irish mathematician W.R. Hamilton and the British mathematician Thomas Kirkman were the first who formulated the TSP mathematically. In 1930, mathematicians from Vienna and Harvard studied the TSP, including Karl Menger. The first mathematical approach of the problem was in 1930 from Merrill M. Flood who was trying to find the optimal route for a school bus. Later Hassler Whitney at Princeton University introduced the definition: “travelling salesman problem”. The travelling salesman problem started to become more popular in the scientific community of Europe and USA in the 1950s and 1960s. This happened because RAND Corporation in Santa Monica funded the research towards better solutions of the problem (Johnson & McGeoch, 1997).
George Dantzig, Delbert Ray Fulkerson and Selmer M. Johnson which worked for the Rand Corporation made some noteworthy contributions towards the optimization of the provided solutions of the TSP. Now regarding the mathematical formulation of the problem, TSP can be considered as an undirected weighted graph. The customers or cities, depending on the problem, can be depicted as the nodes of the graph and the routes or paths are the edges. Also the travel time, distance etc can be considered as the edge’s weight. The goal of the problem is to find the route with the minimum distance that includes all the nodes of the graph and starts and finishes into the same node. Also every node must be visited only one time. In most cases the model is a complete graph, which means that every node is connected with every other node. In the case that two cities or customers are not linked by a path, then we add a path between them with a very high cost, consequently the graph will be completed without any impact on the optimal tour.

The TSP can be formulated as an integer linear program. The cities (nodes) can be represented with numbers from 1 to n. There is a variable

\[ x_{ij} = \begin{cases} 
1, & \text{if a path from node } i \text{ to } j \text{ exists} \\
0, & \text{otherwise}
\end{cases} \]

The variable \( c_{ij} \) can be considered as travel time or cost of moving from city \( i \) to city \( j \).

The total cost of the optimal route can be depicted by the following formulation:

\[
\min \sum_{i=1}^{n} \sum_{j=1, j \neq i}^{n} c_{ij} x_{ij} ;
\]

\[
0 \leq x_{ij} \leq 1 \quad i, j = 1, \ldots, n ;
\]

The equalities below make sure that the arrival to each city is done by only one city and from each city there is a departure to exactly one other city (Bektaş, Demir, & Laporte, 2016).

\[
\sum_{i=1, i \neq j}^{n} x_{ij} = 1 \quad j = 1, \ldots, n ;
\]

\[
\sum_{j=1, j \neq i}^{n} x_{ij} = 1 \quad i = 1, \ldots, n ;
\]

The Traveling Salesman Problem (TSP) is a mathematical problem of finding the optimum path between a large number of points with known distances.
between them. This is one of the most important supply issues, as it is being used by companies to reduce the cost of the route (Taillard & Helsgaun, 2019). This problem first emerged in 1930 by American mathematician Merril M. Flood, who was summoned by the mayor of Washington to identify the best routes for city buses in order to reduce fuel costs (Taillard & Helsgaun, 2019).

In the coming decades, many mathematicians have dealt with the problem. In recent years, and thanks to advances in IT, algorithms have been developed that can provide a solution to this problem with great precision (Benavent, Landete, Salazar-González, & Tirado, 2019). These algorithms apply to almost all of the companies involved in the supply or participate in it. The solution resulting from the use of the algorithm offers the company the ability to minimize the time and cost of routes without reducing the level of service quality (Benavent, Landete, Salazar-González, & Tirado, 2019).

4.1 Nearest neighbor algorithm

One of the first algorithms that were used to solve the travelling salesman problem was the nearest neighbor algorithm. In the solution of the algorithm the “traveler” strarts from one random node and then recursively visits the node with the minimum distance. This method gives a solution, but usually there are other solutions also with less cost. The result of the algorithm is the order by which the “traveler” visits the nodes of the path. In general, the application and implementation of the algorithm is easy. But sometimes the route that the algorithm finds may not be the optimal. A common practice is that if the last two edges of the route have similar cost (length) with the last two edges then the resulting route is reasonable. On the other hand, if their cost is very different then there is a big probability that a better tour exist. Furthermore, another method to check the optimality of the tour is to use the lower bound algorithm. The worst case scenario is that the result of the algorithm will be much different from the optimal solution. Also there is a possibility that the algorithm will not find a solution at all, even if a solution exists.
For example in the above picture, we can see a graph with five nodes and 10 edges. We choose as a starting node the node A. Then according to the nearest neighbor algorithm we choose the edge with the less cost and we transit to node D. Then, from node D the edge with the minimum cost is the edge that leads to node E. Afterwards, using the same logic we go to node C and then to node B. Consequently we have visited all the nodes once. The route of this graph is the following: A->D->E->C->B. The total cost of the route is equal to 185+302+165+305+500=1457. As we mentioned before this route may not be the optimal but is a feasible solution.

The NN algorithm is the first attempt to solve TSP. The algorithmic sequence detects the points in a given space and starting from the first, it sequentially locates the nearest one, until a path that passes through all the points occurs (Kyritsis, Blathras, Gulliver, & Varela, 2017). The algorithm follows some steps to perform its function. For example, space A, with a number of N points, points visited and points not visited u, then the algorithm will be (Todosijević, Mjirda, Mladenović, Hanafi, & Gendron, 2017):

- Convert all points to u

- Select a random point u
- Find the nearest next u point and convert it to v
- Definition of v as the reference point
- Logical control of the presence of other u points in the space
- If yes, repeat 3
- Otherwise, stop

The first applications of the algorithm concerned mainly the finding of optimal routes, between a small number of cities, for transport companies. However, developments in the field allow its extensive use in other areas of science (Todosijević, Mjirda, Mladenović, Hanafi, & Gendron, 2017). Sectors such as meteorology, astrophysics, geology, microbiology, and so on, use the algorithm to perform complex experiments and to gather data that would otherwise not be able to detect (Srinivasan, Satyajit, Behera, & Panigrahi, 2018).

Apart from the scientific field, the algorithm is also used in everyday applications such as GPS, closed-circuit television cameras, face recognition, and more. Various variants of the algorithm are used by online companies to identify patterns in the purchasing trends of the public (Srinivasan, Satyajit, Behera, & Panigrahi, 2018).
5. Vehicle routing problem

The target of Vehicle Routing Problem (VRP) is to visit all customers with the minimum cost using a fleet of vehicles. Also the VRP has been studied from scientists and managers for many years (Laporte, 1992, Laporte et al., 2000, Toth and Vigo, 2002). The VRP is NP-hard, this shows that the problem is complex and difficult in it’s resolution. Scientists use heuristic methods in order to solve combinatorial optimization problems (Cordeau et al., 2002, Laporte et al., 2000) . Vehicle routing problem is a generalization of the travelling salesman problem. Because of the size and frequency of the real world VRPs commercial users frequently use heuristics. The VRP aims to determine a set of routes, S.

We have one route for each vehicle which point of departure and endpoint are the depot. The cost can be calculated in multiple ways such as monetary, distance or travel time. Using mathematical terms the road network can be described as a graph where the demand points are the nodes and the arcs are the roads between them. According to the structure of the road network and the nature of the distribution problem the arcs can be directed or undirected, because there may be one way streets or different costs in each direction. Each arc can have a cost which depends of its length or travel time. Another factor that affects the cost can be the vehicle type. The global cost of each arc or route, the travel cost or the travel time between each customer and the depot must be known. These are the input data of the problem.

The final solution of the problem, which is the optimal routes with the minimum cost is based on these data. For each pair of vertices i and j, there exists an arc (i,j). The sum of the travel times of each arc which consist the route is the total travel time of each optimal route. The objective function of the VRP is not always the same. It depends on the peculiarities of the problem. But there are some general similarities in each problem, for example the final objective is the minimization of the total transportation cost, which is based on the total travelled distance and in some fixed costs such as salaries and the service of the vehicles. Moreover another general goal of the VRP is to minimize the necessary number of vehicles, drivers and the total needed vehicle capacity (Alexiou & Katsavounis, 2015).

Moreover there are variations and specializations of the vehicle routing problem. In later chapters these variations are explained more thoroughly. Now we are going to give a brief explanation of the most well known variations.

1) Vehicle Routing Problem with Pickup and Delivery (VRPPD): An amount of products need to be transferred from a particular pickup location to another delivery point. The objective is to find the optimal routes and the minimum needed amount of vehicles in order to visit the pickup and drop-off locations.
2) Vehicle Routing Problem with LIFO: Similar to the VRPPD, but with one addition. At any customer the item being unloaded must be the item most recently picked up. This system saves time for the driver and the employees because there is no need to unload other products other than the ones that need to be delivered to the particular customer.

3) Vehicle Routing Problem with Time Windows (VRPTW): The products must be delivered in certain time windows and not whenever the driver and the company can.

4) Capacitated Vehicle Routing Problem: CVRP or CVRPTW. The transportation trucks don’t have unlimited capacity. On the contrary their capacity has a certain boundary.

5) Vehicle Routing Problem with Multiple Trips (VRPMT): The vehicles can do more than one itineraries.

6) Open Vehicle Routing Problem (OVRP): Vehicles are not required to return to their depot.

The vehicle routing problem is solved with the help of algorithms, so there are many softwares which are used to solve the multiple VRP problems. Also there is a certain amount of scientific articles and books which analyze the VRP. Finally the VRP has many similarities with the Job Shop Scheduling Problem, although different methods are used to solve this problem (Alexiou & Katsavounis, 2015).

The problem of routing vehicles refers to the process of finding the optimal route. This problem is a development of TSP, considering the cost factor and time-saving. Unlike TSP, the VRP only covers transport services (Braekers, Ramaekers, & Van Nieuwenhuyse, 2016). The first capture of the problem occurred in 1965 by an American oil company. With the increase in demand for fuel, it ended up consuming more oil per transport than the profit of the fuel transported (Montoya-Torres, Franco, Isaza, Jiménez, & Herazo-Padilla, 2015).

However, beyond its ground-based implementation, the problem is being applied to both naval and airborne, with many transatlantic transport companies, to apply its treatment algorithms to reduce their operating costs. The problem is solved by three basic methods (Caceres-Cruz, Arias, Guimaranus, Riera, & Juan, 2015):

- Vehicle Flow Formulations - This method involves the application of integer coefficients in some linear transports, while the solution results from the number of nodes generated by the passage of the vehicle from these areas.
This method is "simplistic" in terms of methods of solving the problem, and overlooks many of the factors that affect the transfer process to the real (Maciejewski, Bischoff, Hörl, & Nagel, 2017).

- Commodity Flow Formulations-this method is essentially a more detailed VFF, essentially an increase in the problem parameters, adding the weight parameter, and the criticality of the commodity. This method is not widely used as it does not offer many benefits to transport companies (Letchford & Salazar-González, 2015).

- Set Partitioning problem - This method implements a partition in the system of paths, with variables relating to each segment separately but also connected. This is an attempt to segment the transfer process, taking into account the delivery intermediaries and other means at the disposal of the company. This method is widely used because it takes into account all of the company's mobile and stationary assets, producing a result of a total reduction in transport costs (Zaghrouti, El Hallaoui, & Soumis, 2018).

5.1 Historical review of the Vehicle Routing Problem

It was first noticed in the bibliography by George Dantzig and John Ramser in 1959 where it was applied in petrol deliveries. The research of the authors introduced a simple matching-based heuristic for its solution and represented it on a toy-sized example. Lateron, several heuristics were made based on principles like savings, geographical proximity, customer matchings, as well as intra-route and inter-route improvement steps. Later Dantzig and Ramser’s approach was enhanced by Clarke and Wright which made an approach which called the savings algorithm. This approach is also the most famous heuristic. Moreover, the advantage of this approach is that it is fast, simple and accurate. In 1981 started the growth of exact algorithms for the VRP with the publishing of two scientific articles by Christofides, Mingozzi and Toth in Networks, Christofides et al. (1981) and Mathematical Programming, Christofides et al. (1981). The first of these papers proposed an algorithm based on dynamic programming with state-space relaxation while the second proposed two mathematical formulations making use of $q$-paths and $k$-shortest spanning trees. A few years later, Laporte, Desrochers and Nobert, Laporte et al. (1984) introduced the first cutting plane approach for a VRP based on the solution of linear relaxation of an integer model. These influential ideas have affected some of the more recent algorithms.
For the latest half of the last century the VRP has been recognized as a subject of much importance in the operational research community. The reason for this is the fact that the study and “solution” of the problem can result in an important saving of costs for companies. Also the application of the VRP in real life problems has a lot of difficulties and a lot of parameters and restrictions which must be taken into account. In particular, the traveling salesman problem (TSP), which is a special case of the VRP, can now be calculated for many vertices. On the other hand, the vehicle routing problem is much more difficult to be solved. For instance, even the most simple form of VRP, which is the capacitated VRP (CVRP), is still difficult to be solved with one hundred and fifty customers or two hundred customers with the help of exact algorithms. Recently, the researchers have focused on the development of powerful metaheuristics.

The first of these papers proposed an algorithm based on dynamic programming with state-space relaxation whereas the second proposed two mathematical formulations making use of $q$-paths and $k$-shortest spanning trees. A few years later, Laporte, Desrochers and Nobert, Laporte et al. (1984) proposed the first cutting plane approach for a VRP based on the solution of linear relaxation of an integer model. These seminal concepts have made their way into some of the more recent algorithms (Bahri, Amor, & Talbi, 2016).

Since then, researchers and scientists have proposed many exact algorithms which are based on mathematical programming formulations. Branch and Cut method is widely used for formulations which have to do with vehicle flow or commodity flow variables. The VRP can also be formulated as a set partitioning problem to which some valid inequalities are added. Fukasawa et al. (2006) and by Baldacci et al. (2008) used these systems in order to do some successful applications (Hoffmann, Chalmers, Urquhart, & Guckert, 2019). In the early 1990s we can notice the beginning of the modern heuristics for the VRP, with the introduction of metaheuristics. The study of the VRP and other similar optimization problems had as a result the growth and understanding of several metaheuristic concepts we know now. The early research in this area was quite fragmented, with an obvious preference towards tabu search-based approaches and some of the algorithms were over engineered, but that situation has changed in recent years (Todosijević, Mjirda, Mladenović, Hanafi, & Gendron, 2017).

