UNIVERSITÉ DE TUNIS
INSTITUT SUPÉRIEUR DE GESTION DE TUNIS
****
UNIVERSITÉ DE NANTES
POLYTECHNIQUE DE NANTES

Mémoire de mastère
en Informatique et gestion de la connaissance

Multi-objective optimization of Vehicle Routing Problem with uncertainty: An approach based on possibility theory

---

Par

Oumayma BAHRI

(Soutenue le 19 Juin 2012)

Président: Philippe Leray, Professeur, Polytech-Universié de Nantes.

Rapporteur: Besma Kaabi, Maître de Conférence, ISG-Université de Tunis.

Directeurs: Nahla Ben Amor, Maître de Conférence, ISG-Université de Tunis.
Talbi EL Ghazali, Professeur, USTL-Université de Lille 1.

Abstract

The vehicle routing problem is considered as one of the greatest successes of operational research, having today many successful real applications in different domains. In our work, we are interested to a variant of VRPs which is the capacitated vehicle routing problem with time windows and stochastic demands (CVRPTW-SD), whose main goal is to find a set of routes with the lowest traveled distance and the minimum tardiness time. Furthermore, this problem is at the confluence of two important lines of research: Multi-objective optimization and Optimization under uncertainty.

In this context, we have observed that the optimization under uncertainty is well studied in the mono-objective case, contrary to the multi-objective case. Besides, we have found that the basic concept of uncertainty is to compare uncertain sets from the decisional space to the objective space. The aim of our study is firstly to propose a new possibilistic approach, in which the uncertainty is expressed using the two measures of possibility and necessity. In this approach and to offer more of flexibility to the decision maker, we propose to represent the stochastic demands by using triangular fuzzy numbers, instead of simple intervals. Consequently, the obtained results are also fuzzy. To taking into account these fuzzy results, we propose to use new dominance relations between triangular fuzzy numbers, instead of classic dominance relations between vectors. These relations are then used to develop a multi-objective local search algorithm for solving the CVRPTW-SD problem.

For the implementation, the software platform in C++ dedicated to multi-objective optimization, ParadisEO-MOEO, is used.

Keywords: Multi-objective Optimization, Vehicle Routing Problem, Stochastic demands, Possibility theory, Triangular fuzzy numbers, Pareto dominance relations, Multi-Objective Local Search, ParadisEo-MOEO.

Résumé

Les problèmes de tournées de véhicules (VRP) font partie des principaux problèmes étudiés en recherche opérationnelle, possèdant un grand nombre d'applications réelles dans de nombreux domaines. Dans ce travail, nous nous sommes intéressés à une variante de VRPs qui est le problème de tournées de véhicules avec fenêtre de temps et demandes stochastiques (CVRPTW-SD), dont le but principal est de trouver un ensemble de routes avec la distance traversée la plus courte et temps de retard le plus faible.

En outre, ce problème se situe à la confluuence de deux axes de recherches: L'optimisation multi-objective et l'optimisation sous incertitude. Dans ce cadre, l'optimisation sous incertitude est bien étudiée dans le cas mono-objectif, contrairement au cas multi-objectif. Par ailleurs, le concept de base de l'incertitude se traduit par la comparaison d'ensembles depuis l'espace décisionnel vers l'espace objectif. Le but de ce travail est de proposer une nouvelle approche possibiliste, dans laquelle l'incertitude est exprimée en utilisant les deux mesures de possibilité et de nécessité. Dans cette approche et afin d'offrir plus de flexibilité au décideur, nous avons choisi de représenter les demandes stochastiques par des nombres flous sous forme triangulaire au lieu d'intervalles simples. Par conséquent, les résultats obtenus sont également floues. Afin de tenir compte de ce type de résultats, nous proposons d'utiliser de nouvelles relations de dominances Pareto entre des nombres floues triangulaires, au lieu des relations de dominance classiques entre des vecteurs ou des intervalles. Ces relations sont par la suite utilisées afin de développer un algorithme multi-objectif de recherche locale pour la résolution du CVRPTW-SD.

Pour l'implémentation, la plateforme logicielle en C++ dédiée à l'optimisation multi-objective, ParadisEO-MOEO, est utilisée.

Mots clés: Optimisation multi-objectif, Problème de tournées de véhicules, Demandes stochastiques, Théorie de possibilité, Nombres floues triangulaires, Relations de dominance Pareto, Recherche locale multi-objectif, ParadisEO-MOEO.

Acknowledgments

I would like to express my sincerest thanks to my advisor, Dr. Nahla Ben Amor (Université de Tunis, Tunisie) for her advice, supervision and her important support throughout this work. She provided me guidance and support in all the time of research and writing of this report. Without her encouragement and understanding it would have been impossible for me to finish this work.

I am deeply grateful to my supervisor, Pr. Talbi El-Ghazali (Université de Lille, France) for accepting me and giving me the opportunity to work in the DOLPHIN team in INRIA laboratory and for his valuable advice and constructive comments. His wide knowledge and his logical way of thinking have been of great value for me.

I am also very thankful for all my teachers including those at ISG Tunis (master IGC) and Polytechnic Nantes (master ECD) for teaching me the necessary backgrounds for research initiation and how to present my ideas in the most rigorous, yet concise and clear way.

Special thanks go to my beloved Brahim Nasraoui, who helped me out in all fields. He is a bright professor, a very nice and intelligent person who supported me during the hard times of this master, from continued personal guidance to configuring my PC and printing my report.

Finally, this work is dedicated to my dear parents, Monia and Taoufik Bahri, for their love and encouragement during all my studies.

Contents

General Introduction **7**

1 Vehicle Routing Problem **9**

1.1 Introduction 9

1.2 The VRP Definition 9

1.3 Formal Description 11

1.4 Resolution methods for VRP 14

1.4.1 Exact methods 15

1.4.2 Approximate methods 16

1.5 Variants of VRP and their resolution methods 16

1.6 From single objective to multi-objective VRP and its resolution approach . 22

1.6.1 Types of objectives 22

1.6.2 Resolution approaches 23

1.7 Conclusion 23

2 Possibilistic approach for CVRPTW with fuzzy demands **26**

2.1 Introduction 26

2.2 Basics of possibility theory 26

2.2.1 Membership function in fuzzy sets theory 27

2.2.2 Possibility distribution 28

2.2.3 Possibility and necessity measures 28

2.3 CVRPTW problem with stochastic demands 30

2.4 Resolution approaches for CVRPTW with stochastic demands 33
2.5 New possibilistic approach for CVRPTW with fuzzy demands 35
2.5.1 Triangular fuzzy demands 35
2.5.2 Fuzzy capacity constraint 35
2.5.3 Constraint satisfaction expressed with possibility theory 36
2.5.4 Triangular multi-objective functions 38
2.6 Ranking triangular fuzzy numbers 39
2.7 Main concepts of Pareto approach 41
2.7.1 New pareto approach for ranking triangular fuzzy numbers 43
2.8 Conclusion 46

3 Algorithmic approach **48**
3.1 Introduction 48
3.2 Metaheuristics for multi-objective optimization 48
3.3 Multi-objective local search 49
3.4 DMLS algorithm 51
3.4.1 Components description 52
3.4.2 Existing algorithms 54
3.5 Implementation using ParadisEO platform 55
3.5.1 Presentation of ParadisEO framework 55
3.5.2 Presentation of ParadisEO-MOEO module 56
3.6 Application to CVRPTW with stochastic demands 57
3.6.1 Experimental analysis 57
3.6.2 Experimental design 58
3.6.3 Experimental results 60
3.7 Conclusion 61

Conclusion **63**

List of Figures

1.1 Classic VRP 10

1.2 Example of VRP 14

1.3 VRP: Resolution methods 15

1.4 Variants of VRP (An arrow moving from problem A to problem B means that B is an extension of A) 17

2.1 Three forms of membership Functions 28

2.2 Example of route failures 31

2.3 Triangular fuzzy demands 35

2.4 Example of measures 37

2.5 Example of ranking two fuzzy triangles [85] 39

2.6 Example of order relation between triangular fuzzy numbers [53] 40

2.7 Example of dominance relation between fuzzy triangles [34] 41

2.8 Certain dominance ($\tilde{y} \prec_c \tilde{y}'$) 43

2.9 Strong uncertain dominance ($\tilde{y} \prec_{us} \tilde{y}'$) 43

2.10 Weak uncertain dominance 1 ($\tilde{y} \prec_{uw} \tilde{y}'$) 44

2.11 Weak uncertain dominance 2: ($\hat{y}$-$\hat{y}'$=0 or $\hat{y}$-$\hat{y}'$>0) 44

2.12 Strong dominance 45

2.13 Weak dominance 46

2.14 Indifference 46

3.1 Local search 50

3.2 DMLS 51

3.3 Modules of ParadisEO framework 56
3.4 Neighborhood operator 2-opt [4] 58
3.5 Deterministic case(C101) 62
3.6 Uncertain case(C101) 62

List of Tables

1.1 Multi-objectives vehicle routing problems in the literature 24
1.2 Real life multi-objectives routing problems 25

2.1 Possibility measure Π 29
2.2 Necessity measure N 30

3.1 DMLS algorithms 55
3.2 Instance C101 59
3.3 Deterministic case results 60
3.4 Uncertain case 61

General Introduction

This memory falls under the domain of combinatorial multi-objective optimization, in particularly in the study of difficult logistic problems such as vehicle routing problems.

In the last decades, the multi-objective optimization has seen a growing interest with the appearance of metaheuristics. However, its principle idea is that the solution of a multi-objective problem is not a unique solution, but a set of solutions called Pareto optimal set. These solutions are those for which the improvement of one of objectives leads systematically to the deterioration in the quality of another objective. Therefore, one of the difficult issues is the identification of the Pareto optimal set, or an approximation of this set for complex problems.

The multi-objective optimization is applied in many academic and industrial areas for example the vehicle routing problems VRPs which are considered as one of the most studied problems in operational research. In this memory, we are interested to a variant of VRP in the multi-objective case.

The VRP is considered as one of the most studied problems in the literature. The well known applications of VRP involve the commercial distributions of many products in order to optimize industrial transport tasks. Examples of applications of VRP in transportation of goods are: the transportation of food, the distribution of oil (Service offshore oil), Newspaper distribution...

In addition, the VRP model can be used not only for applications concerning the delivery or collection of goods but for different real world applications such as school bus routing, transportation of handicapped persons, routing of salespeople, and routing of maintenance units. Note that, the main idea of the basic VRP is to create an optimal set of routes by visiting customers and by considering a single objective that aims often to minimize the transportation costs. However, most of real-world routing problems are multi-objective by nature. For example, in VRP, there are several variants and according to their complexity, multiple constraints can be introduced. Thus, in this case, the problem will be considered as a multi-objective one.

Besides, most of real optimization problems are subject to uncertainties which must be taken into account. But, despite the importance of multi-objective optimization and optimization under uncertainty, few works on multi-objective optimization under uncertainty exists today. For example, the VRP are usually studied in the mono-objective and deterministic form. Yet, there are many multi-objective routing problems presenting uncertainty in their description.

In practice, solving this type of problems with exact methods is often impossible due to the complexity of problem, to the multi-objective aspect and to data uncertainty. Therefore, different metaheuristics likes evolutionary algorithms or local search algorithms have been used to solve such problems. In our work, we are interested to solve a variant of VRP where the customer demands are uncertain. To taking into account this uncertainty of demands, we propose first a new approach based on possibility theory where the uncertain demands are presented by triangular fuzzy numbers and then to compare the obtained fuzzy solutions, we propose a new dominance relations between triangular fuzzy numbers. Finally, we applique the local search algorithms based on these dominance relations to resolve the problem.

This report is organized into three chapters:

- *Chapter 1: Vehicle Routing Problem*: This chapter starts by presenting VRP in general and gives a summary of its resolution methods. Then, it presents the well known variants of VRP with an overview of works proposed in the literature to solve each variant. Also, in this chapter, a more attention is given to multi-objective VRP and its resolution approachs since it represents the core of our work.

- *Chapter 2: Possibilistic approach for CVRPTW with fuzzy demands*: This chapter starts by giving the necessary background of possibility theory and by describing the CVRPTW-SD problem that we treat in our work. In this chapter we suggest a new possibilistic approach to solve our problem. Then, we propose new pareto relations between triangular fuzzy number in the mono-objective and multi-objective cases.

- *Chapter 3: Algorithmic approach*: This chapter presents the algorithmic approach and its application to solve the CVRPTW-SD problem in multi-objective case and by using the dominance relations proposed in the previous chapter.

Chapter 1

Vehicle Routing Problem

1.1 Introduction

The vehicle routing problem, commonly denoted by VRP, is one of the well Known and important combinatorial optimization problem which has an enormous interest and a central role in the design of distributions systems and logistics in many sectors. Thus it serves as model for many practical problems in transport, which are often studied with taking into account a unique objective function. However, most of real problem in the industry are multi-objective by nature and presents many challenges.

The main goal of this chapter is to present an overview of vehicle routing problem in the literature. Therefore, the chapter is organized as follows: Sections 1.2 and 1.3 describe the basic VRP, its specific characteristics and its formal description. A summary on the proposed resolution methods to solve this problem is given in Section 1.4. Then, an overview on the common variants of VRP is given in Section 1.5. Finally, Section 1.6 gives more attention to multi-objective VRPs by summarizing the important works proposed to treat them.

1.2 The VRP Definition

VRP was first introduced by Dantzig and Ramser in 1959. The first publication relative to this problem was proposed in 1971 by Elion [22]. In the last decades, many variants and models of this problem have been discussed and examined in the literature.

The basic VRP consists of a set of identical vehicles, based at a central depot, which must be optimally routed to supply a large number of customers with known demands and

only one vehicle is allowed to visit each customer. Therefore, the main objective aims to find a set of routes at a minimal cost, beginning and ending the route at a single depot.

Example 1.1. *An example of VRP is shown in figure 2.1with 8 customers represented by nodes, a central depot and 3 vehicles.*

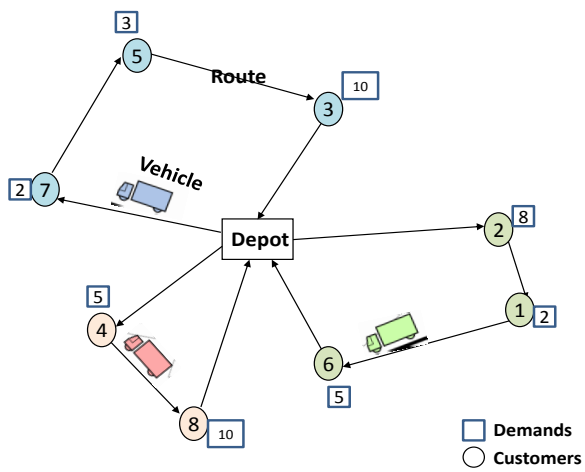

Figure 1.1: Classic VRP

Possible objectives may be to find a set of routes which:

- Minimize the total cost of the route.
- Minimize the total distance traveled.
- Minimize the total time required, the delay time and the waiting time.
- Minimize the number of vehicles required.
- Minimize the size of the fleet.
- Maximize the quality of service offered.
- Maximize the benefit gathered by selection of the customers.

1.3 Formal Description

There are several mathematical formulations of the VRP. In what follows, we choose to present a mathematical model of the most general and basic variant of VRP: CVRP (the capacitated vehicle routing problem).

In this problem, all the customers and the demands are deterministic and known in advance. The vehicles started from a single depot and must deliver the goods to a set of customers and then return to the depot. Also, the vehicles are identical and having a limited capacity. In fact, the CVRP may be defined as follows:

Let $G = (N, A)$ be a weighted directed graph with a node set $N_i = \{0, ..., n\}$ and an arc set A. The node N_0 is the central depot with m vehicles having the same capacity Q (homogenous vehicles). Each node $N_i \in N$, S.t. $N_i \neq N_0$ represents a customer with a known demand d_i. For each arc $(i, j) \in A$ is associated a cost C_{ij} which can be interpreted as the travel cost spent to go from customer i to customer j.

Finally, the main objective of CVRP consists of finding a collection of vehicle routes to serve all the customers, with a minimal total cost. Note that, a route is defined as a simple cycle of graph G passing through depot 0 and a number of customers with respect to the vehicle capacity [65]. The constraints considered in this problem are:

- *The capacity constraint:* For each customer presented by a node i is associated a weight of demand d_i where the sum of customer demands must be less than or equal to the vehicle capacity Q fixed a priori.
- *The total time constraint:* The total time of feasible routes must not exceed the maximum time T fixed a priori.

The definitions and notations leading to the mathematical formulation proposed by Rego and Roucairol [66] are given as follows:

Parameters

N_n: The number of customers

m: The number of vehicles

Q: The capacity of vehicle

d_i: The demand of customer i

T_k: The maximal travel time of vehicle K

t_i^k: the load time needed for vehicle K and the time needed to unload in customer i

t_{ij}^k:the time needed to travel by vehicle k from customer i to customer$j$

C_{ij}: The travel cost (the length or the travel time) from customer i to customer j.

