

# Route optimization for a fleet of vehicles with temporal constraints

Algorithmics Methods and Mathematical Models

Master in Research and Innovation - UPC

Juan Francisco Martínez Vera  
`juan.francisco.martinez@est.fib.upc.edu`

January 10, 2017

### **Abstract**

Optimization problems can appear in almost all situations on life. There are specially important those ones that appears on the industry because if we can achieve an optimal solution, we will improve the efficiency of the industrial process and then get lower manufacturing cost. This improvement of the costs have an effect on the competitiveness of the companies and on the final quality of their products and there are always good news for the customers. The challenging part is that those kind of problem have a really high computational complexity therefore there are several methods to face with them. By one hand we have the, let's say, always-optimality methods that obtain the optimal solution but without taking into account the huge amount of time that it could imply. e.g. ILP<sup>1</sup>. By the other hand we have those methods that concerns about the execution time and are looking for a tradeof between the exeution time and the quality of the solution<sup>2</sup>.

In this pages will be shown this two approaches in order to get the optimal routes for a fleet of vehicles taking into account temporal constraints.

---

<sup>1</sup>Integer Linear Programming

<sup>2</sup>The distance to the optimal

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Structure of the document . . . . .	3
<b>2</b>	<b>Problem definition</b>	<b>1</b>
2.1	Description . . . . .	1
2.2	Formal definition . . . . .	1
2.2.1	Input data . . . . .	1
2.2.2	Decission variables . . . . .	2
<b>3</b>	<b>ILP model</b>	<b>3</b>
3.1	Input data . . . . .	3
<b>4</b>	<b>Meta-heuristics</b>	<b>4</b>
<b>5</b>	<b>Comparissons</b>	<b>5</b>
<b>6</b>	<b>Discussion</b>	<b>6</b>
<b>7</b>	<b>Conclusions</b>	<b>7</b>

# Chapter 1

## Introduction

Optimization problems can appear in almost all situations on life. There are specially important those ones that appears on the industry because if we can achieve an optimal solution, we will improve the efficiency of the industrial process and then get lower manufacturing cost. This improvement of the costs have an effect on the competitiveness of the companies and on the final quality of their products. The challenging part is that those kind of problem have a really high computational complexity therefore there are several methods to face with them. By one hand we have the, let's say, always-optimality methods that obtain the optimal solution but without taking into account the huge amount of time that it could imply. e.g. ILP<sup>1</sup>. By the other hand we have those methods that concerns about the execution time and are looking for a tradeof between the exeution time and the quality of the solution<sup>2</sup>.

In this project has been developed an Integer Linear Programming model in order to optimize the routes for a fleet of vehicles with temporal constraints derived from the distance between locations and from the tasks that has to be done for every location. Also have been developed different meta-heuristic strategies in order to get results more efficiently in terms of exeuction time even if there are not the optimal but close enough. Finnaly comparissons in term of execution time and solution quality have been done.

### 1.1 Structure of the document

The structure of this document is the following: The problem definition is done in chapter 2. Here it is explained the problem that is faced, which constraints is needed to take into account and what we want to optimize. At chapter 3 is explained the mathematical model that has been developed, i.e. the decision variables and also the constraints with a mathematical nomenclature. After this chapter, at chapter 4 is explained how it has been used the heuristics approach in order to face with the problem, two meta-heuristics has been used: GRASP<sup>3</sup> and BRKGA<sup>4</sup>. After perform several executions for those approaches, a comparisson in terms of time and quality of the result is done at chapter 5. Finnally a discussion about the results obtained is done at chapter 6 and the conclusions of the project are explained at chapter 7

---

<sup>1</sup>Integer Linear Programming

<sup>2</sup>The distance to the optimal

<sup>3</sup>Greedy Adaptative Search Procedure

<sup>4</sup>Biased Random-Key Genetic Algorithm

# Chapter 2

## Problem definition

### 2.1 Description

We have been asked to help one logistic company in the design of the daily routes of its fleet of vehicles. The workload for every day consist in some tasks that have to be done at a different locations, therefore, the problem is to identify the optimal routes for the minimum number of vehicles in order to perform all the work in time, i.e. starting at 8 a.m. and finalizing before 8 p.m.

For that purpose we are given a set of locations and a start location  $l_s$  where all the vehicles will start. For our point of view, the locations are not needed at all but we need the distance in terms of time between all of them. Because that we will be provided by the  $dist_{l_1, l_2}$  input data.