The metaheuristics that at the same time conduct a broad and deep search of the solution space and are able to solve multiple versions of the problem are considered the best. Their method has to do with applying multiple operators, like in adaptive large neighbourhood search, or with the combination of genetic search with local search, as for example in the hybrid genetic algorithm which was recently proposed by Vidal et al. (2012).
5.2 Factors affecting the problem

Solving the problem of vehicle routing requires a profound understanding of its growth factors. These factors may vary from geography, the state of the economy, the environment, etc., and it is, therefore, necessary to define certain rigid conditions (Bektaş, Demir, & Laporte, 2016). This problem occurs in many different conditions, either in a provincial area with long distances between destinations or in areas with poor road quality or a high-density urban environment. It is therefore impossible for a developer to create a universal algorithm that will apply to all cases (Qian & Eglese, 2016).

For this reason, three examples will be used to identify the factors. The first is a classic example of the problem of vehicle routing and is a high-density urban environment. The second is a key economy for the economy, and it is a sparsely populated area with long distances between nodes. The third and last one refers to an area where the poor economic situation is reflected by the lack of infrastructure (Qian & Eglese, 2016). In an urban environment, the problem of routing vehicles is the continual rotation of destinations. For example, a packet transport company is forcing its vehicles to run short trips, many times the routes overlap each other. Costs in these environments are particularly high due to the high incidence of accidents, insurance costs and increased use of the company's vehicles (Qian & Eglese, 2016).

Statistically, a vehicle driven exclusively in a high-density urban environment has an increased wear index compared to a vehicle that runs long journeys in sparsely populated areas (Psaraftis, Wen, & Kontovas, 2016). At the same time, an important factor influencing the growth of the problem is also the factor of chance, in a densely populated urban environment there is an increased chance of accident involvement. Accidents not only increase vehicle maintenance costs but also affect the public image of the company, regardless of who was responsible (Psaraftis, Wen, & Kontovas, 2016).

As mentioned above in the first example, compared to a densely populated environment, a sparsely populated environment has certain advantages. However, the problem of vehicle routing remains equally intense for other
factors (Jabir, Panicker, & Sridharan, 2015). In a sparsely populated environment, the routes are far greater than an urban one, basically, this means additional costs for the company. These long journeys remain, in some cases, the only option, without the possibility of finding alternatives (Jabir, Panicker, & Sridharan, 2015).

At the same time, it is well known that the cost of fuel increases depending on the difficulty of transporting it. In an urban center where the existence of a financial hub is at a short distance, the adjusted fuel price will be lower than a provincial city, which is a long way from the junction. In the third case where infrastructure is inadequate, making routes is a serious problem for companies, given the delays and damage caused (Kim, Ong, Heng, Tan, & Zhang, 2015).

It is therefore understood that the existence and increase of the routing problem of vehicles always depend on environmental conditions. However, it is also clear that the problem exists irrespective of the prevailing conditions, which in practice means that in any environment, in any circumstances, the problem of routing will exist, while what changes are the degree, the intensity and the ability to cope (Kim, Ong, Heng, Tan, & Zhang, 2015).

5.3 The need for routing optimization

The applications of VRP are many and vary from company to company. Generally transportation cost is 10% of the total product’s cost. The usage of optimization software can reserve savings to a company up to 5%. Also the transportation sector makes up to 10% of the EU’s GDP. The importance of the transport sector in all areas of everyday life of citizens and the problems caused by this sector are the main reasons for the need to optimize transport. The transport sector is the main part of the modern global economy and social system. At the same time, this sector is particularly sensitive, since at the state level it is heavily influenced by transnational agreements and controversies (Sicilia, Quemada, Royo, & Escuín, 2016).

At the local level, and particularly in large urban centers, the importance of the sector is forcing the authorities to have high priority, with infrastructure and facilities to keep the flow of goods intact. The need to implement optimization systems is necessary because of the great needs in the social and economic structure of large urban centers (Hoffmann, Chalmers, Urquhart, & Guckert, 2019). Practically this need arises for three main reasons (Bahri, Amor, & Talbi, 2016):
- First, speed. Modern society is structured on the foundations of digital systems. All areas of everyday life of citizens have become faster and more efficient compared to the conditions that prevailed just two decades ago. All areas such as shopping, work, communication, and needs have increased their speed (Heilig, Lalla-Ruiz, & Voß, 2017).

- Second environment. Social sensitivity to climate change and environmental policy has changed in recent years. Environmental sensitivity and the continued demand for green technologies and policies have led to a large number of companies to change the way they operate (Kalayci & Kaya, 2016).

- Third, the economy. The third reason is related to the second but with a substantial difference. In the second case, the incentive is economical to environmental sensitivity. The cost of providing a service determines the amount of profit and the extent of the clientele. For this reason, many companies choose to revise their traditional practices in order to increase their clientele and to claim a larger market share (Greiner, Periaux, Quagliarella, Magalhaes-Mendes, & Galván, 2018).

The correlation of route optimization systems in logistics with the above factors is self-evident. In the case of speed, the implementation of these systems allows the company to determine the fastest routes, thus reducing the required service delivery time. In practice, this means that the percentage reduction in time is equal to the percentage of the potential increase of the clientele (Mirmohammadi, Babaee Tirkolaee, Goli, & Dehnavi-Arani, 2017).

With regard to the environment, an increasing number of countries on the planet, with most of them occupying most of the market, have adopted a series of environmental measures to reduce the impact of global warming and prevent it from growing. The adoption of optimization systems by the logistics companies will allow for a reduction in the number of journeys in the already polluted urban centers. This reduction will strengthen the social image of the company, resulting in an increase in the clientele (Mirmohammadi, Babaee Tirkolaee, Goli, & Dehnavi-Arani, 2017).
The third case, which is the cost, is the main reason for creating these systems. Reducing the aforementioned routes will entail reduced direct and indirect costs, while customer growth will translate into net profit (Zuo, Zhu, Huang, & Xiao, 2017). The most important factor in the need for system implementation, however, is the existence of the system itself and its proven benefit. It is one of the main features of the economic world, the need for synchronization and the need to break the previous record. This need lies at the base of a free market where businesses compete with one another in order to grow in size. This conflict offers a variety of benefits to people who enjoy high-quality products and improved services in this way (Zuo, Zhu, Huang, & Xiao, 2017).

5.4 The vehicle routing problem with backhauls

The problem of routing vehicles with backhauls is defined as the pickup/delivery problem where all pickups must be made after all deliveries have been completed. This problem is characterized as the most difficult category of VRPs due to the significant increase in timetables. This case requires a continuous flow of information, which mainly concerns the capacity of the vehicle (Wassan, Wassan, Nagy, & Salhi, 2017).

If all the pickups are made after deliveries, this in practice means that the capacity C will be equal to its maximum value because Q is equal to 0. This way the vehicle has maximum performance as the total of his work is carried out. This practice is often applied to transport companies, especially those carrying out interterritorial supply operations, that is to transport materials from a material production unit to a production unit (Bortfeldt, Hahn, Männel, & Mönch, 2015).
The example above shows an example of the VRPB, the yellow arrows symbolize the deliveries, the red the arrivals, and the blue the “neutral” routes, i.e. the routes that the vehicle must perform without pick up or delivery. This problem has a wide range of applications. However, the main ones are the transport and receipt of sensitive products and materials such as liquids and food (Jairo & Chavez, 2013).

This problem, despite being particularly effective in its applications, also has some limitations. These restrictions are (Dominguez, Guimarans, Juan, & de la Nuez, 2016):

- Customers whose packages are to be picked up are those served after those to be delivered.

- The route is not just about receiving

- All deliveries and total receipts should be within the capacity limits of the vehicle.
- In order to solve the problem, the number of vehicles available to run a route should be assumed from the outset.

- It is assumed that the starting point is one. That is, all customers are served by a particular node.

- To solve the problem, the company’s vehicles are considered homogeneous, that is, they have the same characteristics as capacity.

Solving the problem by using algorithms is necessary, but due to its complexity, it results in low precision results, which do not always offer the highest degree of efficiency. The reason is many constraints that exist due to the problem, and in some cases, the solution is delayed due to the nature of the business (Laporte G., 2018). That is, an enterprise delivering small packages, it makes sense to have a larger number of delivery nodes. This leads to an increase in computational requirements and a reduction in the accuracy of the results (Laporte G., 2018).

5.5 The vehicle routing problem with pickup and delivery

The VRP with pickup and delivery is the most modern version of TSP. This is the case where pickups and deliveries are to be made simultaneously. The main advantage of this method is to reduce the number of routes to the minimum possible. However, this reduction implies increasing the capacity of the vehicles, so that the point of intersection between the capacity, performance, and cost of the service must be found (Männel & Bortfeldt, 2016).

The main advantage of this method is to reduce the number of routes to the minimum possible. However, this reduction implies increasing the capacity of the vehicles, so that the point of intersection between the capacity, performance, and cost of the service must be found (Männel & Bortfeldt, 2016). This version of the problem seems to be increasingly gaining ground in supply services, as it is based on a reduction in delivery and delivery time by all customers, while at the same time it helps to promote a green policy on the part of the company (Avci & Topaloglu, 2016).
The implementation of regular pick up and delivery at the same time has been implemented by several major companies in the sector such as FedEx and UPS. At the same time, it is used by companies that transport homogenous materials such as timber, building materials, etc. Because the only factor determining the order is the weight of the material (Belgin, Karaoglan, & Altiparmak, 2018).

![Figure 5-2 Example of the VRPPD](image)

In the case of VRPPD as data (Belgin, Karaoglan, & Altiparmak, 2018):

- The location of the central depot

- The number of vehicles that can carry out all the work (since many companies have different vehicles per job).

- The available road or transport network.

- The number of delivery nodes.
However, unlike the VRPB, in this case, the fleet may be heterogeneous, e.g., vehicles with different capacity. This can be particularly useful in densely populated urban environments, where heavy vehicles will carry the heaviest materials, while small and flexible vehicles deliver packages from customer to customer (Sitek & Wikarek, 2019). The solution to this problem can be solved in many ways, with the most popular CVRPPAD method of Sitek & Wikarek (2019), which includes the addition of a Parcel point, and allows to avoid repeating paths (Sitek & Wikarek, 2019).

5.6 The vehicle routing problem with time windows
The vehicle routing problem with time windows (VRPTW) is a generalization of the VRP. More specific, the delivery to each customer must be done within specific time limits, which are called time windows. Time windows can be divided in two categories:

1) Soft time windows, when there is no penalty if the time windows are violated
2) Hard time windows, when the prearranged time intervals cannot be ignored.

Time windows are called soft when they can be considered nonbinding for a penalty cost. They are hard when they cannot be violated, i.e., if a vehicle arrives too early at a customer, it must wait until the time window opens, and it is not allowed to arrive late. This is the case we consider here. With the use of information technology and big data, companies try to boost their efficiency and timeliness in their supply chains. In general, managers and researchers use VRPTW in order to model and optimize the operation of their supply chains. VRPTW can have many applications like in supermarkets, bank and postal deliveries, industrial refuse collection, school bus routing, security patrol service and newspaper distribution. Because it is a problem with many practical and realistic applications, there is a broader and deeper research towards at its solution. There is an important progression in the planning of heuristics and the evolution of optimal approximations (Kyritsis, Blathras, Gulliver, & Varela, 2017).
Now we will examine the exact methods for the solution of the VRPTW which are based on column generation. These date back to Desrochers, Desrosiers and Solomon (1992) who used column generation in a Dantzig-Wolfe decomposition framework and Halse (1992) who implemented a decomposition based on variable splitting (also known as Lagrangean decomposition). Later, Kohl and Madsen (1997) developed an algorithm exploiting Lagrangean relaxation. Then, Kohl, Desrosiers, Madsen, Solomon and Soumis (1999); Larsen (1999); Cook and Rich (1999) extended the previous approaches by developing Dantzig-Wolfe based decomposition algorithms involving cutting planes and/or parallel platforms.

Kallehauge (2000) suggested a hybrid algorithm based on a combination of Lagrangean relaxation and Dantzig-Wolfe decomposition. Recently, Chabrier (2005); Chabrier, Danna and Le Pape (2002); Feillet, Dejax, Gendreau and Gueguen (2004); Irnich and Villeneuve (2005); Rousseau, Gendreau and Pesant (2004) have proposed algorithms based on enhanced subproblem methodology.

Advancements in master problem approaches have been made by Danna and Le Pape (2005); Larsen (2004). In the VRPTW we have a set of vehicles, whose symbol is V, a set of customers C, and a graph Q which has different costs in each direction of its edges. Usually we presume that all vehicles are the same. There are |C|+2 vertices in the graph, the customers are represented by 1,2,…n, the depot (the starting point) is represented by 0 and n+1 is the returning depot. Also the arcs, represent the direct links among the depot and the customers and also between the customers. Moreover there are no arcs that end at node 0, there are only arcs that begin from node 0. On the contrary there are no arcs that begin from node n+1, node n+1 is only an ending point for arcs. In each arc (i,j) there is a linked cost (C_{ij}) and an associated time t_{ij} which represents the essential time for the customer i to be served (Wassan, Wassan, Nagy, & Salhi, 2017).

Every customer has a demand which is equal to d_{i} and every vehicle has a capacity which is equal to q. Also every customer has a time window whose symbolism is [a_{i}^{hi}]. Consequently the vehicle must arrive at this particular
node before hi. In the case that the vehicle arrives at the customer before the
time window opens, it has to wait until ai in order to service the customer. The
time windows for the depots are [a0, b0] which is the scheduling horizon. The
fleet is not allowed to leave the point of departure until a0, and must return no
later than b_{n+1}. It is predetermined that variables q, ai, bi, di, ci, are positive or
equal to zero integers and ti are positive integers. All these constraints and
assumptions are necessary in the column generation approach in order to
elaborate an algorithm for the shortest path with resource constraints. Moreover
the triangle inequality must be satisfied for ci and ti (Kadri, Kacem, & Labadi,
2016).

