Universidad Autónoma de Nuevo León

Facultad de Ingeniería Mecánica y Eléctrica

División de Estudios de Posgrado

Exact Solutions for the Agricultural and the Two-Dimensional Packing Problems

por

M.C. Néstor Miguel Cid García

en opción al grado de

Doctor en Ingeniería

con especialidad en Ingeniería de Sistemas

San Nicolás de los Garza, Nuevo León diciembre, 2015

Universidad Autónoma de Nuevo León

Facultad de Ingeniería Mecánica y Eléctrica

División de Estudios de Posgrado

Exact Solutions for the Agricultural and the Two-Dimensional Packing Problems

por

M.C. Néstor Miguel Cid García

en opción al grado de

Doctor en Ingeniería

con especialidad en Ingeniería de Sistemas

San Nicolás de los Garza, Nuevo León diciembre, 2015

Universidad Autónoma de Nuevo León

Facultad de Ingeniería Mecánica y Eléctrica

División de Estudios de Posgrado

Los miembros del Comité de Tesis recomendamos que la Tesis "Exact Solutions for the Agricultural and the Two-Dimensional Packing Problems", realizada por el alumno M.C. Néstor Miguel Cid García, con número de matrícula 1541924, sea aceptada para su defensa como opción al grado de Doctor en Ingeniería con especialidad en Ingeniería de Sistemas.

El Comité de Tesis

Dra. Yasmín Á. Ríos Solís
Directora

Dr. Igor Seminovich Litvinchev
Revisor

Dr. Vincent L. Boyer
Revisor

Dr. Víctor Manuel Albornoz Sanhueza
Revisor

Dra. Sandra Ulrich Ngueveu
Revisora

Vo. Bo.

Dr. Simón Martínez Martínez
División de Estudios de Posgrado

San Nicolás de los Garza, Nuevo León, diciembre, 2015

Dedicatory

To my wife.

Acknowledgments

To God, for the life and for all the skills I have received to do this travel. This research would not be possible without the help of God.

I wish to express an special gratitude and sincere thanks to my Thesis Director, Dra. Yasmín Á. Ríos Solís, for her invaluable guidance, constant encouragement, constructive comments, sympathetic attitude, friendship, and immense motivation. Thanks for all the professional, moral, and financial support.

My deep acknowledgement to my wife "Yolis" for all her love, support, patience, and comprehension along these years, I could not have make this work without her help.

To my parents, Mr. Jaime Cid and Mrs. Ana María García. To my brothers, Jaime, Viri and Faty, and all my family. Thank you for all your unconditional support and for all your tips. Thank you so much for everything.

To my dissertation committee: Dr. Igor Seminovich Litvinchev, Dr. Vincent L. Boyer, Dr. Víctor Manuel Albornoz Sanhueza, and Dra. Sandra Ulrich Ngueveu, for all their comments, contributions and suggestions to improve the quality of this research.

Thank you to all my professors, my friends, partners and personal of the Graduate Program in Systems Engineering (PISIS) at Universidad Autónoma de Nuevo León (UANL). Thank you for your friendship and for all the moments we had. I will never forget you.

Many thanks to Mr. Jean Arlat, Director of the Laboratory for Analysis and Architecture of Systems (LAAS) of the CNRS; to Mr. Christian Artigues, manager of the team ROC (Operations Research in Combinatorial Optimization and Constraints), and to Mrs. Sandra Ulrich Ngueveu who was my host adviser, for accepting me to make a research stay at LAAS in Toulouse, France. Thanks to all the friends and all the people for your friendship and hospitality. It was an adventure that we will always remember.

Finally, but not less important, many thanks to CONACYT (Mexican National Council of Science and Technology) and FIME (Faculty of Mechanical and Electrical Engineering) for the graduate scholarship along of these years.

Contents

ABSTRACT

M.C. Néstor Miguel Cid García.

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

Universidad Autónoma de Nuevo León.
Facultad de Ingeniería Mecánica y Eléctrica.

Título del estudio:

EXACT SOLUTIONS FOR THE AGRICULTURAL AND THE TWO-DIMENSIONAL PACKING PROBLEMS

Número de páginas: 190.