The decision variable :

$$X_{ij}^K = \begin{cases} 1 & \text{if the vehicle } K \text{ travels directly from customer } i \text{ to customer } j \\ 0 & \text{otherwise} \end{cases}$$

The objective function:

$$Min \quad \sum_{i=1}^{n}\sum_{j=1}^{n} C_{ij} \sum_{k=1}^{m} x_{ij}^k \tag{1.1}$$

Under constraints

$$\sum_{i=0}^{n}\sum_{k=1}^{m} x_{ij}^k = 1, \qquad j = 1, \ldots, n \tag{1.2}$$

$$\sum_{j=0}^{n}\sum_{k=1}^{m} x_{ij}^k = 1, \qquad i = 1, \ldots, n \tag{1.3}$$

$$\sum_{i=0}^{n} x_{ip} - \sum_{j=0}^{n} x_{pj} = 0, \qquad k = 1, \ldots, m \quad p = 0 \ldots n \tag{1.4}$$

$$\sum_{i=0}^{n} d_i \ (\sum_{j=1}^{n} x_{ij}^k) \ \leq \ Q, \qquad k = 1, \ldots, m \tag{1.5}$$

$$\sum_{i=0}^{n} t_i^k \sum_{j=0}^{m} x_{ij}^k + \sum_{i=0}^{n}\sum_{j=0}^{n} t_{ij}^k x_{ij}^k \ \leq \ T_k, \qquad k = 1, \ldots, m \tag{1.6}$$

$$\sum_{j=1}^{n} x_{0j}^k \ \leq \ 1, \qquad k = 1, \ldots, m \tag{1.7}$$

$$\sum_{i=1}^{n} x_{i0}^k \ \leq \ 1, \qquad k = 1, \ldots, m \tag{1.8}$$

$$\sum_{i \in S} \sum_{j \in S} x_{ij}^k \leq |S|, \qquad \forall |S| \subset N, \qquad 2 \leq |S| \leq n-1 \tag{1.9}$$

$$x_{ij}^k \in \{0,1\}, \qquad i = 0, \ldots, n, \quad j = 0, \ldots, n, \quad k = 1, \ldots, m \tag{1.10}$$

- **The objective function (1.1):** is to minimize the total cost of the vehicle routes.
- **Equations (1.2) and (1.3):** the constraints assume that only one vehicle is allowed to visit each customer. Means that each customer is visited just once and by one vehicle.
- **Equation (1.4):** the constraint imposes that a vehicle K must leave a customer visited after complete their service.
- **Equation (1.5):** the constraint imposes that the sum of the demands of the customers visited in a specific tour cannot exceed the vehicle Capacity Q.
- **Equation (1.6):** the constraint imposes that the global travel time does not exceed the maximal travel time.
- **Equation (1.7):** the vehicle k starts their route from the central depot 0 to a customer just once.
- **Equation (1.8):** the vehicle K terminates by returning to the depot once, after completing their route.
- **Equation (1.9):** is the connectivity constraint to eliminating the sub tours. It imposes that no solution will contain a sub tour of customers disconnected from the depot and that all vehicle routes will be feasible.

Moreover, the addition, change or remove of some constraints can allow us to switch from one problem to another problem derived from the basic version of VRP. For example, from the mathematical model presented above, we can obtain a modeling for the Distance-Capacity constrained VRP (DCVRP) by adding a constraint of maximum distance.

Example 1.2. *Let us consider the example of CVRP in figure 2.2, which corresponding to 4 vehicles with capacity 10 and a set of 12 customers represented by nodes and served by a central depot. The demand of each customer is represented by a small square associated to each node and the travel costs are specified as labels of edges.*

This example illustrates a solution to the CVRP, which consists of finding a collection of vehicle routes, denoted r_i to serve all the customers with a minimal total cost, while

satisfying operational constraints presented above. First, we must make sure that each customer is covered by one selected route, which is used by just one vehicle. A feasible route has a load less than or equal to the vehicle capacity and is denoted by a chain of nodes starting and ending with the node 0 (the single depot).

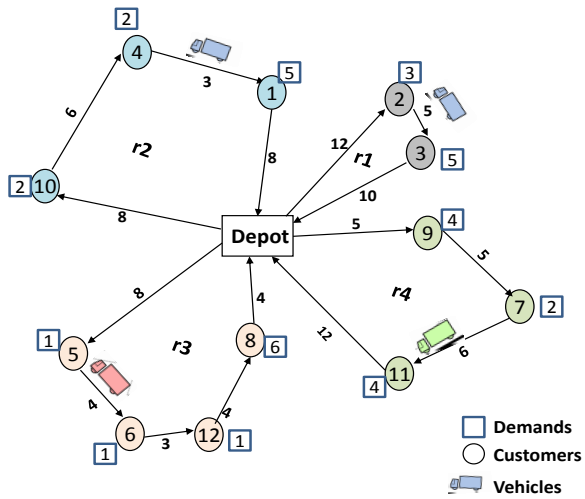

Figure 1.2: Example of VRP

For example consider nodes 2 and 3 served by vehicle 1, the feasible route to serve this two nodes or customers is given by $r_1 = (0, 3, 2, 0)$. *The load of r1 (the sum of the demands of the two customers) is equal to 8* $(d_2 + d_3 = 3 + 5 = 8)$ *which is less than 10 (the vehicle capacity). The cost of this route is 27 (10+5+12=27). Finally a feasible solution for this example is given by* $x = r_1, r_2, r_3, r_4$. *The cost of the solution is 103, which is the optimal solution.*

1.4 Resolution methods for VRP

The VRP is a NP-hard problem and it can be solved by two types of solution methods: by exact methods and by approximate ones. Exact methods are used to find the optimal solution, whereas the approximate methods are developed to find acceptable solutions. Besides, the approximate methods can be classified into two other groups: Heuristics and

Metaheuristics. Figure 2.3 summarizes some of these methods.

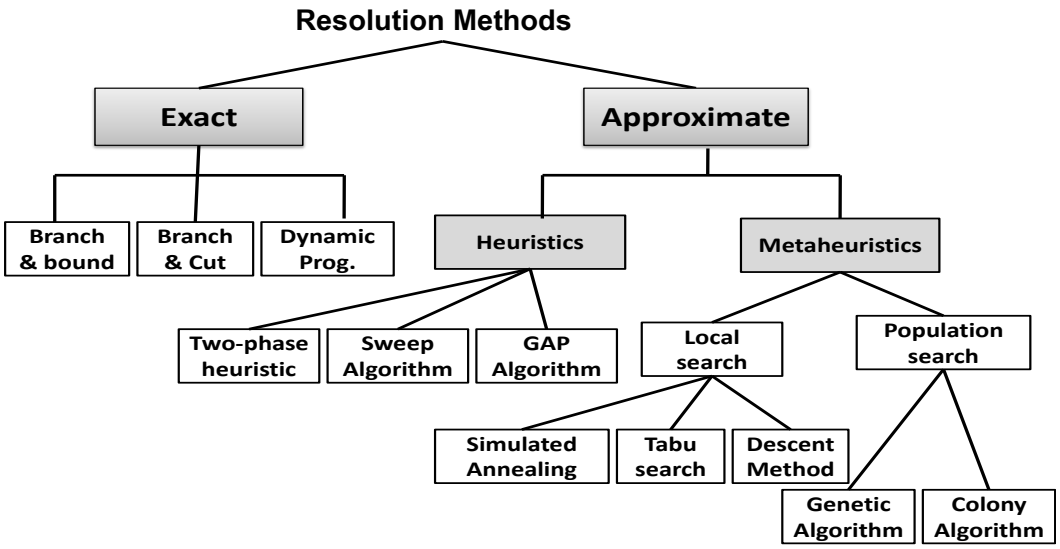

Figure 1.3: VRP: Resolution methods

1.4.1 Exact methods

Several exact methods have been developed to solve the VRP. In this section, we present the important and well known algorithms.

The *branch and bound algorithms* have been introduced until the late 1980 and then analyzed and detailed by Laporte and Nobert in [46]. Christofides and al. [13] have proposed the earliest branch-and-bound algorithm in 1981 to solve only small instances. In few years later, other branch-and-bound algorithms have been studied by several researchers likes Fischetti [24] and Fisher [25].

The *branch and cut algorithms* have been described as a combination of branch and bound algorithms and cutting plane methods. The first complete branch-and-cut approach to solve the CVRP has been proposed in 1995 by Augerat and al. [5]. Then, this approach has been enhanced by several researchers likes [27, 52, 64]. For example, Fukusawa [27] has developed a successful branch-and-cut-and-price algorithm that combining branch-and-cut with the principle of k-routes proposed by Christofides [13]. Currently, branch-and-cut algorithm is considered as the best available exact method for solving the CVRP.

Dynamic programming represent another type of exact methods which is based on the principle of Bellman [7]. This method is mainly used in [66] for solving routing problems with a small number of customers (varies from 10 to 25).

Finally, the serious drawback of all these exact methods is the exponential execution time, that increases with the size of the VRP. Therefore, The exact methods cannot be able to solve problems with large instances (not enough for practical use).

1.4.2 Approximate methods

An impressive number of approximate methods have been proposed in last decades for all kinds of VRP. However, unlike exact methods, approximate methods are useful quickly (with an acceptable execution time) for the problems with large instances and providing near-optimal solutions. As shown in figure 2.3, these methods are also divided into two other classes of algorithms : Heuristic and metaheuristics.

Heuristics: A large amount of heuristics have been used extensively to solve the VRP likes: the sweep algorithm proposed by Gillet and Miller [32] and used to planar VRP instances, the Savings heuristic known as Clarke and Wright algorithms [14], the Fisher and Jaikumar algorithms that uses Generalized Assignment Problem (GAP) [26].

Metaheuristics: Unlike classical heuristics, metaheuristics perform a more thorough search of the solution space and are less likely to end with a local optimum. Besides, contrary to heuristics, the application of a metaheuristic is totally independent from the problem to be treated. In recent decades, several metaheuristics approaches have been proposed to the VRP. These approaches can be divided into two other classes:
(1) Local search, including simulated annealing, deterministic annealing, Descent method and tabu search;
(2) Population search, including evolutionary algorithms likes genetic algorithm and Ant colony algorithm;
A good summaries and surveys of VRP meta-heuristics can be found in [16, 17, 29].

1.5 Variants of VRP and their resolution methods

Several versions and extensions of the basic VRP have been presented in the literature. We summarize in figure 2.4 the main variants of VRP and their interconnections.

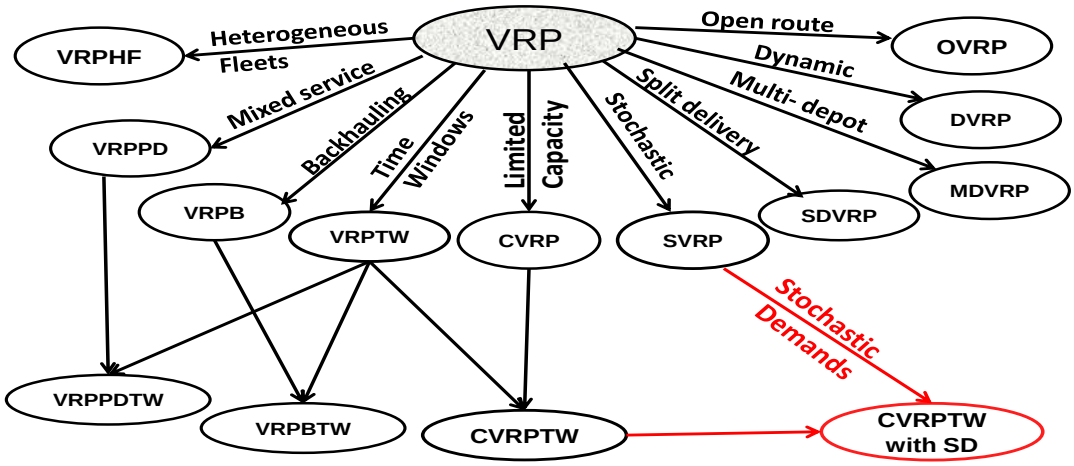

Figure 1.4: Variants of VRP (An arrow moving from problem A to problem B means that B is an extension of A)

CVRP (Capacitated Vehicle Routing Problem)

This problem has been represented in detail above as a basic version of VRP, where only the capacity constraint are considered and the objective is to minimize the total cost or length of the feasible routes. Another variant of CVRP is called the distance and capacity constrained vehicle routing problem (DCVRP), where both the vehicle capacity and the maximum distance constraints are imposed. Thus, DCVRP extends CVRP by adding a maximum distance constraint which imposes that the total length of each route cannot exceed the maximum length fixed a priori.

In the literature, several researchers have tried to solve the CVRP. Indeed, Toth and Vigo [89] present a review of exact algorithms used to solve CVRP such as the branch and bound algorithms. In their paper, only the capacity constraint is imposed and the main objective is to minimize the total cost. Achuthan and al. [1] treat also the CVRP by introducing a new method of cuts generation. This method consists of developing a new cuts and a search procedure to identify for example the violation of sub tours.

Since exact algorithms still not able to find solutions for large problems, several re-

searchers have proposed an approximate methods to deal with this problem for example: Baker and Ayechew [9] who used a genetic algorithm to solve the CVRP. Their main idea is that, once the feasible routes are identified, the vehicle tours are improved by using 2-OPT and 3-OPT methods.
Prins [59] who also adopted a genetic algorithm for this problem. Besides, some other researchers likes Taillard [76] and Toth [88] have chosen to use a tabu approach to solve the CVRP. The main idea of the heuristic of Taillard [76] is to decompose the problem into sub problems and then a tabu search algorithm is used for each sub problem. Finally, for large instances Reimann and al. [67] have proposed another solution based on a composite heuristic inspired by the ant colony algorithm.

VRPTW (Vehicle Routing Problem with Time Windows)

VRPTW is one of the most studied problems. It extends the classic VRP by adding new time constraints: for each customer is associated a time interval $[a_i, b_i]$ called as time window. Thus, the service of each customer must begin and terminate within the associated time window. Then, if the vehicle early arrive at the customer i, it can wait until the time instant ai to starting the service.

Several researchers have tried to solve the VRPTW by different methods, for example, Hwang [35] who presented a graphical interface based on the genetic algorithm to solve the VRPTW, Li and Lim [49] who used a simulated annealing algorithm to solve it, Berger and Barkaoui [49] who developed a new genetic algorithm for the VRPTW, based on a partial relaxation of some constraints and Lau and al. [47] who treated an other variant of VRPTW known as M-VRPTW, in which a limited number of vehicles is available.

VRPB (Vehicle Routing Problem with Backhauls)

VRPB is another variant of basic VRP, consists mainly of fixing an order of visit for the customers to be served and the suppliers to pick up goods. However, this precedence constraint imposes that all deliveries of goods must be performed before the collections. In other words, the suppliers must be visited before the customers to rearranging the load of vehicle along the route. Another problem has been studied in the literature, known as VRPB with Time Windows (VRPBTW) which is a combination of the two previous problems VRPB and VRPTW.

VRPPD (Vehicle Routing Problem with Pickup and Delivery)

VRPPD is similar to the VRPB, but here for each customer, two quantities are associated: a quantity to be delivered and a quantity to be picked. The precedence constraint imposes that, at each customer the delivery of goods is performed before the pickup and by the same vehicle. Thus, the net quantity used for each customer corresponds to the difference between the delivery and the pickup demands. The VRPPD with Time Windows (VRPPDTW) is another problem studied in the literature, which consists of adding the constraint of time windows to the VRPPD.

In recent decades, VRPPDTW has taken an important interest. In fact, some researchers have proposed an exact methods to solve this problem likes the authors Psaraftis [61], Desrosiers, Dumas and Soumis [21] who introduced a dynamic programming methods or the authors Sigurd, Pisinger and Sig 2000 who proposed a branch and bound algorithm to treat this problem. As we mentioned above, the exact methods are limited to obtain solutions for only small number of customers. For this reason, several researchers have chosen to use an approximate methods. For example, Psaraftis [60] has developed an heuristic based on the minimum spanning tree method and capable of handling up 100 customers. Lim and Rodrigues [50] have proposed a resolution method that use the SWO (Squeaky Wheel Optimization) with a local search. Finally, Landrieu, Mati and Binder [44] have developed two resolution methods for VRPPDTW : a classic tabu search with two procedures for exchanging nodes and a probabilistic tabu search in which the probabilities are used to choose the best candidate for the exchange procedure.

VRPHF (Vehicle Routing Problem with Heterogeneous Fleet)

VRPHF is another generalization of the VRP in which the vehicles does not share the same characteristics. In this problem, customers are served by a sets of different types of vehicles, having various capacity and variable costs.

The paper of Renaud and Boctor [68] presents the routing problem with a fleet of heterogeneous vehicles having different capacities. The authors propose a sweep heuristic to generate different routes. Tarantilis and al. [83] treat the same problem in a different way, by using a fixed fleet, means that the fleet in this case is composed by a limited number of heterogenous vehicles. Their paper describes a method based on an acceptation threshold named as Backtracking Adaptive Threshold Accepting (BATA).

MDVRP (Multi-Depot Vehicle Routing Problem)

MDVRP extends the VRP by using more than one depot (a multiple depots). The main objective of this problem is to determining the minimal total length of all the routes, in which a set of vehicles start from a multiple depots to serve all the customers and finally they terminate by returning to their original depots.

SDVRP (Split Delivery Vehicle Routing Problem)

SDVRP is another variant of VRP. Unlike most VRPs which imposes that a customer demand must be less than or equal to the vehicle capacity and each customer must be visited by just one vehicle, in SDVRP the customer demand can exceed the capacity of vehicles and also certain customers can be served by more than one vehicle. This relaxation of constraints may be more beneficial in several real cases.

OVRP (Open Vehicle Routing Problem)

OVRP is another well known variant of VRP. In this variant, the vehicles are not required to return to the depot after completing the service of the last customer on their route but if they are required, they return by using the same route back. Thus, in OVRP, the objective is firstly to minimize the number of vehicles used to serve all the customers and secondly to minimize the traveled distance.

DVRP (Dynamic Vehicle Routing Problem)

Most of real world VRPs are dynamic by nature and change regularly when the vehicles start their tours. This class of problems, called as as dynamic vehicle routing problems DVRPs, was mainly proposed to consider real-time informations. For example, in this problem, new customer demands can appear during tours execution or vehicles down. Consequently, with this dynamic appearance of demands, the tours must be rearranged in order to serve the customers.