The model includes two wholes of decision variables x and s. For each arc (i,j),
where i\neq j, i\neq n+1, j\neq 0 and for each vehicle k we define x_{ijk} as:

\[ x_{ijk} = \begin{cases} 
1, & \text{if vehicle } k \text{ drives directly from vertex } i \text{ to vertex } j, \\
0, & \text{otherwise.} 
\end{cases} \]

For each node i and each vehicle k there is a variable s_{ik} which represents the
time vehicle k starts to service customer i. In the event to vehicle k does not
deliver products to customer i, the variable s_{ik}, does not represent something
and thus has no value or meaning. We conclude that a0=0 and s_{0k}=0, for all k.
6. Exact algorithms

Exact algorithms are the most widely used tool in solving optimization problems. These are algorithms that are repeated until the optimal result is achieved. Algorithms of this type require special handling, as there is the possibility if the variables are not correctly defined, that the algorithms operate indefinitely, without being able to resolve them (Erdoğan, Battarra, & Calvo, 2015). However, these algorithms are the main tools in solving optimization problems, especially in the VRP solution. Their main advantage is that the solution produced has a maximum degree of optimization. In this way, the use of algorithms of this type guarantees the performance of the system (Ortiz-Astorquiza, Contreras, & Laporte, 2019).

At the same time, these algorithms have significant "adaptability" capabilities, as they allow the user to import new data without greatly influencing the performance of the result. However, to achieve maximum performance, these algorithms require a set of complex mathematics to be solved, resulting in increased system cost (Ortiz-Astorquiza, Contreras, & Laporte, 2019).

6.1 Branch and bound algorithm

Branch and binding algorithms are a category of exact algorithms and work to produce the best solution to an optimization problem. These algorithms act as prominent branches at each node of the problem (Kadri, Kacem, & Labadi, 2016). Each branch is a possible solution, new branches are produced for each solution and rechecked. If any of the solutions at the next level is not better than the one existing, then that point stops participating in the solution (Kadri, Kacem, & Labadi, 2016). The process is repeated until there is an option, which is the solution to the problem. The following figure describes the method (Kadri, Kacem, & Labadi, 2016).
The final solution to the problem comes from bounding, i.e. the results begin convergence to arrive at a final unique solution. This synthesis is accomplished in the same way as the branch out, to begin to reduce the number of branches, the algorithm examines the existing solutions, rejecting the inefficient ones (Tanaka & Takii, 2015). The main advantage of this method is universality because the algorithm breaks the problem into a series of solvable equations, the application of the method offers the guarantee that a solution will be found in almost every possible scenario (Tanaka & Takii, 2015). However, this advantage turns into a disadvantage, as possible solutions can reach such a large number so that the solution can be very time-consuming. In other words, the use of this method is costly in terms of the computational process (Wang, Liu, & Chu, 2015).

6.2 Branch and Cut algorithm

The branch and cut method is a transformation of the branch and bound method. The difference between the two methods is in the processing of discontinuous non-linear solutions. In some cases, solutions consisting of integer numbers are omitted in the branch and bound method, thus omitting possible cost-effective solutions (Boccia, Crainic, Sforza, & Sterle, 2018).
In the branch and cut method, these solutions are processed by a subsection of the algorithm, so that they can also be considered as possible solutions without interfering with the process (Boccia, Crainic, Sforza, & Sterle, 2018). This is accomplished by inserting a cropping field, this field separating the continuous from the integral parameters, thus speeding up the process. The creation of branches between the nodes is done in the same way as in the case of BB, but in order to avoid the delays in creating branches and processing them, the process is split into branch creation and parallel processing, saving a lot of time from a computing process, with an equally high-performance index (Gelareh, Monemi, Semet, & Goncalves, 2016).

![Diagram of the Branch and Cut method](image)

*Figure 6-2 Example of the Branch and Cut method*

Depending on the problem, different types of cropping fields apply. In the case of VRP, these fields concern the implementation of different routes, which were not previously considered as alternatives. While in cases of the long distance between nodes, parameters such as parking stations or fuel infrastructure are taken into account (Gelareh, Monemi, Semet, & Goncalves, 2016). These factors, although self-explanatory, are not taken into account in
other methods, because they include interruption of the continuous movement of the vehicle. A problem that few methods have managed to solve (Dalmeijer & Spliet, 2018).

In an environment such as the US, where the distances between provincial cities are very large, these parameters are taken seriously in order to reduce the risk of an accident from a tired driver (Dalmeijer & Spliet, 2018). However, this method also has many applications within urban centers, where factors such as the large traffic problem are sometimes overlooked by some algorithms, which only recognize distances (Rodríguez-Martín, Salazar-González, & Yaman, 2016). This method is a basic choice for developers because it offers great flexibility in terms of parameters. The drawback is that more data should be given in the algorithm than in other cases, in order to obtain the best result (Rodríguez-Martín, Salazar-González, & Yaman, 2016).
7. Heuristic algorithms

Heuristic algorithms are tools for finding solutions based on speed. Unlike exact algorithms, where the solution must be absolutely accurate, heuristic attempts to find the closest solution to the absolute. In other words, the solution resulting from a heuristic algorithm is the maximum approach to the absolute solution of the problem (Koç, Bektaş, Jabali, & Laporte, 2016). The word comes from the Greek word "ευβρίσκω" (evrisko) which means I find it. This is a method that is mainly used in data retrieval systems where speed plays a much more important role than accuracy. This method is especially preferred in applications where data is constantly changing to provide the most immediate solution to the problem (Koç, Bektaş, Jabali, & Laporte, 2016).

In order to achieve speed, the method sacrifices accuracy, optimization, and integration. The "sacrifice" or "exchange" is achieved by applying a series of logical checks that answer the following questions (Abdel-Basset, Manogaran, Abdel-Fatah, & Mirjalili, 2018):

- How necessary is the best solution?
- How indispensable is the availability of alternatives?
- Can the algorithm identify a single solution?

The above queries are expressed by mathematical functions within the algorithm, but they remain active and the user can choose whether to continue with speed or precision. Many programs use heuristic algorithms as an emergency solution, that is, when new data emerges that require an immediate redefinition of solutions (Pooranian, Shojafar, Abawajy, & Abraham, 2015).

7.1 Constructive Heuristics

Algorithms belonging to this category apply a tactic "in and out". That is, they begin their operation with a zero solution, which they extend according to the solutions that arise in the individual problems, and the specified timeframe
This is a method used extensively in Internet search pages, which directly raise the first set of solutions, and then increase as time passes. If the user does not specify a time frame or other criteria (to find a solution), then the algorithm continues to produce results until the available data is exhausted (Abedinnia, Glock, & Brill, 2016).

This method also applies to the solution of VRP, generating direct paths from a theoretical "blank" point. The response of the algorithm is immediate to the point where it continues to produce results while highlighting what it has identified (Luis, Ramli, & Lin, 2016). However, this algorithm has the same disadvantage that everyone in its class possesses. The accuracy of the results depends exclusively on the number of nodes they have to study, that is, the
smaller the number, the more accurate. In applications that require the processing of a large number of nodes, this algorithm responds at high speed, but the results are not accurate (Luis, Ramli, & Lin, 2016).

7.2 Two-phase algorithms

By definition algorithms of this type operate in two independent phases. The first phase is to find all the possible solutions that can be identified on the problem. The second phase is optimizing the results, i.e., rejecting inefficient solutions (Ma & Torquato, 2018). Unlike exact methods, this method projects the results in parallel with the solution. For example, let's say the VRP needs to be resolved, the exact algorithm will proceed to search, edit, and find the solution, and then give it to the user. Instead, the heuristic two-phase algorithm will produce the results it constantly finds on the map, gradually eliminating those that are not profitable (Ma & Torquato, 2018).

![Figure 7-2 Example of the Two-Phase Heuristic Algorithm for VRP solving](image)

The usefulness of this method is that it combines the high degree of precision of exact algorithms with the speed and nematocality of the heuristic. This is the most important class of heuristic algorithms, as it produces the best results (Zuo, Huang, Zhang, Chen, & Asundi, 2016). The implementation of this type of algorithms is mainly in data retrieval applications, but also in more complex processes such as UAV cluster programming, target tracking, and more. Its speed and precision make it ideal for use in military applications as well as urban applications such as stand-alone vehicles (Zuo, Huang, Zhang, Chen, & Asundi, 2016).
7.3 Improvement heuristics

The algorithms of this category are the "unorthodox" part of the heuristic category. These are the algorithms that detect a "random" solution to the problem, and then improve it to the best possible version (Wang, Chu, & Mirjalili, 2016). This method is not widely used because of the randomness factor, as there is no guarantee that the solution resulting from the algorithm is not the worst of the possible solutions. The algorithm works by following these steps (Ortiz-Astorquiza, Contreras, & Laporte, 2019):

- Detecting a random solution to the given problem.

- View the solution

- Parallel search for alternatives based on the first.

- Repeat until there are no other possible solutions.

The problem with this method is the continuous dependence of the identified solutions on the one already identified, thus rejecting all the possible outcomes that could arise from other ramifications of the solution (Bortfeldt, Hahn, Männel, & Mönch, 2015). The major advantage of this method is saving computer resources. Due to the fact that it functions as a "opposite" method to that of exact algorithms, it requires a smaller amount of computational resources. However, this method does not apply to many real-life applications, as the results are extremely inaccurate, its use may increase costs, risk, and all other factors that try to solve problem-solving algorithms (Benavent, Landete, Salazar-González, & Tirado, 2019).
8. Clarke and Wright algorithm

Clarke and Wright's algorithm class is a subclass of heuristic algorithms. This is the method of finding a solution to a problem with ever-changing variables. This method is applied on a costly basis to transport services, mainly small items, and is the most basic method of resolving the VRP (Passos, Lourinho, Alves, & Brito, 2018). In particular, for solving VRP, the algorithm follows the following steps (Passos, Lourinho, Alves, & Brito, 2018):

1. Home Solution. In this step, the algorithm considers that each available vehicle can serve a customer.

2. Calculate the cost reduction. The algorithm performs the process of finding the smallest distance between the pairs of available nodes.

3. Classification. The algorithm classifies the possible solutions in descending order, ie from the highest cost to the lowest.

4. Combination. The algorithm starts combining the sorted solutions by two, ie 1-2, 2-3, 3-4, etc.
5. Repeat 4 until the most efficient solution is found.

The basic advantage of the algorithm is its simplicity in operation and design. Besides complex mathematical formulas, the commands that make up the process are relatively simple and understandable by both the amateur and the professional developer (Hashi, Hasan, & Zaman, 2015). For this reason, this algorithm is particularly widespread in all applications including optimizing a process. From VRP to data analysis, this algorithm is used as it offers the maximum capabilities that a system of this type could offer (Hashi, Hasan, & Zaman, 2015). CWA is divided into two categories, the parallel and sequential, each category has its advantages and disadvantages, more specifically in the following subdivisions. Now we will see some things about the mathematical formulation of the Vehicle Routing Problem and Clark and Wright algorithm.

- \( C=\{1,2,\ldots,n\} \): the set of customers (nodes).
- \( 0 \): point of departure
- \( G=(N,E) \): the graph which depicts the examined network with \( N=\{0,1,\ldots,n\} \) and \( E=\{(i,j):i,j\in N, i<j\} \)
- \( q_j \): demand of cutomer j
- \( Q=\)maximum capacity of the mean of transportation
- \( m \): the set of the vehicles that deliver the products
- \( c_{ij} \): cost of transition between nodes i and j
- \( x_{ij}= 1,0 \)

In both the parallel and sequential version, the following equations are in effect.

The objective function:

\[
\text{minimize } Z = \sum_{i\in N} \sum_{i<j} c_{ij} x_{ij} \text{ if } i\in N, i<j
\]  

(1)
Subject to:

\[ \sum_{i \in C} x_{0i} = 2m \quad , \quad i \in C \] (2)

\[ \sum_{j < i} x_{ij} + \sum_{i < j} x_{ji} = 2 \quad , \quad i \in C \] (3)

Then, we calculate the “savings” \((S_{ij})\) with the following method:

- \(c_{i0}\): the cost of transport from the node \(i\) to the point of departure

- \(c_{ij}\): the cost of transport from node \(i\) to node \(j\)

- \(c_{0j}\): the cost of transport from the point of departure to node \(j\)

**Step 1**: Computation of the savings with the following mathematical equation:

\[ S_{ij} = c_{i0} + c_{0j} + c_{ij} \quad \text{for} \quad i,j = 1,\ldots,n \quad \text{and} \quad i \neq j. \]

Then we sort the savings \((S_{ij})\) in descending order.

**Step 2**: After we created the savings list, we start to process it from the saving with the largest value. Then we choose a saving \(S_{ij}\), the connection \((i,j)\) can be added to the routes if none of the following cases is true.

**Case 1**: In the case of the parallel version of the algorithm, if node \(i\) and node \(j\) are not yet assigned to a route, then a new route is introduced which includes nodes \(i\) and \(j\). As we aforementioned this happens only in the case of the parallel version of Clark and Wright algorithm. In the sequential version we start creating the next route only when the previous route can’t include more nodes.

**Case 2**: Let’s say that we have a \((i,j)\) connection, if one of the two nodes is already included in a route then there are two sub-cases:

- **Subcase 1**: if the node which is included in the route is exterior to the route then the other node of the link is added to the route. For example, in the link \((i,j)\), if node \(i\) exists in the exterior of the route then node \(j\) is added next to the node \(i\) in the exterior of the current route. In other word the link \((i,j)\) is
added to the route. On the other hand, if one of the nodes is already included in the interior of the route, then the other node of the link is not included in the route.