OBJECTIVES AND STUDY METHOD: The objective of this study is to develop exact algorithms that can be used as management tools for the agricultural production planning and to obtain exact solutions for two of the most well known two-dimensional packing problems: the strip packing problem and the bin packing problem.

For the agricultural production planning problem we propose a new hierarchical scheme of three stages to improve the current agricultural practices. The objective of the first stage is to delineate rectangular and homogeneous management zones into the farmer's plots considering the physical and chemical soil properties. This is an important task because the soil properties directly affect the agricultural production planning. The methodology for this stage is based on a new method called "Positions and Covering" that first generates all the possible positions in which the plot can be delineated. Then, we use a mathematical model of linear programming to obtain the optimal physical and chemical management zone delineation of the plot.

In the second stage the objective is to determine the optimal crop pattern that maximizes the farmer's profit taken into account the previous management zones delineation. In this case, the crop pattern is affected by both management zones delineation, physical and chemical. A mixed integer linear programming is used to solve this stage.

The objective of the last stage is to determine in real-time the amount of water to irrigate in each crop. This stage takes as input the solution of the crop planning stage, the atmospheric conditions (temperature, radiation, etc.), the humidity level in plots, and the physical management zones of plots, just to name a few. This procedure is made in real-time during each irrigation period. A linear programming is used to solve this problem.

A breakthrough happen when we realize that we could propose some adaptations of the P&C methodology to obtain optimal solutions for the two-dimensional packing problem and the strip packing.

We empirically show that our methodologies are efficient on instances based on real data for both problems: agricultural and two-dimensional packing problems.

CONTRIBUTIONS AND CONCLUSIONS: The exact algorithms showed in this study can be used in the making-decision support for agricultural planning and two-dimensional packing problems.

For the agricultural planning problem, we show that the implementation of the new hierarchical approach can improve the farmer profit between 5.27% until 8.21%

through the optimization of the natural resources. An important characteristic of this problem is that the soil properties (physical and chemical) and the real-time factors (climate, humidity level, evapotranspiration, etc.) are incorporated.

With respect to the two-dimensional packing problems, one of the main contributions of this study is the fact that we have demonstrate that many of the best solutions founded in literature by others approaches (heuristics approaches) are the optimal solutions. This is very important because some of these solutions were up to now not guarantee to be the optimal solutions.

Firma de la directora: ______________________________

Dra. Yasmín Á. Ríos Solís

Chapter 1

Introduction

In this chapter, we give an overview about the two topics described in this Ph.D. dissertation. The first one is associated to the "Agricultural Planning" problems, and the second one is related to the "Two-Dimensional Packing" problems. Each one of these topics and the proposed methodology is defined in Section 1.1 and 1.2, respectively. Finally, in Section 1.3 we show the dissertation structure.

1.1 Agricultural Planning

1.1.1 Problem statement

The traditional methods implemented by the farmers in the agricultural planning process frequently do not consider the spatial variability of the soil properties inside of the agricultural fields. This is an important problem because the spatial variability of soil properties is one of the main impairments, which affects the productivity and crop quality.

To improve the benefits of the agricultural planning practices, it is necessary to known the spatial variability of the soil properties of the field where the crops are going to be sown. In this context, it is required to do a difference between the physical and chemical soil properties. The physical properties affects in the water supply practices while the chemical properties determine the amount of inputs to apply in the field as: seeds, pesticides, fertilizers.

Delineating the field into site-specific management zones is usually implemented to face within-field variability. Unfortunately, classical zoning methods (clustering methods, mainly), based on soil fertility variables, have a disadvantage: the zones have oval or irregular shapes, which is not practical for the variable rate technology and machinery.

In this work, we present two zoning methods that optimally delineate rectangular homogeneous management zones, using relative variance to guarantee the homogeneity of each zone. The first zoning method relies on a mono-objective integer linear programming model that is efficiently solved to optimality. The objective of this model is minimizing the variance of the zones selected to partitioning the field (see Section 2.1).

The second zoning method is based on a bi-objective integer linear programming model that minimizes the number of zones used to delineate the field, and maximizes the homogeneity level of each zone (see Section 2.3.2). The field is delineated using both properties physical and chemical. Experimental results on real and generated instances validate the methods and enable a graphical visualization of the solution.

