

**UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN
FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA**



TESIS

**SOLVING THE SHIFT SCHEDULING AND THE
IMAGE SEGMENTATION PROBLEMS USING SET
COVERING FORMULATIONS**

PRESENTADA POR

NORBERTO ALEJANDRO HERNÁNDEZ LEANDRO

**COMO REQUISITO PARCIAL PARA OBTENER EL GRADO DE DOCTOR
EN INGENIERÍA CON ESPECIALIDAD EN INGENIERÍA DE SISTEMAS**

OCTUBRE, 2018

UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN
FACULTAD DE INGENIERÍA MECÁNICA Y ELÉCTRICA
SUBDIRECCIÓN DE ESTUDIOS DE POSGRADO



TESIS

**SOLVING THE SHIFT SCHEDULING AND THE
IMAGE SEGMENTATION PROBLEMS USING SET
COVERING FORMULATIONS**

PRESENTADA POR

NORBERTO ALEJANDRO HERNÁNDEZ LEANDRO

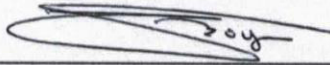
**COMO REQUISITO PARCIAL PARA OBTENER EL GRADO DE DOCTOR
EN INGENIERÍA CON ESPECIALIDAD EN INGENIERÍA DE SISTEMAS**

OCTUBRE, 2018

Universidad Autónoma de Nuevo León
Facultad de Ingeniería Mecánica y Eléctrica
Subdirección de Estudios de Posgrado

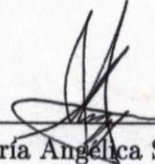
Los miembros del Comité de Tesis recomendamos que la Tesis "Solving the Shift Scheduling and the Image Segmentation problems using Set-Covering Formulations", realizada por el alumno Norberto Alejandro Hernández Leandro, con número de matrícula 1654715, sea aceptada para su defensa como opción al grado de Doctor en Ingeniería con Especialidad en Ingeniería de Sistemas.

El Comité de Tesis



Dr. Vincent André Lionel Boyer

Director




Dra. María Angélica Salazar Aguilar

Co-Director



Prof. Louis-Martin Rousseau

Revisor



Dr. Roger Z. Ríos Mercado

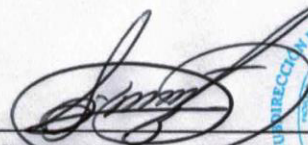
Revisor



Dr. Omar Jorge Ibarra Rojas

Revisor

Vo. Bo.



Dr. Simón Martínez Martínez

Subdirector de Estudios de Posgrado



San Nicolás de los Garza, Nuevo León, Octubre 2018

To my family

CONTENTS

Acknowledgements	xii
Summary	xiii
Resumen	xvi
1 Introduction	1
1.1 The Multi-Activity Shift Scheduling Problem	1
1.2 Image Segmentation Problem	4
1.3 Motivation	6
1.4 Objective	6
1.5 Organization	7
2 Literature Review	8
2.1 Set-Covering Problem	8
2.2 The Multi-Activity Shift Scheduling Problem	10
2.3 Image Segmentation	11

3	Theoretical Framework	14
3.1	The Multi-Activity Shift Scheduling Problem	14
3.1.1	Context-Free Grammar	14
3.2	The Image Segmentation Problem	18
3.2.1	Model	19
3.2.2	The Histogram	20
4	The Multi-Activity Shift Scheduling Problem	23
4.1	Solution Method	25
4.1.1	Lagrangian Relaxation	25
4.1.2	Matheuristic	31
4.2	Experimental results	34
4.2.1	Problem Instances	35
4.2.2	Matheuristic	36
4.2.3	Column Generation Heuristic	40
4.3	Conclusions	41
5	The Image Segmentation Problem	43
5.1	Problem Description and Formulation	43
5.2	AUGMECON	46
5.3	Experimental Results	48
5.3.1	High Definition Images	49

5.3.2 Weizmann Images	60
5.4 Conclusions	61
6 Conclusions	66
7 Future Work	69

LIST OF FIGURES

1.1	Requirements of activity a	2
1.2	Example of assignment of activity a	3
1.3	New requirements of activity a	3
1.4	Example 1	5
3.1	Parse trees of the words aab and abb	16
3.2	DAG Γ of the words of three letters	17
3.3	DAG Γ representation for the feasible shifts	19
3.4	Example of a histogram of an image	21
3.5	Example of a distribution of the histogram of Figure 3.4	22
4.1	Cost propagation for DAG Γ	30
4.2	Flowchart of the Matheuristic	33
5.1	Experimental results for the first example of the high definition images	50
5.2	Experimental results for the second example of the high definition images	52

5.3	Average F-measure	56
5.4	Average time	57
5.5	Distributions of measures	59
5.6	Distributions of processing times	61
5.7	Examples of the experimental results of the Weizmann dataset	62
5.8	Cumulative distribution of the processing time	64

LIST OF TABLES

3.1	Derivation of the words aab (a) y abb (b)	16
3.2	Derivations of the shifts $a_1a_1a_1a_1$ (a) and $a_1a_1a_2a_2$ (b).	18
4.1	Classification of the Lequy instances.	36
4.2	Experimental results for the Demassey instances.	37
4.3	Average solution obtained with the matheuristic for the Lequy instances.	38
4.4	Solutions obtained by the matheuristic for the Lequy instances.	39
4.5	Comparison between MH and CGH for the Demassey instances.	40
4.6	Comparison between MH and CGH for the Lequy instances.	41
5.1	Alternative Models	46
5.2	Summary of the Pareto frontiers with Sigma.freq	53
5.3	Summary of the Pareto frontiers with Sigma.logfreq	53
5.4	Summary of the Pareto frontiers with Media.freq	54
5.5	Summary of the Pareto frontiers with Media.logfreq	54
5.6	Average F-measure	55

5.7	Average time	56
5.8	Homogeneity of the variances test	58
5.9	Normality test	58
5.10	ANOVA	58
5.11	Kruskal-Wallis test	59
5.12	Test of differences	60
5.13	Table of scores	63

ACKNOWLEDGEMENTS

I would like to express my special thanks to my parents Roberto Hernández and Rosa Elba Leandro, for their infinite support and their endless love. I thank my family for being there always I have needed them. A special thanks goes to my girlfriend Mónica Elizondo for her company and for giving me the extra push in the difficult times.

Furthermore, I would like to express my deepest appreciation to my thesis committee, whose contribution and suggestions helped me to improve this work; To Dr. Vincent Boyer and Dra. Angélica Salazar for their guidance, the trust and the advices, they have given me since I started my master degree; To Prof. Louis-Martin Rousseau for his advise and assistance throughout my graduate education.

Likewise, I would like to acknowledge to my friends for the encouragement and all the moments we shared; To the PISIS teachers for all the knowledge, the insight and the wisdom they have provided me along my graduate studies.

Last but not least, thanks to CONACyT for the doctoral fellowship with grant number: 412762; to the Universidad Autónoma de Nuevo León (UANL), and the Facultad de Ingeniería Mecánica y Eléctrica (FIME).

SUMMARY

Norberto Alejandro Hernández Leandro.

Ph.D. candidate in Engineering with a Specialization in Systems Engineering.

Universidad Autónoma de Nuevo León.

Facultad de Ingeniería Mecánica y Eléctrica.

Title of the study: SOLVING THE SHIFT SCHEDULING AND THE IMAGE SEGMENTATION PROBLEMS USING SET COVERING FORMULATIONS.

Number of pages: 77.

OBJECTIVES AND METHODS OF STUDY: The main objective of this research is the study of two applications of the set-covering problem: the multi-activity shift scheduling problem and the image segmentation problem.

The multi-activity shift scheduling problem consists of determining the sequence of activities that each employee has to perform, in order to minimize the over and under covering of the demand of each activity at each period of the planning horizon. In this work we propose to use a set-covering formulation of this problem. Furthermore, we present a matheuristic which uses a Lagrangian relaxation heuristic in order to generate promising shifts, and a restricted set-covering formulation to find the combination of the shifts generated by the Lagrangian heuristic that minimizes the under and over covering cost.

The image segmentation is an image processing problem which consists of dividing the image into sections of similar characteristics, also named segments. The objective of the segmentation is to provide a new representation of an image that eases its analysis. In this work, we propose a bi-objective set-covering formulation for the image segmentation problem, that uses the information of the image histogram. The objectives of this model are to minimize the number of sets of the partition of the histogram, and a heterogeneity measure of the elements of each set with the aim of building the partition with a proper number of groups with similar elements. In order to solve this problem, we propose to use an AUGMECON algorithm, which is a variant of the ϵ -constraint, to find the Pareto frontier of the bi-objective model, with the objective to find efficient partitions for the histogram, and consequently to generate a segmentation from each partition. Finally, we suggest to consider only the solution provided at the first iteration of the AUGMECON, in order to select automatically the number of sets in the partition.

CONTRIBUTIONS: In this thesis we propose a matheuristic method for the multi-activity shift scheduling problem, and a new bi-objective set-covering formulation for the image segmentation problem.

The experiments of the matheuristic show that this algorithm is able to provide better solutions within a relatively low processing times than the best solutions found in the literature; besides, we implement a column generation heuristic to compare with the proposed matheuristic; the experimental results reaffirm the performance of the proposed algorithm.

The result obtained from the bi-objective model for the image segmentation reveals that the solution of the model provides good quality segmentation with a relatively low processing time. The advantage of this work is that the size of the model does not depend on the image size, which means that solution of the model can produce a segmentation of large size images without affecting the processing times considerably. Furthermore, the solution at the first iteration of the solution

method provides the number of segments automatically.

Signature of the faculty advisers: _____

Dr. Vincent André Lionel Boyer

Dra. María Angélica Salazar Aguilar

RESUMEN

Norberto Alejandro Hernández Leandro.

Candidato para obtener el grado de Doctor en Ingeniería con especialidad en Ingeniería de Sistemas.

Universidad Autónoma de Nuevo León.

Facultad de Ingeniería Mecánica y Eléctrica.

Título: SOLVING THE SHIFT SCHEDULING AND THE IMAGE SEGMENTATION PROBLEMS USING SET COVERING FORMULATIONS.

Número de páginas: 77.

OBJETIVO Y METODOLOGÍA DE ESTUDIO: El objetivo de esta investigación es estudiar dos aplicaciones del problema de cobertura de conjuntos: el problema de asignación de horarios a empleados con múltiples actividades, y el problema de segmentación de imágenes.

El problema de asignación de horarios a empleados consiste en asignar la secuencia de actividades que cada empleado debe de realizar, con el objetivo de cubrir la demanda de cada actividad a cada periodo de la mejor manera posible. Para abordar este problema, se propone hacer uso de un modelo basado en el problema de cobertura de conjuntos. Para solucionar este problema, se plantea resolver la relajación Lagrangiana del modelo, con el objetivo de generar turnos promisorios.

Finalmente, se propone resolver el modelo restringido para encontrar la combinación de turnos que minimiza los costos de incumplimiento y sobrecumplimiento de cada actividad a cada periodo del horizonte de planeación.

La segmentación de imágenes es un problema de procesamiento de imágenes. Dicho problema consiste en dividir la imagen en secciones con características similares, las cuales son llamadas segmentos. El objetivo es encontrar una nueva representación de la imagen que ayude a facilitar su análisis. En este contexto, se formula un modelo bi-objetivo basado en el problema de cobertura de conjuntos que hace uso de la información del histograma de la imagen. Los objetivos de este modelo se basan en la idea de encontrar la partición del histograma de la imagen con la menor cantidad de divisiones posible, minimizando una medida de heterogeneidad entre los elementos de cada división. Para resolver dicho problema, se plantea utilizar el algoritmo AUGMECON, que es una variante del algoritmo ϵ -restricción para resolver el problema bi-objetivo y obtener el frente pareto para este problema. Posteriormente, se sugiere tomar la solución obtenida en la primera iteración del AUGMECON para obtener una solución de manera automática para este problema; i.e. el algoritmo no necesita de la guía del usuario para la obtención una segmentación para la imagen.

CONTRIBUCIONES: En este trabajo se propone una matheurística para el problema de asignación de horarios a empleados; además, se formula un nuevo modelo bi-objetivo basado en cobertura de conjuntos para la segmentación de imágenes, y se utiliza el algoritmo AUGMECON para solucionar el modelo.

La experimentación del algoritmo propuesto para solucionar el problema de asignación de horarios a empleados, muestra que es capaz de obtener soluciones de mejor calidad en un tiempo de procesamiento menor, en comparación con las mejores soluciones encontradas en la literatura. Además, se implementó una heurística basada en la generación de columnas para fortalecer los resultados experimentales y confirmar la eficiencia de la matheurística.

La experimentación para solucionar el modelo bi-objetivo propuesto para la

segmentación de imágenes, revela que las soluciones del modelo son de buena calidad y se obtienen en un tiempo de cómputo relativamente bajo. La ventaja de este método radica en que el tamaño del modelo no depende del tamaño de la imagen; lo que significa que a partir del modelo se puede segmentar imágenes tan grandes como se desee sin afectar de manera considerable los tiempos de procesamiento.

Firma de directores: _____

Dr. Vincent André Lionel Boyer

Dra. María Angélica Salazar Aguilar

CHAPTER 1

INTRODUCTION

The Set-Covering Problem (SCP) is a classic discrete optimization problem which consist of covering all the elements of a given set U by the elements of the set $S \subset \mathcal{P}(U)$ minimizing the total cost $c(S)$ of using each element of S , where $\mathcal{P}(U)$ is the power set of U . This problem has been studied since the mid 1960's and, consequently, several applications and solution methods have been proposed. Karp (1972), proved that SCP is a NP-Hard problem; however, there are many methods to solve the SCP efficiently (see: Christofides and Korman (1975) and Caprara et al. (2000)).

In this thesis, we study SCP models applied to two particular problems: the Multi-Activity Shift Scheduling problem and the Image Segmentation problem. The following sections present both problems.

1.1 THE MULTI-ACTIVITY SHIFT SCHEDULING PROBLEM

The Multi-Activity Shift Scheduling Problem (MASSP), also known as personalized multi-activity shift scheduling problem (Côté et al., 2013), consists of assigning shifts to a set of employees. These shifts are represented by a sequence of activities and

can include resting or meal periods. Furthermore, each sequence is restricted by the company rules and the activities that each employee is able to perform.

The activities represent daily operations of the company; besides, each activity has a demand at each period of the planning horizon, given by the number of employees needed to provide an efficient service to the clients.

In this work, we consider a discretized planning horizon; on one hand, it is senseless that an employee performs an activity just by a small period of time and then perform another, because there may be no chance to complete any task in this period; on the other hand, the transitions between activities have to deal with set up times, for this reason is not relevant to consider small periods to perform these transitions.

For example, let us suppose that a is an activity and its demand at each period of the planning horizon is represented by Figure 1.1.

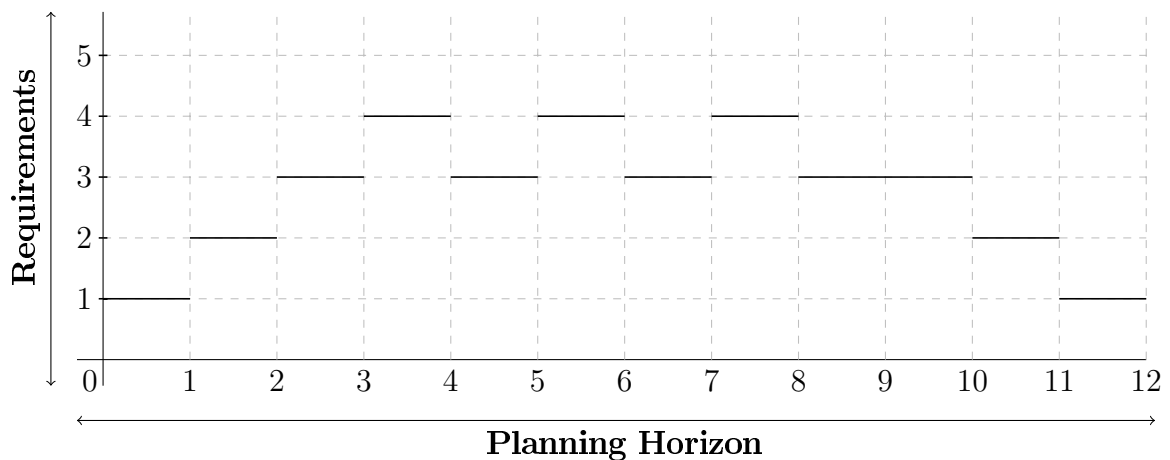


Figure 1.1: Requirements of activity a

If the assignment of the activity a is carried out in the way presented in Figure 1.2, then the requirements of the activity a at each period change. Particularly, the new requirements after the fixing of these shifts are presented in Figure 1.3, where in the periods zero, one, eight, ten and eleven the demand of activity a is met. For the remaining periods, the demand of activity a is partially met; i.e., it is needed to

assign the activity a to more employees, in order to satisfy the requirements of the company.

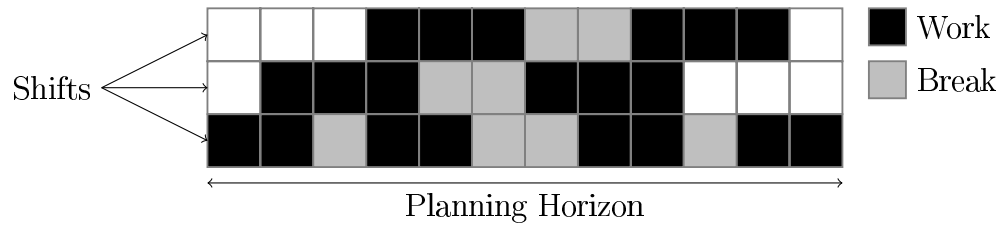


Figure 1.2: Example of assignment of activity a .