**Case 3:** If both nodes of the connection \((i,j)\) are included in the routes, then we have two subcases:

**Subcase 1:** Nodes \(i\) and \(j\) are included in the interior of the routes, so no further action is taken.

**Subcase 2:** Nodes \(i\) and \(j\) are located in the exterior of different routes, then if the sum of the two aforementioned routes doesn’t exceed the capacity of the transportation truck we merge the two different routes into one.

**Step 3:** We continue to do the same until the savings list has been exhausted.

8.1 Parallel version

The parallel form of the algorithm is the most well-known and most efficient in programming. This is a repetitive process that begins with calculating the cost reduction of the process. In the case of VRP, the "cost" refers to the cost of the service (Sörensen, Arnold, & Palhazi Cuervo, 2019). The initial calculation involves calculating the distance between two nodes and the starting point. Then calculate the distance between the two nodes. The algorithm sorts the results and repeats the process for the next node pair, using as the "original" the node from which the first point had the shortest distance (Sörensen, Arnold, & Palhazi Cuervo, 2019).
The procedure is repeated and the results are sorted in descending order. After completing distance calculations, the algorithm begins to combine the paths between them in order to reduce their number. The combination is done by applying criteria that examine the distances between points, and the benefit of the combination (Hong & Kim, 2017). The process continues until a route is created, which includes all the points in the best possible way. However, the algorithm does not stop the loop, but there is the possibility of adding new routes during the process, offering the possibility of recalculating the route immediately if an extraordinary event occurs. This feature makes it ideal for in-motion runway systems, where results have to be produced directly, without reducing the accuracy of the results (Hong & Kim, 2017).

8.2 Sequential version

The sequential form of the algorithm is the least used version. This is the method of comparison-elimination of two possible solutions, until only one remains. The mode of operation of the method resembles the way in which exact algorithms result (Benavent, Landete, Salazar-González, & Tirado, 2019). This method follows the procedure followed and the parallel version with a significant difference, instead of sorting the results and then comparing them, this algorithm calculates the distances, and then rejects the inefficiencies (Bektaş, Demir, & Laporte, 2016).
The reason why the algorithm is not used is because the comparison between the distances poses the risk of discarding a possible solution, which in the later stages was the most profitable. According to research, the sequential version of the algorithm reaches only 20% of the parallel version efficiency, so it can only be used in systems that process a small number of nodes (Benavent, Landete, Salazar-González, & Tirado, 2019). This method was the first form of the CW algorithms, which then evolved into their parallel version. For this reason, the main utility of this method is its application to comparative studies and experiments to compare the various methods of VRP resolution (Halim & Yoanita, 2015).

9. Case study

9.1 Presentation of the company

XYZ company is located in Thessaloniki and is a wholesale company. XYZ distributes tobacco, beverages, alcohol, chips, chocolates, croissants, deodorants, condoms and much more. The majority of XYZ company’s customers are mini markets and kiosks. XYZ also distributes products to bakeries, internet cafes, restaurants etc. In general XYZ distributes products to approximately 1000 customers. In this thesis we will examine the optimal distribution routing for 50 customers. We will solve the Vehicle Routing Problem using the parallel and sequential version of Clark and Wright algorithm.

9.2 Historical review of the company

The first years after WW2 in Thessaloniki, the tobacco market had the following structure, some street vendors were wondering around the city and the suburbs by foot or with the use of a bicycle, selling bulk cigarettes. At 1947, the father of the current owner, left his bicycle and opened a small shop at the east side of the city. At 1975, the current owner of the company took over the company. At 1993, the company started

9.3 Picking systems inside the warehouse

Now i will make a brief presentation of the company’s operations and processes regarding the preparation and shipment of an order.

1) Firstly, the telephone operator calls the customer and asks if he/she want to order something

2) Then, the telephone operator enters the order into the Enterprise Resource Planning(ERP).

3) The ERP sends the order to the Warehouse Management System (WMS).
4) The WMS sends the customer’s order to the printers which are located near the picking area.

5) The order is composed of two parts, the first part has tobacco products like cigarettes, cigars and etc. and the second part has products like chocolates, beverages, toilet paper, croissants, liquids etc.

6) The first order paper which has the tobacco products, is printed near the shelves where the tobacco products are located and the second order paper is printed where the rest of the products are located.

Below there is a diagram regarding the aforementioned event after the receipt of an order.

![Diagram](attachment://diagram.png)

Figure 9-1 The path of the information regarding the orders

Now, regarding the picking of the tobacco products and the preparation of the order, before the tobacco products enter the transport vehicle, the company applies a system which is called pick by light. In particular, there are ninety shelves and each shelf has one type of product. There are three picking zones and each picking zone has three levels. Each level of each picking zone has ten shelves with ten product codes. Beneath the shelves there are led lights. When the order of tobacco products is printed in the tobacco products picking area then the picker of the first zone scans the order paper, then the products of the order which are located in the first zone light with an indication of the right quantity, then the picker of the first zone collects the products, turns off the led lights and puts the products with the order paper into a basket. Then the picker of the second zone takes the order paper and repeats the same process and so on so forth.

![Diagram](attachment://diagram2.png)

Figure 9-2 The process of picking tobacco products

Now regarding the picking of the rest of the products, an order paper with the order lines that does not contain tobacco products is printed from a printer near the picking area, then a picker takes the order and with a trolley goes to the warehouse’s corridors and starts collecting the products. Then the picker puts the products of the order in the loading area and another warehouse employee checks if the products are the right products with the right quantity. Then with the help of the driver the products with the tobacco products are loaded in the truck and when the right time comes are transferred to the customer. To
conclude with, except the system which is called pick by light and is used in the tobacco products picking area, another system which is used in the warehouse is named “pick by list”.

9.4 XYZ company’s WMS

Now lets say some things about the WMS the company uses and some of its applications. First of all, the WMS provides the capability of 3D representation of the warehouse. Also there is surveillance and monitoring in all operations and processes inside the warehouse, like receiving, storing, order picking, loading and order delivering to the final customers. The receiving management is responsible for the identification, marking, and receipt of the incoming final products in the warehouse and their preparation for storage. The receipt management circuit supports all the operations regarding the preparation of the receipt, the management of the products barcodes, the advanced shipping notices. Also the WMS is responsible for the inspection of the received product’s quantities, the standardization of storage units as well as the cross docking. The WMS with the use of wireless and barcode technology achieves fast and without errors receipt of products inside the warehouse’s facilities. The management of returned products and crossdocking are among the operations of the receipt management.

The WMS helps the warehouse to achieve, the best possible service for the customer, more specifically it minimizes the order preparation time and ensures that the order will have the right products. This, results in happier customers. Simultaneously the WMS helps the warehouse management to use all the available resources in an efficient manner (employees-machines). The WMS gets the necessary information about the order from the ERP, then it processes the order, it binds the stock. Furthermore the WMS can group the orders, pre-calculate the load and the package and organize the orders. The WMS targets into minimizing the expired products, thus it applies (First In First Out-First Expired First Out) methods. The WMS has the potential to support all types and methods of order picking like pick by order, pick by group of customers, pick by “waves” of loading, pick by route, pick by zone and pick by batch.

The WMS can collect the appropriate products in every measurement unit (e.g. pallet, box, package, piece). The WMS issues automatically and forwards the orders to the wireless handheld devices of the pickers. Then the WMS takes into account the proximity of the products to the picker’s starting point and finds the order by which the picker must collect the products. In this way, the order picking time and the routes inside the warehouse are minimized. Moreover the WMS does the replenishment of picking positions based on the minimum quantity concept. For example if the stock in one picking position goes under a certain minimum limit then the driver of the clark gets a notification that this particular picking position needs refill. Finally the WMS
supports all types of storage systems (e.g. back-to-back, Drive-In, Life Storage, Push Back) also the user can navigate into the facility of the warehouse using virtual reality. Furthermore, after the tobacco products are collected from pick-by-light, then a weighing check is conducted in order for the system to find out if the right products have been placed into the order.

The WMS supports all types of stocktaking like overall stocktaking, circular stocktaking, stocktaking in phases, stocktaking per product, stocktaking per geographical zone or group of shelves. The stocktaking is conducted with the use of wireless terminals which ensure the precision of the inventory’s stock. The WMS coordinate and optimize all the operations inside the warehouse with the creation of work orders which are sent in real time to the wireless terminals of forklift trucks and pickers. Also the WMS enables the warehouse managers to monitor the performance indexes of the warehouse’s resources, thus they are able to conduct right budgets and reviews of all operations and tasks. The WMS records all orders and procedures, and then uses these data in order to detect mistakes and extract statistical data, which help the administration to take the right decisions. Moreover the WMS has the capability to communicate and connect in real time with external software and hardware systems, like ERP, wireless networks, scales, sorters and pick-by-light systems. Moreover a wide variety of subsystems such as add-ons or modules can be integrated in the basic core of the WMS. The vertebrate structure of the WMS allows mama products to choose and use some standardized subsystems in order to implement some specialized solutions which fully cover the company’s needs. The WMS has a web application which uses the internet in order to exchange data and information which has to do with the supply chain. For example the customers can insert an order with the use of this web application or they can update the information regarding the “expected folders” with the use of internet and extranet. Also with the help of the web application authorized users can keep track of the orders, the stock, the returns and etc.

Moreover the WMS has an add-in which schedules the deliveries and the routes. This add-in plans the routes fast, taking into account all the possible restrictions (like payload and volume), customer delivery days, order priority, maximum number of itineraries, maximum number of delivery points per route, available vehicles and areas which are served by the trucks. Also the WMS has another add-in which has to do with fleet tracking, this is an application which has to do fleet management. This application is based in GIS, GPS και GSM (SMS & GPRS) technologies and gives the capability to all users to monitor in real time, periodically or in the form of report the movements of the fleet through a user-friendly and practical environment with digital maps (GIS). Moreover this fleet tracking add in offers real time information regarding the routes (beginning- end), statistics about the amount and the duration of the
vehicle’s stops, analytical report of the vehicle’s route, report of on-time-off
time deliveries etc.

Also the WMS has the capability to audit that in every retail shop the products
are delivered with the right quantities without shortages or surplus. The system
monitors the delivery of the orders in real time. During the delivery of the
order, the operator scans the barcode of the order and then scans the delivered
products. By this way the system can check that the products that are included
in the order are delivered. When the whole quantity of the order is delivered
then the receiver registers his/hers comments and signs digitally. Also the
company’s back office gets informed in real time about the progress of the
deliveries and the fluctuation of the lead time. In addition the WMS has the
capability of sales forecasting and inventory planning. To be more specific, the
WMS has two functions:

Based on data from the past, like sales, the WMS can calculate the seasonality
and trend of sales, it can compare the seasonality of the item with the
seasonality of the statistical group where the item belongs and carries out the
necessary corrections. Then the system runs automatically the sales forecasting
process for the next periods (days, weeks, months) depending on the needs of
the company. The forecasting can be done for one item, a group of items or for
every item in the warehouse according to the desired information. This process
lasts few minutes and can be modified by the user, according to extra
information which originates from the market or promotional actions designed
by the marketing department.

Also another function of the WMS can calculate automatically the schedule of
stock and procurement, for dynamic time zone. This schedule can be done for
one type of products, for a group of products, or for every item inside the
warehouse. The user can take an export of the results in the form of an excel
spreadsheet. For example the user can see results regarding a specific item,
regarding a time period, anticipated receipts, suggested orders towards the
suppliers, suggested quantity of stock, suggested stock per time period, for
example in summer months the stock of beverages and bottled water must be
higher than usual. Also the WMS can guide the employee who is responsible
for procurement to form whole loads towards the suppliers in order to the
transportation cost to be lower. Also the user, with the help of the WMS, can
predict future problems, like stockouts, lack of storage space, lack of cash flow,
also the user can calculate future workload inside the warehouse and calculate
the premiums based on the predicted stock.

Regarding the stock management of the retail shops, to be more specific the
customers of the company, if the owner of the retail shop wants to cooperate
with its supplier, in our case XYZ company, then the WMS can give guidance
to the retail shop regarding its stock replenishment. To be more specific, the
WMS processes past sales data and based on the lead time, the weekly replenishment rate, the desired level of customer service and other parameters, like promotional actions, sends recommendations to the owner of the retail shop regarding its orders per product and needs. For example if the company owns retail shops then the WMS, based on sales of the past, can calculate the optimal order for each product, then this order can be printed in the warehouse and be prepared by the pickers without any human intervention from the employees of the company’s back office.

The WMS integrates algorithms and forecasting processes, which adjust automatically to the needs and seasonal behaviors of the company’s products (self adaptive forecasting methods), without the need of human intervention and parameter adjustment from the users. The sales of past periods are inserted into the WMS, then the WMS through the interface application transfers all these data to the ERP. These data have to do with quantitative (units) sales per item, day, week or month depending on the prediction group, in which the item belongs. The user has the capability to intervene and correct the historical data, which were caused by occasional situations, or are the result of intense activities. The inventory check and the scheduling of purchases and “production” orders is based on the Time Phased Planning, which is applied with success in MRP and MRP 2 systems. The user can choose, with some criteria, an item or a group of items and creates a schedule. For the chosen schedule the user defines a time horizon, in which he/she desires to execute the stock analysis and to carry out the buying commands. The WMS, without any human intervention, creates automatically the schedule of stocks and purchases. An excel spreadsheet can be used for the display of the results.

The prediction algorithm takes into account the seasonality and trend of the sales and removes the unnecessary data that are caused by random factors. The algorithm compares the seasonality of the item with the seasonality of the statistical group, in which the item belongs and conducts the necessary corrections. The result of this process is the sales forecasting for the next time periods, for example the next 28 days, the next 24 weeks or the next 12 months. Also the user can modify the predicted sales, according to extra information provided by the market or based on future promo actions, which are designed by the marketing department of the company. This correction can be done for one item or for a group of items, with rate fluctuation or with modification of the absolute values. The process of analog conversion (prorating) is conducted automatically by the WMS and converts the sales forecasting of the items, from the period unit of the prediction, to the time unit of the procurement schedule. The time unit of procurement is defined in the parametric data of the item. For example, if the prediction of an item takes place in a month level, while the observation and the scheduling of the item’s procurement is conducted in a
weekly level, then the WMS recalculates the weekly sales, based on the monthly predictions which were calculated by the prediction algorithm.

