THE CREW SCHEDULING PROBLEM OF AN INTERURBAN PUBLIC TRANSPORT BUS COMPANY

ITXASO AMORRORTU GERVASIO

Directores de Tesis:
Unai Apaolaza Perez de Eulate
Victoria Rodríguez Chacón

Tesis dirigida a la obtención del título de
Doctor por Mondragon Unibertsitatea

Departamento de Mecánica y Producción Industrial
Mondragon Unibertsitatea
Diciembre 2017

STATEMENT OF ORIGINALITY

I hereby declare that the research recorded in this thesis and the thesis itself, were developed entirely by myself at the Industrial Management Area, Department of Mechanical and Manufacturing, at the University of Mondragon.

ACKNOWLEDGEMENTS

There are many people who take part in this thesis. Therefore, I would like to take this opportunity to express my gratitude to all who have helped me in one way or another.

First of all, thanks to Victoria Rodríguez for directing this thesis, but mainly for her friendship and personal dedication during these years.

To Unai Apaolaza, for his confidence and effort in the final stage of this thesis; and to Noemi Zabaleta for her advice at the beginning of it.

Also, I want to thank Ramón Perurena, for sharing his problem with me; and Egoitz Basoa and Ignacio Rodríguez, for helping me to solve it.

Additionally, my thanks to the entire Industrial Management Area. Especially, to Alaitz Kortabarria and Leire Markuerkiaga for their friendship, advice and support.

Lastly, I want to thank my family. Aita, ama, Olatz and Joseba, for their support, encouragement and concern during these years and Unai, for giving me the final impulse.

Eskerrik asko guztioi!

LABURPENA

Gidarien lanaren plangintza egoki batek zuzenki eragiten du garraio publikoko enpresen kostu operatiboan. Tripulazioaren plangintzaren zailtasuna bi arrazoiengatik ematen da bereziki (Esclapés 2001, Bonrostro, Yusta 2003, Ernst et al. 2004, Van den Bergh et al. 2013, Ibarra-Rojas et al. 2015, Li et al. 2015): alde batetik, gidarien plangintza beste arazo handiago baten parte da, ibilgailu eta gidarien plangintzaren arazoaren parte. Bestalde, garraio sareen arteko ezberdintasunek, enpresen baliabideen arteko ezberdintasunek edota arautegi edo lan-akordioen arteko ezberdintasunek, enpresa bakoitzarentzako soluzio partikular bat garatzea behartzen dute.

Ikerketa honen helburu nagusia "algoritmo eraginkor bat garatzea da, zeinek exekuzio denbora apropos baten, eta lehen, azken edo beste edozein bitarteko geldialditan errelebua baimenduz, hiriarteko sare baten diharduen garraio publikoko autobus konpainia batek behar duen tripulazioa minimizatzen duen".

Horrela, eta konpainia erreal baten tripulazioaren planifikazioa oinarritzat hartuta, literaturan aurkitutako bi ikerketa-hutsune aztertu dira.

Alde batetik, zenbatetan mugatu ezak eta lehen, azken edo beste edozein bitarteko geldialditan errelebuak baimentzeak daukan inpaktua aztertuko da.

Bestalde, planifikatzerakoan ezaugarri ezberdinak dituzten zerbitzuek errestrikzio ezberdinak kontsideratzea behartzen dutenean, planifikazio prozesua aztertu da. Bi prozedura aztertu dira: arazoa zerbitzuen ezaugarrien araberako planifikazio independentetan banatzea edo errestrikzio gogorrenak kontsideratuta, planifikazio bakar bat osatzea.

Ikerketaren metodologiari dagokionez, Eragiketen Ikerketako (Winston, Goldberg 2004) zazpi urratsak jarraitu dira: (1) arazoa formulatzea, (2) sistemaren behaketa, (3) arazoaren eredua formulatu, (4) eredua egiaztatzea eta aurreikuspenerako erabiltzea, (5) aukera egokia aukeratzea, (6) azterketaren emaitzak eta ondorioak aurkeztea eta (7) gomendioak ezartzea eta ebaluatzea.

Emaitzen arabera, kasu batzuetan ikertu diren bi faktoreek emaitza hobeagoak dakartzatela baieztatu da.

RESUMEN

Una planificación de los conductores adecuada impacta en el coste operacional de las empresas de transporte público. La dificultad de esta tarea se debe principalmente a dos aspectos (Esclapés 2001, Bonrostro, Yusta 2003, Ernst et al. 2004, Van den Bergh et al. 2013, Ibarra-Rojas et al. 2015, Li et al. 2015): por un lado, la planificación de los conductores es parte de un problema mayor, la planificación de los vehículos y conductores. Por otro lado, las diferencias entre las características de las redes de transporte, los recursos de las empresas, las restricciones reglamentarias o los acuerdos laborales hacen que las soluciones sean particulares para cada empresa.

El objetivo principal de esta investigación es "desarrollar un algoritmo eficiente que minimice en un tiempo de ejecución aceptable el problema de la planificación de los conductores de una compañía de autobuses de transporte de pasajeros público interurbano, permitiendo relevos ilimitados en cualquier parada de la red, es decir, al principio, final o cualquier otra parada intermedia de una línea".

De esta manera, haciendo uso de la herramienta en una empresa real, se han examinado dos lagunas de investigación encontradas en el análisis de la literatura. Por un lado, el impacto de permitir relevos ilimitados al principio, al final o en cualquier otra parada intermedia de una línea. Por otro lado, el impacto del proceso de planificación cuando las restricciones a cumplir varían según el tipo de servicio que se incluye en las jornadas. Se han analizado dos procesos: el dividir el problema en problemas independientes según las características de los servicios, o el llevar a cabo una planificación global bajo las restricciones más restrictivas.

Con respecto a la metodología de investigación, se han seguido los siete pasos de la Investigación Operativa (Winston, Goldberg 2004): (1) formular el problema, (2) observar el sistema, (3) formular un modelo del problema, (4) verificar el modelo y usarlo para la predicción, (5) seleccionar una alternativa adecuada, (6) presentar los resultados y conclusiones del estudio e (7) implementar y evaluar las recomendaciones.

Los resultados muestran que en ocasiones vale la pena considerar los factores investigados.

ABSTRACT

A proper crew scheduling impacts on the operational cost of public transport companies. The difficulty of the crew scheduling is due to two main aspects (Esclapés 2001, Bonrostro, Yusta 2003, Ernst et al. 2004, Van den Bergh et al. 2013, Ibarra-Rojas et al. 2015, Li et al. 2015): first, it is part of a larger problem, the Vehicle and Crew Scheduling Problem. Second, the differences among network features, resources of companies, regulatory restrictions or labour agreements make the solutions particular to each company.

The main objective of the present research work is *"to develop an efficient algorithm which minimizes in an acceptable execution time the Crew Scheduling Problem of an interurban passenger public transport bus company, allowing unlimited drivers' reliefs that can occur at first, last or any other intermediate stop of a line".*

So, using this tool on a real company's crew scheduling problem, two research gaps found in the analysis of the literature have been examined. On one hand, the impact of allowing unlimited drivers' reliefs that can occur at first, last or any other intermediate stop of a line. On the other hand, the impact of the scheduling procedure when restrictions vary depending on the type of service that is included in the duty. Two procedures have been studied: dividing the problem into independent problems or scheduling globally under the most limited restrictions.

Concerning the research methodology, the seven steps of Operations Research (Winston, Goldberg 2004) have been followed: (1) formulate the problem, (2) observe the system, (3) formulate a model of the problem, (4) verify the model and use the model for prediction, (5) select a suitable alternative, (6) present the results and conclusion of the study and (7) implement and evaluate the recommendations.

The results show that occasionally it is worthy to consider both investigated factors.

GLOSSARY

The first observation make throughout this research project is that the terminology used in this research area is wide. This is the reason why the aim of this section is to define some words to ensure a correct understanding of this work.

Subsequently, it is explained the meaning given to some key-concepts in this projects. The definitions proposed in this glossary are based on the definitions given by different authors (Hartley 1981, Ceder et al. 2001, Shen, Kwan 2001, Huisman 2004, Gomes et al. 2006, Weider 2007, Michaelis, Schöbel 2009, Li et al. 2015, Ceder 2016). Notice that words are arranged alphabetically.

- Block: the set of trips assigned to a vehicle for a day’s work, including the time taken to leave and return to the depot.
- Break: a rest period for a driver in a duty to get with the aim of getting a pause or having the meal.
- Cover period: it is a time span where a driver is ready to replace scheduled crews who cannot complete their complete for any reason.
- Deadhead: every trip without passengers. Usually, it happens when starting and ending the day (from a depot to the first trip and from the last trip to a depot) or between two trips during the days, when the stops where one trip ends and the other trips starts are different.
- Depot: a place where vehicles and drivers are dispatched from at the start of their work period and returned to at the end of their daily work.
- Driving period: a continuous time span where a crew is driving; it is made up of one or more consecutive pieces of work (not separated by breaks).
- Duty: the sequence of task to be performed by a driver during one day from signing on until signing off at a depot.
- Frequency (of a line): it says how often a service is offered along a line within a (given) time period.
- Line: a path in the public transport network.
- Line concept: a set of lines together with their frequencies.
- Non-driving period: a continuous time span where a crew is on duty although not driving.
- Piece of work: the work between two consecutive relief opportunities on the same vehicle.
- Relief opportunity: a time/location pair, at which a driver can be relieved.
- Relief point: it is a location where drivers can be changed.
- Timetable: it is the result of composition of the lines and trips that will be offered. It defines the starting times and points, the ending time and points, the stops and the frequency of the designed lines.
- Trip / service: a movement of a vehicle in a given path. It is the basic unit of service in the sense that each trip must be operated by a single vehicle.

- Working period: a continuous time span where a crew is on duty. It consists of driving and non-driving periods. Different working periods in a duty are separated by breaks.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF GRAPHS

1 Introduction

The Staff Scheduling Problem determines the number of employees and their work schedules, in order to minimize the labour cost while achieving a desired level of service quality (Baker 1976, Esclapés 2001, Bonrostro, Yusta 2003, Alfares 2004, Ernst et al. 2004, Van den Bergh et al. 2013, De Bruecker et al. 2015). The Staff Scheduling Problem is described in terms of resources and jobs or tasks, so tasks must be assigned to the staff (Esclapés 2001).

As analysed by Van den Bergh et al. (2013), in the last few decades, the Staff Scheduling Problem has gained importance. They provide two main reasons for arguing this increasing interest: the labour cost and the importance of satisfying employee needs.

On one side, pointing out that the labour cost is the major direct cost component for many companies, the demanding research attention could be motivated by economic considerations. Reducing labour cost a slight percent by implementing a new personnel schedule could be significantly beneficial (Van den Bergh et al. 2013).

On the other side, the importance of satisfying employee needs in staffing decisions has increased (companies offer part-time contracts or flexible work hours, for example). In consequence, managing the staff schedule has become more complicated (Van den Bergh et al. 2013).

Among other casuistic, the Staff Scheduling Problem gains significant importance for organizations that have to cover services lasting a whole working day. Hotels, police stations, hospitals, transport operators or airlines are some examples of these organizations (Esclapés 2001, Bonrostro, Yusta 2003, Ernst et al. 2004). This research project deals with the Staff Scheduling Problem on one of these organizations, on a bus transport operator.

The Crew Scheduling Problem is defined as the assignment of drivers to a bus company's regular daily operations (Gomes et al. 2006). Therefore, it refers to the Staff Scheduling Problem in a bus company.

There are three different procedures for solving the Crew Scheduling Problem: the Classic Planning Process (Freling et al. 2003, De Leone et al. 2011), the Independent Crew Scheduling (Huisman 2004) and the Integrated Vehicle and Crew Scheduling (Haase et al. 2001, Freling et al. 2003, Huisman 2004, Huisman et al. 2005). It is important that, independently of the procedure followed, in all cases duties depend on relief points (among other factors).

This is the basis of the first research gap found in the literature. The first aspect refers to the treatment of drivers' reliefs when solving the crew scheduling. It has been found that:

- Relief points are located at the beginning of lines, at the end of lines or at the depot.
- Reliefs are given at the end of the trips, that is, services are not broken in pieces of work. The same driver performs all the service.
- In some examples, vehicle changes in driver's duties are penalized. In some cases, the number of pieces of work is limited to 2 or 3.

Based on these ideas, the questions formulated are these ones:

- What if the relief points were not limited to initial or final stops of services? That is, what if a driver could be changed in an intermediate stop keeping the bus and the passengers inside the bus?
- Furthermore, what if the number of driver's reliefs was not limited?

The second research gap refers to the procedure of solving the crew scheduling when restrictions of duties vary depending on the type of service that is included in the duty. For example, this fact happens in Spain, when solving the crew scheduling of a public bus transport operator which specifically operates inter-city lines that have more and less than 50 kilometres. In this case, the operator must solve the Crew Scheduling Problem considering two regulations, the Regulation (EC) 561/2006, that refers to lines of more than 50 kilometres, and the Spanish Real Decreto 902/2007, that refers to lines of less than 50 kilometres. In this situation, there are two procedures to solve the Crew Scheduling Problem:

1. On one hand, to divide the lines that form the network considering their length (more and less than 50 kilometres) and schedule the operations separately as if they were two different networks.
2. On the other hand, to schedule all the operations jointly considering the most limited restrictions, that is, the ones that correspond to lines of more than 50 kilometres.

So, the question to answer in relation to this second research gap is this one:

- When solving the crew scheduling of an inter-city network that has lines of more and less than 50 kilometres, dividing the problem in two independent problems is better than scheduling globally under the most limited restrictions?

In order to answer the research gaps found in the literature, an appropriate scheduling tool is needed. Because of this reason, the objectives of this research project are summarized as follows:

"To develop an efficient algorithm which minimizes in an acceptable execution time the Crew Scheduling Problem of an interurban passenger public transport bus company, allowing unlimited drivers' reliefs that can occur at first, last or any other intermediate stop of a line".

"To evaluate the impact of allowing unlimited drivers' reliefs that can occur at first, last or any other intermediate stop of a line".

“To evaluate the procedure of scheduling, that is, to evaluate if scheduling separately under different restrictions is better than scheduling globally under the most limited restrictions.”

In order to achieve these objectives, a seven steps methodology of Operations Research proposed by Winston and Goldberg (2004) will be followed: (1) formulate the problem, (2) observe the system, (3) formulate a model of the problem, (4) verify the model and use the model for prediction, (5) select a suitable alternative, (6) present the results and conclusion of the study and (7) implement and evaluate recommendations.

Structure of the document

The present document is organized in 13 chapters. The first chapter deals with the introduction of the research project. The second chapter is a critical literature review. Chapter number three defines the research framework. Then, research objectives, hypothesis and methodology are described in the fourth chapter. Subsequently, the next six chapters are focused on the steps of the methodology. And finally, the last three chapters work on the conclusions, the bibliographic references and the annexes, consecutively. The content of the document is summarized in Table 1.

Table 1.Chapters and a brief description

Chapter	Title	Description
1	Introduction	Introduction to the present research project, detailing the purpose of the study and its objectives.
2	Scientific and technological background	Discussion of the literature review. Vehicle and Crew Scheduling Problems, mathematical models and heuristic solution techniques.
3	Research framework	Critical analysis of the literature review and identification of the research gaps.
4	Research objectives, hypothesis and methodology	Definition of the research objectives, the research hypothesis and the research methodology.
5	Formulation of the problem	Analysis of the main objectives of the research.
6	Observation of the system	Analysis of the key parameters in order to understand how they influence in the crew scheduling.

Chapter	Title	Description
7	Formulation of the model	Definition of the model and the solution technique developed.
8	Verification of the model	Verification of the algorithm with real-world data instances.
9	Suitable alternative selection and presentation of the results of the analysis	Definition of a method to adjust the input parameters of the algorithm and presentation of the results.
10	Implementation and evaluation (Research Results)	Analysis of the objectives that concern the project.
11	Conclusions and recommendations	Submission of contributions of this thesis and suggestion of future research.
12	References	References used within the research work.
13	Annexes	The last chapter shows the survey used, the universities surveyed and the statistical results for a better understanding of what is stated in the document.

2 Scientific and technological background

The scientific and technological background presented here is divided into three different sections.

The Crew Scheduling Problem is one of the subproblems of the Vehicle and Crew Scheduling Problem. So, the first section reviews the Vehicle and Crew Scheduling Problem in order to understand the task of crew scheduling. On one hand, the Classic Vehicle and Crew Scheduling Problem is detailed and concretely, its five optimization subproblems are described: Line Planning Problem, Timetabling Problem, Vehicle Scheduling Problem, Crew Scheduling Problem and Rostering. On the other hand, later proposals that contrast with the Classic Process are explained.

The second section focuses on the mathematical models of the Crew Scheduling Problem. The set covering formulation, the set partitioning formulation, the implicit formulation and other alternative formulations are exposed.

Finally, on the third section, heuristic solution techniques are discussed. In particular, Greedy Randomized Adaptive Search Procedure (GRASP), Tabu Search and Genetic Algorithm are analysed. After a brief explanation, a description of the improvements on The Bus Driver Scheduling Problem is given for each case.

2.1 Vehicle and Crew Scheduling Problem

Vehicle and Crew Scheduling Problem is stated as “given a set of trips within a fixed planning horizon, find a minimum cost schedule for the vehicles and the crew, so that both the vehicle and the crew schedules are feasible and mutually compatible” (Freling et al. 2003).

Addressed as a whole, this problem is not resolvable because of its huge volume and complexity (Desaulniers, Hickman 2003). So, it is divided into a set of subproblems that are solved sequentially at three stages of the planning process. On one hand, the subproblems are (1) *Line Planning Problem,* (2) *Timetabling Problem,* (3) *Vehicle Scheduling Problem,* (4) *Crew Scheduling Problem* and (5) *Rostering Problem*. On the other hand, the stages of the planning process are (1) *Strategic*, (2) *Tactical* and (3) *Operational*.

Different authors have worked on these stages (Ortuzar, Willumsen 1994, Karlaftis 2001, Desaulniers, Hickman 2003, McNally 2008, Ibarra-Rojas et al. 2015). The principal ideas proposed by these authors are summarized in Table 2.

Table 2. Stages of the planning process

Stage	Objective	Problems to solve	Time reference
Strategic	Maximizing service quality under budgetary restrictions	Network Design Problems	Long term decisions
Tactical	Maximizing service quality under budgetary restrictions	Set the frequencies of services and timetables	Seasonal decisions
Operational	Minimizing total cost to offer the proposed service	Vehicle Scheduling Crew Scheduling	Once per month Once per day

With regard to the resolution of the Vehicle and Crew Scheduling Problem, the literature contains various procedures. The traditional procedure is the *Classic Planning Process* (Freling et al. 2003, De Leone et al. 2011), where the five subproblems are solved sequentially. This procedure and the description of each subproblem are described in the next section.

More recently, different authors modify this procedure and present other alternatives. These alternatives are described onwards in the section *2.1.2 Reordering the Classic Planning Process*.

Results related to Vehicle and Crew Scheduling Problem are compiled in volumes of papers on Computer-Aided Scheduling of Public Transport as well as in other papers (Wren, Rousseau 1995, Desaulniers, Hickman 2003, Thangiah 2003, Hickman et al. 2008, Ceder 2016).

2.1.1 Classic Planning Process of the Vehicle and Crew Scheduling Problem

The Classic Planning Process of the Vehicle and Crew Scheduling Problem (De Leone et al. 2011, Freling et al. 2003, Ma et al. 2016) solves the five subproblems sequentially. It starts defining lines and timetables. Then, it solves the vehicle scheduling whose solution is the set of feasible vehicle blocks to be carried out by the buses. The next phase corresponds to the division of these vehicle blocks in some breakpoints, which usually are the *relief points*, i.e. locations where drivers can be changed. Each division of the vehicle blocks is defined as a *piece of work,* that is, the work between two consecutive *relief opportunities* on the same vehicle. A *relief opportunity* is a time/location pair, where a driver can be relieved. Next, the feasible driver's duties will be constructed by joining feasibly different pieces of work. Finally, weekly or monthly rosters will be generated. Figure 1 represents the sequence.

Figure 1. The subproblems of the Vehicle and Crew Scheduling Problem

Line Planning Problem, Timetabling Problem, Vehicle Scheduling Problem, Crew Scheduling Problem and Rostering Problem are described in the following sections. For each case, its particular objective and characteristics are described as well as some actual examples are given.

2.1.1.1 Line Planning Problem

The Line Planning Problem tackles the problem of finding lines and their corresponding frequencies in a public transport network so that a given travel demand can be satisfied taking into account at least two objectives: the transport company wishes to minimize its operating cost and the passengers request short travel times (Borndörfer, Karbstein 2012). A *line* is a path in the public transport network. The *frequency* of a line says how often a service is offered along a line within a (given) time period. A *line concept* is a set of lines together with their frequencies (Michaelis, Schöbel 2009). So, the objective of The Line Planning Problem is to find a line concept:

- which is feasible in the sense that it can be operated,
- which ensures that public transport is convenient for the passengers, and
- whose costs are small.

Taking into account these three aspects, there are two conflicting objectives when defining a line concept: maximizing the service level and minimizing costs (Schöbel 2012). In consequence, this conflict appears also in models referring to the Line Planning Problem. Schöbel (2012) classified them into two types: cost-oriented and service-oriented models. In cost oriented models, the line concept has to cover a given demand with the minimum costs (Claessens et al. 1998, Bussieck et al. 2004, Goossens 2004). In contrast, in service-oriented models a budget is given and it should be used in a way that is more advantageous for the passengers. Some examples of this appropriate use are maximizing the number of direct travellers (Bussieck et al. 1996; Bussieck 1998) (Kaspi, Raviv 2013) or minimizing the traveling time of the passengers (Scholl 2005, Pfetsch, Borndörfer 2006, Schöbel, Scholl 2006, Kaspi, Raviv 2013).

Papers reviewing models, mathematical approaches, algorithms and real applications of Line Planning Problem can be found in the literature. Some examples are described in Table 3.

Table 3. Examples of papers related to Line Planning Problem

Reference	Description
(Quak 2003)	He proposes an own algorithm to solve the Line Planning Problem: a passenger-oriented approach of the construction of a global line network and an efficient timetable. The author tests it in a case study and gets good results: • The number of line runs decreases more than 36%. • The total drive time decreases more than 42%. • The mean travel time for the passenger reduces. • The mean detour time for the passenger can be reduced by more than 51%. • The mean quadratic detour time can be reduced by more than 79%.
(Barabino 2009)	He proposes a new heuristic model and its resolution for the Transit Bus Route Network Design. For its resolution, a complex two-phase heuristic algorithm is used. Firstly, lines are characterized and secondly, the frequencies are determined, respecting geometric, operative and congruence constraints. Greedy resolution techniques are employed in the first part, while the second part is resolved through a random optimization process. The model is tested on a medium-sized city in Italy. Effective results are shown, with specific regard to the business service and a 22% reduction of the total travel time against the previous scenario.
(Laporte et al. 2011)	This article reviews some indices for the quality of a rapid transit network, as well as mathematical models and heuristics that can be used to design networks.
(Goerigk et al. 2013)	They analyse the impact of different line planning models by comparing not only typical characteristics of the line plans, but also their impact on timetables and their robustness against delays. They develop a simulation platform LinTim which enables them to compute a timetable for each line concept and to experimentally evaluate its performance under delays.
(Jaramillo-Alvarez et al. 2013)	First, they present an overview of suitable optimization models for the public transportation system. Then, they developed an optimization model which objective is to minimize transfers. Finally, results according to the proposed model are discussed.

Reference	Description
(Gattermann et al. 2014)	They propose a novel algorithmic approach to solve line planning problems. They investigate, under which conditions on the line planning model a passenger's best-response can be calculated efficiently and which properties are needed to guarantee convergence of the best-response algorithm. They also present some small computational examples.
(Laporte, Mesa 2015)	They provide an account of some of the most important results on rapid transit location planning. First, the main objectives and indices used in the assessment of rapid transit systems are described. Then, the main models and algorithms used to design such systems are reviewed. Finally, the location of stations on an already existing network is analysed.
(Martins de Sá et al. 2015)	They study a hub location problem in which the hubs to be located must form a set of inter-connecting lines. The objective is to minimize the total weighted travel time between all pairs of nodes while taking into account a budget constraint on the total set-up cost of the hub network. A mathematical programming formulation, a Benders-branch-and-cut algorithm and several heuristic algorithms, based on variable neighbourhood descent, greedy randomized adaptive search, and adaptive large neighbourhood search, are presented and compared to solve the problem. Numerical results on two sets of benchmark instances with up to 70 nodes and three lines confirm the efficiency of the proposed solution algorithms.

2.1.1.2 Timetabling Problem

Once the lines are defined the next step is to solve the Timetabling Problem. The timetable is the result of composition of the lines and trips that will be offered. The timetable defines the starting times and points, the ending time and points, the stops and the frequency of the designed lines (Ceder et al. 2001).

According to Ceder et al. (2001) there are three levels of decision problems that have to be addressed before generating the timetables:

1. Selecting the type of headway: even or uneven headways.
2. Selecting a method for setting the frequencies: maximum load or load profile.
3. Selecting one or more objective functions.

Also, Ibarra-Rojas and Rios-Solis (2012) found the following three key components to optimize timetable generation.

- Passenger transfers: travel from one point to another might imply passenger transfer between lines. Passenger waiting time is a key component. It is significant that concerning passenger transfers, synchronization has been studied in depth. Ceder et al. (2001) define the synchronization as “the simultaneous arrival of two buses”. Later, Eranki (2004) redefined the concept as “the arrival of two trips at a synchronization node with a separation time within a small time window instead of simultaneous arrivals”.
- Bus bunching: to avoid bus bunching between sub-lines or between different lines using trip separation is essential.
- Almost evenly spaced departures: a large variation in the time between consecutive trips affects the behaviour of passenger demand, even in small planning periods.

A review about the Timetabling Problem is described by Ceder (2007) , Bruno et al. (2009) and Ibarra-Rojas and Rios-Solis (2012). The Table 4 cites references mentioned by these authors on each optimization problem as well as other later works found in the literature.

Table 4. Timetabling review

Problem	References
Passenger waiting time optimization considering evenly spaced departure times	(Bookbinder, Desilets 1992, Ceder, Wilson 1986, Chakroborty et al. 1995, Daduna, Voß 1995, Cevallos, Zhao 2006, Wong et al. 2008, Daganzo, Anderson 2016, Dou et al. 2016)
Synchronization of timetabling	(Ceder et al. 2001, Eranki 2004, Zhigang et al. 2007, Ibarra-Rojas, Rios-Solis 2012, Ibarra Rojas et al. 2015, Fouilhoux et al. 2016, Ibarra-Rojas, Muñoz 2016)
Bus bunching optimization	(Adamski 1993, Berrebi et al. 2015, Daganzo 2009, Sidhu 2016)
Behaviour of passenger demand	(Bar-Yosef et al. 2013, Tirachini et al. 2013, Batarce et al. 2016, Cats et al. 2016)

2.1.1.3 Vehicle Scheduling Problem

After Timetabling, both Vehicle Scheduling Problem and also Crew Scheduling Problem correspond to a specific type of scheduling problem, called *Interval Scheduling* Problem, also known as *Fixed Job Scheduling Problem* (Kolen et al. 2007). In *Fixed Job Scheduling Problem*, not only the processing times of the *jobs* but also their starting times are given. In relation to public bus transport, *jobs* refer to trips to be carried out at the time previously defined.

In Vehicle Scheduling Problem, the lines and the timetable of each line are given and *trips* can be defined. *Trips* are minimal paths which have to be operated by the same vehicle, usually between the first stop and the last stop of a line. In consequence, Vehicle Scheduling Problem is defined as the assignment of buses to the trips to be operated in such a way that the total number of buses required to operate all the trips and the total cost are minimized (Shen, Xia 2009). In Vehicle Scheduling Problem, the aim is to construct *blocks* of consecutive trips. A *block* is the set of trips assigned to a vehicle for a day's work, including the time taken to leave and return to the *depot*. The *depot* is the place where vehicles and drivers are dispatched from at the start of their work period and returned to at the end of their daily work.

A vehicle schedule is feasible if (1) each trip is assigned to a vehicle and if (2) each vehicle performs a feasible sequence of trips (Michaelis, Schöbel 2009). Related to sequence feasibility, two trips, $trip_1$ and $trip_2$, can be served by the same bus if the arrival time at the end stop of $trip_1$ plus the time needed to drive from the end stop of $trip_1$ to the start stop of $trip_2$ is smaller than the departure time at the start stop of $trip_2$ (Michaelis, Schöbel 2009).

A literature review of Vehicle Scheduling Problem can be found in Daduna and Paixão (1995), Li et al. (2007), Bunte and Kliewer (2009), Visentini et al. (2014) and Shen et al. (2016). The principal ideas of these works are resumed in Table 5.

Table 5. Literature reviews of Vehicle Scheduling Problem

Reference	Description
(Daduna et al. 1995)	This paper shows the state of the research in vehicle scheduling and its practical application in urban mass transit companies. A brief outline of the historical developments, different restrictions for the vehicle scheduling problem and some models and their mathematical formulation are described.
(Li et al. 2007)	A hyper-heuristic based on column generation for finding near-optimal solutions is proposed. The performance of the proposed algorithm is compared with the approaches in the literature. Computational results on real-life instances are presented and discussed.
(Bunte, Kliewer 2009)	This paper discusses the model approaches for different kinds of vehicle scheduling problems and gives an up-to-date and comprehensive overview on the basis of a general problem definition. Although the authors concentrate on the presentation of model approaches, also the basic ideas of solution approaches are given.
(Visentini et al. 2014)	This paper presents a comprehensive review on methods for real-time schedule recovery in transportation services. Vehicle assignment and

Reference	Description
	rescheduling are analysed when one or more severe disruptions such as vehicle breakdowns, accidents, and delays occur. Real-time vehicle schedule recovery problems (RTVSRP) are defined and for each class, models are classified based on problem formulations and solution strategies.
(Shen et al. 2016)	This paper proposes a new VSP model based on variable trip times. Instead of being a fixed value, the duration of a trip falls into a time range. Computational results show that this model can increase the on-time performance of resulting schedules without increasing the fleet size.

2.1.1.4 Crew Scheduling Problem

The Crew Scheduling Problem is defined as the assignment of drivers to a bus company's regular daily operations (Gomes et al. 2006). It finds covering a pre-defined vehicle timetable with the minimum number of feasible duties (Gomes et al. 2006). The operations are divided into *pieces of works* and the objective is to create efficient drivers' *duties*. A *piece of work* represents the minimum portion of work that can be assigned to a driver, which is the work between two *relief points*. A *relief point* is a location where drivers can be changed. A *duty* is defined as the sequence of task to be performed by a driver during one day from signing on until signing off at a depot.

When creating duties, there are some restrictions that the scheduler have to consider, according to regulations and the labour rules of the company. These restrictions are listed below (Smith, Wren 1988, Portugal et al. 2009, Shen, Xia 2009, Chen, Niu 2012):

- Minimum and maximum duty duration, the total spreadover of a duty.
- Maximum working time: there is a limit on driving time, i.e. the total bus running time on a duty;
- Minimum and maximum every break duration, included the length of the meal break.
- Limited times to have a meal break: each driver must have his/her meal during the customary lunch or dinner time.
- Types of duties: the restrictions can differ depending on the type of duty.

Concerning the type of duty, different classifications have been proposed in the literature. Smith and Wren (1988) propose one of the earliest classifications. The authors divide duties into two main types, straight duties and split duties. Their main characteristics are these ones:

- On one hand, straight duties correspond to a normal working day, with a maximum driving time of perhaps 8 hours and a short meal break in the middle of the duty. Moreover, the authors subdivide the straight duties into early, late, day and middle duties, according to which period of the day they cover (Smith, Wren 1988).

- On the other hand, split duties have a longer spreadover, say up to 12 hours or more, with a long break in the middle of the duty; the maximum driving time is usually the same as for straight duties. In bus operations, there are morning and evening peaks in the number of buses in operation, and the main purpose of split duties is to allow one duty to cover both peaks (Smith, Wren 1988).

More recent classifications (Chen, Niu 2012, Li et al. 2015), follow also the division of Smith and Wren (1988). Chen and Niu (2012) propose three types of duties, according to when drivers start and end their duty:

- Early duty: it covers morning peak hours and the range of working time is generally from 6:00 to 13:00,
- Late duty: it covers evening peak hours, and the range of working time is generally from 15:00 to 22:00.
- Day duty: it covers two peak hours, and the range of working time is generally from 9:00 to 18:00.

In this sense, Li et al. (2015) adjust the time ranges to five types of duty:

- Early Duty: it starts early in the morning and the working time of a day that this duty covers is between 05:00 and 12:00.
- Late Duty: it starts in the afternoon and ends in the night. The working time of a day that this duty covers is between 12:00 and 20:00.
- Night Duty: it works in the late evening buses returning the buses to the garage. The working time of a day that this duty covers is between 22:00 and 05:00.
- Day Duty: it starts in the morning and ends in the afternoon. The working time of a day that this duty covers is between 08:00 and 16:00.
- Middle Duty: works during the period of the morning and the evening peaks. The working time of a day that this duty covers has two parts. One is between 05:00 and 08:00 and the other is between 16:00 and 22:00.This is a split duty.

As it is the centre of this research, mathematical approaches, algorithms and real applications of Crew Scheduling Problem are detailed later in this document, in sections *2.2 Modelling the Crew Scheduling Problem* and *2.3 Heuristic solution techniques.*

2.1.1.5 Rostering Problem

The Rostering Problem involves the assignment of duties to employees in a time horizon longer than a day, usually a week or month, taking into account factors as the rest time between consecutive days, the total rest time in a week, the work load per week or the total number of shift changes per person (morning, afternoon and night) (Shen, Xia 2009). Referring to public bus transport, after scheduling the buses and drivers, the rostering is the process of combining daily driver duties into sets of work for actual drivers on a daily, weekly or monthly basis (Shen, Xia 2009).

A focused review on the Rostering Problem can be found in the work presented by Alfares (2004) who reviews and classifies employee rostering literature published since 1990. The author presents a set covering formulation, a goal programming formulation and an implicit modelling formulation. Moreover, he classifies different solution techniques into ten categories and makes a comparison of them. The analysed solution techniques are: (1) manual solution, (2) integer programming, (3) implicit modeling, (4) decomposition, (5) goal programming, (6) working set generation, (7) LP-based solution, (8) construction and improvement, (9) metaheuristics, and (10) other methods.

Also, Ernst et al. (2004) present a review of applications, methods and models of the Rostering Problem. The authors decompose the Rostering Problem into six different modules: demand modelling, days off scheduling, shift scheduling, line of work construction, task assignment and staff assignment. The authors stablish that depending on the problem needed to be solved in each particular case, modules combination will be different and in consequence, model definition will vary. The applications areas analysed by the authors are transport systems (airlines, railways, mass transit and buses), call centres, heath care systems, protection and emergency services, civic services and utilities, venue management, financial services, hospitality and tourism, retail, and manufacturing.

Examples of papers that refer to the Rostering Problem are summarized in Table 6:

Table 6. Examples of the Rostering Problem

Reference	Description
(Esclapés 2001)	The author describes the way that ten different transport companies define their rostering. After this analysis, she concludes that it is impossible to define a common general solution to the Rostering Problem due to the wide variety of restrictions and particularities of each case. In the end, the author chooses a specific case, defines its restrictions and solves the problem with a greedy algorithm.

Reference	Description
(De Causmaecker, Vanden Berghe 2011)	This paper examines different companies and their way of scheduling is described according to three independent parameters: time, personnel and duties. From this classification, the authors identify four different types of scheduling: permanence, mobility, fluctuation and project centred planning. They point out that all of the studied cases exhibit additional properties and complexities when compared to similar cases. Furthermore, the companies' current solutions and future demands for scheduling are discussed.
(Mesquita et al. 2012)	The authors propose a methodology for planning bus driver rosters with days off patterns in public transit companies. The new methodology was tested on instances of two companies operating in Portugal. The computational experiment shows that the proposed framework can be used as a tool to evaluate and discuss different days off patterns within public transit companies.

2.1.2 Reordering the Classic Planning Process

There are some newer ideas that have changed the order of the Classic Planning Process procedure or have introduced some modifications on it. Kliewer et al. (2012) mention that in order to improve cost efficiency two concepts have been developed over the last years:

1. In order to obtain better flexibility when scheduling crews, vehicle and crew scheduling problems are tackled simultaneously.
2. In order to extend flexibility while scheduling vehicles, variable trip departure and arrival times are considered.

Moreover, other proposals have also been found in the literature review. Defining the vehicle scheduling before timetabling, the Independent Crew Scheduling, the Integrated Vehicle and Crew Scheduling and the Crew Timetabling Problem are detailed in the following sections.

2.1.2.1 Vehicle scheduling before timetabling

The drawback of the Classic Planning Process is that the main factors for the total costs, the number of vehicles and drivers needed, are determined at the end of the process

(Michaelis, Schöbel 2009). This is the reason why Michaelis and Schöbel (2009) suggest reordering the classic sequence of the planning steps: they first design the vehicle routes, then they split them to lines and finally, they calculate a (periodic) timetable. Figure 2 represents this procedure. From their point of view, this procedure has two advantages:

1. Costs can be controlled during the whole process.
2. The objective in all three steps is customer-oriented.

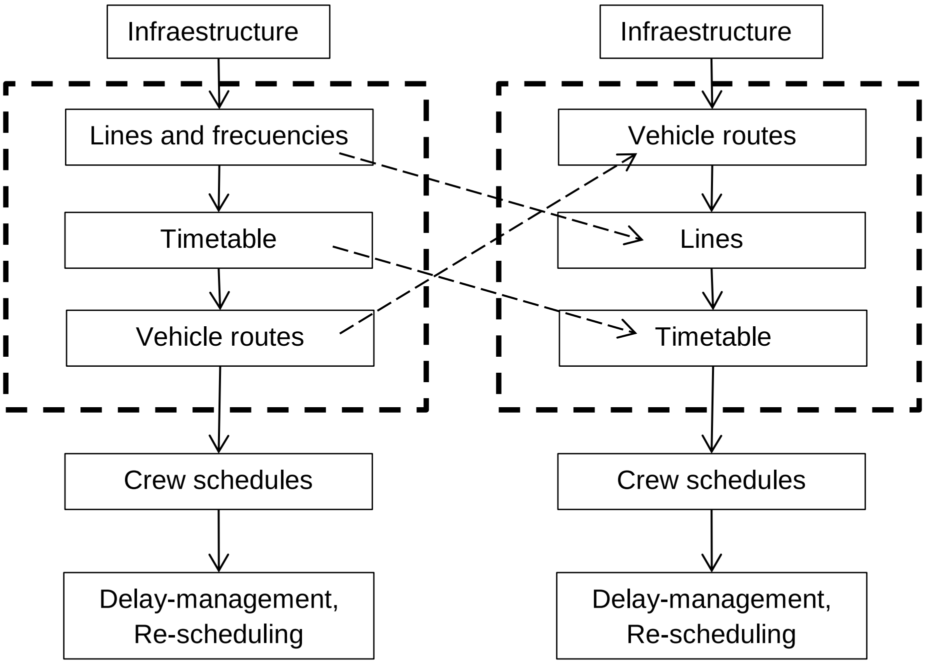


Figure 2. Structure of Vehicle Scheduling before Timetabling. (Michaelis, Schöbel 2009)

Weiszer (2011) also works on defining vehicle scheduling before timetabling. His objective is to discuss integrated planning approach employing multiobjective evolutionary algorithm. The basic structure of the model is shown in Figure 3.