Once the field has been delineated with the physical and chemical management zones, the next step is to select the optimal crop pattern, which maximizes the farmer profit (see Section 2.2). The optimal crop pattern means that the model is going to select the best crop for each plot considering their physical and chemical management zones. For this purpose, another mixed integer linear programming model has been developed where, for each zone, the supply of inputs is given in a site-specific way. For the case of the bi-objective model for management zones, the previous model helps to determine which delineating is the best for each parcel.

Finally, we show another mathematical model that at each irrigation turn decides how much and which plots must be watered such to maximize the total final yields (see Section 2.2). The model considers the physical management zones and the humidity level of each parcel to determine in real-time the optimal amount of water for each crop. This is a critical decision in countries where water shortages are frequent. In this study we integrate these stages in a hierarchical process for the agriculture planning and empirically prove that our methodologies are efficient on instances based on real data.

1.1.2 Objectives

General objective

To develop a decision-making tool that helps to the farmers to improve the agricultural planning practices, with the objective of maximizing the farmer profit and optimizing the available resources through the whole production cycle.

Specific objectives

- To develop a decision-making tool that optimally delineates rectangular homogeneous management zones inside of agricultural fields.
- To develop a decision-making tool to select the optimal crop pattern which maximizes the farmer profit at the end of the production cycle.
- To develop a decision-making tool to determine the optimal amount of water to be irrigated in each crop considering its real-time water requirements and its phenological stage of growing, which maximizes the farmer profit at the end of the production cycle.

1.1.3 Hypothesis

The development of a decision-making tool will improve the agricultural planning practices, maximizing the farmer profit, minimizing the production costs and optimizing all the available resources for the whole production cycle.

1.1.4 Scientific methodology

The next methodology is used to develop each one of the objectives proposed in Section:

- Bibliography review. The objective of this stage is to find approaches related with the agricultural planning practices and see which methodologies have been proposed to solve this problem: exact and heuristic methods. Our methodology is going to be compared with these methods.
- Mathematical formulation. A linear programming mathematical model is going to be developed for each phase of the agricultural planning problem. The CPLEX solver will be used to solve these models.
- Computational implementation and experimental results. In this stage, we generate some instances using random data to validate the models performance. The models are solved with the branch and bound algorithm of CPLEX.
- Results analysis. The experimental results are analyzed to verify the correct performance of the proposed models.

1.2 Two-Dimensional Packing Problems

1.2.1 Problem statement

The Two-Dimensional Packing Problems are classical problems that we can found in several industries process where is necessary to allocate a set of rectangular items inside of larger rectangular sheets of material. For example: wood, paper, steel or glass industries.

Inside of this class of problems, we can distinguish two kinds of problems: The Bin Packing Problem and the Strip Packing Problem. In the Two-Dimensional Bin

Packing Problem we have an unlimited number of identical rectangular bins with fixed width and height, and a set of rectangular items each one with specific width and height. The objective of this problem is to allocate the set of rectangular items using the minimum number of bins.

For the Two-Dimensional Strip Packing Problem we have a rectangular strip with fixed width but infinite height, and a set of rectangular items each one with specific width and height. In this case, the objective consist in allocate all the set of rectangular items into the strip using the minimum height of the strip.

Both problems, Strip and Bin Packing, are NP-Hard in the strong sense, since a reduction of this problems can be easily made for the one-dimensional bin packing problem, that is strongly NP-Hard. Due to the combinatorial complexity of the problem, many studies have been focused to use heuristics and metaheuristics methods to solve this problem, and just a few approaches used exact methods to give a solution for the problem.

In this study, we present a new formulation and a methodology, called Positions and Covering (P&C), to obtain exact solutions for the Two-Dimensional Bin and Strip Packing Problems, see Section 3.1 and 3.2, respectively. The methodology is based on a two-stage procedure where is solved a Set-Covering formulation of integer linear programming. The P&C methodology was tested using the benchmark of literature for Bin and Strip Packing problems. P&C was able to solve small and medium instances, but for large instances P&C cannot solve them.

1.2.2 Objective

To develop a new methodology that optimally solves the classical NP-Hard two-dimensional bin and strip packing problems.