Stochastic-VRP (Stochastic Vehicle Routing Problem)

SVPR is another well-studied variant of VRP that has a great importance and several applications in real-life. Unlike classic VRP, when all the data of problem are known a priori, SVRP has some random parameters such as random set of customers to be visited, uncertain demands and random travel and service times. In fact, this class of problems is

NP-hard and arises in situations with large random elements. For example, in SVRP with stochastic demands, when all the parameters except the demands are deterministic, it's so difficult and even impossible to determine the routes that satisfy all the demand points. Finally, the SVRP has different categories:

- ***VRP with Stochastic Demands VRPSD***: where the customer demands are uncertain.
- ***VRP with Stochastic Customers VRPSC***: where the existence of each customer on the route is uncertain.
- ***VRP with Stochastic Customers and Demands VRPSCD***: where the existence and the demands of each customer are unknown.
- ***VRP with Stochastic Traveling Time VRPSTT***: where the traveling time between two customers is a stochastic process that depends on traffic, weather or road conditions.

Tillman [87] introduced a method based on the Savings algorithm to solve the M-DVRP with stochastic demands. Stewart and Golden [74] proposed a Savings algorithm based on CCP (constraints chance programming) to treat the SVRP with stochastic demands. A review of a number of algorithms used to solve the SVRP with stochastic customers are presented by Bertsimas in [68]. Laporte and al [45] suggested a CCP model for the SVRP with stochastic traveling times. Finally, Kenyon and Morton tried to solve the SVRP with random travel and service times by developing two models: the first model is meant to minimizing the expected completion time and the second is meant to maximizing the probability that the operation is complete before a target time. The both models used the branch and cut techniques.

CVRPTW with Stochastic Demands

CVRPTW is one of the most studied variants of routing problems because of its great and practical relevance in real life. It can be defined as a combination of the two problems presented previously: CVRP and VRPTW. Thus, in addition to the vehicle capacity constraint introduced by the CVRP, the vehicles must respect the constraint of time windows associated with each customer and provided by the VRPTW.

In our work, we are interested to CVRPTW with stochastic demands . This problem is also a combination of two other problems or variants as shown in figure 2.4: CVRPTW and VRPSD. Therefore, this problem is based firstly on the principle of classic CVRPTW, where all the data are deterministic, excepting the customer demands which are stochastic

or uncertain. In the literature, CVRPTW with stochastic demands is defined as one of approaches that transform a mono-objective vehicle routing problem into a multi-objective one. Thus, in the next section, we are interested to the multiobjective class of vehicle routing problems.

1.6 From single objective to multi-objective VRP and its resolution approach

VRPs are frequently used to model real world applications. However, as shown previously, most of classical variants are often limited to a single objective of minimizing the total cost of the solution. Although, in real life, the majority of transport problems are multi-objective by nature and may not be limited to the total cost.

In our research, we are interested to this case of multi-objective problems. First, we must note that the multi-objective routing problems are mainly used to extend classic academic problems and to study real-life cases. In addition, unlike single-objective routing problem that provides a single optimal solution, in multi-objective case, we tend to determine a set of points representing the compromise solutions between different objectives.

1.6.1 Types of objectives

In routing problems, the objectives may be classified by the component to which they are related, for example, the objectives on tour, the objectives on resources and the objectives on node/arc activities. In this section, we present the most common types of objectives in the literature.

- Objectives on tour: Several studies seek to minimize the total cost of the tours. This cost can be expressed in terms of the total traveled distance, the total time needed or also the number of customers visited. On the other hand, there are several other studies that ignore the costs and that seek to minimize the Makespan which is the duration of the longest tour. Another type of natural objective consists of inducing an equilibration between tours, for example, by defining the quantity of delivered goods, the length of the tour or the required time. This kind of objective can be used to provide a notion of equity between employees and customers.

- Objectives on resources: For vehicle routing problems, the resources are mainly the vehicles and the goods. However, in the literature, the most studied objective is to minimize the number of vehicles. This objective can has the following economic significance: "Fewer is the number of vehicles, fewer is the investment". An other common objective about

vehicles consists of maximizing their profit in terms of capacity or time used by the vehicle. Besides, an objective related to goods may be the minimization of the damage of goods.

- Objectives on node/arc activities: Most of studies including objectives on nodes or arcs treat the routing problems with time windows. In this case, the time window can be ignored and replaced by an objective that minimizes the number of violated time windows or the waiting time of customers or drivers, and in some cases, the both objectives can be considered in the same time.

1.6.2 Resolution approaches

In the last decades, many techniques have been proposed for solving multiobjective routing problems. These techniques can be classified into three general categories: (1) scalar methods, (2) Pareto methods and (3) Other methods that not belong to the two previous categories.

In the literature, most of studies such as [48, 57, 58, 75, 92] use scalar methods to treat the different objectives. However, the majority of them use an aggregation method that transform a multi-objective problem to a mono-objective one. Other different scalar methods are also used like the Programmed Order proposed in [57, 58] and the ϵ-constraint approach introduced in [55]. An other approach based on lexicographic method has been also proposed in [38, 39]. On the other hand, several other studies propose to use the Pareto approaches to solve multi-objective problems, instead of using scalar methods.

Table 3.1 summarizes different studies in the literature for multi-objective VRPs and table 3.2 describes the resolution methods for some real life multi-objective routing problems [78].

1.7 Conclusion

In this chapter, we have presented the vehicle routing problem in general and the multi-objective variant in particularly. Starting from its mathematical formulation and its main variants to its state of the art, we have enumerated the different resolution methods proposed to resolve such vehicle routing problems. The next chapter will focus on the resolution approach of a specific variant of VRP namely CVRPTW-SD in which the demands are uncertain and two objective are to be optimized.

Authors	**Problem**	**Method**	**Tour**	**Nodes/Arcs**	**Resources**
Baran and Scharer [6]	Vehicle routing problem	Ant-colony Algo.	Min. the total time	Min. the total delivery times	Min. the number of vehicles
Geiger[28] (2001,2003)	Vehicle routing problem with time windows	Pareto approach Genetic algorithm	Min. the total distance	Min. the total deviations from the time window bounds	Min. the number of vehicles
Rahoual and al.(2001)[63] Djoukh. [62]	Vehicle routing problem with time windows	Pareto approach	Min. the traveled distance	Min.the number of violated constraints	Min.the number of vehicles
Jozefowiez and al., [38, 36, 37]	Covering tour problem	Pareto A Genetic Algorithm ϵ-Constr. Tabu S.	Min. the length	Min. the cover	
Tan and al.[82]	Vehicle routing problem with TW	Pareto A Genetic Hybrid A	Min. the total length		Min. the number of vehicles
Ribeiro and Lourenco [69]	Multi-period vehicle routing problem	Aggregat. Iterated local S.	Min. the traveled distance Optimize balance	Marketing: driver/client relationship	
Ombuki and al.[54]	Vehicle routing Problem with TW.	Pareto A Weighted -sum Genetic Algorithm	Min. the total length		Min. the number of vehicles
Chitty and Hernandez (2009)[12]	Dynamic vehicle routing problem	Ant Colony Optimizat. system	Min. the total mean transit time. Min. the total variance in transit time		
Lee and Ueng [48]	Vehicle routing problem	Aggregat. Heuristic	Min. the traveled distance Optimize balance of the load		

Table 1.1: Multi-objectives vehicle routing problems in the literature

Authors	**Problem**	**Method**	**Tour**	**Nodes/Arcs**	**Resources**
El-Sherbeny (2001) [23]	Vehicle routing problem adapted to the case of a Belgian Transportation company	Pareto approach Simulated Annealing	Min. the total time Optimize the balance. Max. the flexibility	Min. the waiting time.	Min. the number of trucks Min. the number of covered trucks Min. the number of uncovered trucks
Corboran and al. (2002)[15]	Rural schoolbus routing pb.	ϵ-constr. approach scatter S.	Min the makespan		Min. the number of vehicles
Bowerman and al. (1995)[10]	Urban schoolbus routing	Aggregat. Heuristics	Min. the total length Optimize the balance of the load	Min. the student walking distance	
Giannikos (1998)[31]	Location and routing for hazardous waste treatment	Goal prog-ramming	Min. the total cost Min. the total perceived risk	Min of the individual perceived risk or individual disutility	
Zografros and Androus (2004) [92]	Vehicle routing with TW for hazardous product transportation	Aggregat. Heuristics	Min. the length Min. the risk		
Tan et al. (2006) [41]	Truck and trailer VRP	Pareto A Hybrid genetic A.	Min. the total length	Min. the number of vehicles	

Table 1.2: Real life multi-objectives routing problems

Chapter 2

Possibilistic approach for CVRPTW with fuzzy demands

2.1 Introduction

In real world problems, our information or knowledge are often imperfect. This imperfection can be resulted from a lack of information, in this case we talk about uncertainty or also it can be caused by incomplete description of a reality known as imprecision. These two different concepts of imprecision and uncertainty are closly related in the sense: more the information is precise, more it is uncertain and vice versa. In this work, we are interested to uncertainty aspect. In fact, an information is said to be uncertain, if we do not have full knowledge about the veracity of any information. In the last decades, several tools have been proposed to deal with uncertain data. In our research, we are intereseted primaly to possibility theory.

Thus, we start this chapter by recalling some important concepts in Section 2.2. Then, Section 2.3 summarizes the problem we are interested which is the CVRPTW-SD. Finally, in the remaining sections of this chapter, we especially suggest a new possibilistic approach to handle our problem and a new pareto relations for rankig the obtained solutions.

2.2 Basics of possibility theory

Possibility theory have been first introduced by Zadeh [91] as an extension of fuzzy set theory and then developed by Dubois and Prade [20]. The basic idea of this theory is to use fuzzy sets, no to just represent the gradual aspect of vague concepts such as "high", but also to represent both imprecise and uncertain knowledge by introducing new concepts

of possibility and necessity measures. In fact, this theory informs us in which measure the realization of an event is possible and in which measure we are sure without having any evaluation of probabilities.

Before introducing the basic principles of possibility theory, we must define some concepts of fuzzy sets theory which are mainly related with possibility theory.

2.2.1 Membership function in fuzzy sets theory

The main idea of the fuzzy sets theory is to represent many situations that involve uncertainty in a suitable way and to deal with natural language uncertainty (or linguistics variables) like 'small' or 'big' with a gradual aspect presented by a graded membership. A graded membership is used to represent and qualify the meaning of these elements by associating a level of satisfaction for each element known as membership degree $\mu(x)$ and it is expressed by a number between 0 and 1 (with top denoted 1 and bottom denoted 0). More formally, a fuzzy set $A \neq \emptyset$ is characterized by a membership function:

$$\mu_A : X \rightarrow [0, 1] \tag{2.1}$$

This membership function indicates the degree that an element x, of the universel set X belongs to the fuzzy set A as following:

$$\mu(x) \begin{cases} 1 & \text{means x belongs to A} \\ 0 & \text{means x not belongs to A} \\ ]0..1[& \text{means x belongs partially to A} \end{cases} \tag{2.2}$$

In general, a membership function takes several forms such as triangular, rectangular or trapezoidal shape. Selection of the form depends on the type of sets to be represented. Figure 2.1 illustrates an example of three linear representations of a membership function. Finally, the above theory is in complete accordance with the basic principles of possibility theory where gradual notions of possibility and necessity are present. In other words, starting from the idea of gradual membership in fuzzy theory, possibility theory is meant to provide a gradual/possibilistic semantic to natural form of uncertainty.

Besides, contrary to other uncertainty theories, possibility theory is considered as the simplest one, mathematically and it offers a flexible framework for handling incomplete data in two different ways: a quantitative way (by numeric values) or a qualitative way (ranking relations between events).

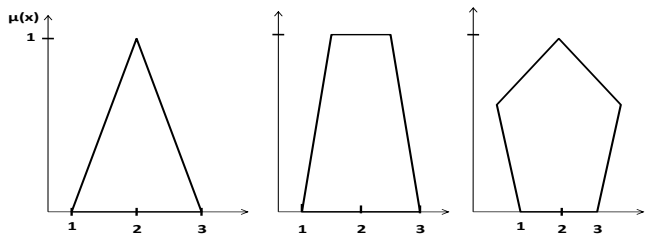

Figure 2.1: Three forms of membership Functions

2.2.2 Possibility distribution

The basic notion of possibility theory is the possibility distribution, denoted π, which is similar to probability density and which assigns to each element u in a set of alternatives U a degree of possibility $\pi \in [0, 1]$ of being the correct description of a state of knowledge.

Indeed, a possibility distribution function (attached to a single-valued variable x) is defined to be numerically equal to the membership function of a fuzzy set F: $\pi_x = \mu_F$ and it represents the more or less plausible value of the unknown variable x. Thus, technically, a possibility distribution is a normal fuzzy set (at least one membership grade equal to 1).

However, possibility theory can also be derived without reference to fuzzy sets. In fact, $\pi_x(u) = 1$ means that $x = u$ is a totally possible or plausible situation but it could also not happen, so this situation expresses the case of total ignorance. Whereas, $\pi_x(u) = 0$ is an impossible situation (the event sure will not happen) and this situation expresses the complete knowledge about x. In the context of constraint satisfaction problem, x represents a decision variable and the problem will be to select a suitable value of x. In this case, the fuzzy set of admissible values for x according to the associated constraint, can be naturally described by means of possibility distribution discussed above.

2.2.3 Possibility and necessity measures

Formally, possibility theory provides two dual measures: the possibility measure Π and the necessity measure N. These measures are used to transform a decision problem under uncertainty into deterministic terms.

Possibility measure: The possibility measure of any subset A, $\Pi(A)$ expresses the possibility degree that the event A will be realized. More this degree is near to 1, more

the event is possible. Whereas, a possibility degree equal to 0, indicates that the event is totally impossible. Table 3.1 summarizes the main properties of possibility measure.

Table 2.1: Possibility measure Π

$\Pi(A) = 1$ and $\Pi(\bar{A}) = 0$	A is certainly true
$\Pi(A) = 1$ and $\Pi(\bar{A}) \in]0, 1[$	A is somewhat certain
$\Pi(A) = 1$ and $\Pi(\bar{A}) = 1$	total ignorance
$\Pi(A) > \Pi(B)$	A is more plausible than B

Besides, if the subset is a disjunction of events, the possibility measure will be defined by the maximum of the possibility degrees of events in the disjunction. Formally, given two unrelated events A and B from the possibility distribution $\pi : R \longrightarrow [0, 1]$, the possibility measure of this subset is determined by the axiom of Maxitivity as the following:

$$\forall A, B \subset R, \quad \Pi(A \cup B) = max(\Pi(A), \Pi(B)) \tag{2.3}$$

On the other hand, when a fuzzy logic is adopted, the possibility measure will be expressed in terms of the supremum degree of the membership function:

$$\Pi(A) = sup \;\; \mu_A(x) \tag{2.4}$$

Finally, the possibility measure is insufficient to determine if an event A will be realized or not because we can having in the same time $\Pi(A) = 1$ and $\Pi(\bar{A}) = 1$, which expresses the case of total ignorance. Thus, to eliminate this case, we necessite new information about the possibility of the contrary event. Thus, necessity measure is meant for this reason.

Necessity measure: The necessity measure of any subset A, N(A) expresses the certainty degree that the event A will be realized. More this degree is near to one, more we are certain about the realization of this event. However, given a conjunction of events, this measure will be determined by the minimum of the necessity degrees of events in the conjunction. This axiom of Minitivity is the following:

$$\forall A, B \subset R, \quad N(A \cap B) = min(N(A), N(B)) \tag{2.5}$$

Table 3.2 gives the main properties of necessity measures. Note that N(A) is always inferior or equal to $\Pi(A)$.

Possibility- Necessity duality: From the above definitions of possibility and necessity measures, it is clear that these two functions are related to each other by the following dual relationship.

$$N(A) = 1 - \Pi(\bar{A}) \tag{2.6}$$

Table 2.2: Necessity measure N

$N(A) = 1$ and $N(\bar{A}) = 0$	A is certain
$N(A) \in]0, 1[$ and $N(\bar{A}) = 0$	A is somewhat certain
$N(A) = 0$ and $N(\bar{A}) = 0$	total ignorance

This indicates that A is necessarily true ($N(A) = 1$) only when $\bar{A}$ is not possible ($\Pi(\bar{A}) = 0$). From this duality , we can extract the following properties :

- $N(A) = 1$ and $\Pi(A) = 1$ ($\Pi(\bar{A}) = 0$) expresses the total certainty.
- N(A)=0 and $\Pi(A) = 1$ ($\Pi(\bar{A}) = 1$) expresses the total uncertainty.
- N(A)=0 and $\Pi(A) = 0$ ($\Pi(\bar{A}) = 1$) expresses the total certainty of not A.

2.3 CVRPTW problem with stochastic demands

In our work, we are interested to a well known variant of VRP (CVRPTW-SD), with a limited vehicle capacity to serve the customers, a time windows to respect and a stochastic demands. In fact, as we already mentioned in the previous chapter, this problem results from the combination of two other variants of VRP:

- CVRPTW that represents a generalization of the classic CVRP with the addition of time constraints provided by the VRPTW.
- VRP with Stochastic Demands: means that the existence of customer demands in the route is uncertain. In this case, we should add a set of constraints to represent the most probable situations due to route failures. However, in the literature two common strategies are used to solve this kind of problem:
  - First strategy: A priori optimization in which, when the vehicle becomes empty or its capacity becames insufficient to serve the remaining customers, it returns back to the depot to restock [30].
  - Second strategy: A preventive restocking, in which the vehicle returns to the depot specially when it is near to it and its capacity is low [8]. Note that our work focuses on a priori optimization strategy illustrated in the figure (2.2).

Formally, the mathematical model of this problem is based on the basic VRP formulation presented in the previous chapter with the addition of further constraints. This means that always there are:

- A node 0 that represents the central depot and the other nodes $\{1...n\}$ that represent the customers but with unknown demands dem_i.