The sheet of stock time-scheduling shows the overall view of stock and purchases over time. By using this sheet the user is able to display the quantities in other measurement units (for example: value, volume, weight, means of transportation), to convert the quantities from the predefined time unit in other time unit (for example from week to month), to search information about specific items, group of items, or all the items with their aggregated quantities expressed in a variety of stock measurement units (for example value, weight, volume, pallets), also the user has the possibility to change the quantities, so the WMS recalculates automatically the procurement scheduling. With the scheduling sheet the user has the capability to examine results per item and per time period, amongst others, information like: predicted demand per period, expected arrivals, suggested orders towards the suppliers and safety stock per period. The WMS creates the items and quantities that are included in the dispatch notes, towards the suppliers. Moreover the WMS indicates the urgent orders that should be made in order to avoid future stock outs.

Based on the stock scheduling sheet, the user can create a receipt plan. This receipt plan gives the opportunity to the user to obtain an overall view regarding the progress of the receipts. With the help of the WMS the user can observe the progress of stock over time, by displaying to the user the future evolution of stock. Consequently, the user, based on the provided information from the WMS, can detect future problems, such as insufficiency of the storage space and problems regarding the cash flow. Also the user can obtain a prediction regarding the future workload in the warehouse and calculate the premiums based on the quantity of future stocks. The WMS provides the capability of sensitivity analysis. Sensitivity analysis helps the user to examine the consequences of emergency events, such as an unusually large order on behalf of a customer which is going to cause a stock-out, or, for example, the delay of an import which is going to cause problems in the operation of the company.

All these simulations and sensitivity analysis are done with a very simple and user-friendly way, with the use spreadsheet interfaces of the WMS. The user has the capability to examine and test multiple scenarios. Moreover the WMS provides the capability to its users to form orders which fill the whole capacity of the truck in orders to minimize the transportation cost with the use of economies of scale. These full loads can be transported with the use of trucks, containers, carriages etc. In order to full loads to be achieved, the employee whose task is to make the orders towards the suppliers, redirects future or past orders and fluctuates the quantities of the orders. Moreover the WMS provides the capability of budget creation regarding the sales. This budget can be created with two ways, firstly, the system executes the process of statistical forecasting
and transforms these sales forecasts into budget, where the user can modify these calculated quantities. The other way is the following: the user with the help of an interface program, inserts into the wms the annual quantities, then the system divides these quantities to each month, based on the item’s seasonality indexes.

The WMS has the capability to conduct ABC analysis for every item inside the warehouse, or for a group of items, or for a specific time period. This analysis can be done based on value or quantity, with the measurement unit that is chosen by the user, like piece, box, pallet etc. Also the WMS provides the capability to the user to create a new item or to delete an item from the database. The recently created item, inherits the historical data of the deleted item, consequently the sales forecasting is capable. The wms provides the capability of sales forecasting with a small amount of historical data. If the data are not enough, then the system uses data from the sales budget. Additionally, if the item has seasonal behavior and the available historical data are inadequate for the calculation of seasonal indexes, then the wms uses the overall seasonality of the group, in which the item belongs.

The wms has a complementary module, which is responsible for the stock replenishment of the company’s retail shops. In contrast with the replenishment system which is used in the warehouses, which is based in the MRP model, the replenishment of the retail shops is automated based on a model whose name is “Optimal Point of Reordering” of the classic theory of stock management, which is estimated from the wms. This approach is effective, because the replenishment times are small, in contrast with the replenishment times of the warehouses. The process of stock-replenishment in retail stores is fully automated, because the wms issues automatically the replenishment orders of the retail shops, based on the replenishment schedule. The user of the wms can define for each retail shop, the lead time, the weekly frequency of replenishment and the desirable level of customer service. In case that, the company carries out promotional actions (offers, leaflets, discount coupons, etc), and the sales are expected to be higher, the user activates the alteration of the basic parameters, for the period when the promotion is active. After the expire date of the promotion the wms takes control and restores automatically the parameters into their optimal values.

To sum up, the wms provides the following benefits to the company,

1) Decrease of stock and storage cost  
2) Better customer service and decrease of shortages in the customer’s orders  
3) Decrease of the required storage space  
4) Decrease of the transportation cost with the creation of whole loads  
5) Decrease of cost due to the disdain of products
6) Decrease of workload in the department of procurement
7) Decrease of workload inside the warehouse
8) Improvement in the cooperation with the suppliers and decrease of the lead time
9) In time diagnosis of future problematic situations
10) Improvement of cooperation between the department of the company (sales, marketing, procurement, warehouse)
11) Capability of investigation of extreme situations (what if analysis)
12) Appropriate scheduling in the operations of the supply chain

9.4 The application of pick-by-light systems in XYZ company

In today’s competitive environment of supply chain management, the success of a company is based on the in-time and without mistakes delivery of an order. This cannot be achieved only by using traditional systems of picking or batch processes. The logistic companies which are able to prepare and deliver orders without mistakes, with relatively low cost and are able to satisfy the increasing needs of their customers, can acquire a competitive advantage in their sector. The prerequisite of the right level of automation in a distribution center is the clear understanding and the definition of:

1) the company’s activities
2) the characteristics of the products
3) the frequency and the volume of the executed orders
4) the labor costs inside the warehouse

The pick-by-light technology is the result of the combination of cutting edge technologies like wireless communication (RF), microcomputers, microelectronics, barcode technology and software technology, in one uniform, effective and integrated system. The pick by light technology makes good use of the data which have to do with the orders that must be executed and transfer these data to the picking zones, without using picking lists or check lists. Moreover the employees in the order picking zone, by having both their hands free, they are able to collect the orders without doing time-consuming and needless actions, like reading from a list, searching for the location inside the warehouse where the product is placed, checking of the list etc. The result of the above is the decrease of the order picking time and the elimination of the mistakes during the picking. The pick-by-light technology which is used by XYZ company abolishes the reading of picking lists from the pickers and the data entries to the system, practically eliminating the mistakes during the order picking.

In general, all the different products the a company distributes doesn’t have the same traffic speed. There some product codes with big traffic speed, which have a large volume of sales and other product codes with low traffic speed
which have low sales. Depending on the market in which a company does business, (production, wholesaler, retail), there is a number of curves which define the ratio between product codes with large traffic speed and product codes with low traffic speed. Due to the fact that all product codes doesn’t have the same traffic speed, we apply different methods of order picking for each category of products. The pick-by-light systems are usually preferred for the fast moving products. Now, if we do a brief analysis of the time that an employee spends in order to collect and prepare an order, we can get the following results:

1) Handling of lists: 10%
2) Correction of mistakes during the order collection: 8%
3) Picking, placement: 16%
4) Packaging: 3%
5) Checking of the order: 13%
6) Detection of the next product of the picking list and transition to this position.

With the use of pick-by-light systems, we can notice a decrease or elimination in the needed time for processing the picking list, in the needed time for the correction of the mistakes during the picking process and in the necessary time for detecting the right picking position. With the use of pick-by-light systems we can notice an elimination of the order picking and preparation process. Before the installation of pick by light systems in the picking process of the tobacco products, the warehouse managers of the XYZ company noticed the following allocation in the mistakes which were made:

1) Mistakes in the counting of the products: 2%
2) Mistakes during the reading of the picking list: 23%
3) The pickers choose a product that is not included in the order: 30%
4) The pickers forget to put a product which is included in the order: 45%

After the installation of pick-by-light in the XYZ company, the warehouse’s productivity was increased by 300%. The mistakes during the picking process were reduced up to 95%, while the depreciation time of the investment was less than twelve months. The pick by light system is comprised by micro-computers whose size is equal to a chocolate. These micro-computers are placed in the picking positions of the tobacco products. These microcomputers posses a light bulb, a bright digital indication, as well as user control buttons. Consequently the pick by light system is able to connect in real time with the company’s information system.

Regarding the picking process, as we aforementioned, the picker scans the order, then the light bulbs which lie in front of the picking positions of the
codes that are included in the order light, the indication on the light bulb is equal with the desired quantity. In the case that the picker notices a shortage of this code’s stock, then he/she can lower the picking quantity by pressing the function button. For each picking position from which the collection of the products were made, the light indication turns off. Then when the last line of the order has been collected the completion display lights up and a sound is made which informs the picker that the order has been collected.

In the case of XYZ company, because the number of fast-moving codes is large, the picking area of tobacco products is divided into three zones, where each zone has one picker. So the picker in each zone collects the products that exist in his/her zone and moves forward the order to the next picker in the next zone. This process continues until the order has passed from all the system’s zones.

9.5 XYZ company’s ERP

XYZ company as every modern and competitive company uses an ERP. With the use of ERP systems, XYZ company is able to improve its flexibility, to insert new systems and operations faster, and of course to increase its productivity, so that the company can remain competitive. XYZ company uses the ERP in order to make the operations and processes inside the company more efficient and to make the communication amongst the various company’s departments easier.

In general, the ERP covers the following four main operational sectors inside the company:

1) Economic management: general accounting, analytic accounting, standing orders, securities, cash flow, reports creation etc.
2) Commercial management and distribution: warehouse, sales, procurement, credit turnovers, contracts etc.
3) Retail(POS-Touch Screen)

The ERP makes the receiving of the orders and the delivery of the orders to the customers. In particular:

1) It improves the scheduling operations
2) Synchronizes the procurement with the demand
3) Efficiently manages the inventory
4) Manages the traceability of the products

The ERP is built in a way which accepts modules, depending on the company’s needs. Also the ERP is able to integrate with a variety of commercial software products that are released in the market. Also one of the advantages of the ERP is that it is flexible, it is user-friendly and it has client/server function. Moreover the ERP can provide to its users wide information by collecting data from various sources inside the company. This capability can help the company’s managers to make more well-calculated and precise decisions. The
result of the aforementioned is the increase of the business performance due to the more rational management and utilization of data and resources. The ERP has some business intelligence tools, consequently every decision made in a strategic or operational level, is supported always by the right information. In this way, the organization can take the right decisions and simultaneously, control every aspect of the company’s operation.

9.6 XYZ company’s mobile ERP

XYZ company uses a mobile version erp in the sales person’s tablets. This mobile ERP helps the company to synchronize the sales team, service and marketing, and to schedule their activities inside and outside of the company by optimize their communication and interaction.

Part B: The application of the algorithm

10. The application of Clark and Wright algorithm in XYZ company

Due to time constraints the application of the algorithm was made only with forty nine customers of XYZ company. The company’s customers didn’t wanted their personal data to be written in this thesis, so I will not display their brand name neither their address. In the following figure we can see the distance matrix.

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>48</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>49</td>
<td>48</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>50</td>
<td>49</td>
<td>48</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>51</td>
<td>50</td>
<td>49</td>
<td>48</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>52</td>
<td>51</td>
<td>50</td>
<td>49</td>
<td>48</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>53</td>
<td>52</td>
<td>51</td>
<td>50</td>
<td>49</td>
<td>48</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>54</td>
<td>53</td>
<td>52</td>
<td>51</td>
<td>50</td>
<td>49</td>
<td>48</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>55</td>
<td>54</td>
<td>53</td>
<td>52</td>
<td>51</td>
<td>50</td>
<td>49</td>
<td>48</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>56</td>
<td>55</td>
<td>54</td>
<td>53</td>
<td>52</td>
<td>51</td>
<td>50</td>
<td>49</td>
<td>48</td>
<td>0</td>
</tr>
</tbody>
</table>

Figure 10-1 The distance matrix between each retail store

In each row and in each column there is one retail store, whose symbol is a number from increasing numbering. The depot’s symbol is zero while the symbols of retail stores (customers) are integer numbers from 1 to 49.

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Travel time in minutes</td>
<td>31</td>
<td>4</td>
<td></td>
<td>2</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>1</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>Distance in meters</td>
<td>24700</td>
<td>1100</td>
<td></td>
<td>400</td>
<td>850</td>
<td>850</td>
<td>1000</td>
<td>230</td>
<td>1200</td>
<td>1000</td>
</tr>
<tr>
<td>Weighted mean of transportation cost</td>
<td>28,0476</td>
<td>3,3</td>
<td></td>
<td>1,2</td>
<td>2,55</td>
<td>2,55</td>
<td>3</td>
<td>0,69</td>
<td>3,6</td>
<td>3</td>
</tr>
</tbody>
</table>

Figure 10-2 The displayed distances between retail store number two and other retail stores
The distance information for each retail store is presented in three rows. In the first row of each retail store we can see the travel time in minutes between retail store two and the rest of the retail stores. In the second row, we can see the distance between each retail store presented in meters. In the third row we can see the weighted distance between the retail store whose id-number is presented in the left side of the picture and the rest of the retail stores. In the fourth row we can see the weighted mean of the travel cost between the retail store whose id-number is presented in the left side of the picture and the rest of the retail stores.

For example the weighted mean cost between retail store x and retail store y is calculated using the following formula:

Weighted mean distance = 0.7*Travel Time + 0.3*Distance in meters

By using this formula and the programming language Visual Basic for Applications the cost matrix is calculated. To be more specific the code which was used in order to calculate the weighted distance is the following.

```
Sub data_final()
  Dim x, y As Integer
  For x = 6 To 202
    For y = 3 To 52
      If (Cells(x - 1, y) <> "") Then
        Cells(x, y) = Cells(x - 1, y) * 0.7 + Cells(x - 3, y) * 0.3
      End If
    Next
  Next
End Sub
```

Figure 10-3 The VBA code which was used to calculate the weighted distance
The cost matrix of all retail stores, the cost matrix displays the cost of transition between node x and node y.

![Figure 10-4](image)

The weighted distance between retail store, whose id number is 10, and other retail stores.