1.2.3 Hypothesis

The new methodology will obtain optimal solutions for the classical NP-Hard two-dimensional bin and strip packing problems.

1.2.4 Scientific methodology

The next methodology is used to develop each one of the objectives proposed in Section:

- Bibliography review. The objective of this stage is to find approaches related with the methodologies proposed for two-dimensional packing problems and see which methodologies have been proposed to solve these problems: exact and heuristic methods. Our methodology is going to be compared with these methods.
- Mathematical formulation. A linear programming mathematical model is going to be developing for each two-dimensional packing problem. The CPLEX solver will be used to solve these models.
- Computational implementation and experimental results. In this stage, we use the benchmark instances, for each two-dimensional packing problem, to validate the models performance. The models are solved with the branch and bound algorithm of CPLEX.
- Results analysis. The experimental results are analyzed to verify the correct performance of the proposed models.

1.3 Dissertation structure

The Ph.D. dissertation is structured for 3 chapters. Each ones is described next:

- Chapter 2 shows the topics related to the *Agricultural Planning Problems*. Section 2.1 presents a new method to "Delineate Rectangular and Homogeneous Management Zones". Section 2.2 shows "A Crop Planning and Real Time Irrigation Method Based on Site-specific Management Zones and Linear Programming". Finally, in Section 2.3 is presented "A Hierarchical Planning Scheme Based on Precision Agriculture" to improve the agricultural practices.

- In Chapter 3 are presented the topics related to exact solutions for the *Two-Dimensional Packing Problems*. In this sense, we present a new methodology called "Positions and Covering (P&C)" to obtain exact solutions for the Bin and Strip Packing Problems, see Sections 3.1 and 3.2, respectively. A short description about a decomposition approach to solve large scale instances of the Bin and Strip Packing Problems, using the P&C methodology, is showed in Appendix B.

- Chapter 4 presents the final conclusions and the future work for both topics tackled in this scientific work: the agricultural and two-dimensional packing problems.

- Appendix A presents some basic concepts related to the optimization area, a short history about the operations research, and a brief description of some terms used in this Ph.D. dissertation, as linear programming and multi-objective optimization, are showed too.

- Appendices C and D show a summary of notation for the crop planning problem (CPP) and the real-time irrigation problem (RTIP), respectively.

Chapter 2

Agricultural Planning

2.1 Delineation of Rectangular and Homogeneous Management Zones

This research produced the article: Cid-Garcia, N. M., Albornoz V., Rios-Solis, Y. A., & Ortega R. (2013). "Rectangular shape management zone delineation using integer linear programming". *Computers and electronics in agriculture*, 93, 1-9.

Abstract

The spatial variability of the soil properties is one of the main impairments to the productivity and crop quality in agriculture. Delineating the field into site-specific management zones is usually implemented to face within-field variability. Classical zoning methods, based on soil fertility variables, have a disadvantage: the zones have oval shapes which are not practical for the variable rate technology and machinery. In this work, we present a new zoning method that optimally delineates rectangular homogeneous management zones, using relative variance to guarantee the homogeneity. This zoning method, based on soil properties, relies on an integer linear programming model that is efficiently solved to optimality. Experimental results on real and generated instances validated the method and enabled a graphical visualization of the solution.

2.1.1 Introduction

One of the main aspects of precision agriculture is to provide farming management methods to respond to within-field variability. It relies on new technologies like satellite imagery, information technology, and geospatial tools to improve the decision-making process in agricultural production. As mentioned in Ortega y Santibáñez (2007), in contrast with “traditional” uniform field management, precision agriculture permits the application in a site-specific manner of agronomic practices such as fertilization, weed and pest control, as a function of the information compiled from collected field data. The impact of precision agriculture derives from the fact that most factors determining crop yield and quality are variable in space and time. To be more efficient, management decisions must be time- and site-specific and not rigidly programmed.