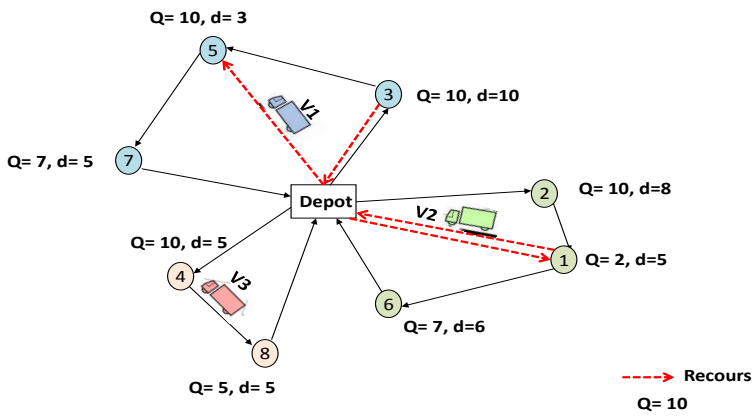

Figure 2.2: Example of route failures

- A set of homogenous vehicles, having a limited capacity Q which is not exceeded (Vehicle capacity constraint).
- For each arc (i, j) is associated a non-negative value C_{ij} that corresponds to the cost between two nodes i and j. Note that C_{ij} is proportional to distance and to travel time, meaning that $C_{ij} = d_{ij} = T_{ij}$.

In addition to these classic constraints, in CVRPTW-SD, two other important constraints are considered : the time windows constraint and the travel distance constraint.

Travel distance constraint: This constraint imposes that each vehicle with a limited capacity Q, must deliver goods to the customers according to their stochastic demands, with the minimal expected transportation costs in term of traveling time or distance. In other words, if the capacity constraint of this vehicle is not satisfied, the route fails and causes an additional costs. More formally, in CVRPTW-SD:

- the customer demands are stochastic means that the actual demand is only known when the vehicle arrives at the customer location
- A feasible route is represented by $r = (v_0, v_1, v_2, ..., v_s, v_0)$ where s is the size of the route.

- Each vehicle K has a symetric distance $d^k_{v_i v_j}$ to travel from node i to node j. This travel distance is proportional to the travel time T_{ij} and to the cost C_{ij}.

Therefore, to calculate the travel distance on a route $r = (v_0, v_1, v_2, ..., v_s, v_0)$ defined a priori, three different situations may be found (FIG. 2.2):

1- **The customer demand is smaller than the vehicle capacity ($\sum_{i=1}^{f} d_{v_i} < Q$):** In this case the vehicle will serve the current customer f and then move to the next one $(f+1)$.

2- **The customer demand is equal to the vehicle capacity ($\sum_{i=1}^{f} d_{v_i} = Q$):** In this case, a priori optimization strategy is used. Thus, the vehicle leaves the depot v_0 to serve the first customer f with its total capacity and as it becames empty, this vehicle will return to the depot to upload and go after from the depot to the next customer $(f+1)$. The total traveled distance is calculated as follows:

$$D(r) = d_{v_0 v_1} + \sum_{i=1}^{f-1} d_{v_i v_{i+1}} \; + \; d_{v_f v_0} + d_{v_0 v_{f+1}} + \sum_{i=f+1}^{s-1} d_{v_i v_{i+1}} + \; d_{v_s v_0} \tag{2.7}$$

3- **The customer demand is bigger than the vehicle capacity ($\sum_{i=1}^{f} d_{v_i} > Q$):** In this case the vehicle will serve the customer f with its remaining capacity $(Q - \sum_{i=1}^{f-1} d_{v_i})$, go to the depot to upload, return back to the same customer f to give him the remaining amount of his demand and then move to the next customer $(f+1)$. The total traveled distance in this case is calculated as follows:

$$D(r) = d_{v_0 v_1} + \sum_{i=1}^{s-1} d_{v_i v_{i+1}} \; + d_{v_f v_0} + d_{v_0 v_f} + \; d_{v_s v_0} \tag{2.8}$$

Time constraint: After defining the distance constraint, we must also defining the times required in our problem. In CVRPTW-SD, the time constraint (added mainly by the variant VRPTW) uses the following notations:

- Each client i will be served within his time windows $[e_i, l_i]$ by exactly one vehicle, where the lower bound e_i represents the earliest arrival time and the upper bound l_i represents the latest arrival time for the visit of vehicles.

- The time windows of the central depot is $[0, l_0]$, which means that each vehicle that leaves the depot at a time 0 go back to the depot before the time L_0.

- Each client imposes a service time S_i^k that corresponds to the goods loading/unloading time used by the vehicle K.

- A_i, B_i refers respectively to the arrival time and the departure time at customer i.
- A waiting time W_i, means that the vehicles must arrive before the upper bound of the window l_i
- T_{ij} refers to the travel time from customer i to customer j (which is proportional to C_{ij})

Firstly, the constraint considered to taking into account the travel time (T_{ij}) between two consecutive customers i and j is defined as follows:

$$x_{ij}^k(S_i^k + T_{ij} + S_j^k) \leq 0 \quad with \quad A_i \leq S_i^k \leq B_i \tag{2.9}$$

Besides, the vehicles must arrive between the time windows of each customer $[e_i, l_i]$, and when they arrived before the lower bound e_i, they must wait a while W_i (time wait). However, this waiting time is calculated as follows:

$$W_i = \begin{cases} 0 & \text{if } A_i \geq e_i; \\ e_i - A_i & \text{otherwise} \end{cases} \tag{2.10}$$

where, the arrival time at customer i is calculated by: $A_i = B_{i-1} + T_{i,i-1}$ and the departure time is equal to: $B_i = A_i + W_i + S_i$
Similarly,when the vehicle arrives at a customer i after the end of its time windows l_i, a delay time must be calculated as follows:

$$R_i = \begin{cases} 0 & \text{if } A_i \leq l_i; \\ A_i - l_i & \text{otherwise} \end{cases} \tag{2.11}$$

In case of route failure, an additional costs in term of the tardiness time will appear (FIG.2.2). Finally, all these above constraints combined with the constraints of the basic VRP presented in the previous chapter, models the problem CVRPTW with stochastic demands.

In our work, the objectives to be satisfied by this problem are: minimization of the total travel distance and the total tardiness time. Note that, when stochastic demands are represented by fuzzy numbers, the problem will be named as CVRPTW with fuzzy demands (to avoid any confusion).

2.4 Resolution approaches for CVRPTW with stochastic demands

Our interest in this work, is mainly to solve the CVRPTW with stochastic demands (CVRPTW-SD) problem described in the previous section with an approach based on

possibility theory. To summarize, CVRPTW-SD is an NP-hard vehicle routing problem where both the vehicle capacity and the time windows constraints are imposed and where the customer demands are stochastic and cannot be precisely determined in advance.

In the last decades, this variant of problem has taken an important interest. In fact, several researchers have tried to solve it and have proposed different approaches based on uncertainty theories to express the unknown customer demands. For example, in the majority of studies like [11, 70, 72, 86], a fuzzy approach based on fuzzy sets theory has been proposed to solve the CVRPTW-SD, in which the uncertainty of demands are represented by fuzzy sets such as triangular or trapezoidal fuzzy numbers. In this case the problem will be named as VRPTW with fuzzy demands. In some other studies, like [81, 84] uncertain demands follows simply a normal distribution such as classic probability or possibility distributions. While in [4, 51], a different approach based on evidence theory has been proposed to solve this problem, in which the uncertainty of demands is represented by a an interval of belief and plausibility.

An other interesting approach has been proposed in [33] which is based on both possibility theory and fuzzy set theory to solve this kind of problem. In this approach, the researchers choose first to use triangular fuzzy number to represent uncertain demands, then to use possibility theory to express the degree of satisfaction of the fuzzy constraint resulting from these fuzzy demands. Next, they tried to transform this fuzzy constraint (expressed with possibility theory) to a crisp equivalent formula by introducing two thresholds α and β for the possibility and necessity measures that can be viewed as the minimum levels of the constraint satisfaction. Finally, in this work, the CVRPTW-SD has been treated as a mono objectif problem by using an aggregation function of the set of objectives.

Our study will be an extension of this work. However, we use the same principle of this latter approach based on the two theories of uncertainty: fuzzy theory and possibility theory. We choose also to represent uncertainty of demands by triangular fuzzy numbers because this natural form of uncertainty offers more of flexibility for the decision maker and more facilitates the expression of customer demands. For example, based on experience, one can estimate that a demand of a customer is around 30 units, between 20 and 40, in this case, the demand will be simply represented by the fuzzy triangular number (20,30,40). Then, we choose to integrate the possibility theory to handle the fuzzy capacity constraint resulted, by using the two basics measures of possibility and necessity.

But contrary to the previous proposed approach, in our approach, the CVRPTW-SD will be considered as a multi-objective problem where the set of objectives to be optimized are respectively: minimizing the total traveled distance and the tardiness time. Besides, in our work, we remplace the idea of the two thresholds introduced for the possibility and necessity measures by using a linear combination of these latter.

2.5 New possibilistic approach for CVRPTW with fuzzy demands

In this section, we present our approach to solve the CVRPTW with fuzzy demands. Section 2.6.1 and Section 2.6.2 present the concept of the triangular fuzzy demands and the fuzzy constraint resulted in this case. Then, Section 2.6.3 adresses how to exprimate the satisfaction of this fuzzy constraint with the possibility theory.

2.5.1 Triangular fuzzy demands

The demand of each customer $i \in V$, is represented by a triangular fuzzy number denoted $\tilde{d}$, which consists of a triplet of real numbers ($\underline{d}$, $\hat{d}$, $\bar{d}$)with $\underline{d} < \hat{d} < \bar{d}$. A membership function μ is associated with each fuzzy number and it has a triangular representation (figure 2.3). The left boundary $\underline{d}$ and right boundary $\bar{d}$ are respectively the lower and higher limits of the demand, their corresponding membership degrees are equal to zero. The term $\hat{d}$ (the most possible demand) is the standardization value which corresponds to the maximum membership degree equals to one.

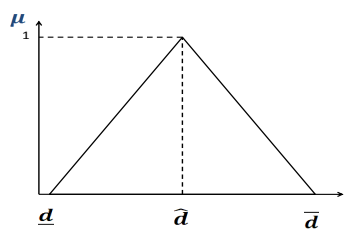

Figure 2.3: Triangular fuzzy demands

Since, the vehicle tours must be planned in advance to taking into account the fuzzy demands and to estimate the additional costs due to route failures. Note that, a route is called failure when the residual capacity of the vehicle is not sufficient to serve the planned customers,or simply if the capacity constraint is violated.

2.5.2 Fuzzy capacity constraint

In this section, we must remember the a priori optimization strategy and the three possible situations of vehicle tours formulated in the previous section to estimate the additional costs. These situations are related absolutely to the vehicle capacity constraint which

imposes that the total customer demands on each route should be equal to or less than the vehicle capacity i.e., $\sum\sum d_i x_{ij} \leq Q$. Besides, as the demands are now represented by fuzzy numbers, the total demands of the served customers will be also fuzzy numbers $\widetilde{D_k}$ defined by:

$$\widetilde{D_k} = \sum_{i=0}^{n}\sum_{j=1}^{n} \widetilde{d}_i x_{ij}^k = (\sum_{i=0}^{n}\sum_{j=1}^{n} \underline{d_i} x_{ij}^k + \sum_{i=0}^{n}\sum_{j=1}^{n} \widehat{d}_i x_{ij}^k + \sum_{i=0}^{n}\sum_{j=1}^{n} \overline{d_i} x_{ij}^k) = (\underline{D_k}, \widehat{D_k}, \overline{D_k}) \quad (2.12)$$

Thus, the capacity constraint changes to a fuzzy constraint defined as the following:

$$\widetilde{D_k} \ \widetilde{\leq} \ Q \quad (2.13)$$

where the symbol $\widetilde{\leq}$ is used to designate the comparison between a fuzzy number $\widetilde{D_k}$ and an ordinary number Q [33]. In this case, we cannot directly verify if the fuzzy capacity constraint is satisfied or not. For example, with a fuzzy demand $\widetilde{D_k} = (30, 40, 55)$ and a vehicle capacity $Q = 40$, how verify if the fuzzy capacity constraint is not satisfied? Thus, to handle the satisfaction of the fuzzy constraint, we propose to use the possibility theory with its two measures Π and N.

2.5.3 Constraint satisfaction expressed with possibility theory

In this step, the satisfaction of the fuzzy capacity constraint (Equation 2.13) is expressed in the possibility framework. The possibility and necessity measures which correspond to the degrees of satisfaction of this constraint, are defined as the following:

$$\Pi(\widetilde{D_v} \leq Q) = sup_{x \leq Q} \ \mu_{\tilde{D}}(x) = \begin{cases} 1 & , \widehat{D_v} \leq \text{Q} \\ \frac{Q - \underline{D_v}}{\widehat{D_v} - \underline{D_v}} & , \underline{D_v} \leq \text{Q} \leq \widehat{D_v} \\ 0 & , \text{Q} \leq \underline{D_v} \end{cases} \quad (2.14)$$

$$N(\widetilde{D_v} \leq Q) = 1 - \Pi(\widetilde{D_v} > Q) = \begin{cases} 1 & , \overline{D_v} \leq \text{Q} \\ \frac{Q - \widehat{D_v}}{\overline{D_v} - \widehat{D_v}} & , \widehat{D_v} \leq \text{Q} \leq \overline{D_v} \\ 0 & , \text{Q} \leq \widehat{D_v} \end{cases} \quad (2.15)$$

Example 2.1. *An example of constraint satisfaction expressed by the possibility and necessity measures is shown in figure 2.4, where the vehicle capacity* $Q = 50$ *and the fuzzy demand* $\widetilde{D} = (30, 40, 55)$*. In this case, the* $\Pi(\widetilde{D} \leq 50) = 1$ *because* $\widehat{D} = 40 \leq Q = 50$*, and the* $N(\widetilde{D} \leq 50) = \frac{50-40}{55-40} = 0.66$ *because* $40 \leq Q \leq 55$*.*

Finally, an often used definition, says that the possibility Π measures the best case and shows the most optimist attitude toward the events, while the necessity N measures the

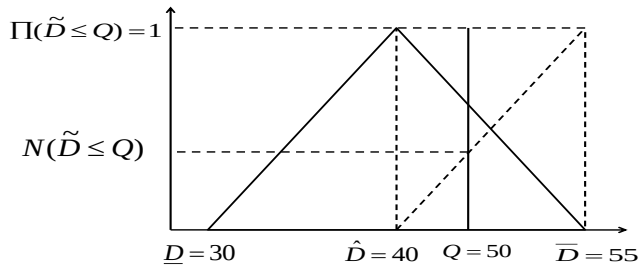

Figure 2.4: Example of measures

worst case and shows the most pessimist attitude. In the other hand, as we presented above, these two measures are related to each other by a dual relationship, so a combination of them allows the expression of both the optimistic and pessimistic attitude of the decision maker.

From these definitions, we can conclude that it is more simpler and efficient at this step, to use the linear combination of the possibility and the necessity measures proposed by [11], rather than treating each measure separatly. This linear combination is defined as the following: Given an event A, its weight denoted by W(A) which consists of the combination of the weighted possibility and necessity, is expressed by:

$$W(A) = \lambda \Pi(A) + (1 - \lambda) N(A) \tag{2.16}$$

Where the parameter $\lambda \in [0, 1]$, measures the degree of optimism or confidence of the decision maker such that:

$$\lambda = \begin{cases} 1 & \text{Total optimistic case} \\ 0 & \text{Total pessimistic Case} \\ 0.5 & \text{Neither optimistic nor pessimistic} \end{cases} \tag{2.17}$$

Once, the weight W(A) is computed, we should now verify if a solution of the weighted possibility and necessity satisfies the capacity constraint or not, we must use a threshold α of constraint satisfaction. Therefore, at confidence value α, the fuzzy capacity constraint is satisfied iff:

$$W(\widetilde{D} \leq Q) \geq \alpha, \quad with \;\; \alpha \;\; \in \;\; [0, 1] \tag{2.18}$$

This formula indicates that the weight measure, must be higher than a given threshold α (fixed by the decision maker). The higher it is, the greater the capacity constraint will

be satisfied and then less capacity defaults will occur, but larger will be then the generated cost. So, a lower value of this threshold generates a high risk.

2.5.4 Triangular multi-objective functions

We should first introduce the standard mathematical form of a multi-objective optimization problem to see the difference with the triangular form obtained in our case.

Definition 2.1. ***Multi-objective Optimization problem*** *A multi-objective optimization problem consists of solving a mathematical program of form:*

$$f\colon X \mapsto Y \subseteq R^n$$

$$f(x) : \begin{pmatrix} f_1(x) \\ .. \\ f_n(x) \end{pmatrix} = y = \begin{pmatrix} y_1 \\ ... \\ y_n \end{pmatrix} \tag{2.19}$$

where n ($n \geq 2$) is the number of objectives, $x=(x_1, x_2,...,x_n)$ is the vector representing the decision variables, and $y=(y_1, y_2,...,y_n)$ represents the set of feasible solutions which are clearly an exact values. $f(x)=(f_1(x),f_2(x),...,f_n(x))$ is the vector of objectives to be optimized.

The CVRPTW with fuzzy demands we are interested in this work, is a multi-objective problem where the objectives to optimized are the minimization of the total traveled distance and the tardiness time. Since, in our work the fuzzy demands are represented by a triangular fuzzy numbers, the solutions of our problem will not be an exact values but an uncertain values represented also as a triangular fuzzy sets. Therefore, the objective function changes formally to:

$$\mathrm{f\colon X \mapsto Y} \subseteq R^n$$

$$f(x) : \begin{pmatrix} f_1(x) \\ .. \\ f_n(x) \end{pmatrix} = y = \begin{pmatrix} [\underline{y_1}, \widehat{y_1}, \overline{y_1}] \\ ... \\ [\underline{y_n}, \widehat{y_n}, \overline{y_n}] \end{pmatrix} \tag{2.20}$$

Consequently, we will obtain the traveled distance and the tardiness time for our problem represented respectively by $[\underline{\mathrm{D}}, \hat{\mathrm{D}}, \bar{\mathrm{D}}]$ and $[\underline{\mathrm{T}}, \hat{\mathrm{T}}, \bar{\mathrm{T}}]$. At this step, the issue we are interested, is how ranking these triangular fuzzy solutions especially in a multi-objective case.