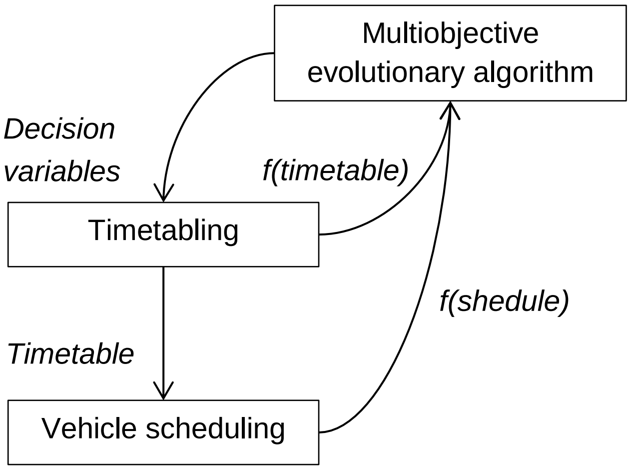


Figure 3. Structure of the integrated model. (Weiszer 2011)

As it is shown, the timetable is constructed from decision variables according to the type of timetable. If the timetable is periodical, only the offset of the first departure is needed. In other cases, the decision variables can be exact departure times or offsets from initial timetable. Once the timetable is defined, it will be the input in the vehicle scheduling. The multiobjective evolutionary algorithm, a Fast Non-dominated Sorting Genetic Algorithm NSGA-II (Deb et al. 2002), facilitates that individual objective functions are optimized simultaneously. In order to explain this algorithm, Weiszer et al. (2010) show results from a simple test case which illustrates the effectiveness of such approach. Also, Fedorko and Weiszer (2012) define how they configure the parameters of the algorithm in an extension of the study presented by Weiszer (2011).

In the same context, The Simultaneous Vehicle Scheduling and Passenger Service Problem defined by Petersen et al. (2012) is based on the Vehicle Scheduling Problem, with two modifications:

1. The trips of the timetable are allowed to be shifted by a few minutes to an earlier or later departure time. Without introducing significant changes to the timetable, this modification makes the scheduling more flexible and it can lead to a lower operational cost.
2. Secondly, a measure of passenger service is introduced for the evaluation of solutions, with the purpose of controlling the effects of this time shifting. An application from the Greater Copenhagen Area is studied and solution improvements for different problem sizes are shown.

2.1.2.2 Independent Crew Scheduling

Huisman (2004) defines The Independent Crew Scheduling as follows: given a set of trip tasks corresponding to a set of trips, and given the travelling times between each pair of locations, find a minimum cost crew schedule in which all trips are covered by exactly one duty and all duties are feasible. Vehicles are not considered in this problem. The main difference between the Independent Crew Scheduling and the Classic Planning Process is that the number of possible duties is much higher because the vehicle schedule is not solved in advance.

2.1.2.3 Integrated Vehicle and Crew Scheduling

The Integrated Vehicle and Crew Scheduling is stated as follows: “given a set of timetabled trips and a fleet of vehicles assigned to several depots, find minimum-cost vehicle blocks and valid driver duties such that each active trip is covered by one block, each active trip segment is covered by one duty, and each deadhead used in the vehicle schedule is also covered by one duty” (Freling et al. 2003, Huisman 2004).

In general, vehicle scheduling is performed before crew scheduling in the operational planning process of a public transit agency. However, a very efficient vehicle schedule may lead to a poor duty schedule or even to an infeasible crew scheduling. This fact can arise because there can be vehicles that do not pass a relief location for hours (Haase et al. 2001, Freling et al. 2003, Huisman 2004, Huisman et al. 2005).

The Classic Planning Process was first criticized by Ball et al. (1983). They create an Integrated Vehicle and Crew Scheduling at the Baltimore Metropolitan Transit Authority and they develop a mathematical model for it. However, they propose to solve this model by decomposing it into its vehicle and duty scheduling parts. So, the model is integrated, but the solution method is sequential.

Haase et al. (2001) work on Integrated Vehicle and Crew Scheduling in urban mass transit systems. For the vehicle aspect of the problem, they consider the single-depot, homogeneous fleet case. For crew aspect, they consider that all drivers are identical and can be assigned to working days of different types.

Freling et al. (2003) consider a complete integration of Vehicle and Crew Scheduling Problem and propose a mathematical formulation for it. To evaluate the effectiveness of their approach, the authors present a computational study where they make a comparison among the result obtained with the Classic Planning Process of the Vehicle and Crew Scheduling Problem, the Independent Crew Scheduling Problem and the integration they proposed. They show that the integration obtains better results. Furthermore, they conclude that the improvement depends on the relation between crew and vehicle costs: if crew cost is higher the integration becomes more attractive.

Huisman (2004) and Huisman et al. (2005) extend the work of Haase et al. (2001). The authors work on integrating vehicle and crew scheduling in multiple-depot networks. Some years later, Steinzen et al. (2010), Mesquita and Paias (2008) and Mesquita et al. (2009) present different models related to multiple-depot network.

To end with, other real-world example are described in the literature. Borndörfer et al. (2008), proposes a Lagrangean relaxation approach to solve integrated duty and vehicle scheduling problems arising in public transport. Computational results for large-scale real-world integrated vehicle and duty scheduling problems with up to 1,500 timetabled trips are reported. The authors compared their results with the results of a classical sequential approach. They conclude that integrated scheduling offers remarkable potentials in savings and drivers' satisfaction.

2.1.2.4 Crew Timetabling Problem

Gomes et al. (2006) define a new problem called Crew Timetabling Problem, which is an extension of the Crew Scheduling Problem. The aim of the Crew Timetabling Problem is to improve the results obtained in the Crew Scheduling by using driver's *non-driving periods* as *cover periods*. A *non-driving period* is a continuous time span where a crew is on duty although not driving. A *cover period* is a time span where a driver is ready to replace scheduled crews.

As in the generalized Crew Scheduling Problem, the authors separate the Crew Scheduling in a three stage process:

1. The timetables offered to the public are defined. (Timetabling)
2. Crew duties that cover those timetables are created. (Crew Scheduling)
3. Crew duties are assigned to workers on a rotating basis (Crew Rostering)

However, they incorporate a new feature in the second stage. They explain that sometimes, once the Crew Scheduling is solved, contractual rules (related to starting and finishing times and breaks) force duties to be longer so it is necessary to add some non-driving periods. Usually, in these non-driving periods drivers will carry out additional tasks, for example, act as *cover crews*, ready to replace scheduled crews who do not complete their work for any reason. The aim of The Crew Timetabling Problem is to optimize the number of cover crews available along the working day.

As a result, Gomes et al. (2006) separate the crew duties construction into two problems:

1. Crew Scheduling Problem: consists in covering a pre-defined vehicle timetable with the minimum number of (feasible) duties. The solution defines the *driving periods* for each duty. A *driving period* is a continuous time span where a crew is driving; one or more pieces of work comprises it.
2. Crew Timetabling Problem: given a Crew Scheduling Problem solution, find a duty sheet (Crew Timetabling Problem solution) so as to obtain a well-balanced cover crew profile.

The authors display three Crew Timetabling Problem duties corresponding to the same Crew Scheduling Problem duty obtained by assigning differently the start and the end of the first working period. As it is shown in Figure 4, the earliest and the latest duty starts and the earliest and the latest break starts are defined and must be completed. For each solution driving periods, non-driving periods, cover periods and breaks are represented.

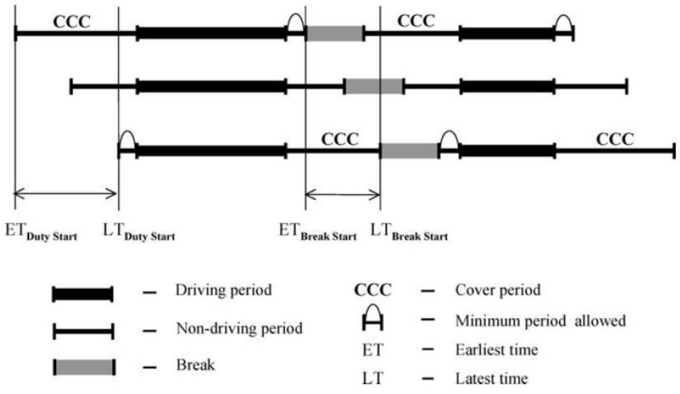


Figure 4. Shifting of working periods. (Gomes et al. 2006)

As their non-driving periods are long enough, two of the Crew Timetabling Problem duties display two cover periods each. However, this does not happened in the one represented in the second place on the Figure 4, so apparently this solution will be the worst. A Lisbon Underground case study is detailed on the article.

2.2 Modelling the Crew Scheduling Problem

Once defined the context around the Crew Scheduling Problem, this sections reviews the mathematical models defined to its resolution.

Traditionally, the Crew Scheduling Problem has been described as a single objective integer linear program and several linear programming methods have been proposed to solve it, many of them based on Dantzig's set covering model (Dantzig 1954, Darby-Dowman, Mitra 1985, Portugal et al. 2009, Kwan 2011). However, other alternatives formulations have been developed; formulations that consider more than one objective (Patrikalakis, Xerocostas 1992, Freling et al. 2003).

Taking this evolution into account, this point starts explaining the well-known set covering formulation proposed by Dantzig (Dantzig 1954) and the set partitioning formulation proposed by Darby-Dowman and Mitra (1985). Then, an implicit model proposed by Bechtold and Jacobs (1990) is described and finally other alternative formulations found in the literature are mentioned.

2.2.1 Set covering formulation

Dantzig (1954) was the first author who modelled the labour scheduling problem as a mathematical programme, he proposed a set covering model. The objective of Dantzig's set covering model is to minimize the cost of the scheduled shifts in the planning horizon. In this case, a homogeneous workforce is employed, that is, the skills of employers are not considered and the objective is to minimize the number of employees.

This formulation requires the generation of a wide range of possible duties, and the calculation of their associated cost. As mentioned by Portugal et al. (2009), one of the advantages of this formulation is that the duty generation module is separated from the duty selection module. In the first step, duties are generated taking into account the union agreements or legal operation rules of each company. In the second step, duties to be implemented will be selected, corresponding to the main objective. According to Portugal et al. (2009), only the first module (generation) will be adapted to each company, the second one (selection) will not suffer any change.

The set covering problem can be expressed as an integer linear programming problem.

$$Minimise\ \sum_{j=1}^{n} c_j x_j \qquad (2.1)$$

$$subject\ to:$$

$$\sum_{j=1}^{n} a_{ij}\, x_j \geq 1 \qquad i = 1,2,\dots,m \qquad (2.2)$$

$$x_j\ \in [0,1] \qquad j = 1,2,\dots,n \qquad (2.3)$$

- the *m* constraint corresponds to the pieces of work that must be covered by at least one duty,
- the *n* variable corresponds to the duties of the generated set of duties,
- a_{ij} is a binary value, if the *j*-th duty covers the piece of work *i* the value of a_{ij} is 1, otherwise its value is 0,
- the cost associated to duty *j* is c_j and the variable x_j is a binary variable whose value is 1 if the *j*-th day is part of the final solution, 0 in other cases,

- restrictions indicate that each piece of work must be covered by at least one duty.

Other formulations based on the set covering formulation have been proposed for solving the crew-scheduling problem. Smith and Wren (1988) add constraints relative to types of duties. Ramalhinho et al. (2001) present a model based on the set covering formulation considering multiple objective functions. Their objective functions take into account the service quality, usually in conflict with the cost minimization. The authors define the quality of the service in terms of: the number of pieces of work not covered, the unfitness value measured by the amount of over covered pieces of work, the total number of duties, the total number of duties with only one piece of work and the number of vehicle changes. Huisman (2004) works on the Integrated and Dynamic Vehicle and Crew Scheduling, he solves the crew scheduling based on the set covering model. Kunkun and Shen (2016) proposes a new grey shift evaluation approach, which contains eight parameters.

2.2.2 Set partitioning formulation

If the inequalities of the set covering problem are replaced by equalities implies that each piece of work is done exactly once and the problem becomes a set partitioning problem (Darby-Dowman, Mitra 1985). Works focused on bus transportation and based on the set partitioning formulations have been proposed by different authors (Lee et al. 2008, Mesquita, Paias 2008, Crawford et al. 2009, Dong et al. 2011, Fuegenschuh 2011, Shen et al. 2013).

2.2.3 Implicit formulation

The size of the resulting integer model with set covering formulation has been found to be very large to solve optimally in most practical applications. With the aim to solve this problem, Bechtold and Jacobs (1990) propose an implicit model which requires a smaller number of variables and uses a novel idea for modelling break placements implicitly. Nevertheless, this formulation was not applicable to the general form due to its assumptions: (1) the system operates less than 24 hours daily; (2) planning periods are equal in length; (3) each shift has a single break; (4) the break duration is identical for all shifts; (5) the break duration is one or more periods; (6) each shift has a single break window associated with its work span; (7) breaks should start and end during the shifts; (8) no extraordinary break window overlap exists; and (9) no understaffing is allowed.

Aykin (1996) presented a new approach in which a set of break variables is introduced for every shift-break type combination, requiring substantially smaller number of variables and computer memory than the model of Bechtold and Jacobs (1990). Aykin (2000) compared both models solving 220 problems taking into account two criteria: model reliability and solution time. The results show that Aykin’s formulation offers better model reliability and requires on overage one-third of the time needed in the formulation of Bechtold and Jacobs.

More recently, with the aim of solving the nine restrictions mentioned before, Addou and Soumis (2007) proposed a Bechtold-Jacobs generalized model for shift scheduling with extraordinary overlap. Rekik et al. (2010) propose an implicit shift scheduling with multiple fractional breaks and work stretch duration restrictions (a fractional break is a break that can be divided into sub-breaks, so that the sum of them is equal to the total length of the break needed).

2.2.4 Other formulations

Easton and Rossin (1996) develop a stochastic goal programming model for a mix workforce with different skills. Penalties for overstaffing and understaffing exist and labour demands are estimated from probability distributions.

Patrikalakis and Xerocostas (1992) present the first mathematical formulation for the integrated Vehicle and Crew Scheduling Problem. Later, Freling et al. (2003) consider a complete integration of Vehicle and Crew Scheduling Problem and propose a mathematical formulation for it. Extensions to the multi-depot case have been developed by Huisman et al. (2005) and De Groot and Huisman (2008).

De Leone et al. (2011) propose a mathematical formulation for the Crew Scheduling Problem under special constraints imposed by Italian transport regulations.

The objective of the model presented by Ma et al. (2017) is to minimize a bus company’s total costs, which include standard and additional salary payments to drivers, the cost of potentially unfair working time and the cost of the average total working time and idle time.

2.3 Heuristic solution techniques

Bartholdi (1981) shows that the Staff Scheduling Problem is NP-complete. In consequence, several approaches have been proposed to solve it based on different solution methods. Ernst et al. (2004) classifies solution methods used to solve Staff Scheduling and Rostering Problems into six categories: demand modelling, artificial intelligence, constraint programming restrictions, metaheuristics, linear programming and integer linear programming. In the case of Alfares (2004), the author classifies solution techniques into ten categories: manual solution, integer programming, implicit modelling, decomposition, goal programming, working set generation, linear programming based solution, construction and improvement, metaheuristics and other methods.

In most practical cases the amount of data and the corresponding execution time, make integer programming approaches unviable for obtaining the optimal solution (Alfares 2004). In contrast, the higher speed of metaheuristic procedures facilitates the execution. However, metaheuristics have a disadvantage, they do not assure the global optimal (Alfares 2004). Surveys about heuristics and metaheuristics can be found in the literature (Blum et al. 2011, Boussaïd et al. 2013, Pirlot 1996, Malczewski, Rinner 2015).

The literature shows that metaheuristic are used also in the The Crew Scheduling Problem. In fact, there are some papers that compare the use of different metaheuristics in this problem. Ramalhinho et al. (1998) describe metaheuristics for solving real crew scheduling problems in a public transport bus company. The authors focus on the GRASP, Tabu Search and Genetic Algorithm. Dos Santos and Mateus (2007) solve crew scheduling problem with an exact column generation algorithm improved by metaheuristics. The metaheuristics used in this case are GRASP and Genetic Algorithm. All heuristics improved the column generation algorithm. Finally, Lopez et al. (2009) have experimentally analysed a total of 12 techniques grouped into four categories: bioinspired methods, metaheuristics, constraint-based methods and market-based methods. Lopez et al. (2009) analyse GRASP, Tabu Search, Genetic Algorithm and Ant colony optimization.

As it can be seen, in these comparative papers, the most common used metaheuristics are GRASP, Tabu Search and Genetic Algorithm. Centred on this idea, these three metaheuristics are described in next lines. Besides, for each metaheuristic the developments of The Crew Scheduling Problem are explained.

2.3.1 Greedy Randomized Adaptive Search Procedure (GRASP)

2.3.1.1 Description

Greedy Randomized Adaptive Search Procedure (GRASP) was proposed by Feo & Resende (1995). GRASP is a procedure that combines constructive methods with local search. The pseudocode of the GRASP algorithm is presented in Figure 5:

```
GRASP
Repeat
        Build a feasible solution using a randomized greedy heuristic
        Apply a local search starting from the built solution
until the stopping criterion is satisfied
return the best solution met
```

Figure 5. Template for the GRASP algorithm. (Boussaïd et al. 2013)

In the first phase, the construction, a feasible solution is built using a randomized iterative function. In each iteration, all the elements are ordered in a candidate list taking into account the benefit of selecting each element. This list of candidates is called *The Restricted Candidate List* (RCL). After that, an element is randomly selected and included in the solution. It is said that GRASP is adaptive because the benefit associated with every element is recalculated after the selection of the candidate in every iteration to reflect the influence of the last selection. Using this technique different solutions are obtained in each GRASP iteration.

The second phase corresponds to a local search. Basically, the local search consists in moving from a solution to another one in its neighbourhood according to some well-defined rules. For instance, if X is a set of binary vectors and $x \in X$, a neighbourhood $V(x)$ of x can be defined as the set of all solutions $x \in X$ obtained from x by flipping a single coordinate from 0 to 1 or conversely. In other words, it means that the neighbours of x are the solutions obtained from x by an elementary move (Pirlot 1996). So, in this phase of the GRASP, it is necessary to define the neighbourhood and look for other solutions.

The construction and the local search procedures are repeated until the stopping criterion is satisfied and the best overall solution is returned as the result.

Festa and Resende (2009a) and Festa and Resende (2009b) contain detailed bibliographies of the GRASP literature from 1989 to 2008. Festa and Resende (2011) give an overview of GRASP describing its basic components and developments to the basic procedure.

Successful implementation techniques, alternative solution construction mechanisms and techniques to speed up the search, discussion about implementation strategies or hybridizations with other metaheuristics are shown in other works (Resende 2008, Resende, Ribeiro 2010, Resende, Ribeiro 2014, Duarte et al. 2015).

2.3.1.2 GRASP for The Crew Scheduling Problem

Ramalhinho et al. (1998) propose a two phases GRASP based on the Set Covering Problem. In the construction phase a feasible solution is generated by adding working days to the current solution. Specifically, in each iteration the algorithm compares the non–assigned pieces of work when a working day is added. Corresponding to the local search, they propose an exchange neighbourhood, i.e. remove a column of the solution and add a new column that covers at least an uncovered line.

Lopez et al. (2009) formulate the problem using the service approach. For the constructive phase all the services are ordered by departure time. A list of all possible drivers that can complete a service is made and the cost of assigning the service to that driver is calculated. A driver from the list is randomly selected and all of the variables are updated. This procedure is repeated until all the services have a driver assigned to them. The local search attempts to reduce the cost by reducing the number of assigned drivers. With this aim, the algorithm looks for drivers that perform only one service and finds out if another driver is able to do it.

Vaquerizo (2010) proposes a method that determines optimal shifts in transport, for any local public transport companies. This problem has been solved using GRASP and a Bio-inspired Algorithm.

De Leone (2011) proposes a mathematical formulation for The Bus Driver Scheduling Problem under special constraints imposed by transport rules in Italy. However, this model can only be usefully applied to small or medium size problem instances. For large instances, a GRASP procedure is proposed. Results are reported for a set of real-word problems and a comparison is made with an exact method. Moreover, the computational results obtained with the GRASP procedure are compared with the results obtained by Huisman et al. (2005).

Vaquerizo et al. (2012) propose another method of two stages. In a first stage, a GRASP algorithm is used to generate a viable solution. In a second stage, this preliminary solution is adjusted, in order to obtain an optimal one, by using a Scatter Search Algorithm.

2.3.2 Tabu Search

2.3.2.1 Description

The Tabu Search was first proposed by Glover (1986). Tabu Search can be described as a local search technique guided by the use of adaptive or flexible memory structure which allows the exploration of different regions of the search space (short term memory) and the intensification of the search in promising areas (long term memory) (Lopez et al. 2009).

Referring to the exploration, short – term memory, a *tabu list* records the last encountered solutions (or some attributes of them) and prohibits these solutions (or solutions containing one of these attributes) from being visited again, as long as they are in the list. If the length of the tabu list is low, the search will concentrate on small areas of the search space. On the contrary, a high length forbids revisiting a higher number of solutions so this fact forces the exploration of larger regions. The structure of a simple Tabu Search algorithm is presented in Figure 6.

```
TABU SEARCH
Choose, at random, an initial solution $s$ in the search space
$TabuList \leftarrow \emptyset$
while the stopping criteria is not satisfied do
        Select the best solution $s' \in N(s) \setminus TabuList$
        $s \leftarrow s'$
        Update TabuList
End
return the best solution met
```

Figure 6. Algorithm for the simple tabu search method (Boussaïd et al. 2013)

Longer-term memory processes are incorporated in order to intensify and diversify the search. The *aspiration criteria*, that is a set of rules, greatly improves the search process. For example, a move forbidden by the tabu list which leads to a solution better than all those visited by the search in the preceding iterations, does not have any reason to be prohibited. So, the aspiration criteria is used to override tabu restrictions and allow this type of move.

Some works related to tabu search can be found in the literature (Melián et al. 2003, Glover et al. 2007, Adamuthe, Bichkar 2012).

2.3.2.2 Tabu Search for The Crew Scheduling Problem

Cavique et al. (1999) present a Tabu Search method for crew scheduling. The method constructs an initial schedule using a method called *a run-cutting algorithm* which covers a pre-defined timetable. After this, *Tabu-crew*, a refining algorithm, is applied to reduce the number of duties. *Tabu-crew* iteratively removes some ‘inefficient duties’ (with only a single piece of work) as well as their ‘adjacent’ duties from the current solution, and then applies the run-cutting algorithm to construct duties to cover the broken schedule. Cavique et al. (1999) also present another Tabu Search algorithm called *Run-ejection*, which considers compound moves. It results from a succession of steps in which an element is assigned to a new state, with the outcome of ejecting some other element from its current state. The ejected element is then assigned to a new state, sequentially ejecting another element, and so forth, creating a chain of such operations.

Shen and Kwan (2001) focus their work on solving the driver scheduling problem using a constructive approach with *windows of relief opportunities*, defined as a time/location pair, at which a driver can be relieved. According to the author, the tabu search approach proposed is very fast and achieves results comparable to those based on mathematical programming approaches.

2.3.3 Genetic Algorithm

2.3.3.1 Description

A Genetic Algorithm (Holland 1975, Goldberg 1989, Mitchell 1998) is a computational model base on a biological evolution. The main difference between Genetic Algorithm and other local search heuristics is that Genetic Algorithm is based on a population of solutions instead of on a single solution (Pirlot 1996).

Genetic Algorithms are able to develop complex structures (Forrest, Mitchell 1993). These structures, called individuals or chromosomes, represent solutions to problems. First, a population of individuals is created randomly or by processes that use prior knowledge of the specific problem. Then, all the individuals are evaluated and ranked according to their relative fitness. After that, the new generations are created by the operators. The main operators are selection, crossover and mutation (Dias et al. 2002):

1. Selection: taking into account the relative fitness, the selection is the fact of choosing the individuals. There are a number of different selection schemes but the most commonly used are tournament selection, rank-based selection, and proportionate reproduction (Goldberg, Deb 1991). Selection mechanisms in genetic algorithms are defined by different authors (Bäck, Hoffmeister 1991, Goldberg, Deb 1991, Thierens, Goldberg 1994, Blickle, Thiele 1995, Miller, Goldberg 1996, Zhang, Kim 2000).
2. Crossover: two or more of the selected individuals are chosen to mate and it combines pieces of them to form new and possibly better individuals. According to Goldberg (2007) the combination of crossover and selection operators is the core of the innovation process that explains the success of Genetic Algorithms. The crossover operation is analysed in many papers (Ortiz-Boyer et al. 2007, Abido, Elazouni 2012, Ramli et al. 2013, Yuan et al. 2013, Chuang et al. 2015).
3. Mutation: performs changes in a single individual. Mutation randomly searches in the neighbourhood of a particular solution. Its role is very important to

guarantee that the whole search space is reachable. Many studies analyse the mutation operation (Abido, Elazouni 2012, Woodward, Swan 2012, Ramli et al. 2013).

Once the operation is carried out, the new individuals are evaluated and they will replace the worst individuals of the current population. This process is repeated until a satisfactory solution is achieved or a pre-fixed number of generations are performed. There are many ways of implementing this idea, a prevalent implementation is the one introduced by Holland (Holland 1975, Goldberg 1989, Forrest 1993, Forrest 1996) and it is illustrated in Figure 7:

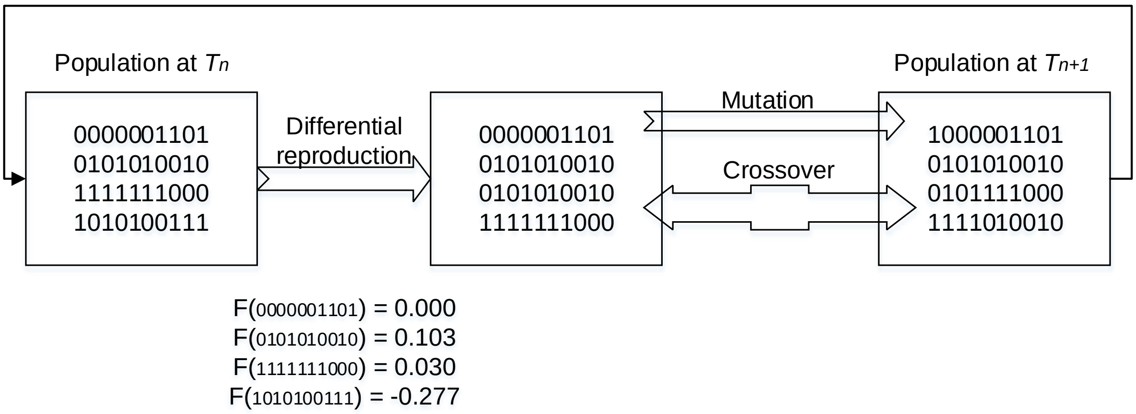


Figure 7. Genetic algorithm overview. (Forrest 1996)

In the figure a population of four individuals is shown. The fitness value of each individual is calculated according to the function $F(x, y) = yx^2 - x^4$. On the basis of these fitness values, the selection phase assigns the first individual (0000001101) one copy, the second (0101010010) two copies, the third (1111111000) one copy, and the fourth (1010100111) zero copies. After the selection, the genetic operators are applied probabilistically; the first individual has its first bit mutated from 0 to 1, and a crossover operation combines the last two individuals into two new ones. The resulting population is shown in the box labeled $T_{(N+1)}$ (Forrest 1996).

It is also outstanding that the literature distinguish between a traditional Genetic Algorithm and a Hybrid Genetic Algorithm. The traditional Genetic Algorithm usually uses a binary coding alphabet and the crossover and mutation operators do not include any knowledge about the structure and domain of the problem. In a Hybrid Genetic Algorithm problem specific knowledge is taken into account in the operators as well as in the coding scheme. Hybrid Genetic Algorithms are usually applied in difficult problems (Davis 1991, Dias et al. 2001).

Many reviews related to Genetic Algorithm have been developed last decades (Koza 1992, Forrest 1993, Alander 1995, Forrest 1996, Goldberg et al. 1997, Mitchell 1998, Alander 2000a, Alander 2000b, Gen, Cheng 2000, Aytug et al. 2003, Chaudhry, Luo 2005, Melián et al. 2009, Malhotra et al. 2011, Younas 2014).

2.3.3.2 A Genetic Algorithm for The Crew Scheduling Problem

Regarding the use of a Genetic Algorithm for solving the set covering problem, different authors have work on it (Al-Sultan et al. 1996, Beasley, Chu 1996). In the case of the set partitioning, Levine (1996) define a parallel Genetic Algorithm for the set partitioning.

Referring particularly to The Crew Scheduling Problem other works have also been presented. Wren and Wren (1995) and Kwan et al. (1999) developed a simple genetic algorithm the Crew Scheduling Problem. Later, Cai and LI (2000) work on a genetic algorithm for scheduling crews of mixed skills and Kwan et al. (2001) centred their work in driver's reliefs. More recently, Shen et al. (2013) present computational results based on 11 real-world crew scheduling problems in China. They assure that their algorithm works fast and that it achieves results close to the lower bounds obtained by a standard linear programming.

One of the differences found in the literature refers to what the chromosome structure represents. Song et al. (2015) define that structure of GA could be divided into two classes: one is the shift-based chromosome structure and the other is called the piece-based chromosome structure.

A frequently used representation is a binary vector with fixed length, where each gene is associated with a duty, and its value is either one or zero, according to the presence or absence of the duty in the solution (Beasley, Chu 1996, Levine 1996). Figure 8 shows an example of this representation. If for a given problem there are 14 candidate duties and a solution is created by duties 1,2, 6, 7, and 13 the corresponding chromosome will have a length of 14 bits with the following structure:

1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	1	0	0	0	1	1	0	0	0	0	0	1	0

Figure 8. A binary chromosome representation

Dias et al. (2001) and Dias et al. (2002) define a pieces of work coding scheme, associating the duties and the set of pieces of work. Each piece of work corresponds to a gene of the chromosome, and each gene is characterized by the duty that covers the piece of work in that particular solution. Figure 9 represents a solution for a problem with 14 pieces of work.

1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	5	3	2	1	4	1	0	2	3	5	2	4	0

Figure 9. Chromosome representation. (Dias et al. 2001)

Shen et al. (2013) propose a hybrid Genetic Algorithm whose chromosome length may vary adaptively during the process (Figure 10). First, its initial value is chosen as the lower bound of the number of shifts to be used in an unachievable optimal solution. Next, the hybrid Genetic Algorithm with such a short chromosome length is employed to find a feasible schedule. During the Genetic Algorithm process, the adaptation on chromosome lengths is realized by genetic operations of crossover and mutation with removal and replenishment strategies aided by a simple greedy algorithm. If a feasible schedule cannot be found when the Genetic Algorithm's termination condition is met, the Genetic Algorithm will restart adding one more gene. The process is repeated until a feasible solution is found.

j_1	j_2	j_3	…	j_{k-1}	j_k	j_{k+1}	…	j_{L-2}	j_{L-1}	j_L

Figure 10. Variable length chromosome representation. (Shen et al. 2013)

The given target L not only defines the length of a chromosome, but also indicates the number of shifts to be included in a schedule. Therefore, each gene in a chromosome must be assigned a sole value, that is, the same shifts are not allowed to be selected into a chromosome; otherwise, the number of shifts contained in the corresponding schedule is less than the given target. Note that a schedule represented by above-defined chromosome may be infeasible, especially under a tight target such as L_o.

3 Research framework

Once analysed the Vehicle and Crew Scheduling Problem and its subproblems, examined the models of the Crew Scheduling Problem and identified some heuristic solution techniques, the present chapter deals with the critical analysis of the literature review and the identification of the research gaps.

3.1 Research gap 1: drivers' reliefs at intermediate stops

As defined in the literature review, the resolution of the Crew Scheduling Problem has been tackled in different ways.

In the case of the Classic Planning Process (Freling et al. 2003, De Leone et al. 2011), the crew scheduling is solved after the vehicle scheduling. Firstly, lines and timetables of each line are given and *trips* that have to be operated by the same bus are defined. The aim is to construct *blocks*, that is, the set of trips assigned to vehicles for a day's work. Secondly, in order to solve the Crew Scheduling Problem, the *blocks* are divided into *pieces of work* and the objective is to create efficient drivers' *duties*. A *piece of work* represents the minimum portion of work that can be assigned to a driver, which is the work between two *relief points*. A *relief point* is a location where drivers can be changed.

In the Independent Crew Scheduling (Huisman 2004) vehicles are not considered. The main difference of the Independent Crew Scheduling in contrast to the Classic Planning Process is that the set of possible duties is much larger in the Independent Crew Scheduling. This fact happens because the vehicle scheduling is not solved in advance, so, vehicle blocks do not exist and in consequence, do not limit the crew scheduling.

Finally, in the Integrated Vehicle and Crew Scheduling (Haase et al. 2001, Freling et al. 2003, Huisman 2004, Huisman et al. 2005), both problems are considered jointly. The aim is to find minimum-cost vehicle blocks and valid driver duties such that each active trip is covered by one block, each active trip segment is covered by one duty, and each deadhead used in the vehicle schedule is also covered by one duty.

However, although the processes are different, it is observed that in all cases, (1) duties depend on pieces of work and (2) pieces of work depend on relief points. So, it could be affirmed that duties depend on relief points. But, how do relief points affect the crew scheduling? How are the locations and restrictions of reliefs analysed in the bibliography?

With the aim of answering these questions, various works of the literature have been examined in greater detail, in particular, the ones that focus on the Bus Crew Scheduling Problem and emphasize how they have addressed this issue (Table 7**).**

Table 7. Literature detailing content about reliefs

Reference	Content found in relation to reliefs
Smith and Wren (1988)	In this work relief opportunities depend on vehicle blocks and breaking trips is not allowed when defining pieces of work. Besides, the number of buses a driver drives is limited. Duties are classified into three categories: two-bus duties, three bus duties and one-bus duties.
Ramalhinho et al. (1998)	They highlight that one aspect that measures the quality of the service is the number of vehicle changes. The change of a vehicle driver can disrupt the operational functioning of the company, and cause complains from the drivers. Therefore, some companies are mostly worried about minimizing the number of changes.
Haase et al. (2001)	Each line of the bus system is defined by a start location, an end location, and several intermediate stops where passengers can get on and off the bus. Among the starts and the end locations, some locations are considered as relief points. The depot is also a relief point. Trips has to be serviced by exactly one bus, no bus exchanges are possible at intermediate stops. From a driver point of view, there are two types of duties: a first type may impose that a driver remains on the same bus all along its duty while a second type may allow up to two bus changes during the same duty.
Kwan et al. (2001)	In this paper relief opportunities depend on vehicle blocks and breaking trips is not allowed when defining pieces of work.
Ramalhinho et al. (2001)	In this work relief opportunities depend on vehicle blocks and breaking trips is not allowed when defining pieces of work.
Shen and Kwan (2001)	Relief opportunities depend on vehicle blocks and breaking trips is not allowed when defining pieces of work. Each duty consists of a sign-on activity, a sign-off activity, and a set of spells, which are continuous vehicle work to be operated by a driver without break. Each spell includes at least two ROs, the first and the last ROs are called active relief opportunities (AROs for short). Besides, the first and the last RO of a spell are called departure-ARO and arrival-ARO respectively. A pair of time and depot location for signing on is called a sign-on-ARO, which is treated as an arrival-ARO because it is followed by a departure-ARO in a spell. Similarly for signing off, a sign-off-ARO is treated as a departure-ARO.

Reference	Content found in relation to reliefs
Dias et al. (2002)	The authors define different techniques to reduce the number of feasible duties when solving the set covering. A proposal focuses on the examination of all the possible pieces of work to determine whether two consecutive pieces of work can be combined together. As each duty must start, finish or have a break at a relief point, the reduction of the number of these points will lead to a lower number of potential feasible duties. Referring to the genetic algorithm, they propose a pieces of work coding scheme, associating the two fundamental types of information contained in a solution: the duties and the set of pieces of work. Each piece of work corresponds to a gene of the chromosome. The genes are ordered by bus and for each bus they are ordered by time. So, as buses are joined to trips, it means that pieces of work are composed of non-divided trips.
Freling et al. (2003)	The authors remark that duties consist of a number of pieces with a given maximum number of pieces. In practice this maximum is very often equal to 2 or 3. The mathematical formulation they propose is a combination of the quasi-assignment formulation for the vehicle scheduling problem, and the set partitioning formulation for crew scheduling. They used data from the RET, the public transport company in Rotterdam, in the computational analysis. Concerning the reliefs, the restrictions that they have taken into account, are as follows: a driver can only be relieved by another driver at the start or end of a trip or at the depot and continuous attendance is required. This implies that a trip always corresponds to exactly one trip task and the number of relief points is equal to twice the number of trips.
Huisman (2004)	Based on Freling et al. (2003), the author considers the same scenario, that is, the public transport company in Rotterdam. In this work, duties are divided into two main types: full-time and part-time duties. Full-time duties consist of two pieces and one break in between at the central station, which is the only relief location.
Huisman et al. (2005)	Among the restrictions considered in this work, there are two related to reliefs: (1) there is continuous attendance, i.e., there is always a driver on site if the bus is outside the depot and (2) changeovers are allowed.

Reference	Content found in relation to reliefs
	When testing their algorithms with real-world data instances, they define that a driver can only be relieved by another driver at the start or end of a trip at certain specified locations or at the depot.
De Groot and Huisman (2008)	Each trip has fixed starting and ending times, and can be assigned to a vehicle and a crew member from a certain set of depots. Besides, a driver can only be relieved by another driver at the start or end of a trip at certain specified locations or at the depot.
Mesquita et al. (2009)	In this paper, each task, the minimum portion of work that can be assigned to a driver, corresponds to a deadhead trip followed by a trip and the crew duties can start (end) at a depot or at an end location of a trip. Besides, a driver may start/end its duty at the end location of each task or at a depot. They have considered that a changeover might occur if and only if the location, where the driver leaves the first vehicle, is the same where he picks up the second one.
Portugal et al. (2009)	Some of possible desired rules are when evaluating the solution are these ones: total (Normal Work + Extra Work) duty duration, vehicle changes number, relief's out of the depot and duty type percentage. So, it means that relief are penalized when evaluating the solution.
Shen and Xia (2009)	In this paper, the following rules are enforced on duties: (1) A driver signs on and signs off at the same depot; (2) Three types of duties are allowed, which are split duties (longer than standard duties, e.g. up to 12 h or more, plus a long break in the middle), straight duties (with a meal break in the middle), and single-spell duties (containing a short straight run, say 2–5 h, without meal breaks). For a split duty, the longest gap between two consecutive spells is not shorter than a given minimum length of time. (3) Meal break (i.e. the time allowance between two spells of work for the driver to travel to the canteen, have a meal and then take over the next bus) is not shorter than a specified minimum length of time, and the meal is taken within a given lunch or dinner time range; (4) A driver is usually restricted to a single bus, at most two buses, and must have knowledge of the type of the buses; (5) A bus is operated by one or two drivers, at most three, but only when necessary;

Reference	Content found in relation to reliefs
	(6) There is a limit on driving time, i.e. the total bus running time on a duty; (7) There is a maximum working time for a duty. The calculation of the working time of a duty (e.g. whether non-driving time is partially or fully counted as working time) depends on the operator and the type of the duty. Taking into account these rules, a new method for constructing an initial schedule is designed, which consists of the following steps: 1. Build feasible duties with one bus. 2. Build feasible duties with two buses. 3. Pair uncovered spells.
De Leone et al. (2011)	In this work relief opportunities depend on vehicle blocks and breaking trips is not allowed when defining pieces of work.
Chen and Niu (2012)	The crew scheduling problem in this study is to arrange the work plan for crew in a single bus line. The kind of urban bus line is a circle line, which means that the starting station and terminal are the same bus station. The authors propose three types of duties, according to when drivers start and end their duty; early duty, late duty and day duty.
Shen et al. (2013)	In this paper, a shift starts with the crew signing on at depot. The crew then works on one spell (called a single spell shift) or n (normally between 2 and 4) spells with breaks in between (called an n-spells shift) until signing off at depot. Breaking trips is not allowed when defining pieces of work.
Li et al. (2015)	In this work relief opportunities depend on vehicle blocks and breaking trips is not allowed when defining pieces of work.
(Shen et al. 2017)	In this work relief opportunities depend on vehicle blocks and breaking trips is not allowed when defining pieces of work.