Within precision agriculture, an important area is the site-specific nutrient management since there is a need of delineating management zones within fields before planting the crops to improve the overall yield. More precisely, a management zone is a sub-region of a field that is relative homogeneous with respect to soil parameters, and for which a specific rate of inputs is needed (Roudier *et al.*, 2008). Indeed, variable rate technology uses equipment to apply inputs at a precise location to achieve site-specific application rates of inputs to reduce input and labor costs, maximize productivity, and to reduce the impact wastage on the environment. Mainly, variable rate technology in agriculture includes fertilizer, lime, seeding, and pesticides.

As mentioned by Doerge (1999), the most meaningful factors to include in a management zone strategy are those with the most direct effects on crop yield: soil moisture relationships, soil pH, soil pathogen infestation, and extremes in soil nutrient levels (see also Cambardella *et al.* (1994); Ortega y Flores (1999)).

Trying to delineate management zones efficiently and accurately is a mayor challenge where decision support systems are needed (McBratney *et al.*, 2005). In this study we used data of the soil properties to propose the Integer Linear Programming Management Zone delineation method (ILPMZ for short) based on a mathematical model that could be easily inserted in any decision support system. The main advantage of the zones that the ILPMZ zoning method computes, is that they have a rectangular shape which is an important characteristic for agriculture machinery. Moreover, rectangular parcels (or portions of them) allow easier adoption of variable rate technologies based on prescription maps than irregular parcels. Additionally, this zoning method could also be applied for drip irrigation designs.

There are several approaches in the literature for properly determining site-specific management zones. Most of them are based on clustering algorithms, i.e., they are classification-based approaches.

- Approaches based on information of the soil. For example Schepers *et al.* (2004) and Fraisse *et al.* (2001) use soil and relief information; Carr *et al.* (1991) base their zoning on topographic maps; while methods of Mulla (1991), Mortensen *et al.* (2003), and Bhatti *et al.* (1991) need soil sampling.

- Approaches based on yield maps, combining data from several seasons. We can cite Blackmore (2000), Diker *et al.* (2004), and Pedroso *et al.* (2010). Doerge (1999) pointed out that crop yield patterns from yield maps may not be stable enough across seasons to accurately define management zones without supplemental information.

- Integration of the two previous approaches as in Whelan *et al.* (2003), Franzen y Nanna (2003), Hornung *et al.* (2003, 2006). In Roudier *et al.* (2008) they use a watershed segmentation algorithm where the user can introduce morphologies of the desired zones.

The combination of the different layers of information can be performed by a cluster procedure using K-means or Fuzzy K-means methods (Ortega *et al.*, 2002; Li *et al.*, 2005; Jiang *et al.*, 2011), or principal component analysis with a cluster method (Ortega y Flores, 1999). The Fuzzy K-means algorithm is widely used and the choice of the data layers processed by the clustering is an issue (Jaynes *et al.*, 2005). A major drawback is the resulting fragmentation of the zones (Simbahan y Dobermann, 2006; Frogbrook y Oliver, 2007; Li *et al.*, 2005). Moreover, this zones are oval shaped and disjoint due to the clustering methods.

Figure 2.1 schemes the real word problematic we solve in this work: Some farmers become aware that precision agriculture leads to important saving (e.g., in fertilizers) by delineating management zones. For this, they invest in soil samples of their fields that are then analyzed in a laboratory (dots in the maps are the places where the soil samples where taken). The results of the properties of the soil samples