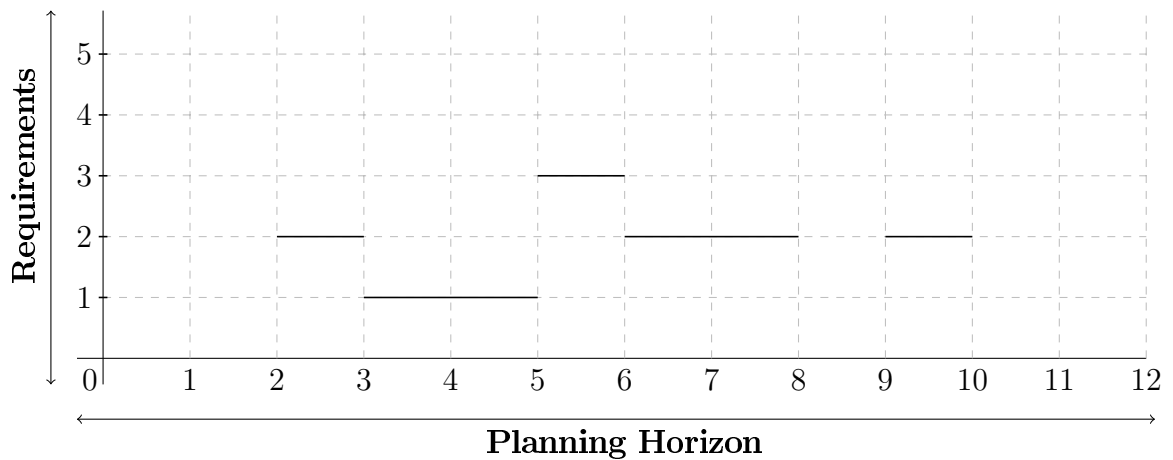


Figure 1.3: New requirements of activity a

The objective of the MASSP is to assign the sequence of activities that each employee has to perform, in order to minimize the total cost of undercovering or overcovering activities at each period of time. The undercovering of a demand happens when there are not enough employees performing a given activity at a certain period, whilst the overcovering occurs when there are an excess of employees performing an activity.

In this thesis, we consider two variants of the MASSP. Firstly, the working periods of the shifts are fixed a priori and, secondly, the working periods of the shifts are not fixed. Furthermore, the next characteristics are also considered:

- A discretized planning horizon.

- Each employee can perform a subset of activities.
- Each activity must be performed between a maximum and a minimum span.
- Each activity has a demand at each period of the planning horizon.
- The under and over covering of each activity at each period have a fixed cost.
- The undercovering is more expensive than the overcovering.
- The transition between activities has a fixed cost.

In this this context, if all the feasible shifts could be generated, then the MASSP turns into the problem of covering the demand for each activity at each period of the planning horizon with a feasible shift per employee. Particularly, each set of the SCP is represented by the sequence of activities of each feasible shift and the working periods of the planning horizon covered by each shift.

1.2 IMAGE SEGMENTATION PROBLEM

In recent years, Image Segmentation Problem (ISP) has been a very useful tool in image processing; specifically to respect to pattern recognition, detection of objects and edge detection. ISP consist of dividing an image into groups of similar characteristics (color, intensity, texture, etc.) with the objective to obtaining a new representation of the image, to facilitate its analysis.

The solution of an ISP is the set of segments that represent the objects of the image. For example, let us suppose we want to detect the lamp in the image presented in Figure 1.4a, then the IS algorithm must return a segment like the one presented in Figure 1.4b in red.

In this work, we consider the Image Segmentation Problem (ISP) which consist of detecting an image. In order to evaluate the algorithm, we select a set of images



(a) Original image

(b) Object to detect

Figure 1.4: Example 1

to detect an object and its respective ground truth, which is represented by the desired solution. In this order, the objective is to maximize the similarity between the segment that represents the object, obtained by the segmentation algorithm, and the ground truth of the tested images.

Additionally, if we consider the set of all the possible segments of an image; then the ISP can be seen as the problem of finding the segment that covers the object presented in the ground truth image, with the objective of minimizing the number of pixels which are not covered by the ground truth. Note that if the image size grows, then the number of the possible segments increase considerably. For this

reason, we propose to use the information provided by the histogram of the image, in order to make tractable this problem.

1.3 MOTIVATION

The SCP is a well known and studied problem in operations research, therefore the use of the models and the tools proposed, in the literature, to solve this problem can help to provide an efficient solution to other problems and, in particular the MASSP and the ISP.

On one hand, providing a good quality solution to the MASSP will help with the hard task of managing workforce of a company and, consequently, will reduce the operational cost. Furthermore, the efficient management of human resources will define whether the demand of an activity will be met or not. A highly unmet demand of an activity impacts directly in the utilities of the company and in the costumer service.

On the other hand, proposing an algorithm that solves the ISP efficiently, will help to simplify the analysis of an image. In particular, the object detection in the image can be useful in several applications. For example, in medical imaging where the diagnostic of cancer is made from the images provided by a Magnetic Resonance Imaging MRI, the detection of anomalies in the digital image will help in the early diagnosis of cancer and, consequently, will reduce the mortality due to this disease.

1.4 OBJECTIVE

The main objective of this research is to present a model based on SCP for the MASSP and the ISP. Moreover, we design and propose solution methods using the knowledge of the SCP, in order to provide efficient solutions to both problems.

Particularly, the objectives for the ISP is to introduce a mathematical model based on integer programming, and to present an automatic algorithm capable to obtain the number of segments of an image without the guidance of the user.

Finally, the algorithms proposed in this thesis are tested and compared with the best methods found in the literature.

1.5 ORGANIZATION

This thesis is organized as follows: In Chapter 2, we present a literature review of the problems treated in this work; Chapter 3 shows some definitions and notations used to formulate the SCP models for the MASSP and the ISP; Chapter 4 describe the SCP model for the MASSP, the matheuristic proposed to solve this problem, and the experimental results of the solution method; Chapter 5 introduces a new bi-objective set covering model for the image segmentation problem, proposes an AUGMECON algorithm to solve it, and presents the respective experimental results. Finally, in chapters 6 and 7 the conclusions and the future work are discussed.

CHAPTER 2

LITERATURE REVIEW

In this chapter, we present a literature review of the most recent and relevant works related to the SCP, MASSP, and IS.

2.1 SET-COVERING PROBLEM

The SCP is a well studied problem in operations research. Christofides and Korman (1975), Vemuganti (1998), Caprara et al. (2000), and Farahani et al. (2012) present a compilation of algorithms and applications of the SCP. Christofides and Korman (1975) and Caprara et al. (2000) show a classification of heuristic and exact methods proposed to solve the SCP. Vemuganti (1998) makes a review of more than 900 references to show the application of the SCP in capital budgeting, crew scheduling, cutting stock, facility location, personnel scheduling, vehicle routing, and timetable scheduling among others. Farahani et al. (2012) present a collection of 160 works that show the use of SCP to formulate facility location problems.

Additionally, metaheuristics have been used to solve the SCP. The most used algorithms are bioinspired metaheuristics.

Among other algorithms to solve the SCP, we can find the ones based in Lagrangian relaxation. Balas and Ho (1980) design a branch and bound method, where

the reduced costs obtained from solving the relaxation with the subgradient method are used to fix variables to generate feasible solutions to the SCP.

Beasley (1990) proposes a heuristic method, in which the reduced costs obtained from the subgradient method are used to generate promising columns to the problem. Balas and Carrera (1996) design a heuristic, in which the subgradient method is embedded into the branch and bound scheme. The difference between this method and the one presented by Balas and Ho (1980) is that this method makes a preselection of the columns. Later, they branch over the columns with a reduced cost equal to zero and with Lagrangian multipliers strictly positive.

Ceria et al. (1998), Caprara et al. (1999), and Yagiura et al. (2006) use the reduced costs of the Lagrangian relaxation to fix to zero the variables with a reduced cost greater to a threshold and, consequently, reducing the size of the problem. The resultant variables are fixed by using a greedy procedure.

Umetani and Yagiura (2007) solve the problem using a variant of the column generation procedure known as sifting method (Bixby et al., 1992). This algorithm use the information of the reduced costs of the Lagrangian relaxation to fix rows of the problem. Later, they fix variables using the reduced costs, then they generate feasible solutions using a greedy heuristic.

Caserta (2007) designs a tabu search using an algorithm based on Lagrangian relaxation in the intensification phase. Subsequently, he computes the reduced costs, in order to generate a subproblem similar to the one handled by Umetani and Yagiura (2007) and, finally, a feasible solution is produced by fixing variables and using a greedy heuristic.

The papers reviewed in this section show that the solution methods based on Lagrangian relaxation are efficient to solve SCP. For this reason, in this work we propose to use the Lagrangian relaxation to solve the MASSP using the SCP model for this problem similar to the ones presented by Côté et al. (2013) and Boyer et al. (2014).

2.2 THE MULTI-ACTIVITY SHIFT SCHEDULING PROBLEM

To the best of our knowledge, few researchers have addressed this problem. Demassez et al. (2006) model the subproblems of a column generation to solve the MASSP using automatas and constraint programming. Lequy et al. (2012a) propose three integer programming models and two algorithms to solve them; the first method consists in a branch and bound for solving small instances, and the second is based on a rolling-horizon heuristic for large instances. In this paper, they consider the case where the working periods of the shifts are fixed a priori.

Quimper and Rousseau (2010) design a large-neighborhood search and use formal languages and context-free grammars to model the constraints of the shifts. Dahmen and Rekik (2012) use a branch and bound procedure for the improvement, intensification, and diversification phases of a tabu search algorithm for solving the MASSP. Côté et al. (2011a), Côté et al. (2011b) and Côté et al. (2013) propose two grammar-based models and use a branch-and-price approach to solve them. In the paper of Côté et al. (2013), the results obtained for the instances of Demassez et al. (2006) and Lequy et al. (2012a) show an improvement with respect of the solutions reported in the literature. This algorithm is able to solve instances with up to 100 employees and 10 activities over a planning horizon of 7 days, although in the worst case it takes more than two hours to obtain a solution with a relative gap of 1%. Côté et al. (2013) propose to solve the linear relaxation as a restricted master problem (RMP) with a column generation which is embedded into a branch and price scheme to solve the MASSP. Finally, Restrepo et al. (2012) present a case of study of the MASSP which consists in assigning the employees to car parks in Bogotá, Colombia. They use a column generation algorithm coupled with an auxiliary shortest path problem with resource constraints to solve this problem.

Elahipanah et al. (2013) present the variant of the MASSP considering multiple

tasks and use a two-phase heuristic for solving it; in the first phase, a partial shift is generated where tasks are assigned using a mixed integer linear model, and in the second phase, activities are assigned using a rolling-horizon procedure. Lequy et al. (2012b) solve the multi-task MASSP using a two-stage heuristic. In the first stage, the authors use a mixed integer linear model in order to assign the tasks, and in the second they employ a column generation heuristic to assign activities and reassign the tasks. Boyer et al. (2014) present an multi-task MASSP with precedence constraints. The authors propose two formulations for the precedence constraints on the tasks and three branching strategies.

To the best of our knowledge, there is no work in the literature that uses algorithms based on the Lagrangian relaxation. For this reason, we propose to use the Lagrangian relaxation to design an algorithm capable to compete with the best algorithms found in the literature for solving the MASSP, especially in the larger instances.

2.3 IMAGE SEGMENTATION

In the literature, there are several applications and solution algorithms for the Image Segmentation (IM). Pal and Pal (1993), Udupa et al. (2006), Wei and Mandava (2010), and Garcia et al. (2015) present a recompilation of papers that use the IM for the digital image analysis. In particular, Pham et al. (2000), Noble and Boukerroui (2006), Heimann and Meinzer (2009), Gurcan et al. (2009), Ma et al. (2010), and Menze et al. (2015) show a set of papers of the medical application of the IM. In these reviews, the IM is used to analyze medical images for the detection of cancer, cardiovascular diseases, and other medical conditions.

The k -means based algorithms (see Ng et al. (2006), Piqueras et al. (2015) and Sammouda et al. (2015)) and artificial neural networks (see Torbati et al. (2014), Havaei et al. (2015) and Zhang et al. (2015)) have been extensively used for the image

segmentation. Furthermore, there are several articles that show the importance of the information provided by the histogram in order to design efficient segmentation algorithms. Tobias and Seara (2002) propose to use a threshold in order to divide the histogram of a gray scale image; a threshold is the critical value where the histogram is cut. Also, in order to improve the quality of the segmentation, they suggest to use fuzzy sets. The weakness of this algorithm is that it only works efficiently over images with bimodal or nearly bimodal histogram. Tan and Isa (2011), Rajinikanth and Couceiro (2015), Bhandari et al. (2016), and He and Huang (2017) segment RGB images by computing the histogram of each channel of color and applying a threshold algorithm over each channel histogram.

Tan and Isa (2011) propose a fuzzy k -means algorithm which considers the information of a multithresholding histogram algorithm in order to handle with the sensibility of the algorithm. The experimental results show that this approach is able to improve the results over the tested datasets, and provides a fewer number of segments and more homogeneous sections in the segmentation. Rajinikanth and Couceiro (2015), and He and Huang (2017) obtain the optimal threshold for the three channels of color by using a firefly algorithms; the results show effectiveness of the methods for multilevel color image segmentation.

Finally, Bhandari et al. (2016) design four algorithms to segment RGB images: the differential evolution, the wind driven optimization, the particle swarm optimization, and the cuckoo search algorithm. These algorithms are based on the information of the histogram of each channel of color and they are used to find the optimal threshold combination. The algorithms proposed are tested over a satellital and natural image dataset, the disadvantage of the methods is that they need the number of thresholds to divide the histogram.

Among the most recent work found in the literature are Pont-Tuset et al. (2017), Zheng et al. (2015), and Liu et al. (2015). Pont-Tuset et al. (2017) design a multiscale combinatorial group that uses a hierarchical segmentation algorithm to

detect objects in the images. This approach is able to provide contours, hierarchical regions, and object proposals within 5 seconds. Zheng et al. (2015) present a generalized hierarchical fuzzy k -means algorithm, which considers the spatial information and the pixel value with the objective to perform a segmentation over images with noise and with non euclidean distance data. They consider this type of images given that its characteristics are the reason of the malfunctioning of the k -means and fuzzy k -means algorithms. The results of this algorithm show that it is able to find good segmentations that cannot be assured with the standard fuzzy k -means. Liu et al. (2015) use the information of a Markov random field to perform a semantic segmentation which is solved by using a convolutional neural network. This approach achieves a 77.5% average accuracy on the detection of objects on the tested dataset.

To the best of our knowledge, there are algorithms that solve clustering problem, however there is no work in the literature that makes use of an optimization model in order to segment an image. The main idea lays on the exploitation of the information provided by the histogram of an image to design a bi-objective integer programming model based on the set-covering formulation for the segmentation of gray scale images.

CHAPTER 3

THEORETICAL FRAMEWORK

3.1 THE MULTI-ACTIVITY SHIFT SCHEDULING PROBLEM

In this section, we present definitions and some notations that are used to formulate the SCP model for the MASSP. For more details of context-free grammar see Hopcroft et al. (2001) and Sipser (2006).

3.1.1 CONTEXT-FREE GRAMMAR

A context-free grammar G is defined by the tuple (Σ, N, P, S) where

- Σ is an alphabet of symbols, called terminals;
- N is a set of nonterminal symbols;
- P is a set of productions of the form $X \rightarrow \alpha$, where $X \in N$ and α is a sequence conformed by terminal and nonterminal symbols;
- S is the starting nonterminal symbol.

A context-free grammar provides the rules and the symbols to generate a language, which is conformed by a set of sequences of terminal symbols named words. The set of productions rules is defined by the relations between the nonterminals symbols and a sequence of terminals/nonterminal symbols. This relation determines if a word belongs to the language or not.

EXAMPLE 3.1 *If we consider the grammar $G(\Sigma, N, P, S)$ where:*

- $\Sigma = a, b;$
- $N = A, B;$
- *and P is conformed by:*

$$- S \rightarrow AB, A \rightarrow AA|a \text{ and } B \rightarrow BB|b$$

then G produces the language $\{a^n b^m : n, m > 0\}$. The symbol $|$ represents the logical OR operator.

From example 3.1, if we want to generate the words of length three; then the production rules should be applied iteratively, starting from symbol S , until a sequence of three terminal symbols is produced. Table 3.1 shows the two possible words that can be generated with grammar G in example 3.1. The column P displays the production rules applied and the column Result gives the produced sequences after applying the production rules. Note that the word aab is produced by applying the production $S \rightarrow AB$; then the sequence AAB can be derived from the production rule $A \rightarrow AA$, and finally each nonterminal symbol A and B can be substituted by the terminals a and b by performing the productions $A \rightarrow a$ and $B \rightarrow b$, respectively. Moreover, one should consider that each terminal/nonterminal symbol represents a node in a graph and the outgoing arcs of a node depict the execution of a production rule; consequently, a derived word of a language can be represented by a tree, named parse tree. Figure 3.1 presents the parse trees of the words aab and abb .

Table 3.1: Derivation of the words aab (a) y abb (b)

(a)		(b)	
P	Result	P	Result
-	S	-	S
$S \rightarrow AB$	AB	$S \rightarrow AB$	AB
$A \rightarrow AA$	AAB	$B \rightarrow BB$	ABB
$A \rightarrow a$	aAB	$B \rightarrow b$	ABb
$A \rightarrow a$	aaB	$B \rightarrow b$	Abb
$B \rightarrow b$	aab	$A \rightarrow a$	abb

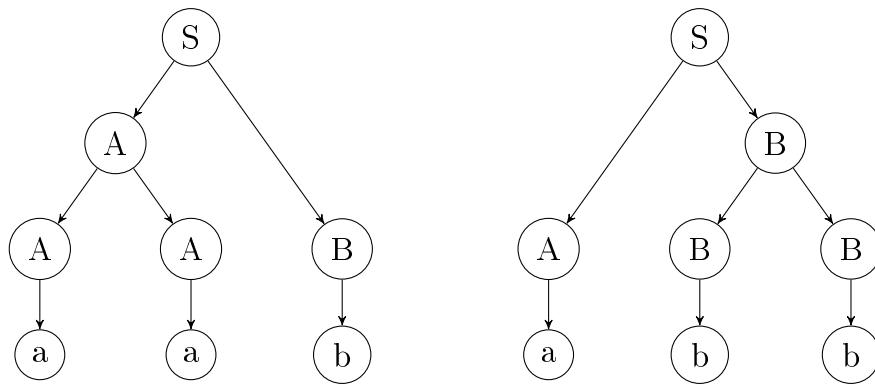


Figure 3.1: Parse trees of the words aab and abb

Finally, all the parse trees of the words of a certain length can be embedded in a Directed Acyclic Graph (DAG Γ). Figure 3.2 shows the DAG Γ of the words of three letters produced with the grammar G ; the DAG Γ is an AND/OR graph where the AND-nodes are denoted with A and the OR-nodes with O . The nodes $A_{i,j}^{p,n}$ represent the AND-node that applies the production rule p to generate the n -th sequence starting in the position i with length j . Besides, the nodes O_{ij}^X is the OR-node represented by the terminal/nonterminal symbol X , which derives a sequence starting in the position i with length j . Note that the inner AND-nodes and OR-nodes represent production and non-terminal symbols, respectively; whilst the leaves of the DAG Γ correspond to the terminals symbols in their respective

position in the word. In order to derive any word with the DAG Γ of this example, we should start in the $O_{1,3}^S$ and choose one outgoing arc from the OR-nodes; then we must select all the outgoing arcs, from the AND-nodes, until the leaves of the DAG Γ are reached.