After this analysis, the main ideas obtained are these ones:

- In all cases, relief points are located at the beginning of lines, at the end of lines or at the depot.
- In all cases, reliefs are given at the end of the trips, that is, services are not broken in pieces of work. The same driver performs all the service.
- In many of the examples, vehicle changes in driver's duties are penalized. In some cases, the number of pieces of work is limited to 2 or 3.

So, the questions formulated on this first research gap are these ones:

- What if the relief points were not limited to initial or final stops of services? That is, what if a driver could be changed in an intermediate stop maintaining the bus and the passengers inside the bus?
- Also, what if the number of driver's reliefs was not limited?

3.2 Research gap 2: crew scheduling of inter-city lines that have more and less than 50 kilometres

Regulation (EC) 561/2006 provides a common set of European Union (EU) rules for maximum daily and fortnightly driving times, as well as daily and weekly minimum rest periods for all drivers of road haulage and passenger transport vehicles, subject to specified exceptions and national derogations. The question is that one of these exceptions refers to vehicles used for the carriage of passengers on regular services where the route covered does not exceed 50 kilometres. For this case, national law has to be considered. In the case of Spain, the Real Decreto 902/2007 is the law needed to follow.

This project focuses on public bus transport operators which operate inter-city lines that have more than 50 kilometres as well as inter-city lines that have less than 50 kilometres. In this casuistic, the operator must solve the Crew Scheduling Problem considering both regulations, the Regulation (EC) 561/2006 and the Spanish Real Decreto 902/2007. When facing the crew scheduling, the principal differences of these regulations correspond to two aspects: the maximum working time limitations and the break restrictions. Besides, it is significant that schedules that combine lines of more and less than 50 kilometres must respect the regulation of lines of more than 50 kilometres.

So, if duties will have different restrictions, depending on which lines are joined, it means that the scheduling procedure carried out could affect the solution. That is, in this situation it seems that the scheduling could be carried out in two different ways, and the solutions would be different:

1. Perform two separate plans: divide the operations of more and less of 50 kilometres and schedule them as independent networks. The advantage of this option is that the management is easier: any change on the regulation, network or services would affect only to a part of the scheduling.

2. Create schedules jointly considering the toughest restrictions.

Joined to these procedures, De Groot and Huisman (2008) highlight that, in the literature on vehicle and crew scheduling, not much attention has been paid to the problem of splitting up large instances into several smaller ones such that a good overall solution is obtained. According to the authors, if a real-world instance has to be solved and it seems to be too large for the algorithm to solve it, the problem is just split up into several smaller instances, divided according to some logical rules. Then, the algorithm is used to solve those smaller instances and the results are combined such that there is an overall solution. However, it is remarkable that different divisions can result in a completely different final solution. Besides, even if the algorithm itself provides an optimal solution, optimality for the overall problem is likely to be lost.

With the aim of determining if this problem or a similar one has been tackle before, the literature has been examined in greater detail. We have focused on papers that classify duties into different types and specifically, two factors have been analysed:

1. The classification of duties itself, that is, the parameters considered and in particular, if differences in breaks and working times could happened among duties.
2. The procedure, that is, if the fact that dividing instances has been analysed.

Table 8. Literature detailing content about duty types

Reference	Content found in relation to duty types
Smith and Wren (1988)	In this paper, five types of two- and three-bus duty can be created: (1) early duties, taking buses out of the garage before the morning peak; (2) day duties, which either take over early buses on the road from another crew, or occasionally take a later-starting bus out of the garage; (3) late duties, working on late-evening buses returning to the garage; (4) middle duties, which either finish at the garage shortly after the evening peak, or hand over to late duties on the road; and (5) split duties. Feasible pieces of work are created first and then, they are combined respecting the restrictions defined, e.g. earliest sign-on time, maximum spreadover, and so on. The procedure of dividing instances is not analysed.
Haase et al. (2001)	They consider two types of duty differing mainly by the number of pieces of work a duty contains. Feasible pieces of work are created first and then, they are combined respecting the restrictions defined: the length of a duty, the length of each piece of work, the length of a break and the total work time in a duty. The procedure of dividing instances is not analysed.
Freling et al. (2003)	In this case, there are four different types of duties classified by the number of pieces of work they have. The procedure of dividing instances is not analysed.

Reference	Content found in relation to duty types
Huisman (2004)	In this paper, minimum and maximum values are given for the number of pieces of work contained in a duty. The factors considered when creating duties are the maximum length of the duty, the minimum length of a break in the duty and a maximum working time in the duty. The procedure of dividing instances is not analysed.
Chen and Niu (2012)	According to the different peak hours, authors divide crew duties into three types: early, day, and late mode. The considered work rules involve time shift, work intensity, and duty type compatibility constraints. The procedure of dividing instances is not analysed.
Li et al. (2015)	In this paper, the authors adjust the time ranges to five types of duty: early duty, late duty, night duty, day duty and middle duty The procedure of dividing instances is not analysed.

As it can be seen, in relation to the type of duty different classifications have been proposed. In general, duties are classified according to three aspects:

1. When the duty is performed.
2. The number of pieces of work that form the duty.
3. Whether they are straight or split duties.

So, it is observed that the division of duties considering differences in breaks or working hours and the procedure to solve our problem has not been threatened in the bibliography. Thus, the question to answer concerning this second research gap is this one:

- When solving the crew scheduling of an inter-city network that has lines of more and less than 50 kilometres, dividing the problem into two independent problems is better than scheduling globally under the most limited restrictions?

4 Research objectives, hypothesis and methodology

Once research gaps have been detailed, this chapter describes the research objectives, the research hypothesis and the research methodology of this thesis.

It is outstanding that this research project focuses on public bus transport operators which operate inter-city lines that have more than 50 kilometres as well as inter-city lines that have less than 50 kilometres. In this casuistic, the operator must solve the Crew Scheduling Problem considering both regulations, the Regulation (EC) 561/2006 and the Spanish Real Decreto 902/2007. So, the objectives as well as the hypothesis refer to this scenario.

4.1 Research objectives

To work on the research gaps defined in the previous chapter, a scheduling tool that answers to the new scenario is needed. This is the reason why the first objective of this research thesis is stated as follows:

Objective 1: *"To develop an efficient algorithm which minimizes in an acceptable execution time the Crew Scheduling Problem of an interurban passenger public transport bus company, allowing unlimited drivers' reliefs that can occur at first, last or any other intermediate stop of a line".*

Once the scheduling tool is developed, two other objectives can be addressed:

Objective 2: *"To evaluate the impact of allowing unlimited drivers' reliefs that can occur at first, last or any other intermediate stop of a line."*

Objective 3: *"To evaluate the procedure of scheduling, that is, to evaluate if scheduling separately under different restrictions is better than scheduling globally under the most limited restrictions."*

4.2 Research hypothesis

With the aim of testing the research framework developed through Chapter 3, we have defined the following hypotheses:

Hypothesis A: *allowing unlimited drivers' reliefs that can occur at first, last or any other intermediate stop of a line might improve the result of the crew scheduling. The quality of the solution will be measured in number of duties.*

As explained in Section 3.1, no matter the procedure used, the Classic Planning Process (Freling et al. 2003, De Leone et al. 2011), the Crew Scheduling (Huisman 2004) or the Integrated Vehicle and Crew Scheduling (Haase et al. 2001, Freling et al. 2003, Huisman 2004, Huisman et al. 2005), the solution of the crew scheduling depends on relief points. Besides, it has been found that the treatment of relief points has been similar in the literature: (1) relief points are located at the beginning of lines, at the end of lines or at the depot; (2) reliefs are given at the end of the trips; and (3) vehicle changes in driver's duties are penalized.

This first hypothesis will allow us to answer to the questions related to the first research gap: "what if the relief points were not limited to initial or final stops of services?" and "what if the number of driver's reliefs was not limited?"

Hypothesis B: *referring to the scheduling procedure, scheduling separately under different restrictions achieves a better solution than scheduling globally under the toughest restrictions. The quality of the solution will be measured in number of duties.*

As defined in Section 3.2, not much attention has been paid to the problem of splitting up large instances into several smaller ones when a problem seems to be too large for the algorithm to solve it (de Groot, Huisman 2008). However, different divisions can result in a completely different final solution and besides, even if the algorithm itself provides an optimal solution, optimality for the overall problem is likely to be lost.

When tackling our problem, the procedure to carry out could vary. This second hypothesis will allow us to choose between these two procedures:

1. Perform two separate plans: divide the operations of more and less of 50 kilometres and schedule them as independent networks.
2. Create schedules jointly considering the toughest restrictions.

4.3 Research methodology

4.3.1 Description of the methodology

According to the methodology, as authors in this research area do (Peters et al. 2007, Portugal et al. 2009, Li et al. 2015), the seven steps of the methodology proposed by Winston and Goldberg (Winston, Goldberg 2004) to face operations research related to problems will be followed. These seven steps are these ones: (1) formulate the problem, (2) observe the system, (3) formulate a model of the problem, (4) verify the model and use the model for prediction, (5) select a suitable alternative, (6) present the results and conclusion of the study and (7) implement and evaluate recommendations. Each step is defined in next lines.

1. *Formulate the problem:*

The first step involves defining the main objectives of the research and the parts of the organization that must be studied before the problem can be solved.

2. *Observe the system:*

The second step is to collect data to estimate the value of parameters that affect the problem of the organization. These approximations are used to develop and evaluate a model in steps 3 and 4.

3. *Formulate a model*:

The third step discusses the advantages and disadvantages of traditional models and selecting one or developing a new one which solves the restrictions of the current problem.

4. *Verify the model*:

The aim of the fourth step is to determine if the model developed in step 3 is an accurate representation of reality. It involves testing the model with sets of instances that represent different planning situations or problems.

5. *Select a suitable alternative:*

Given a model and a set of alternatives, the next step is to choose the alternative that best meets the organization’s objectives.

6. *Present the results of the analysis:*

In this step, the model and recommendation from step 5 are presented to the decision making individual or group. In some situations, the analyst may find that the organization does not approve the recommendation. This may result from incorrect definition of the organization’s problems or from failure to involve the decision maker from the start of the project. In this case, the operations researcher should return to step 1, 2, or 3.

7. *Implement and evaluate:*

If the organization has accepted the study, then the researcher aids in implementing the recommendations. The system must be constantly monitored (and updated dynamically as the environment changes) to ensure that the recommendations enable the organization to meet its objectives.

4.3.2 Implementation of the methodology in this research project

But how we will implement the seven steps described in the previous section in this particular project? Next, the work carried out in each step is detailed.

1. *Formulation of the problem:*

This step details further the research objectives described in Section 4.1 and analyses the meaning of the words that describe this objectives.

2. *Observation of the system:*

Two different fields are analysed in this second step of the methodology: the components of a transport network and the restrictions related to drivers' duties.

In the analysis of these fields, a two stage-procedure is carried out. Firstly, the information is collected and secondly, the observations based on this information are made.

3. *Formulation of a model:*

As new restrictions are included in the Crew Scheduling Problem, a new formulation of the model is developed at this point.

Besides, in this third step the GRASP is selected as the solution technique and the algorithm has been developed. It is remarkable that, when executing the algorithm, there are some parameters related to how it works that affect the result. These parameters are defined in this step.

4. *Verification of the model:*

With the aim of determining if the model is an accurate representation of reality, a real enterprise is selected. First, it is validated that the chosen scenario fits the characteristics of the model. Then, as it is appropriate, real values are assigned to the data needed in the algorithm and finally, its validation is carried out.

5. *Suitable alternative selection:*

The method used to adjust the parameters of the algorithm is the Design of Experiments (DoE). Lye (2005) defined DoE as a method for systematically applying statistics to experimentation. Montgomery (2008) defined it as a "*series of tests in which purposeful changes are made to the input variables of a process or system so that one may observe and identify the reasons for these changes in the output response*".

6. *Presentation of the results of the analysis:*

As result of the previous analysis, the best values for the input parameters are defined.

7. *Implementation and evaluation:*

The objectives that concern the project are tested and evaluated in different transport networks.

5 Formulation of the problem

The first step involves defining the main objectives of the research and the parts of the organization that must be studied in order to solve the problem. With this aim, an analysis in detail of each of the three research objectives described in Section 4.1 is developed.

5.1 Analysis of Objective 1

The first objective of this research project is stated as follows:

> **Objective 1:** *"To develop an efficient algorithm which minimizes in an acceptable execution time the Crew Scheduling Problem of an interurban passenger public transport bus company, allowing unlimited drivers' reliefs that can occur at first, last or any other intermediate stop of a line".*

The analysis of the keywords of this first objective is detailed below:

Efficient algorithm. *What is considered "efficient"?*

Aykin (2000) compares his two models taking into account two factors: model reliability and solution time. In Portugal et al. (2009), the criterion for evaluating their model is the capacity to generate real and useful schedules that can be implemented without many manual adjustments or modifications. Authors consider the following measures of the quality of the model: simplicity, solution quality and applicability.

Based on these works, the criteria to consider when developing the scheduling tool are these ones: model reliability and simplicity, solution time, solution quality and applicability. Next, these criteria are described:

1. Model reliability and simplicity: all the restrictions of the model will compulsorily be respected. So, the solution will be totally feasible and no manual adjustments will be necessary once the solution is obtained.
2. Solution time: as well as getting a good solution, getting it fast will be a key aspect when executing the tool.
3. Solution quality: the solution quality (the goodness of the solution) will be measured in duties. The aim will be to solve the problem with the minimum number of drivers' duties.
4. Applicability: the tool has to answer to a real problem of a real company; that means that it is able to answer to a problem of large instances.

Acceptable execution time. *What is considered “acceptable”?*

The execution time will be considered “acceptable” if the tool solves the problem faster than a traffic manager would solve it manually. That means that the company saves time when using the scheduling tool to complete the crew scheduling.

Interurban passenger public transport bus company. *What are the factors to study in an interurban transport network?*

When The Crew Scheduling Problem is analysed in the literature, the same aspects are described. These aspects are divided into two categories:

1. Information about the network: services, depots, relief points and pieces of work.
2. Restrictions related to driver’s duties, specifically, the information related to: (1) straight and split duties, (2) breaks in lines with more than 50 kilometres and (3) breaks in lines with less than 50 kilometres.

Unlimited drivers’ reliefs. *What do unlimited drivers’ reliefs mean?*

According to the scenario described in the Research Framework, these are the aspects to be considered:

1. The reliefs will not be penalized in the algorithm.
2. The driver can change the bus as often as necessary.
3. The crew scheduling will be solved independently of the vehicle scheduling. It assumes that drivers will access relief points on buses or cars, anyway to respect the crew scheduling.

Reliefs at first, last or any other intermediate stop of a line. *What does the fact that reliefs can occur also at intermediate points mean?*

Based on the content described in the Research Framework, these are aspects to consider:

1. Reliefs can occur at any time of the duty and at any point of the network. However, with the aim of reducing the number of pieces of work and in consequence, the execution time, it will be necessary to analyse whether all stops are equally suitable to have a relief.
2. It is possible to have a relief within a service, i.e. at an intermediate stop, without being necessary to adjust them to idle-times between services.
3. In order not to disturb the passenger, one new restriction must be taken into account: during the execution of a service, only a driver relief can be carried out. This aspect will be crucial in our model.

5.2 Analysis of Objective 2

The second objective of this research project is stated as follows:

> **Objective 2:** *"To evaluate the impact of allowing unlimited drivers' reliefs that can occur at first, last or any other intermediate stop of a line."*

Afterward the analysis of the keywords of this second objective is summarized:

Evaluate the impact. *How can the impact be measured?*

In order to assess this impact, a comparison of two different schedules will take place. On one hand, drivers' reliefs at intermediate stops will not be allowed. On the other hand, reliefs at intermediate stops will be permitted.

When evaluating the solutions obtained, these two parameters will be compared: the number of duties obtained and the execution time required to solve the problem.

5.3 Analysis of Objective 3

The third objective of this research project is stated as follows:

> **Objective 3:** ***"****To evaluate the procedure of scheduling, that is, to evaluate if scheduling separately under different restrictions is better than scheduling globally under the most limited restrictions."*

Evaluate the procedure of scheduling. *How can the planning process be assessed?*

In order to evaluate this impact, a comparison of two different scheduling procedures will be considered:

1. Perform two separate plans: divide the operations of more and less of 50 kilometres and schedule them as independent networks. The advantage of this option is that the management is easier: any change on the regulation, network or services would affect only to a part of the scheduling.

2. Create schedules jointly considering the toughest restrictions.

As in the previous goal, when evaluating the solutions obtained, these two parameters will be compared: the number of duties obtained and the execution time required to solve the problem.

6 Observation of the system

Once the problem is formulated, the next step is to observe the system. The aim is to examine the key parameters in order to understand how they influence in the crew scheduling.

Two different fields have been analysed in this second step of the methodology: the components of a transport network and the restrictions related to driver's duties.

In the analysis of these fields, a two stage-procedure has been carried out. Firstly, the *information* is collected and secondly, the *observations* are made. The following descriptions explain the difference between both concepts:

1. Collecting information: it includes descriptive data collection by the observation of the system.
2. Observation: it is considered a contribution based on the information collected. The objective is to identify how the information affects the crew scheduling activity.

6.1 Description of the network

6.1.1 Information: services, depots, relief points and pieces of work

Services

Lines which have different services during a day comprise the transport network. As regards public transport, lines and timetables are defined in advance. In consequence, services can be defined by a departure time, a departure place and an arrival time and an arrival place. The Figure 11 shows the information needed to identify the services of a line.

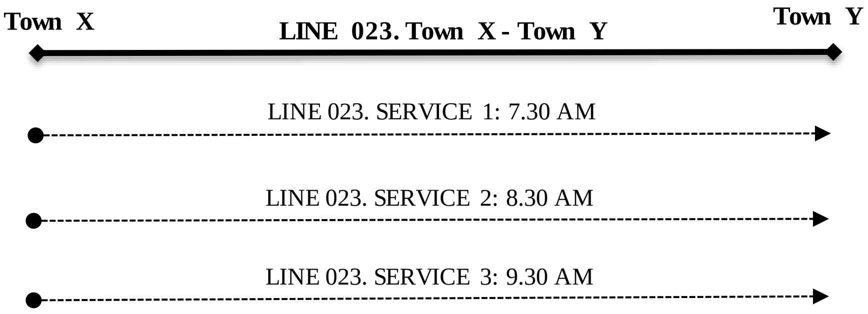


Figure 11. Representation of services of a line

Depots

There are *n* depots in the network. Depots are garages where drivers start and end their workday.

Relief points

Reliefs can occur at any time of the duty and at any point of the network. Furthermore, it is possible to have a relief within a service at an intermediate stop. So, it is not necessary to adjust reliefs to idle-times between services.

Nevertheless, to avoid disturbing passengers, one restriction has taken into account: only one driver relief can be carried out during the execution of a service.

Pieces of work

As defined by Shen and Kwan (2001), a piece of work is “an indivisible period of driving work, between two windows of relief opportunities”. Considering this definition, services have to be divided into pieces of work taking into account the relief points. The result is a set of pieces of work $SPOW = \{POW_1, POW_2, \ldots, POW_n\}$ where each POW_i represents a piece of work that must be covered and n represents the total number of pieces of work. Lines, services, relief points and pieces of work are represented in Figure 12.

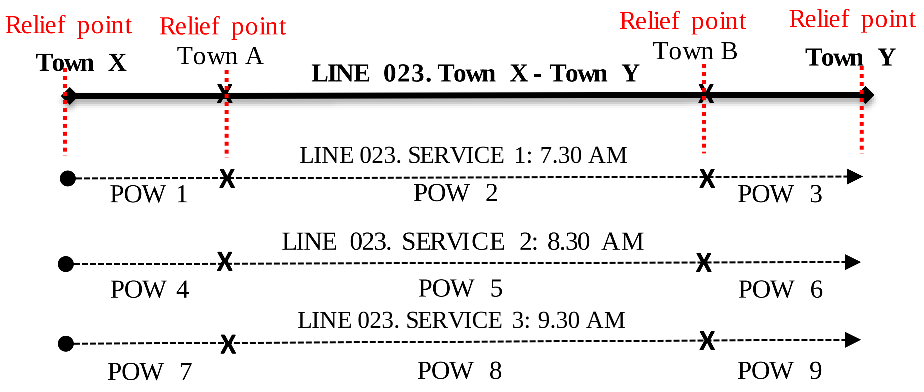


Figure 12. Representation of the network

As represented, the timetables of services are known. Therefore, the information of the pieces of work can be deduced. A piece of work $(\boldsymbol{POW_i})$ is described by an origin point $(\boldsymbol{op_i})$, a destination point $(\boldsymbol{dp_i})$, an origin time $(\boldsymbol{ot_i})$, a destination time $(\boldsymbol{dt_i})$, a line $(\boldsymbol{l_i})$ it belongs to and the correspondent timetable or service $(\boldsymbol{tt_i})$, therefore, $\boldsymbol{POW_i} = (\boldsymbol{op_i}, \boldsymbol{dp_i}, \boldsymbol{ot_i}, \boldsymbol{dt_i}, \boldsymbol{l_i}, \boldsymbol{tt_i})$. For example, a piece of work could be defined as represented in Table 9. In this example, the piece of work number 4, starts in Town X at 8:30, ends in Town A at 9:00, belongs to line 023 and corresponds to the service 2.

Table 9. Example of the information of a piece of work

POW_i	op_i	dp_i	ot_i	dt_i	l_i	tt_i
4	Town X	Town A	8.30	9.00	023	2

It is significant that all the pieces of work with the same values of (l_i, tt_i) constitute a service. For example, the Table 10 defines the pieces of work of the different services of the line number 023.

Table 10. Example of the information of pieces of work of the same line

POW_i	op_i	dp_i	ot_i	dt_i	l_i	tt_i
1	Town X	Town A	7.30	8.00	023	1
2	Town A	Town B	8.00	8.15	023	1
3	Town B	Town Y	8.15	8.30	023	1
4	Town X	Town A	8.30	9.00	023	2
5	Town A	Town B	9.00	9.15	023	2
6	Town B	Town Y	9.15	9.30	023	2
7	Town X	Town A	9.30	10.00	023	3
8	Town A	Town B	10.00	10.15	023	3
9	Town B	Town Y	10.15	10.30	023	3

Moreover, in addition to the pieces of work resulted of the timetabling, drivers also perform distances without passengers. These trips are called *deadheads.* This fact can occur in these circumstances:

- at the beginning of the day when the driver approaches the first header of the day,
- at the end of the day when the driver should approach the garage or
- during the day when in order to avoid wasting time, the driver drives from one point to another point of the network out of service.

6.1.2 Observation 1: suitable relief points

Reliefs can occur at any time of the duty and at any stop of the network. However, with the aim of reducing the number of pieces of work and in consequence, the execution time, there are two parameters that make some stops more suitable than others for a relief:

1. The distance between the depot and the stop:
   When starting or ending a duty, the driver will spend more time to return to the depot (when ending) or to arrive to the relief point (when starting) depending on the location of the relief. So, further the stop is, the productivity of the duty will decrease because the deadhead time increases.
2. The number of lines that cross the stop:
   The more lines that go through a stop, the more drivers that will coincide there. If drivers coincide in the same point, deadheads times will decrease.

In summary, in order to simplify the problem and reduce the volume of possible solutions, the appropriate relief points could be selected at the first step. The selection would prioritize the following relief points:

1. Relief points closer to the depot.
2. Relief points in which lines coincide.
3. Relief points that fulfil both conditions.

6.1.3 Observation 2: aspects to consider when introducing pieces of work in the duties

When creating duties, pieces of work are joined to complete a duty. When joining pieces of work, restrictions are considered.

Suppose that the last piece of work introduced into a duty has associated the following information: $POW_i = op_i, dp_{i,}ot_i, dt_i, l_i, tt_i$. Imagine also that the time limit of the working period that is being generated is *WP*. In that moment there will be a group of pieces of works candidates to follow this piece of work in the creation of the duty. In this situation, a piece of work j with this associated information $POW_j = \left(op_j, dp_j, ot_j, dt_j l_i tt_i\right)$ could be a candidate if the following conditions are met:

- If $(op_j = \ dp_i)$ the condition to be met is $dt_i \leq ot_j$ (6.1)
  else the condition to be met is $(dt_i + time_{dp_i - \ op_j}) \leq ot_j$ (6.2)
  where $time_{dp_i - \ op_j}$refers to the time required to arrive from dp_i to op_j.
- Moreover, if the ending of the duty is near, the deadhead to arrive to the depot has to be considered so,
  If $(dt_j + time_{dpj-depot}) \leq WP$ (6.3)
  else $dt_j \leq WP$ (6.4)
  where $time_{dpj-depot}$ refers to the time required to arrive from dp_j to depot.

However, these conditions are not enough. As highlighted before, the number of reliefs in a service is limited to one. But, which consequences has this limitation in the scheduling?

1. It does not limit the reliefs a driver can suffer per day.
2. It means that pieces of work from a service can be at most part of two duties. So, the dependence among pieces of work of the same service is crucial when generating duties. Figure 13 shows an example with the purpose of explaining this condition. It refers to a service divided into four pieces of work. As represented, the division of this service could be proposed in four different ways.

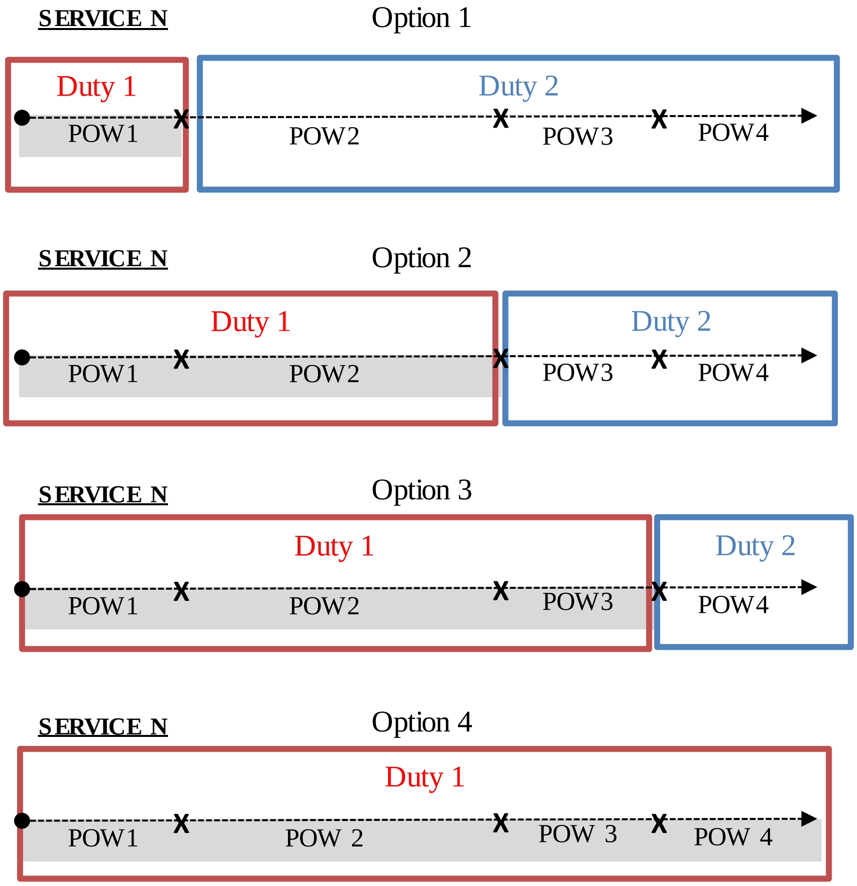


Figure 13. Examples of service breaks

And how does this condition affect the evaluation of the candidate pieces of work? Different situations can occur:

1. Situation 1: in the service of the candidate piece of work, there is one or more pieces of work introduced in a prior duty.
   In that case, the candidate and the pieces of work of its service that have not been assigned yet, perform as a block. So, the scheduler has to check not only if the candidate piece of work is appropriate but also that the ones that are not assigned yet can be introduced in the duty. Figure 14 represents these situations.

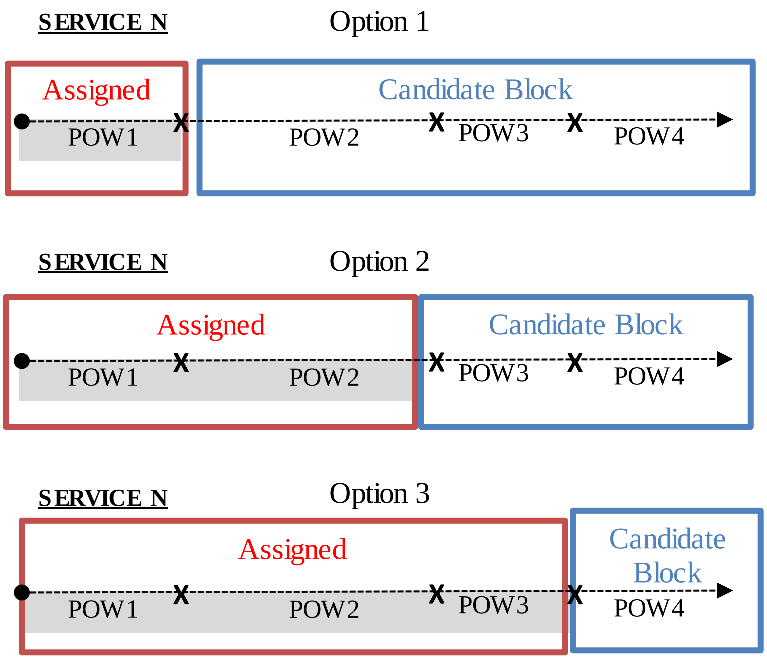


Figure 14. Selection of a piece of work. Example 1

In Option 1, POW 1 has been assigned to a prior duty. So, POW2, POW3 and POW4 will perform as a block. In consequence, in the evaluation of the block, if the last piece of work introduced into the duty has associated the following information $POW_i = op_i, dp_{i,}ot_i, dt_i, l_i, tt_i,$ these equations have to be considered:

$$op_{block} = op_{POW2} \tag{6.5}$$
$$dp_{block} = dp_{POW4} \tag{6.6}$$
$$ot_{block} = ot_{POW2} \tag{6.7}$$
$$dt_{block} = dt_{POW4} \tag{6.8}$$

- If $(op_{block} = dp_i)$ the condition to be met is $dt_i \leq ot_{block}$ (6.9)
  else the condition to be met is $(dt_i + time_{dp_i - op_{block}}) \leq ot_{block}$ (6.10)

- Moreover, if the ending of the duty is near, the deadhead must be considered so,
  If $(dt_{block} + time_{dp_{block} - depot}) \leq WP$ (6.11)
  else $dt_{block} \leq WP$ (6.12)

In Option 2, POW1 and POW2 have been assigned to a prior duty. So, POW3 and POW4 will perform as a block. In Option 3, POW1, POW2 and POW3 have been assigned to a prior duty. So, POW4 will perform alone.

2. Situation 2: in contrast, imagine that in the service of the candidate piece of work, there is not any piece of work that has been introduced in a prior duty. In this case the position of the candidate piece of work is crucial. Figure 15 shows the situations in this case.

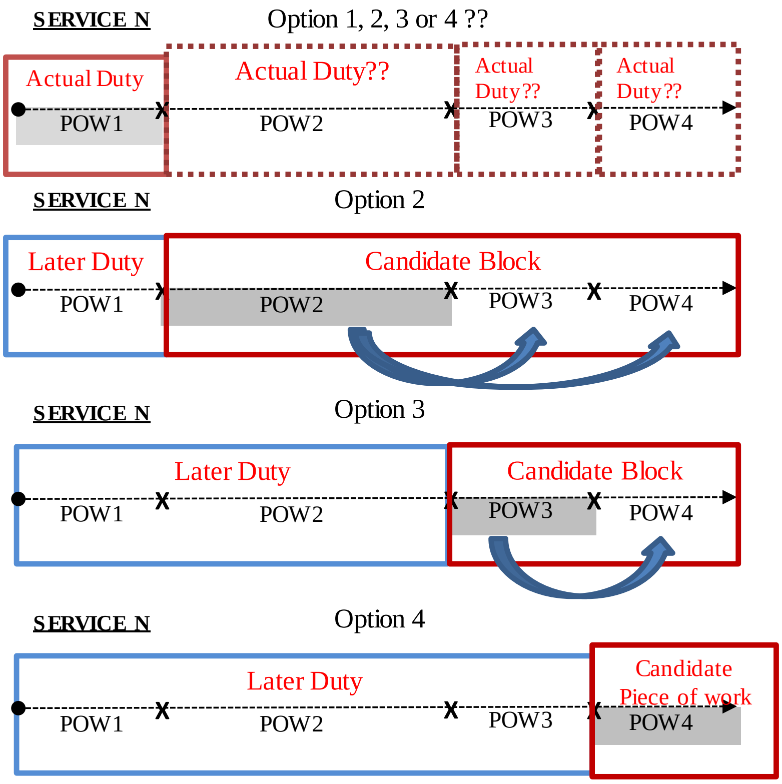


Figure 15. Selection of a piece of work. Example 2

As it can be seen in Option 1, if the candidate piece of work is the first one, it does not impact on the rest, so it will be evaluated alone.

However, if the candidate is in a position different from the first one (Option 2, Option 3 or Option 4), it supposes that:

i. the pieces of works that follow the candidate, must be introduced in this duty. So, not only the candidate piece of work will be evaluated but also the following ones, all of them will be evaluated as a block.
ii. the pieces of work that precede the candidate will have to be introduced in a later duty.

6.2 Restrictions related to driver's duties

Regulation (EC) 561/2006 provides a common set of EU rules for maximum daily and fortnightly driving times, as well as daily and weekly minimum rest periods for all drivers of road haulage and passenger transport vehicles, subject to specified exceptions and national derogations. The question is that one of these exceptions refers to vehicles used for the carriage of passengers on regular services where the route covered by the service in question does not exceed 50 kilometres. For these cases, there is not any European regulation defined, so national regulation has to be considered. This research project focuses on the Spanish Real Decreto 902/2007.

In this context, this project focuses on public bus transport operators which operate inter-city lines that have more than 50 kilometres as well as inter-city lines that have less than 50 kilometres. In consequence, they have to take into account the Regulation (EC) 561/2006 as well as the Spanish Real Decreto 902/2007. Joined to crew scheduling it is noteworthy that duties that combine lines of more and less than 50 kilometres must comply with the regulation of lines of more than 50.

The information resumed in this point considers the restrictions described in both regulations as well as the restrictions mentioned in the section *2.1.1.4 Crew Scheduling Problem* of this document:

- Maximum duty duration.
- Maximum working time.
- Minimum and maximum every break duration, included the length of the meal break.
- Limited times to have a meal break: each driver must have his/her meal during the customary lunch or dinner time.
- Types of duties: straight and split duties have been distinguished.

6.2.1 Information: straight and split duties

As proposed in the literature (Smith, Wren 1988, Chen, Niu 2012, Li et al. 2015) duties are divided into two types: straight and split duties.

The following lines resume the restrictions that must be considered:

1. All drivers start and end their duty at the depot.
2. The maximum duration of a straight duty is, *Straight_ Dur_Max.*
3. The maximum duration of a split duty is *Split_Dur_Max.*
4. Straight duties do not have meal breaks.
5. The minimum duration of the meal break of a split duty is *Split_Meal_Break*
6. The meal break is carried out at the depot and at a specific time window, limited by *Split_Break_Start* and *Split_Break_End*.
7. The sum of the working periods of a split duty cannot exceed *Split_Sum_WP*.

6.2.2 Observation 3: conditions to create split duties.

Considering the restrictions mentioned in the previous section, there are some conditions to take into account when scheduling split duties.

Figure 16 represents two short duties that could be joined in a new split duty. As shown, the most relevant data refers to the starting and ending times of each duty. Afterwards, the conditions to be met are defined:

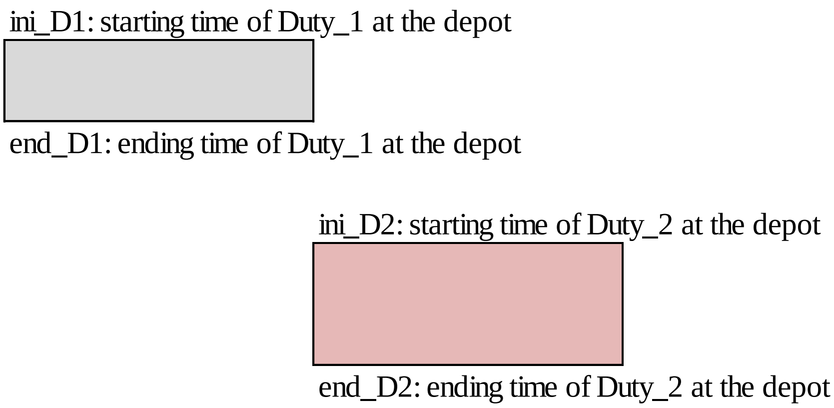


Figure 16. Representation of two short duties

Conditions:

$$Split\_Dur\_Max \geq (end\_D2 - ini\_D1) \tag{6.13}$$

$$Split\_Meal\_Break \geq (ini\_D2 \ - end\_D1) \tag{6.14}$$

$$Split\_Sum\_WP \geq (end\_D1 - \ ini\_D1 \ + \ end\_D2 - \ ini\_D2\ ) \tag{6.15}$$

$$ini\_D2 \geq Split\_Break\_Start \tag{6.16}$$

$$ini\_D2 \leq (Split\_Break\_End \ - Split\_Meal\_Break) \tag{6.17}$$

6.2.3 Information: breaks in lines with more than 50 kilometres

In lines of more than 50 kilometres, after a maximum working period *(L_Max_WP)* a driver must take a minimum uninterrupted break (*L_ Break)*.

If the scheduler wants, *L_Break* can be replaced by two shorter breaks, *L_Break*$_1$ and *L_Break*$_{2.}$ However, these conditions must consider:

- The minimum values of *L_Break*$_1$ and *L_Break*$_2$ are fixed and they sum the duration of *L_Break.*
- The order of the breaks has to be respected compulsory, i.e. *L_Break*$_1$ happens before *L_Break*$_2$.
- Before *L_Break*$_1$ happens, idle-times which take less time than the duration of $Break_1$ will be considered as non-driving periods.
- Between *L_Break*$_1$ and *L_Break*$_2$, all the idle-times which take less than *L_Break*$_2$ will be considered as non-driving periods.

Figure 17 represents both options. Afterwards, the conditions related to these restrictions are formulated:

Option 1:

L_Max_WP	*L_Break*

Option 2:

L_WP_1	*L_Break_1*	*L_WP_2*	*L_Break_2*

Figure 17. Daily break times in lines with more than 50 kilometres

$$L\_Max\_WP = L\_WP_1 + L\_WP_2 \quad (6.18)$$

$$L\_Break = L\_Break_1 + L\_Break_2 \quad (6.19)$$

Notice that in Option 2, *L_WP_1* and *L_WP_2* are not limited on time, these values depend on when *L_Break_1* occurs.