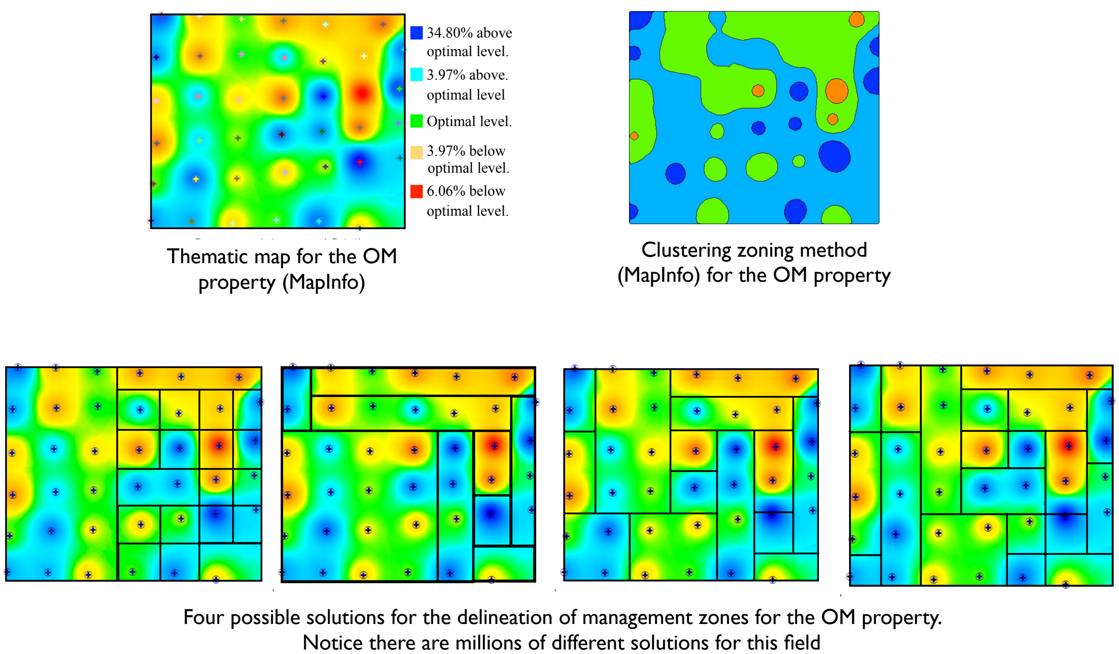


Figure 2.1: Left upper map: thematic map for the organic matter property (OM) obtained with MapInfo. Right upper map: Clustering zoning method from MapInfo for the OM property. Each color of this map represents a management zone (green, orange, light blue, and dark blue). Bottom maps: four (out of millions) different solutions of management zones obtained by making a grid based on the smallest size of a zone.

can be visualized as thematic maps like the one in the left upper part of Figure 2.1

(Organic Matter (OM) is used as soil property and MapInfo as visualization software). By using a clustering method (e.g., the one in MapInfo software) with the organic matter as soil chemical property we obtain the upper right map of the same figure. Each color of this map represents a management zone (green, orange, light blue, and dark blue). We can notice that the resulting zones are disjoint and are irregular shaped which difficulty the use of machinery and therefore the application of resources and inputs. Farmers then think about tracing a grid and delineate their own zones based on the clustering zones or on the thematic map which results in any of the solutions presented in the bottom of Figure 2.1. The drawback of this approach is that there is an exponential number of different possible management zone delineations. In order to find the best delineation, farmers must try all of them (e.g., compute the costs of fertilizers) and this would literally take years to be completed. With the ILPMZ method we offer the farmer the best management zones delineation in minutes such that they are rectangular and the most homogeneous possible within each zone.

To the best of our knowledge, this is the first approach that directly offers rectangular shape zones which is an important characteristic for variable rate technologies since it facilitates the work and operation of machinery. Indeed, broadcast seeders (used for spreading lime or fertilizer), manure spreader or sprayers are usually towed behind a tractor. If a management zone is rectangular then it is easier for the farmer to indicate its limits to the tractor driver. Therefore, agricultural inputs are spread exactly in the management zone that requires them.

The ILPMZ method delineates the most homogeneous rectangular management zones from a field with respect to the properties of the soil. It consists of three main stages:

a) **Instance generation.** In this stage, data from grid soil sampling of a given field is processed: for each soil sample we have its coordinates, and a set of soil properties (pH, organic matter, phosphorus, nitrogen, etc.). Then, a thematic map of the field is created with respect to the wished soil property (or properties) and all the quarters (or potential zones) are computed together with their variances.

b) Mathematical model. With the input of stage a), we propose an Integer Linear Programming (ILP) that is solved with a branch and bound algorithm (see Section A.3). The aim of the ILP is to find a set of the most homogeneous quarters that minimizes the sum of their variance and that covers the whole field.

A main contribution is the insertion of the relative variance into the model to guarantee the homogeneity of the managements zones.

c) Visualization. This module translates the solution given by the mathematical model into a graphical view of the most homogeneous rectangular zones for the field.