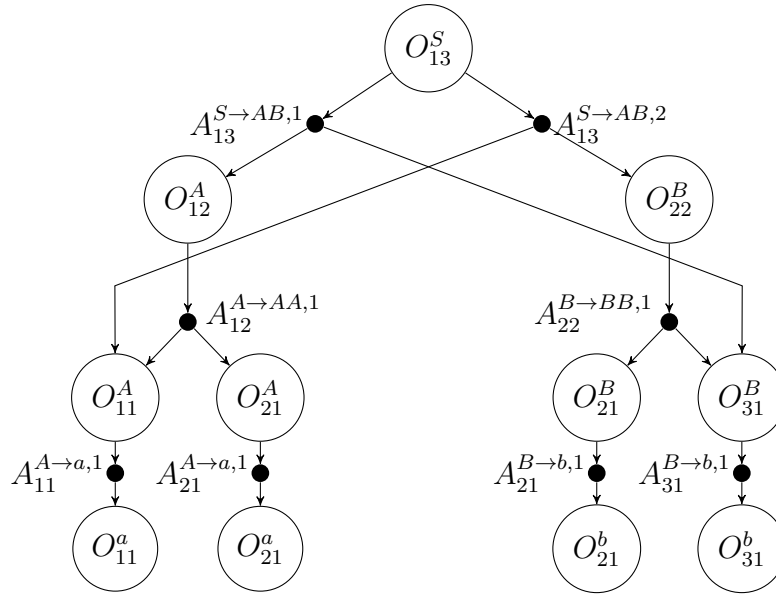


Figure 3.2: DAG Γ of the words of three letters

In the MASSP, a shift is represented by the sequence of the activities to be performed; additionally, if the activities are represented as terminal symbols and the constraint of the shifts are modeled by production rules, then the shifts of the MASSP can be modeled using a context-free grammar. Modeling a shift with a context-free grammar will allow to produce a feasible shift easily.

EXAMPLE 3.2 *Let us suppose there is an employee who has a shift with 4 periods of one hour and he/she is able to perform two activities (a_1 and a_2). Let us consider that activity a_1 must be perform before a_2 , and the activities cannot be assigned for less than 2 hours. Consequently, the feasible shifts for this employee can be modeled by the following context-free grammar G :*

- $\Sigma = \{a_1, a_2\}$;
- $N = \{J_1, J_2, A_1, A_2\}$;

- P is defined by:

$$\begin{aligned}
 S &\rightarrow_{[4,4]} J_1 J_1 | J_1 J_2 \\
 J_1 &\rightarrow A_1 A_1 & J_2 &\rightarrow A_2 A_2 \\
 A_1 &\rightarrow A_1 a_1 | a_1 & A_2 &\rightarrow A_2 a_2 | a_2.
 \end{aligned}$$

where the symbol “ $\rightarrow_{[l,r]}$ ” indicates that the sequence derived from this production must have a length between l and r .

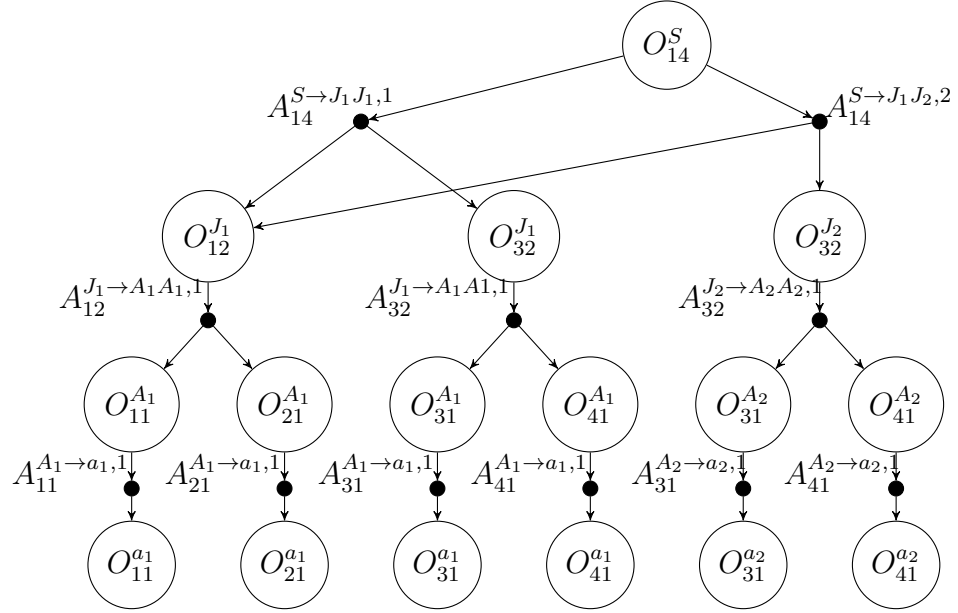
To generate the sequences of length four with the grammar of the example 3.2, the production rules presented in Table 3.2 must be applied. Consequently, this grammar produces the two feasible shift for the employee: $a_1 a_1 a_1 a_1$ and $a_1 a_1 a_2 a_2$. Figure 3.3 show the DAG Γ representation for the feasible shifts.

Table 3.2: Derivations of the shifts $a_1 a_1 a_1 a_1$ (a) and $a_1 a_1 a_2 a_2$ (b).

(a)		(b)	
P	Result	P	Result
-	S	-	S
$S \rightarrow J_1 J_1$	$J_1 J_1$	$S \rightarrow J_1 J_2$	$J_1 J_2$
$J_1 \rightarrow A_1 A_1$	$A_1 A_1 J_1$	$J_1 \rightarrow A_1 A_1$	$A_1 A_1 J_2$
$J_1 \rightarrow A_1 A_1$	$A_1 A_1 A_1 A_1$	$J_2 \rightarrow A_2 A_2$	$A_1 A_1 A_2 A_2$
$A_1 \rightarrow a_1$	$a_1 A_1 A_1 A_1$	$A_1 \rightarrow a_1$	$a_1 A_1 A_2 A_2$
$A_1 \rightarrow a_1$	$a_1 a_1 A_1 A_1$	$A_1 \rightarrow a_1$	$a_1 a_1 A_2 A_2$
$A_1 \rightarrow a_1$	$a_1 a_1 a_1 A_1$	$A_2 \rightarrow a_2$	$a_1 a_1 a_2 A_2$
$A_1 \rightarrow a_1$	$a_1 a_1 a_1 a_1$	$A_2 \rightarrow a_2$	$a_1 a_1 a_2 a_2$

3.2 THE IMAGE SEGMENTATION PROBLEM

In this section we present the definitions and notations that will be used to model the ISP by a SCP model.

Figure 3.3: DAG Γ representation for the feasible shifts

3.2.1 MODEL

The ISP consists of dividing the image into groups of pixels with similar characteristics (color, intensity, texture, etc.) with the objective to obtain a new representation of the image to make easier its analysis.

For this purpose, we propose the next mixed integer linear programming model:

$$\text{Min } z = \sum_{i=1}^N \sum_{j=1}^i d_{ij} x_{ij} \quad (3.1)$$

subject to

$$x_{ij} = x_{ji} \quad \forall i, j = 1, \dots, N \quad (3.2)$$

$$\sum_{i=1}^N x_{ij} \geq 1 \quad \forall j = 1, \dots, N \quad (3.3)$$

$$x_{ij} + x_{jk} \leq 1 + x_{ik} \quad \forall i, j, k = 1, \dots, N \quad (3.4)$$

$$x_{ij} + x_{jk} \geq 2x_{ik} \quad \forall i, j, k = 1, \dots, N \quad (3.5)$$

$$x_{ij} \in \{0, 1\} \quad \forall i, j = 1, \dots, N \quad (3.6)$$

where d_{ij} is a parameter that measures the dissimilarity between pixel i and pixel j , and N is the total number of pixels in the image. Furthermore, the decision variables are defined as follows:

$$x_{ij} = \begin{cases} 1 & \text{if } i\mathcal{R}j \\ 0 & \text{otherwise} \end{cases}$$

where the notation $i\mathcal{R}j$ indicates if pixel i is related with pixel j or not; i.e. if the pixels i and j are assigned to the same group.

Constraints (3.2) ensure that pixel i will be related with pixel j if and only if pixel j is related with pixel i . Constraints (3.3) establish that pixel j is related with at least another pixel. In other words, these constraints forbid groups with only one pixel. Likewise, constraints (3.4) and (3.5) settle a transitive relation between pixels; i.e. if pixel i is related with j and j is related with k , then i is related with k . The objective (3.1) of this model minimizes the dissimilarity between the related pixels.

Finally, it is relevant to point out that the cardinality of the set of constraints (3.4) and (3.5) grows in order $\mathcal{O}(2N^3)$. This means that if we want to segment an image of 1 megapixel (1 million of pixels), the model would have 1×10^{12} variables and more than 2×10^{18} constraints, which makes the model unsolvable and unpractical to implement. For this reason, we propose to use the information provided by the image histogram under the hypothesis that this information contains relevant characteristics to make an efficient segmentation.

3.2.2 THE HISTOGRAM

The main idea of this work lies in the use of the information provided by the histogram of an image, in order to reduce the number of variables and constraints involved in the mathematical model. The histogram is a graphical representation of the distribution of the intensity of the pixel of an image. Figure 3.4 shows an image

and its respective histogram. Note that the histogram is composed by different distributions, for example the distribution of pixels over the interval $[0, 130]$ is different to the one found in $[130, 200]$.

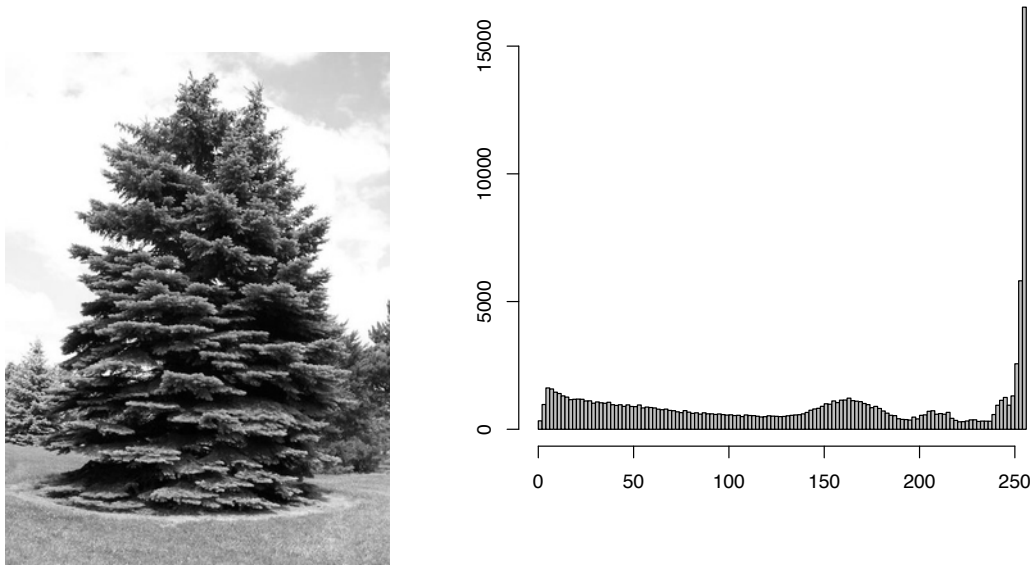


Figure 3.4: Example of a histogram of an image

Figure 3.5 shows the distribution of the interval $[0, 130]$ and its respective pixels represented in the original image. It is easy to note that the tree in the image is contained in the distribution over the interval $[0, 130]$ in the histogram. This observation gives us a clue that a good partition of the histogram can provide an efficient segmentation for the image.

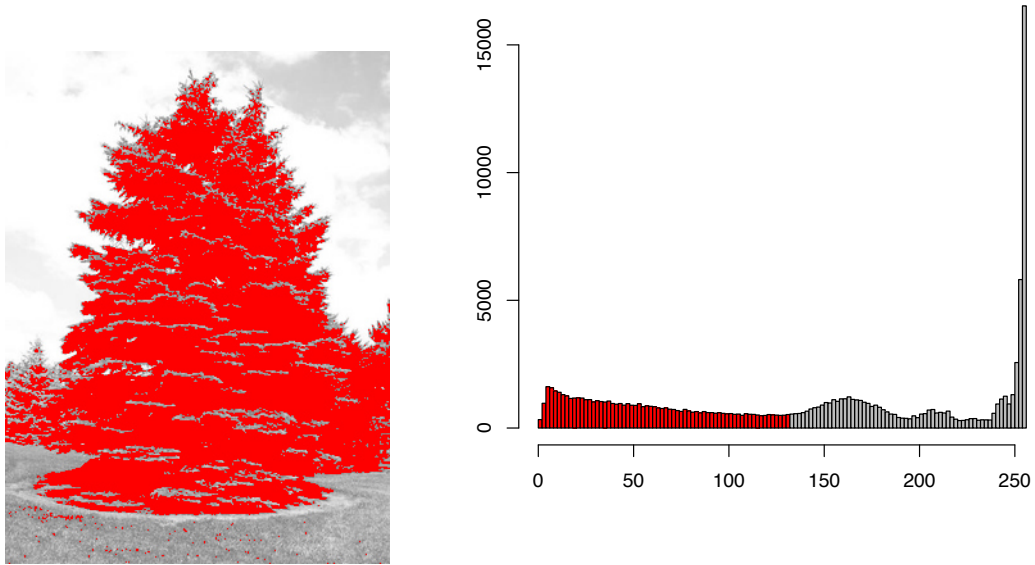


Figure 3.5: Example of a distribution of the histogram of Figure 3.4

CHAPTER 4

THE MULTI-ACTIVITY SHIFT SCHEDULING PROBLEM

In this chapter, we present the mathematical model for solving the MASSP. Besides, we describe the solution method proposed to solve this problem. Finally, we test the proposed algorithm, and compare the experimental results with the best results presented in the literature.

In this problem, we consider a set E of available employees; a set Ω^e of all feasible shifts of employee $e \in E$; a set A of activities to be performed; a set I of time periods in the planning horizon, and b_{ia} which is the demand of activity $a \in A$ at period $i \in I$. The set of feasible shifts Ω^e of each employee $e \in E$ is determined by the abilities of employee e , the duration of the shift, the resting periods, and the company policies. Moreover, each activity in shift s must be performed between a minimum and a maximum number of consecutive periods.

Additionally, we consider parameter δ_{ias}^e which is equal to one if activity $a \in A$ is performed at period $i \in I$ in shift $s \in \Omega^e$ of employee $e \in E$, and zero otherwise. The planning horizon I is discretized into periods of equal length. Besides, each shift $s \in \Omega^e$ has a cost c_s^e that considers the transition between activities in the shift S , and the undercovering and overcovering demand b_{ia} of activity a at period i are

penalized by c_{ia}^u and c_{ia}^o , respectively.

The objective of the MASSP is to assign a shift $s \in \Omega^e$ to each employee $e \in E$, so as to minimize the cost of assigning the shift s and the total undercovering and overcovering costs.

From these parameters, the MASSP can be modeled with the following Set-Covering Model (SCM) proposed by Côté et al. (2013):

$$\text{(SCM) Min } z = \sum_{e \in E} \sum_{s \in \Omega^e} c_s^e x_s^e + \sum_{i \in I} \sum_{a \in A} (c_{ia}^u u_{ia} + c_{ia}^o o_{ia}) \quad (4.1)$$

subject to

$$\sum_{e \in E} \sum_{s \in \Omega^e} \delta_{ias}^e x_s^e + u_{ia} - o_{ia} = b_{ia} \quad i \in I, a \in A \quad (4.2)$$

$$\sum_{s \in \Omega^e} x_s^e = 1 \quad e \in E \quad (4.3)$$

$$x_s^e \in \{0, 1\} \quad e \in E, s \in \Omega^e \quad (4.4)$$

$$u_{ia} \geq 0 \quad i \in I, a \in A \quad (4.5)$$

$$o_{ia} \geq 0 \quad i \in I, a \in A \quad (4.6)$$

where

- $x_s^e = 1$ if shift s is assigned to employee e , and 0 otherwise.
- u_{ia} represents the undercovering of activity a in period i .
- o_{ia} represents the overcovering of activity a in period i .

The objective (4.1) minimizes the cost of assigning a shift to an employee and the overcovering and undercovering costs. Constraints (4.2) ensure that the demand for each activity a in each period i is satisfied. Constraints (4.3) ensure that only one shift is assigned to each employee.

The set Ω^e can be modeled by a context-free grammar and represented by its DAG Γ as proposed by Côté et al. (2013). Notice, that the feasible shifts are not explicitly generated by modeling the set Ω^e with a context-free grammar. Nevertheless, the DAG Γ representation will be useful to solve this MASSP in the next sections.

4.1 SOLUTION METHOD

In this section, we present our matheuristic (MH), which uses Lagrangian relaxation and the subgradient method to generate promising shift for the MASSP. Later, these shifts are used to build a restricted SCM where the sets Ω^e are replaced by $\Omega'_e \subseteq \Omega^e$; notice that the restricted model is easier to solve because the size of Ω'_e is considerably smaller than Ω^e .

The principal reason for using Lagrangian relaxation to generate the shifts for the MASSP lays under the hypothesis that the Lagrangian relaxation is able to produce good quality shifts. Given that at each iteration of the subgradient method the updated Lagrangian multipliers generate shifts that cover the demand of activities that were not covered in the previous iteration.

4.1.1 LAGRANGIAN RELAXATION

In this work, we propose to relax the covering constraints (4.2) because they make the SCM difficult to solve. The Lagrangian relaxation of the SCM (called RSCM) is as follows:

$$\begin{aligned}
 \text{(RSCM)} \quad \text{Min } z = & \sum_{e \in E} \sum_{s \in \Omega^e} c_s^e x_s^e + \sum_{i \in I} \sum_{a \in A} (c_{ia}^u u_{ia} + c_{ia}^o o_{ia}) \\
 & + \sum_{i \in I} \sum_{a \in A} \lambda_{ia} \left(\sum_{e \in E} \sum_{s \in \Omega^e} \delta_{ias}^e x_s^e + u_{ia} - o_{ia} - b_{ia} \right)
 \end{aligned}$$

subject to

$$\begin{aligned} \sum_{s \in \Omega^e} x_s^e &= 1 & e \in E \\ x_s^e &\in \{0, 1\} & e \in E, s \in \Omega^e \\ u_{ia} &\geq 0 & i \in I, a \in A \\ o_{ia} &\geq 0 & i \in I, a \in A \\ \lambda_{ia} &\in \mathbb{R} & i \in I, a \in A \end{aligned}$$

where λ_{ia} are the Lagrangian multipliers that can take any real value because of the nature of the relaxed constraints.