6.2.4 Observation 4: breaks in lines with more than 50 kilometres

Regarding the working periods and the breaks of duties of lines of more than 50 kilometres, it has been observed that there is an important feature which will be definitive when scheduling: the duration of idle-times between services. Considering this parameter, two types of transport networks can be distinguished: networks that have long idle-times, and on the contrary, networks that have short idle-times.

On one hand, if it refers to a network where idle-times generally last longer than *L_ Break*, these idle-times could change into breaks and in consequence, breaks will rarely have to be forced by the scheduler. Duties result of this situation are represented in Figure 18.

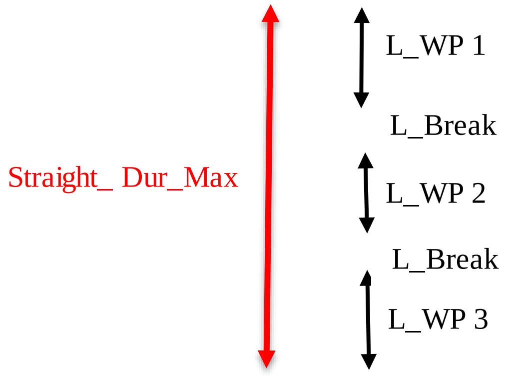


Figure 18. Representation of a duty with longer idle-times than L_Break

On the other hand, if it is a network where idle-times take less than *L_ Break*, these idle-times will not be considered breaks and in consequence, the scheduler will have to force the breaks when creating the duty.

With the aim of defining the types of duty result of this situation, another step is given and another observation is carried out. Taking into account the values of the variables, a duty will have at least two working periods. The values of both working periods will be different, the duration of one of them will be $\mathrm{L\_Max\_WP}$ and the duration of the other one will be shorter. So, these conditions will be respected:

$$L\_Max\_WP + L\_Break < Straight\_Dur\_Max < 2 * L\_Max\_WP + L\_Break \qquad (6.20)$$

$$\mathrm{WP1} = \mathrm{L\_Max\_WP} \qquad (6.21)$$

$$\mathrm{WP1} > \mathrm{WP2} \qquad (6.22)$$

Considering these values, it has been concluded that in networks with short idle-times, there are four types of duties, depending on when and how breaks and working periods are established. Figure 19 resumes these four options:

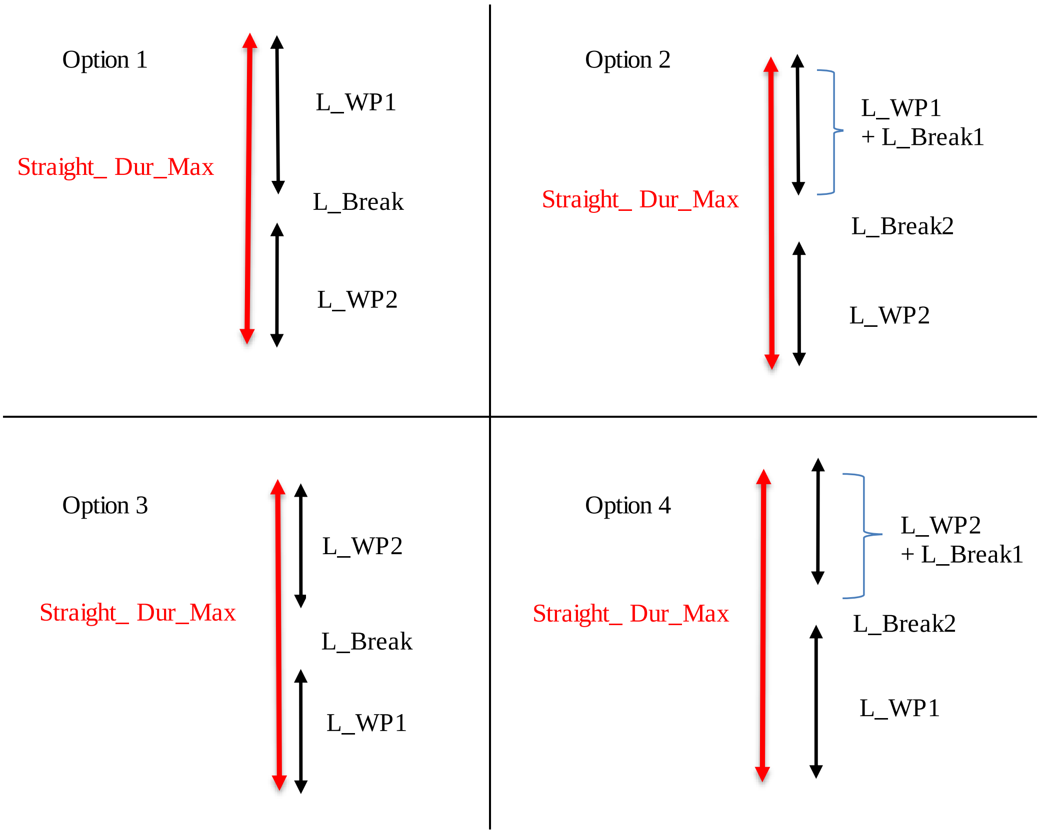


Figure 19. Options of duties with smaller idle-times than L_Break

6.2.5 Information: breaks in lines with less than 50 kilometres

In lines of less than 50 kilometres, the minimum total break time during the duty is limited (S_Break). Nevertheless, S_Break can be divided as the scheduler needs. That is, any idle-time in the duty could be considered as part of the daily break.

6.2.6 Observation 5: breaks in lines with less than 50 kilometres

Imagine, a case where *S_Break* consists of one minute breaks. Although this duty is legal considering the regulation, the driver would not have a real rest time during the duty and in consequence, the scheduler would never propose it.

With the aim of resolving this situation, a new input parameter has been defined. *S_Min_Break* is the minimum duration of a break in duties with less than 50 kilometres. This means that idle-times which are shorter than *S_Min_Break* will not be considered as break times.

Related to this new parameter new formulation is defined:

$$S_{Break_i} \geq S\_Min\_Break \qquad (6.23)$$

$$\sum_{i=1}^{n} S\_Break_i \geq S\_Break \qquad (6.24)$$

7 Formulation of the model

After observing the system, the next step contains the formulation of the model, and has two objectives: first, to define the model in which this problem is based on; second, to describe the solution technique developed.

7.1 The model

The three necessary elements to define a model are (1) the objective function, (2) the input data and (3) the problem restrictions.

Firstly, considering that the result of the scheduling tool will be measured in duties, as defined in the section 5.1, the objective function of the model looks for minimizing the number of duties. It is defined as follows:

$$Minimise \ \sum_{j=1}^{m} d_j \qquad (7.1)$$

Secondly, the data needed is the following one:

- Pieces of work: each piece of work *i* (POW_i) is described by an origin point (op_i), a destination point (dp_i), an origin time (ot_i), a destination time (dt_i), the line (l_i) that belongs to and the correspondent timetable (tt_i), so, it is defined as $POW_i = (op_i, dp_i, ot_{i,} dt_i, l_i, tt_i)$. Note that (l_i) defines if the line has less or more than 50 kilometres.
- Depot: the garage where drivers start and finish their workday.
- Time (minutes) between each pair of stops of the network and time between each stop and the depot.

Finally, the restrictions are listed below:

- All drivers start and end their duty at the depot.
- The minimum duration of a duty is limited.
- The maximum duration of a straight duty is limited (*Straight_ Dur_Max*).
- The maximum duration of a split duty is limited (*Split_Dur_Max*).
- The sum of the working periods of a split duty cannot exceed *Split_Sum_WP*.
- The minimum duration of the meal break of a split duty is limited (*Split_Meal_Break*).
- The meal break is carried out at the depot and at a specific time window, limited by *Split_Break_Start* and *Split_Break_End*.
- In lines of more than 50 kilometres, after a maximum working period *(L_Max_WP)* a driver must take a minimum uninterrupted break (*L_ Break)*. If the scheduler considers it more convenient, *L_Break* can be replaced by two shorter breaks, *L_Break*$_1$ and *L_Break*$_2$. The minimum values of *L_Break*$_1$ and *L_Break*$_2$ are fixed

and they sum the duration of *L_Break.* The order of the breaks has to be respected compulsory, i.e. *L_Break*$_1$ happens before *L_Break*$_2$.

- In lines of less than 50 kilometres, the minimum total break time during the duty is limited (*S_Break*).
- When scheduling lines of less than 50 kilometres, the minimum duration of a break is limited (*S_Min_Break*).
- During the execution of a service, only one driver relief can be carried out.
- The number of reliefs a driver can suffer per day is unlimited.

7.2 Description of the GRASP

7.2.1 Selection of the solution technique

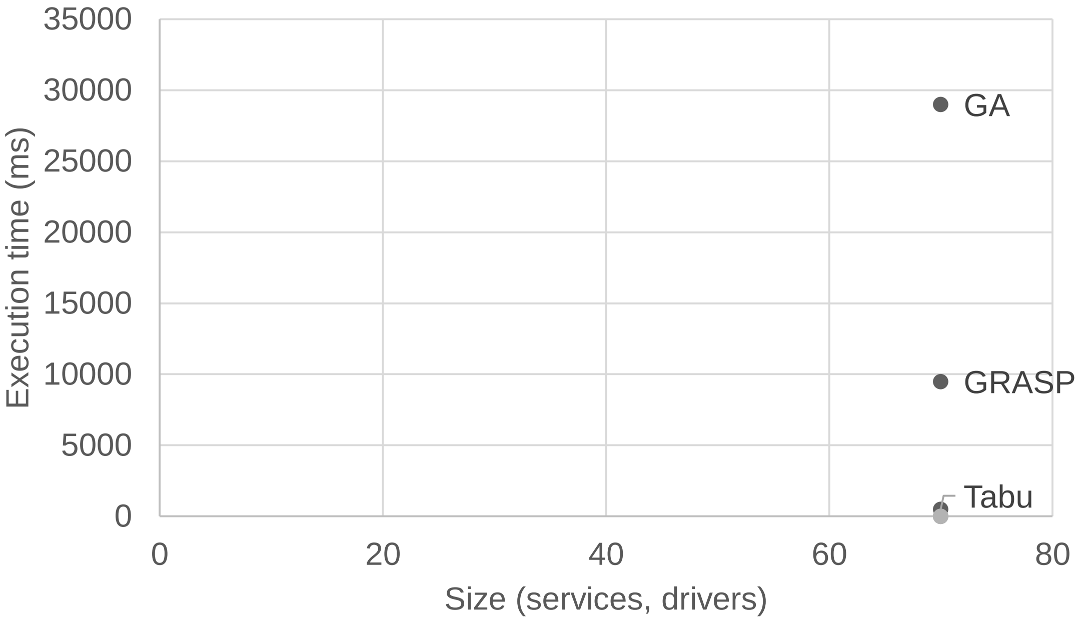
As presented at section 2.3, in most practical cases the amount of data and the necessary execution time, make integer programming approaches unviable for obtaining the optimal solution (Alfares 2004) when solving the Crew Scheduling Problem. However, the higher speed of metaheuristic procedures and the high quality of the obtained solutions facilitates its execution. As shown before, in the case of the Crew Scheduling Problem, commonly used metaheuristics are GRASP, Tabu Search and Genetic Algorithm.

In order to choose one of these three solution techniques, the work shown by López et al. (2009) has been taken as reference. The authors experimentally analysed a set of representative techniques on the Crew Scheduling Problem of a road passenger transportation problem. On one hand, the similarity is that they focus on an inter-urban scenario. On the other hand, the difference is that they focus on private transport.

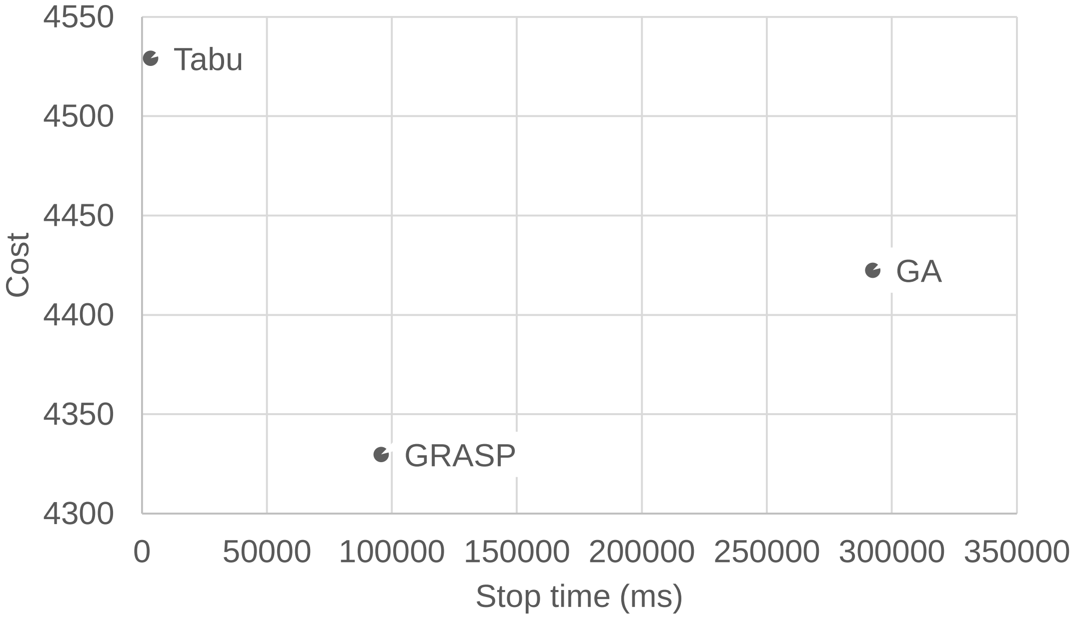
This work experimentally analyses some different solution techniques, GRASP, Genetic Algorithm and Tabu among them. With this aim, up to 70 problem instances were generated with different degree of complexity. Next, the results of these three techniques are resumed:

Table 11. Summary of the analysis: techniques and their properties. (Lopez et al. 2009)

Property	Authors’ definition of the property	Tabu	GRASP	GA
Modeling expressiveness	Whether the methods allow the specification of constraints as part of the program (constraints coded), or whether constraints should be explicitly provided (the declarative way) by means of either compatible variable-value pairs or with the use of functions. When constraints must be explicitly provided and the problem is large, some kind of pre-processing step is required to obtain the constraint set, even though the specification language sometimes provides a way to specify them by means of complex expressions.	Coded	Coded	Coded
Anytime	Whether the algorithm can be stopped and its execution resumed, giving the best solution found so far or not, or only partially (can stop but not resume).	Yes	Yes	Yes
Time complexity	Whether the method as tested is able to solve up to the 70th test case (low), up to the 20th case (medium) or few cases (high).	Low	Low	Low
Memory complexity	The amount of memory required by the method, to store either constraints or internal data. High means that the method requires a lot of memory, so, dynamic memory or other kinds of programming tricks should be used to keep handling memory in an efficient way.	Medium	Medium	Low
Parameter tuning	Whether the method requires several runs in order to tune the parameters required. In this sense, the label ‘‘Yes’’ indicates that with the current parameter estimations the algorithm has not found the best solution.	Yes	Yes	Yes
Tool	Whether there is a free license tool on the shell to test the problem or not. Note that tool availability could force the problem to be modelled according to the tool requirements. In addition, the tools available are not always the most efficient ones,	No	No	No



Graph 1. Maximum problems managed by the methods and the associated execution time. (Lopez et al. 2009)



Graph 2. Time (ms) required to find a solution for the 70 case and the cost of the solution. Only MIP finds the optimal solution. (Lopez et al. 2009)

As defined in section 5.1, the criteria chosen to consider when developing the scheduling tool are these ones: model reliability and simplicity, solution time, solution quality and applicability. Table 12 resumes the analysis carried out.

Table 12. Comparison of the properties of GRASP, Tabu and GA

Criterion	Description	Comments
Model reliability and simplicity	All the restrictions of the model will compulsorily be respected. So, the solution will be totally feasible and no manual adjustments will be necessary once the solution is obtained.	The three options can be coded, so all restrictions could be coded. There is not any tool available in any of the cases.
Solution time	As well as getting a good solution, getting it fast will be a key aspect when executing the tool.	Tabu is the fastest; GRASP is faster than GA.
Solution quality	The solution quality (the goodness of the solution) will be measured in duties. So, the aim will be to solve the problem with the minimum number of drivers' duties.	GRASP finds the lowest cost; GA finds a lower cost than Tabu.
Applicability	The tool has to answer to a real problem of a real company; that means that it is able to answer a problem of large instances.	The three metaheuristics are able to solve up to the 70th test case, that is, the most complex problem.

Based on this analysis, GRASP has been chosen as the solution method of this project. The main reason is that GRASP gets the best solution. Furthermore, although it is not the fastest method, it seems fast enough.

As described in 2.3.1.2, GRASP has two phases: a construction and a local search. Next, the construction phase and the local search particularly developed in this research project are described. As restrictions in the networks of more and less than 50 kilometres are different, the operation of the algorithm in both cases is different too.

7.2.2 Construction phase of the GRASP

The aim of the construction phase is to provide a feasible solution. The construction procedure is repeated as many times as the scheduler requires. In fact, the number of iterations or repetitions is an input parameter of the algorithm. It is remarkable that higher this parameter is, longer it will take to finish this phase. Figure 20 resumes the construction phase.

The first step is to create a feasible solution composed of a set of feasible duties. This solution will be temporally defined as the best solution. After that, each iteration repeats the procedure: first, a new feasible solution is generated, then it is compared with the best solution got before and the best of them, the one which is made up by less duties, will be considered as the best solution. Once all the iterations are carried out, the best solution found will be the solution of the construction phase.

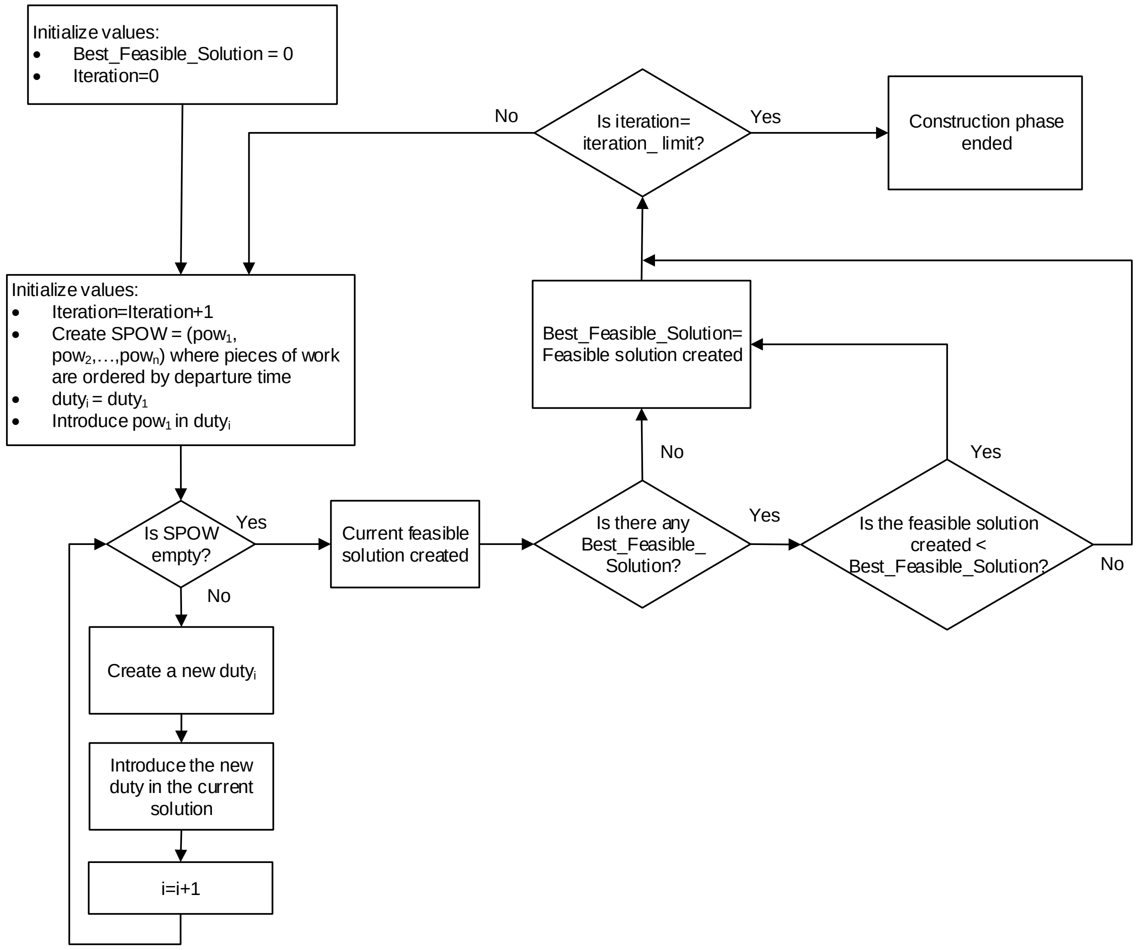


Figure 20. Creation of a feasible solution in the construction phase

One of the key points of the construction phase is that the solution got in each iteration is different. Figure 21 illustrates in more detail the steps followed when creating a duty.

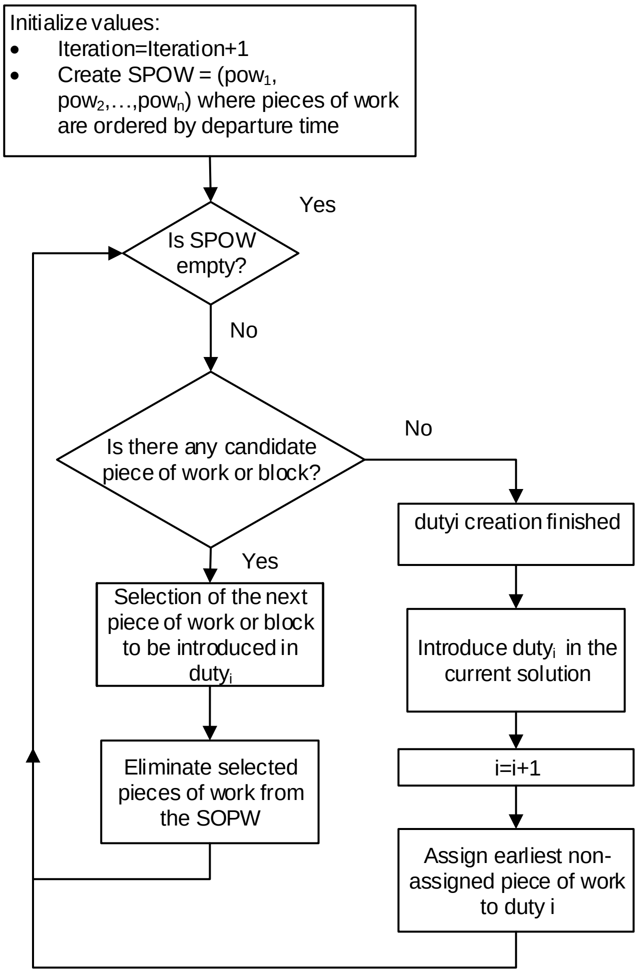


Figure 21. Creation of a feasible duty in the construction phase

At this point, the outstanding issues when creating a duty are two:

1. To know if there is any candidate piece of work or any candidate block to be introduced in the duty.
2. The selection of the next piece of work or block to be introduced.

Next, these two steps are described.

7.2.2.1 Is there any candidate piece of work or block?

As defined in sections 6.1.3, duties are created by joining individual pieces of work or blocks of pieces of work.

Moreover, a key value to keep in mind at all times is the working period (*WP*). As shown, *WP* takes different values along the creation of the duty depending on when breaks must happen. Besides, because of the differences between regulations, the evolution of *WP* varies when scheduling lines of more 50 kilometres or less than 50 kilometres.

- Evolution of the WP in lines with more than 50 kilometres

As described in section 6.2.4, in duties made up of pieces of work of lines of more than 50 kilometres, two different situations can occur: networks where idle-times generally last longer than the minimum uninterrupted break (*L_ Break)* or in contrast, networks where idle-times generally last shorter than *L_ Break*.

In the first situation (Figure 18), when the break is not generated in the introduction of pieces of work itself, the algorithm forces it. In cases where the break is generated, the algorithm recalculates the new value of WP and it does not force any break.

In the second situation, as shown in Figure 19, there are four types of duties, depending on when and how breaks and working periods are established. Figure 22 shows how the algorithm controls breaks and how it enables that the four options can occur. This second fact depends on the value of two new input variables, prob2 and prob4. Specifically, prob2 defines the probability to generate a duty of type 3 and prob4 defines the probability to generate a duty of type 4. Indirectly, these values also define the probability to generate duties of type 1 and 2.

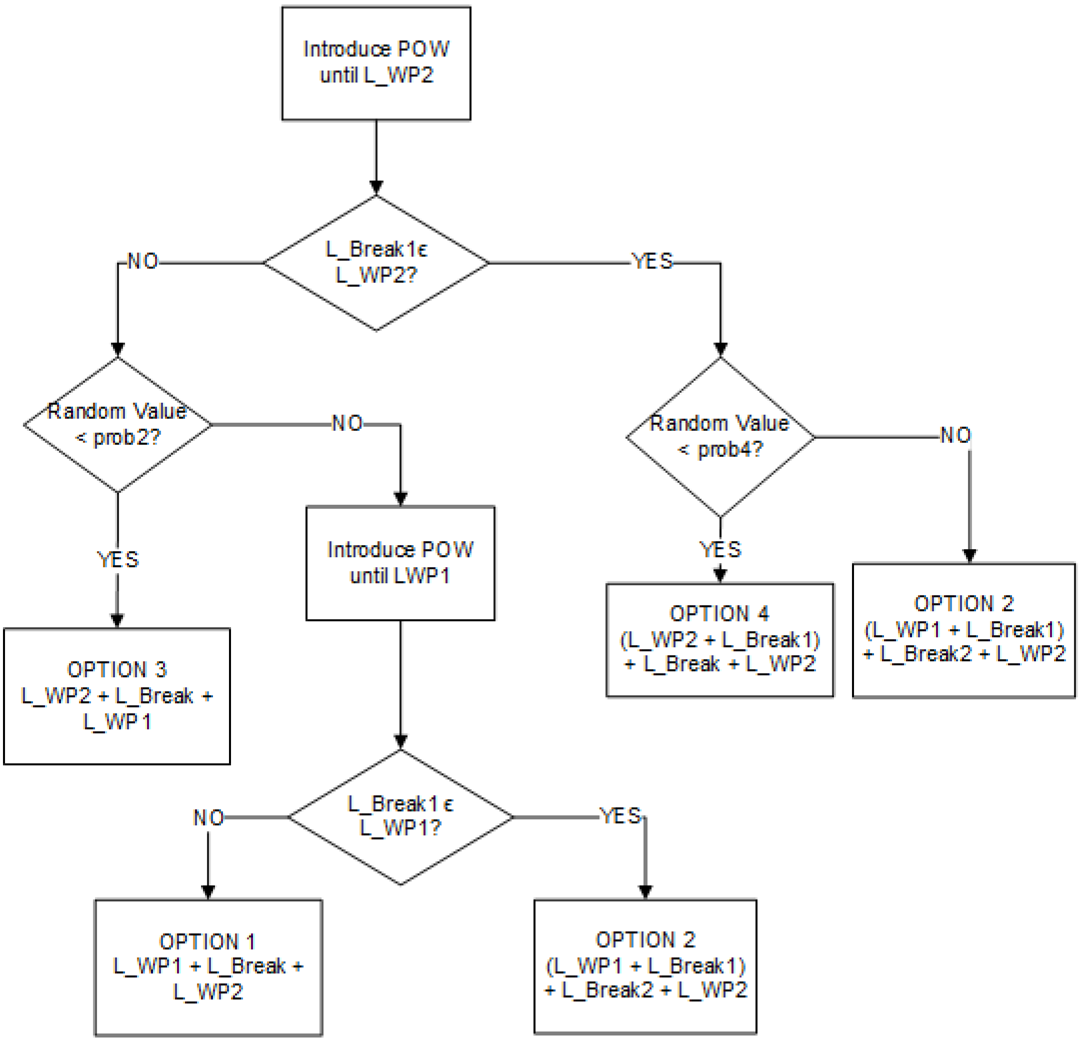


Figure 22. Procedure of the algorithm in lines with more than 50 kilometres (situation 2)

- <u>Evolution of the WP in lines with less than 50 kilometres</u>

The logic the algorithm follows when creating these duties is shown in Figure 23. As observed, the algorithm controls the breaks occurring throughout the duty. As there is a mandatory minimum break, at first, the value of *Limit_WP* corresponds to the maximum duration of the duty minus mandatory minimum break (*S_Break*). As breaks occur, the value of WP is updated until it reaches a maximum value, the maximum duration of the duty (*Straight_Dur_Max*). Just the idle-times that take longer than the minimum duration *S_Min_Break* will be considered as breaks.

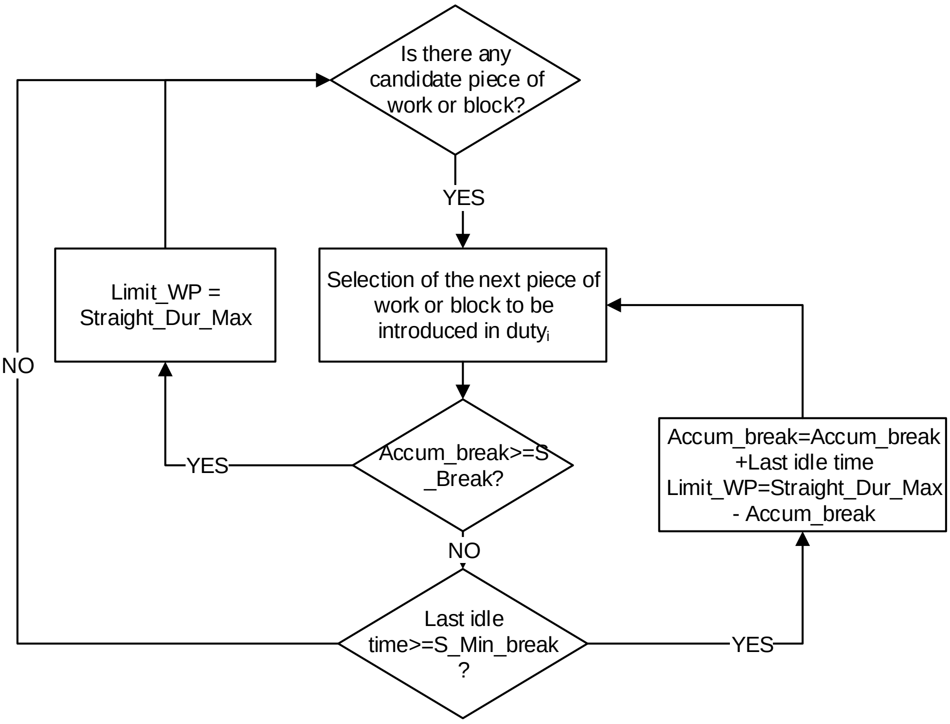


Figure 23. Procedure of the algorithm in lines with less than 50 kilometres

7.2.2.2 Selection of the next piece of work or block to be introduced in a duty

As mentioned, the solution is different every time a duty is generated. This is due to how pieces of work or blocks are chosen at all times.

As found in the literature, in each iteration, all the elements are ordered in a candidate list taking into account the benefit of selecting each element. This list of candidates is called *The Restricted Candidate List* (RCL). After that, an element is selected and included in the solution.

In our case, the benefit of selecting a piece of work, will be inversely proportional to the idle-time that will generate its selection, i.e., the smaller the timeout is, the greater will be the likelihood of the candidate to be selected.

7.2.3 Local search phase of the GRASP

The aim of this second phase is to reprocess the best solution got in the construction phase. The local search is divided into two steps:

1. First, duties that are not efficient enough are divided into new parts. After that, these parts are re-joined to try to get a better solution, that is, to answer to these pieces of work with less straight duties.
2. Then, the possibility of creating split duties linking inefficient duties is analysed.

Next, these steps are developed in detail.

7.2.3.1 Creation of new straight duties

Once the construction phase finishes, the first step analyses the possibility to reduce the number of duties of the solution got, reprocessing the ones that are not efficient enough. The following lines describe how this task is carried out:

1. Calculate the efficiency of each duty created in the construction phase.

First, the total duration of the duty is calculated, since the leaving time from the depot until the arriving to it (except on duties shorter than the minimum duration of the duty, in these cases, the total duration corresponds to the minimum duration). Then, the driving time in service of each duty is added. Finally, the efficiency of the duty is measured by dividing the time in service by the total duration of the duty.

2. Select duties to be reprocessed because they are not efficient enough.

The efficiency is an input parameter of the local search, i.e., the traffic manager decides which duties are good enough to keep and not to be reprocessed. Notice that the less stringent the efficiency is, the less work to reprocess, and in consequence, the smaller the execution time.

3. Divide non-effective duties into new parts.

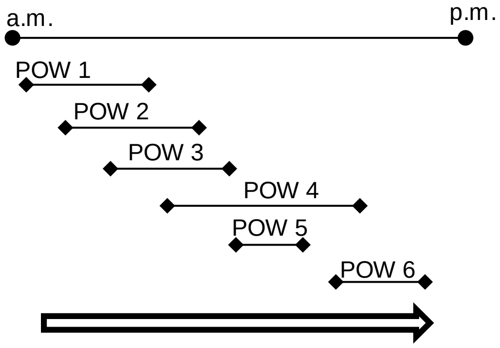
Inefficient duties are broken into new parts. A new input parameter defines the criterion followed to define these new parts: the idle-time allowed between pieces of work. So, instead of considering relief points and dividing duties in pieces of work as in the construction phase, the time between pieces of work has been analysed. That is, unions made in the construction phase that are considered good enough are kept. Pieces of work are just divided when idle-times are longer than the value that the traffic manager has defined. Notice that depending on this value, parts will be different; the less stringent this parameter is, the less work to reprocess, and in consequence, the smaller the execution time.

4. Creation of new straight duties.

The creation of duties in the local search is simpler than the one in the construction phase. The goal in this second phase is to carry out a targeted search instead of covering the whole field of possible solutions again.

The first step consist of ordering the pieces of work by departure time. Then, with the aim of getting different solutions, five different processes are carried out.

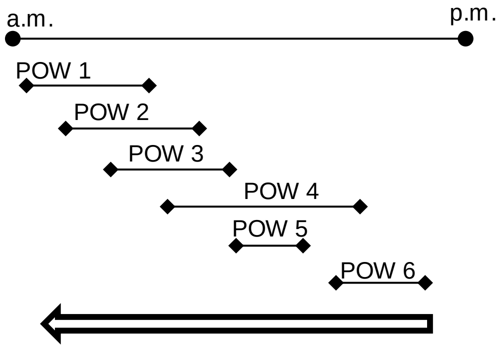
1. The duty creation starts from the earliest free piece of work. At the time of selecting the next part to be assigned, the algorithm assigns the nearest candidate, that is, the one that supposes less deadhead



Solution 1:
Duty 1 : POW 1 + POW 4
Duty 2 : POW 2 + POW 5 + POW 6
Duty 3 : POW 3

Figure 24. Local search example. Solution 1.

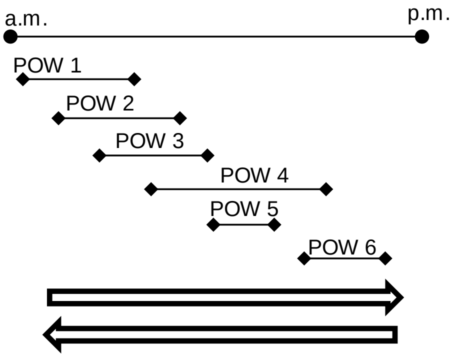
2. The duty creation starts from the earliest free piece of work. At the time of selecting the next part to be assigned, the algorithm assigns the nearest candidate, that is, the one that supposes less deadhead.



Solution 2:
Duty 1 : POW 3 + POW 5 + POW 6
Duty 2 : POW 2 + POW 4
Duty 3 : POW 1

Figure 25. Local search example. Solution 2.

3. The duty creation starts from the earliest or latest free piece of work, depending on whether the duty is odd or even. If a duty is odd, it is created from the beginning; if a duty is even, it is created from the end. At the time of selecting the next part to be assigned, the algorithm assigns the nearest candidate, that is, the one that supposes less deadhead.

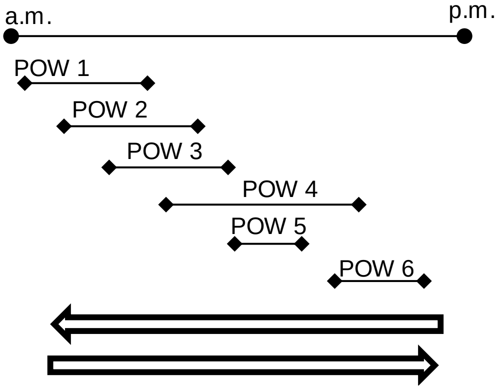


Solution 3:
Duty 1 : POW 1 + POW 4
Duty 2 : POW 3 + POW 5 + POW 6
Duty 3 : POW 2

Figure 26. Local search example. Solution 3.

4. The duty creation starts from the earliest or latest free piece of work, depending on whether the duty is odd or even. If a duty is even, it is created from the beginning; if a duty is odd, it is created from the end. At the time of selecting the

next part to be assigned, the algorithm assigns the nearest candidate, that is, the one that supposes less deadhead.



Solution 4:
Duty 1 : POW 3 + POW 5 + POW 6
Duty 2 : POW 1 + POW 4
Duty 3 : POW 2

Figure 27. Local search example. Solution 4.

5. The duty creation depends on a binary random value. If the value is 1 the duty is created from the beginning; if it is 0 from the end. At the time of selecting the next part to be assigned, the algorithm assigns the nearest candidate, that is, the one that supposes less deadhead.

7.2.3.2 Creation of split duties

Once recalculated the new straight duties, the final step is the creation of split duties.

First, the following indicators are calculated for each duty: driving time, total duration, starting time, ending time, break time, deadhead time, idle-time, leaving place of the first piece of work, leaving stop of the first piece of work, arriving time of the last piece of work and arriving stop of the last piece of work.

Then, the second step is to verify whether split duties can be defined considering the restrictions mentioned in section 6.2.2.

7.2.4 Input parameters of the algorithm

When describing the algorithm, some input parameters that affect the result of the algorithm have been mentioned. These parameters are listed in Table 13:

Table 13. List of input parameters of the algorithm

Input parameter	Description	Phase	Influence
iterations	Number of repetitions of the construction phase	Construction phase	+50 km schedules -50 km schedules
prob2	Probability to generate type 3 duties	Construction phase	+50 km schedules

Input parameter	Description	Phase	Influence
prob4	Probability to generate type 4 duties	Construction phase	+50 km schedules
set_efficiency	Minimum efficiency defined to consider a duty good enough to avoid the local search	Local search	+50 km schedules -50 km schedules
set_stop	Maximum time allowed between pieces of work in the creation of duties before the local search	Local search	+50 km schedules -50 km schedules

8 Verification of the model

In order to verify the performance of the algorithm, a real company whose scheduling meets the characteristics of the studied problem in this research project has been chosen as a test scenario. To respect the anonymity of the company hereafter, we will refer to the company as XYZ.

The scenario is described following the structure of the sixth chapter of this report, the observation of the system. The operations of the company as well as the scheduling restrictions are detailed and compared with the characteristics of the model.