For example the weighted distance between retail store 10 and retail store 1 is equal to 28.3949, the distance between retail store 1 to retail store 5 is equal to 2.321875, the distance between retail store 1 and retail store 9 is equal 2.2125 and so on.

Then i calculated the savings matrix between each retail store. To be more specific,

![Figure 10-5](image)

The savings matrix is calculated using the following way:

\[ \text{Savings} = \text{Cost} - \text{Weighted Distance} \]

Let's say that we have the following distances in the cost matrix

![Figure 10-6](image)

The savings matrix is calculated using the following way:

\[ \text{Savings} = \text{Cost} - \text{Weighted Distance} \]

For example the “saving” value between node 1 and node 2 means that the company will spare that amount if includes node 1 and node 2 in the same route. To be more specific:

The initial solution that the Clark and Wright algorithm gives is that every single node in our customer base is served by its own route and vehicle. Of course this initial solution is not efficient at all, but the algorithm based on this initial solution and after a number of repetitions provides us a more efficient solution. So in order to calculate
the saving between node 1 and node 2 we follow the following mathematical formula:

\[
\text{Saving between node 1 and node 2} = \text{weighted distance between node 1 and the depot} + \text{weighted distance between node 2 and the depot} - \text{weighted distance between node 1 and node 2}
\]

Consequently:

the saving of node 1 and node 2 is equal to \(26.559 + 29.1718 - 3.90625 = 51.82455\)

the saving of node 1 and node 3 is equal to \(26.559 + 28.7128 - 2.75938 = 52.51242\)

the saving of node 2 and node 1 is equal to \(28.9333 + 25.8974 - 3.51 = 51.3207\)

the saving of node 2 and node 3 is equal to \(28.9333 + 28.7128 - 1.44 = 56.2061\)

and using the same formula i calculated the rest of the savings matrix.

The VBA code which was used for the calculation of the savings matrix is the following:

```
Sub savings_matrix()
    Dim x, y, i, z As Integer
    i = 2
    For x = 3 To 51
        z = 2
        For y = 3 To x
            If (x <> y) Then
                Worksheets("savings_matrix").Cells(i, z) = Worksheets("cost_matrix").Cells(i, y) + Worksheets("cost_matrix").Cells(x, y) - Worksheets("cost_matrix").Cells(x, y)
            End If
            z = z + 1
        Next
        i = i + 1
    Next
End Sub
```

Figure 10-8 The VBA code which was used for the calculation of the savings matrix

But this form of the savings matrix is not final. As you can see in the above matrix, the matrix has values above the diagonal and under the diagonal. This is not right because each pair of nodes has two saving values and the clark and wright algorithm can’t work with two “savings” values. For example:

1)The saving between node 3 and 7 is equal to 53.15 while the saving between nodes 7 and 3 is equal to 56.28269

2)The saving between node 6 and 11 is equal to 59.49174 while the saving between node 11 and 6 is equal to 55.14487.

3)The savings between node 11 and node 12 is equal to 51.01923 and the savings between node 12 and 11 are equal to 51.97179
So I put above the diagonal the minimum value of the two pairs and I deleted the values below the diagonal, for instance:

3->7: savings value=53.15
7->3: savings value=56.28269

So in this case we put above the diagonal between the node 3 and 7 the minimum value from both. In the same way, for the nodes 6 and 11 we have:

6->11: savings value= 59.49174
11->6: savings value= 55.14487

So we put the value 55.14487 above the diagonal which is the minimum and for nodes 11 and 12 we have

11->12: savings value=51.01923
12->11: savings value=51.97179

So we choose the minimum value which is 51.01923.

Consequently the final version of the savings matrix is the following.

![Figure 10-9 The final form of the savings matrix.](image)

Then we sort the values of the savings matrix in a decreasing order and we have the following matrix.
The values of the savings matrix sorted in decreasing order.

For the solution of the Vehicle Routing Problem both parallel and sequential versions were used. Because XYZ company has a big variety of different products, big boxes are used for the arrangement and distribution of the products. Each truck has a maximum capacity of 100 boxes, consequently the variable Q is equal to 100. So the average demand of each retail store is measured in boxes, the following picture displays the average demand of each retail store.

<table>
<thead>
<tr>
<th>ID of retail store</th>
<th>Demand</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>14</td>
</tr>
<tr>
<td>2</td>
<td>13</td>
</tr>
<tr>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>4</td>
<td>11</td>
</tr>
<tr>
<td>5</td>
<td>12</td>
</tr>
<tr>
<td>6</td>
<td>13</td>
</tr>
<tr>
<td>7</td>
<td>15</td>
</tr>
<tr>
<td>8</td>
<td>14</td>
</tr>
<tr>
<td>9</td>
<td>12</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>12</td>
<td>12</td>
</tr>
<tr>
<td>13</td>
<td>14</td>
</tr>
<tr>
<td>14</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td></td>
</tr>
</tbody>
</table>

Figure 10-11 The average daily demand, in boxes, of retail stores 1 to 15

<table>
<thead>
<tr>
<th>ID of retail store</th>
<th>Demand</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>15</td>
</tr>
<tr>
<td>17</td>
<td>13</td>
</tr>
<tr>
<td>18</td>
<td>11</td>
</tr>
<tr>
<td>19</td>
<td>15</td>
</tr>
<tr>
<td>20</td>
<td>13</td>
</tr>
<tr>
<td>21</td>
<td>11</td>
</tr>
<tr>
<td>22</td>
<td>10</td>
</tr>
<tr>
<td>23</td>
<td>14</td>
</tr>
<tr>
<td>24</td>
<td>12</td>
</tr>
<tr>
<td>25</td>
<td>15</td>
</tr>
<tr>
<td>26</td>
<td>13</td>
</tr>
<tr>
<td>27</td>
<td>12</td>
</tr>
<tr>
<td>28</td>
<td>10</td>
</tr>
<tr>
<td>29</td>
<td>11</td>
</tr>
<tr>
<td>30</td>
<td>12</td>
</tr>
<tr>
<td>31</td>
<td>14</td>
</tr>
</tbody>
</table>

Figure 10-12 The average daily demand, in boxes, of retail stores 16 to 31

<table>
<thead>
<tr>
<th>ID of retail store</th>
<th>Demand</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>14</td>
</tr>
<tr>
<td>33</td>
<td>12</td>
</tr>
<tr>
<td>34</td>
<td>13</td>
</tr>
<tr>
<td>35</td>
<td>15</td>
</tr>
<tr>
<td>36</td>
<td>12</td>
</tr>
<tr>
<td>37</td>
<td>15</td>
</tr>
<tr>
<td>38</td>
<td>11</td>
</tr>
<tr>
<td>39</td>
<td>14</td>
</tr>
<tr>
<td>40</td>
<td>15</td>
</tr>
<tr>
<td>41</td>
<td>12</td>
</tr>
<tr>
<td>42</td>
<td>11</td>
</tr>
<tr>
<td>43</td>
<td>15</td>
</tr>
<tr>
<td>44</td>
<td>10</td>
</tr>
<tr>
<td>45</td>
<td>11</td>
</tr>
<tr>
<td>46</td>
<td>12</td>
</tr>
<tr>
<td>47</td>
<td>11</td>
</tr>
<tr>
<td>48</td>
<td>10</td>
</tr>
<tr>
<td>49</td>
<td></td>
</tr>
</tbody>
</table>
10.1 Clark and Wright algorithm (Parallel Version)

Now after we sorted the savings in decreasing order we can start to execute the parallel version of clark and wright algorithm. As we mentioned before, we start from the savings list.

1. saving: 59,571       nodes: 6-21

We create the first route

1st route: 0-6-21-0       Route’s demand=24 boxes       Route’s cost: 63,78

2. saving: 58,436       nodes: 6-29

Node 6 is an external node in the first route, the current demand of the route plus the demand of node 29 doesn’t surpass the maximum capacity of the truck which is equal to 100 boxes of products. Consequently we can add node 29 to the first route.

1st route: 0-29-6-21-0       Route’s demand=35 boxes       Route’s cost: 68,08

3. saving: 57,948       nodes: 6-25

Nothing happens because node 6, which already belongs to the 1st route, is an internal node to the route, so node 25 can not be added to the route.

1st route: 0-29-6-21-0       Route’s demand=35 boxes       Route’s cost: 68,08

4. saving: 57,75558313      nodes: 2-20

Nodes 2 and 20 doesn’t already exist in a route, so: since we execute the parallel version of Clark and Wright algorithm, we create another route with these two nodes

1st route: 0-29-6-21-0       Route’s demand= 35 boxes       Route’s cost: 68,08
2nd route: 0-2-20-0       Route’s demand= 26 boxes       Route’s cost: 60,03

5. saving: 57,74358974      nodes: 11-21

1st route: 0-29-6-21-11-0       Route’s demand= 45 boxes       Route’s cost: 66,01
2nd route: 0-2-20-0       Route’s demand= 26 boxes       Route’s cost: 60,03

6. saving: 57,70       nodes: 25-29

1st route: 0-25-29-6-21-11-0       Route’s demand= 60 boxes       Route’s cost: 67,32
2nd route: 0-2-20-0       Route’s demand= 26 boxes       Route’s cost: 60,03

7. saving: 57,47461538      nodes: 21-29
1st route: 0-25-29-6-21-11-0  
Route’s demand= 60 boxes  Route’s cost: 67,32  
2nd route: 0-2-20-0  Route’s demand= 26 boxes  Route’s cost: 60,03  

8. saving: 57,45820513  nodes: 24-25  
1st route: 0-24-25-29-6-21-11-0  
Route’s demand= 72 boxes  Route’s cost: 67,54  
2nd route: 0-2-20-0  Route’s demand= 26 boxes  Route’s cost: 60,03  

9. saving: 57,44219114  nodes: 21-27  
This case is the same with the case in the saving no: 3, so nothing happens.  
1st route: 0-24-25-29-6-21-11-0  
Route’s demand= 72 boxes  Route’s cost: 67,54  
2nd route: 0-2-20-0  Route’s demand= 26 boxes  Route’s cost: 60,03  

10. saving: 57,43361823  nodes: 5-29  
1st route: 0-24-25-29-6-21-11-0  
Route’s demand= 72 boxes  Route’s cost: 67,54  
2nd route: 0-2-20-0  Route’s demand= 26 boxes  Route’s cost: 60,03  

11. saving: 57,41367521  nodes: 6-24  
1st route: 0-24-25-29-6-21-11-0  
Route’s demand= 72 boxes  Route’s cost: 67,54  
2nd route: 0-2-20-0  Route’s demand= 26 boxes  Route’s cost: 60,03  

12. saving: 57,3063689  nodes: 20-29  
1st route: 0-24-25-29-6-21-11-0  
Route’s demand= 72 boxes  Route’s cost: 67,54  
2nd route: 0-2-20-0  Route’s demand= 26 boxes  Route’s cost: 60,03  

13. saving: 57,28148148  nodes: 24-29  
1st route: 0-24-25-29-6-21-11-0  
Route’s demand= 72 boxes  Route’s cost: 67,54  
2nd route: 0-2-20-0  Route’s demand= 26 boxes  Route’s cost: 60,03  

14. saving: 57,23076923  nodes: 5-6  
1st route: 0-24-25-29-6-21-11-0  
Route’s demand= 72 boxes  Route’s cost: 67,54  

63
<table>
<thead>
<tr>
<th></th>
<th>2nd route: 0-2-20-0</th>
<th>Route’s demand= 26 boxes</th>
<th>Route’s cost: 60,03</th>
</tr>
</thead>
<tbody>
<tr>
<td>15. saving:</td>
<td>56,9114984</td>
<td>nodes: 6-28</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1st route: 0-24-25-29-6-21-11-0</td>
<td>Route’s demand= 72 boxes</td>
<td>Route’s cost: 67,54</td>
</tr>
<tr>
<td></td>
<td>2nd route: 0-2-20-0</td>
<td>Route’s demand= 26 boxes</td>
<td>Route’s cost: 60,03</td>
</tr>
<tr>
<td>16. saving:</td>
<td>56,89564103</td>
<td>nodes: 11-29</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1st route: 0-24-25-29-6-21-11-0</td>
<td>Route’s demand= 72 boxes</td>
<td>Route’s cost: 67,54</td>
</tr>
<tr>
<td></td>
<td>2nd route: 0-2-20-0</td>
<td>Route’s demand= 26 boxes</td>
<td>Route’s cost: 60,03</td>
</tr>
<tr>
<td></td>
<td>3rd route: 0-7-10-0</td>
<td>Route’s demand= 30 boxes</td>
<td>Route’s cost: 56,81</td>
</tr>
<tr>
<td>17. saving:</td>
<td>56,80769231</td>
<td>nodes: 7-10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1st route: 0-24-25-29-6-21-11-0</td>
<td>Route’s demand= 72 boxes</td>
<td>Route’s cost: 67,54</td>
</tr>
<tr>
<td></td>
<td>2nd route: 0-2-20-0</td>
<td>Route’s demand= 26 boxes</td>
<td>Route’s cost: 60,03</td>
</tr>
<tr>
<td></td>
<td>3rd route: 0-7-10-0</td>
<td>Route’s demand= 30 boxes</td>
<td>Route’s cost: 56,81</td>
</tr>
<tr>
<td>18. saving:</td>
<td>56,78317308</td>
<td>nodes: 6-14</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1st route: 0-24-25-29-6-21-11-0</td>
<td>Route’s demand= 72 boxes</td>
<td>Route’s cost: 67,54</td>
</tr>
<tr>
<td></td>
<td>2nd route: 0-2-20-0</td>
<td>Route’s demand= 26 boxes</td>
<td>Route’s cost: 60,03</td>
</tr>
<tr>
<td></td>
<td>3rd route: 0-7-10-0</td>
<td>Route’s demand= 30 boxes</td>
<td>Route’s cost: 56,81</td>
</tr>
<tr>
<td>19. saving:</td>
<td>56,76622458</td>
<td>nodes: 6-16</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1st route: 0-24-25-29-6-21-11-0</td>
<td>Route’s demand= 72 boxes</td>
<td>Route’s cost: 67,54</td>
</tr>
<tr>
<td></td>
<td>2nd route: 0-2-20-0</td>
<td>Route’s demand= 26 boxes</td>
<td>Route’s cost: 60,03</td>
</tr>
<tr>
<td></td>
<td>3rd route: 0-7-10-0</td>
<td>Route’s demand= 30 boxes</td>
<td>Route’s cost: 56,81</td>
</tr>
<tr>
<td>20. saving:</td>
<td>56,76354167</td>
<td>nodes: 2-10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1st route: 0-24-25-29-6-21-11-0</td>
<td>Route’s demand= 72 boxes</td>
<td>Route’s cost: 67,54</td>
</tr>
<tr>
<td></td>
<td>2nd route: 0-7-10-2-20-0</td>
<td>Route’s demand= 56 boxes</td>
<td>Route’s cost: 60,07</td>
</tr>
<tr>
<td>21. saving:</td>
<td>56,74461538</td>
<td>nodes: 2-21</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1st route: 0-24-25-29-6-21-11-0</td>
<td>Route’s demand= 72 boxes</td>
<td>Route’s cost: 67,54</td>
</tr>
<tr>
<td></td>
<td>2nd route: 0-7-10-2-20-0</td>
<td>Route’s demand= 56 boxes</td>
<td>Route’s cost: 60,07</td>
</tr>
</tbody>
</table>
22. saving: 56,743,108,971
   nodes: 14-20
   1st route: 0-24-25-29-6-21-11-0
   Route’s demand= 72 boxes   Route’s cost: 67,54
   2nd route: 0-7-10-2-20-14-0
   Route’s demand= 68 boxes   Route’s cost: 60,38