The RSCM can be simplified as follows:

$$\begin{aligned} \text{Min } z &= \sum_{i \in I} \sum_{a \in A} ((c_{ia}^u + \lambda_{ia}) u_{ia} + (c_{ia}^o - \lambda_{ia}) o_{ia}) \\ &+ \sum_{e \in E} \sum_{s \in \Omega^e} \left(c_s^e + \sum_{i \in I} \sum_{a \in A} \lambda_{ia} \delta_{ias}^e \right) x_s^e \\ &- \sum_{i \in I} \sum_{a \in A} \lambda_{ia} b_{ia} \end{aligned}$$

subject to

$$\begin{aligned} \sum_{s \in \Omega^e} x_s^e &= 1 & e \in E \\ x_s^e &\in \{0, 1\} & e \in E, s \in \Omega^e \\ u_{ia} &\geq 0 & i \in I, a \in A \\ o_{ia} &\geq 0 & i \in I, a \in A \\ \lambda_{ia} &\in \mathbb{R} & i \in I, a \in A \end{aligned}$$

We propose to use the classical subgradient method (Shor, 1985) to solve the RSCM. Note that if the Lagrangian multipliers are fixed, then the RSCM can be divided into two subproblems. On one hand, SubP1 determines the values of the covering variables; On the other hand, SubP2 assigns a shift to each employee, such

that the cost of assigning the shifts and the total weight of performing each activity at each period are minimized. SubP1 and SubP2 are defined as follows:

$$\begin{aligned} \text{(SubP1}(\Lambda)) \text{ Min } z_1 = & \sum_{i \in I} \sum_{a \in A} (c_{ia}^u + \lambda_{ia}) u_{ia} + \sum_{i \in I} \sum_{a \in A} (c_{ia}^o - \lambda_{ia}) o_{ia} \\ & - \sum_{i \in I} \sum_{a \in A} \lambda_{ia} b_{ia} \end{aligned}$$

subject to

$$\begin{aligned} u_{ia} & \geq 0 & \forall i \in I, a \in A \\ o_{ia} & \geq 0 & \forall i \in I, a \in A \end{aligned}$$

$$\text{(SubP2}(\Lambda)) \text{ Min } z_2 = \sum_{e \in E} \sum_{s \in \Omega^e} \left(c_s^e + \sum_{i \in I} \sum_{a \in A} \lambda_{ia} \delta_{ias}^e \right) x_s^e$$

subject to

$$\begin{aligned} \sum_{s \in \Omega^e} x_s^e & = 1 & \forall e \in E \\ x_s^e & \in \{0, 1\} & \forall e \in E, s \in \Omega^e \end{aligned}$$

At the beginning of the subgradient method, the Lagrangian multiplier should be fixed. Later, SubP1 and SubP2 are solved to recompute the multipliers. Finally the steps of solving the subproblems and recomputing the multipliers are iterated until a given maximum number of iterations is reached or when the lower bound cannot be improved.

Notice that the shifts generated by SubP2 are also feasible for the SCM, given that the DAG Γ for each employee considers the constraints of a valid shift. The unfeasibility of the solution provided by RSCM lays on the values of the covering variables.

4.1.1.1 SOLVING THE SUBPROBLEMS

SubP1 has one set of constraints which state that the decision variables u_{ia} and o_{ia} are positive, $\forall i \in I$ and $\forall a \in A$. Furthermore, the u_{ia} and o_{ia} variables can be bounded, given that the maximum value of u_{ia} occurs when there are no shift performing activity a at period i , and the maximum value of o_{ia} take place when all the employees are performing activity a at period i . Hence, this variables are fixed to the maximum value if its respective coefficient is negative, and fixed to zero otherwise. Algorithm 1 shows the solution procedure for SubP1

Algorithm 1 Solution method for SubP1.

Input: Matrix Λ .

Output: Solution $[U, O]$.

```

1: for all  $i \in I$  do
2:   for all  $a \in A$  do
3:     if  $(c_{ia}^u + \lambda_{ia}) < 0$  then
4:        $u_{ia} \leftarrow b_{ia}$ 
5:     else
6:        $u_{ia} \leftarrow 0$ 
7:     end if
8:     if  $(c_{ia}^o - \lambda_{ia}) < 0$  then
9:        $o_{ia} \leftarrow |E| - b_{ia}$ 
10:    else
11:       $o_{ia} \leftarrow 0$ 
12:    end if
13:  end for
14: end for
15: return  $[U, O]$ 

```

In SubP2, the Lagrangian multiplier λ_{ia} can be seen as the weight of performing activity a at period i . The objective of SubP2 is to assign one shift to each employee,

such that the total weight and the cost of assigning the shifts are minimized. Since c_s^e is a parameter that depends on the transition cost, then the problem is reduced to finding the shift that minimizes the number of transitions and the total weight. Constraints (4.3) are trivially satisfied since we choose one shift for each employee.

To solve SubP2, we propose to use the dynamic programming procedure based on the DAG Γ by Côté et al. (2013). This algorithm consists of labeling each leave node corresponding to activity a at period i with weight λ_{ia} . Later, the weights should be backtracked in the DAG Γ to obtain the minimum weighted shift; where the OR-nodes are labeled with the minimum label of its children, and the AND-nodes are labeled with the sum of its children weights. Finally, when the root node is labeled, we track the path that generates the sequence of activities with minimum weight.

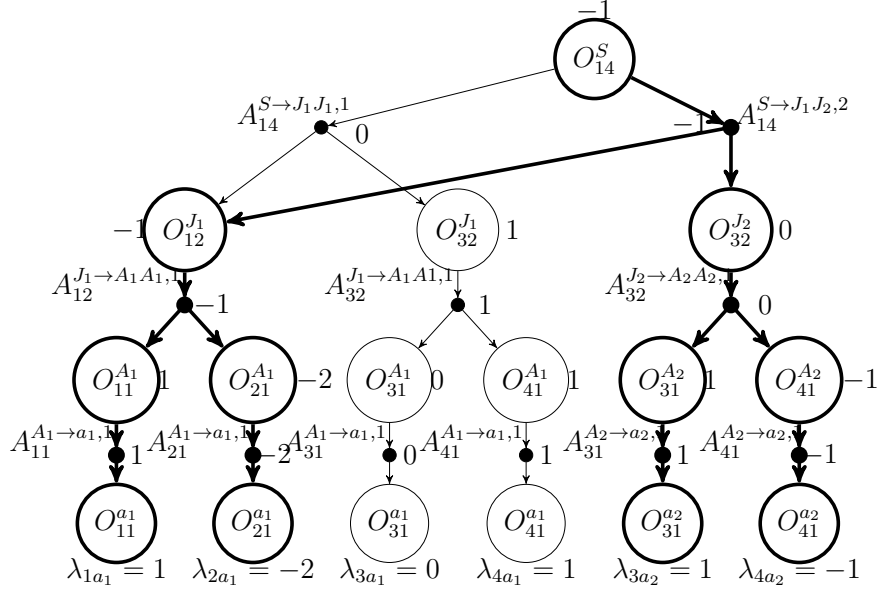
For instance, consider the DAG Γ from Figure 3.3 generated by the grammar presented in example 3.2 with the following values of the Lagrangian multipliers:

$$\Lambda = \begin{pmatrix} 1 & 0 \\ -2 & 0 \\ 0 & 1 \\ 1 & -1 \end{pmatrix}$$

The sequence with minimum weight is $a_1a_1a_2a_2$. Figure 4.1 shows the weight propagation that generates this solution; the path of this sequence is highlighted in bold. Note that transition costs are associated with any AND-node. Specifically, the transition of the sequence $a_1a_1a_2a_2$ occurs in the node $A_{14}^{S \rightarrow J_1 J_2, 2}$ in Figure 4.1. The transition cost is added to the label of the corresponding AND-node.

4.1.1.2 UPDATING THE LAGRANGIAN MULTIPLIERS

Once the subproblems are solved, the Lagrangian multiplier must be recomputed using the classical subgradient method introduced by Shor (1985). We use the

Figure 4.1: Cost propagation for DAG Γ .

following equation to update the Lagrangian multipliers at the k -th iteration (λ_{ia}^k):

$$\lambda_{ia}^k = \lambda_{ia}^{k-1} + \left[\frac{s_{ia}^{k-1} \cdot \epsilon_{k-1} (f_{k-1} - f'_{k-1})}{\|s_{ia}^{k-1}\|_2} \right] \quad (4.7)$$

where $\epsilon_{k-1} \in (0, 2]$, and f_{k-1} and f'_{k-1} are the best bounds for SCM and RSCM found at the $(k-1)$ -th iteration, respectively. Additionally,

$$s_{ia}^{k-1} = \sum_{e \in E} \sum_{s \in \Omega^e} \delta_{ias}^{e(k-1)} x_s^{e(k-1)} + u_{ia}^{k-1} - o_{ia}^{k-1} - b_{ia}.$$

where $x_s^{e(k-1)}$, $e \in E$ and $s \in \Omega_e$, denotes the RSCM solution obtained at the $(k-1)$ -th iteration.

Conventionally, parameter ϵ_1 is fixed to two, and it is divided by two after a given number of iterations without improvement in the relaxed bound value. The procedure stops when the Lagrangian multipliers do not causes significant changes in the solution of RSCM; i.e. when, at iteration k , $\epsilon_k < \varepsilon$, where ε is small enough.

The shifts generated with SubP2 at each step of the Lagrangian relaxation are feasible to MASSP; nevertheless, the quality of the solution is poor. Furthermore, the Lagrangian relaxation provides lower bounds for SCM which can be used to compute the optimality gap.

In algorithm 2 the subgradient method used to solve RSCM is presented.

Algorithm 2 Subgradient Method.

Input: RSCM, ε , N

Output: Set Ω

- 1: $\Omega \leftarrow \emptyset$
- 2: $\epsilon_R \leftarrow 2$
- 3: $LB \leftarrow -\infty$
- 4: $\Lambda \leftarrow 0_{|I|,|A|}$
- 5: **while** $\epsilon_R > \varepsilon$ **do**
- 6: $LB1 \leftarrow SubP1(\Lambda)$
- 7: $LB2 \leftarrow SubP2(\Lambda)$
- 8: $S \leftarrow GetSolution(SubP2(\Lambda))$
- 9: $\Omega \leftarrow \Omega \cup S$
- 10: $LB \leftarrow \max(LB, LB1 + LB2)$
- 11: **if** LB does not change in N iterations **then**
- 12: $\epsilon_R \leftarrow \epsilon_R/2$
- 13: **end if**
- 14: Update Λ
- 15: **end while**
- 16: **return** Ω

4.1.2 MATHEURISTIC

We now present a general description of MH. Under the hypothesis that the Lagrangian relaxation is able to obtain good-quality shifts at each step, then the new problem is to find the combination of these shifts that provides an efficient solution to SCM.

Model SSCM(Ω), which allows only a subset Ω of the feasible shifts, is as follows:

$$\begin{aligned}
(\text{SSCM}(\Omega)) \quad \text{Min } z = & \sum_{e \in E} \sum_{s \in \Omega'_e} c_s^e x_s^e \\
& + \sum_{i \in I} \sum_{a \in A} (c_{ia}^u u_{ia} + c_{ia}^o o_{ia})
\end{aligned}$$

subject to

$$\begin{aligned}
\sum_{e \in E} \sum_{s \in \Omega'_e} \delta_{ias}^e x_s^e + u_{ia} - o_{ia} &= b_{ia} & i \in I, a \in A \\
\sum_{s \in \Omega'_e} x_s^e &= 1 & e \in E \\
x_s^e &\in \{0, 1\} & e \in E, s \in \Omega'_e \\
u_{ia} &\geq 0 & i \in I, a \in A \\
o_{ia} &\geq 0 & i \in I, a \in A
\end{aligned}$$

where $\Omega = \bigcup_{e \in E} \Omega'_e$ and $\Omega'_e \subset \Omega^e$ is the set of feasible shifts for employee e , generated at each iteration of the subgradient method.

Algorithm 3 shows the MH procedure. The function `Subgradient_Method` refers to the method described in Section 4.1.1, and the function `Solve` uses an exact method (included in CPLEX) to solve $\text{SSCM}(\Omega)$. In figure 4.2 a flowchart of the whole procedure is presented.

Algorithm 3 Matheuristic based on Lagrangian relaxation.

Input: Problem data

Output: Solution Sol

- 1: $\Omega \leftarrow \text{Subgradient_Method}(RSCM)$
 - 2: $Sol \leftarrow \text{Solve}(SSCM(\Omega))$
 - 3: **return** Sol
-

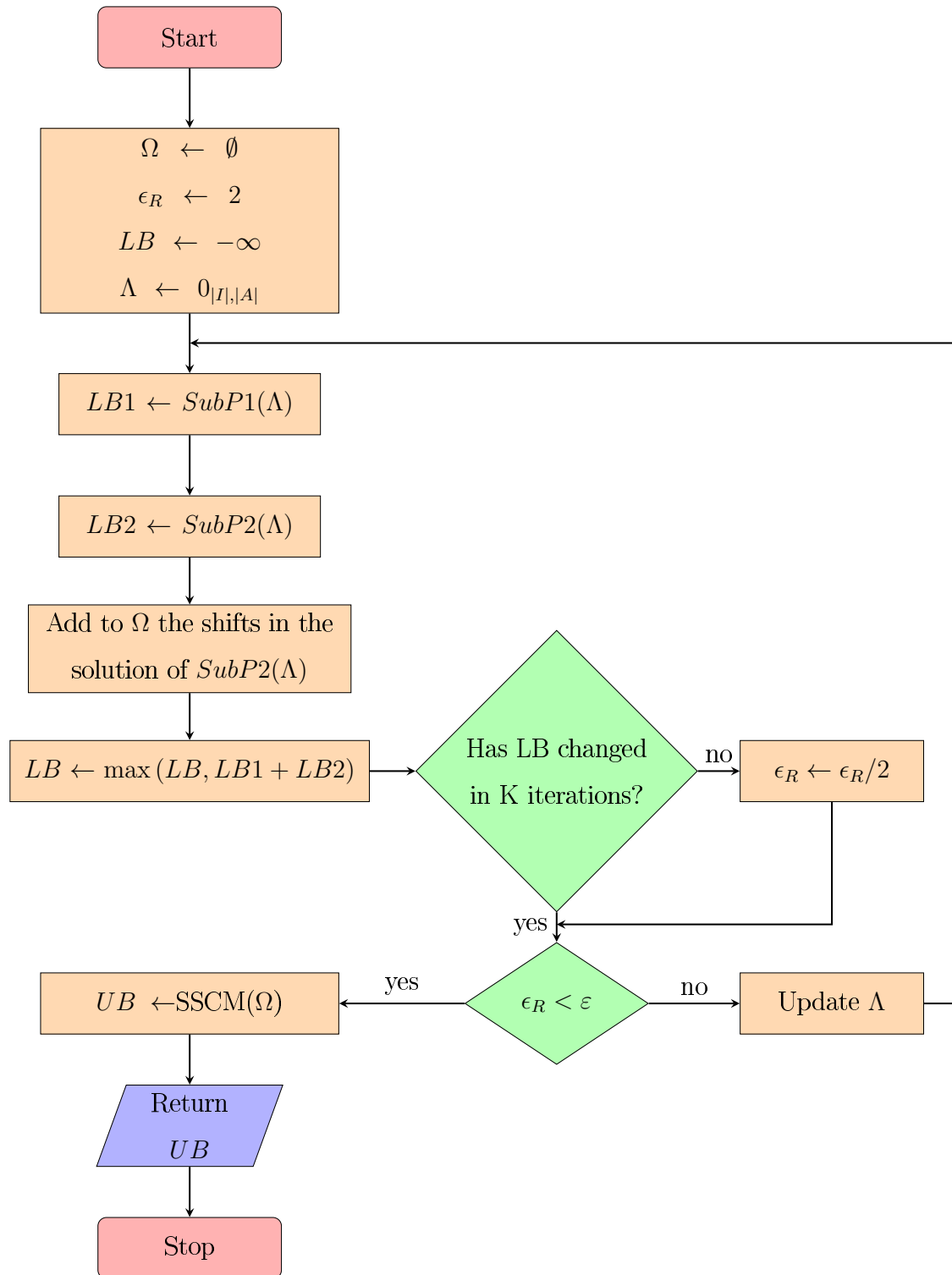


Figure 4.2: Flowchart of the Matheuristic

4.2 EXPERIMENTAL RESULTS

In this section, we present the experimental results obtained with the MH over two instance sets: the Demassey instances from Demassey et al. (2006) and the Lequy instances from Lequy et al. (2012a). The difference between these sets of instances is that Lequy et al. (2012a) consider shifts where the starting period and the duration are fixed; In Demassey et al. (2006) the shift can start at any period of the planning horizon and two durations for a shift are considered: part-time shifts and full-time shifts. In both sets the planning horizon is divided into periods of 15 minutes. The results obtained with the MH are compared with the ones reported by Côté et al. (2013), which report the best solution presented in the literature for both benchmarks and where obtained with a Branch and Price method (BP). Côté et al. (2013) execute the BP on an Intel Xeon 2.4 GHz with 48 GB of RAM and using CPLEX 11.2. For the BP, the Demassey results are obtained within a 2 hours limit; whilst, the Lequy instances in the classes 1,3,4 and 7 are solved within 1 hour limit, and for classes 2,5,6,y 8 the time is not limited.

The MH was coded using C++ and the SSCM(Ω) is solved using CPLEX 12.6. The experiments were performed on an Intel Xeon E5-2687W 3.1 GHz with 64 GB of RAM. We initialize the parameters $\epsilon_1 = 2$ and $\varepsilon = 0.0001$ for the Lagrangian relaxation. Additionally, the Lagrangian multipliers are updated every 375 and 75 iterations for the Demassey and the Lequy instances, respectively. CPLEX stops when the optimality gap and the partial solution of SSCM(Ω) is less than 1% or if after 300 seconds there is no improvement in the solution.

For the MH, the Demassey results are obtained within a 2 hours limit; whilst, the Lequy instances in the classes 1,3,4 and 7 are solved within 1 hour limit, and for classes 2,5,6,y 8 the time is not limited.