8.1 Description of XYZ’s network: depots, lines, services, relief points and pieces of work

8.1.1 Information about depots

The company owns two work centres located 42 kilometres away one from each other. Drivers start and end each workday at these depots. Their size is different, while Depot A concentrates 180 drivers and 57 vehicles Depot B concentrates 44 drivers and 9 vehicles.

8.1.2 Information about lines

The network is made of 20 lines, 18 belong to Depot A and the remaining 2 belong to Depot B.

As for the length of the lines, both lines of Depot B have more than 50 kilometres. Among the lines of Depot A, 3 have more than 50 kilometres and 15 have less than 50 kilometres.

8.1.3 Information about services

There are different services for each line per day. All services are defined by its line, the departure time, the departure place, the arrival time and the arrival place. *Annexe A: Lines and services of XYZ* shows the number of services run per each line as well as their distance in both directions, outward and return. As one of the issues to be addressed in this research is the convenience of grouping lines when scheduling the information is divided into three groups: 1) services of Depot B, 2) services of Depot A with more than 50 kilometres and 3) services of Depot A with less than 50 kilometres. The summary of the characteristics of each group is shown in Table 14:

Table 14. Characteristics of each group of lines of XYZ (services and kilometres)

Group	Description	Services	Services (%)	Total Km.	Km (%)	Average Km/Services
Group 1	Lines from Depot B	65	9,29%	4601	21,60%	70,78
Group 2	Lines with more than 50 km from depot A	79	11,29%	4223	19,82%	53,46
Group 3	Lines with less than 50 km from depot A	556	79,43%	12.478	58,58%	22,44
TOTAL		700		21.302		

It is noted that the type of operations to be planned are different per group, that is, the ratios *% of service* and *kilometres* do not correspond. While services of Group 3, i.e. the lines of more than 50 kilometres, suppose the biggest amount of the services, the 79,43%, they only suppose the 58,58% of the kilometres. On the contrary, the sum of the Group 1 and Group 2, i.e. the lines of more than 50 kilometres, suppose a small number of services but the 41,42% of the kilometres. So, on one hand, we find a huge quantity of short operations to be scheduled while from the other hand, we find a smaller quantity of shorter operations to be scheduled.

8.1.4 Information about relief points

One of the key points of this model is that reliefs can occur at any time of the duty and at any stop of the network, not only in the header or in the final stops of services, as commonly happens. However, considering the observation in point 7.1.2, among all the stops of a network, besides the header and the final stops, some other appropriate relief points can be chosen.

In relation to the relief points in the case of XYZ, among the stops the network has, 35 have been selected considering these three parameters:

1. The time from the stop to the nearer depot.
2. The number of lines that coincide in the stop.
3. If the stop is a header or final stop.

Annexe B: Selected relief points resumes the information about the final selected relief points. It is noted that within the group of 35 relief points, there are 2 points which meet the three criteria, 13 that meet two of the three criteria and the remaining 20 that meet a single criterion.

8.1.5 Information about pieces of work

Once services and relief points have been defined, pieces of work are generated. *Annexe C: Pieces of work to the scheduled in XYZ* resumes the information of the pieces of work to the scheduled in XYZ. In this case also, the previous three groups have been considered. Table 15 resumes the information about the grouped configurations:

Table 15. Characteristics of pieces of work of XYZ (services and pieces of work)

Group	Description	Services	Serv (%)	POW	POW (%)	POW/Serv.
Group 1	Lines from Depot B	65	9,29%	194	11,45%	2,98
Group 2	Lines with more than 50 km from depot A	79	11,29%	336	19,83%	4,25
Group 3	Lines with less than 50 km from depot A	556	79,43%	1164	68,71%	2,09
TOTAL		700		1694		

The first conclusion is reasonable; allowing handovers during services complicates the scheduling problem, as it increases from 700 operations to 1694.

Also, concerning the ratio POWs / SERVICES shown in the last column, it can be seen that its value is higher in the groups of more than 50 kilometres. This fact happens because longer a line is, more relief points finds in its way and consequently, it is broken in more pieces of work. Therefore, it can be concluded that, in this case, allowing reliefs during services complicates especially the scheduling of lines that measure more than 50 kilometres, where the number of operations triple and quadruple.

8.2 Restrictions related to driver’s duties in XYZ

Once the information about the operations of the company has been collected, the next step analyses the scheduling restrictions in order to validate that the selected scenario is suitable to operate the algorithm. Table 16 lists the restrictions discussed in point *7.1.2 Restriction related to driver’s duties* and the values they have in the XYZ. From this comparison, it is concluded that XYZ’s scheduling fits the requirements of the model.

Table 16. Restrictions related to driver's duties and their values in XYZ

Model	Values in XYZ
Driver's reliefs	
Reliefs can occur at any time of the duty and at any stop of the network. However, there are some stops more suitable than others.	There are 35 relief points where reliefs occur.
The number of reliefs in a service is limited to one.	The number of reliefs in a service is limited to one.
Straight and split duties	
All drivers start and end their duty at the same depot.	All drivers start and end the working day in a specific depot, in Depot A or in Depot B.
The minimum duration of a duty is limited.	The minimum duration of a duty is 7 hours 10 minutes.
The maximum duration of a straight duty is *Straight_ Dur_Max.*	The maximum duration of a straight duty is 9 hours.
The maximum duration of a split duty is *Split_Dur_Max.*	The maximum duration of a split duty is 11 hours.
The sum of the working periods of a split duty cannot exceed *Split_Sum_WP*.	The sum of the working periods of a split duty cannot exceed 9 hours.
The minimum duration of the meal break of a split duty is *Split_Meal_Break*	The minimum duration of the meal break of a split duty is 2 hours.

Model	Values in XYZ
The meal break is carried out at the depot and at a specific time window, limited by *Split_Break_Start* and *Split_Break_End*.	The meal break is carried out at the depot and at a specific time window, from 11.30 to 15.30.
<u>Breaks in lines with more than 50 kilometres</u>	
After a maximum working period *(L_Max_WP)* a driver must take an uninterrupted break (*L_ Break)*.	After a maximum working period of 4,5 hours a driver must take an uninterrupted break of 45 minutes.
L_Break can be replaced by two shorter breaks, L_Break_1 and L_Break_2. However, these conditions must be considered: • The minimum values of L_Break1 and L_Break2 are fixed and they sum the duration of L_Break. • The order of the breaks have to be respected compulsory, i.e. L_Break1 happens before L_Break2. • Before L_Break1 happens, idle-times which take less time than the duration of $Break_1$ will be considered as non-driving periods.	The break of 45 minutes can be replaced by two shorter breaks, the first one of 15 minutes at least and the second one of 30 minutes at least. • The values of 15 minutes and 30 minutes are fixed and they sum the duration of 45 minutes. • The order of the breaks have to be respected compulsory, i.e. the break of 15 minutes happens before the break of 30 minutes. • Before the break of 15 minutes happens, idle-times which take less time than the duration of 15 minutes are considered as non-driving periods.
Between L_Break_1 and L_Break_2, all the idle-times which take less than L_Break_2 will be considered as non-driving periods.	Between the break of 15 minutes and the break of 30 minutes, all the idle-times which take less than 30 minutes are considered as non-driving periods.

Model	Values in XYZ
A duty has at least two working periods, L_WP1 and L_WP2. The duration of *L_WP_1* and *L_WP_2* are not limited, these values depend on when *L_Break_1* occurs. Nevertheless, their maximum values are limited.	A duty has at least two working periods, the maximum duration is 4,5 hours for one working period and 3 hours and 45 minutes for the other one.
Breaks in lines with less than 50 kilometres	
The minimum total break time during the duty is S_Break.	The minimum total break time during the duty is 30 minutes. Nevertheless, S_Break can be divided attending scheduler's needs.
The minimum duration of a break is *S_Min_Break.*	The minimum duration of a break is 10 minutes.

9 Suitable alternative selection and presentation of the results of the analysis

The next step consists in making the adjustment of the input parameters. The aim is to identify their influence in the operation of the algorithm. For that, a Design of Experiments (DoE) has been developed.

9.1 Description of the testing scenario and the DoE

The planning process carried out by XYZ is the basis of this analysis. The company schedules its operations based on the group each line belongs to (Table 14 and Table 15). Thus, XYZ divides the total problem into three different networks:

- Group 1: lines with more than 50 kilometres from Depot B.
- Group 2: lines with more than 50 kilometres from Depot A.
- Group 3: lines with less than 50 kilometres from Depot A.

As mentioned, this division has also been considered. First, the influence of the parameters has been examined by group. After that, the independent results have been compared in order to get global results. The reasons why this process has been carried out are:

1. It allows a better understanding of the results since they can be compared with the reality.
2. The three networks have different characteristics, so, the behaviour of the parameters in different situations can be examined.

As for the values of the parameters refers, three different levels have been defined for each of them. Table 17 shows the description and the values of each parameter.

Table 17. Description and levels of the parameters in the DoE

Input parameter	Description	Level 0	Level 1	Level 2
iterations	Number of repetitions of the construction phase.	10	30	90
prob2	Probability to introduce to type 3 duties.	0.1	0.25	0.5
prob4	Probability to introduce to type 4 duties.	0.1	0.25	0.5

Input parameter	Description	Level 0	Level 1	Level 2
set_efficiency	Minimum efficiency (%) defined to consider a duty good enough to avoid the local search.	50	60	70
set_stop	Maximum time (minutes) allowed between pieces of work in the creation of the duties before the local search.	5	10	15

9.2 Analysis of the results obtained in the DoE

This section summarizes the results of the DoE classified by group (Table 18, Table 20 and Table 22). For further information, *Annexe D: Results of the DoE* contains the results obtained in the experiments concerning these parameters:

1. Solution in number of duties.
2. Runtime needed.
3. Effects of the parameters and their interactions in the number of duties.
4. Effects of the parameters and their interactions in the runtime.

Notice that prob2 and prob4 do not affect Group 3. This group corresponds to lines of less than 50 kilometres. Nevertheless, these two variables just impact on the scheduling of lines of more than 50 kilometres.

Each experiment has been replicated three times. As for the analysis of the results refers, based on the analysis of section 5.1, the solution is measured considering two aspects: the number of duties obtained and the execution time of the algorithm. It is considered that an effect or interaction is significant when the value of its p_value is smaller than 0,05.

Table 18. Results of Group 1

<table>
<tr><td>Duties</td><td>
<p>The statistically significant effects are <i>iterations</i>, <i>set_stop</i> and <i>prob4</i>. What happens is that the effects are very slight, a whole duty never comes down.</p>
<p>The interactions of the following factors are statistically significant: <i>iterations</i> * <i>set_stop, iterations</i> * <i>prob2, set_efficiency</i> * <i>set_stop, set_efficiency</i> * <i>prob4</i> and <i>set_stop</i> * <i>prob4</i>. However, it is remarkable that in these cases also, the influence in number of duties is negligible.</p>
</td></tr>
<tr><td>Runtime</td><td>
<p>The only parameter that has a really significant impact on the runtime is <i>iterations</i> (as expected). When its value is high (90), runtime increases.</p>
</td></tr>
<tr><td>Best values</td><td>
<p>On the basis of the results obtained, the parameters should be fed as shown in Table 19. The best results have been obtained with the selected values. Besides, the significant factors are highlighted.</p>
<p>Table 19. Appropriate values of the parameters in Group 1</p>
<table>
<tr><th>Parameter</th><th>Appropriate level</th><th>Appropriate value</th></tr>
<tr><td>iterations</td><td>2</td><td>90</td></tr>
<tr><td>set_efficiency</td><td>0</td><td>50</td></tr>
<tr><td>set_stop</td><td>1</td><td>10</td></tr>
<tr><td>prob2</td><td>1</td><td>0.25</td></tr>
<tr><td>prob4</td><td>1 and 2</td><td>0.25 and 0.5</td></tr>
</table>
</td></tr>
</table>

Table 20. Results of Group 2

<table>
<tr>
<td>Duties</td>
<td>The statistically significant effects are the number of <i>iterations, set_efficiency, set_stop</i> and <i>prob4</i>. However, the effects are very slight, except for <i>set_efficiency</i>.
It is significant that when the value of <i>set_efficiency</i> is high (70), the total number of duties increases.
The interactions of these pairs of factors are statistically significant: <i>set_efficiency</i> * <i>prob2</i>, <i>set_efficiency</i> * <i>prob4, set_stop</i> * <i>prob2</i>, <i>set_stop</i> * <i>prob4</i>. The results do not add information to the results achieved in the analysis of the effects.</td>
</tr>
<tr>
<td>Runtime</td>
<td>The only parameter that has a really significant impact on the runtime is <i>iterations</i> (as expected). When its value is high (90), runtime increases.</td>
</tr>
<tr>
<td>Best values</td>
<td>On the basis of the results obtained, the parameters should be fed as shown in Table 21. The best results have been obtained with the selected values. Besides, the significant factors are highlighted.
Table 21. Appropriate values of the parameters in Group 2
<table>
<tr><th>Parameter</th><th>Appropriate level</th><th>Appropriate value</th></tr>
<tr><td>iterations</td><td>2</td><td>90</td></tr>
<tr><td>set_efficiency</td><td>0</td><td>50</td></tr>
<tr><td>set_stop</td><td>0</td><td>5</td></tr>
<tr><td>prob2</td><td>0</td><td>0.1</td></tr>
<tr><td>prob4</td><td>2</td><td>0.5</td></tr>
</table>
</td>
</tr>
</table>

Table 22. Results of Group 3

<table>
<tr><td>Duties</td><td>The only statistically significant effect is <i>set_stop</i>. For small values of this parameter the number of duties is smaller.

The only statistically significant interaction is <i>iterations*set_efficiency</i>. When the value of <i>iterations</i> is high (90) and the value of <i>set_efficiency</i> is low (50) the number of duties decreases. However, when the value of <i>iterations</i> is low (10) and the value of <i>set_efficiency</i> is high (70), the number of duties increases.</td></tr>
<tr><td>Runtime</td><td>Only <i>iterations</i> has a really significant impact on the runtime (as expected). When its value is high (90), runtime increases.</td></tr>
<tr><td>Best values</td><td>On the basis of the results obtained, the parameters should be fed as shown in Table 23. The best results have been obtained with the selected values. Besides, the significant factors are highlighted.

Table 23. Appropriate values of the parameters in Group 3
<table>
<tr><th>Parameter</th><th>Appropriate level</th><th>Appropriate value</th></tr>
<tr><td>iterations</td><td>2 (interaction)</td><td>90</td></tr>
<tr><td>set_efficiency</td><td>0 (interaction)</td><td>50</td></tr>
<tr><td>set_stop</td><td>0</td><td>5</td></tr>
</table>
</td></tr>
</table>

But, do the input parameters perform similarly in all the cases? Is there any common result? A global comparison has been carried out in order to answer these questions. Table 24 shows the best values got for each group. The significant factors are highlighted.

Table 24. Comparison of the best values for each parameter

	Group 1	Group 2	Group 3
iterations	90	90	90
set_efficiency	50	50	50
set_stop	10	5	5
prob2	0.25	0.1	X
prob4	0.25 and 0.5	0.5	X

It is noteworthy that all parameters except *prob2* have influence on the result.

Besides, it seems that the most appropriate significant parameters coincide when scheduling lines of more than 50 kilometres and less than 50 kilometres. There is just a little difference with *set_stop* which values do not coincide in groups 1 and 2, both of more than 50 kilometres.

To end with, the most appropriate values are summarized in Table 25. Note that in cases that there are differences, Group 2 and Group 3 values have been chosen. These groups have more pieces of work than Group 1. So, it has considered that this fact could suit better the reality.

Table 25. Chosen best values of the variables

Parameter	Appropriate level	Appropriate value
iterations	2	90
set_efficiency	0	50
set_stop	0	5
prob2	0	0.1
prob4	2	0.5

10 Implementation and evaluation (Research Results)

Once the adjustment of the parameters is finished, the scheduling tool is ready to analyse the objectives that concern the project.

10.1 Implementation to evaluate Objective 1

The first objective corresponds to the development of the scheduling tool.

Objective 1: “To develop an efficient algorithm which minimizes in an acceptable execution time the Crew Scheduling Problem of an interurban passenger public transport bus company, allowing unlimited drivers’ reliefs that can occur at first, last or any other intermediate stop of a line”.

At the beginning of this project XYZ carries out its planning by grouping lines that share territorial areas, so that, manual scheduling is possible once the problem is divided. However, this manual planning has some associated problems. The most important ones are listed below:

1. Only 60% of the working hours of drivers correspond to effective working hours.
2. The scheduling takes a long time, nearly one week. Traffic managers must discuss and analyse diverse alternatives. In consequence, the personnel cost of this task is too high.
3. The information is not accessible.
4. Analysing new proposals or improvements is very arduous. Scheduling is too tight.

Nowadays, XYZ uses the scheduling tool developed in this research project to face its crew scheduling. Reliefs are allowed during services and pieces of work are defined based on the new relief points, as it was proposed in the model defined.

With the aim of evaluating the scheduling tool, some results obtained in both stages have been compared (Table 26).

Table 26. Initial and final scheduling results

	Initial scheduling (manual)	Final scheduling (scheduling tool)
Execution time (hours)	160	66
Duties	93	101
Time scheduled (hours)	761	1000
Driving time in service (hours)	529	610
Idle time (hours) + Break time(hours)	172	228

The difference between the indicator *Driving time in service* means that the operations to be scheduled are different. That is, there have been changes in the transport network of the company during the development of this research project. As this fact may affect the comparison of the results, in order to make it more objective, Table 27 shows the relative ratios obtained from the values in Table 26.

Table 27. Ratios of initial and final scheduling results

	Initial scheduling (manual)	Final scheduling (scheduling tool)	Comments
Execution time (hours)/ Time scheduled(hours)	0,30	0,11	The time needed to complete a schedule is much higher before the implementation.
Duties / Driving time in service (hours)	0,18	0,17	Less duties are needed in the second stage. The tool has helped to improve this parameter.
Driving time in service (hours)/ Duties	5,69	6,04	The driving time in service per duty is higher in the second stage. Duties could be more efficient or just longer (considering the next ratio, it seems so).
Driving time in service (hours) / Time scheduled (hours)	0,70	0,61	Duties in the manual scheduling are more efficient.
Time scheduled (hours) / Driving time in service (hours)	1,44	1,64	In the second stage, more time scheduled is needed to complete a driving hour. Longer but fewer duties are generated.
Idle time (hours) + Break time (hours) / Driving time in service (hours)	0,33	0,37	Break and idle-times are more when using the scheduling tool

10.2 Evaluation of Objective 1

The comparison between the initial schedule and the one obtained at the end of the project shows that allowing the handover during services modifies the result. In relation to this modification, there are two competing ideas:

1. In the second stage, the problem is solved with fewer duties.
2. Duties in the initial stage are more efficient than in the latest stage; they are shorter and idle-times are less.

In this case, company's goal, as well as the goal of this project, is reducing the number of drivers; so, the second scenario is more desirable. In short, we conclude that the fact of allowing reliefs during services results in a better performance of the crew scheduling in XYZ.

Besides, a detailed evaluation has been contrasted with the traffic manager of XYZ. The evaluation has consisted on an analysis of the evaluation criteria proposed in 5.1. Table 28 resumes its results.

Table 28. Evaluation of Objective 1

Criterion	Traffic manager’s evaluation
Model reliability and simplicity	The traffic manager of XYZ verifies that all the restrictions of the model are respected. The solution is totally feasible and no manual adjustment is necessary once the solution is obtained. So, the *Crew Scheduling Problem of an interurban passenger public transport bus company allowing unlimited drivers’ reliefs that can occur at first, last or any other intermediate stop of a line* is solved correctly when using the scheduling tool.
Solution time	The traffic manager of XYZ considers that the execution time is “*acceptable*”. The tool solves the problem much faster than manually. It has been proven that the work of one week could be carried out in one day. It supposes a reduction of around the 80% of the initial execution time.
Solution quality	Measured in duties, the tool obtains a similar result as manually. The traffic manager of XYZ considers that the solution is good enough.
Applicability	The tool answers to a real problem of a real company; it is able to answer a problem of large instances.

As result it can be concluded that the scheduling tool completes the specifications. In consequence, it can be determined that the first objective has been arranged.

10.3 Implementation to evaluate Objective 2 and Objective 3

As detailed in 4.1 and 4.2, the research objectives and hypothesis evaluate the influence in the scheduling of two factors:

1. The relief points at intermediate stops of the transport network.

   Objective 2: *"To evaluate the impact of allowing unlimited drivers' reliefs that can occur at first, last or any other intermediate stop of a line."*

   Hypothesis A: *allowing unlimited drivers' reliefs that can occur at first, last or any other intermediate stop of a line might improve the result of the crew scheduling. The quality of the solution will be measured in number of duties.*

2. The creation of schedules of lines of more and less than 50 kilometres jointly or separately.

   Objective 3: *"To evaluate the procedure of scheduling, that is, to evaluate if scheduling separately under different restrictions is better than scheduling globally under the most limited restrictions."*

   Hypothesis B: *referring to the scheduling procedure, scheduling separately under different restrictions achieves a better solution than scheduling globally under the toughest restrictions. The quality of the solution will be measured in number of duties.*

In order to evaluate these two aspects, six different scenarios have been created. Each scenario refers to a different transport network generated from data of the lines of XYZ.

With the aim of analysing different situations, the amount of lines of more and less than 50 kilometres is different in each network. The selection of lines has been randomly carried out. Table 29 resumes the composition of each network.

Table 29. Network composition

	Lines		Number of pieces of work			
			Reliefs at first, last or intermediate stops		Reliefs at first or last stops	
Networks	+50	-50	+50	-50	+50	-50
Network 1 (N1)	4	6	321	372	74	85
Network 2 (N2)	2	6	228	372	43	85
Network 3 (N3)	2	6	93	292	31	69
Network 4 (N4)	3	5	198	272	50	64
Network 5 (N5)	3	5	300	300	67	67
Network 6 (N6)	4	5	321	300	74	67

Besides, for each network four possible schedules have been created. These four schedules result of the combination of both analysis proposed in 5.2 and 5.3. On one hand, drivers' reliefs at intermediate stops are allowed or forbidden and on the other hand, the scheduling is carried out separately or globally. So, the four schedules correspond to:

1. Reliefs at first, last or intermediate stops and global scheduling.
2. Reliefs at first, last or intermediate stops and separated scheduling.
3. Reliefs at first or last stops and global scheduling.
4. Reliefs at first or last stops and separated scheduling.

In all cases, results of the schedules are measured in number of duties and runtime.

Finally, concerning to the variables of the algorithm, their values are the ones defined as the most appropriate in Table 25.

10.3.1 Results

As previously described, six different networks have been created and for each of them, four possible schedules have been evaluated. The results achieved are shown in Table 30 and in Table 31. Among the four results obtained for each network, the best result is highlighted.

Table 30. Results of the implementation (number of duties)

	N1	N2	N3	N4	N5	N6
Reliefs at first, last or intermediate stops and global scheduling	54	44	34	40	48	49
Reliefs at first, last or intermediate stops and separated scheduling	46	38	29	39	44	50
Reliefs at first or last stops and global scheduling	49	42	28	38	42	45
Reliefs at first or last stops and separated scheduling	47	38	27	35	41	47

In a general comparison of the results in number of duties (Table 30), the most remarkable idea is that the best result is not always obtained with the same scheduling criterion. In three cases, the best result is obtained when intermediate reliefs are forbidden and the scheduling is solved separately. However, in the three remaining cases, other options are the best. The best result is never got permitting intermediate reliefs and scheduling globally.

Table 31. Results of the implementation (runtime, minutes)

	N1	N2	N3	N4	N5	N6
Reliefs at first, last or intermediate stops and global scheduling	43	42	33	27	30	29
Reliefs at first, last or intermediate stops and separated scheduling	52	58	47	49	48	52
Reliefs at first or last stops and global scheduling	18	18	21	18	18	24
Reliefs at first or last stops and separated scheduling	34	48	45	40	42	40

In relation to the runtime (Table 31), it is observed that the scheduling tool is always faster when reliefs at intermediate stops are not allowed and the scheduling is solved globally.

However, what happens if the features are analysed independently? The following two sections answer to this question.

10.4 Evaluation of Objective 2 (reliefs at intermediate stops)

In relation to the second objective, there are two possible comparisons to be made: the results of scheduling globally and the results of scheduling separately.

Table 32: Evaluation Objective 2 (Number of duties)

	Global scheduling		Separated scheduling	
	Reliefs at first, last or intermediate stops	Reliefs at first or last stops	Reliefs at first, last or intermediate stops	Reliefs at first or last stops
N1	54	49	46	47
N2	44	42	38	38
N3	34	28	29	27
N4	40	38	39	35
N5	48	42	44	41
N6	49	45	50	47

Concerning duties (Table 32), results coincide in most of the cases, the best result is obtained when reliefs at intermediate stops are forbidden. However, there are two exceptions. In N1, when scheduling separately, the best result is obtained when reliefs at intermediate stops are allowed. In N2, the same result is got permitting or forbidding reliefs at intermediate stops.

Table 33. Evaluation Objective 2 (Runtime, minutes)

	Global scheduling		Separated scheduling	
	Reliefs at first, last or intermediate stops	Reliefs at first or last stops	Reliefs at first, last or intermediate stops	Reliefs at first or last stops
N1	43	18	52	34
N2	42	18	58	48
N3	33	21	47	45
N4	27	18	49	40
N5	30	18	48	42
N6	29	24	52	40

In relation to the runtime (Table 33), in all cases the results match. Planning is solved earlier when reliefs happen at first or last stops.

So, concerning Objective 2 and Hypothesis A, it could be concluded that:

- The second objective has been achieved.
- Hypothesis A is TRUE,
  The results have proven that in some cases (N1 and N2), allowing unlimited drivers' reliefs that can occur at first, last or any other intermediate stop of a line improves the result of the crew scheduling.

10.5 Evaluation of Objective 3 (scheduling procedure)

In relation to the third objective, there are two possible comparisons to be made: the results of scheduling permitting reliefs at intermediate stops and the results of scheduling forbidding reliefs at intermediate stops:

Table 34. Evaluation Objective 3 (Number of duties)

	Reliefs at first, last or intermediate stops		Reliefs at first or last stops	
	Global scheduling	Separated scheduling	Global scheduling	Separated scheduling
N1	54	46	49	47
N2	44	38	42	38
N3	34	29	28	27
N4	40	39	38	35
N5	48	44	42	41
N6	49	50	45	47

In what corresponds to duties (Table 34), it is observed that whether reliefs at intermediate stops are allowed or not, results coincide in all the networks except on N6. In N1, N2, N3, N4 and N5 the best results are got scheduling separately. In N6, the best result is got scheduling globally.

Table 35. Evaluation Objective 3 (Runtime, minutes)

	Reliefs at first, last or intermediate stops		Reliefs at first or last stops	
	Global scheduling	Separated scheduling	Global scheduling	Separated scheduling
N1	43	52	18	34
N2	42	58	18	48
N3	33	47	21	45
N4	27	49	18	40
N5	30	48	18	42
N6	29	52	24	40

Concerning to the runtime (Table 35), planning is always solved earlier when it is solved globally.

So, concerning Objective 3 and Hypothesis B, it could be concluded that:

- The third objective has been achieved.
- Hypothesis B is FALSE.
  The results have proven that in some cases (N6), scheduling separately under different restrictions is worse than scheduling globally under the most limited restrictions.

11 Conclusions and recommendations

This chapter is divided into two parts. First, research contributions are explained. Then, research limitations and future research lines are detailed.

11.1 Research contributions

The Crew Scheduling Problem is one of the five subproblems of the Vehicle and Crew Scheduling Problem, the one that focuses on the optimization of the staff. The impact a proper crew scheduling has on the operational cost of public transport companies is evident (Esclapés 2001, Bonrostro, Yusta 2003, Ernst et al. 2004, Van den Bergh et al. 2013, Ibarra-Rojas et al. 2015, Li et al. 2015). What stands out the literature is the difficulty of this task due to:

1. The crew scheduling is part of a larger problem, the Vehicle and Crew Scheduling Problem, which consists of five sub-problems: (1) Line Planning Problem, (2) Timetabling Problem, (3) Vehicle Scheduling Problem, (4) Crew Scheduling Problem and (5) Rostering Problem. Dependences among the solutions of these subproblems is a factor hindering the resolution of the Crew scheduling Problem.
2. The differences among network features, resources of companies, regulatory restrictions or labour agreements make the solutions particular to each company.

As it has been shown through the literature review, two research gaps have been found in the analysis of the literature.

The first aspect refers to the treatment of drivers' reliefs when solving the crew scheduling. It has been found that:

- Generally, relief points are located at the beginning of lines, at the end of lines or at the depot.
- Generally, reliefs are given at the end of the trips, that is, services are not broken in pieces of work. The same driver performs all the service.
- In some examples, vehicle changes in driver's duties are penalized. In some cases, the number of pieces of work is limited to 2 or 3.

The second research gap refers to the procedure of solving the crew scheduling when restrictions of duties vary depending on the type of service that is included in the duty. Specifically, the restrictions analysed correspond to breaks and working hours in drivers' duties. This research gap focuses on a situation that occurs in Spain, specifically, the crew scheduling of a public bus transport operator which operates inter-city lines that have more and less than 50 kilometres. In this casuistic, the operator must solve the Crew Scheduling Problem considering two regulations, the Regulation (EC) 561/2006 and the Spanish Real Decreto 902/2007.

After analysing the literature, it is observed that the division of duties considering differences in breaks or working hours and the procedure to solve this problem have not been studied in the bibliography.

So, with the aim of analysing both aspects, as the main research contribution, *“an efficient algorithm which minimizes in an acceptable execution time the Crew Scheduling Problem of an interurban passenger public transport bus company, allowing unlimited drivers’ reliefs that can occur at first, last or any other intermediate stop of a line*” has been developed (objective 1).

Once the performance of the scheduling tool has been validated, concerning the research gaps, it has been found that:

- Allowing unlimited drivers’ reliefs that can occur at first, last or any other intermediate stop of a line improves the result of the crew scheduling in some of the cases examined.
- Scheduling separately under different restrictions is better than scheduling globally under the most limited restrictions scheduling in some of the cases examined.

11.1.1 Other results

1. Definition of the key aspects of the problem:

Observation 1: suitable relief points. Reliefs can occur at any time of the duty and at any stop of the network. However, in order to simplify the problem and reduce the volume of possible solutions, the appropriate relief points could be selected as the first step. The selection would prioritize the following relief points: (1) relief points that are nearer from the depot, (2) relief points from where lines coincide and (3) relief points that fulfil the two previous conditions.

Observation 2: aspects to consider when introducing pieces of work in the duties. When joining two pieces of work as well as respecting the conditions related to the arrival time and place of the first piece of work and the departure time and place of the second piece of work, a new restriction is defined: the need to limit to one (or zero) the number of drivers’ reliefs in the same service. After analysing this question, it is observed that the pieces of work belonging to the same service can be at most part of two duties. So, considering the dependence among pieces of work of the same service is crucial in our model.

Observation 3: conditions to create split duties. With the aim of respecting the restrictions found in the literature, conditions needed to consider when scheduling split duties have been defined.

Observation 4: breaks in lines with more than 50 kilometres. When scheduling duties of lines of more than 50 kilometres, the duration of idle-times between services is an important feature. Considering this parameter, networks that have long idle-times and networks that have short idle-times can be distinguished. After analysing both cases, five types of duties could be distinguished. In all cases, the conditions that must be met in relation to driving times and rest times have been defined.

Observation 5: breaks in lines with less than 50 kilometres. The need of defining a new input parameter when planning duties with lines of less than 50 kilometres is stablished. *S_Min_Break* is the minimum duration of a break. This means that idle-times which are shorter than *S_Min_Break* will not be considered as break times.

2. The solution technique developed:

Different solution techniques are proposed in the literature to solve The Crew Scheduling Problem. In this context, it is outstanding that in most practical cases the amount of data and the corresponding execution time, make integer programming approaches unviable for obtaining the optimal solution (Alfares 2004). So, commonly heuristic procedures are chosen (Alfares 2004). There are some papers that compare the use of different metaheuristics in this problem (Ramalhinho et al. 1998, Dos Santos, Mateus 2007, Lopez et al. 2009). In these comparative papers, the most common used metaheuristics are GRASP, Tabu Search and Genetic Algorithm.

The solution technique chosen is GRASP. Its performance is joined to some input parameters. With the aim of defining their most appropriate values a DoE has been developed. However, it has been found that, in most cases, their values do not impact significantly.

11.2 Research limitations and future research lines

As with any research, several limitations should be taken into account in the analysis and interpretation of results. Thereby, this section collects the limitations of the present thesis.

The first objective of this thesis corresponds to the development of the scheduling tool. Although its operation has been validated in different networks, it seems a research limitation that it has only been contrasted with a particular company. For this reason, it is proposed to share the tool with other companies that have the same or a similar problem in order to gather more experiences and opinions about it.

Concerning the second objective, the results show that there are cases in which allowing intermediate reliefs improves the result. However, it is not clear on what situations or scenarios this factor impacts. For this reason, it is proposed that future researches focus on the characteristics of the network. The aim is to know if aspects as the length of pieces of work, the number of pieces of work, the proximity of relief points or the frequency of lines influence on the result.

Referring to the results of the third objective, they show that scheduling globally improves the result in some cases. As before, the characteristics of these cases have not been examined. So, it is proposed that future researches focus on this issue.

Finally, regarding to the algorithm:

1. The input parameters have been examined in three groups of different lines. It has concluded that they have little influence on the result (except for *iterations* in the runtime). It would be interesting to analyse the influence of these values in other scenarios, for example, in networks with lines of greater lengths. Longer pieces of work could impact more on working times and consequently, vary the creation of duties.
2. It has been explained that when creating duties the selection of the next pieces of work has a random component. This criterion is applied throughout all the creation of the duty. However, it seems that a targeted selection in some specific moments could improve the result. It is recommended to examine the selection at the end of the duty or when mandatory breaks are close, in order to prioritize pieces of work that suppose less deadheads.
3. In addition to GRASP, the bibliography gathers the suitability to solve the Crew Scheduling Problem of Tabu Search and the Genetic Algorithm. It seems interesting to develop other algorithms based on these two heuristics so that, the comparison of the results should be possible.

12 References

ABIDO, M. A. and ELAZOUNI, A. A. *Method of Generating Precedence-Preserving Crossover and Mutation Operations in Genetic Algorithms.* U.S. Patent No 8,250,007. 21 Aug., 2012.

ADAMSKI, A. Public Transport Lines Synchronization Problem*. AUTOMATICS; Sci.Bulletins of AGH*, 1993, no. 64, pp. 221-234.

ADAMUTHE, A. and BICHKAR, R. *Tabu Search for Solving Personnel Scheduling Problem.* , Communication, Information & Computing Technology (ICCICT), 2012.

ADDOU, I. and SOUMIS, F. Bechtold-Jacobs Generalized Model for Shift Scheduling with Extraordinary Overlap*. Annals of Operations Research*, 2007, vol. 155, no. 1, pp. 177-205. ISSN 0254-5330.

ALANDER, J. *Indexed Bibliography of Genetic Algorithms and Artificial Intelligence.* Technical Report, 2000a.

ALANDER, J. Indexed Bibliography of Genetic Algorithms in Operations Research, 2000b.

ALANDER, J. Indexed Bibliography of Genetic Algorithms in Computer Aided Design, 1995.

ALFARES, H. K. Survey, Categorization, and Comparison of Recent Tour Scheduling Literature*. Annals of Operations Research*, 2004, vol. 127, no. 1-4, pp. 145-175. ISSN 0254-5330.

AL-SULTAN, K. S.; HUSSAIN, M. F. and NIZAMI, J. S. A Genetic Algorithm for the Set Covering Problem*. Journal of the Operational Research Society*, 1996, pp. 702-709.

AYKIN, T. A Comparative Evaluation of Modeling Approaches to the Labor Shift Scheduling Problem*. European Journal of Operational Research*, 2000, vol. 125, no. 2, pp. 381-397. ISSN 0377-2217.

AYKIN, T. Optimal Shift Scheduling with Multiple Break Windows*. Management Science*, 1996, vol. 42, no. 4, pp. 591-602. ISSN 0025-1909.

AYTUG, H.; KHOUJA, M. and VERGARA, F. E. Use of Genetic Algorithms to Solve Production and Operations Management Problems: A Review*. International Journal of Production Research*, 2003, vol. 41, no. 17, pp. 3955-4009.

BÄCK, T. and HOFFMEISTER, F. Extended Selection Mechanisms in Genetic Algorithms, 1991.

BAKER, K.R. Workforce Allocation in Cyclical Scheduling Problems - Survey*. Operational Research Quarterly*, 1976, vol. 27, no. 1, pp. 155-167. ISSN 0030-3623.

BALL, M.; BODIN, L. and DIAL, R. A Matching Based Heuristic for Scheduling Mass Transit Crews and Vehicles*. Transportation Science*, 1983, vol. 17, no. 1, pp. 4-31.

BARABINO, B. *Transit Bus Route Network Design: A Model and its Application in a Real Network*, 2009.

BARTHOLDI, J. A Guaranteed-Accuracy Round-Off Algorithm for Cyclic Scheduling and Set Covering*. Operations Research*, 1981, vol. 29, no. 3, pp. 501-510.

BAR-YOSEF, A.; MARTENS, K. and BENENSON, I. A Model of the Vicious Cycle of a Bus Line*. Transportation Research Part B: Methodological*, 2013, vol. 54, pp. 37-50.

BATARCE, M.; MUÑOZ, J. C. and DE DIOS, J. Valuing Crowding in Public Transport: Implications for Cost-Benefit Analysis*. Transportation Research Part A: Policy and Practice*, 2016, vol. 91, pp. 358-378.

BEASLEY, J. E. and CHU, P. C. A Genetic Algorithm for the Set Covering Problem*. European Journal of Operational Research*, 1996, vol. 94, no. 2, pp. 392-404.

BECHTOLD, S.E. and JACOBS, L.W. Implicit Modeling of Flexible Break Assignments in Optimal Shift Scheduling*. Management Science*, 1990, vol. 36, no. 11, pp. 1339-1351. ISSN 0025-1909.

BERREBI, S.; WATKINS, K. and LAVAL, J. A Real-Time Bus Dispatching Policy to Minimize Passenger Wait on a High Frequency Route*. Transportation Research Part B: Methodological*, 2015, vol. 81, pp. 377-389.

BLICKLE, T. and THIELE, L. A Comparison of Selection Schemes used in Genetic Algorithms, 1995.

BLUM, C., et al. Hybrid Metaheuristics in Combinatorial Optimization: A Survey*. Applied Soft Computing*, 9, 2011, vol. 11, no. 6, pp. 4135-4151. ISSN 1568-4946.

BONROSTRO, J.A. and YUSTA, S. Estudio Comparativo De Diferentes Estrategias Metaheurísticas Para La Resolución Del Labor Scheduling Problem*. Estudios De Economía Aplicada*, 2003, vol. 21, no. 3, pp. 537-557.

BOOKBINDER, J. and DESILETS, A. Transfer Optimization in a Transit Network*. Transportation Science*, 1992, vol. 26, no. 2, pp. 106-118.

BORNDÖRFER, R. and KARBSTEIN, M. *A Direct Connection Approach to Integrated Line Planning and Passenger Routing.* , 2012 SCOPUS.

BORNDÖRFER, R.; LÖBEL, A. and WEIDER, S. A Bundle Method for Integrated Multi-Depot Vehicle and Duty Scheduling in Public Transit*. Computer-Aided Systems in Public Transport*, 2008, pp. 3-24.

BOUSSAÏD, I.; LEPAGNOT, J. and SIARRY, P. A Survey on Optimization Metaheuristics*. Information Sciences*, 2013, vol. 237, pp. 82-117.