23. saving: 56,722,167,081
   nodes: 20-21
   1st route: 0-24-25-29-6-21-11-0
   Route’s demand= 72 boxes   Route’s cost: 67,54
   2nd route: 0-7-10-2-20-14-0
   Route’s demand= 68 boxes   Route’s cost: 60,38

24. saving: 56,657,109,561
   nodes: 6-27
   1st route: 0-24-25-29-6-21-11-0
   Route’s demand= 72 boxes   Route’s cost: 67,54
   2nd route: 0-7-10-2-20-14-0
   Route’s demand= 68 boxes   Route’s cost: 60,38

25. saving: 56,635,318,441
   nodes: 10-20
   1st route: 0-24-25-29-6-21-11-0
   Route’s demand= 72 boxes   Route’s cost: 67,54
   2nd route: 0-7-10-2-20-14-0
   Route’s demand= 68 boxes   Route’s cost: 60,38

With the same way of thinking and the same rules the saving’s list is processed with the help of VBA code in excel. The results of the execution of the VBA code are presented further down.

Now let’s see the VBA code which was used for the execution of the parallel version of Clark and Wright algorithm in excel.
Sub clarkewright_parali()
' Dim fortio, forticio, fortio_still As Integer
Dim numb, num, x, i, fortio, forticio, fortio(s), x, help1, help2, a(0 To 49), k, q, solver, flagl, flag2, place
Worksheets("clerk").Activate

For x = 13 To 21
For still = 6 To 22
Cells(still,x) = 0
Next
Cells(still, 25) = 0
Next
Defl = 0
Defl = 0
z = 0
Next

For i = 6 To 54
Cells(i, 25) = 0
Next

For i = 13 To 21
For x = 25 To 53
Cells(i, x) = 0
Next
Next

End Sub

Figure 10-14 The part of the code which initializes the variables and arrays.

For x = 1 To Cells(1, 1) + Cells(1, 1) Then Help = 1
If Cells(i, 25) = Cells(i, 25) Then Help = 1
Next

For x = 1 To Cells(1, 1) + Cells(1, 1) Then Help = 1
If Cells(i, 25) = Cells(i, 25) Then Help = 1
Next

For x = 1 To Cells(1, 1) + Cells(1, 1) Then Help = 1
If Cells(i, 25) = Cells(i, 25) Then Help = 1
Next

If Help = 1 And Help = 1 Then
    If Cells(i, 25) = Cells(i, 25) Then
        If Cells(i, 25) = Cells(i, 25) Then
            Cells(i, 25) = Cells(i, 25) + 1
        End If
    End If
End If

If Help = 1 And Help = 1 Then
    If Cells(i, 25) = Cells(i, 25) Then
        If Cells(i, 25) = Cells(i, 25) Then
            Cells(i, 25) = Cells(i, 25) + 1
        End If
    End If
End If

If Help = 1 And Help = 1 Then
    If Cells(i, 25) = Cells(i, 25) Then
        If Cells(i, 25) = Cells(i, 25) Then
            Cells(i, 25) = Cells(i, 25) + 1
        End If
    End If
End If

If Help = 1 And Help = 1 Then
    If Cells(i, 25) = Cells(i, 25) Then
        If Cells(i, 25) = Cells(i, 25) Then
            Cells(i, 25) = Cells(i, 25) + 1
        End If
    End If
End If

Figure 10-16 The part of the code which combines two routes, part 1
Figure 10-17 The part of the code which combines two routes, part2

Next
Cells(place1, 25) = Cells(place1, 25) + Cells(place2, 25)
Cells(place2, 25) = 0
Cells(place1, 29) = Cells(place2, 29)
Cells(place2, 29) = 0
Cells(place1, 31) = Cells(place1, 31) - Cells(place2, 31)
Cells(place2, 31) = 0
Cells(place2, 39) = 0
Cells(place2, 39) = 0
Cells(place2, 32) = 0
ElseIf place1 > place2 Then
For i = 0 To Cells(place1, 33) - 1
def1 = Cells(place1, 32) + 1 + 1
def2 = Cells(place1, 31) + 1
Cells(place2, def1) = Cells(place1, def2)
Cells(place1, Cells(place1, 31).Value + 1) = 0
Next
Cells(place2, 25) = Cells(place2, 25) + Cells(place1, 25)
Cells(place2, 50) = Cells(place1, 50)
Cells(place2, 32) = Cells(place2, 32) + Cells(place1, 33)
Cells(place2, 33) = Cells(place2, 33) + Cells(place1, 33)
Cells(place2, 31) = 0
Cells(place2, 39) = 0
Cells(place2, 39) = 0
Cells(place2, 29) = 0
Cells(place1, 29) = 0
End If
End If
End If

If flag1 = 2 And flag2 = 2 Then
If (Cells(place1, 25) + Cells(place2, 25) < 100) Then
If place1 < place2 Then
For i = 0 To Cells(place2, 33) - 1
Cells(place1, Cells(place1, 32).Value + (i + 1)) = Cells(place2, Cells(place2, 32).Value - i)
Cells(place2, Cells(place2, 32).Value - i) = 0
Next
Cells(place2, 35) = 0
Cells(place1, 30) = Cells(place2, 30)
Cells(place1, 32) = Cells(place1, 32) + Cells(place2, 33)
Cells(place1, 33) = Cells(place1, 33) + Cells(place2, 33)
Cells(place2, 30) = 0
Cells(place2, 29) = 0
Cells(place2, 31) = 0
Cells(place2, 32) = 0
Cells(place2, 33) = 0
ElseIf place1 > place2 Then
For i = 0 To Cells(place1, 33) - 1
Cells(place2, Cells(place2, 32).Value + (i + 1)) = Cells(place1, Cells(place1, 32).Value - i)
Cells(place2, Cells(place2, 32).Value - i) = 0
Next
Cells(place2, 35) = 0
Cells(place1, 30) = Cells(place2, 30)
Cells(place1, 32) = Cells(place1, 32) + Cells(place2, 33)
Cells(place1, 33) = Cells(place1, 33) + Cells(place2, 33)
Cells(place2, 30) = 0
Cells(place2, 29) = 0
Cells(place2, 31) = 0
Cells(place2, 32) = 0
Cells(place2, 33) = 0
End If
End If
End If
End If

If flag1 = 2 And flag2 = 2 Then
If (Cells(place1, 25) + Cells(place2, 25) < 100) Then
If place1 < place2 Then
For i = 0 To Cells(place2, 33) - 1
Cells(place1, Cells(place1, 32).Value + (i + 1)) = Cells(place2, Cells(place2, 32).Value - i)
Cells(place2, Cells(place2, 32).Value - i) = 0
Next
Cells(place2, 35) = 0
Cells(place1, 30) = Cells(place2, 30)
Cells(place1, 32) = Cells(place1, 32) + Cells(place2, 33)
Cells(place1, 33) = Cells(place1, 33) + Cells(place2, 33)
Cells(place2, 30) = 0
Cells(place2, 29) = 0
Cells(place2, 31) = 0
Cells(place2, 32) = 0
Cells(place2, 33) = 0
ElseIf place1 > place2 Then
For i = 0 To Cells(place1, 33) - 1
Cells(place2, Cells(place2, 32).Value + (i + 1)) = Cells(place1, Cells(place1, 32).Value - i)
Cells(place2, Cells(place2, 32).Value - i) = 0
Next
Cells(place2, 35) = 0
Cells(place1, 30) = Cells(place2, 30)
Cells(place1, 32) = Cells(place1, 32) + Cells(place2, 33)
Cells(place1, 33) = Cells(place1, 33) + Cells(place2, 33)
Cells(place2, 30) = 0
Cells(place2, 29) = 0
Cells(place2, 31) = 0
Cells(place2, 32) = 0
Cells(place2, 33) = 0
End If
End If
End If
End If

If flag1 = 2 And flag2 = 2 Then
If (Cells(place1, 25) + Cells(place2, 25) < 100) Then
If place1 < place2 Then
For i = 0 To Cells(place2, 33) - 1
Cells(place1, Cells(place1, 32).Value + (i + 1)) = Cells(place2, Cells(place2, 32).Value - i)
Cells(place2, Cells(place2, 32).Value - i) = 0
Next
Cells(place2, 35) = 0
Cells(place1, 30) = Cells(place2, 30)
Cells(place1, 32) = Cells(place1, 32) + Cells(place2, 33)
Cells(place1, 33) = Cells(place1, 33) + Cells(place2, 33)
Cells(place2, 30) = 0
Cells(place2, 29) = 0
Cells(place2, 31) = 0
Cells(place2, 32) = 0
Cells(place2, 33) = 0
ElseIf place1 > place2 Then
For i = 0 To Cells(place1, 33) - 1
Cells(place2, Cells(place2, 32).Value + (i + 1)) = Cells(place1, Cells(place1, 32).Value - i)
Cells(place2, Cells(place2, 32).Value - i) = 0
Next
Cells(place2, 35) = 0
Cells(place1, 30) = Cells(place2, 30)
Cells(place1, 32) = Cells(place1, 32) + Cells(place2, 33)
Cells(place1, 33) = Cells(place1, 33) + Cells(place2, 33)
Cells(place2, 30) = 0
Cells(place2, 29) = 0
Cells(place2, 31) = 0
Cells(place2, 32) = 0
Cells(place2, 33) = 0
End If
End If
End If
End If
Figure 10-18 The part of the code which combines two routes, part3
ElseIf flag1 = 1 And help2 = 0 Then
If Cells(place1, 31) - 1 > 0 Then
  If Cells(place1, Cells(place1, 31) - 1) = 0 And solver = 0 Then
    If Cells(place1, 25) + fortio1 <= q Then
      Cells(place1, Cells(place1, 31) - 1) = num2
      Cells(place1, 25) = Cells(place1, 25) + fortio2
      Cells(place1, 29) = num2
      Cells(place1, 31) = Cells(place1, 31) - 1
      Cells(place1, 33) = Cells(place1, 33) + 1
      z = z + 1
      Cells(25, 5 + z) = num2
    End If
  End If
End If
ElseIf flag2 = 1 And help1 = 0 Then
If Cells(place2, 31) - 1 > 0 Then
  If Cells(place2, Cells(place2, 31) - 1) = 0 And solver = 0 Then
    If Cells(place2, 25) + fortio1 <= q Then
      Cells(place2, Cells(place2, 31) - 1) = num1
      Cells(place2, 25) = Cells(place2, 25) + fortio1
      Cells(place2, 29) = num1
      Cells(place2, 31) = Cells(place2, 31) - 1
      Cells(place2, 33) = Cells(place2, 33) + 1
      z = z + 1
      Cells(25, 5 + z) = num1
    End If
  End If
End If
ElseIf flag1 = 2 And help2 = 0 Then
If Cells(place1, 32) + 1 = 0 Then
  If Cells(place1, Cells(place1, 32) + 1) = 0 And solver = 0 Then
    If Cells(place1, 25) + fortio1 <= q Then
      Cells(place1, Cells(place1, 32) + 1) = num2
      Cells(place1, 30) = num2
      Cells(place1, 25) = Cells(place1, 25) + fortio2
      Cells(place1, 32) = Cells(place1, 32) + 1
      Cells(place1, 33) = Cells(place1, 33) + 1
      z = z + 1
      Cells(25, 5 + z) = num2
    End If
  End If
End If
ElseIf flag2 = 2 And help1 = 0 Then
If Cells(place2, 32) + 1 = 0 Then
  If Cells(place2, Cells(place2, 32) + 1) = 0 And solver = 0 Then
    If Cells(place2, 25) + fortio1 <= q Then
      Cells(place2, Cells(place2, 32) + 1) = num1
      Cells(place2, 25) = Cells(place2, 25) + fortio1
      Cells(place2, 32) = Cells(place2, 32) + 1
      Cells(place2, 33) = Cells(place2, 33) + 1
      Cells(25, 5 + z) = num1
    End If
  End If
End If

Figure 10-19 The part of the code which add nodes into the routes
ElseIf help1 = 0 And help2 = 0 And solver = 0 Then
  i = 13
  Do While i <= 21 And solver = 0
    If Cells(i, 15) = 0 Then
      Cells(i, 15) = num1
      Cells(i, 16) = num2
      Cells(i, 25) = fortio1 + fortio2
      Cells(i, 29) = num1
      Cells(i, 30) = num2
      Cells(i, 31) = 15
      Cells(i, 32) = 16
      Cells(i, 33) = 2
      z = z + 1
      Cells(25, 5 + z) = num1
      z = z + 1
      Cells(25, 5 + z) = num2
      solver = 1
    End If
    i = i + 1
  Loop
End If
i = 0