The main novelty of procedure ILPMZ is that step a) is not only an instance generator, it also computes the different potential management zones of the parcels. This way step b) becomes tractable since the mathematical model can be exactly solved in seconds.

The rest of this paper is as follows. In Section 2.1.2 we introduce our ILPMZ delineation method. The instance generation phase is described in Section 2.1.2, the Mathematical model is introduced in Section 2.1.2 while the visualization phase is described in Section 2.1.2. In Section 2.1.3 we experimentally test our methodology, prove its efficiency, and accuracy. Finally, Section 2.1.4 concludes the work.

2.1.2 Materials and Methods

The ILPMZ method delineates rectangular management zones from a field with respect to some properties of the soil. As mentioned, it consists of three main stages: Instance generation, Mathematical model, and Visualization. Each one of these parts of the ILPMZ methodology is explained with more details in this section.

Instance generation

The objective of this stage is to generate an instance for the Mathematical Model stage from the soil samples that have been taken from the field. These soil samples are approximately equidistant in the field since they were generated from a systematic grid sampling with the help of the Software SMS Mobile and a GPS receiver model AgLeader 1500 with e-diff. Sample positions were collected in geographic coordinates (lon, lat) using the WGS84 datum. Coordinates were then converted to UTM zone 19 S and from them to Cartesian coordinates, in meters.

An example of the soil sample data needed by our methodology can be visualized in Table 2.1. This table presents the data of 40 soil samples of an agriculture field close to Santiago, Chile (we use this field along the work and call it Real Field instance). This field has 256 meters width and 305.6 meters long (around 7.82 *ha*). The samples are approximately spaced by 50 meters one from each other so four soil samples are needed to cover an *ha*. Then, the samples are labelled (first and fourth column of the table) and their positions are translated into a Cartesian map (coordinates (x, y) in the table). Finally, the information about each soil property is presented (pH, organic matter rate (OM), amount of phosphorus (P), and sum of bases (SB). Phosphorus is the most limiting factor in Chilean soils while SB and OM are good indicators of overall soil fertility. Organic matter was determined by the wet oxidation method, extractable P by the Olsen method, while SB corresponds to the sum of bases determined by the CH3COONH4 method of INIA (2006).

The visualization of this data is called a thematic map. In Figure 2.2 two thematic maps are presented from the data of Table 2.1. The left map is the thematic map for the OM property and the right one is for the P property. Here we used MapInfo with the default grid and the inverse distance weighting interpolator. It can be noted that any interpolation method such as kriging, nearest neighbour or other could have been used, since data is spatially dependent.

Table 2.1: Coordinates and values of the soil properties for each one of the 40 soil samples taken from a 7.82 *ha* field close to Santiago, Chile. OM is in %, P in $mg\ kg^{-1}$ and SB in $Cmol(+)\ kg^{-1}$.