BRUNO, G.; IMPROTA, G. and SGALAMBRO, A. Models for the Schedule Optimization Problem at a Public Transit Terminal*. OR Spectrum*, 2009, vol. 31, no. 3, pp. 465-481.

BUNTE, S. and KLIEWER, N. An Overview on Vehicle Scheduling Models*. Public Transport*, 2009, vol. 1, no. 4, pp. 299-317.

BUSSIECK, M. R.; LINDNER, T. and LÜBBECKE, M. E. A Fast Algorithm for Near Cost Optimal Line Plans*. Mathematical Methods of Operations Research*, 2004, vol. 59, no. 2, pp. 205-220.

CAI, X. and LI, K.N. A Genetic Algorithm for Scheduling Staff of Mixed Skills Under Multi-Criteria*. European Journal of Operational Research*, 2000, vol. 125, no. 2, pp. 359-369.

CATS, O.; WEST, J. and ELIASSON, J. A Dynamic Stochastic Model for Evaluating Congestion and Crowding Effects in Transit Systems*. Transportation Research Part B: Methodological*, 2016, vol. 89, pp. 43-57.

CAVIQUE, L.; REGO, C. and THEMIDO, I. New Heuristic Algorithms for the Crew Scheduling Problem*. Meta-Heuristics*, 1999, pp. 37-47.

CEDER, A. *Public Transit Planning and Operation: Modeling, Practice and Behavior.* CRC Press, 2016.

CEDER, A. *Public Transit Planning and Operation: Theory, Modeling and Practice.* Elsevier, Butterworth-Heinemann, 2007.

CEDER, A.; GOLANY, B. and TAL, O. Creating Bus Timetables with Maximal Synchronization*. Transportation Research Part A: Policy and Practice*, 2001, vol. 35, no. 10, pp. 913-928.

CEDER, A. and WILSON, N. Bus Network Design*. Transportation Research Part B: Methodological*, 1986, vol. 20, no. 4, pp. 331-344.

CEVALLOS, F. and ZHAO, F. Minimizing Transfer Times in Public Transit Network with Genetic Algorithm*. Transportation Research Record: Journal of the Transportation Research Board*, 2006, vol. 1971, no. 1, pp. 74-79.

CHAKROBORTY, P.; DEB, K. and SUBRAHMANYAM, P. S. Optimal Scheduling of Urban Transit Systems using Genetic Algorithms*. Journal of Transportation Engineering*, 1995, vol. 121, no. 6, pp. 544-553.

CHAUDHRY, S. and LUO, W. Application of Genetic Algorithms in Production and Operations Management: A Review*. International Journal of Production Research*, 2005, vol. 43, no. 19, pp. 4083-4101.

CHEN, M. and NIU, H. A Model for Bus Crew Scheduling Problem with Multiple Duty Types*. Discrete Dynamics in Nature and Society*, 2012, vol. 2012.

CHUANG, Y.; CHEN, C. and HWANG, C. A Real-Coded Genetic Algorithm with a Direction-Based Crossover Operator*. Information Sciences*, 2015, vol. 305, pp. 320-348.

CLAESSENS, M. T.; VAN DIJK, N. M. and Z, P. J. Cost Optimal Allocation of Rail Passenger Lines*. European Journal of Operational Research*, 1998, vol. 110, no. 3, pp. 474-489.

CRAWFORD, B.; CASTRO, C. and MONFROY, E. *A Two-Phase Set Partitioning Model for Passenger Transportation.* , 2009. ISBN 978-1-4244-5244-6.

DADUNA, J. R.; BRANCO, I. and PAIXÃO, J. *Computer-Aided Transit Scheduling: Proceedings of the Sixth International Workshop on Computer-Aided Scheduling of Public Transport.* Springer, 1995.

DADUNA, J. R. and VOSS, S. Computer-Aided Transit SchedulingSpringer, 1995*. Practical Experiences in Schedule Synchronization*, pp. 39-55.

DAGANZO, C. F. A Headway-Based Approach to Eliminate Bus Bunching: Systematic Analysis and Comparisons*. Transportation Research Part B: Methodological*, 2009, vol. 43, no. 10, pp. 913-921.

DAGANZO, C. F. and ANDERSON, P. Coordinating Transit Transfers in Real Time*. UCCONNECT Final Reports*, 2016.

DANTZIG, G. Letter to the Editor-A Comment on Edie's “Traffic Delays at Toll Booths”*. Journal of the Operations Research Society of America*, 1954, vol. 2, no. 3, pp. 339-341.

DARBY-DOWMAN, K. and MITRA, G. An Extension of Set Partitioning with Application to Scheduling Problems*. European Journal of Operational Research*, 1985, vol. 21, no. 2, pp. 200-205. ISSN 0377-2217.

DAVIS, L. *Handbook of Genetic Algorithms.* Van Nostrand Reinhold New York, 1991.

DE BRUECKER, P., et al. Workforce Planning Incorporating Skills: State of the Art*. European Journal of Operational Research*, 2015, vol. 243, no. 1, pp. 1-16.

DE CAUSMAECKER, P. and VANDEN BERGHE, G. A Categorisation of Nurse Rostering Problems*. Journal of Scheduling*, 2011, vol. 14, no. 1, pp. 3-16.

DE GROOT, S. and HUISMAN, D. Computer-aided Systems in Public TransportSpringer, 2008*. Vehicle and Crew Scheduling: Solving Large Real-World Instances with an Integrated Approach*, pp. 43-56.

DE LEONE, R.; FESTA, P. and MARCHITTO, E. A Bus Driver Scheduling Problem: A New Mathematical Model and a GRASP Approximate Solution*. Journal of Heuristics*, AUG, 2011, vol. 17, no. 4, pp. 441-466. ISSN 1381-1231.

DEB, K., et al. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II*. Transactions on Evolutionary Computation*, 2002, vol. 6, no. 2, pp. 182-197.

DESAULNIERS, G. and HICKMAN, M. Public Transit*. Transportation, Handbooks in Operations Research and Management Science*, 2003, pp. 69-127.

DIAS, T. G.; SOUSA, J. P. and CUNHA, J. F. *A Genetic Algorithm for the Bus Driver Scheduling Problem.* Citeseer, 2001.

DIAS, T. G.; DE SOUSA, J.,P. and CUNHA, J. F. Genetic Algorithms for the Bus Driver Scheduling Problem: A Case Study*. Journal of the Operational Research Society*, MAR, 2002, vol. 53, no. 3, pp. 324-335. ISSN 0160-5682.

DONG, G., et al. An Exact Algorithm for Vehicle Routing and Scheduling Problem of Free Pickup and Delivery Service in Flight Ticket Sales Companies Based on Set-Partitioning Model*. Journal of Intelligent Manufacturing*, OCT, 2011, vol. 22, no. 5, pp. 789-799. ISSN 0956-5515.

DOS SANTOS, A. and MATEUS, G. *Hybrid Approach to Solve a Crew Scheduling Problem: An Exact Column Generation Algorithm Improved by Metaheuristics.* IEEE, 2007.

DOU, X., et al. Time Control Point Strategy Coupled with Transfer Coordination in Bus Schedule Design*. Journal of Advanced Transportation*, 2016, vol. 50, no. 7, pp. 1336-1351.

DUARTE, A., et al. Greedy Randomized Adaptive Search Procedure with Exterior Path Relinking for Differential Dispersion Minimization. *Information Sciences*, 2015, vol. 296, pp. 46-60.

EASTON, F.E. and ROSSIN, DF. A Stochastic Goal Program for Employee Scheduling. *Decision Sciences*, SUM, 1996, vol. 27, no. 3, pp. 541-568. ISSN 0011-7315.

ERANKI, A. A Model to Create Bus Timetables to Attain Maximum Synchronization Considering Waiting Times at Transfer Stops, 2004.

ERNST, A.T., et al. Staff Scheduling and Rostering: A Review of Applications, Methods and Models. *European Journal of Operational Research*, FEB 16, 2004, vol. 153, no. 1, pp. 3-27. ISSN 0377-2217.

ESCLAPÉS, C. *Asignación De Conductores a Jornadas De Trabajo En Empresas De Transporte Colectivo.* . COROMINAS, A. ed., Universitat Politècnica de Catalunya. Departament d'Estadística i Investigació Operativa, 2000-11-14, 2001 TDX/TDR. ISBN 8469968823.

FEDORKO, G. and WEISZER, M. *Configuring the Parameters of Multi-Objective Evolutionary Algorithm for Integrated Timetabling and Vehicle Scheduling in Public Transport.* Jeseník , Czech Republic, EU ed. Carpathian Logistics Congress, 9.11.2012, 2012.

FEO, T. and RESENDE, M. Greedy Randomized Adaptive Search Procedures. *Journal of Global Optimization*, 1995, vol. 6, no. 2, pp. 109-133.

FESTA, P. and RESENDE, M. GRASP: Basic Components and Enhancements. *Telecommunication Systems*, 2011, vol. 46, no. 3, pp. 253-271.

FESTA, P. and RESENDE, M. An Annotated Bibliography of GRASP–Part I: Algorithms. *International Transactions in Operational Research*, 2009a, vol. 16, no. 1, pp. 1-24.

FESTA, P. and RESENDE, M. An Annotated Bibliography of GRASP–Part II: Applications. *International Transactions in Operational Research*, 2009b, vol. 16, no. 2, pp. 131-172.

FORREST, S. Genetic Algorithms. *ACM Computing Surveys (CSUR)*, 1996, vol. 28, no. 1, pp. 77-80.

FORREST, S. Genetic Algorithms: Principles of Natural Selection Applied to Computation. *Science*, 1993, vol. 261, no. 5123, pp. 872-878.

FORREST, S. and MITCHELL, M. What Makes a Problem Hard for a Genetic Algorithm? Some Anomalous Results and their Explanation*. Machine Learning*, 1993, vol. 13, no. 2-3, pp. 285-319.

FOUILHOUX, P., et al. Valid Inequalities for the Synchronization Bus Timetabling Problem*. European Journal of Operational Research*, 2016, vol. 251, no. 2, pp. 442-450.

FRELING, R.; HUISMAN, D. and WAGELMANS, APM. Models and Algorithms for Integration of Vehicle and Crew Scheduling*. Journal of Scheduling*, JAN-FEB, 2003, vol. 6, no. 1, pp. 63-85. ISSN 1094-6136.

FUEGENSCHUH, A. A Set Partitioning Reformulation of a School Bus Scheduling Problem*. Journal of Scheduling*, AUG, 2011, vol. 14, no. 4, pp. 307-318. ISSN 1094-6136.

GATTERMANN, P.; SCHIEWE, A. and SCHMIDT, M. The Line Planning Routing Game*. ERIM Report Series Reference no.ERS-2014-017-LIS*, 2014.

GEN, M.; and CHENG, R. *Genetic Algorithms and Engineering Optimization.* John Wiley & Sons, 2000.

GLOVER, F. Future Paths for Integer Programming and Links to Artificial Intelligence*. Computers & Operations Research*, 1986, vol. 13, no. 5, pp. 533-549. ISSN 0305-0548.

GLOVER, F.; LAGUNA, M. l. and MARTI, R. Principles of Tabu Search*. Approximation Algorithms and Metaheuristics*, 2007, vol. 23, pp. 1-12.

GOERIGK, M.; SCHACHTEBECK, M. and SCHÖBEL, A. Evaluating Line Concepts using Travel Times and Robustness: Simulations with the LinTim Toolbox*. Public Transport*, 2013, vol. 5, no. 3, pp. 267-284 SCOPUS.

GOLDBERG, D. *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison-Wesley, 1989.

GOLDBERG, D. and DEB, K. A Comparative Analysis of Selection Schemes used in Genetic Algorithms*. Foundations of Genetic Algorithms*, 1991, vol. 1, pp. 69-93.

GOLDBERG, D. and SASTRY, K. *Genetic Algorithms: The Design of Innovation.* Springer, 2007.

GOLDBERG, D., et al. Genetic Algorithms: A Bibliography*. Urbana*, 1997, vol. 51, pp. 61801.

GOMES, M.C.; CAVIQUE, L. and THEMIDO, I. The Crew Timetabling Problem: An Extension of the Crew Scheduling Problem*. Annals of Operations Research*, 2006, vol. 144, no. 1, pp. 111-132 SCOPUS.

GOOSSENS, J.H.M. Models and Algorithms for Railway Line Planning Problems, 2004.

HAASE, K.; DESAULNIERS, G. and DESROSIERS, J. Simultaneous Vehicle and Crew Scheduling in Urban Mass Transit Systems*. Transportation Science*, 2001, vol. 35, no. 3, pp. 286-303.

HARTLEY, T. A Glossary of Terms in Bus and Crew Scheduling*. Computer Scheduling of Public Transport*, 1981, pp. 353-359.

HICKMAN, M.; MIRCHANDANI, P. and VOSS, S. *Computer-Aided Systems in Public Transport.* Springer, 2008.

HOLLAND, J.H. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence.* U Michigan Press, 1975.

HUISMAN, D. *Integrated and Dynamic Vehicle and Crew Scheduling.* Erasmus School of Economics (ESE), 2004.

HUISMAN, D.; FRELING, R. and WAGELMANS, APM. Multiple-Depot Integrated Vehicle and Crew Scheduling*. Transportation Science*, NOV, 2005, vol. 39, no. 4, pp. 491-502. ISSN 0041-1655.

IBARRA-ROJAS, O.J.; LOPEZ IRARRAGORRI, F. and RIOS SOLIS, Y.A. Multiperiod Synchronization Bus Timetabling*. Transportation Science*, 2015.

IBARRA-ROJAS, O.J. and MUÑOZ, J.C. Synchronizing Different Transit Lines at Common Stops Considering Travel Time Variability Along the Day*. Transportmetrica A: Transport Science*, 2016, pp. 1-19.

IBARRA-ROJAS, O.J. and RIOS SOLIS, Y.A. Synchronization of Bus Timetabling*. Transportation Research Part B: Methodological*, 2012, vol. 46, no. 5, pp. 599-614.

IBARRA-ROJAS, O.J., et al. Planning, Operation, and Control of Bus Transport Systems: A Literature Review*. Transportation Research Part B: Methodological*, 7, 2015, vol. 77, no. 0, pp. 38-75. ISSN 0191-2615.

JARAMILLO-ALVAREZ, P.; GONZALEZ-CALDERON, C. and GONZALEZ-CALDERON, G.. Route Optimization of Urban Public Transportation*. Dyna*, 2013, vol. 80, no. 180, pp. 41-49.

KARLAFTIS, M. Handbook of Transport Modelling*. Journal of Transportation Engineering*, 2001, vol. 127, no. 6, pp. 546-546.

KASPI, M. and RAVIV, T. Service-Oriented Line Planning and Timetabling for Passenger Trains*. Transportation Science*, 2013, vol. 47, no. 3, pp. 295-311 SCOPUS.

KLIEWER, N.; AMBERG, B. and AMBERG, B. Multiple Depot Vehicle and Crew Scheduling with Time Windows for Scheduled Trips*. Public Transport*, 03/01, 2012, vol. 3, no. 3, pp. 213-244. ISSN 1866-749X.

KOLEN, A., et al. Interval Scheduling: A Survey*. Naval Research Logistics*, AUG, 2007, vol. 54, no. 5, pp. 530-543. ISSN 0894-069X.

KOZA, J. *Genetic Programming: On the Programming of Computers by Means of Natural Selection.* MIT press, 1992.

KWAN, A.; KWAN, R. and WREN, A. Driver Scheduling using Genetic Algorithms with Embedded Combinatorial Traits*. Computer-Aided Transit Scheduling*, 1999, pp. 81-102.

KWAN, R; KWAN, A and WREN, A. Evolutionary Driver Scheduling with Relief Chains*. Evolutionary Computation*, 2001, vol. 9, no. 4, pp. 445-460.

KWAN, R. Case Studies of Successful Train Crew Scheduling Optimisation*. Journal of Scheduling*, OCT, 2011, vol. 14, no. 5, pp. 423-434. ISSN 1094-6136.

LAPORTE, G. and MESA, J.A. Location science Springer, 2015*. The Design of Rapid Transit Networks*, pp. 581-594.

LAPORTE, G., et al. Planning Rapid Transit Networks*. Socio-Economic Planning Sciences*, 2011, vol. 45, no. 3, pp. 95-104 SCOPUS.

LEE, Y. H., et al. A Heuristic for Vehicle Fleet Mix Problem using Tabu Search and Set Partitioning*. Journal of the Operational Research Society*, JUN, 2008, vol. 59, no. 6, pp. 833-841. ISSN 0160-5682.

LEVINE, D. A Parallel Genetic Algorithm for the Set Partitioning Problem*. Meta-Heuristics*, 1996, pp. 23-35.

LI, H., et al. A Column Generation Based Hyper-Heuristic to the Bus Driver Scheduling Problem*. Discrete Dynamics in Nature and Society*, 2015, vol. 501, pp. 638104.

LI, J.; MIRCHANDANI, Pitu B.and BORENSTEIN, Denis. The Vehicle Rescheduling Problem: Model and Algorithms*. Networks*, 2007, vol. 50, no. 3, pp. 211-229.

LOPEZ, B., et al. Experimental Analysis of Optimization Techniques on the Road Passenger Transportation Problem*. Engineering Applications of Artificial Intelligence*, 2009, vol. 22, no. 3, pp. 374-388.

LYE, LM. *Tools and Toys for Teaching Design of Experiments Methodology.* , 2005.

MA, J., et al. A Case Study of Beijing Bus Crew Scheduling: A Variable Neighborhood-based Approach*. Journal of Advanced Transportation*, 2016, vol. 50, no. 4, pp. 434-445.

MA, J. et al. Fairness in Optimizing Bus-Crew Scheduling Process*. PloS One*, 2017, vol. 12, no. 11, pp. e0187623.

MALCZEWSKI, J. and RINNER, C. Multicriteria Decision Analysis in Geographic Information ScienceSpringer, 2015*. Heuristic Methods*, pp. 145-189.

MALHOTRA, R.; SINGH, N. and SINGH, Y. Genetic Algorithms: Concepts, Design for Optimization of Process Controllers*. Computer and Information Science*, 2011, vol. 4, no. 2, pp. p39.

MARTINS DE SÁ, E.; CONTRERAS, I. and CORDEAU, J.F. Exact and Heuristic Algorithms for the Design of Hub Networks with Multiple Lines*. European Journal of Operational Research*, 2015, vol. 246, no. 1, pp. 186-198.

MCNALLY, M. The Four Step Model, 2008.

MELIÁN, B.; MORENO, J. A. and MORENO, J. M. Metaheurısticas: Una Visión Global*. Revista Iberoamericana De Inteligencia Artificial*, 2003, vol. 19, pp. 7-28.

MELIÁN, B.; MORENO, J. A. and MORENO, J. M. Algoritmos Genéticos. Una Visión Práctica*. Números*, 2009, vol. 71, pp. 4.

MESQUITA, M.; PAIAS, A. and RESPÍCIO, A. Branching Approaches for Integrated Vehicle and Crew Scheduling*. Public Transport*, 2009, vol. 1, no. 1, pp. 21-37.

MESQUITA, M., et al. *Planning Bus Driver Rosters.* , 2012 SCOPUS.

MESQUITA, M. and PAIAS, A. Set Partitioning/Covering-Based Approaches for the Integrated Vehicle and Crew Scheduling Problem*. Computers & Operations Research*, MAY, 2008, vol. 35, no. 5, pp. 1562-1575. ISSN 0305-0548.

MICHAELIS, M. and SCHÖBEL, A. Integrating Line Planning, Timetabling, and Vehicle Scheduling: A Customer-Oriented Heuristic*. Public Transport*, 2009, vol. 1, no. 3, pp. 211-232 SCOPUS.

MILLER, B. and GOLDBERG, D. Genetic Algorithms, Selection Schemes, and the Varying Effects of Noise. *Evolutionary Computation*, 1996, vol. 4, no. 2, pp. 113-131.

MITCHELL, M. *An Introduction to Genetic Algorithms.* MIT press, 1998.

MONTGOMERY, D. Applications of Design of Experiments in Engineering. *Quality and Reliability Engineering International*, AUG, 2008, vol. 24, no. 5, pp. 501-502. ISSN 0748-8017.

ORTIZ-BOYER, D.; HERVÁS-MARTÍNEZ, C. and GARCÍA-PEDRAJAS, N. Improving Crossover Operator for Real-Coded Genetic Algorithms using Virtual Parents. *Journal of Heuristics*, 2007, vol. 13, no. 3, pp. 265-314.

ORTUZAR, J. and WILLUMSEN, L. *Modelling Transport.* , 1994.

PATRIKALAKIS, I. and XEROCOSTAS, D. Computer-Aided Transit SchedulingSpringer, 1992. *A New Decomposition Scheme of the Urban Public Transport Scheduling Problem*, pp. 407-425.

PENG K.; and SHEN, Y. An Evolutionary Algorithm Based on Grey Relational Analysis for Crew Scheduling. *Journal of Grey System*, -, 2016, vol. 28, no. 3, pp. 75 egs. ISSN 09573720.

PETERS, E; DE MATTA, R. and BOE, W.. Short-Term Work Scheduling with Job Assignment Flexibility for a Multi-Fleet Transport System. *European Journal of Operational Research*, JUL 1, 2007, vol. 180, no. 1, pp. 82-98. ISSN 0377-2217.

PETERSEN, H., et al. The Simultaneous Vehicle Scheduling and Passenger Service Problem. *Transportation Science*, 2012, vol. 47, no. 4, pp. 603-616.

PFETSCH, M. and BORNDÖRFER, R. Routing in Line Planning for Public Transport. *Operations Research Proceedings 2005*, 2006, pp. 405-410.

PIRLOT, M. General Local Search Methods. *European Journal of Operational Research*, 1996, vol. 92, no. 3, pp. 493-511.

PORTUGAL, R.; RAMALHINHO, H. and PINTO, J. Driver Scheduling Problem Modelling. *Public Transport*, 2009, vol. 1, no. 2, pp. 103-120.

QUAK, CB. Bus Line Planning. *Master's Thesis, TU Delft*, 2003.

RAMALHINHO, H.; PINTO, J. and PORTUGAL, R. Multiobjective Metaheuristics for the Bus-Driver Scheduling Problem. *Transportation Science*, AUG, 2001, vol. 35, no. 3, pp. 331-343. ISSN 0041-1655.

RAMALHINHO, H.; PINTO, J. and PORTUGAL, R. Metaheuristics for the Bus-Driver Scheduling Problem*. Transportation Science*, 1998, vol. 3, no. 35, pp. 331-343.

RAMLI, R.; IBRAHIM, H. and SHUNG, L. Innovative Crossover and Mutation in a Genetic Algorithm Based Approach to a Campus Bus Driver Scheduling Problem with Break Consideration and Embedded Overtime*. Appl.Math*, 2013, vol. 7, no. 5, pp. 1921-1928.

REKIK, M; CORDEAU, J.F. and SOUMIS, F. Implicit Shift Scheduling with Multiple Breaks and Work Stretch Duration Restrictions*. Journal of Scheduling*, FEB, 2010, vol. 13, no. 1, pp. 49-75. ISSN 1094-6136.

RESENDE, M. Metaheuristic Hybridization with GRASP*. Tutorials in Operations Research, INFORMS*, 2008.

RESENDE, M. and RIBEIRO, C. Search methodologiesSpringer, 2014*. GRASP: Greedy Randomized Adaptive Search Procedures*, pp. 287-312.

RESENDE, M. and RIBEIRO, C. Greedy Randomized Adaptive Search Procedures: Advances, Hybridizations, and Applications*. Handbook of Metaheuristics*, 2010, pp. 283-319.

SCHÖBEL, A. and SCHOLL, S. *Line Planning with Minimal Traveling Time.* Internationales Begegnungs-und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, 2006.

SCHÖBEL, A. Line Planning in Public Transportation: Models and Methods*. OR Spectrum*, 2012, vol. 34, no. 3, pp. 491-510 SCOPUS.

SCHOLL, S. Customer-Oriented Line Planning, 2005.

SHEN, Y. and KWAN, R. *Tabu Search for Driver Scheduling.* Springer, 2001.

SHEN, Y.; XU, J. and WU, X. Vehicle Scheduling Based on Variable Trip Times with Expected On-time Performance*. International Transactions in Operational Research*, 2016.

SHEN, Y., et al. Evolutionary Crew Scheduling with Adaptive Chromosomes*. Transportation Research Part B: Methodological*, 10, 2013, vol. 56, no. 0, pp. 174-185. ISSN 0191-2615.

SHEN, Y., et al. An Estimation of Distribution Algorithm for Public Transport Driver Scheduling*. International Journal of Operational Research*, 2017, vol. 28, no. 2, pp. 245-262.

SHEN, Y. and XIA, J. Integrated Bus Transit Scheduling for the Beijing Bus Group Based on a Unified Mode of Operation*. International Transactions in Operational Research*, MAR, 2009, vol. 16, no. 2, pp. 227-242. ISSN 0969-6016.

SIDHU, B. *Exploring Data Driven Models of Transit Travel Time and Delay.* California Polytechnic State University, San Luis Obispo, 2016.

SMITH, B. and WREN, A. A Bus Crew Scheduling System using a Set Covering Formulation*. Transportation Research Part A-Policy and Practice*, MAR, 1988, vol. 22, no. 2, pp. 97-108. ISSN 0965-8564.

SONG, C., et al. Improved Genetic Algorithm with Gene Recombination for Bus Crew-Scheduling Problem*. Mathematical Problems in Engineering*, 2015, vol. 2015.

STEINZEN, I., et al. A Time-Space Network Approach for the Integrated Vehicle- and Crew-Scheduling Problem with Multiple Depots*. Transportation Science*, AUG, 2010, vol. 44, no. 3, pp. 367-382. ISSN 0041-1655.

THANGIAH, S. Computer-Aided Scheduling of Public Transport*. Journal of Scheduling*, 2003, vol. 6, no. 4, pp. 415-418.

THIERENS, D.; and GOLDBERG, D. *Elitist Recombination: An Integrated Selection Recombination GA.* IEEE, 1994.

TIRACHINI, A.; HENSHER, D. and ROSE, J. Crowding in Public Transport Systems: Effects on Users, Operation and Implications for the Estimation of Demand*. Transportation Research Part A: Policy and Practice*, 2013, vol. 53, pp. 36-52.

VAN DEN BERGH, J., et al. Personnel Scheduling: A Literature Review*. European Journal of Operational Research*, 5/1, 2013, vol. 226, no. 3, pp. 367-385. ISSN 0377-2217.

VAQUERIZO, M. Computational Intelligence in Security for Information Systems 2010Springer, 2010*. Intelligent Methods for Scheduling in Transportation*, pp. 231-238.

VAQUERIZO, M.; BARUQUE, B. and CORCHADO, E. Springer Berlin Heidelberg, *Combining Metaheuristic Algorithms to Solve a Scheduling Problem*, pp. 381-391. ISBN 978-3-642-28930-9.

VISENTINI, M., et al. Review of Real-Time Vehicle Schedule Recovery Methods in Transportation Services*. Journal of Scheduling*, 2014, vol. 17, no. 6, pp. 541-567.

WEIDER, S. *Integration of Vehicle and Duty Scheduling in Public Transport.* Cuvillier Verlag, 2007.

WEISZER, M.; FEDORKO, G. and ČUJAN, Z. Multiobjective Evolutionary Algorithm for Integrated Timetable Optimization with Vehicle Scheduling Aspects*. Perne's Contacts*, 2010, vol. 5, no. 4, pp. 286-294.

WEISZER, M. Integrated Timetable and Scheduling Optimization with Multiobjective Evolutionary Algorithn*. The International Journal of TRANSPORT & LOGISTICS*, 2011, vol. 20, pp. 47.

WINSTON, W. and GOLDBERG, J. *Operations Research: Applications and Algorithms.* Duxbury press Boston, 2004.

WONG, R., et al. Optimizing Timetable Synchronization for Rail Mass Transit*. Transportation Science*, 2008, vol. 42, no. 1, pp. 57-69.

WOODWARD, J. and SWAN, J. *The Automatic Generation of Mutation Operators for Genetic Algorithms.* ACM, 2012.

WREN, A. and ROUSSEAU, J.M. *Bus Driver Scheduling—an Overview.* Springer, 1995.

WREN, A. and WREN, D. A Genetic Algorithm for Public Transport Driver Scheduling*. Computers & Operations Research*, 1, 1995, vol. 22, no. 1, pp. 101-110. ISSN 0305-0548.

YOUNAS, I. Using Genetic Algorithms for Large Scale Optimizationof Assignment, Planning and Rescheduling Problems, 2014.

YUAN, S., et al. A New Crossover Approach for Solving the Multiple Travelling Salesmen Problem using Genetic Algorithms*. European Journal of Operational Research*, 2013, vol. 228, no. 1, pp. 72-82.

ZHANG, B. and KIM, J. Comparison of Selection Methods for Evolutionary Optimization*. Evolutionary Optimization*, 2000, vol. 2, no. 1, pp. 55-70.

ZHIGANG, L., et al. Regional Bus Timetabling Model with Synchronization*. Journal of Transportation Systems Engineering and Information Technology*, 2007, vol. 7, no. 2, pp. 109-112.

13 Annexes

13.1 Annexe A: Lines and services of XYZ

Table 36 shows the number of services run per line as well as their distance in both directions, outward and return. Table 36 resumes the information divided into three groups: 1) services of Depot B, 2) services of Depot A with more than 50 kilometres and 3) services of Depot A with less than 50 kilometres.

Table 36. Lines and services of XYZ

Group	Code	Depot	Direction	Km.	Services	Total Km.
Group 1	A3915	Depot B	Outward	69	16	1.104
	A3915	Depot B	Return	68	16	1.088
	A3916	Depot B	Outward	73	16	1.168
	A3916	Depot B	Return	73	17	1.241
Group 2	A3912	Depot A	Outward	53	16	848
	A3912	Depot A	Return	52	17	884
	A3924	Depot A	Outward	53	8	424
	A3924	Depot A	Return	54	8	432
	A3925	Depot A	Outward	54	15	810
	A3925	Depot A	Return	55	15	825
Group 3	A3911	Depot A	Outward	35	17	595
	A3911	Depot A	Return	34	16	544
	A3913	Depot A	Outward	13	15	195
	A3913	Depot A	Return	11	15	165
	A3914	Depot A	Outward	12	18	216
	A3914	Depot A	Return	11	17	187

Group	Code	Depot	Direction	Km.	Services	Total Km.
	A3917	Depot A	Outward	36	21	756
	A3917	Depot A	Return	35	17	595
	A3918	Depot A	Outward	32	16	512
	A3918	Depot A	Return	33	15	495
	A3919	Depot A	Outward	39	10	390
	A3919	Depot A	Return	39	11	429
	A3920	Depot A	Outward	44	6	264
	A3920	Depot A	Return	44	5	220
	A3923	Depot A	Outward	42	16	672
	A3923	Depot A	Return	43	18	774
	A3926	Depot A	Outward	48	17	816
	A3926	Depot A	Return	48	16	768
	A3927	Depot A	Outward	35	15	525
	A3927	Depot A	Return	35	15	525
	A3928	Depot A	Outward	27	3	81
	A3928	Depot A	Return	29	3	87
	A3929	Depot A	Outward	21	4	84
	A3929	Depot A	Return	22	4	88
	A3931	Depot A	Outward	8	4	32
	A3931	Depot A	Return	8	3	24
	A3932	Depot A	Outward	6	101	606
	A3932	Depot A	Return	6	100	600
	A3933	Depot A	Outward	32	21	672

Group	Code	Depot	Direction	Km.	Services	Total Km.
	A3933	Depot A	Return	33	17	561

13.2 Annexe B: Selected relief points

This annexe contains the information about the final selected relief points. For each stop of the network of XYZ, three parameters have been examined (Table 37):

1. The time from the stop to the nearer depot.
2. The number of lines that coincide in the stop.
3. If the stop is a header or final stop.

The results show that there are 2 stops which meet the three criteria, 13 that meet two criteria and the remaining 20 meet a single criterion.

Table 37. Selected relief points

	Shortest time to depot (Depot A or Depot B)	Nº of lines that coincide	Is it a header or final stop?
Galdakao	15	6	Yes
Hospital	10	7	Yes
Artea	20	3	Yes
Bi/Abando	25	4	Yes
Bi/Bailen	25	3	Yes
Bi/Termibus	30	7	Yes
Durango	20	9	Yes
Ermua	25	4	Yes
Le/Surtidor	0	5	No
Metro/Galdakao	15	1	Yes
Mutriku	15	1	Yes
Ondarroa	15	2	Yes
Otxandio	35	1	Yes
Restop	10	6	No

	Shortest time to depot (Depot A or Depot B)	Nº of lines that coincide	Is it a header or final stop?
Zeanuri	25	3	Yes
Metro/Basauri	20	1	Yes
Amorebieta	5	2	No
Arrasate	40	1	Yes
Arrazola	40	1	Yes
Arrigorriaga	20	5	No
Baranbio	35	1	Yes
Basauri	20	5	No
Berriatua	0	2	No
Eibar	25	2	Yes
Elorrio	35	2	Yes
Garai	30	1	Yes
Ibarra	45	2	Yes
Leioa	30	1	Yes
Markina	15	2	No
Miraballes	20	5	No
Orduña	45	1	Yes
Orozko	40	2	Yes
Ubidea	45	1	Yes
Uribiarte	45	1	Yes
Zeberio	20	1	Yes

13.3 Annexe C: Pieces of work to the scheduled in XYZ

Table 38 resumes the information of the pieces of work to the scheduled in XYZ.

Table 38. Pieces of work to the scheduled in XYZ

Group	Code	Depot	Direction	Total POW/line
Group 1	A3915	Depot B	Outward	64
	A3915	Depot B	Return	64
	A3916	Depot B	Outward	32
	A3916	Depot B	Return	34
Group 2	A3912	Depot A	Outward	96
	A3912	Depot A	Return	102
	A3924	Depot A	Outward	24
	A3924	Depot A	Return	24
	A3925	Depot A	Outward	45
	A3925	Depot A	Return	45
Group 3	A3911	Depot A	Outward	85
	A3911	Depot A	Return	80
	A3913	Depot A	Outward	15
	A3913	Depot A	Return	15
	A3914	Depot A	Outward	18
	A3914	Depot A	Return	17
	A3917	Depot A	Outward	63
	A3917	Depot A	Return	51
	A3918	Depot A	Outward	64
	A3918	Depot A	Return	60
	A3919	Depot A	Outward	40

Group	Code	Depot	Direction	Total POW/line
	A3919	Depot A	Return	44
	A3920	Depot A	Outward	24
	A3920	Depot A	Return	20
	A3923	Depot A	Outward	48
	A3923	Depot A	Return	54
	A3926	Depot A	Outward	34
	A3926	Depot A	Return	32
	A3927	Depot A	Outward	30
	A3927	Depot A	Return	30
	A3928	Depot A	Outward	12
	A3928	Depot A	Return	12
	A3929	Depot A	Outward	16
	A3929	Depot A	Return	16
	A3931	Depot A	Outward	4
	A3931	Depot A	Return	3
	A3932	Depot A	Outward	101
	A3932	Depot A	Return	100
	A3933	Depot A	Outward	42
	A3933	Depot A	Return	34

13.4 Annexe D: Results of the DoE

13.4.1 Group 1

13.4.1.1 Results of experiments of Group 1

Table 39. Results of experiments of Group 1

	Values of parameters					Replica 1		Replica 2		Replica 3	
Exp.	iterations	set_efficiency	set_stop	prob2	prob4	Nº duties	Runtime (s)	Nº duties	Runtime (s)	Nº duties	Runtime (s)
1	0	0	0	0	0	27	120,29	26	125,39	26	115,73
2	0	0	0	0	1	26	120,39	26	126,78	26	114,39
3	0	0	0	0	2	26	123,58	26	130,48	26	111,09
4	0	0	0	1	0	27	118,38	27	114,83	27	114,38
5	0	0	0	1	1	25	123,5	28	126,1	28	111,51
6	0	0	0	1	2	28	123,12	27	122,99	27	115,44
7	0	0	0	2	0	28	116,34	27	122,5	27	112,1
8	0	0	0	2	1	26	128,05	25	122,17	25	111,92
9	0	0	0	2	2	26	130,83	27	122,32	27	108,53
10	0	0	1	0	0	28	124,07	27	123,69	27	117,06
11	0	0	1	0	1	27	122,57	28	126,22	28	118,37
12	0	0	1	0	2	28	126,75	27	128,5	27	118,39
13	0	0	1	1	0	27	121,76	28	126,53	28	111,93
14	0	0	1	1	1	25	125,75	29	124,33	29	115,49
15	0	0	1	1	2	26	125,49	27	121,77	27	117,58
16	0	0	1	2	0	26	119,03	27	128,11	27	112
17	0	0	1	2	1	26	119,35	26	119,89	26	116,16
18	0	0	1	2	2	26	125,52	26	125,98	26	116,21
19	0	0	2	0	0	27	122,85	26	129,55	26	112,63
20	0	0	2	0	1	28	123,16	25	136,18	25	123,64
21	0	0	2	0	2	27	126,66	27	127,61	27	114,32
22	0	0	2	1	0	25	125,56	26	122,87	26	116,66
23	0	0	2	1	1	27	122,51	28	123,58	28	115,95
24	0	0	2	1	2	27	127,37	26	127,75	26	121,26

	Values of parameters					Replica 1		Replica 2		Replica 3	
Exp.	iterations	set_efficiency	set_stop	prob2	prob4	Nº duties	Runtime (s)	Nº duties	Runtime (s)	Nº duties	Runtime (s)
25	0	0	2	2	0	26	120,79	26	123,74	26	116,77
26	0	0	2	2	1	26	123,74	26	126,02	26	115
27	0	0	2	2	2	27	128,25	25	132,02	25	118,66
28	0	2	0	0	0	25	126,38	26	123,85	26	117,57
29	0	2	0	0	1	26	122,13	26	129,87	26	116,33
30	0	2	0	0	2	27	126,72	28	129,18	28	121,15
31	0	2	0	1	0	26	127,59	27	122,4	27	115,65
32	0	2	0	1	1	25	122,59	27	126,56	27	113,67
33	0	2	0	1	2	26	132,07	26	125,83	26	117,95
34	0	2	0	2	0	26	124,34	28	130,28	28	118,64
35	0	2	0	2	1	27	123,31	27	126,37	27	116,38
36	0	2	0	2	2	26	129,26	26	130,73	26	118,96
37	0	2	1	0	0	26	126,23	26	126,05	26	117,41
38	0	2	1	0	1	27	125,15	25	125,82	25	115,46
39	0	2	1	0	2	27	126,62	26	127,26	26	122,22
40	0	2	1	1	0	29	126,04	26	127,14	26	114,54
41	0	2	1	1	1	25	121,41	26	124,19	26	119,26
42	0	2	1	1	2	26	125,38	26	127,61	26	122,12
43	0	2	1	2	0	28	123,96	27	126,5	27	117,26
44	0	2	1	2	1	27	127,01	27	130,62	27	115,21
45	0	2	1	2	2	27	131,04	25	126,2	25	118,95
46	0	2	2	0	0	28	122,09	26	122,06	26	117,64
47	0	2	2	0	1	26	131,32	28	127,78	28	113,04
48	0	2	2	0	2	27	127,44	28	129,53	28	117,76
49	0	2	2	1	0	27	124,31	25	125,2	25	116,14