Figure 10-20 The part of the code which creates new routes

The execution of Clark and Wright algorithm (Parallel Version) was made using

The initial view of excel is the following:

Figure 10-21 The initial display of the application
The rectangle with zeros is the space where the routes will be displayed. Each line represents a route. The vertical line of zeros next to the rectangle of zeros represents the total demand of the route, in other words, the total load of the truck that serves the route. Moreover in the horizontal line under the rectangle the nodes that are already in a route are placed. If we click on the initialization button, all our matrices are filled with zeros. If we click on the button "RUN CLARK AND WRIGHT ALGORITHM PARALLEL VERSION" then the algorithm starts to be executed and we get the following display:

![Image of the program interface](image)

*Figure 10-22 The program asks us which is the maximum capacity of the truck in boxes*
In our case the maximum capacity of the trucks is equal to 100, consequently we type “100” in the dialog box and we get the following results:

As we can see in the above picture, all 49 retail stores can be served by 7 routes and trucks. To be more specific,

1\textsuperscript{st} Route: 5-24-25-29-6-21-11-28 \hspace{1cm} \text{Total demand}=94,
\text{Total travel cost of the route}= 69,06

2\textsuperscript{nd} Route: 7-10-2-20-14-16-27 \hspace{1cm} \text{Total demand}=95,
\text{Total travel cost of the route}= 62,18

3\textsuperscript{rd} Route: 42-34-46-45-36-47-49 \hspace{1cm} \text{Total demand}=85
\text{Total travel cost of the route}=63,65

4\textsuperscript{th} Route: 18-8-12-19-4-9-32-1 \hspace{1cm} \text{Total demand}=99
\text{Total travel cost of the route}=60,38

5\textsuperscript{th} Route: 48-39-37-40-35 \hspace{1cm} \text{Total demand}=65
\text{Total travel cost of the route}=59,46

6\textsuperscript{th} Route: 26-13-15-33-31-30-23 \hspace{1cm} \text{Total demand}=91
\text{Total travel cost of the route}=60,40

7\textsuperscript{th} Route: 44-41-43-17-22-38-3 \hspace{1cm} \text{Total demand}=86
\text{Total travel cost of the route}= 68,93
The total demand is equal to \(94+95+85+99+65+91+86=615\) displays. Moreover, the total travel cost for the supply of all these retail stores is equal to \(69.06+62.18+63.65+60.38+59.46+60.40+68.93=444.05\). Due to the fact that the maximum capacity of each truck is 100 displays, 7 trucks will be used.

Figure 10-24 The first route of the optimal VRP solution (Parallel Version)

Figure 10-25 The second route of the optimal VRP solution (Parallel Version)
Figure 10-26 The third route of the optimal VRP solution (Parallel Version)

Figure 10-27 The fourth route of the optimal VRP solution (Parallel Version)
Figure 10-28 The fifth route of the optimal VRP solution (Parallel Version)

Figure 10-29 The sixth route of the optimal VRP solution
10.2 Clark and Wright algorithm (Sequential Version)

Now we will execute the sequential version of Clark and Wright algorithm. As we mentioned before, the difference with the parallel version is that the next route starts to grow only when the precedent route cannot take any more customers and not in a parallel manner. Now we will start to process a part of the savings list.

1. saving: 59,571 nodes: 6-21

We create the first route

1st route: 0-6-21-0 Route’s demand=24 boxes Route’s cost: 63,78

2. saving: 58,436 nodes: 6-29

Node 6 is an external node in the first route, the current demand of the route plus the demand of node 29 doesn’t surpass the maximum capacity of the truck which is equal to 100 boxes of products. Consequently we can add node 29 to the first route.

1st route: 0-29-6-21-0 Route’s demand=35 boxes Route’s cost: 68,08

3. saving: 57,948 nodes: 6-25

Nothing happens because node 6, which already belongs to the 1st route, is an internal node to the route, so node 25 can not be added to the route.

1st route: 0-29-6-21-0 Route’s demand=35 boxes Route’s cost: 68,08

4. saving: 57,755,58313 nodes: 2-20
Node 2 and node 20 are not included in the route, in this version of the algorithm we cannot create another route while the previous route isn’t completed. So we won’t add any node.

5. saving: 57,74358974  nodes: 11-21
   1st route: 0-29-6-21-11-0
   Route’s demand= 45 boxes  Route’s cost: 66,01

6. saving: 57,70  nodes: 25-29
   1st route: 0-25-29-6-21-11-0
   Route’s demand= 60 boxes  Route’s cost: 67,32

7. saving: 57,47461538  nodes: 21-29
   1st route: 0-25-29-6-21-11-0
   Route’s demand= 60 boxes  Route’s cost: 67,32

8. saving: 57,45820513  nodes: 24-25
   1st route: 0-24-25-29-6-21-11-0
   Route’s demand= 72 boxes  Route’s cost: 67,54

9. saving: 57,44219114  nodes: 21-27
   This case is the same with the case in the saving no: 3, so nothing happens.
   1st route: 0-24-25-29-6-21-11-0
   Route’s demand= 72 boxes  Route’s cost: 67,54

10. saving: 57,43361823  nodes: 5-29
    1st route: 0-24-25-29-6-21-11-0
    Route’s demand= 72 boxes  Route’s cost: 67,54

11. saving: 57,41367521  nodes: 6-24
    1st route: 0-24-25-29-6-21-11-0
    Route’s demand= 72 boxes  Route’s cost: 67,54

12. saving: 57,3063689  nodes: 20-29
    1st route: 0-24-25-29-6-21-11-0
    Route’s demand= 72 boxes  Route’s cost: 67,54

13. saving: 57,28148148  nodes: 24-29
1st route: 0-24-25-29-6-21-11-0
Route’s demand= 72 boxes Route’s cost: 67.54

14. saving: 57,23076923 nodes: 5-6

1st route: 0-24-25-29-6-21-11-0
Route’s demand= 72 boxes Route’s cost: 67.54

15. saving: 56,9114984 nodes: 6-28

1st route: 0-24-25-29-6-21-11-0
Route’s demand= 72 boxes Route’s cost: 67.54

16. saving: 56,89564103 nodes: 11-29

1st route: 0-24-25-29-6-21-11-0
Route’s demand= 72 boxes Route’s cost: 67.54

17. saving: 56,80769231 nodes: 7-10

1st route: 0-24-25-29-6-21-11-0
Route’s demand= 72 boxes Route’s cost: 67.54

18. saving: 56,78317308 nodes: 6-14

1st route: 0-24-25-29-6-21-11-0
Route’s demand= 72 boxes Route’s cost: 67.54

19. saving: 56,76622458 nodes: 6-16

1st route: 0-24-25-29-6-21-11-0
Route’s demand= 72 boxes Route’s cost: 67.54

20. saving: 56,76354167 nodes: 2-10

1st route: 0-24-25-29-6-21-11-0
Route’s demand= 72 boxes Route’s cost: 67.54

21. saving: 56,74461538 nodes: 2-21

1st route: 0-24-25-29-6-21-11-0
Route’s demand= 72 boxes Route’s cost: 67.54

22. saving: 56,74310897 nodes: 14-20

1st route: 0-24-25-29-6-21-11-0
Route’s demand= 72 boxes Route’s cost: 67.54

23. saving: 56,72216708 nodes: 20-21
1st route: 0-24-25-29-6-21-11-0
Route’s demand= 72 boxes     Route’s cost: 67,54

24. saving: 56,65710956       nodes: 6-27

1st route: 0-24-25-29-6-21-11-0
Route’s demand= 72 boxes     Route’s cost: 67,54

25. saving: 56,63531844       nodes: 10-20

1st route: 0-24-25-29-6-21-11-0
Route’s demand= 72 boxes     Route’s cost: 67,54

With the same way of thinking and the same rules the saving’s list is processed with
the help of VBA code in excel. The results of the execution of the VBA code are
presented further down.

If we click on the “RUN CLARK AND WRIGHT ALGORITHM SEQUENTIAL VERSION” then
the excel application executes the sequential version of the algorithm. And we get the
following result:

![Figure 10-31 The execution of the sequential version of clark and wright algorithm](image-url)
Similar to the parallel version, each row represents a route and next to the route we can see the sum of the route’s demand, in other words, the total load of the truck.

Figure 10-32 The first route of the VRP (Sequential Version)
Figure 10-33 The second route of the VRP (Sequential Version)

Figure 10-34 The third route of the VRP (Sequential Version)
Figure 10-35 The fourth route of the VRP (Sequential Version)

Figure 10-36 The fifth route of the VRP (Sequential Version)
As we can see in the above picture, all 49 retail stores can be served by 7 routes and trucks. To be more specific,

1\textsuperscript{st} Route: 5-24-25-29-6-21-11-2  
Total demand=97  
Total travel cost of the route=69,42

2\textsuperscript{nd} Route: 47-36-34-46-45-37-35  
Total demand=93  
Total travel cost of the route=66,17

3\textsuperscript{rd} Route: 3-7-9-28-4-27-32-14  
Total demand=94
Total travel cost of the route= 73,81
4th Route: 31-19-23-12-15-8-33  Total demand=94

Total travel cost of the route= 68,02
5th Route: 20-38-16-1-30-18-10  Total demand=95

Total travel cost of the route= 83,12

Total travel cost of the route= 83,52
7th Route: 44-17-43-48  Total demand=46

Total travel cost of the route= 61,99

The total demand is equal to 97+93+94+94+95+96+46=615 boxes of products. Due to the fact that the maximum capacity of each truck is 100 boxes, 7 trucks will be used.

Now we will calculate the cost of each route and the total cost of all routes, in other words, the total cost of the whole distribution.

So the overall cost is equal to 69,42+ 66,17+ 73,81+ 68,02+ 83,12+ 83,52+ 61,99=506,04

In the present, before the execution of the Clark and Wright algorithm, the routes that XYZ company uses in order to serve the 49 customers which we examined previously, are 9 and are presented below:

1st Route: 25-29-6-21-11-2  Cost of the route: 69,14
2nd Route: 16-27-14-5-24  Cost of the route: 60,46
3rd Route: 33-7-28-13-15  Cost of the route: 58,64
4th Route: 23-31-10-26-32  Cost of the route: 59,88
5th Route: 47-34-37-42-40  Cost of the route: 65,01
6th Route: 8-9-19-4-20  Cost of the route: 61,69
7th Route: 45-35-48-36-30  Cost of the route: 71,55
8th Route: 22-41-46-49-39-43  Cost of the route=74,40
9th Route: 1-44-38-3-12-17-18  Cost of the route= 69,06

The total cost of the above routes is equal to 589,83. The total travel cost of the whole distribution in the parallel version of the algorithm is equal to 444,05 while the total cost of the sequential version of the algorithm is equal to 506,04. So the optimal...
routes are the routes which resulted from the execution of the parallel version of Clark and Wright algorithm.

Conclusions

In this thesis we saw the application of both versions of Clark and Wright algorithm in a wholesale company. We saw an instance where information technology and mathematics can be applied in the logistics sector. Of course, in Greek companies these systems are not widely know and applied, also often these methods and algorithms are met with disbelief and disdain. But when these methods, reduce the costs and improve the processes inside the company then the results speak by themselves. At this stage, XYZ company, doesn’t use any routing software or algorithm in order to determine which load will be loaded to which truck and which truck will to go to which customer. All these decisions are made based on the experience of the employees. This happens because the company has a policy where the receipt of the order from the customer and the delivery of the products to the customer take place in the same day. This makes the scheduling and consolidation of the freight distribution very hard. Of course, the work experience of the employees is very important and necessary but the improvement which it can provide to the whole operation of the company has limits, in order to overcome these limits we need the help of mathematics and information technology which provide more accuracy and velocity in the processes inside the company.

One disadvantage of the application of Clark and Wright algorithm was that the data regarding the travel “cost” between the retail stores were based on searches in google map, which represents the reality in some point but not entirely. From time to time google maps give different results regarding the travel time between the same points. Consequently, the initial data were not a 100% accurate representation of the reality.

Regarding the execution of Clark and Wright algorithm with the help of excel worksheets and VBA code, the advantage is that the program which was made for the purposes of this thesis can be executed everyday, and everyday of the week it can provide the company with the optimal routes. The data regarding the customer’s demand and location are entered in an excel worksheet, and then the program takes these data and processes according to the program’s code, so since the data are entered in an excel worksheet they can be processed and altered easily. Also, as we can notice from the results of the execution of the algorithm, the parallel version of Clark and Wright algorithm gives us better results regarding the total cost of the routes. As it seems, one of the reasons that this happens is because the parallel version makes better use of the trucks. Moreover, as the results showed us, if Clark and Wright algorithm will be applied really inside XYZ company, then we will have a better utilization of the truck’s space in accordance with the cubic meters of the boxes which are used for transportation of the products. Moreover there is a chance that the company will make savings regarding the cost of the fuels and the workhours of the drivers.
Future research
In this thesis we dealt with the capacitated vehicle routing problem, in another thesis the researchers can examine the variations:

1. Vehicle Routing Problem with time windows
2. Vehicle Routing Problem with backhauls
3. Vehicle Routing Problem with pick up and delivery

The Capacitated Vehicle Routing Problem (CVRP) is an relatively simple and easy version of the Problem. Other companies may have a structure and operations which are more suitable to other versions of the problem which are more expertized. In this thesis we created an application with the help of VBA and excel, as an alternative, this application could be written in JAVA or C++.

References


investopedia


Knowledge-guided local search for the vehicle routing problem *Computers & Operations Research 14*


On the history of combinatorial optimization (till 1960)


Βαντσιούρης, Σ. (2013). Διαχείριση στόλου μεταφοράς και διανομής στην βιομηχανία γάλακτος. Θεσσαλονίκη.