As is already mentioned, in every location (but not in  $l_s$ ) there is a task that has to be done, then, also the information about how many time is needed for every task is provided as input data. We are talking about  $task_l$ .

Finally one more constraint is imposed by this logistic company. Every task has to be done but can not be started at any hour of the day. There is a temporal windows for every task that describes when an specific task can be started. This windows consists on a lower boundary  $min_l$  and on a upper boundary  $max_l$ . Then the starting time can not be before  $min_l$  and after  $max_l$ .

Considering that the company has an unlimited number of vehicles, the goal of the project is to find the minimum number of vehicles needed to visit all locations and perform all tasks, and define their routes. Given two solutions with the same number of vehicles it is preferred the one in which the latest vehicle arrives at its final destination sooner.

### 2.2 Formal definition

In this section is defined the input variables, decision variables, the constraints and also the objective of this optimization problem. According the definition above, this variables are provided and describes a single instance of this problem.

#### 2.2.1 Input data

The following list describe all the input data that is provided.

- Number of locations: Is the number of locations that the vehicles have visit.

$$n \in \mathbb{N}$$

From here we can define the  $L$  set that is the locations set.

- Starting location: The location where all vehicles are placed at the beginning of times.

$$l_s \in L$$

- Distances: Are the distances in minutes from one locations to the other. It is needed in order to know the time that a route spends. Note that when  $a = b$ , then  $dist_{l_a, l_b} = 0$

$$dist_{l_a, l_b} \quad \forall l_a, l_b \in L$$

- Lower boundary for the temporal windows for every location. It indicates from when a task can be executed. If one vehicles arrives before this time, it must wait. The temporal unit are the minutes. This minimum windows can be at most 720 that is the journey time in minutes ( $8h * 60min/h = 720min$ )

$$min_l \in [0, 720] \quad \forall l \in L$$

- Upper boundary for the temporal windows for every location. It indicates until when a task can be executed. Take into account that does not have sense that  $max_l < min_l$

$$max_l \in [min_l, 720] \quad \forall l \in L$$

- Time that a task spends for every location.

$$task_l \in [0, 720], \forall l \in L$$

### 2.2.2 Decission variables

We are facing this problem in order to optimize the solution. The solution consists on the description of the routes that the fleet of vehicles have to be done. On the next lines a relation of the required decission variables are explained.

- We can represent the towns and routes as a directed weighted complete graph, then, we can describe a route of every vehicle by mean of a set of edges that describes a cycle that implmes the  $l_s$  node.

$$G = (V, A) \quad \text{where} \quad V: \text{set of nodes}, A: \text{set of directed edges} \quad (2.1)$$

$$|V| = n \quad (2.2)$$

$$cycle_i \subseteq A, \text{ s.t. } 1 \leq i \leq \text{Number of cycles} \quad (2.3)$$

$$cycle_i \cap cycle_j = \emptyset \text{ s.t. } i \neq j \quad (2.4)$$

In equation 2.3 and 2.4 are defining this decission variable and will be constraints in the ILP model at 3 chapter. They are defining that our decission variable is a set of sets that are subsets of  $A$ , i.e. are sets of directed edges. The 2.4 equation is saying that we can not have two routes that have a common route.

- One of the most important decission variables is the numbers of vehicles needed. It will be equal as the number of cycles in the graph.
- From the company requirements we know that if there are two possible solutions, we have to take this one that returns before to the  $l_s$ . Then it is as well a decission variable and depends directly on the taked routes.

The decission variables above are the basic ones. As you will see on chapter 3 more decissions variables are used for the ILP model in order to calcule good values for the times of the routes.

# Chapter 3

## ILP model

ILP<sup>1</sup> is the first of the two methods that has been used in this project. This method is always concerned on having the optimal solution without taking into account the amount of computational resources needed. This model is developed in CPLEX and the model implemented is described at the next sections.

### 3.1 Input data

From chapter 2 we already know which is the data that is provided. Now we need them in an specific shape in order to deal with them in CPLEX.

- Number of locations
- Taked routes by the vehicles. The subsets of directed edges described in the previous chapter is represented by a boolean matrix in CPLEX. The rows of the matrix are the source of the route and the columns are the target location. Therefore, if a cell is set to 1 means that this route has been taked.

---

<sup>1</sup>Integer Linear Programming

## Chapter 4

# Meta-heuristics



## Chapter 5

# Comparisons

## Chapter 6

# Discussion

## Chapter 7

# Conclusions