	Values of parameters					Replica 1		Replica 2		Replica 3	
Exp.	iterations	set_efficiency	set_stop	prob2	prob4	Nº duties	Runtime (s)	Nº duties	Runtime (s)	Nº duties	Runtime (s)
50	0	2	2	1	1	26	126,13	25	126,07	25	118,06
51	0	2	2	1	2	28	125,31	27	126,15	27	119,89
52	0	2	2	2	0	27	123,98	28	130,48	28	118,79
53	0	2	2	2	1	26	125,41	27	133,1	27	119,55
54	0	2	2	2	2	28	126,12	28	130,68	28	115,94
55	0	1	0	0	0	25	123,82	25	130,19	25	115,8
56	0	1	0	0	1	26	126,55	25	132,05	25	119,49
57	0	1	0	0	2	25	127,85	24	134,42	24	118,52
58	0	1	0	1	0	26	122,85	28	131,95	28	115,42
59	0	1	0	1	1	27	126,96	25	131,84	25	121,07
60	0	1	0	1	2	25	123,5	27	133,8	27	120,56
61	0	1	0	2	0	26	124,38	27	129,35	27	116,58
62	0	1	0	2	1	23	126,94	28	130,4	28	117,37
63	0	1	0	2	2	26	122,41	25	134,74	25	121,54
64	0	1	1	0	0	26	128,73	28	131,45	28	120,47
65	0	1	1	0	1	26	124,46	25	131,85	25	122,88
66	0	1	1	0	2	25	131,96	26	133,67	26	124,1
67	0	1	1	1	0	25	131,6	28	132,2	28	120,15
68	0	1	1	1	1	28	132,98	27	124,3	27	121,3
69	0	1	1	1	2	26	127,67	27	132,01	27	123,37
70	0	1	1	2	0	27	130,23	28	125,55	28	120,98
71	0	1	1	2	1	26	132,8	25	128,99	25	120,63
72	0	1	1	2	2	24	129,64	28	131,21	28	116,78
73	0	1	2	0	0	28	128,53	25	123,98	25	117,99
74	0	1	2	0	1	25	126,32	25	129,21	25	117,37

	Values of parameters					Replica 1		Replica 2		Replica 3	
Exp.	iterations	set_efficiency	set_stop	prob2	prob4	Nº duties	Runtime (s)	Nº duties	Runtime (s)	Nº duties	Runtime (s)
75	0	1	2	0	2	26	130,48	28	134,76	28	120,21
76	0	1	2	1	0	24	128,81	27	128,98	27	118,92
77	0	1	2	1	1	27	124,02	25	131,54	25	121,04
78	0	1	2	1	2	27	128,8	26	131,8	26	122,48
79	0	1	2	2	0	29	127,07	27	130,9	27	120,26
80	0	1	2	2	1	27	130,31	28	133,58	28	117,73
81	0	1	2	2	2	25	130,65	26	129,44	26	116,24
82	1	0	0	0	0	29	334,61	27	344,23	27	309,15
83	1	0	0	0	1	26	334,65	27	349,58	27	317,41
84	1	0	0	0	2	26	341,48	26	351,65	26	309,38
85	1	0	0	1	0	27	330,61	26	337,18	26	317,26
86	1	0	0	1	1	27	329,79	29	337,2	29	317,02
87	1	0	0	1	2	27	336,35	25	345,15	25	323,33
88	1	0	0	2	0	27	338,43	26	343	26	309,42
89	1	0	0	2	1	25	342,45	27	348,55	27	315,83
90	1	0	0	2	2	27	342,58	27	346,6	27	317,22
91	1	0	1	0	0	28	329,41	27	353,67	27	305,56
92	1	0	1	0	1	28	339,38	28	359,06	28	319,61
93	1	0	1	0	2	27	343,06	26	349,85	26	317,1
94	1	0	1	1	0	27	334,52	28	344,51	28	313,16
95	1	0	1	1	1	26	336,67	27	348,47	27	312,92
96	1	0	1	1	2	27	345,38	27	350,38	27	322,73
97	1	0	1	2	0	26	331,22	27	357,93	27	313,17
98	1	0	1	2	1	27	341,75	26	347,63	26	312,66
99	1	0	1	2	2	25	339,35	27	363,25	27	314,86

	Values of parameters					Replica 1		Replica 2		Replica 3	
Exp.	iterations	set_efficiency	set_stop	prob2	prob4	Nº duties	Runtime (s)	Nº duties	Runtime (s)	Nº duties	Runtime (s)
100	1	0	2	0	0	27	346,85	26	345,05	26	309,39
101	1	0	2	0	1	28	352,15	25	346,07	25	312,78
102	1	0	2	0	2	26	346,53	26	348,59	26	320,66
103	1	0	2	1	0	26	349,38	28	349,58	28	314,04
104	1	0	2	1	1	28	353,7	25	346,05	25	314,95
105	1	0	2	1	2	26	348,81	26	358,62	26	317,37
106	1	0	2	2	0	25	337,23	28	350,78	28	310,32
107	1	0	2	2	1	28	335,62	26	349,24	26	320,97
108	1	0	2	2	2	27	345,4	26	361,23	26	314,48
109	1	2	0	0	0	26	341,05	27	348,93	27	321,65
110	1	2	0	0	1	27	356,92	26	357,74	26	314,92
111	1	2	0	0	2	26	362,36	25	358,38	25	321,83
112	1	2	0	1	0	26	349,64	25	350,62	25	323,02
113	1	2	0	1	1	27	342,05	27	347,09	27	322,85
114	1	2	0	1	2	29	348,15	27	362,51	27	326,12
115	1	2	0	2	0	25	343,85	28	349,16	28	325,08
116	1	2	0	2	1	28	347,62	27	357,32	27	324,38
117	1	2	0	2	2	25	352,61	27	353,42	27	326,06
118	1	2	1	0	0	26	353,74	27	357,81	27	316,01
119	1	2	1	0	1	25	360,56	27	355,72	27	316,09
120	1	2	1	0	2	26	363,14	29	384,09	29	327,77
121	1	2	1	1	0	26	347,12	26	353,2	26	322,84
122	1	2	1	1	1	27	343,37	25	353,88	25	318,77
123	1	2	1	1	2	26	357,44	25	357,38	25	324,25
124	1	2	1	2	0	26	354,8	26	355,64	26	327,16

	Values of parameters					Replica 1		Replica 2		Replica 3	
Exp.	iterations	set_efficiency	set_stop	prob2	prob4	Nº duties	Runtime (s)	Nº duties	Runtime (s)	Nº duties	Runtime (s)
125	1	2	1	2	1	27	348,86	28	352,96	28	316,12
126	1	2	1	2	2	27	351,51	25	357,88	25	322,36
127	1	2	2	0	0	28	352,14	26	354,76	26	327,44
128	1	2	2	0	1	25	353,95	26	356,48	26	325,59
129	1	2	2	0	2	25	358,59	24	360,53	24	329,69
130	1	2	2	1	0	28	344,47	25	362,42	25	328,26
131	1	2	2	1	1	26	354,34	26	364,93	26	318,21
132	1	2	2	1	2	27	354,02	30	359,66	30	333,42
133	1	2	2	2	0	25	352,34	26	355,9	26	323,06
134	1	2	2	2	1	26	351,62	26	370,33	26	327,05
135	1	2	2	2	2	26	359,32	26	368,19	26	335,31
136	1	1	0	0	0	26	343,65	27	343,45	27	329,96
137	1	1	0	0	1	26	354,09	26	352,62	26	324,82
138	1	1	0	0	2	28	356,37	27	350,97	27	316,98
139	1	1	0	1	0	23	345,79	26	349,96	26	322,82
140	1	1	0	1	1	26	348,8	28	351,42	28	325,9
141	1	1	0	1	2	25	357,82	26	366,82	26	331,36
142	1	1	0	2	0	26	353,1	25	350,6	25	325,73
143	1	1	0	2	1	26	341,14	26	358,55	26	326,78
144	1	1	0	2	2	25	361,18	26	358,14	26	326,9
145	1	1	1	0	0	27	344,02	28	345,15	28	323,71
146	1	1	1	0	1	26	351,36	26	348,71	26	321,07
147	1	1	1	0	2	25	351,74	26	368,54	26	322,28
148	1	1	1	1	0	26	351,79	27	361,73	27	318,32
149	1	1	1	1	1	27	351,55	26	364,13	26	324,4

	Values of parameters					Replica 1		Replica 2		Replica 3	
Exp.	iterations	set_efficiency	set_stop	prob2	prob4	Nº duties	Runtime (s)	Nº duties	Runtime (s)	Nº duties	Runtime (s)
150	1	1	1	1	2	29	348,83	25	361,38	25	318,69
151	1	1	1	2	0	27	362,02	28	358,28	28	325,2
152	1	1	1	2	1	27	359,16	25	364,31	25	322,16
153	1	1	1	2	2	26	359,19	28	363,16	28	330,09
154	1	1	2	0	0	26	345,32	27	350,82	27	324,75
155	1	1	2	0	1	26	350,87	26	351,25	26	327,29
156	1	1	2	0	2	27	341,46	25	349,99	25	328,92
157	1	1	2	1	0	26	345,38	27	345,74	27	327,42
158	1	1	2	1	1	26	351,5	27	352,42	27	322,75
159	1	1	2	1	2	28	358,68	25	353,84	25	328,55
160	1	1	2	2	0	27	346,36	25	342,24	25	317,79
161	1	1	2	2	1	25	350,19	27	356,35	27	319,81
162	1	1	2	2	2	26	357,81	25	353,85	25	321,78
163	2	0	0	0	0	25	985,48	26	1007,55	26	910,98
164	2	0	0	0	1	25	1020,21	25	1004,17	25	923,31
165	2	0	0	0	2	26	1008,48	25	1015,36	25	925,42
166	2	0	0	1	0	25	999,59	26	993,27	26	927,53
167	2	0	0	1	1	26	1033,61	25	1016,44	25	928,14
168	2	0	0	1	2	27	1026,9	26	997,84	26	935,44
169	2	0	0	2	0	24	1018,56	27	1008,88	27	925,58
170	2	0	0	2	1	25	1025,58	28	1031,54	28	948,67
171	2	0	0	2	2	27	1012,11	26	1048,08	26	952,73
172	2	0	1	0	0	25	1013,21	27	998,34	27	936,53
173	2	0	1	0	1	25	1019,35	26	1000,18	26	937,61
174	2	0	1	0	2	26	1014,94	26	1023,15	26	945,54

	Values of parameters					Replica 1		Replica 2		Replica 3	
Exp.	iterations	set_efficiency	set_stop	prob2	prob4	Nº duties	Runtime (s)	Nº duties	Runtime (s)	Nº duties	Runtime (s)
175	2	0	1	1	0	27	1003,58	25	1008,84	25	929,81
176	2	0	1	1	1	28	1011,91	26	1007,91	26	942,93
177	2	0	1	1	2	27	1027,14	27	1033,16	27	956,49
178	2	0	1	2	0	28	987,14	27	1018,04	27	938,12
179	2	0	1	2	1	26	1004,62	28	1030,12	28	943,38
180	2	0	1	2	2	26	1026,41	26	1052,18	26	951,63
181	2	0	2	0	0	26	1019,5	25	1030,57	25	941,5
182	2	0	2	0	1	26	1038,31	26	1041,14	26	940,99
183	2	0	2	0	2	27	1024,17	26	1039,98	26	948,43
184	2	0	2	1	0	25	1005,41	26	1005,14	26	941,06
185	2	0	2	1	1	27	1015,16	28	1043,31	28	947,61
186	2	0	2	1	2	27	1034,82	26	1044,07	26	939,04
187	2	0	2	2	0	28	1016,67	25	1017,44	25	931
188	2	0	2	2	1	28	1013,52	26	1012,74	26	934,84
189	2	0	2	2	2	27	1023,82	25	1029,12	25	947,66
190	2	2	0	0	0	25	1021,45	28	1013,97	28	938,09
191	2	2	0	0	1	29	1019,51	27	1037,79	27	937,89
192	2	2	0	0	2	27	1044,64	25	1045,22	25	942,66
193	2	2	0	1	0	26	1018,54	27	1024,42	27	936,13
194	2	2	0	1	1	25	1028,87	26	1022,21	26	930,52
195	2	2	0	1	2	28	1025,97	27	1044,47	27	960,79
196	2	2	0	2	0	27	1027,96	25	1004,34	25	928,75
197	2	2	0	2	1	26	1034,68	25	1013,87	25	952,21
198	2	2	0	2	2	25	1041,66	25	1016,93	25	952,43
199	2	2	1	0	0	26	1024,73	26	1004,14	26	940,06

	Values of parameters					Replica 1		Replica 2		Replica 3	
Exp.	iterations	set_efficiency	set_stop	prob2	prob4	Nº duties	Runtime (s)	Nº duties	Runtime (s)	Nº duties	Runtime (s)
200	2	2	1	0	1	25	1019,57	28	1017,64	28	945,31
201	2	2	1	0	2	26	1039,43	27	1019,63	27	945,86
202	2	2	1	1	0	25	1007,5	25	1008,86	25	932,11
203	2	2	1	1	1	25	1035,7	26	993,63	26	947,45
204	2	2	1	1	2	26	1019,21	26	1019,22	26	944,59
205	2	2	1	2	0	29	1022,44	27	1010,06	27	919,05
206	2	2	1	2	1	27	1020,56	25	1037,6	25	935
207	2	2	1	2	2	24	1028,79	24	1050,68	24	939,36
208	2	2	2	0	0	27	1030,99	27	1014,27	27	938,84
209	2	2	2	0	1	26	1006,92	27	1009,15	27	936,25
210	2	2	2	0	2	27	1038,44	26	1017,76	26	940,35
211	2	2	2	1	0	25	1020,25	25	1013,42	25	924,59
212	2	2	2	1	1	26	1017,43	25	1022,76	25	918,96
213	2	2	2	1	2	25	1041,8	28	1010,75	28	936,04
214	2	2	2	2	0	26	1023,78	26	992,54	26	924,58
215	2	2	2	2	1	25	1008,59	24	1008,31	24	940,13
216	2	2	2	2	2	24	1027,01	28	1031,25	28	936,46
217	2	1	0	0	0	27	1021,56	25	1016,89	25	918,68
218	2	1	0	0	1	26	1035,62	28	1020,6	28	931,62
219	2	1	0	0	2	26	1049,85	26	1018,64	26	939,25
220	2	1	0	1	0	26	1004,63	25	1020,47	25	911,35
221	2	1	0	1	1	24	994,75	28	1050,63	28	921,46
222	2	1	0	1	2	25	1002,33	24	1042,51	24	934,95
223	2	1	0	2	0	27	1007,15	27	1010,38	27	912,74
224	2	1	0	2	1	25	1006,71	25	1005,08	25	924,58

	Values of parameters					Replica 1		Replica 2		Replica 3	
Exp.	iterations	set_efficiency	set_stop	prob2	prob4	Nº duties	Runtime (s)	Nº duties	Runtime (s)	Nº duties	Runtime (s)
225	2	1	0	2	2	26	1012,14	25	1019,52	25	944,85
226	2	1	1	0	0	28	1010,09	25	1005,13	25	933,19
227	2	1	1	0	1	26	1027,66	28	1028,76	28	929,01
228	2	1	1	0	2	25	1029,73	26	1029,46	26	937,99
229	2	1	1	1	0	26	990,63	26	1029,49	26	922,74
230	2	1	1	1	1	26	1016,1	25	1008,76	25	921,4
231	2	1	1	1	2	26	1026,56	26	1012,13	26	935,87
232	2	1	1	2	0	27	993,06	25	987,06	25	924,76
233	2	1	1	2	1	24	1024,35	25	992,83	25	936,98
234	2	1	1	2	2	25	1016,42	27	993,57	27	941,65
235	2	1	2	0	0	28	1024,65	26	1014,44	26	931,99
236	2	1	2	0	1	26	1019,47	27	1014,7	27	935,55
237	2	1	2	0	2	27	1022,55	26	1042,93	26	945,82
238	2	1	2	1	0	25	1018,76	26	1017,94	26	923,39
239	2	1	2	1	1	26	967,56	25	1000,18	25	931,51
240	2	1	2	1	2	25	1014,92	26	1005	26	949,95
241	2	1	2	2	0	26	1012,47	28	1002,54	28	932,07
242	2	1	2	2	1	27	1002,47	26	1010,9	26	927,98
243	2	1	2	2	2	27	1020,04	26	1009,99	26	925,34

13.4.1.2 Effects of the parameters and their interactions in relation to the number of duties (Group 1)

Table 40. General Linear Model: duties versus factors (Group1).

Factor	Type	Levels	Values
iterations	fixed	3	0. 1. 2 / 10; 30; 90
set_efficiency	fixed	3	0. 1. 2 / 50; 60; 70
set_stop	fixed	3	0. 1. 2 / 5; 10; 15
prob2	fixed	3	0. 1. 2 / 0.1; 0.25; 0.5
prob4	fixed	3	0. 1. 2 / 0.1; 0.25; 0.5

Table 41. Analysis of Variance for duties, using Adjusted SS for Tests (Group1).

Source	DF	Seq SS	Adj SS	Adj MS	F
Iterations	2	100,686	100,686	50,343	33,52
set_efficiency	2	4,529	4,529	2,265	1,51
set_stop	2	15,056	15,056	7,528	5,01
prob2	2	4,521	4,521	2,261	1,51
prob4	2	31,410	31,410	15,705	10,46
iterations*set_efficiency	4	14,145	14,145	3,536	2,35
iterations*set_stop	4	21,001	21,001	5,250	3,50
iterations*prob2	4	32,302	32,302	8,075	5,38
iterations*prob4	4	11,981	11,981	2,995	1,99
set_efficiency*set_stop	4	19,108	19,108	4,777	3,18
set_efficiency*prob2	4	13,075	13,075	3,269	2,18
set_efficiency*prob4	4	15,940	15,940	3,985	2,65
set_stop*prob2	4	8,499	8,499	2,125	1,41
set_stop*prob4	4	24,623	24,623	6,156	4,10
prob2*prob4	4	9,355	9,355	2,339	1,56

Table 42. P values for duties (Group1).

Source	P value
Iterations	0,000
set_efficiency	0,222
set_stop	0,007
prob2	0,223
prob4	0,000
iterations*set_efficiency	0,053
iterations*set_stop	0,008
iterations*prob2	0,000
iterations*prob4	0,094
set_efficiency*set_stop	0,013

Source	P value
set_efficiency*prob2	0,070
set_efficiency*prob4	0,032
set_stop*prob2	0,227
set_stop*prob4	0,003
prob2*prob4	0,184

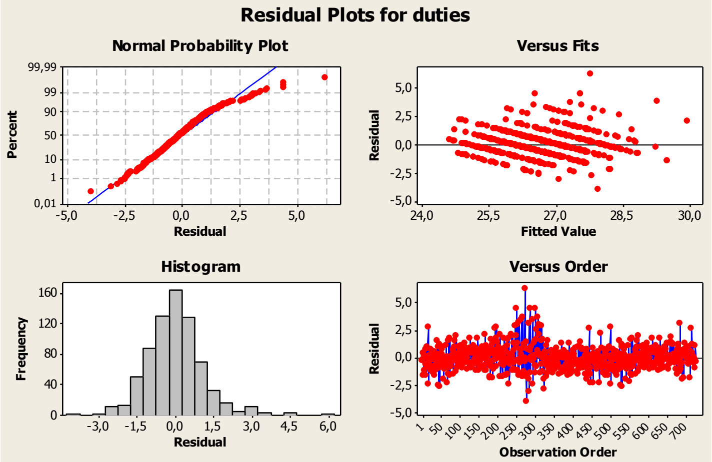


Figure 28. Residual Plots for Duties (Group 1).

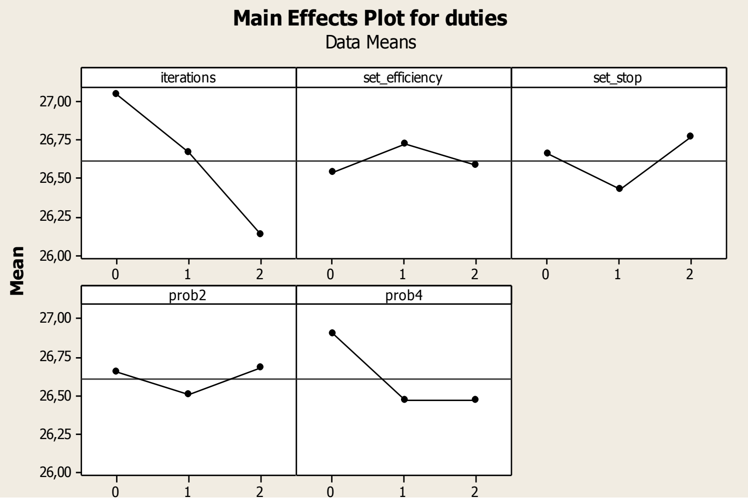


Figure 29. Main Effects Plot for Duties. Data Means (Group 1)

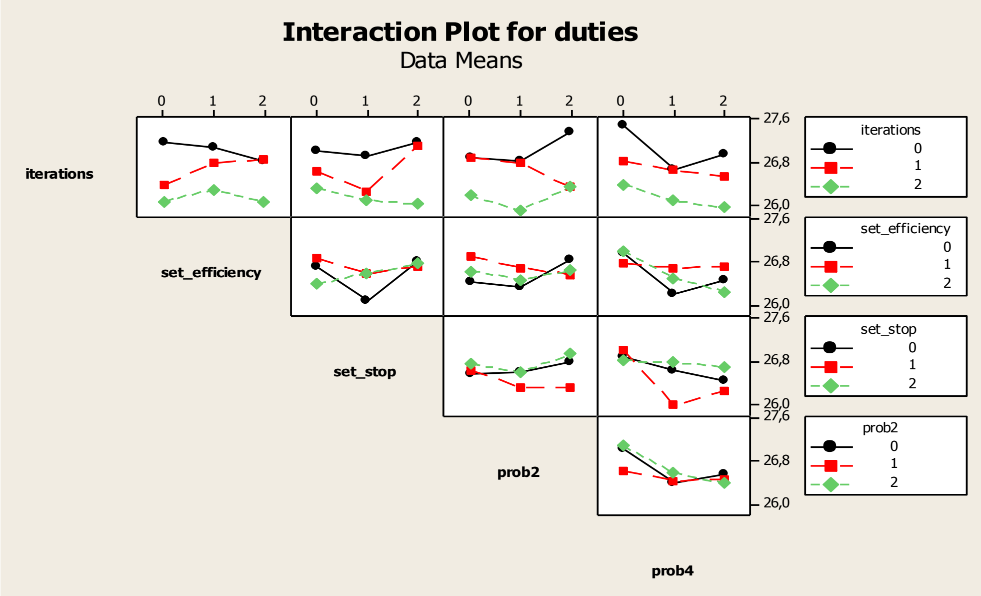


Figure 30. Interaction Plot for Duties. Data Means (Group 1).

13.4.1.3 Effects of the parameters and their interactions in relation to the runtime (Group 1).

Table 43. General Linear Model: comp.time versus factors (Group 1).

Factor	Type	Levels	Values
iterations	fixed	3	0. 1. 2 / 10; 30; 90
set_efficiency	fixed	3	0. 1. 2 / 50; 60; 70
set_stop	fixed	3	0. 1. 2 / 5; 10; 15
prob2	fixed	3	0. 1. 2 / 0.1; 0.25; 0.5
prob4	fixed	3	0. 1. 2 / 0.1; 0.25; 0.5

Table 44. Analysis of Variance for comp.time, using Adjusted SS for Tests (Group1).

Source	DF	Seq SS	Adj SS	Adj MS	F
Iterations	2	15213128	15213128	7606564	767280,43
set_efficiency	2	8029	8029	4014	404,94
set_stop	2	2753	2753	1377	138,87
prob2	2	195	195	97	9,83
prob4	2	1612	1612	806	81,29
iterations*set_efficiency	4	2739	2739	685	69,06
iterations*set_stop	4	1565	1565	391	39,46
iterations*prob2	4	364	364	91	9,18
iterations*prob4	4	1324	1324	331	33,39
set_efficiency*set_stop	4	2627	2627	657	66,25
set_efficiency*prob2	4	525	525	131	13,25
set_efficiency*prob4	4	21	21	5	0,52

Source	DF	Seq SS	Adj SS	Adj MS	F
set_stop*prob2	4	310	310	78	7,82
set_stop*prob4	4	20	20	5	0,50
prob2*prob4	4	24	24	6	0,61

Table 45. P values for comp.time (Group1).

Source	P
iterations	0,000
set_efficiency	0,000
set_stop	0,000
prob2	0,000
prob4	0,000
iterations*set_efficiency	0,000
iterations*set_stop	0,000
iterations*prob2	0,000
iterations*prob4	0,000
set_efficiency*set_stop	0,000
set_efficiency*prob2	0,000
set_efficiency*prob4	0,723
set_stop*prob2	0,000
set_stop*prob4	0,737
prob2*prob4	0,657

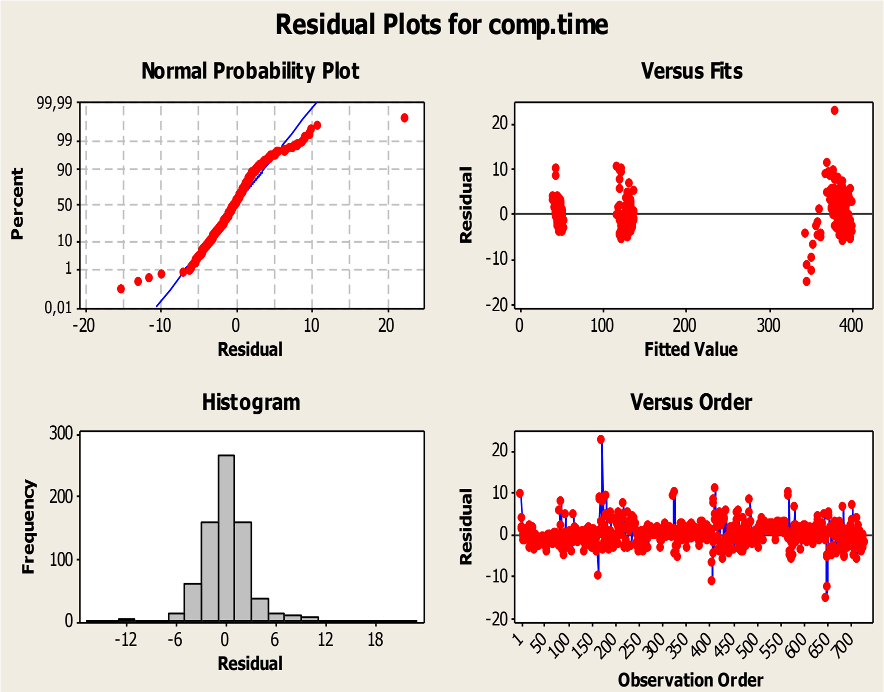


Figure 31. Residual Plots for Computational Time (Group 1).

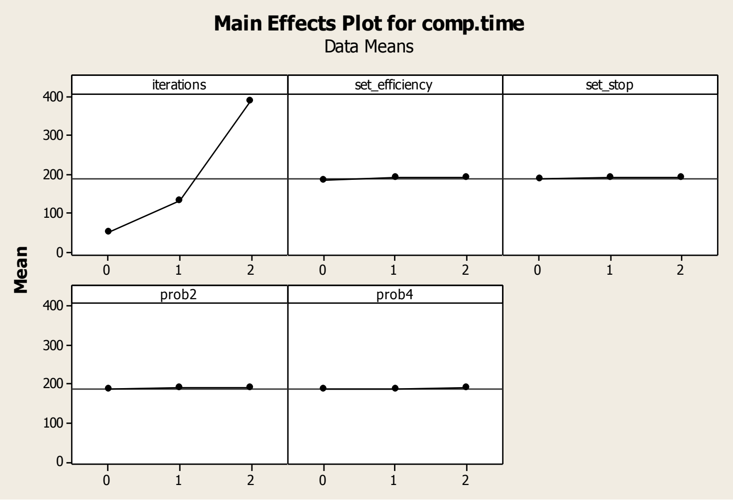


Figure 32. Main Effects Plot for Computational Time. Data Means (Group 1).

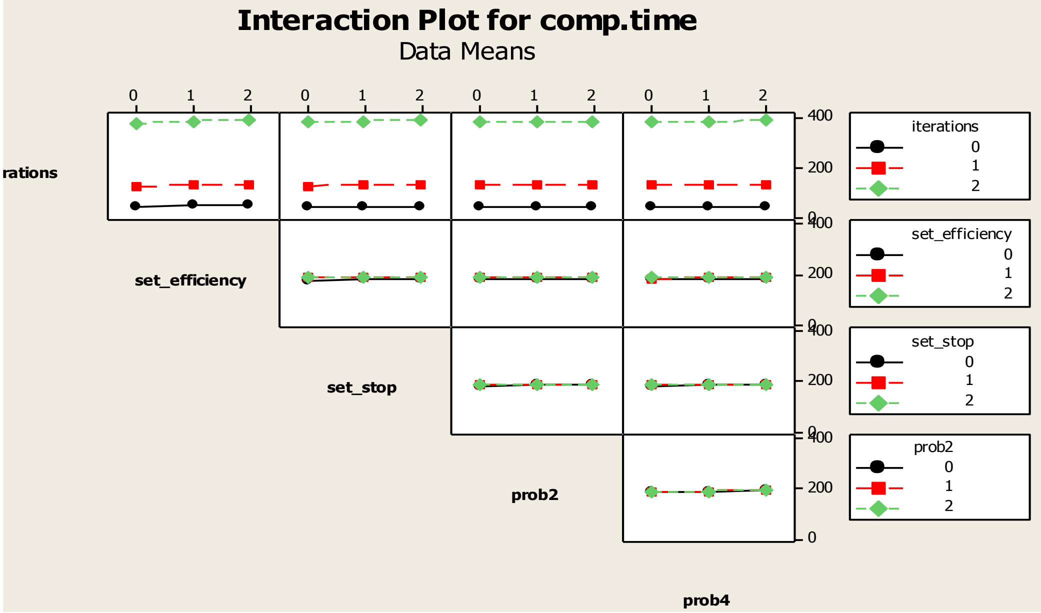


Figure 33. Interaction Plot for Computational Time. Data Means (Group 1)

13.4.2 Group 2

13.4.2.1 Results of experiments of Group 2

Table 46. Results of experiments of Group 2

	Values of parameters					Replica 1		Replica 2		Replica 3	
Exp.	iterations	set_efficiency	set_stop	prob2	prob4	Nº duties	Runtime (s)	Nº duties	Runtime (s)	Nº duties	Runtime (s)
1	0	0	0	0	0	43	157,09	43	144,44	43	146,69
2	0	0	0	0	1	41	149,17	41	144,66	41	146,62
3	0	0	0	0	2	41	148,6	41	145,7	41	145,77
4	0	0	0	1	0	42	144,55	42	143,66	42	144,8
5	0	0	0	1	1	43	145,57	43	144,65	43	143,66
6	0	0	0	1	2	45	145,78	45	145,47	45	144,75
7	0	0	0	2	0	43	145,69	43	145,73	43	145,42
8	0	0	0	2	1	45	142,97	45	142,74	45	142,32
9	0	0	0	2	2	43	145,41	43	145,32	43	144,54
10	0	0	1	0	0	46	149,18	46	144,72	46	143,99
11	0	0	1	0	1	44	144,8	44	145,02	44	144,11
12	0	0	1	0	2	41	146,72	41	145,69	41	146,03
13	0	0	1	1	0	42	145,22	42	144,64	42	144,57
14	0	0	1	1	1	44	143,99	44	145,06	44	144,83
15	0	0	1	1	2	43	147,03	43	146,6	43	146,56
16	0	0	1	2	0	40	143,33	40	142,99	40	142,88
17	0	0	1	2	1	44	143,87	44	144,17	44	143,61
18	0	0	1	2	2	42	145,03	42	145,77	42	144,29
19	0	0	2	0	0	44	144,16	44	144,28	44	143,13
20	0	0	2	0	1	44	144,27	44	144,38	44	143,29
21	0	0	2	0	2	44	146,75	44	146,42	44	146,04
22	0	0	2	1	0	46	157	46	145,68	46	145,36
23	0	0	2	1	1	42	201	42	146,09	42	202,84
24	0	0	2	1	2	42	200,56	42	147,22	42	207,9

	Values of parameters					Replica 1		Replica 2		Replica 3	
Exp.	iterations	set_efficiency	set_stop	prob2	prob4	Nº duties	Runtime (s)	Nº duties	Runtime (s)	Nº duties	Runtime (s)
25	0	0	2	2	0	43	197,88	43	145,92	43	214,62
26	0	0	2	2	1	44	196,66	44	146,72	44	212,46
27	0	0	2	2	2	41	201,97	41	146,06	41	207,26
28	0	2	0	0	0	43	206,14	43	145,68	43	208,65
29	0	2	0	0	1	46	193,5	46	145,61	46	212,3
30	0	2	0	0	2	44	209,95	44	147,21	44	216,02
31	0	2	0	1	0	46	202,78	46	146,93	46	214,67
32	0	2	0	1	1	41	202,04	41	146,4	41	222,66
33	0	2	0	1	2	44	208,16	44	148,08	44	221,77
34	0	2	0	2	0	45	198,59	45	146,34	45	221,6
35	0	2	0	2	1	43	204,61	43	146,53	43	216,2
36	0	2	0	2	2	45	210,31	45	146,05	45	218,29
37	0	2	1	0	0	44	208,71	44	147,44	44	207,26
38	0	2	1	0	1	42	210,4	42	147,35	42	216,96
39	0	2	1	0	2	43	203,3	43	147,21	43	212,21
40	0	2	1	1	0	44	200,71	44	155,92	44	215,05
41	0	2	1	1	1	43	203,7	43	182,72	43	204,61
42	0	2	1	1	2	46	196,51	46	193,83	46	214,62
43	0	2	1	2	0	42	198,19	42	198,48	42	209,14
44	0	2	1	2	1	42	196,93	42	208,07	42	214,44
45	0	2	1	2	2	44	206,64	44	203,08	44	208,05
46	0	2	2	0	0	46	199,03	46	193,21	46	214,11
47	0	2	2	0	1	42	212,81	42	199,82	42	211,2
48	0	2	2	0	2	44	202,85	44	200,17	44	218,26
49	0	2	2	1	0	44	197,6	44	200,08	44	219,16
50	0	2	2	1	1	42	206,31	42	192,74	42	216,36

	Values of parameters					Replica 1		Replica 2		Replica 3	
Exp.	iterations	set_efficiency	set_stop	prob2	prob4	Nº duties	Runtime (s)	Nº duties	Runtime (s)	Nº duties	Runtime (s)
51	0	2	2	1	2	42	207,19	42	199,15	42	209,38
52	0	2	2	2	0	43	211,99	43	191,97	43	211,4
53	0	2	2	2	1	42	206,6	42	200,81	42	211,63
54	0	2	2	2	2	42	203,94	42	200,29	42	216,56
55	0	1	0	0	0	44	206,45	44	198,83	44	213,45
56	0	1	0	0	1	42	205,22	42	203,94	42	216,59
57	0	1	0	0	2	44	211,04	44	198,58	44	217,37
58	0	1	0	1	0	45	202,19	45	205,84	45	220
59	0	1	0	1	1	40	212,21	40	200,4	40	223,47
60	0	1	0	1	2	43	207,15	43	202,34	43	212,53
61	0	1	0	2	0	42	199,98	42	198,56	42	217,37
62	0	1	0	2	1	41	200,46	41	201,69	41	204,49
63	0	1	0	2	2	42	203,91	42	201,96	42	215,81
64	0	1	1	0	0	44	204,26	44	195,66	44	218,82
65	0	1	1	0	1	44	210,6	44	201,14	44	216,54
66	0	1	1	0	2	44	203,85	44	201,2	44	215,75
67	0	1	1	1	0	40	210,8	40	197,26	40	207,31
68	0	1	1	1	1	42	205,11	42	203,42	42	226,32
69	0	1	1	1	2	42	206,16	42	196,74	42	211,43
70	0	1	1	2	0	45	206,27	43	190,24	43	205,33
71	0	1	1	2	1	40	202,48	40	195,86	40	223,42
72	0	1	1	2	2	43	207,01	43	211,02	43	229,68
73	0	1	2	0	0	44	207,85	44	204,64	44	221,34
74	0	1	2	0	1	44	203,09	44	203,02	44	212,02
75	0	1	2	0	2	43	204,31	43	210,27	43	206,95
76	0	1	2	1	0	44	204,44	44	196,16	44	217,4

	Values of parameters					Replica 1		Replica 2		Replica 3	
Exp.	iterations	set_efficiency	set_stop	prob2	prob4	Nº duties	Runtime (s)	Nº duties	Runtime (s)	Nº duties	Runtime (s)
77	0	1	2	1	1	44	207,2	44	193,55	44	215,04
78	0	1	2	1	2	42	201,03	42	198,8	42	213,3
79	0	1	2	2	0	40	205,88	40	197,31	40	212,93
80	0	1	2	2	1	44	206,7	44	202,41	44	217,01
81	0	1	2	2	2	39	213,58	39	206,51	39	227,21
82	1	0	0	0	0	43	557,95	43	544,29	43	589,2
83	1	0	0	0	1	42	558,59	42	557,83	42	601,77
84	1	0	0	0	2	43	570,15	43	560,89	43	600,46
85	1	0	0	1	0	43	573,53	43	557,91	43	597,3
86	1	0	0	1	1	40	580,98	40	546,84	40	596,81
87	1	0	0	1	2	43	566,48	43	560,96	43	590,33
88	1	0	0	2	0	43	579,92	43	553,29	43	599,97
89	1	0	0	2	1	42	551,6	42	539,59	42	598,1
90	1	0	0	2	2	44	563,77	44	559,21	44	618,23
91	1	0	1	0	0	43	559,26	43	568,56	43	589,99
92	1	0	1	0	1	42	559	42	564,68	42	609,64
93	1	0	1	0	2	43	572,14	43	544,16	43	613,78
94	1	0	1	1	0	43	588,25	43	523,55	43	599,55
95	1	0	1	1	1	43	584,47	43	533,61	43	598,81
96	1	0	1	1	2	43	584,35	43	556,69	43	603,61
97	1	0	1	2	0	44	581,32	44	537,5	44	604,69
98	1	0	1	2	1	41	573,75	41	531,83	41	586,39
99	1	0	1	2	2	43	566,78	43	562,67	43	608,84
100	1	0	2	0	0	43	573,07	43	555,23	43	607,98
101	1	0	2	0	1	44	571,66	44	551,54	44	605,08
102	1	0	2	0	2	41	567,36	41	542,07	41	613,09

	Values of parameters					Replica 1		Replica 2		Replica 3	
Exp.	iterations	set_efficiency	set_stop	prob2	prob4	Nº duties	Runtime (s)	Nº duties	Runtime (s)	Nº duties	Runtime (s)
103	1	0	2	1	0	43	574,97	43	532,43	43	603,88
104	1	0	2	1	1	42	564,42	42	551,4	42	617,95
105	1	0	2	1	2	42	573,89	42	532,13	42	618,62
106	1	0	2	2	0	42	560,97	42	535,52	42	626,76
107	1	0	2	2	1	44	580,55	44	561,99	44	614,6
108	1	0	2	2	2	42	562,49	42	565,63	42	608,3
109	1	2	0	0	0	43	576,63	43	552,69	43	619,46
110	1	2	0	0	1	44	575,75	44	571,52	44	627,33
111	1	2	0	0	2	45	590,38	45	555,61	45	633,84
112	1	2	0	1	0	44	574,95	44	565,13	44	618,08
113	1	2	0	1	1	41	581,48	41	539,06	41	616,9
114	1	2	0	1	2	42	566,53	42	570,76	42	616,18
115	1	2	0	2	0	45	576,3	45	557,85	45	602,68
116	1	2	0	2	1	44	599,86	44	562,79	44	608,95
117	1	2	0	2	2	42	587,64	42	576,17	42	629,51
118	1	2	1	0	0	46	609,17	46	550	46	623,04
119	1	2	1	0	1	45	598,6	45	535,87	45	612,31
120	1	2	1	0	2	44	596,84	44	584,64	44	596,2
121	1	2	1	1	0	47	584,69	47	556,49	47	611,18
122	1	2	1	1	1	44	578,01	44	553,32	44	597,21
123	1	2	1	1	2	44	577	44	562,31	44	604,74
124	1	2	1	2	0	44	581,01	44	562,99	44	607,34
125	1	2	1	2	1	45	594,14	45	562,81	45	614,43
126	1	2	1	2	2	45	592,51	45	559,59	45	618,47
127	1	2	2	0	0	41	602	41	565,96	41	622,56
128	1	2	2	0	1	45	587,11	45	557,89	45	600,85