2.6 Ranking triangular fuzzy numbers

In the litterature, different methods for ranking fuzzy numbers have been proposed by several authors likes [18, 19, 34, 39, 40, 53, 85, 90]. Among these methods, we are interested to those suggested for comparing fuzzy numbers with triangular form, specially those based on preference relation as comparison functions or cutting technique.

In this section, three famous methods for ranking triangular fuzzy numbers are described and illustrated by examples. Thus, let $\tilde{y}$ be a triangular fuzzy number represented by a triplet $[\underline{y}, \hat{y}, \bar{y}]$, where $\hat{y}$ is the point whose membership value is 1 and $\underline{y}$, $\bar{y}$ are respectively the left and the right hands of triangle.

First, [85] proposes a simple comparison method consists of three steps tested progressively:

1. Comparing removals of each pair of triangles. The value of removal is computed by $R = (\underline{y} + 2 * \widehat{y} + \overline{y})/4$. Then if $R_y < R_{y'}$, $\tilde{y}$ is said smaller than $\tilde{y}$'. But if the removals are equals, we must move to the next criterion of comparison.

2. Comparing modes (If the modes are also equals then we moves to the third criterion).

3. Comparing divergences.

Example 2.2. *Let $\tilde{y}$=[3, 10, 11] and $\tilde{y}$'=[2, 10, 12] be two triangular fuzzy numbers as shown in figure 2.5. The removals of these two fuzzy triangles are: $R_y = \frac{3+2\cdot 10+11}{4} = 8,5$ and $R'_y = \frac{2+2\cdot 10+12}{4} = 8,5$. As the two removals are equals, the second criterion must be applied. Then, the both modes of the two fuzzy numbers are also equal to 10. In this case, the third criterion must be now considered. In fact, the divergences are respectively $D_y = 11 - 3 = 8$ and $D'_y = 12 - 2 = 10$. Finally, as D_y is inferior to D'_y, we can conclude that $\tilde{y}$ is smaller than $\tilde{y}$'($\tilde{y}$< $\tilde{y}$').*

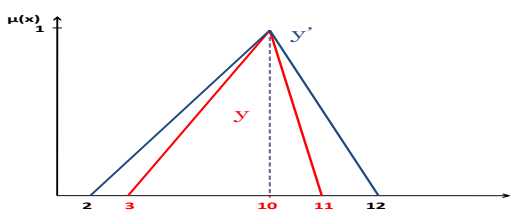

Figure 2.5: Example of ranking two fuzzy triangles [85]

This method gives no good solution for several situations because it classifies the fuzzy triangles only in the sense of ỹ is inferior to ỹ' without taking into account that the smaller triangle may be dominating by the other one.

In [53], a second method have been proposed, in which the ranking of two triangular fuzzy numbers consists of defining an order relation between them by using a comparison measure proposed by [71]. This measure is called 'Acceptability index'(A-index) and is defined as follows:

$$A(\tilde{y} \prec \tilde{y}) = \frac{\hat{y'} - \hat{y}}{(\bar{y} - \hat{y}) + (\hat{y'} - \underline{y'})} \quad with \begin{cases} \tilde{y} = [\underline{y}, \hat{y}, \bar{y}] \\ \tilde{y'} = [\underline{y'}, \hat{y'}, \bar{y'}] \end{cases} \tag{2.21}$$

Using this A-index, the following ranking orders or dominance relations among two triangular fuzzy numbers is defined as follows:

- If A(ỹ< ỹ') ≥ 1, ỹ is said to be 'totally dominating' over ỹ' in case of minimization (pessimistic choice) and ỹ' is said to be 'totally dominating' over ỹ in case of maximization(optimistic choice). This is denoted by ỹ$\prec$ ỹ'.
- If $0 <$ A(ỹ< ỹ') < 1, ỹ is said to be 'partially dominating' over ỹ' in the sense of minimization and the case is converse in the sense of maximization. This is denoted by ỹ$\prec_p$ ỹ'
- If A(ỹ< ỹ')= 0 and $\underline{y}$= $\underline{y}$' or $\bar{y} = \bar{y}$', ỹ and ỹ' are said to be non dominating.

Example 2.3. *Let us compare the following two fuzzy triangles illustrated in figure 2.6 and represented by: ỹ=[2, 5, 10] and ỹ'=[6, 12, 15]. Using the A-index, we have A(ỹ$\prec$ ỹ')=* $\frac{12-5}{(10-5)+(12-6)} = 0.63 < 1$. *Therefore, ỹ partially dominates ỹ' with a degree of satisfaction equal to 0.63. This order relation is denoted by ỹ$\prec_p$ỹ'.*

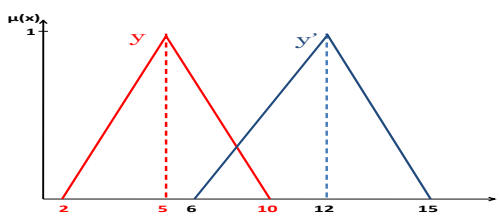

Figure 2.6: Example of order relation between triangular fuzzy numbers [53]

This second method gives a good ranking results between two fuzzy triangles because firstly it is based on the principle of dominance approach and secondly because it exprimates the decision maker choice and its degree of satisfaction.

The thrid method for ranking triangular fuzzy numbers is suggested by [34] and it is mainly based on the principle of $\alpha - cut$. In order to determine the ranking order, this method uses the left and the right spreads at some $\alpha - cuts$ levels of the triangular fuzzy numbers. Therefore, a strong dominance relation like $\ll y'$ strong dominates $y \gg$ is determined when all the $\alpha - cuts$ of y' dominates the $\alpha - cuts$ of y. This relation is denoted by $y \preceq_s y'$. However, if one $\alpha - cut$ level not verifying the strong relation, a weak dominance is determined. Finally, if all the $\alpha - cuts$ are equals, an indifference relation results.

Example 2.4. *In this example,we consider the two triangular fuzzy numbers $\tilde{y}$=[3, 8, 10] and $\tilde{y}$'=[2, 11, 15]. From the figure 2.7, we can clearly identify that the $\alpha - cuts$ levels (α1, α2...,etc.) of y' dominates those of y. Thus, a strong dominance between them is verified by $\ll y'$ strong dominates $y \gg$. Then, an optimistic person whould prefer y' while a pessimistic person would prefer y.*

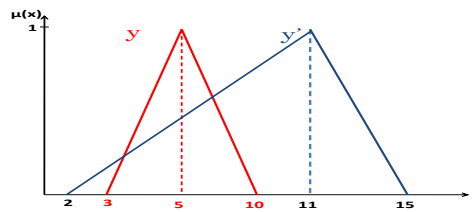

Figure 2.7: Example of dominance relation between fuzzy triangles [34]

This method is considered more complex compared to the two previous ones and is failed often to descriminate between two fuzzy numbers that have the same spreads. Finally, we clearly observe that each of the three above methods treats only some situations in general case. For example, the method [34] treats only the cases of strong and weak dominances whereas the method in [53] presents only the total and partial dominances. In the next section, a new Pareto approach for ranking triangular fuzzy number is proposed to deal with the maximum possible situations, but before describing our solution, we must defining some necessary concepts of pareto approach.

2.7 Main concepts of Pareto approach

In contrast to mono-objective optimization, a multi-objective optimization problem implies obtaing not only a single solution but a set of solutions called as Pareto optimal solutions. An elegant definition offered by Talbi EL ghazali in [77] states that: "A solution is Pareto optimal if it is not possible to improve a given objective without deteriorating at least

another objective". In other words, this set of solutions represents the compromise solutions between the different conflicting objectives. This section presents the main concepts of a Pareto approach such as Pareto dominance, Pareto optimality, Pareto optimal set and Pareto front.

Definition 2.2. ***Pareto dominance***
An objective vector u = (u_1, ...,u_n) is said to dominate another objective vector v = (v_1, ...,v_n)(denoted by $u \prec_p v$) if and only if no component of v is smaller than the corresponding component of u and at least one component of u is strictly smaller:

$$\forall i \in 1, ..., n : u_i \leq v_i \wedge \exists i \in 1, ..., n : u_i < v_i. \tag{2.22}$$

Definition 2.3. ***Pareto optimality***
A solution $x^* \in X$ is Pareto optimal (also known as efficient, non-dominated or non-inferior) if for every $x \in X$, F(x) does not dominate $F(x^*)$, that is, F(x)$\not\prec_p$ $F(x^*)$.

Definition 2.4. ***Pareto optimal set***
For a given MOP(F,X), the Pareto optimal set P^* is defined as:

$$P^* = \{x \in X / \exists x' \in X, F(x') \not\prec_p F(x)\}. \tag{2.23}$$

Definition 2.5. ***Pareto Front***
The Pareto front is the image of Pareto optimal set in the objective space. For a given MOP(F,X) and its Pareto optimal set P^*, the Pareto front is defined as:

$$PF^* = \{F(x), x \in P^*\} \tag{2.24}$$

The main goal of multi-objective optimization is to obtain the Pareto front, for which the decision maker can select an optimal solution based on the current situation [77]. Note that the Pareto relations depends mainly on the dimension of the objective functions. For example, the above pareto definitions are meant especially for a deterministic multi-objective case, where the dimension of objective functions is suitable for certain and exact values. Therefore, in the case of imprecise multi-objective functions, the normal pareto concepts must be modified to handle such imprecise functions. For example, in [51, 4] a new pareto relations have been proposed to handle imprecise objective functions represented by simple intervals.

Since, in our approach the imprecise objective functions are represented by triangular fuzzy numbers, the issue becames how to ranking these fuzzy triangles in the sense of pareto approach.

2.7.1 New pareto approach for ranking triangular fuzzy numbers

In this section, we first define a dominance relations between triangular fuzzy numbers in a mono-objective case. Then, we extend these dominance relations for the multi-objective case. Note that, in all the following pareto relations, the dominance is always defined in the sense of minimization.

A- Pareto relation in mono-objective case

Definition 2.6. ***Certain dominance****(Figure 2.8)*

Let $\tilde{y}$= [$\underline{y}$, $\hat{y}$, $\bar{y}$] $\subseteq \mathbb{R}$ and $\tilde{y}$'= [$\underline{y}$', $\hat{y}$', $\bar{y}$'] $\subseteq \mathbb{R}$ be two fuzzy triangles. $\tilde{y}$ ***dominates*** *$\tilde{y}$'* ***certainly or totally*** *(denoted by* $\tilde{\boldsymbol{y}} \prec_c \tilde{\boldsymbol{y}}$**'**) *iff:* $\bar{\boldsymbol{y}} < \underline{\boldsymbol{y}}$**'**

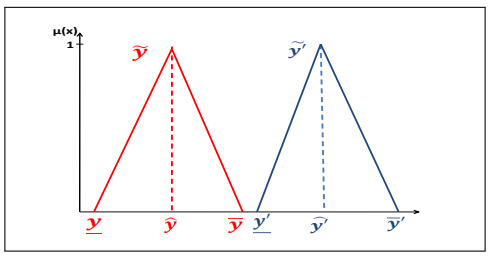

Figure 2.8: Certain dominance ($\tilde{y} \prec_c \tilde{y}$')

Definition 2.7. ***Strong uncertain dominance*** *(Figure* **??***)*

Let $\tilde{y}$= [$\underline{y}$, $\hat{y}$, $\bar{y}$] $\subseteq \mathbb{R}$ and $\tilde{y}$'= [$\underline{y}$', $\hat{y}$', $\bar{y}$'] $\subseteq \mathbb{R}$ be two fuzzy triangles. $\tilde{y}$ ***strong dominates*** *$\tilde{y}$'* ***uncertainly or partially*** *(denoted by $\tilde{y} \prec_{us} \tilde{y}$') iff:* $(\bar{\boldsymbol{y}} \geq \underline{\boldsymbol{y}}') \wedge (\underline{\boldsymbol{y}} \leq \underline{\boldsymbol{y}}') \wedge (\bar{\boldsymbol{y}} \leq \bar{\boldsymbol{y}}')$

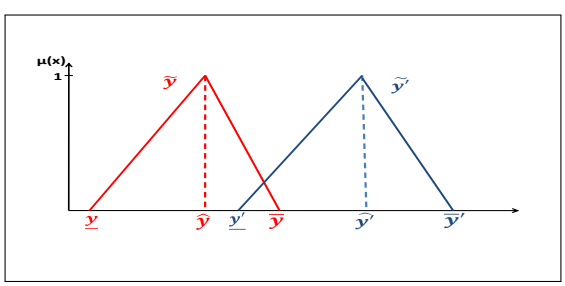

Figure 2.9: Strong uncertain dominance ($\tilde{y} \prec_{us} \tilde{y}$')

Definition 2.8. ***Weak uncertain dominance*** *(Figures 2.10 and 2.11)*

Let $\tilde{y}$= [$\underline{y}$, $\hat{y}$, $\bar{y}$] $\subseteq \mathbb{R}$ and $\tilde{y}$'= [$\underline{y}$', $\hat{y}$', $\bar{y}$'] $\subseteq \mathbb{R}$ be two fuzzy triangles. ***$\tilde{y}$ weak dominates $\tilde{y}$' uncertainly or partially*** *($\tilde{y} \prec_{uw} \tilde{y}$') iff:* ***($\bar{y} \geq \underline{y}$') $\wedge$ ($\bar{y} \leq \bar{y}$' $\wedge$ $\underline{y} > \underline{y}$') $\vee$ ($\underline{y} \leq \underline{y}$' $\wedge$ $\bar{y} > \bar{y}$').***

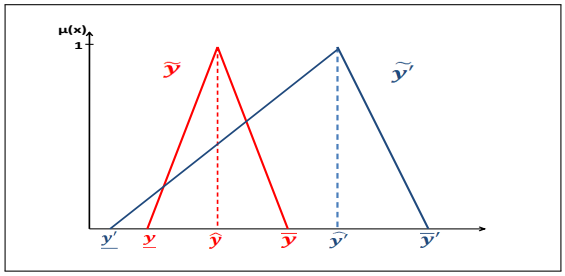

Figure 2.10: Weak uncertain dominance 1 ($\tilde{y} \prec_{uw} \tilde{y}$')

However, in some situations, two fuzzy triangles can be incomparable with this weak dominance. In this case, we conclude that the weak dominance relation between two fuzzy triangles depends mainly on the order relation between the modes of these two triangles: $\hat{y}$ and $\hat{y}$'. To identify the incomparable situations, we introduce a comparison measure which consists in comparing the modes of two fuzzy triangles. This measure is defined as follow:

$$\widehat{y} - \widehat{y'} = \begin{cases} < 0 & \text{Clearly, } \tilde{y} \text{ weak dominates } \tilde{y}' \\ \geq 0 & \tilde{y} \text{ and } \tilde{y}' \text{ can be incomparable} \end{cases}$$

Since the indifferent situations are identified, the issue we are interested now, is how to handle them. Our idea is to propose an other criterion of comparison which consists in comparing the discard between the both triangles as follow:

If ($\underline{y}$ - $\underline{y}$') $\leq$ ($\bar{y}$' - $\bar{y}$), then $\tilde{y}$ weak dominates $\tilde{y}$'.

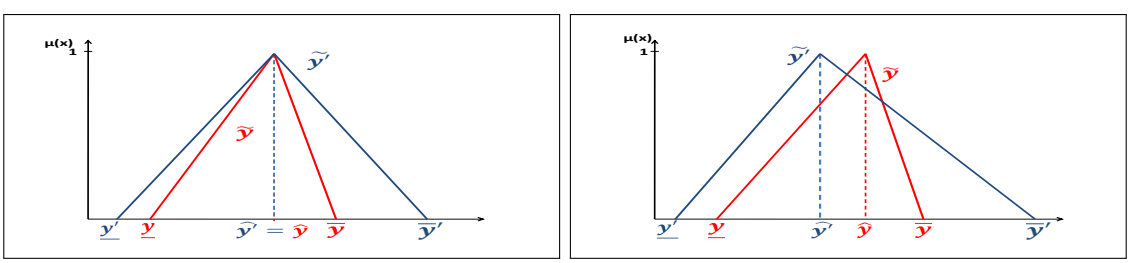

Figure 2.11: Weak uncertain dominance 2: ($\hat{y}$-$\hat{y}$'=0 or $\hat{y}$-$\hat{y}$'>0)

Clearly, in the mono-objective case, we obtained a total pre order between two triangular fuzzy numbers. However, the evaluation of fuzzy numbers in a multi-objective problem is one of the more difficult and complex tasks. So, it is very difficult and even impossible to obtain a pre-order between the fuzzy numbers in this case. In the next section, we propose an extension of our pareto relations to the multi-objective case.

B- Pareto relation in multi-objectibe case: To faciliate the comparison of triangular fuzzy numbers in the multi-objective case, the following definitions uses mainly the three dominance relations proposed previously, for the mono-objective case:
Certain dominance ($\tilde{y} \prec_c \tilde{y}$'), Strong uncertain dominance ($\tilde{y} \prec_{us} \tilde{y}$') and Weak uncertain dominance ($\tilde{y} \prec_{uw} \tilde{y}$').