Furthermore, we have implemented a column generation heuristic (CGH). The CGH replaces the Lagrangian relaxation of the MH with a column generation of the

linear relaxation of SCM that uses the reduced costs to generate feasible shifts. Later, the set of shifts generated by the column generation are passed to the SSCM(Ω), which is solved as in the MH. The results obtained with CGH are compared with the corresponding results of MH.

4.2.1 PROBLEM INSTANCES

4.2.1.1 DEMASSEY INSTANCES

Demasse et al. (2006) introduce a set of 100 instances for the MASSP. This set is conformed by 10 classes of 10 instances each, based on the number of available activities, from 1 to 10. The employees have the ability to perform all the activities, and they can start to work at any period of the planning horizon considering that they should comply with a full-time or a part-time schedule. The part-time shift consists of a working time with a duration between 3 and 6 hours and includes a break of 30 minutes. The full time shift has between 6 and 8 hours of work with 2 breaks of 30 minutes and a lunch break of 1 hour. Finally, the breaks must be located between two different activities.

4.2.1.2 LEQUY INSTANCES

Lequy et al. (2012a) propose a set of 40 instances divided into 8 classes of 5 instances. The employees have skills and the starting period and the duration of a shift are fixed. Besides, the activities should be performed between a minimum and a maximum number of consecutive periods. Finally, the transitions in the shifts have a fixed cost.

Table 4.1 shows a description of each class: the first column indicates the name of the class, the second shows the number of working days, the third presents the

number of employees available and the fourth gives the number of activities to be performed. Note that class 1 has the same size as class 2, and class 4 has the same size as class 5. However, classes 1 and 4 have larger shifts and the employees have more qualifications.

Table 4.1: Classification of the Lequy instances.

Set of Instances	Days	Employees	Activities
Class 1	1	50	10
Class 2	1	50	10
Class 3	2	75	12
Class 4	7	20	5
Class 5	7	20	5
Class 6	7	50	7
Class 7	7	50	10
Class 8	7	100	15

4.2.2 MATHEURISTIC

4.2.2.1 DEMASSEY INSTANCES

Table 4.2 shows a summary of the solutions found by the MH, in the columns from 2 to 4, and the solutions reported by Côté et al. (2013), in the columns from 5 to 7. The column Average Time presents, for the MH, the average processing time tested over all the instances of each class; whilst, the average time for the BP is computed only considering the processing time of the instances solved within 0.01% gap, for the instances within 1% gap the limit time is reached. The columns NbS(0.01%) and NbS(1%) represent the number of instances of each class where the solution reports an optimality gap of 0.01% and 1%, respectively. For the MH, the optimality gap

is computed with the formula $100 \cdot (z - z_l)/z_l$, where z is the objective function value of the feasible solution bound obtained with the MH and z_l is the lower bound provided by the Lagrangian relaxation.

Table 4.2: Experimental results for the Demassey instances.

Instances	Matheuristic			BP		
	NbS(0.01%)	NbS(1%)	Average Time (s)	NbS(0.01%)	NbS(1%)	Average Time* (s)
Class 1	7	10	12.01	5	10	62.00
Class 2	8	9	75.64	6	9	100.00
Class 3	9	10	97.51	6	8	2074.00
Class 4	6	7	180.35	5	9	2096.00
Class 5	9	9	35.29	0	10	7200.00
Class 6	7	7	95.91	9	10	915.00
Class 7	7	9	131.64	5	9	2426.00
Class 8	4	7	113.77	7	10	2163.00
Class 9	7	10	54.61	5	7	1886.00
Class 10	6	8	172.14	6	8	3754.00
Average:	7	8.60	96.89	5.40	9	2267.60

*Average processing times of the BP are those reported by Côté et al. (2013)

The MH is able to find solution within 0.01% gap for 70 out of 100 instances, while the BP achieves this result only in 54 instances. Furthermore, MH solves all the instances within a 0.470% average gap. Although the experimental results were obtained with different machines, the experimental results show that MH is able to achieve better solution than the BP, without reaching the limit in the processing times.

4.2.2.2 LEQUY INSTANCES

Table 4.4 shows the results obtained with the MH for each instance; the column MH solution presents the feasible bounds obtained with the MH, and the column Gap(%) shows the optimality gap computed with the feasible bound of the MH and the lower bound provided by the Lagrangian relaxation. The columns related to BP show the solutions and the processing times reported by Côté et al. (2013). Observe

that the MH is able to find better or equivalent feasible bounds than the BP in 38 out of 40 instances; however, the instances where BP solution is better than the MH differ only in 15 units which represents the cost of one transition.

Table 4.3 summarizes the results showed in the table 4.4. The column Gap (MH/LR) presents the average gap between the MH solution and the Lagrangian relaxation bound; the column Gap(BP/LR) corresponds to the average gap between the BP solution and the Lagrangian relaxation bound, and Gap(BP/MH) shows the average gap between the BP bound and the MH bound.

Table 4.3: Average solution obtained with the matheuristic for the Lequy instances.

Instances	Gap(MH/LR)	Gap(BP/LR)	Gap(BP/MH)	Time	Time
	(%)	(%)	(%)	MH (s)	BP* (s)
Class 1	2.34	3.08	-0.70	49.72	1630.20
Class 2	1.79	1.63	0.16	9.06	53.78
Class 3	4.50	6.59	-1.92	2447.56	3600.00
Class 4	0.47	0.69	-0.21	94.81	411.40
Class 5	0.00	0.00	0.00	12.79	0.66
Class 6	0.28	0.28	0.00	231.07	36.59
Class 7	1.24	1.62	-0.37	1209.29	2014.80
Class 8	1.87	2.12	-0.24	2320.83	4936.84
Average:	1.56	2.00	-0.41	796.89	1585.53

*Average processing times of the BP are those reported by Côté et al. (2013)

On average, the MH reports a 1.56% of gap against the 2.00% obtained with the BP. Additionally, the average Gap(BP/MH) is -0.41% which means that the MH obtains better solution than the BP. Finally, although the experimental results were obtained on different machines, the MH improves the quality of the solutions reported by Côté et al. (2013) in a relatively low processing time.

Table 4.4: Solutions obtained by the matheuristic for the Lequy instances.

Instance (Id)	MH solution	Gap (%)	Time MH (s)	BP solution	Time BP (s)
Class 1: 1 day, 50 employees, and 10 activities.					
1808	3225	4.90	71.46	3270	3600.00
5066	2440	2.43	36.69	2440	935.00
5135	2580	0.54	19.12	2580	8.00
5226	2725	0.34	20.72	2725	8.00
8854	2740	3.46	100.63	2800	3600.00
Class 2: 1 day, 50 employees, and 10 activities.					
342	1875	1.60	8.02	1875	26.70
369	2315	0.66	12.48	2315	146.08
71	2050	0.00	4.83	2050	1.26
737	2065	2.82	9.73	2065	53.23
896	1890	3.88	10.20	1875	41.67
Class 3: 2 days, 75 employees, and 12 activities.					
1855	5960	4.27	5314.58	6265	3600.00
2106	6540	5.99	2632.76	6525	3600.00
2435	5850	3.74	1740.24	6050	3600.00
4225	6150	5.19	1682.04	6255	3600.00
9863	5870	3.31	868.20	5870	3600.00
Class 4: 7 days, 20 employees, and 5 activities.					
1024	7220	0.56	81.49	7220	11.00
1773	6345	0.01	86.14	6360	9.00
2732	7420	0.29	121.35	7420	19.00
4657	6400	1.34	78.36	6400	2003.00
5553	7535	0.16	106.70	7600	15.00
Class 5: 7 days, 20 employees, and 5 activities.					
1024	2940	0.00	15.73	2940	0.80
1773	2770	0.00	13.24	2770	0.76
2732	3820	0.00	9.90	3820	0.54
4657	3210	0.00	14.22	3210	0.61
5553	3270	0.00	10.88	3270	0.59
Class 6: 7 days, 50 employees, and 7 activities.					
5600	8440	0.64	200.03	8440	59.49
592	7345	0.27	235.63	7345	33.03
8597	7645	0.12	275.13	7645	31.58
9445	7900	0.06	169.51	7900	14.67
949	8155	0.33	275.07	8155	44.19
Class 7: 7 days, 50 employees, and 10 activities.					
1007	14085	1.37	1157.05	14115	1321.00
156	13420	0.79	1139.32	13420	1793.00
237	13455	0.93	875.391	13610	3600.00
4369	13630	1.95	1703.35	13675	1536.00
5216	14770	1.15	1171.37	14800	1824.00
Class 8: 7 days, 100 employees, and 15 activities.					
530	15155	1.48	2430.48	15200	5818.36
1024	15435	2.63	2790.49	15420	4602.99
2596	15765	1.76	2180.99	15855	10806.10
6384	15235	1.47	2100.24	15250	2064.81
7862	15880	1.98	2101.96	15940	1391.95

4.2.3 COLUMN GENERATION HEURISTIC

In the previous sections, the results obtained with the MH are compared with the results presented in Côté et al. (2013). However, the BP algorithm is not tested over the same machine given that the algorithm is not available to perform these experiments. For this reason, we implement and execute a column generation heuristic in order to measure the performance of the MH.

Table 4.5 shows a summary of the result obtained with the CGH. The column $\text{Gap}(\text{CGH}/\text{CG})$ presents the average gap between the feasible solution obtained with the CGH and the lower bound provided by the column generation; The column $\text{Gap}(\text{MH}/\text{CGH})$ corresponds to the average between the bound of the MH and the bound of the CGH; The column $\text{NbS}(\text{MH} \geq \text{CGH})$ shows the number of instances where the bound of the MH is lesser or equal to the bound of the CGH, and the last two columns display the average processing time of the MH and CGH, respectively. One can note that the MH is able to find better or equivalent than the CGH in 98 out of 100 instances. Furthermore, the average $\text{Gap}(\text{MH}/\text{CGH})$ is -1.8% which means that the MH obtains better solutions than the CGH, and the bounds of MH are 1.8% better than the CGH bounds. Finally, the MH finds better solution than the CGH in the 98% of the instances, with an improvement of 81.4% in the average processing time of the CGH.

Table 4.5: Comparison between MH and CGH for the Demassey instances.

Number of Activities	$\text{Gap}(\text{CGH}/\text{CG})$ (%)	$\text{Gap}(\text{MH}/\text{CGH})$ (%)	$\text{NbS}(\text{MH} \leq \text{CGH})$	Time MH(s)	Time CGH(s)
1	2.06	-1.84	9	12.01	1.06
2	6.19	-4.32	9	75.64	5.83
3	2.24	-2.11	10	97.51	16.09
4	1.37	-1.15	10	180.35	131.55
5	0.76	-0.74	10	35.29	158.62
6	1.31	-1.17	10	95.91	617.7
7	1.94	-1.72	10	131.64	919.97
8	3.08	-2.32	10	113.77	1163.15
9	0.74	-0.73	10	54.61	827.03
10	2.25	-1.94	10	172.14	1369.66
Average:	2.19	-1.8	9.8	96.89	521.07

Table 4.6 reports a summary of the results obtained with the CGH for the

Lequy instances. The results show that the MH is able to find better or equivalent solutions for 39 out of 40 instances, and obtains better bounds than the CGH in 3.73% of the instances. Although the CGH reports, on average, a better processing time (727.89s), the difference with the MH (796.89s) is not significant. Furthermore, in the Class 8 conformed with the largest instances, the MH reports a better processing times than the CGH. Also, both algorithms present a similar behavior (considering the $\text{Gap}(\text{MH}/\text{CGH})$) in all the classes except Class 8, where the MH reports bounds 26.26% better than the ones obtained with the CGH. This behavior can be due to the size of the instances; given that planning horizon is over one week with 100 employees and 15 activities, the number of feasible shift increase considerably.

Table 4.6: Comparison between MH and CGH for the Lequy instances.

Instances	$\text{Gap}(\text{CGH}/\text{CG})$	$\text{Gap}(\text{MH}/\text{CGH})$	NbS ($\text{MH} \leq \text{CGH}$)	Time	
	(%)	(%)		MH (s)	CGH (s)
Class 1	2.99	-0.97	5	49.72	112.78
Class 2	1.83	-0.25	4	9.06	8.81
Class 3	4.11	-1.88	5	2447.56	1558.66
Class 4	0.47	-0.08	5	94.81	15.97
Class 5	0.00	0.00	5	12.79	2.78
Class 6	0.20	-0.07	5	231.07	50.06
Class 7	0.65	-0.34	5	1209.29	1055.64
Class 8	87.12	-26.26	5	2320.83	3018.39
Average:	12.17	-3.73	4.87	796.89	727.89

4.3 CONCLUSIONS

In this chapter, we have presented a matheuristic based on Lagrangian relaxation, where the subgradient method was used to generate promising shifts and the best combination of the shifts was found by solving the $\text{SSCM}(\Omega)$. The subgradient method takes advantage of the context-free grammar representation to generate the shifts. Additionally, the shifts generated with the subgradient method allow us to reduce the number of variables and the complexity of the problem.

The solutions for the Demassey instances show that the MH obtains better solutions than the BP and the CGH, in a relatively low processing time. The MH finds solution within 0.01% of optimality gap for 70% of the instances. Furthermore, the bounds obtained with the MH are, on average, 1.8% better than the ones obtained with the CGH, also the MH reports better solution than the CGH in the 98% of the instances, improving the average processing time of the CGH in 81.4%.

For the Lequy instances, the MH obtains solutions within a 1.56% of optimality gap with respect to the lower bound provided by the Lagrangian Relaxation. Additionally, the MH finds better or equivalent solution than the BP for 38 of the 40 instances. Besides, the MH has better behavior than the CGH in the largest instances, where the MH reports bounds 26.26% better than the CGH, improving the average processing time of the CGH in 23.1%.

Finally, the MH obtains better or equivalent solutions than the CGH for the 97.8% of the instances, which highlights the efficiency of the MH.

THE IMAGE SEGMENTATION PROBLEM

In this chapter we introduce a bi-objective set-covering model for the ISP. Furthermore, we propose to solve the bi-objective model using the AUGMECON algorithm (Mavrotas, 2009). Finally, we present a detailed analysis of the results obtained with the model.

5.1 PROBLEM DESCRIPTION AND FORMULATION

The ISP consists of dividing a given image into segments with similar characteristics. Given that the model presented in the Section 3.2.1, it is impractical to solve or even to implement, we propose to use the histogram of the image to reduce the number of variables of the model. In this case, the problem can be seen as the problem of partitioning the histogram. Considering that we must cluster only pixels with similar colors, then the partition of the histogram has to be performed with sets of consecutive intensities. The objectives of this problem are to minimize the total number of sets considered in the partition and to minimize the maximum variance of the sets in the partition.

Moreover, if all the possible partition sets are generated, then the problem becomes a bi-objective set-covering problem which consists of finding the minimum

number of sets that cover the histogram, minimizing the maximum variance of the sets in the partition. In this thesis, we consider that the minimum set size is two and the maximum is $\lceil N/2 \rceil$, where N is the number of intensities (colors) in the image. We consider the minimum set size as two, because it has no sense to have segments with only one element.

We use the next parameters, to formulate the bi-objective set-covering model (BSCM):

- i is the index of the color in the histogram ($i \in \{1, \dots, N\}$);
- j is the size of a set ($j \in \{2, \dots, \lceil N/2 \rceil\}$);
- $s_{ij} = \{i, \dots, i + j - 1\}$ is the set of size j starting from color index i ;
- σ_{ij} is the variance of set s_{ij} ;
- δ_{ij}^k is a parameter equal to 1 if $k \in s_{ij}$, and 0 otherwise.

Using the previous notation, the decision variables are defined as follows:

$$x_{ij} = \begin{cases} 1, & \text{if } s_{ij} \text{ is a partition of the histogram;} \\ 0, & \text{otherwise.} \end{cases}$$

Consequently, the proposed BSCM for the image segmentation is formulated as follows:

$$\text{(BSCM) Min } z_1 = \max_{\substack{i=1, \dots, N \\ j=1, \dots, \lceil N/2 \rceil}} (\sigma_{ij}^2 x_{ij}) \quad (5.1)$$

$$\text{Min } z_2 = \sum_{i=1}^N \sum_{j=1}^{\lceil N/2 \rceil} x_{ij} \quad (5.2)$$

subject to

$$\sum_{i=1}^N \sum_{j=1}^{\lceil N/2 \rceil} \delta_{ij}^k x_{ij} = 1 \quad \forall k = 1, \dots, N \quad (5.3)$$

$$x_{ij} \in \{0, 1\} \quad \forall i = 1, \dots, N, j = 1, \dots, \lceil N/2 \rceil \quad (5.4)$$

The objective (5.1) minimizes the maximum variance of the sets in the partition, whilst the objective (5.2) minimizes the number of sets in the partition. Constraints (5.3) ensure that only one set covers each intensity in the histogram. Note that the BSCM model is not linear because of objective (5.1); however BSCM can be linearized as follows:

$$\text{(LBSCM) Min } z_1 = y \quad (5.5)$$

$$\text{Min } z_2 = \sum_{i=1}^N \sum_{j=1}^{\lceil N/2 \rceil} x_{ij} \quad (5.6)$$

subject to

$$\sum_{i=1}^N \sum_{j=1}^{\lceil N/2 \rceil} \delta_{ij}^k x_{ij} = 1 \quad \forall k = 1, \dots, N \quad (5.7)$$

$$\sigma_{ij}^2 x_{ij} \leq y \quad \forall i = 1, \dots, N, j = 1, \dots, \lceil N/2 \rceil \quad (5.8)$$

$$x_{ij} \in \{0, 1\} \quad \forall i = 1, \dots, N, j = 1, \dots, \lceil N/2 \rceil \quad (5.9)$$

$$y \in \mathbb{R}^+ \quad (5.10)$$

where y is a positive real variable and constraints (5.8) are the classical min-max linearization constraints.

Furthermore, if we consider an 8-bit image, which contains 256 intensities, the LBSCM model has 24,384 variables, the same number of linearization constraints, and 256 covering constraints. It is relevant to remark that the size of LBSCM depends only in the pixel format of the image; which means that if we consider two images of different size with the same pixel format, then the number of variables and constraints does not change.

Note that the function z_1 in model BSCM represents a heterogeneity measure of the elements in each set, and can be replaced by other measure. For example,

$$\text{Min } z_3 = \max_{\substack{i=1,\dots,N \\ j=1,\dots,\lceil N/2 \rceil}} (\sigma_{ij}^2 x_{ij})(\mu_\sigma - \sigma^2 x_{ij}), \quad (5.11)$$

where, μ_σ is the average of the variances over all the sets. This new function can be linearized using the same technique as in z_1 .