	Values of parameters					Replica 1		Replica 2		Replica 3	
Exp.	iterations	set_efficiency	set_stop	prob2	prob4	Nº duties	Runtime (s)	Nº duties	Runtime (s)	Nº duties	Runtime (s)
129	1	2	2	0	2	42	584,97	42	542,62	42	625,92
130	1	2	2	1	0	44	598,18	44	549,91	44	613,46
131	1	2	2	1	1	46	595,81	46	563,03	46	606,64
132	1	2	2	1	2	43	594,21	43	558,85	43	615,5
133	1	2	2	2	0	42	586,51	42	544,5	42	587,79
134	1	2	2	2	1	42	595,89	42	556,43	42	613,77
135	1	2	2	2	2	44	611,57	44	552,58	44	625,12
136	1	1	0	0	0	42	605,7	42	550,23	42	611,32
137	1	1	0	0	1	40	589,85	40	566,78	40	604,83
138	1	1	0	0	2	42	591,89	42	578,03	42	627,9
139	1	1	0	1	0	43	601,41	43	562,1	43	610,46
140	1	1	0	1	1	41	594,05	41	569,35	41	605,66
141	1	1	0	1	2	41	600,35	41	544,5	41	604,81
142	1	1	0	2	0	42	594,19	42	547,99	42	623,16
143	1	1	0	2	1	43	584,64	43	543,44	43	595,62
144	1	1	0	2	2	41	588,07	41	558,4	41	623,88
145	1	1	1	0	0	39	574,51	39	543,94	39	631,32
146	1	1	1	0	1	43	574,78	43	547,67	43	619,99
147	1	1	1	0	2	45	579,2	45	554,58	45	614,56
148	1	1	1	1	0	42	572,03	42	557,05	42	605,31
149	1	1	1	1	1	42	581,48	42	564,77	42	603,71
150	1	1	1	1	2	45	595,27	45	563,38	45	613,56
151	1	1	1	2	0	44	568,9	44	570,21	44	624,24
152	1	1	1	2	1	42	579,34	42	558,37	42	624,55
153	1	1	1	2	2	42	593,41	42	545,04	42	600,1
154	1	1	2	0	0	45	616,43	45	541,24	45	615,48

	Values of parameters					Replica 1		Replica 2		Replica 3	
Exp.	iterations	set_efficiency	set_stop	prob2	prob4	Nº duties	Runtime (s)	Nº duties	Runtime (s)	Nº duties	Runtime (s)
155	1	1	2	0	1	42	589,99	42	555,62	42	613,82
156	1	1	2	0	2	42	600,54	42	589,42	42	608,23
157	1	1	2	1	0	43	610,9	43	542,23	43	609,16
158	1	1	2	1	1	43	585,02	43	544,84	43	622,07
159	1	1	2	1	2	42	591,78	42	567,91	42	618,44
160	1	1	2	2	0	42	591,37	42	543,7	42	607,55
161	1	1	2	2	1	43	574,42	43	563,01	43	617,6
162	1	1	2	2	2	43	625,89	43	553,04	43	608,68
163	2	0	0	0	0	43	1729,9	43	1644,8	43	1852,63
164	2	0	0	0	1	41	1709,69	41	1625,38	41	1796,5
165	2	0	0	0	2	42	1702,73	42	1652,53	42	1829,59
166	2	0	0	1	0	44	1697,75	44	1638,88	44	1807,65
167	2	0	0	1	1	44	1704,97	44	1654,21	44	1798,56
168	2	0	0	1	2	41	1750,11	41	1650,78	41	1811,93
169	2	0	0	2	0	43	1725,16	43	1612,46	43	1774,5
170	2	0	0	2	1	44	1696,53	44	1637,3	44	1816,22
171	2	0	0	2	2	42	1720,59	42	1658,66	42	1848,96
172	2	0	1	0	0	42	1683,94	42	1657,5	42	1819,26
173	2	0	1	0	1	43	1701,8	43	1657,54	43	1841,26
174	2	0	1	0	2	42	1680,88	42	1665,67	42	1830,14
175	2	0	1	1	0	42	1691,58	42	1680,16	42	1836,95
176	2	0	1	1	1	43	1696,19	43	1706,14	43	1841,14
177	2	0	1	1	2	42	1716	42	1689,14	42	1811,3
178	2	0	1	2	0	42	1709,17	42	1632,7	42	1796,69
179	2	0	1	2	1	45	1707,38	45	1603,11	45	1785,79
180	2	0	1	2	2	43	1704,97	43	1642,93	43	1839,48

	Values of parameters					Replica 1		Replica 2		Replica 3	
Exp.	iterations	set_efficiency	set_stop	prob2	prob4	Nº duties	Runtime (s)	Nº duties	Runtime (s)	Nº duties	Runtime (s)
181	2	0	2	0	0	45	1680,51	45	1599,03	45	1762,88
182	2	0	2	0	1	44	1715,13	44	1654,67	44	1801,27
183	2	0	2	0	2	46	1744,73	46	1648,39	46	1866,58
184	2	0	2	1	0	43	1718,39	43	1609,78	43	1806,15
185	2	0	2	1	1	42	1676,86	42	1629,05	42	1800,48
186	2	0	2	1	2	43	1738,98	43	1698,47	43	1791,24
187	2	0	2	2	0	43	1666,04	43	1684,01	43	1831,1
188	2	0	2	2	1	43	1692,66	43	1710,9	43	1795,83
189	2	0	2	2	2	44	1706,97	44	1660,64	44	1843,71
190	2	2	0	0	0	45	1754,83	45	1672,77	45	1840,08
191	2	2	0	0	1	45	1705,21	45	1655,3	45	1802,16
192	2	2	0	0	2	45	1743,15	45	1674,18	45	1864,28
193	2	2	0	1	0	46	1724,85	46	1634,76	46	1812,95
194	2	2	0	1	1	44	1710,01	44	1656,77	44	1788,82
195	2	2	0	1	2	41	1747,2	41	1686,58	41	1821,07
196	2	2	0	2	0	43	1696,69	43	1651,03	43	1815,2
197	2	2	0	2	1	43	1700,66	43	1662,15	43	1783,23
198	2	2	0	2	2	42	1706,89	42	1669,47	42	1839,52
199	2	2	1	0	0	44	1701,17	44	1654,12	44	1824,89
200	2	2	1	0	1	42	1715,06	42	1640,11	42	1829,53
201	2	2	1	0	2	46	1715,88	46	1672,15	46	1831,95
202	2	2	1	1	0	43	1717,84	43	1652,57	43	1789,41
203	2	2	1	1	1	45	1698,99	45	1657,82	45	1770,93
204	2	2	1	1	2	44	1699,96	44	1631,81	44	1804,9
205	2	2	1	2	0	45	1716,66	45	1684,58	45	1789,64
206	2	2	1	2	1	42	1701,54	42	1657,91	42	1838,98

	Values of parameters					Replica 1		Replica 2		Replica 3	
Exp.	iterations	set_efficiency	set_stop	prob2	prob4	Nº duties	Runtime (s)	Nº duties	Runtime (s)	Nº duties	Runtime (s)
207	2	2	1	2	2	44	1748,41	41	1669,81	41	1846,15
208	2	2	2	0	0	43	1705,13	43	1674,09	43	1795,11
209	2	2	2	0	1	45	1689,29	45	1676	45	1818,17
210	2	2	2	0	2	45	1713,74	45	1685,2	45	1834,24
211	2	2	2	1	0	43	1714,62	43	1702,43	43	1814,76
212	2	2	2	1	1	43	1697,44	43	1662,96	43	1826,03
213	2	2	2	1	2	43	1728,93	43	1676,22	43	1811,84
214	2	2	2	2	0	46	1726,61	46	1684,71	46	1788,9
215	2	2	2	2	1	44	1707,83	44	1696,91	44	1821,21
216	2	2	2	2	2	45	1693,42	45	1673,31	45	1803,25
217	2	1	0	0	0	41	1713,22	41	1687,59	41	1780,51
218	2	1	0	0	1	42	1741,14	42	1705,79	42	1799,17
219	2	1	0	0	2	42	1729,21	42	1721,38	42	1849,43
220	2	1	0	1	0	42	1704,93	42	1681,81	42	1806,61
221	2	1	0	1	1	41	1718,41	41	1661,99	41	1805,01
222	2	1	0	1	2	43	1722,27	43	1666,68	43	1791,76
223	2	1	0	2	0	43	1680,76	43	1638,85	43	1790,85
224	2	1	0	2	1	42	1675,12	42	1655,36	42	1813,39
225	2	1	0	2	2	44	1722,48	44	1670,05	44	1834,4
226	2	1	1	0	0	43	1720,82	43	1659,4	43	1792,47
227	2	1	1	0	1	41	1689,65	41	1647,1	41	1805,75
228	2	1	1	0	2	42	1709,6	42	1649,7	42	1812,21
229	2	1	1	1	0	42	1695,98	42	1701,56	42	1834,37
230	2	1	1	1	1	41	1729,38	41	1618,94	41	1806,97
231	2	1	1	1	2	43	1741,35	43	1637,64	43	1845,3
232	2	1	1	2	0	43	1727,23	43	1632,42	43	1817,92

	Values of parameters					Replica 1		Replica 2		Replica 3	
Exp.	iterations	set_efficiency	set_stop	prob2	prob4	Nº duties	Runtime (s)	Nº duties	Runtime (s)	Nº duties	Runtime (s)
233	2	1	1	2	1	42	1714,46	42	1654,6	42	1795,29
234	2	1	1	2	2	39	1746,72	39	1658,8	39	1811,6
235	2	1	2	0	0	44	1741,43	44	1650,24	44	1768,92
236	2	1	2	0	1	44	1762,04	44	1621,81	44	1794,35
237	2	1	2	0	2	41	1775,68	41	1671,6	41	1814,94
238	2	1	2	1	0	43	1710,29	43	1633,54	43	1790,54
239	2	1	2	1	1	44	1714,85	44	1676,65	44	1810,05
240	2	1	2	1	2	45	1746,58	45	1694,72	45	1817,11
241	2	1	2	2	0	44	1750,01	44	1660,8	44	1804,45
242	2	1	2	2	1	45	1762,8	45	1692,75	45	1835,64
243	2	1	2	2	2	45	1715,46	45	1609,13	45	1833,79

13.4.2.2 Effects of the parameters and their interactions in relation to the number of duties (Group 2)

Table 47. General Linear Model: duties versus factors (Group 2).

Factor	Type	Levels	Values
iterations	fixed	3	0. 1. 2 / 10; 30; 90
set_efficiency	fixed	3	0. 1. 2 / 50; 60; 70
set_stop	fixed	3	0. 1. 2 / 5; 10; 15
prob2	fixed	3	0. 1. 2 / 0.1; 0.25; 0.5
prob4	fixed	3	0. 1. 2 / 0.1; 0.25; 0.5

Table 48. Analysis of Variance for duties, using Adjusted SS for Tests (Group 2).

Source	DF	Seq SS	Adj SS	Adj MS	F
Iterations	2	12,848	12,848	6,424	4,99
set_efficiency	2	268,914	268,914	134,457	104,50
set_stop	2	37,045	37,045	18,523	14,40
prob2	2	3,185	3,185	1,593	1,24
prob4	2	9,737	9,737	4,868	3,78
iterations*set_efficiency	4	14,239	14,239	3,560	2,77
iterations*set_stop	4	5,416	5,416	1,354	1,05
iterations*prob2	4	8,115	8,115	2,029	1,58
iterations*prob4	4	10,700	10,700	2,675	2,08
set_efficiency*set_stop	4	9,868	9,868	2,467	1,92
set_efficiency*prob2	4	24,198	24,198	6,049	4,70
set_efficiency*prob4	4	16,560	16,560	4,140	3,22
set_stop*prob2	4	14,979	14,979	3,745	2,91
set_stop*prob4	4	3,342	3,342	0,835	0,65
prob2*prob4	4	15,449	15,449	3,862	3,00

Table 49. P values for duties (Group 2).

Source	P
Iterations	0,007
set_efficiency	0,000
set_stop	0,000
prob2	0,291
prob4	0,023
iterations*set_efficiency	0,027
iterations*set_stop	0,380
iterations*prob2	0,179
iterations*prob4	0,082
set_efficiency*set_stop	0,106

Source	P
set_efficiency*prob2	0,001
set_efficiency*prob4	0,013
set_stop*prob2	0,021
set_stop*prob4	0,628
prob2*prob4	0,018

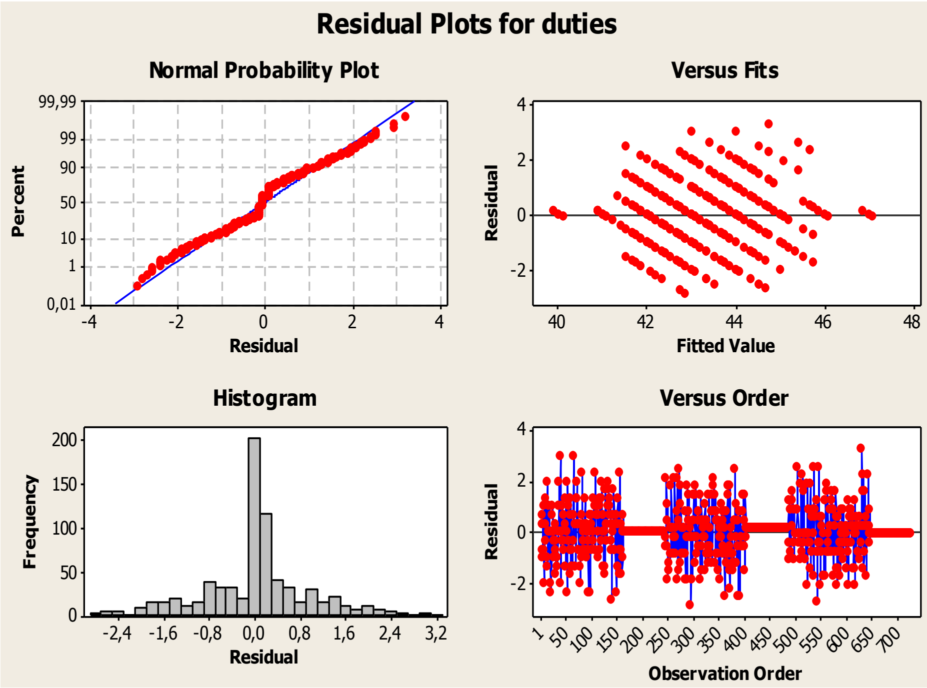


Figure 34. Residual Plots for Duties (Group 2).

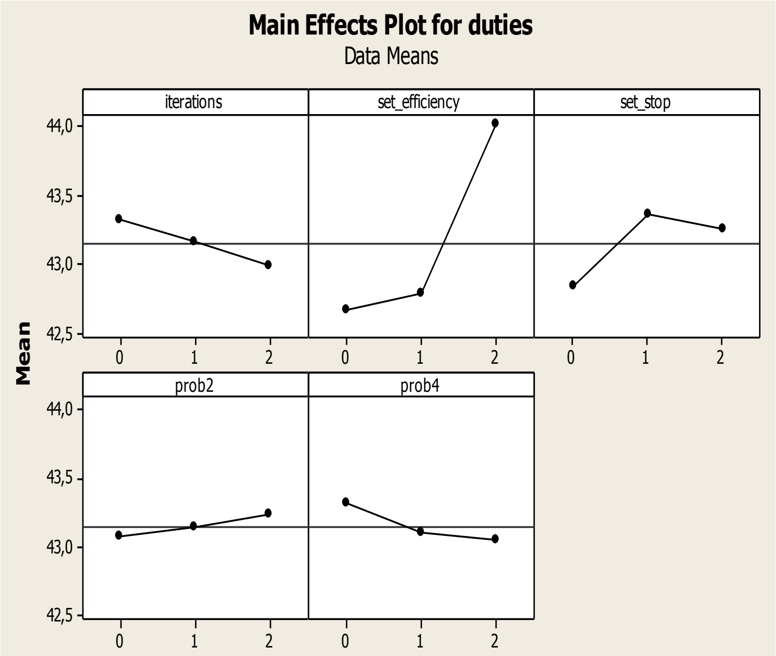


Figure 35. Main Effects Plot for Duties. Data Means (Group 2)

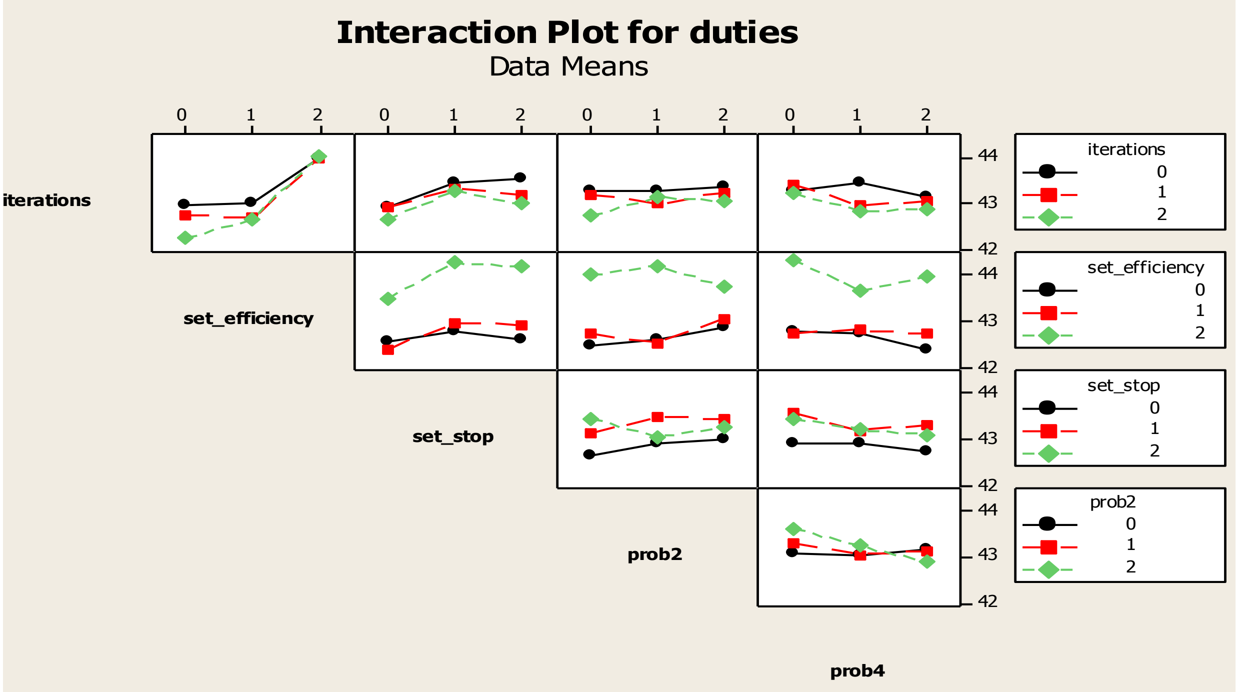


Figure 36. Interaction Plot for Duties. Data Means (Group 2).

13.4.2.3 Effects of the parameters and their interactions in relation to the runtime (Group 2).

Table 50. General Linear Model: comp.time versus factors (Group 2)

Factor	Type	Levels	Values
iterations	fixed	3	0. 1. 2 / 10; 30; 90
set_efficiency	fixed	3	0. 1. 2 / 50; 60; 70
set_stop	fixed	3	0. 1. 2 / 5; 10; 15
prob2	fixed	3	0. 1. 2 / 0.1; 0.25; 0.5
prob4	fixed	3	0. 1. 2 / 0.1; 0.25; 0.5

Table 51. Analysis of Variance for comp.time, using Adjusted SS for Tests (Group 2).

Source	D F	Seq SS	Adj SS	Adj MS	F
Iterations	2	82705342	82705342	41352671	400643,36
set_efficiency	2	5917	5917	2958	28,66
set_stop	2	7118	7118	3559	34,48
prob2	2	1118	1118	559	5,42
prob4	2	2950	2950	1475	14,29
iterations*set_efficiency	4	14372	14372	3593	34,81
iterations*set_stop	4	3663	3663	916	8,87
iterations*prob2	4	1303	1303	326	3,15
iterations*prob4	4	1361	1361	340	3,30
set_efficiency*set_stop	4	4177	4177	1044	10,12

Source	D F	Seq SS	Adj SS	Adj MS	F
set_efficiency*prob2	4	681	681	170	1,65
set_efficiency*prob4	4	155	155	39	0,38
set_stop*prob2	4	2272	2272	568	5,50
set_stop*prob4	4	322	322	80	0,78
prob2*prob4	4	670	670	168	1,62

Table 52. P values for comp.time (Group 2).

Source	P
Blocks	0,000
Iterations	0,000
set_efficiency	0,000
set_stop	0,000
prob2	0,005
prob4	0,000
iterations*set_efficiency	0,000
iterations*set_stop	0,000
iterations*prob2	0,014
iterations*prob4	0,011
set_efficiency*set_stop	0,000
set_efficiency*prob2	0,161
set_efficiency*prob4	0,826
set_stop*prob2	0,000
set_stop*prob4	0,539
prob2*prob4	0,167

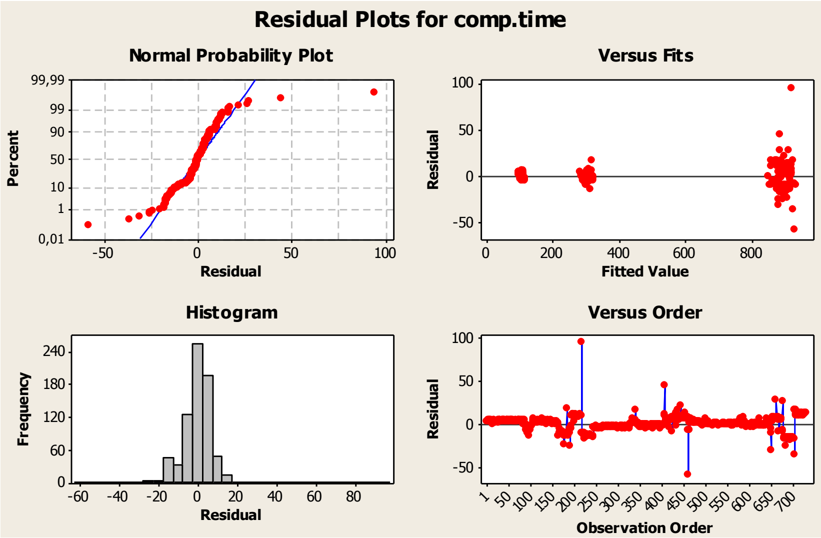


Figure 37. Residual Plots for Computational Time (Group 2).

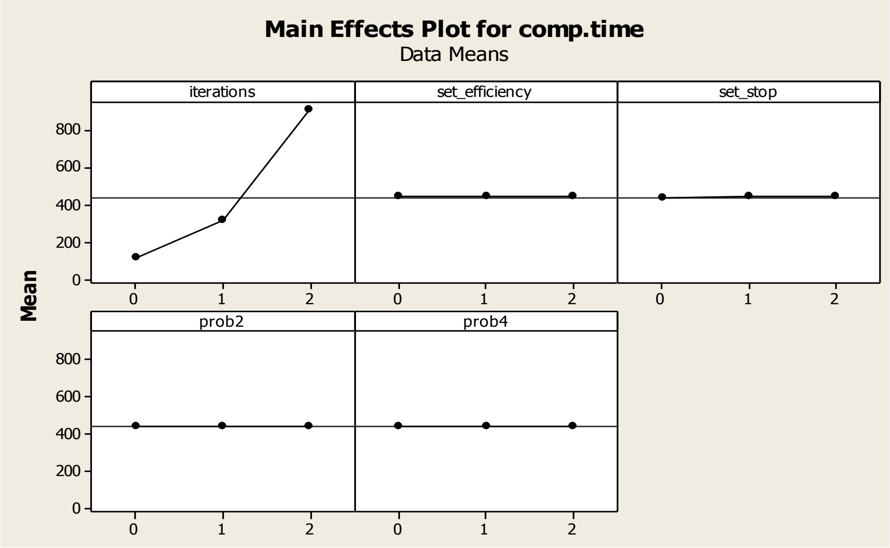


Figure 38. Main Effects Plot for Computational Time. Data Means (Group 2)

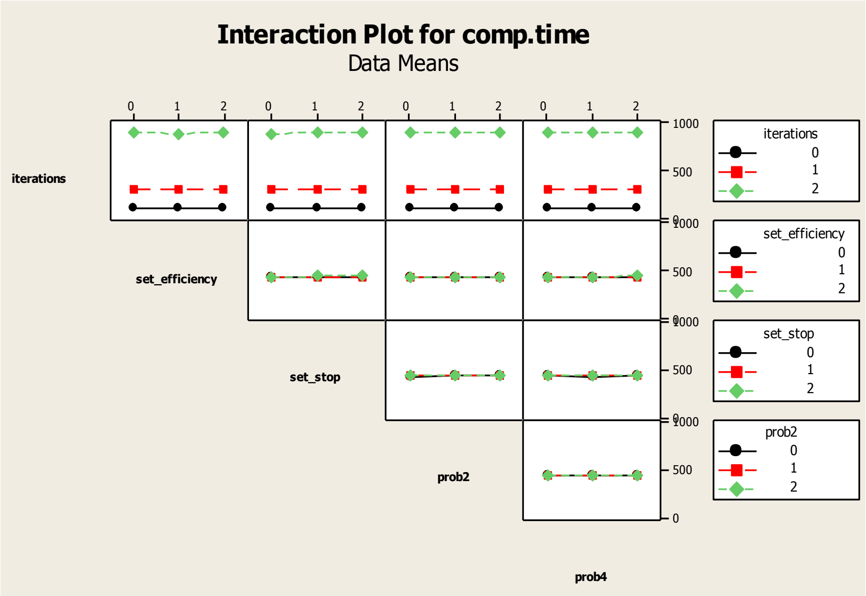


Figure 39. Interaction Plot for Computational Time. Data Means (Group 2)

13.4.3 Group 3

13.4.3.1 Results of experiments of Group 3

Table 53. Results of experiments of Group 3

	Values of parameters			Replica 1		Replica 2		Replica 3	
Exp.	iterations	set_efficiency	set_stop	Nº duties	Runtime (s)	Nº duties	Runtime (s)	Nº duties	Runtime (s)
1	0	0	0	50	551,35	50	427,94	50	610,15
2	0	0	1	56	530,17	56	417,09	56	620,81
3	0	0	2	56	511,49	56	420,63	56	580,5
4	0	2	0	51	517,05	51	416,92	51	616,31
5	0	2	1	54	525,98	54	422,02	54	602,05
6	0	2	2	55	507,95	55	420,09	55	605,11
7	0	1	0	56	527,71	56	423,74	56	618
8	0	1	1	55	518,92	55	420,7	55	602,83
9	0	1	2	55	526,01	55	423,62	55	611,07
10	1	0	0	58	1490,66	58	1229,32	58	1752,13
11	1	0	1	55	1491,13	55	1232,96	55	1750,14
12	1	0	2	51	1507,31	51	1228,22	51	1730,12
13	1	2	0	55	1529,72	55	1460,27	55	1750,42
14	1	2	1	52	1540,57	52	1813,46	52	1780,27
15	1	2	2	54	1553,21	54	1862,63	54	1766,79
16	1	1	0	50	1508,68	50	1817,73	50	1769,54
17	1	1	1	54	1584,59	54	1876,74	54	1755,7
18	1	1	2	54	1576,96	54	1857,5	54	1746,89
19	2	0	0	55	4698,48	55	5547,21	55	5196,08
20	2	0	1	53	4613,08	53	5554,06	53	5244,11
21	2	0	2	55	4364,69	55	6939,65	55	5258,11
22	2	2	0	54	4399,71	52	5653,66	52	5324,81
23	2	2	1	49	4371,68	49	5625,14	49	5325,49
24	2	2	2	53	4491,74	53	5776,27	53	5634,4

	Values of parameters			Replica 1		Replica 2		Replica 3	
Exp.	iterations	set_efficiency	set_stop	Nº duties	Runtime (s)	Nº duties	Runtime (s)	Nº duties	Runtime (s)
25	2	1	0	51	4344,7	51	5822,24	51	5855,36
26	2	1	1	51	4508,73	51	5725,18	51	6027,45
27	2	1	2	54	4467,96	54	5517,53	54	6091,02

13.4.3.2 Effects of the parameters and their interactions in relation to the number of duties (Group 3)

Table 54. General Linear Model: duties versus factors (Group 3).

Factor	Type	Levels	Values
iterations	fixed	3	0. 1. 2 / 10; 30; 90
set_efficiency	fixed	3	0. 1. 2 / 50; 60; 70
set_stop	fixed	3	0. 1. 2 / 5; 10; 15

Table 55. Analysis of Variance for duties, using Adjusted SS for Tests (Group 3).

Source	DF	Seq SS	Adj SS	Adj MS	F
Iterations	2	10,889	10,889	5,444	2,49
set_efficiency	2	8,296	8,296	4,148	1,90
set_stop	2	70,222	70,222	35,111	16,05
iterations*set_efficiency	4	35,704	35,704	8,926	4,08
iterations*set_stop	4	7,778	7,778	1,944	0,89
set_efficiency*set_stop	4	10,370	10,370	2,593	1,19

Table 56. P values for duties (Group 3).

Source	P
Iterations	0,091
set_efficiency	0,159
set_stop	0,000
iterations*set_efficiency	0,005
iterations*set_stop	0,476
set_efficiency*set_stop	0,326

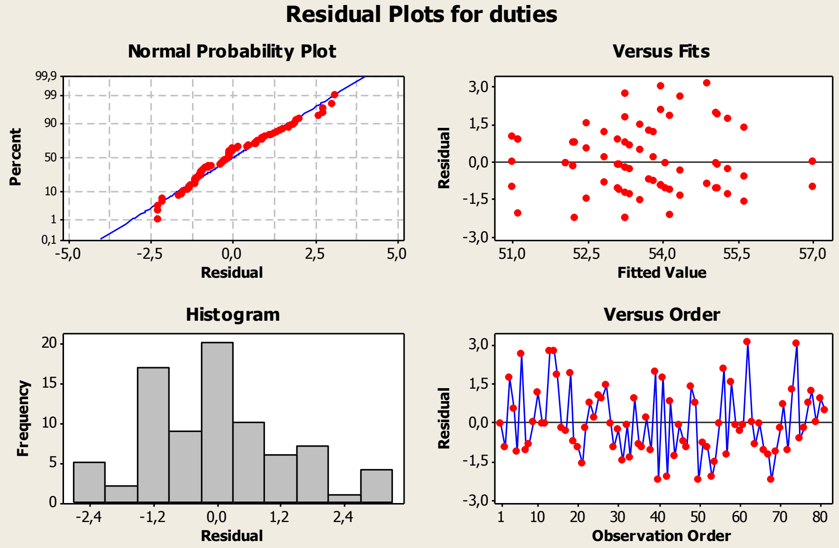


Figure 40. Residual Plots for Duties (Group 3).

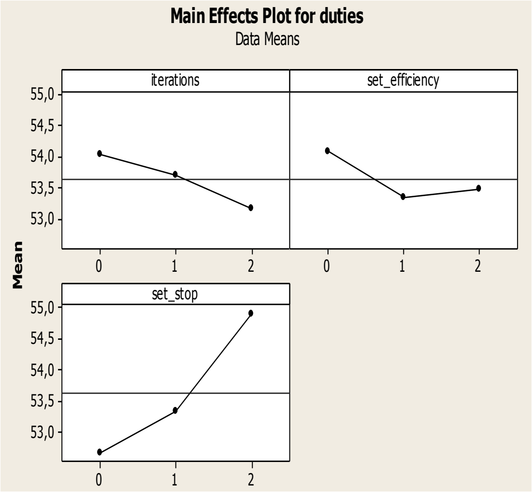


Figure 41. Main Effects Plot for Duties. Data Means (Group 3)

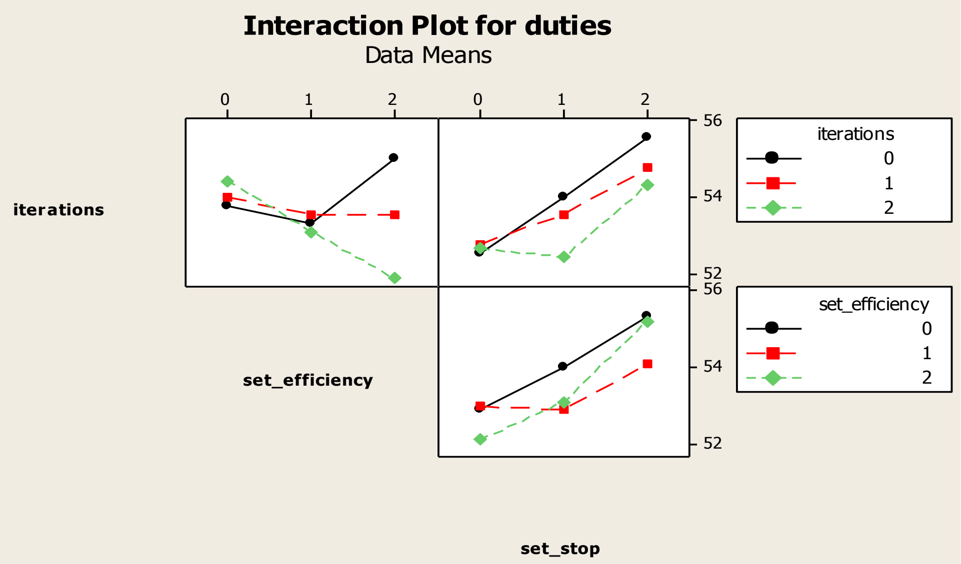


Figure 42. Interaction Plot for Duties. Data Means (Group 3).

13.4.3.3 Effects of the parameters and their interactions in relation to the runtime (Group 3).

Table 57. General Linear Model: comp.time versus factors (Group 3).

Factor	Type	Levels	Values
iterations	fixed	3	0. 1. 2 / 10; 30; 90
set_efficiency	fixed	3	0. 1. 2 / 50; 60; 70
set_stop	fixed	3	0. 1. 2 / 5; 10; 15

Table 58. Analysis of Variance for comp.time, using Adjusted SS for Tests (Group 3).

Source	DF	Seq SS	Adj SS	Adj MS	F
iterations	2	100395895	100395895	50197947	117887,32
set_efficiency	2	22398	22398	11199	26,30
set_stop	2	1051	1051	525	1,23

Source	DF	Seq SS	Adj SS	Adj MS	F
iterations*set_efficiency	4	18909	18909	4727	11,10
iterations*set_stop	4	888	888	222	0,52
set_efficiency*set_stop	4	4855	4855	1214	2,85

Table 59. P values for comp.time (Group 3)

Source	P
iterations	0,000
set_efficiency	0,000
set_stop	0,298
iterations*set_efficiency	0,000
iterations*set_stop	0,720
set_efficiency*set_stop	0,031

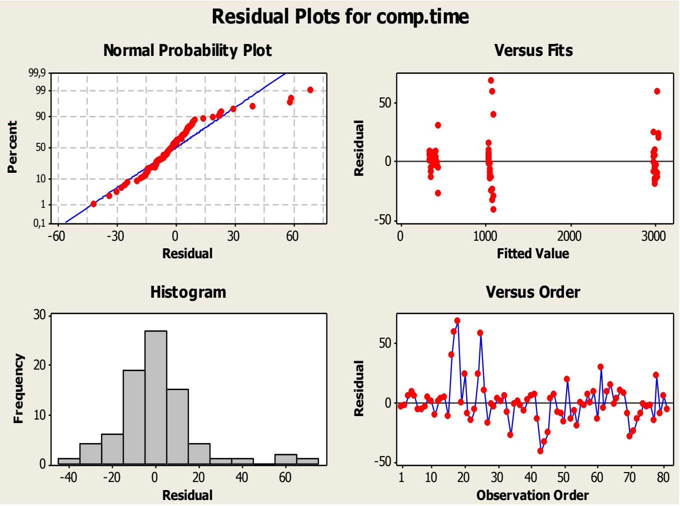


Figure 43. Residual Plots for Computational Time (Group 3).

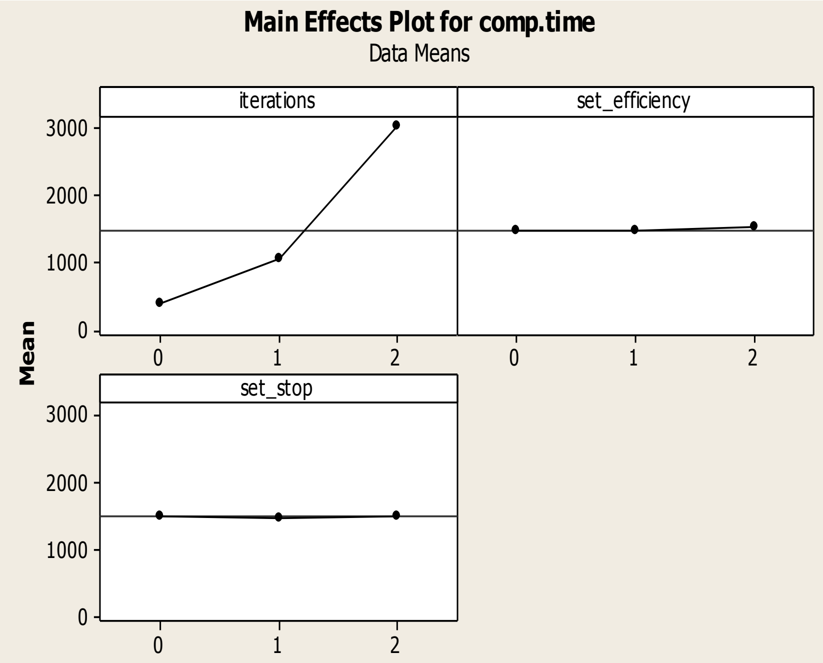


Figure 44. Main Effects Plot for Computational Time. Data Means (Group 3)

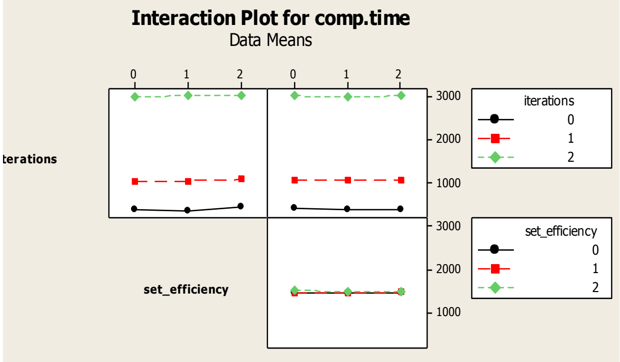


Figure 45. Interaction Plot for Computational Time. Data Means (Group 3)