Definition 2.9. ***Strong dominance*** *(Figure 2.12)*

A vector y (of triangular fuzzy numbers) strongly Pareto dominates another vector y' (denoted by $\tilde{y} \prec_S \tilde{y}$') iff:

$$\forall i \in 1..n : y_i \prec_c y'_i \vee$$
$$\forall i \in 1..n : y_i \prec_{us} y'_i \vee$$
$$\exists i \in 1..n : y_i \prec_c y'_i \wedge y_i \prec_{us} y'_i.$$

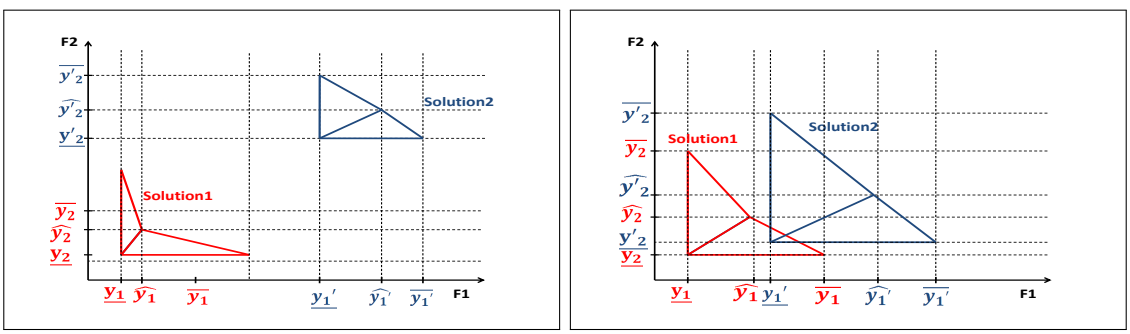

Figure 2.12: Strong dominance

Definition 2.10. ***Weak dominance*** *(Figure 2.13)*

A vector y (of triangular fuzzy numbers) weakly Pareto dominates another vector y' (denoted by $\tilde{y} \prec_W \tilde{y}$') iff:

$$\forall i \in 1..n : y_i \prec_{uw} y'_i \wedge$$
$$\exists i \in 1..n : (y_i \prec_c y'_i \vee y_i \prec_{us} y'_i).$$

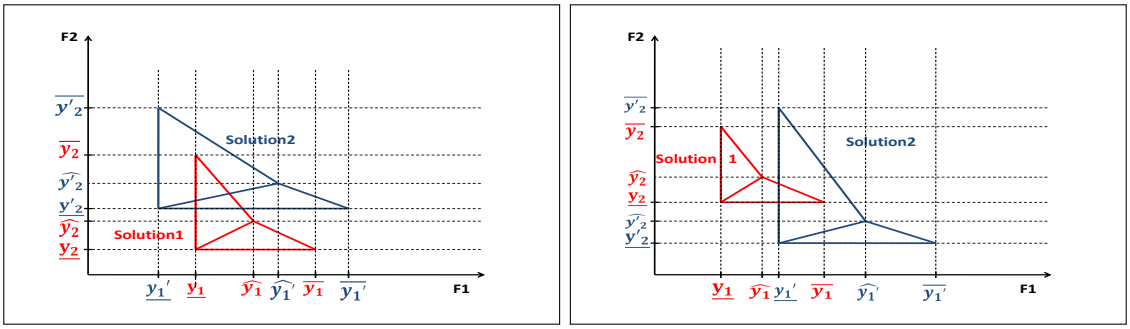

Figure 2.13: Weak dominance

Definition 2.11. ***Indifference*** *(Figure 2.14) Two vectors of triangular fuzzy numbers are indifferent or incomparable (denoted by y‖y') iff:*

$$\forall i \in 1..n : y_i \prec_{uw} y'_i$$

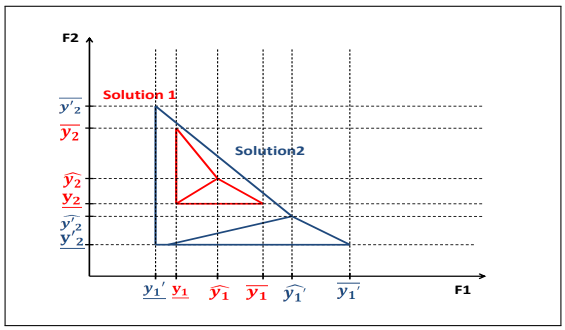

Figure 2.14: Indifference

Therefore, weak dominance will only hold if y at least weak dominates y'(uncertainly) in all objectives and certain or strong dominates y' in one, while strong dominance holds if y certain or strong dominates y' in all objectives. Finally, the case of indifference holds only if y weak dominates y' in all objectives.

2.8 Conclusion

In this chapter, we have firstly presented the necessary background for our work such as, the possibility theory, the multi-objective optimization and the Pareto approach. Also, we

have briefly presented the problem we are interested and reviewd some approaches proposed to solve it.

Besides, the chapter summarizes our contribution to resolve the CVRPTW-SD problem. In a first stage, we have proposed a new possibilistic approach to deal with CVRPTW with this problem, in which the uncertainty is represented by triangular fuzzy numbers. Consequently, to evaluate the fuzzy solutions obtained, a ranking method should be used. Thus, in the second stage, we have proposed a new pareto approach for ranking triangular fuzzy numbers in mono-objective and multi-objective cases. The implementation and experimentation of our approaches will be the focus of the next chapter.

Chapter 3

Algorithmic approach

3.1 Introduction

In this chapter, we present the implementation of the new possibilistic approach and the new Pareto relations proposed in the previous chapter in order to resolve the CVRPTW-SD which is considered in our work as a multi-objective optimization problem.

First, we start the chapter by presenting the main concepts of metaheuristics for multi-objective optimization, specially the multiobjective local search concepts and also detailing those based on dominance approach, respectively in Section 3.2, Section 3.3 and Section 3.4. Then, the next section 3.5 is dedicated to the implementation of our algorithm under ParadisEO framework. Finally, the remaining sections provides the design, the implementation and the experimental analysis of this algorithmic and also summarizes the obtained results.

3.2 Metaheuristics for multi-objective optimization

In recent decades, the use of metaheuristics on hard multi-objective optimization problems has seen a greater interest and gained a lot of popularity. Unlike exact methods, metaheuristics do not guarantee obtaining an optimal solutions but they provide a satisfactory ones in a reasonable time. Metaheuristics can be divided into two different families of algorithms:

- Single solution-based metaheuristics (S-Meta): such as local search algorithms, Tabu search, Simulated annealing. The S-Meta are based on the exploitation of the best solution found (the manipulation of a single solution during the search).

- Population-based metaheuristics (P-Meta): such as evolutionary algorithms, Ant colonies, Scatter search. The P-Meta are generally exploration search algorithms and they allow a better diversification of the search space [77].

In the multi-objective case where the solution is a Pareto optimal set, the main goal of metaheuristics becomes to obtain an approximation of this set of Pareto optimal solutions. Note that, a multi-objective metaheuristic is based on three main search components [77]:

- Fitness assignment: means that a scalar value, called fitness, is affected to each vector objective function. Its main role is to guide the search algorithm toward the Pareto optimal front for a better quality of solutions.
- Diversity preservation: Its role is to generate a diverse set of Pareto solutions by deteriorating solutions with high neighborhood.
- Elitism: the goal here is the preservation and the use of the elite solutions, which are the non-dominated or Pareto optimal solutions founded by the algorithm during the search.

After introducing the main concepts of metaheuristics, our interest in the next sections is focused on the multi-objective local search method which is the heart of our algorithmic approach.

3.3 Multi-objective local search

Local search is known as the oldest and simplest metaheuristic method in the literature and it refers to the family of S-Metaheuristics, means that it is often used for single-objective optimization [77]. But compared to the considerable number of multi-objective evolutionary algorithms proposed over the last two decades, the quantity of multi-objective local search algorithms still very low, in particularly those based on a dominance relation such as Pareto dominance.

In our work, we are interested to this type of metaheuristics. The main idea of a classical local search algorithm is that, it starts at a random initial solution. Then, at each iteration, the current solution is replaced by its best neighbor (that improves the most the objective function) until reaching a local optimum (FIG. 3.1). Therefore, the local search algorithm is mainly based on the neighborhood strategy to restrict the search space. At this level, it is necessary to recall some important concepts used by this algorithm.

Definition 3.1. ***(Neighborhood structure)***
A neighborhood structure is a function $N : X \rightarrow 2^x$ that assigns to each solution $x \in X$, a set of solutions $N(x) \subset X$. A solution x' in the neighborhood of x ($x' \in N(X)$) is called a neighbor of x (FIG.3.1).

Definition 3.2. ***(local optimum)***
For a given neighborhood function N, a solution $x \in X$ is a local optimum if it has a better quality than all its neighbors, means that, $f(x) \leq f(x')$ for all $x' \in N(x)$ (FIG. 3.1).

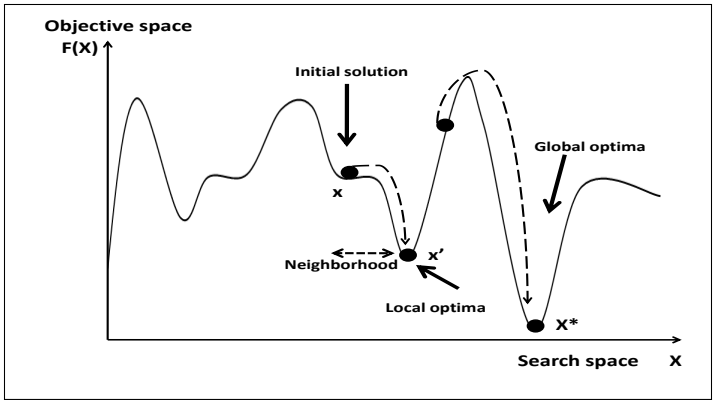

Figure 3.1: Local search

In the case of multi-objective optimization where a Pareto optimal solutions must be found, the notion of local optimum must be defined in terms of Pareto optimality.

Definition 3.3. ***(Local Pareto optimal solution)***
A solution $x \in X$ in a neighborhood N is called a local Pareto optimal solution if and only if there is no neighboring solution $x' \in N(x)$ such as x' dominates x ($x' \prec x$).

In recent years, several multi-objective local search algorithms have been proposed and are considered as an efficient methods to solve complex problems in science and industry. These methods are usually based on classical single-objective local search. Besides, the local search methods meant to find an approximation of the Pareto optimal set of a multi-objective optimization problem, belongs mainly to two categories [3]:

- Scalar approaches: This class consists in transforming a multi-objective problem into a mono-objective one. Many methods belonging to this first class exists in the literature such as aggregation methods, the weighted metrics, the ϵ-constraint methods. These methods are well described in [77].

- Pareto approaches: The goal of this second class is to defining an acceptance criterion of the Local search in terms of dominance relation such as the Pareto dominance. A review of the different algorithms belonging to this class is given in [56].

In our work, we are interested to the methods based on Pareto approaches, known as DMLS (Dominance-based Multi-objective Local Search).

3.4 DMLS algorithm

The idea of Dominance-based Multi-objective Local Search algorithm (DMLS) is to store the non dominated solutions in an archive, to explore the neighborhood of the existed solutions in the archive, and then to update this later with explored neighbors solutions, this until the archive does not improves. Also, one particularity of the DMLS methods is that they can handle a population with variable size. Many variants based on this basic idea have been proposed and investigated in the literature, such as the algorithms PLS (Pareto Local Search) [43], PAES (Pareto Archived Evolution Strategy) [42] and moRBC (Random Bit Climber using the archive for restarts) [2]. Note that these algorithms share many common concepts. A unified model generalizing this class of algorithms for multi-objective optimization, is illustrated in the figure 3.2 and the algorithm 1 taken from [4].

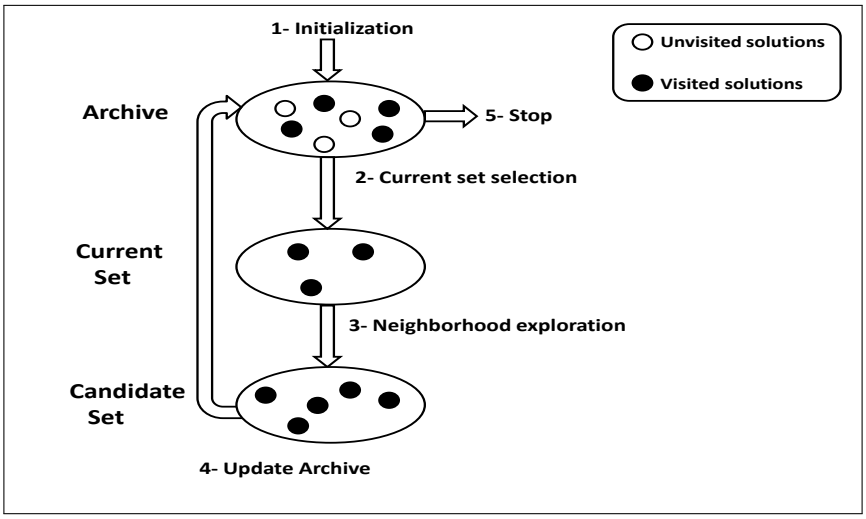

Figure 3.2: DMLS

Starting by initializing an archive with a set of non dominating solutions, at each iteration the DMLS applies the following three steps until a stopping condition is satisfied. First,

Algorithm 1 DMLS

1: Initialization. An initial set of solutions P provided by parameters, or randomly generated; initialize an archive A with the non-dominated solutions of P.
2: Current set selection. Select a set of non-visited solutions from the archive and add it to the current set $P_{current}$.
3: Neighborhood exploration. For each solution x of the current set $P_{current}$, explore the neighborhood of x according to some strategy and add the explored non-dominated solutions to the candidate set $P_{candidate}$. In the case of an exhaustive exploration of the neighborhood, x is marked as visited.
4: Update archive. Update the archive A using the solutions of the candidate set $P_{candidate}$.
5: Stopping condition. If the solutions of the archive A are all marked as visited or if another stop condition is satisfied, return the non-dominated solutions from the archive A. Stop. Otherwise, go to step 2.

a subset of unvisited solutions is selected from the archive to build the current set. Then, the neighborhood of each solution of the current set is explored to given the candidate set. Finally, the archive is updated with the solutions of the candidate set.

3.4.1 Components description

The common search components for the design of a DMLS algorithm can be defined as follows:

Representation. The representation of a solution is the starting task for designing any type of metaheuristic. A solution must be represented both in the decision space and in the objective space. In fact, the choice of this representation is considered as an important designing step. Many representations can be applied such as the representations based on real valued vectors, binary vectors, or permutation vectors.

Initialization strategy. The way to initialize a solution (or a population of solutions) is closely related to the problem to be solved and to the representation chosen. In most cases, the initial population is randomly generated. However, in some other case, the initial solution is built using a specific heuristic.

Evaluation. This step corresponds to the computation and the affectation of objective vectors for a given solution. Therefore, the evaluation associates to each feasible solution

an objective vector. Then, each objective vector element must quantify the quality of the solution considered with respect to the corresponding objective function. Note that, there is no total order but, only a partial order between the solutions. In practice, the evaluation is the most expansive step in terms of execution time.

Neighborhood structure. The definition of a neighborhood structure is the key issue for the local search efficiency. A neighborhood function N affects a set of neighboring solutions $N(x) in X$ for each solution $x \in X$. Thus, a solution $x' \in N(x)$ is called a neighbor of x. A neighbor can be generated: deterministically, randomly, or also adaptively.

Incremental evaluation. since evaluating a solution is often the most expensive step of a DMLS algorithm, an efficient way to evaluate a neighboring solution is rather appreciated, when it is possible to compute it.

Current set selection strategy. The initial phase of a DMLS iteration focuses on the selection of a set of archived solutions from which the neighborhood will be explored. Note that, if some archived solutions are marked as visited, they must be eliminated from the current set selection. Generally, in the DMLS model two main strategies can be distinguished for the current set selection:

- An exhaustive selection, where all non-visited solutions from the archive are selected.
- A partial selection, where only a subset of solutions is selected.

Neighborhood exploration strategy. At this step, some candidate solutions selected from the current set, must be generated by using a neighborhood structure. This candidate set is obtained by a repeated local transformation of every solution existed in the current set. In addition, two main strategies can be identified for the exploration task:

- An exhaustive neighborhood exploration, where the neighborhood is evaluated in a complete and deterministic way.
- A partial neighborhood exploration, where only a subset of moves are applied. For this partial strategy, different classes can be distinguished:
  - *Random neighbor:* A single random neighbor for the current solution is proposed as candidate solution for integrating the archive.
  - *First non-dominated neighbor:* For each solution x from the current set, neighbors are evaluated until a non-dominated one, with respect to x, is found.

- *First dominating neighbor:* For each solution x from the current set, neighbors are evaluated until a dominating one, with respect to x, is found. Then, all evaluated non-dominated neighbors are proposed as candidate solutions for integrating the archive.

Archive management. The archive allows to store the non-dominated solutions founded during the search process. However, updating the content of an archive using a new potential non-dominated solutions, is based mainly on the Pareto dominance relation. Two different archiving techniques can be distinguished:

- An unbounded archive : used to save the whole set of non-dominated solutions.
- A bounded archive (with fixed size): used mainly for combinatorial multi-objective problems containing an exponential number of non-dominated solutions.

Stopping condition. The most classic examples consists in stopping the algorithm after a given number of iterations or a given run times. In fact, in DMLS algorithms, the stopping criterion arises when all archived solutions are marked as visited.

3.4.2 Existing algorithms

The following table (taken from [4]) summarizes the existing algorithms as an instances of DMLS model in the literature such as; PLS-1 (Pareto Local Search) [43], PLS-2 [80], PAES (Pareto Archived Evolution Strategy) [42] and moRBC(Random Bit Climber using the archive for restarts) [2]. This table presents the algorithms as a simple instances of the unified DMLS model. Clearly, all the algorithms presented above uses the Pareto dominance relation in their original version. Then, the independent components of each algorithm are identified. For example, at each iteration, the algorithm PlS-1 select randomly one unvisited solution from the archive and it explore exhaustively its neighborhood. Besides, it has a natural stopping criterion which is verified where all the solutions are marked as visited. Whereas, at each iteration of the algorithm PAES, μ solutions are selected from the archive and then a partial neighborhood exploration is performed (up to λ random neighbors by solution). Note that λ and μ are fixed to one in the general version of PAES algorithm.

In our work, we chooses to adapt the PLS-1 algorithm to solve the CVRPTW problem with stochastic demands. Then, an entire section will be devoted to present the application of this algorithm to resolve our problem. But, we must first introduce the ParadisEO platform used to implement it.