Furthermore, in this work two types of histograms are considered: the original histogram and the one with logarithmic frequencies. In fact, some values of the histogram are very large which can cause numerical difficulties when the problem is solved in a computer. The logarithmic frequencies allows to tackle these problems; moreover, the new scale could identify different segments, particularly when the peaks are pronounced. Table 5.1 shows the different variants of the LBSM discussed in the following sections.

Table 5.1: Alternative Models

Model	Objective Function	Frequencies
Sigma.freq	$z_1 = \max_{i,j} (\sigma_{ij}^2 x_{ij})$	Original
Sigma.logfreq	$z_1 = \max_{i,j} (\sigma_{ij}^2 x_{ij})$	Logarithmic
Media.freq	$z_3 = \max_{i,j} (\mu_\sigma - \sigma_{ij}^2 x_{ij})$	Original
Media.logfreq	$z_3 = \max_{i,j} (\mu_\sigma - \sigma_{ij}^2 x_{ij})$	Logarithmic

Notice that the models generated from the variants showed in Table 5.1 differ only in the objective function and the scale of the data.

5.2 AUGMECON

We propose to solve LBSCM using an AUGMECON algorithm, introduced by Mavrotas (2009), which is an iterative method used to generate the Pareto frontier for a multiobjective problem. The AUGMECON is a variant of ε -constraint method, the difference between the two methods is that ε -constraint optimize one of the objective function and uses the other objective functions as constraints; the AUGMECON

uses the same formulation as ε -constraint and adds, in the objective function, the excess or slack variable of each constraint related to the objective functions, in order to push their value as much as possible.

In our case, the algorithm solves a mono objective model at each iteration of the algorithm by using one of the objectives as a constraint of the model, and penalizing this change in the objective function. The mono objective version of the LBSCM is obtained by limiting the maximum number of segments G in the partition, and is formulated as follows:

$$(\text{TLBSCM}(G)) \text{ Min } z = y - \frac{\epsilon}{N} S \quad (5.12)$$

$$\begin{aligned} & \text{subject to} \\ & \sum_{i=1}^N \sum_{j=1}^{\lceil N/2 \rceil} x_{ij} + S = G \end{aligned} \quad (5.13)$$

$$\sum_{i=1}^N \sum_{j=1}^{\lceil N/2 \rceil} \delta_{ij}^k x_{ij} = 1 \quad \forall k = 1, \dots, N \quad (5.14)$$

$$\sigma_{ij}^2 x_{ij} \leq y \quad \forall i = 1, \dots, N, j = 1, \dots, \lceil N/2 \rceil \quad (5.15)$$

$$x_{ij} \in \{0, 1\} \quad \forall i = 1, \dots, N, j = 1, \dots, \lceil N/2 \rceil \quad (5.16)$$

$$y \in \mathbb{R}^+ \quad (5.17)$$

$$S \in \mathbb{R}^+ \quad (5.18)$$

where S is a slack variable for the constraint (5.13) and ϵ is a small value which helps to minimize the objective z_2 of the LBSCM, while z_1 is minimized. Note that the objective function in TLBSCM will maximize variable S subject to constraint (5.13). Consequently, the number of groups in the solution will be push down as much as possible.

Algorithm 4 presents the AUGMECON proposed to solve LBSCM. Function Solve() obtains the exact solution Sol for the TLBSCM(i); function Add() incorporate the solution Sol to the Pareto frontier, and the function NumberOfSegments() obtains the number of groups in the solution Sol . In this case, we fix G with the

value $\lceil N/2 \rceil$ at the first iteration, given that this is the maximum number of groups in a solution considering that the minimum size of a set is two. Notice that the Pareto frontier is conformed by all the possible segmentations.

Algorithm 4 Solution method for the (LBSCM)

Input: Histogram Data

Output: Pareto Frontier PF

- 1: $PF \leftarrow \emptyset$
 - 2: $G \leftarrow \lceil N/2 \rceil$
 - 3: **while** $G > 1$ **do**
 - 4: $Sol \leftarrow \text{Solve}(\text{TLBSCM}(G))$
 - 5: $PF \leftarrow \text{Add}(Sol)$
 - 6: $GR \leftarrow \text{NumberOfSegments}(Sol)$
 - 7: $G \leftarrow GR - 1$
 - 8: **end while**
 - 9: **return** PF
-

5.3 EXPERIMENTAL RESULTS

In this section, we present the performance of the AUGMECON approach over the four variants of the LBSCM showed in Table 5.1. This algorithm is tested using two datasets: the Weizmann benchmark conformed by 100 gray scale images of one object, and the high definition dataset introduced in this work. The details of the testing sets can be found in the next sections. In both cases, the objective is to find the segment which better fits to the object in each image.

The datasets contain only grayscale images with an 8-bit pixel format, which means that the size of the histogram has $N = 256$ intensities. Before solving $\text{TLBSCM}(G)$, a preprocessing of the image must be executed to compute all the possible partitions and the variance of each set in each partition. Besides, parameter ϵ in the

AUGMECON approach is fixed to 0.0001 for convention.

Finally, the algorithm is coded in C++ and the TLBSCM(G) is solved using CPLEX 12.3. Additionally, the stopping criteria are a relative optimality gap less than 1% and one hour of maximum processing time per iteration. The experiments were performed on a machine with an Intel Xeon 3.1 GHz and 64 GB of RAM. The statistical analysis presented in the next section was accomplished using the statistical software R.

In order to measure the quality of each segmentation generated from the algorithm, we use the function F (also named F-measure) computed as follows:

$$F = 2 \frac{PR}{P + R}$$

where $P = \frac{|PO \cap PS|}{|PS|}$ and $R = \frac{|PO \cap PS|}{|PO|}$ respectively. PO is the set of pixels that conforms the object in the image, and PS is the set of pixels contained in the segment related with the object. Furthermore, the set PO is indicated in the ground truth image (generated by a human). A good segmentation implies that the F-measure is close or equal to one.

5.3.1 HIGH DEFINITION IMAGES

We introduce a new set of high definition images of one object. Additionally, we present the experimental results of the AUGMECON. The new dataset is composed by 4 classes of 16 grayscale images taken from one object, where each class is represented by the size of the images: 2.4, 8, 12 and 16 mega-pixels (MP). Also, the ground-truth of each image is provided.

Furthermore, we carried out a statistical analysis in order to identify the best variant of the model (see table 5.1). Figures 5.1 and 5.2 show some examples of the segmentation performed by the AUGMECON algorithm.

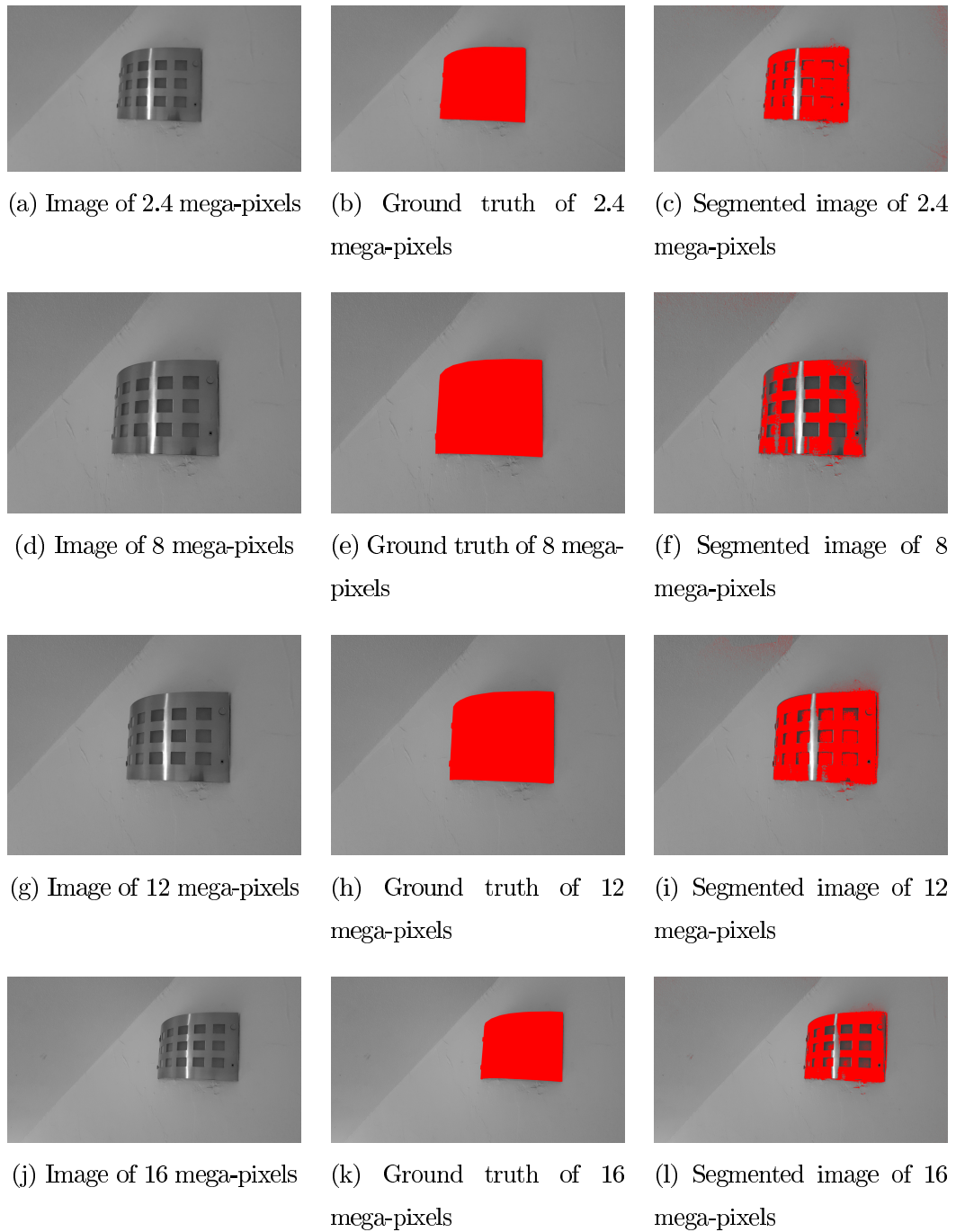


Figure 5.1: Experimental results for the first example of the high definition images

5.3.1.1 THE PARETO FRONTIER

In this section, we present the results obtained with the AUGMECON, which returns the Pareto frontier, that is conformed by a set of solutions that generate the possible segmentations of an image. Notice that the first efficient solution obtained with the AUGMECON provides the minimum value for the heterogeneity measure with the maximum number of segment. Consequently, the next solutions lead to a lower number of segments but with an increase of the heterogeneity value. In order to evaluate the quality of the segmentations generated from the Pareto frontier, we compute the F-measure for each image in the dataset.

In practice, we have noted that the algorithm never reach the maximum processing time, and discards a lot of possible number of segments at the first iteration, because the slack variable added in the objective function of the TLBSCM is pushing the number of groups as much as possible. Furthermore, if only the first solution is taken, then the algorithm can be considered as an automatic procedure; i.e. the algorithm is able to determine the number of segments and the segmentation. However, the algorithm provides a set of possible segmentations that gives to the decision maker the possibility of choosing the best segmentation for him/her among all the segmentations.

Tables 5.2, 5.3, 5.4 and 5.5 summarize the Pareto frontiers obtained with each variant of the LBSCM for each class in the high definition dataset. The first two columns corresponds to the average of the number of segments and the F-measure for each image size with the first solution; the next three columns present the average of the number of iteration, the number of segments and the F-measure of the best segmentation obtained with the frontier; the following two columns show the

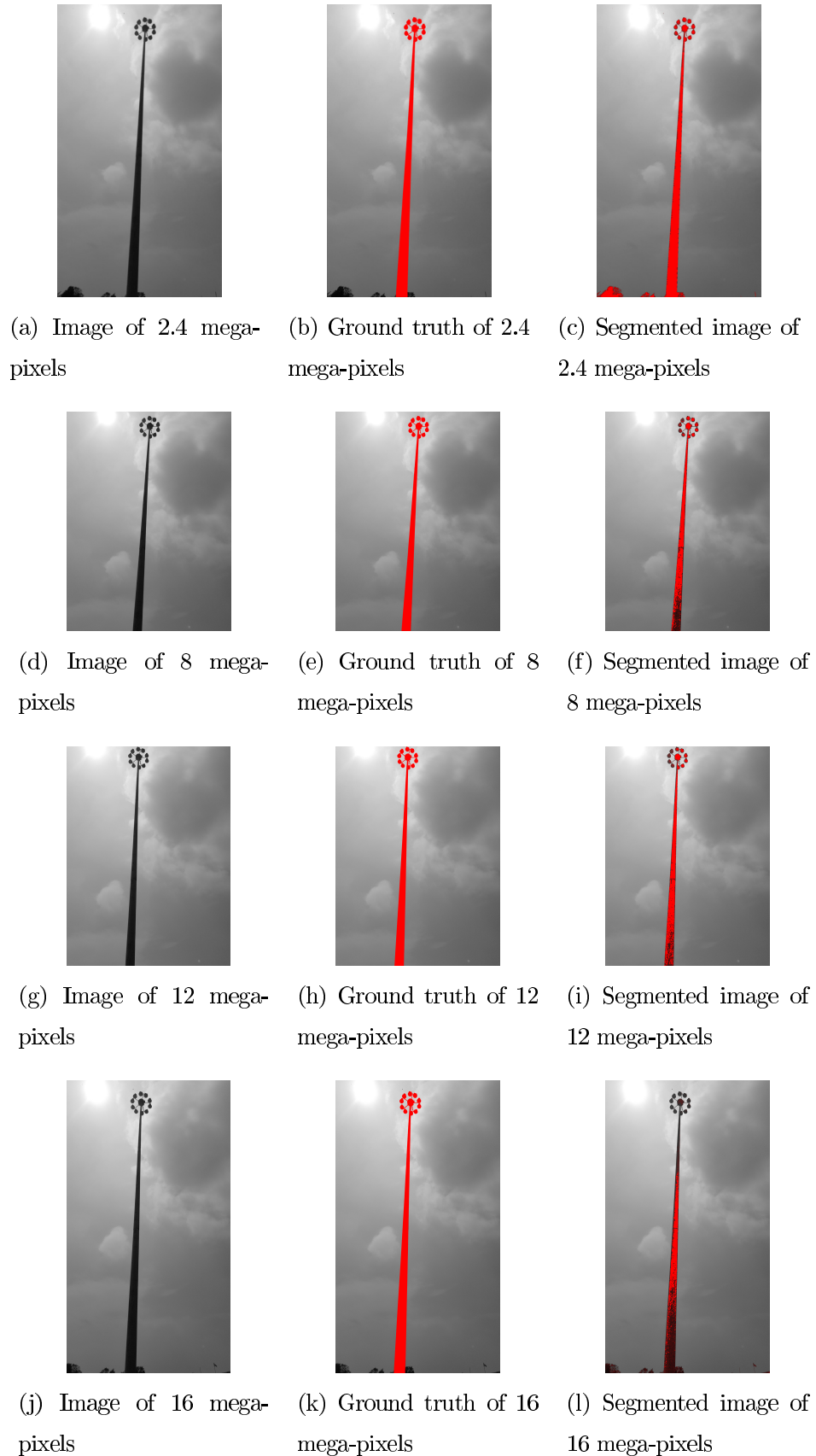


Figure 5.2: Experimental results for the second example of the high definition images

average F-measure of the segmentations generated from Pareto frontier and its respective standard deviation. Finally, the column Gap F(Best-First)% displays the gap between the average F-measure of the best and the first solution.

Table 5.2: Summary of the Pareto frontiers with Sigma.freq

First Solution			Best Solution		Pareto Frontier			Gap
Segments	F-measure	Iteration	Segments	F-measure	F-Avg	Std-Dev	F(Best-First)	
							%	
2.4 M	13.50	0.43	6.88	7.50	0.50	0.43	0.06	11.27
8 M	12.25	0.47	8.00	5.19	0.53	0.47	0.06	12.43
12 M	10.50	0.47	6.31	5.13	0.54	0.46	0.07	12.01
16 M	12.63	0.43	6.94	6.69	0.50	0.42	0.06	15.10
Average:	12.22	0.45	7.03	6.13	0.51	0.45	0.06	12.70

Table 5.3: Summary of the Pareto frontiers with Sigma.logfreq

First Solution			Best Solution		Pareto Frontier			Gap
Segments	F-measure	Iteration	Segments	F-measure	F-Avg	Std-Dev	F(Best-First)	
							%	
2.4 M	12.38	0.46	7.31	6.00	0.51	0.44	0.07	10.59
8 M	12.50	0.48	8.56	4.94	0.53	0.47	0.06	10.57
12 M	10.56	0.48	5.75	5.81	0.53	0.46	0.07	9.54
16 M	12.81	0.44	7.44	6.38	0.50	0.43	0.06	9.97
Average:	12.06	0.46	7.27	5.78	0.52	0.45	0.07	10.17

Table 5.4: Summary of the Pareto frontiers with Media.freq

	First Solution			Best Solution		Pareto Frontier		
	Segments	F-measure	Iteration	Segments	F-measure	F-Avg	Std-Dev	Gap
								F(Best-First) %
2.4 M	3.56	0.41	1.44	3.13	0.47	0.41	0.07	6.64
8 M	3.19	0.44	1.44	2.75	0.48	0.43	0.05	4.59
12 M	3.06	0.45	1.44	2.63	0.46	0.44	0.02	1.41
16 M	3.25	0.41	1.56	2.69	0.44	0.40	0.05	4.25
Average:	3.27	0.43	1.47	2.80	0.46	0.42	0.05	4.22

Table 5.5: Summary of the Pareto frontiers with Media.logfreq

	First Solution			Best Solution		Pareto Frontier		
	Segments	F-measure	Iteration	Segments	F-measure	F-Avg	Std-Dev	Gap
								F(Best-First) %
2.4 M	4.44	0.43	2.19	3.25	0.47	0.42	0.07	7.17
8 M	4.50	0.47	2.44	3.06	0.51	0.45	0.08	6.64
12 M	4.50	0.45	2.13	3.38	0.51	0.45	0.07	9.38
16 M	4.50	0.43	2.06	3.44	0.48	0.41	0.09	8.24
Average:	4.48	0.45	2.20	3.28	0.49	0.43	0.08	7.86

The results show that the first segmentation generated has not always the best F-measure evaluation, although this solution has the minimum heterogeneity value. Actually, the average gap between the F-measure from the first and the best solution is between 4.22% and 12.70%. However the method can take hours to obtain the best solution because it explores the complete frontier first. Also, we cannot estimate the quality of a segmentation a priori, because it is no possible to obtain information about the F-measure of a solution without computing it.

From the results, the best performance is reported by Sigma.logfreq model, which can achieve an average F-measure of 0.52 with the best solution in the frontier, and 0.46 with the first solution. Considering that the algorithm can take hours to obtain the best solution, and given that the gap between the F-measures of the first and the best solution is relatively low, it make sense to consider only the first solution in order to obtain an automatic segmentation and reduce the processing

times of the algorithm. In the next section, we present the computational result of the algorithm considering only the first solution of the AUGMECON. Additionally, we show a detailed statistical analysis of the results.

5.3.1.2 COMPARISON OF THE MODELS

In this section, we summarize the results obtained by choosing the first solution returned by the AUGMECON. Table 5.6 and figure 5.3 show the average measure obtained with each variant of the model tested over each image size set. The results suggest that the quality of the segmentation has no relation with the size of the image, given that the F-measure is maintained stable with respect of each variant of the model. Actually, the average F-measure is around 0.4 in all cases, and the variation is not greater than 0.03 in the average F-measures over each model.

Furthermore, the average processing times are presented in table 5.7 and figure 5.4. In general, the results does not show an explicit relation between the average processing times and the size of the images; Although the processing times with the Media.freq shows that the greater the image size, the greater the processing times. The behaviors of the results suggest that the difference observed in the processing time seems related to the convergence of CPLEX. Additionally, the models Sigma.logfreq, Media.freq, and Media.logfreq obtains solution within a minute of processing time.

Table 5.6: Average F-measure

Size	Sigma.freq	Sigma.logfreq	Media.freq	Media.logfreq
	F-measure	F-measure	F-measure	F-measure
2.4 M	0.39	0.45	0.40	0.43
8 M	0.37	0.47	0.42	0.46
12 M	0.38	0.47	0.41	0.45
16 M	0.35	0.44	0.42	0.42

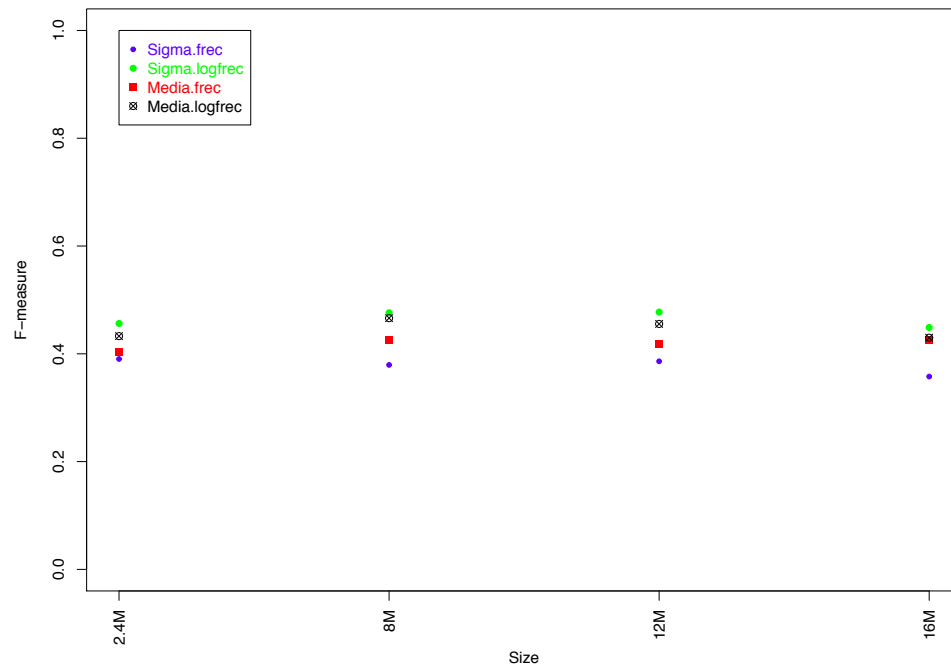


Figure 5.3: Average F-measure

Table 5.7: Average time

Size	Sigma.freq Time (S)	Sigma.logfreq Time (S)	Media.freq Time (S)	Media.logfreq Time (S)
2.4 M	162.78	50.74	32.59	29.02
8 M	182.48	71.50	39.87	43.53
12 M	164.80	44.41	48.66	39.99
16 M	192.98	66.01	59.11	67.24

Consequently, we suggest to use the Sigma.logfrec given that is the best evaluated model and it takes a relatively low processing time to obtain a solution. In the next sections, only the Sigma.logfrec is considered to perform a detailed experiment and a deeper statistical analysis of the results.

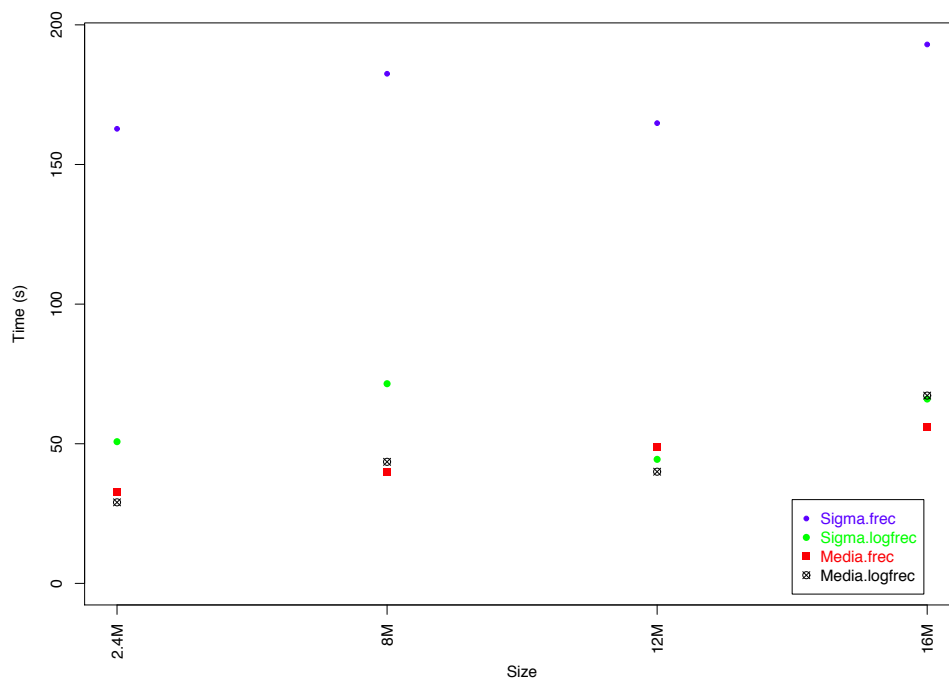


Figure 5.4: Average time

5.3.1.3 F-MEASURE ANALYSIS

In this section, we present a statistical analysis of differences of the F-measure data with respect to the size of the image. In this work, we propose to use an ANOVA (Fisher, 1925) to identify if there is a difference or not, in order to determine if the image size is a factor that impacts on the quality of the segmentation.

Before applying the ANOVA, we should corroborate that the data accomplish with its assumptions. First, the independence of the treatments is assured because each treatment is represented by the image size, which is totally different in each class; second, the homogeneity of the variance is confirmed with 95% of confidence by the Fligner-Killeen test presented in Table 5.8, given that the p-value in the table is greater to 0.05; finally, the normality assumption is validated with 95% of confidence by the Anderson-Darling test presented in Table 5.9, given that the p-value is greater to 0.05.

Table 5.8: Homogeneity of the variances test

Fligner-Killeen Test			
chi-squared	Degrees of freedom	p-value	Critical value
0.081264	3	0.994	7.81

Table 5.9: Normality test

Anderson-Darling Test			
Size	A-value	p-value	Critical value
2.4M	0.55435	0.1276	2.92
8M	0.29086	0.5633	
12M	0.43949	0.2555	
16M	0.33393	0.4666	

The ANOVA test for the difference in the F-measure is presented in Table 5.10. The analysis shows that there is not a significant difference in the values among the image sizes with a confidence level of 95%; given that the $Pr(> F)$ value is greater to 0.05 and the F-value (0.039) is lesser than the Critical Value (2.75). Therefore, the quality of the solution does not depend of the size of the image. Figure 5.5 presents a boxplot of the F-measure results to support the analysis.

Table 5.10: ANOVA

	DF	Sum Sq	Mean Sq	F-value	Pr(>F)	Critical Value
Treatments	3	0.10	0.0032	0.039	0.99	2.75
Residuals	60	5.01	0.0836			

5.3.1.4 TIME ANALYSIS

The statistical analysis of differences in the processing times cannot be performed by using an ANOVA, because the samples does not comply with the normality assumption. For this reason, we propose to use the non parametric Kruskal-Wallis test (Kruskal and Wallis, 1952) to identify the differences in the processing times

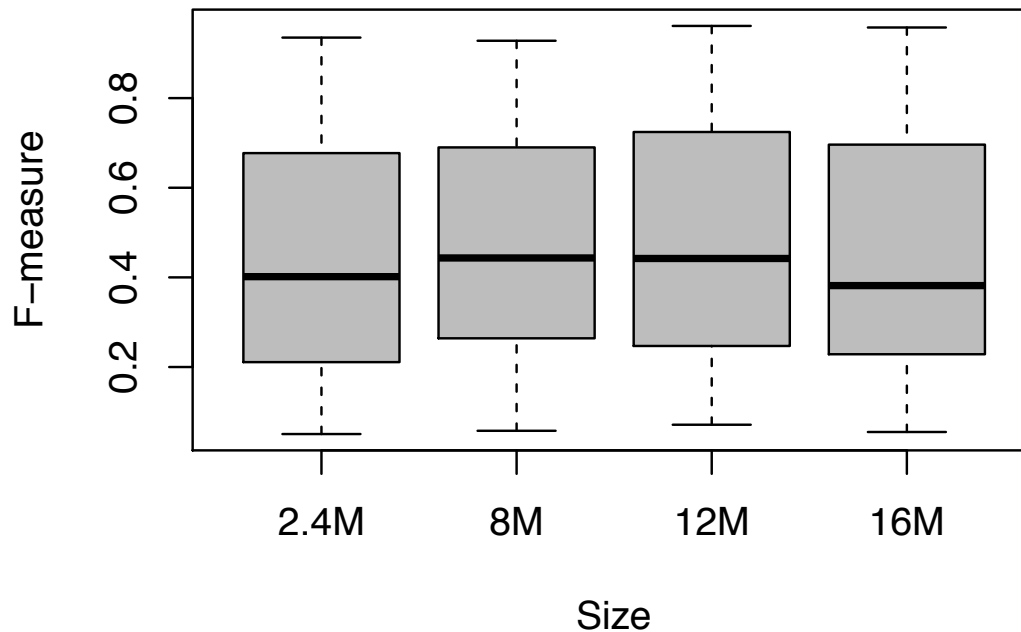


Figure 5.5: Distributions of measures

among the size of the images. Table 5.11 displays the results of the Kruskal-Wallis test and shows that there is a difference in the processing times with a confidence level of 95% given that the p-value is less to 0.05. Also, the value of the χ -squared (10.65) for this experiment is greater to the critical value (7.81) which corroborates the results of this test. This means that there is a difference in the processing times for, at least, two different image size, although the model maintains the same number of variables an constraint for all the images.

Table 5.11: Kruskal-Wallis test

Kruskal-Wallis Test			
chi-squared	Degrees of freedom	p-value	Critical value
10.652	3	0.01376	7.81

Consequently, the Nemenyi test is performed in order to identify the differences

in the preprocessing times. This test is presented in Table 5.12 and indicates that the difference in the processing times occurs, with 95% of confidence, in the comparison of the results of the 2.4M and 16M datasets; given that the respective p-value is less than 0.05. The results of the test suggest that there is a statistical difference in the processing times corresponding to the 2.4M and 16M datasets. Figure 5.6 shows that there is a small increase in the processing times, however this increment can be due to the convergence of CPLEX because the number of variables and constraints of the model does not change.

Furthermore, Figure 5.5 shows that the algorithm is more stable with the 12M set. Given that this case presents the smallest deviation in the processing times.

Table 5.12: Test of differences

Nemenyi Test		
Size	Differences	p-value
8M-2.4M	8.00	0.6171
12M-2.4M	10.50	0.3814
16M-2.4M	21.25	0.0068
12M-8M	2.50	0.9814
16M-8M	13.25	0.1832
16M-12M	10.75	0.3599

5.3.2 WEIZMANN IMAGES

In this section, we present the computational results of the algorithm tested over the Weizmann benchmark, which is conformed by 100 grayscale images of 8-bit with one object. Also, this dataset provides the ground truth for each image. The objective is to obtain the segment that better fits to the principal object. This experiment was executed considering only the first solution of the Sigma.logfreq, since this model have reported the best F-measure. Figure 5.7 shows some examples of the segmentations obtained with the first solution of the AUGMECON. Table 5.13 compares our score with the best scores reported in the literature for this benchmark.

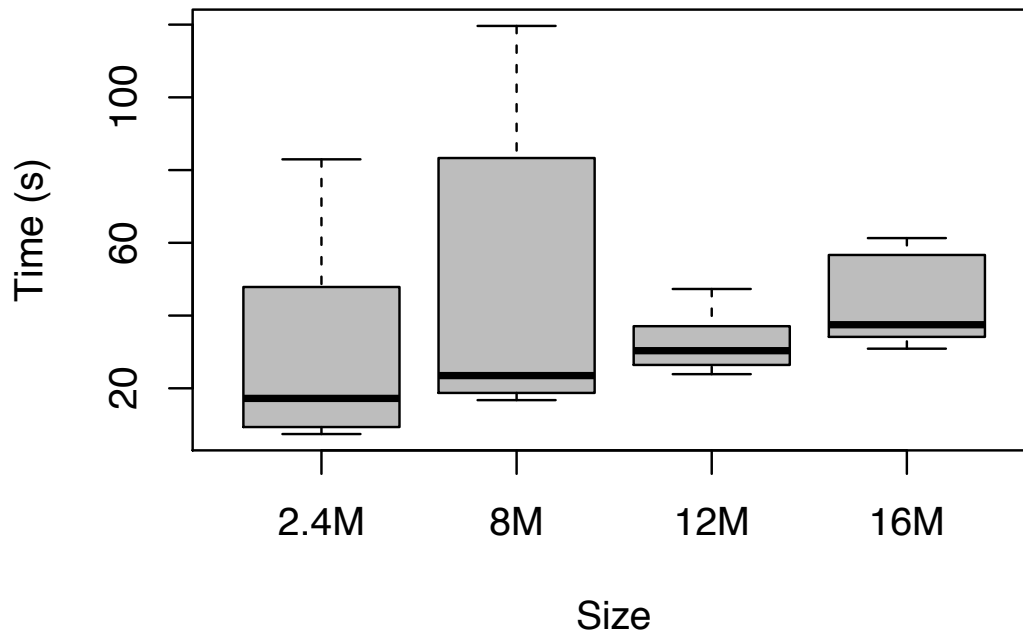


Figure 5.6: Distributions of processing times

Table 5.13 shows that the proposed approach achieves an average F-measure of 0.59 with 0.032 of variance. These results place our algorithm at the fifth position among the best scores reported. Furthermore, the algorithm provides a segmentation within 41.42 seconds of computational time. Also, we observe that 90% of the images were segmented in less than 100 seconds. These results are supported by Figure 5.8, which presents the cumulative distribution of the processing time.

5.4 CONCLUSIONS

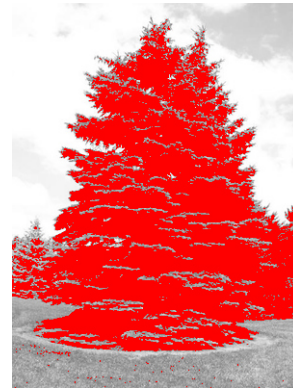
In this chapter, we have formulated a novel bi-objective set-covering model for the image segmentation based on the data of the histogram of a given image. Likewise, we have presented four variants of this model combined with two different



(a) Example 1



(b) Ground truth from Example 1



(c) Segmented image from Example 1



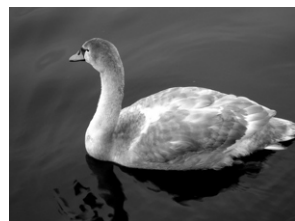
(d) Example 2



(e) Ground truth from Example 2



(f) Segmented image from Example 2



(g) Example 3



(h) Ground truth from Example 3



(i) Segmented image from Example 3



(j) Example 4



(k) Ground truth from Example 4



(l) Segmented image from Example 4

Figure 5.7: Examples of the experimental results of the Weizmann dataset

Table 5.13: Table of scores

Algorithm	Score	Remarks
Bagon et al. (2008)	0.87 ± 0.010	Interactive Segmentation
Alpert et al. (2012)	0.86 ± 0.012	Automatic Segmentation
Galun et al. (2003)	0.83 ± 0.016	Automatic Segmentation
Shi and Malik (2000)	0.72 ± 0.018	Automatic Segmentation
Sigma.logfreq	0.59 ± 0.032	Automatic Segmentation
Comaniciu and Meer (2002)	0.57 ± 0.023	Automatic Segmentation

heterogeneity measures and two scales for the frequencies of the histogram. The advantage of considering the data of the histogram is that the models does not depend on the images size, which conduced to a model with considerably less variables and constraints. Furthermore, we have proposed to use the AUGMECON approach to provide a set of efficient solutions to the BSCM, that gives the decision maker the possibility of choosing the best segmentation based on his/her experience. Notice that if the first solution of the Pareto frontier is taken, then the algorithm is able to determine the number of segments automatically in a relatively low computational time.

Additionally, we performed a detailed experimental work of the AUGMECON approach through the use of a new high definition dataset. The results show the robustness of the method, maintaining the stability in the quality of the solutions and in the processing times independently of the size of the input image. We have identified that the Sigma.logfrec model provides the best quality segmentations.