Components	**PLS-1**	**PlS-2**	**PAES**	**moRBC**
Dominance relation	Pareto dominance	Pareto dominance	Pareto dominance	Pareto dominance
current set selection	partial 1 random solution	exhaustive all solutions	partial μ solutions	partial 1 less crowded solution
neighborhood exploration	exhaustive all neighbors	exhaustive all neighbors	partial λ random neighbors	partial 1 dominating neighbor
archiving	unbounded	unbounded	bounded adaptive hypergrid	bounded crowding distance
stopping condition	natural all solutions visited	natural all solutions visited	user-defined defined by the user	natural all solutions visited

Table 3.1: DMLS algorithms

3.5 Implementation using ParadisEO platform

To implement our PLS-1 algorithm which is based mainly on our dominance relations proposed in the previous chapter, we use the framework ParadisEO, specially the platform dedicated to multiobjective optimization: ParadisEO-MOEO [79].

3.5.1 Presentation of ParadisEO framework

ParadisEO is a free open-source object-oriented software framework implemented in C++ and available at the following URL: http://paradiseo.gforge.inria.fr. This framework is dedicated to the reusable design of metaheuristics for the resolution of continuous, discrete and combinatorial optimization problems and it offers a conceptual separation between the search algorithm and the problem to be solved. Based on EO (Evolving Objects) library, ParadisEO is portable on unix, windows and MacOS. Besides, it contains the following four interconnected modules which constitute the global platform as shown in the figure (3.3):

1. **ParadisEO-EO** for implementation of population-based metaheuristics, including evolutionary algorithms or particle swarm optimization techniques.

2. **ParadisEO-MO** for implementation of single solution-based metaheuristics, including local search, simulated annealing, Tabu search, etc.

3. **ParadisEO-MOEO** for implementation of metaheuristics dedicated to multi-objective optimization.

4. **ParadisEO-PEO** for implementation of parallel and distributed metaheuristics.

In addition, thanks to the interconnection between the different modules, the global framework ParadisEO offers a variety of components for the implementation of hybrid metaheuristics. In our work, we are interested to the module dedicated to multiobejctive optimization, that is ParadisEO-MOEO.

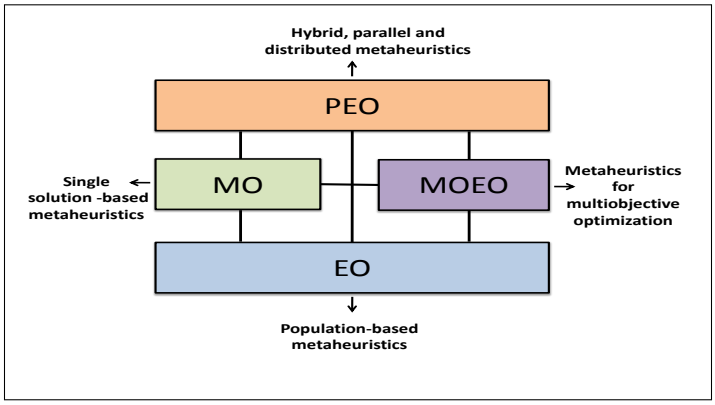

Figure 3.3: Modules of ParadisEO framework

3.5.2 Presentation of ParadisEO-MOEO module

ParadisEO-MOEO is an extension of the ParadisEO framework by including some components related to Pareto-based multiobjective optimization. It represents a white-box, flexible and modular framework for the design and the implementation of multiobjective metaheuristics. On each level of its architecture, a set of abstract and instantiated classes, corresponding to the different strategies of literature are provided. These provided classes allows a generic definition of the DMLS algorithm by a simple instantiation. Furthermore, the main goals of the ParadisEO-MOEO platform are:

- providing a clear and maximal distinction between the method and the problem representations.
- making an easy and quick development with the minimal problem-specific code.
- doing the development process in an incremental way
- allowing the programmer to rewrite as little code as possible (all coded programs might be reusable).

- being extensible, flexible and adaptable platform, means that some new elements must easily be added or modified without implicating other components, the existing components must be adaptable and extensible to offer the most possible flexibility.

Finally, by using paradisEO-MOEO platform, it is possible to build any combinatorial multiobjective application with a minimal effort.

3.6 Application to CVRPTW with stochastic demands

This section is dedicated to the experimental analysis and design to the resolution of our problem detailed in the previous chapter, the CVRPTW-SD using the the PLS-1 algorithm and the platform ParadisEO-MOEO presented above.

3.6.1 Experimental analysis

The specific components related to the resolution of our problem using the metaheuristics based on local search algorithms are presented as the following:

1. Representation: The problem considered in our work, in the general case, is a permutation vehicle routing problem, in which a solution with a size N (the number of customers) is represented as a chromosome. Each gene in the chromosome represents a specific customer or node and the order of genes that constitute the chromosome gives the order of visit of customers;

2. Initialization: The initial solutions are generated using random permutation;

3. Evaluation: This step consists simply in evaluating all the chromosomes (solutions) of our problem by computing the objective functions which are the total traveled distance D and the total tardiness time T. Note that, a chromosome is a set of n customers $\{x_1, x_2, x_3, ..., x_n\}$, each customer x_i has a fuzzy triangle demand [$\underline{d}$, $\hat{d}$, $\bar{d}$]. For each value of this triple, we compute the total traveled distance and the total tardiness time, then we obtain the results represented also by fuzzy triangles: [$\underline{D}$, $\hat{D}$, $\bar{D}$] and [$\underline{T}$, $\hat{T}$, $\bar{T}$]. After that, these fuzzy triangular solutions will be ranked in the population, by using the Pareto dominance relations defined in the previous chapter to ranking fuzzy triangular numbers. Following this ranking, the archive is updated;

4. Neighborhood: Several neighborhood operators exists in the literature for optimization problems such as the neighborhoods based on the position of elements and the neighborhoods based on the order of elements. In our problem, the order of customers

is most important than their position. Thus, the second type of neighborhood seems the most adequate to our problem, for example the operator 2-opt. This operator is often used for routing problems and offers a good results. The main idea of 2-opt is to checking a pairs of non adjacent edges in a given route, rearranging these pairs by exchanging the terminal nodes of the two edges in each pair as shown in figure 3.4 and finally computing the improvement in the route length;

5. Incremental evaluation: No incremental evaluation is considered, each neighbor will be evaluated from a zero;

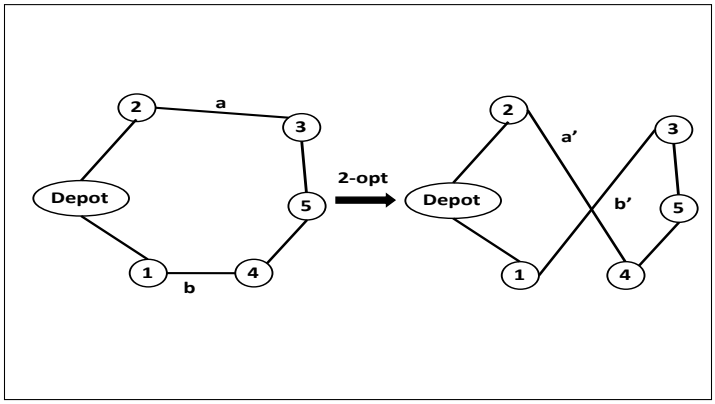

Figure 3.4: Neighborhood operator 2-opt [4]

3.6.2 Experimental design

In our work, we use the PLS-1 algorithm (shown in Algorithm 2). The main idea of this algorithm, is that at each iteration one solution is randomly selected among the unvisited solutions exists in the archive. Besides, in this algorithm the neighborhood is generated exhaustively and the archive used is an unbounded archive based on dominance between fuzzy triangles proposed in the previous chapter.

The algorithm is implemented with the version 1.3-beta of ParadisEO, specially with the ParadisEO-MOEO module under Linux. Then, to evaluate our approach, a new instances have been developed from the solomon's benchmark instances [73]. The Solomon's benchmark is a basic reference for the evaluation of several methods proposed for the CVRPTW. It includes 56 problems. Each problem contains 100 customers and a single depot. All the vehicles are homogenous, having a same capacity Q=60 and a constant speed fixed to 1 (distance unit/time unit). For that, the travel time between two customers is proportional

Algorithm 2 PLS-1

1: Initialization. initialize the archive A with a solution $x \in X$ randomly generated.
2: Current set selection. Select randomly a solution x marked as unvisited between the members of the archive.
3: Neighborhood exploration. Generate all the neighbors of x and add the explored non-dominated solutions to the candidate set $P_{candidate}$. Then, x is marked as visited.
4: Update archive. Update the archive A using the solutions of the candidate set $P_{candidate}$.
5: Stopping condition. If the solutions of the archive A are all marked as visited, return the non-dominated solutions from the archive A. Otherwise, go to step 2.

to the Euclidean distance between them. This benchmark contains six different categories of problems: C1, C2, R1, R2, RC1, RC2. The category R corresponds to a uniform customer distributions in the space, the category C corresponds to a distribution of customers in clusters, while the category RC is a combination of the two previous modes of customer distributions. Finally, the numbers 1 and 2 indicates the size of the time windows (1 for closely time windows and 2 for large time windows).

In our work, we adapted the solomon's instances to taking into account the triangular fuzzy demands by replacing the precise demands with a triplet of values generated aleatory. Table 3.2 shows an example of the instance C101 adapted with a triplet of demands.

Customer n°	X.Coord	Y.Coord	d1	d2	d3	Ready time (minute)	Due date (minute)	Service time (minute)
0	40	50	0	0	0	0	1236	0
1	45	68	6	10	12	912	967	90
2	45	70	26	28	35	825	870	90
3	42	66	5	8	15	65	146	90
4	42	68	6	8	14	727	782	90
5	42	65	7	10	11	15	67	90
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.
98	58	75	18	20	24	30	84	90
99	55	80	7	9	10	743	820	90
100	55	85	15	18	20	647	726	90

Table 3.2: Instance C101

Finally, we tested six instances: C101, C201, R101 R201, RC101, RC201, firstly in

the deterministic case by using the classic version of solomon's benchmark and secondly in uncertain case by adapting the six instances with a triangular fuzzy demands. Note that, 10 executions have been performed for each instance.

3.6.3 Experimental results

Since each instance has been executed 10 times, we obtained 10 sets of solutions that represents the Pareto optimal sets. Each solution shows the lowest distance and tardiness time, which are represented by a precise values in the deterministic case and by a triangular form (triplet of values) in the uncertain case.

To facilitate the description of our results, we choose to present only one execution for each instance. Therefore, tables 3.3 and 3.4 summarizes the results obtained for the six solomon's instances tested in the two cases: deterministic and uncertain. Each table shows the optimal solutions obtained for one execution and the elapsed time for each instance with the PLS-1 algorithm.

	Distance (km)	**Tardiness (second)**	**Elapsed Time (minute)**
C101	2333.98	503916	4.04
	2379.12	527340	
	2365.8	528431	
C201	2499.29	461297	4.71
	2398.15	501654	
	2620.91	513502	
R101	1979.98	129575	3.79
	1999.54	139873	
	2001.99	141212	
R201	2200	119412	4.32
	2313	101761	
	2317	131314	
RC101	2843	166193	3.85
	2926	179236	
	2843	186513	
RC201	2983.96	145086	4.00
	3082.96	169361	
	3054.34	147654	

Table 3.3: Deterministic case results

	Distance (Km)	**Tardiness (second)**	**Elapsed time (minute)**
C101	[2447 3162 3183]	[501545 527751 527992]	7.24
	[2438 3157 3192]	[501078 527543 528748]	
	[2421 3148 3197]	[499337 526974 529095]	
	[2414 3169 3200]	[498804 528803 529400]	
	[2429 3190 3337]	[498763 529235 536746]	
C201	[3060 3374 3443]	[476866 493845 495065]	6.23
	[3022 3284 3473]	[473540 488131 495427]	
	[3051 3392 3443]	[476507 494345 495073]	
R101	[2486 2740 2890]	[154723 166453 166603]	6.41
	[2495 2741 2883]	[154536 166042 166704]	
	[2501 2729 2879]	[154927 166389 166539]	
R201	[2124 2603 2635]	[117118 135419 135656]	6.61
	[2076 2583 2663]	[116633 134954 135943]	
RC101	[3258 3556 3681]	[166075 184797 186459]	8.02
	[3377 3514 3579]	[168445 184305 185655]	
	[3238 3649 3734]	[165209 187170 187447]	
RC201	[2976 3467 3729]	[145278 171700 175494]	7.03
	[3063 3532 3662]	[148285 174091 174443]	

Table 3.4: Uncertain case

According to these results, we clearly observe that the tardiness value is always considerable compared with the distance value. This can be due to benchmark's data, specially to customers distribution and service times , or we can also explain this by the fact that a lateness time is often imposed in many cases. Note that in our case, we can not compare our results with others because of incompatibilities between the objectives to be optimized. Figure 3.5 and 3.6 shows the Pareto front obtained for an execution of the instance C101, in both deterministic and uncertain cases. Finally, we observe that the Pareto front obtained with the new dominance relations (for the uncertain case) is presented as a set of triangles to reflect fuzzy values.

3.7 Conclusion

In this chapter, we have presented the experimental analysis, the design and the implementation of a multi-objective local search algorithm, based on dominance relations between triangular fuzzy numbers to solve our multi-objective vehicle routing problem with stochastic demands.

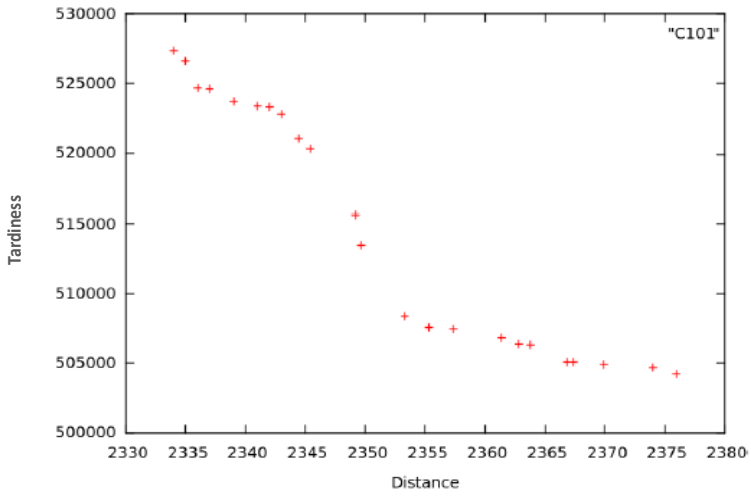

Figure 3.5: Deterministic case(C101)

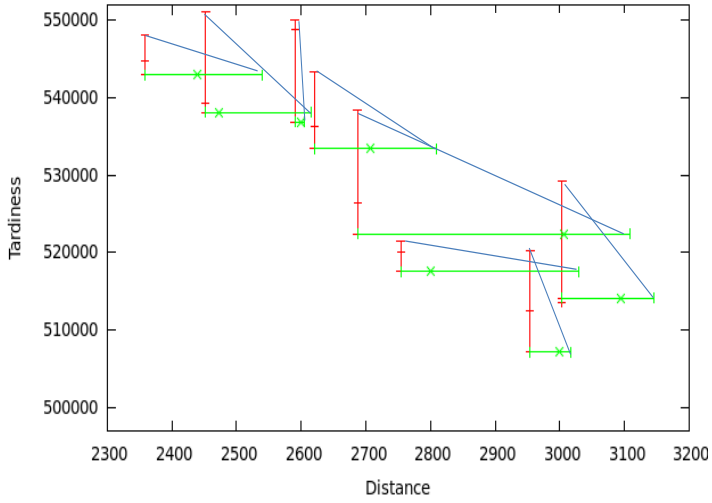

Figure 3.6: Uncertain case(C101)

Conclusion

This work has focused on the resolution of a multi-objective variant of vehicle routing problem with uncertainty in demands, namely, CVRPTW-SD (capacitated vehicle routing problem with time windows and stochastic demands). This problem includes two important lines of research: the multi-objective context and the uncertainty aspect.

However, we have chosen to treat this kind of problems because we have found that, in recent years the optimization under uncertainty is widely studied in mono-objective case, but very limited in multi-objective case. In this context, we are interested to possibility theory to express the uncertainty and to metaheuristics, specially the local search for the multi-objective optimization.

In the first stage, we have addressed the question of combining the fuzzy sets with the possibility theory. For that, we have first introduced a new possibilistic approach to solve our problem, in which the uncertainty of demands is represented by fuzzy numbers under triangular form. Consequently, the objective functions are disrupted by this form of uncertainty.

In the second stage, we have observed that the basic concept of uncertainty is the comparison of uncertain sets from the decisional space to the objective space. Since, in our case the solutions of our problem (the objective functions) are also represented by triangular fuzzy numbers, we have proposed a new Pareto dominance relations to compare these solutions, in both mono-objective and multi-objective cases. These Pareto relations have been after introduced in a DMLS (Dominance based Multi-objective Local Search) algorithm to solve the CVRPTW-SD problem.

Finally, this algorithm was developed under an object C++ framework dedicated to multi-objective optimization : ParadisEO-MOEO. This later offers a great number of components for multi-objective optimization, which can be easily used, modified and extended by new search components. The algorithm was then validated on a few VRP instances adapted from solomon's VRPTW benchmarks problems. Subsequently, we have obtained encouraging results.

This work allowed not only the resolution of a multi-objective vehicle routing problem under uncertainty, but also to better know the different characteristic of VRP, its variants and its recent solving methods, the multi-objective optimization, the theories of uncertainty specially the possibility theory, the different methods used for ranking fuzzy number and also the local search algorithms. In addition to all the theoretical knowledge acquired, i have also gained some practical knowledge, particularly relating to development environment such as: practicing linux commands, programming with C++ , and implementing under ParadisEO Platform.

Finally, while working on this memory, some areas to improve further have been arisen:

- The revaluation of our proposed solutions by using a performance assessments like the Hypervolume Indicator.
- The adaptation of our approach to treat a vehicle routing problem with both uncertain demands and uncertain times or also to treat a dynamic multi-objective variant of the problem.
- The application of our approach in a real distribution of goods.
- The application of the proposed dominance relations to solve other multi-objective combinatorial optimization problems like the Flow-shop problem.
- The improvement of our dominance relations for ranking different forms of fuzzy numbers at a time, for example dominance relations between rectangular and triangular fuzzy numbers.

All these perspectives may be lines for future works and improvements, while taking into account the requirements of this domain.

Bibliography

[1] N.R. Achuthan, L. Caccetta, and S.P. Hill. An improved branch-and-cut algorithm for the capacitated vehicle routing problem. In *Transportation Science, 37(2)*, pages 153–169, 2003.

[2] H. Aguirre and K. Tanaka. Random bit climbers on multiobjective mnk-landscapes : Effects of memory and population climbing. In *IEICE Trans. Fundamentals, 88-A(1)*, pages 334–345, 2005.

[3] L. Arnaud, E-G. Talbi, H. Jérémie, M. Salma, and J. Laetitia. On dominance-based multiobjective local search: design, implementation and experimental analysis on scheduling and traveling salesman problems. In *J Heuristics*, page 317–352, 2011.

[4] S. Asma, E-G. Talbi, M. Aïder, and A. Liefooghe. Multi-objective local search with epistemic uncertainty : Application to multi-objective vehicle routing problem with uncertain demands. In *ISOR'11*, 2011.

[5] Augerat and al. Computational results with a branch and cut code for the capacitated vehicle routing problem. In *Rapport de recherche- IMAG*, 1995.

[6] B. Baran and M. Schaerer. A multiobjective ant colony system for vehicle routing problem with time windows. In *Proceedings of the Twenty-first IASTED International Conference on Applied Informatics*, page 97–102, 2003.

[7] R. Bellman and S. Dreyfus. Applied dynamic programming. In *Princeton University Press, Princeton*, 1962.

[8] L. Bianchi, M. Birattari, M. Chiarandini, M. Manfrin, M. Mastrolilli, L. Paquete, O. Rossi-Doria, and T. Schiavinotto. Metaheuristics for the vehicle routing problem with stochastic demands. In *Parallel Problem Solving from Nature - PPSN VIII, volume 3242*, pages 450–460, 2004.

[9] M.A. Ayechew B.M. Baker. A genetic algorithm for the vehicle routing problem. In *Computers and OR 30(5)*, pages 787–800, 2003.

[10] R. Bowerman, B. Hall, and P. Calamai. A multi-objective optimization approach to urban school bus routing: Formulation and solution method. In *Transportation Research Part A 29*, page 123–197, 1995.

[11] J. Brito, J. A. Morino, and J. L. verdegay. Fuzzy optimization in vehicle routing problems. In *IFSA-EUSFLAT*, 2009.

[12] D.M. Chitty and M.L. Hernandez. A hybrid ant colony optimisation technique for dynamic vehicle routing. In *in: K. Deb et al. (Eds.), GECCO 2004, Lecture Notes in Computer Science, vol. 3102, Springer-Verlag*, page 48–59.

[13] N. Christofides, A. Mingozzi, and P. Toth. Exact algorithms for the vehicle routing problem, based on spanning tree and shortest path relaxations. In *Mathematical Programming, 20*, pages 255–282, 1981.

[14] G. Clarke and J. Wright. Scheduling of vehicles from a central depot to a number of delivery points. In *Operations Research, 12*, pages 568–581, 1964.

[15] A. Corberan, E. Fernandez, M. Laguna, and R. Marti. Heuristic solutions to the problem of routing school buses with multiple objectives. In *Journal of the Operational Research Society 53*, page 427–435.

[16] J. Cordeau and G. Laporte. Tabu search heuristics for the vehicle routing problem. In *Metaheuristic Optimization via Memory and Evolution: Tabu Search and Scatter Search*, page 145–163, 2004.

[17] J. F. Cordeau, M. Gendreau, A. Hertz, G. Laporte, and J. Sormany. New heuristics for the vehicle routing problem. logistics systems: Design and optimization. In *Metaheuristic Optimization via Memory and Evolution: Tabu Search and Scatter Search A. Langevin and D. Riopel, eds: Springer, New York*, 2005.

[18] M. Delgado, J.L. Verdegay, and M.A. Vila. A procedure for ranking fuzzy numbers using fuzzy relations. In *Fuzzy Sets and Systems. 26*, pages 49–62, 1988.

[19] D. Dubois and H. Prade. A ranking fuzy numbers in the setting of possiblity theory. In *Information Sciences. 30*, pages 183–224, 1983.

[20] D. Dubois and H. Prade. Possibility theory : Qualitative and quantitative aspects. In *In D. Gabbay and P. Smets (Eds.), Handbook on Defeasible Reasoning and Uncertainty Management Systems, 1*, pages 169–226, 1988.

[21] Y. Dumas, J. Desrosiers, and J. Soumis. A dynamic programming solution of the large-scale single vehicle dial-ride problem with time windows. In *American Journal of Mathematical and Management Science, 16*, page 301–325, 1986.

[22] Eilon and al. Distribution management. Griffin, London, 1971.

[23] N. El-Sherbeny. Resolution of a vehicle routing problem with multi-objective simulated annealing method. In *These de doctorat, Faculte Polytechnique de Mons, Mons, Belgique*, 2001.

[24] M. Fischetti and P. Toth. An additive bounding procedure for combinatorial optimization problems. In *Operations Research*, pages 319–328, 1989.

[25] M. L. Fisher. Optimal solution of vehicle routing problems using minimum ktrees. In *Operations Research*, pages 626–642, 1994.

[26] M. L. Fisher, R. Jaikumar, and L. N. V. Wassenhove. A multiplier adjustment method for the generalized assignment problem. In *Management Science, 32(9)*, pages 1095–1103, 1986.

[27] R. Fukasawa and al. Robust branch-and-cut-and-price for the capacitated vehicle routing problem. In *Mathematical Programming, 106*, pages 491–511, 2006.

[28] M.J. Geiger. Multi-criteria und fuzzy systeme in theorie und praxis, chapter a computational study of genetic crossover operators for multi-objective vehicle routing problem with soft time windows. In *Deutscher Universitats-Verlag*, page 191–207, 2003.

[29] M. Gendreau, G. Laporte, and J.-Y. Potvin. Metaheuristics for the capacitated vrp. In *In P. Toth and D. Vigo (Eds.), The Vehicle Routing Problem*, pages 129–154, 2001.

[30] M. Gendreau, G. Laporte, and R. Seguin. An exact algorithm for the vehicle routing problem with stochastic demands and customers. In *Transportation science, 29(2)*, pages 143–155, 1995.

[31] I. Giannikos. A multiobjective goal programming model for locating treatment sites and routing hazardous wastes. In *European Journal of Operational Research 104*, page 333–342, 1998.

[32] B. E. Gillett and L. R. Miller. A heuristic algorithm for the vehicle-dispatch problem. In *Operations Research, 22*, pages 340–349, 1974.

[33] Gilles Goncalves, Tienté Hsu, and Xu Jian. Vehicle routing problem with time windows and fuzzy demands: an approach based on the possibility theory. In *Int. J. Advanced Operations Management, Vol. 1*, 2009.

[34] ANTONIO GONZALEZ and MARIA AMPARO VILA. Dominance relations on fuzzy numbers. In *INFORMATION SCIENCES, 64*, pages 1–16, 1991.

[35] H.S. Hwang. An improved model for vehicle routing problem with time constraint based on genetic algorithm. In *Computer Industrial Engineering, 42*, pages 361–369, 2002.

[36] N. Jozefowiez. Modelisation et resolution approchees de problemes de tournees multiobjectif. Ph.D. thesis, Laboratoire d'Informatique Fondamentale de Lille, Universite des Sciences et Technologies de Lille, Villeneuve d'Ascq, France, 2004.

[37] N. Jozefowiez, F. Semet, and E-G. Talbi. Parallel and hybrid models for multi-objective optimization : Application to the vehicle routing problem. In *in J.J. Merelo Guervos et al.(Eds.), Parallel Problem Solving from Nature VII, Lecture Notes in Computer Science, vol. 2439, Springer-Verlag*, page 271–280.

[38] N. Jozefowiez, F. Semet, and E-G. Talbi. Multi-objective vehicle routing problems. In *In European Journal of Operational Research*, 2007.

[39] A. Kaufmann. Le probleme du classement des nombres flous en un ordre total. In *Note de travail n. 112*, 1993.

[40] A. Kaufmann and M. Gupta. Fuzzy mathematical models in engineering and managment science. In *Elseiver Science Publishers, Amsterdam, Netherlands*, 1988.

[41] T.H. Chew K.C. Tan and L.H. Lee. A hybrid multiobjective evolutionary algorithm for solving truck and trailer vehicle routing problems. In *European Journal of Operational Research 172*, page 855–885, 2006.

[42] J.D. Knowles and D. Corne. pproximating the nondominated front using the pareto archived evolution strategy. In *Evolutionary Computation*, page 149–172.

[43] M. Chiarandini L. Paquete and T. Stutzle. Pareto local optimum sets in the biobjective traveling salesman problem : An experimental study. In *In Metaheuristics for Multiobjective Optimisation, chapter 7*, page 177–199.

[44] A. Landrieu, Y. Mati, and Z. Binder. A tabu search heuristic for the single vehicle pickup and delivery problem with time windows. In *Journal of Intelligent Manufacturing. 12*, pages 497–508, 2001.

[45] G. Laporte and F. Louveaux. The vehicle routing problem with stochastic travel times. In *Transportation Science, 26(3)*, pages 161–170, 1992.

[46] G. Laporte and Y. Nobert. Exact algorithms for the vehicle routing problem. In *Annls of Discrete Mathematics, Vol.31*, pages 147–184, 1987.

[47] H.C. Lau, M. Sim, and K.M. Teo. Vehicle routing problem with time windows and a limited number of vehicles. In *European Journal of Operational Research, Volume 148*, pages 559–569(11), 1 August 2003.

[48] T-R. Lee and J-H. Ueng. A study of vehicle routing problem with load balancing. In *International Journal of Physical Distribution and Logistics Management, 29*, page 646–648.

[49] H. Li and A. Lim. Local search with annealing-like restarts to solve vrptw. In *European Journal of Operational Research, Volume 150*, pages 115–127, 1 August 2003.

[50] A. Lim, H. Li, and Rodrigues B. Solving the pickup and delivery problem with time windows using squeaky wheel optimization with local search. In *In Proceedings of 2002 Americas Conference on Information Systems*, 2002.

[51] Philipp Limbourg. Multi-objective optimization of problems with epistemic uncertainty. In *EMO'05 Proceedings of the Third international conference on Evolutionary Multi-Criterion Optimization*, pages 413–427.

[52] J. Lysgaard, A. N. Letchford, and R. W. Eglese. A new branch-and-cut algorithm for the capacitated vehicle routing problem. In *Mathematical Programming, Vol.100*, pages 423–445, 2004.

[53] SK. MD. ABU NAYEEM. Shortest path problem on a network with imprecise edge weight. In *Fuzzy Optimization and Decision Making. 4*, pages 93–312, 2005.

[54] B. Ombuki, B.J. Ross, and F. Hanshar. A multiobjective ant colony system for vehicle routing problem with time windows. In *Multi-objective genetic algorithm for vehicle routing problem with time windows, Applied Intelligence 24*, page 17–30, 2006.

[55] J. Pacheco and R. Marti. Tabu search for a multi-objective routing problem. In *Rapport technique TR09-2003, Universite de Valance*, 2003.

[56] L. Paquete. Stochastic local search algorithms for multiobjective combinatorial optimization: Methods and analysis. In *IOS Press*, 2006.

[57] Y. Park and C.Koelling. A solution of vehicle routing problems in multiple objective environment. page 121–132, 1986.

[58] Y. Park and C.Koelling. An interactive computerized algorithm for multicriteria vehicle routing problems. page 477–490, 1989.

[59] C. Prins. simple and effective evolutionary algorithm for the vrp. In *Computers and Operations Research*, pages 143–147, 2001.

[60] H.N. Psaraftis. Analysis of an o(n2) heuristic for the single vehicle many-to-many euclidean dial-a-ride problem. In *Transportation Research. 17*, pages 133–145, 1983.

[61] H.N. Psaraftis. An exact algorithm for the single vehicle many to-many immediate request dial-a-ride problem with time windows. In *Transportation Science, 17*, pages 351–357, 1983.

[62] M. Rahoual and W. Djoukhdjoukh. Metaheuristique hybride pareto pour le probleme de tournees de vehicules avec fenetres horaires. In *In Evolution Artificielle 2003*, page 380–395, 2003.

[63] M. Rahoual, B. Kitoun, M. Mabed, V. Bachelet, and F. Benameur. Multicriteria genetic algorithms for the vehicle routing problem with time windows. In *In Metaheuristic International Conference (MIC'2001)*, page 527–532, 2001.

[64] T. K. Ralphs, L. Kopman, W. R. Pulleyblank, and L. E. Trotter. On the capacitated vehicle routing problem. In *Mathematical Programming, 94*, pages 343–359, 2003.

[65] T. K. Ralphs, L. Kopman, W. R. Pulleyblank, and L.E. Trotter. On the capacitated vehicle routing problem. 2001.

[66] C. Rego and C. Roucairol. Le probleme de tournees de vehicules : Etude et resolution approchee. In *Technical Report,inria*, 1994.

[67] M. Reimann, K. Doerner, and R.F Hartl. Analyzing a unified ant system for vrps and some of its variations. In *Applications of Evolutionary Computing, Springer LNCS 2611*, pages 300–310, 2003.

[68] J. Renaud and F.F Boctor. A sweep-based algorithm for the fleet size and mix vehicle routing problem. In *European Journal of Operational Research, 140*, pages 618–628, 2002.

[69] R. Ribeiro and H.R. Lourenco. A multi-objective model for a multi-period distribution management problem. In *In : Metaheuristic International Conference 2001 (MIC'2001)*, page 91–102, 2001.

[70] Eddy Roflin. Genetic algorithm approach for capacitated vehicle routing problem with fuzzy demand. In *Jurnal Penelitian Sains*, 2010.

[71] A. Sengupta and T. K. Pal. Theory and methodology on comparing interval numbers. In *European Journal of Operational Research 127*, 2000.

[72] Chang shi Liu and Ming yong lai. The vehicle routing problem with uncertain demand at nodes. In *Transportation Research*, page 517–524, 2009.

[73] M.M. Solomon. Algorithms for the vehicle routing and scheduling problem with time window constraints. In *Operations Research. 41*, pages 469–488, 1987.

[74] W. R. Stewart and B. L. Golden. A chance-constrained approach to the stochastic vehicle routing problem. In *Paper presented at the 1980 Northeast AIDS Conference, Philadelphia*, 1980.

[75] C. Sutcliffe and J. Board. Optimal solution of a vehicle routing problem : Transporting mentally handicapped adults to an adult training centre. In *Journal of the Operational Research Society, 41*, page 61–67.

[76] E.D. Taillard. Parallel iterative search methods for vehicle routing problems. pages 661–673, 1993.

[77] E-G. Talbi. Metaheuristics : from design to implementation. In *Wiley*, 2009.

[78] E-G. Talbi, N. Jozefowiez, and F. Semet. Multi-objective vehicle routing problems. In *In European Journal of Operational Research*, 2007.

[79] E-G. Talbi, Arnaud L., Matthieu B., and Laetitia J. Paradiseo-moeo: A framework for evolutionary multi-objective optimization. In *Evolutionary Multi-Criterion Optimization (EMO 2007)*, 2007.

[80] E-G. Talbi, M. Rahoual, M.H. Mabed, and C.Dhaenens. A hybrid evolutionary approach for multicriteria optimization problems : Application to the flow shop. In *In First International Conference on Evolutionary Multi-criterion Optimization (EMO 2001), volume 1993 de Lecture Notes in computer Science*, pages 416–428, 2001.

[81] K.C. Tan, C. Cheong, and C. Goh. Solving multiobjective vehicle routing problem with stochastic demand via evolutionary computation. In *In European Journal of Operational Research, 17*, pages 813–839, 2007.

[82] K.C. Tan, T.H. Lee, Y.H. Chew, and L.H.Lee. A multiobjective evolutionary algorithm for vehicle routing problems with time windows. In *In IEEE Internatinal Conference on System, Man, and Cybernetics*, pages 361–366, 2003.

[83] C.D. Tarantilis, C.T. Kiranoudis, and V.S Vassiliadis. A list based threshold accepting metaheuristic for the heterogeneous fixed fleet vehicle routing problem. In *J. Oper. Res. Soc, 54*, pages 65–71, 2003.

[84] D. Teodorovic and P. Lucic. Intelligent vehicle routing system. In *Proceedings of the 3rd IEEE Conference on Intelligent Transportation Systems, Dearborn, MI*, pages 482–487, 2000.

[85] Dusan Teodorovic and Shinya Kikuchi. Application of fuzzy sets theory to the saving based vehicle routing algorithm. In *Civil Engineering Systems*, pages 87–93, 1991.

[86] Dusan Teodorovic and Goran Pavkovic. The fuzzy set theory approach to the vehicle routing problem when demand at nodes is uncertain. In *Fuzzy Sets and Systems, 82*, pages 307–317, 1996.

[87] F. A. Tillman. The multiple terminal delivery problem with probabilistic demands, 3. pages 192–204, 1996.

[88] P. Toth and D. Vigo. Models, relaxations and exact approaches for the capacitated vehicle routing problem. In *Discrete Applied Mathematics, 123*, pages 487–512, 2002.

[89] P. Toth and D. Vigo. The vehicle routing problem. In *Monographs on Discrete Mathematics and Applications*, 2002.

[90] R. R. Yager and D. Filev. On ranking fuzzy numbers using valuations. page 1249–1268, 1999.

[91] L. A. Zadeh. Fuzzy sets as a basis for a theory of possibility. pages 3–28, 1978.

[92] K.G. Zografos and K.N. Androustsopoulos. A heuristic algorithm for solving hazardous material distribution problems. In *European Journal of Operational Research, 152*, page 507–519, 2004.