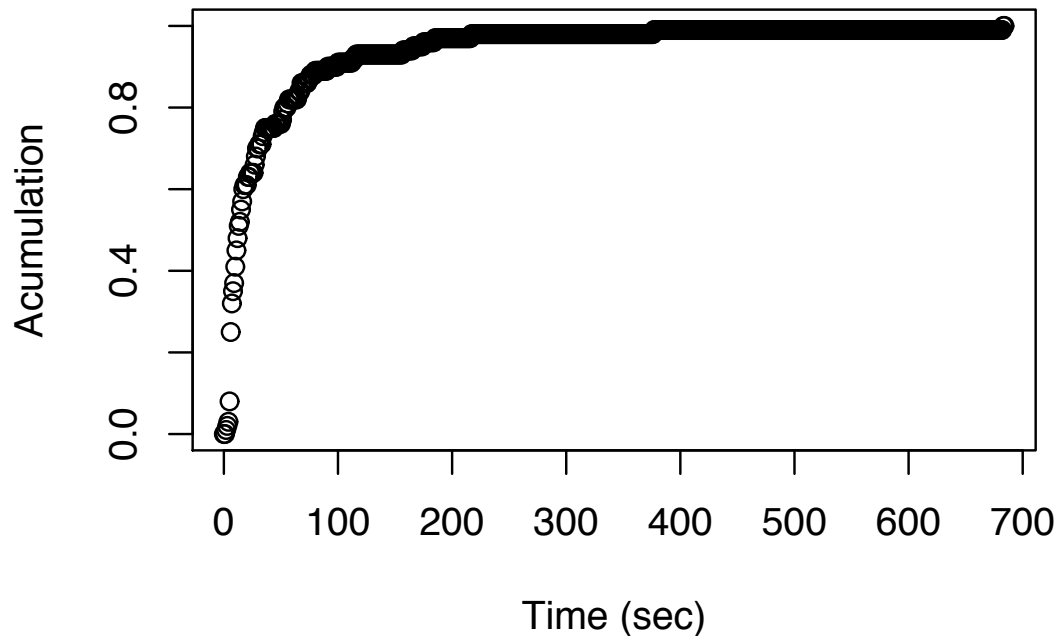


Figure 5.8: Cumulative distribution of the processing time

Likewise, the statistical analysis proves that the quality of the segmentation is not affected by the number of pixels in the image; although the analysis shows a slight increase in the processing times given a greater image size. This increment in the processing times can be due to the convergence of the solver (CPLEX).

Furthermore, the proposed algorithm was tested and compared considering the Weizmann dataset. Our algorithm with the Sigma.logfrec variant of the model was able to obtain an average F-measure of 0.59 with a variance of 0.032, which position our algorithm in the fifth place among the best results reported in the literature. Finally, the procedure take less than 50 seconds to find the first solution of the frontier for 80% of the images.

Finally, the proposed approach gives to the user the flexibility to provide differ-

ent heterogeneity measures and ways to define the partition sets, in order to improve the quality of the segmentation. Notice that this algorithm is relevant in the case of very large size images because it can perform an efficient segmentation within a relatively low processing time.

CHAPTER 6

CONCLUSIONS

In this thesis we have presented two applications for the set-covering problem: the multi-activity shift scheduling problem and the image segmentation. The SCP was very useful to model each problem and to derive efficient dedicated solution methods.

On one hand, the structure of SCP for the MASSP allowed us to divide this problem into smaller ones thanks to a Lagrangian relaxation. Also, we took advantage of the context-free grammar representation of the feasible shifts of each employee to facilitate the solution of the subproblems. In this context, we have proposed a matheuristic method which uses the set of the feasible shifts generated at each step of the subgradient method, in order to build a restricted set-covering model which is solved exactly to provide the best combination of the shifts. Note that the SSCM has considerably less variables than the SCM.

The MH for the MASSP was tested over two datasets provided in the literature: the Lequy instances and the Demassez instances. The experimental results show that the MH is able to find better or equivalent solution for 137 out of 240 instances compared with the column generation heuristic for both datasets. Furthermore, the MH is able to provide better or equivalent solutions for 38 of 40 instances of the Lequy dataset; this information cannot be obtained for the Demassez instances given that Côté et al. (2013) only show the summary of the results; however, from

table 4.2, it can be seen that the MH is able to find better quality solutions than the BP with an improvement of 95.7% of the BP average processing time. Finally, it is important to remark that the MH is able to obtain better quality solution in a shorter processing time than the CGH and the BP algorithms, specially for the biggest instances in both datasets.

On the other hand, we introduce a new bi-objective set-covering formulation for the image segmentation, which takes advantage of the histogram information in order to reduce the number of variables of the model. Besides, we propose four variants for the BSCM that differ on the heterogeneity measure and on the scale of the histogram. Also, we propose to use the first solution obtained with AUGMECON to obtain the number of groups automatically. This idea is justified for the relatively low difference between the first and the best solution in the frontier and the high processing time taken to obtain the best evaluated solution. Nevertheless, the AUGMECON algorithm provides a set of possible segmentation that gives to the decision maker the flexibility of choosing the best solution based in his/her own criteria.

The experimental results show that the model with a heterogeneity measure based on the variance of the segments with logarithmic scale has the best performance with respect to the F-measure. The statistical analysis shows that there is no significant difference in the measure with respect to the image size. However, the results show a statistical difference in times between the 2.4MP and 16MP images; although, we note that the difference is relatively low.

Furthermore, the proposed algorithm was tested and compared considering the Weizmann dataset. Our algorithm with the Sigma.logfrec variant of the model was able to obtain an average F-measure of 0.59 with a variance of 0.032, which position our algorithm in the fifth place among the best results reported in the literature. Finally, the procedure takes less than 50 seconds to find the first solution of the frontier for 80% of the cases.

Finally, the advantage of taking the first solution provided by the algorithm is that it is able to identify the number of segments of the image.

CHAPTER 7

FUTURE WORK

As a future work, we propose to reduce the processing times of the proposed metaheuristic for the MASSP. The exact solution of the model takes the most of the processing times; for this reason, we propose to use a heuristic method to solve the SSCM based on one of the several heuristics that can be found in the literature (see for instance Christofides and Korman (1975) and Caprara et al. (2000)). The reason of this idea is that, in the literature, there are several works that show that the set-covering problem can be solved efficiently using heuristic and metaheuristic algorithms.

Furthermore, the MASSP can be extended to the multi-task variant where a task is a mandatory activity which is not performed on a daily basis. Additionally, this variant presents precedence constraints and time windows for the tasks. For example, the machinery maintenance in a company is not performed daily and it is planned to be performed in certain periods of time.

For the images segmentation problem, it is planned to test this algorithm over medical images in order to make the detection of anomalies. Furthermore, the histogram partition can be executed using a heuristic approach in order to achieve better processing times. Also, we propose to design a fuzzy version of the BSCM in order to improve the F-measure evaluation; given that BSCM consider only the frequencies of each intensity in the image to make a partition, then the algorithm

is not able to group those pixels that conforms an object that have very different intensity. In this order, the fuzziness in the intensities can give to the BSCM the possibility of grouping these pixels.

BIBLIOGRAPHY

- Alpert, S., Galun, M., Brandt, A., and Basri, R. (2012). Image segmentation by probabilistic bottom-up aggregation and cue integration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(2):315–327.
- Bagon, S., Boiman, O., and Irani, M. (2008). What is a good image segment? a unified approach to segment extraction. In *Forsyth D., Torr P., Zisserman A. (eds) Computer Vision – ECCV 2008*, pages 30–44. Lecture Notes in Computer Science, vol 5305. Springer, Berlin, Heidelberg.
- Balas, E. and Carrera, M. (1996). A dynamic subgradient-based branch-and-bound procedure for the set-covering problem. *Operations Research*, 44(5):875–890.
- Balas, E. and Ho, A. (1980). Set covering algorithms using cutting planes, heuristics, and subgradient optimization: A computational study. In *Padberg M.W. (eds) Combinatorial Optimization. Mathematical Programming Studies*, volume 12, pages 37–60. Springer, Berlin, Heidelberg.
- Beasley, J. (1990). A lagrangian heuristic for set-covering problems. *Naval Research Logistics*, 37(1):151–164.
- Bhandari, A., Kumar, A., Chaudhary, S., and Singh, G. (2016). A novel color image multilevel thresholding based segmentation using nature inspired optimization algorithms. *Expert Systems with Applications*, 63:112–133.
- Bixby, R., Gregory, J., Lustig, I., Marsten, R., and Shanno, D. (1992). Very large-

- scale linear programming: A case study in combining interior point and simplex methods. *Operations Research*, 40(5):885–897.
- Boyer, V., Gendron, B., and Rousseau, L. (2014). A branch-and-price algorithm for the multi-activity multi-task shift scheduling problem. *Journal of Scheduling*, 17(2):185–197.
- Caprara, A., Fischetti, M., and Toth, P. (1999). A heuristic method for the set covering problem. *Operations Research*, 47(5):730–743.
- Caprara, A., Toth, P., and Fischetti, M. (2000). Algorithms for the set covering problem. *Annals of Operations Research*, 98(1-4):353–371.
- Caserta, M. (2007). Tabu search-based metaheuristic algorithm for large-scale set covering problems. In *Doerner K.F., Gendreau M., Greistorfer P., Gutjahr W., Hartl R.F., Reimann M. (eds) Metaheuristics. Operations Research/Computer Science Interfaces Series*, volume 39, pages 43–63. Springer, Boston, MA.
- Ceria, S., Nobili, P., and Sassano, A. (1998). A lagrangian-based heuristic for large-scale set covering problems. *Mathematical Programming*, 81(2):215–228.
- Christofides, N. and Korman, S. (1975). Note—a computational survey of methods for the set covering problem. *Management Science*, 21(5):591–599.
- Comaniciu, D. and Meer, P. (2002). Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619.
- Côté, M.-C., Gendron, B., Quimper, C.-G., and Rousseau, L.-M. (2011a). Formal languages for integer programming modeling of shift scheduling problems. *Constraints*, 16(1):54–76.
- Côté, M.-C., Gendron, B., and Rousseau, L.-M. (2011b). Grammar-based integer programming models for multiactivity shift scheduling. *Management Science*, 57(1):151–163.

- Côté, M.-C., Gendron, B., and Rousseau, L.-M. (2013). Grammar-based column generation for personalized multi-activity shift scheduling. *INFORMS Journal on Computing*, 25(3):461–474.
- Dahmen, S. and Rekik, M. (2012). Solving multiactivity personalized shift scheduling problems with a hybrid heuristic. Technical report, CIRRELT.
- Demasse, S., Pesant, G., and Rousseau, L. (2006). A cost-regular based hybrid column generation approach. *Constraints*, 11:315–333.
- Elahipanah, M., Desaulniers, G., and Lacasse-Guay, È. (2013). A two-phase mathematical-programming heuristic for flexible assignment of activities and tasks to work shifts. *Journal of Scheduling*, 16(5):443–460.
- Farahani, R., Asgari, N., Heidari, N., Hosseini, M., and Goh, M. (2012). Covering problems in facility location: A review. *Computers & Industrial Engineering*, 62(1):368–407.
- Fisher, R. A. (1925). *Statistical methods for research workers*. Genesis Publishing Pvt Ltd.
- Galun, M., Sharon, E., Basri, R., and Brandt, A. (2003). Texture segmentation by multiscale aggregation of filter responses and shape elements. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 716–723. IEEE.
- Garcia, V., Ochoa, H., and Mederos, B. (2015). Analysis of discrepancy metrics used in medical image segmentation. *Latin America Transactions, IEEE (Revista IEEE America Latina)*, 13(1):235–240.
- Gurcan, M., Boucheron, L., Can, A., Madabhushi, A., Rajpoot, N., and Yener, B. (2009). Histopathological image analysis: A review. *IEEE reviews in biomedical engineering*, 2:147–171.

- Havaei, M., Davy, A., Warde-Farley, D., Biard, A., Courville, A., Bengio, Y., Pal, C., Jodoin, P.-M., and Larochelle, H. (2015). Brain tumor segmentation with deep neural networks. *arXiv preprint arXiv:1505.03540*.
- He, L. and Huang, S. (2017). Modified firefly algorithm based multilevel thresholding for color image segmentation. *Neurocomputing*, 240:152–174.
- Heimann, T. and Meinzer, H. (2009). Statistical shape models for 3d medical image segmentation: a review. *Medical image analysis*, 13(4):543–563.
- Hopcroft, J., Motwanu, R., and Ullman, J. (2001). Introduction to automata theory, languages and computation. chapter 5, pages 169–216. Addison-Wesley.
- Karp, R. (1972). Reducibility among combinatorial problems. In *Complexity of computer computations*, pages 85–103. Springer.
- Kruskal, W. H. and Wallis, W. A. (1952). Use of ranks in one-criterion variance analysis. *Journal of the American statistical Association*, 47(260):583–621.
- Lequy, Q., Bouchard, M., Desaulniers, G., Soumis, F., and Tachefine, B. (2012a). Assigning multiple activities to work shifts. *Journal of Scheduling*, 15(2):239–251.
- Lequy, Q., Desaulniers, G., and Solomon, M. (2012b). A two-stage heuristic for multi-activity and task assignment to work shifts. *Computers & Industrial Engineering*, 63(4):831–841.
- Liu, Z., Li, X., Luo, P., Loy, C.-C., and Tang, X. (2015). Semantic image segmentation via deep parsing network. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1377–1385.
- Ma, Z., Tavares, J., Jorge, R., and Mascarenhas, T. (2010). A review of algorithms for medical image segmentation and their applications to the female pelvic cavity. *Computer Methods in Biomechanics and Biomedical Engineering*, 13(2):235–246.

- Mavrotas, G. (2009). Effective implementation of the ε -constraint method in multi-objective mathematical programming problems. *Applied Mathematics and Computation*, 213(2):455–465.
- Menze, B., Jakab, A., Bauer, S., Kalpathy-Cramer, J., Farahani, K., Kirby, J., Burren, Y., Porz, N., Slotboom, J., Wiest, R., et al. (2015). The multimodal brain tumor image segmentation benchmark (BRATS). *IEEE Transactions on Medical Imaging*, 34(10):1993–2024.
- Ng, H., Ong, S., Foong, K., Goh, P., and Nowinski, W. (2006). Medical image segmentation using k-means clustering and improved watershed algorithm. In *2006 IEEE Southwest Symposium on Image Analysis and Interpretation*, pages 61–65. IEEE.
- Noble, J. and Boukerroui, D. (2006). Ultrasound image segmentation: a survey. *IEEE Transactions on medical imaging*, 25(8):987–1010.
- Pal, N. and Pal, S. (1993). A review on image segmentation techniques. *Pattern recognition*, 26(9):1277–1294.
- Pham, D., Xu, C., and Prince, J. (2000). Current methods in medical image segmentation. *Annual review of biomedical engineering*, 2(1):315–337.
- Piqueras, S., Krafft, C., Beleites, C., Egodage, K., von Eggeling, F., Guntinas-Lichius, O., Popp, J., Tauler, R., and de Juan, A. (2015). Combining multiscale resolution and segmentation for hyperspectral image analysis of biological tissues. *Analytica chimica acta*, 881:24–36.
- Pont-Tuset, J., Arbelaez, P., Barron, J., Marques, F., and Malik, J. (2017). Multi-scale combinatorial grouping for image segmentation and object proposal generation. *IEEE transactions on pattern analysis and machine intelligence*, 39(1):128–140.
- Quimper, C. and Rousseau, L. (2010). A large neighbourhood search approach to the multi-activity shift scheduling problem. *Journal of Heuristics*, 16:373–392.

- Rajinikanth, V. and Couceiro, M. (2015). Rgb histogram based color image segmentation using firefly algorithm. *Procedia Computer Science*, 46:1449–1457.
- Restrepo, M., Lozano, L., and Medaglia, A. (2012). Constrained network-based column generation for the multi-activity shift scheduling problem. *International Journal of Production Economics*, 140(1):466–472.
- Sammouda, R., Aboalsamh, H., and Saeed, F. (2015). Comparison between k mean and fuzzy c-mean methods for segmentation of near infrared fluorescent image for diagnosing prostate cancer. In *Computer Vision and Image Analysis Applications (ICCVIA), 2015 International Conference on*, pages 1–6. IEEE.
- Shi, J. and Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence*, 22(8):888–905.
- Shor, N. (1985). Minimization methods for non-differentiable functions. volume 3 of *Springer Series in Computational Mathematics*, pages 22–47. Springer Berlin Heidelberg.
- Sipser, M. (2006). Introduction to the theory of computation. Course Technology.
- Tan, K. and Isa, N. (2011). Color image segmentation using histogram thresholding-fuzzy c-means hybrid approach. *Pattern Recognition*, 44(1):1–15.
- Tobias, O. and Seara, R. (2002). Image segmentation by histogram thresholding using fuzzy sets. *IEEE transactions on Image Processing*, 11(12):1457–1465.
- Torbati, N., Ayatollahi, A., and Kermani, A. (2014). An efficient neural network based method for medical image segmentation. *Computers in biology and medicine*, 44:76–87.
- Udupa, J., Leblanc, V., Zhuge, Y., Imielinska, C., Schmidt, H., Currie, L., Hirsch, B., and Woodburn, J. (2006). A framework for evaluating image segmentation algorithms. *Computerized Medical Imaging and Graphics*, 30(2):75–87.

- Umetani, S. and Yagiura, M. (2007). Relaxation heuristics for the set covering problem. *Journal of the Operations Research*, 50:350–375.
- Vemuganti, R. (1998). Applications of set covering, set packing and set partitioning models: A survey. In *Handbook of combinatorial optimization*, pages 573–746. Springer.
- Wei, B. and Mandava, R. (2010). Multiobjective optimization approaches in image segmentation-the direction and challenges. *ICSRs Publication*, 2:41–65.
- Yagiura, M., Kishida, M., and Ibaraki, T. (2006). A 3-flip neighborhood local search for the set covering problem. *European Journal of Operational Research*, 172:472–499.
- Zhang, W., Li, R., Deng, H., Wang, L., Lin, W., Ji, S., and Shen, D. (2015). Deep convolutional neural networks for multi-modality isointense infant brain image segmentation. *NeuroImage*, 108:214–224.
- Zheng, Y., Jeon, B., Xu, D., Wu, Q., and Zhang, H. (2015). Image segmentation by generalized hierarchical fuzzy c-means algorithm. *Journal of Intelligent & Fuzzy Systems*, 28(2):961–973.

AUTOBIOGRAPHY

Norberto Alejandro Hernández Leandro

Candidato para obtener el grado de
Doctor en Ingeniería
con Especialidad en Ingeniería de Sistemas

Universidad Autónoma de Nuevo León
Facultad de Ingeniería Mecánica y Eléctrica

Tesis:

SOLVING THE SHIFT SCHEDULING AND THE IMAGE SEGMENTATION
PROBLEMS USING SET COVERING FORMULATIONS

Norberto Hernández was born in Saltillo, Coahuila, on June 24, 1989. He is the eldest child of Mr. Roberto Hernández and Mrs. Rosa Elba Leandro. He holds a Bachelor degree in Applied Mathematics from Universidad Autónoma de Coahuila. In 2015, he obtained a master of science in Systems Engineering from the Graduate Program in Systems Engineering at Universidad Autónoma de Nuevo León (UANL). In 2015 he started his doctoral studies in the Graduate Program in Systems Engineering at UANL.