Sample	Coordinates	Soil properties				Sample	Coordinates	Soil properties			
	(x, y)	pH	OM	P	SB		(x, y)	pH	OM	P	SB
1	0.00, 9.14	5.2	11.8	8.0	5.89	21	297.68, 166.36	5.6	10.4	4.0	8.26
2	48.97, 8.46	5.5	12.8	4.0	7.97	22	253.87, 160.20	5.4	18.7	11.0	8.88
3	97.52, 5.57	5.2	14.9	10.0	7.63	23	206.99, 157.26	5.6	10.5	11.0	6.03
4	150.52, 9.42	5.4	14.0	7.0	11.44	24	158.29, 155.16	5.5	16.8	3.0	9.48
5	201.07, 8.25	5.5	11.2	4.0	6.36	25	105.27, 153.53	5.4	14.8	5.0	7.85
6	250.24, 0.00	5.4	14.7	4.0	9.31	26	56.47, 156.87	5.5	12.6	5.0	5.38
7	298.57, 84.00	5.6	12.5	6.0	10.03	27	6.15, 151.48	5.4	15.1	7.0	6.50
8	249.94, 78.89	5.6	9.6	4.0	7.99	28	6.33, 204.03	5.4	11.7	5.0	5.88
9	208.71, 73.33	5.5	14.3	6.0	8.20	29	58.83, 205.57	5.5	16.0	4.0	8.09
10	160.73, 66.20	5.5	15.0	6.0	9.23	30	108.59, 207.64	5.4	13.8	4.0	8.18
11	102.69, 59.51	5.4	14.5	5.0	6.64	31	159.65, 203.22	5.6	12.6	3.0	7.95
12	53.66, 58.30	5.4	11.1	6.0	6.00	32	206.04, 199.18	5.4	14.4	6.0	7.50
13	2.81, 52.71	5.3	14.1	5.0	5.67	33	255.23, 205.16	5.4	15.4	5.0	8.23
14	6.93, 101.13	5.3	16.3	6.0	5.51	34	303.14, 212.73	5.7	11.2	5.0	9.51
15	58.25, 105.04	5.4	12.7	7.0	6.36	35	278.06, 242.75	5.2	16.6	22.0	7.30
16	104.05, 107.24	5.4	14.2	6.0	7.80	36	208.60, 243.31	5.5	15.6	8.0	9.21
17	156.53, 111.44	5.5	11.4	5.0	6.72	37	158.68, 247.47	5.5	16.1	5.0	9.51
18	204.49, 114.91	5.5	11.5	8.0	6.11	38	108.00, 249.65	5.4	13.9	6.0	6.90
19	250.37, 119.77	5.4	16.7	6.0	8.75	39	58.16, 253.69	5.5	15.4	5.0	9.69
20	296.17, 124.74	5.5	13.5	5.0	7.81	40	12.72, 254.37	5.4	10.7	4.0	7.71

These thematic maps reveal the diversity of the soil in the field. It is easy to conclude that applying the same amount of inputs (seed, fertilizers, pesticides, water, etc.) throughout the field, would result in few zones at optimum level. With only these thematic maps, it is an extremely difficult task to delineate the most homogeneous managements zones of the field. Moreover, the management zones should be rectangular to be a realistic solution for a farmer. It is known that fields with rectangular shape are better in terms of machinery efficiency. Additionally, for

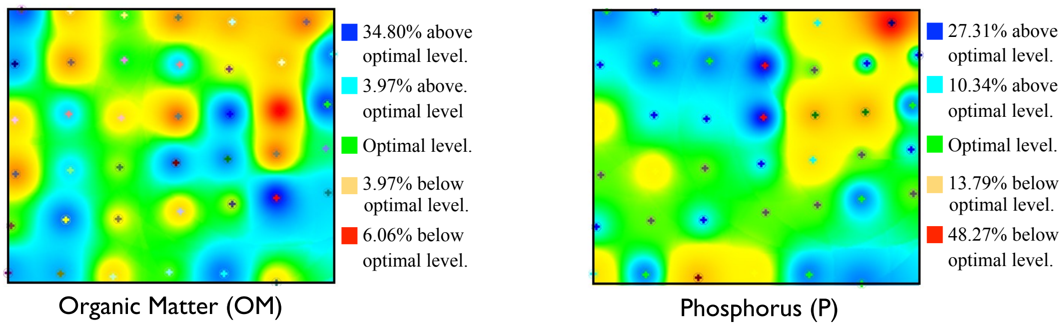


Figure 2.2: Thematic maps of organic matter and phosphorus generated with the information of Table 2.1.

drip irrigation design, rectangular fields are usually used. This leads to the ILPMZ management zone delineation problem which belongs to the NP-hard class. In this article, a new integer linear programming is proposed in order to obtain the best solution among the combinatorial number of zoning patterns of the field.

With the information of the samples, we proceed to generate all the potential zones (or quarters) of the field. Unlike a cutting stock problem, here we are not generating all possible patterns of a field. Instead, we are forming the potential rectangles that could be a zone. This search is the key point of ILPMZ and it can be done in $\mathcal{O}(WidthF \times LengthF)$ where $WidthF$ is the number of samples in the width of the field while $LengthF$ is the number of samples in its length.

An illustration is given in Figure 2.3. The left hand side of this figure shows a thematic map of a small field with nine samples. On the right side, all potential zones are marked. For this example, we have a total of 36 rectangular quarters (generated by Algorithm 1 presented below). Usually, the soil samples are almost equidistant, e.g., four soil samples (two width for two long) are needed to cover an ha and the minimum area covered is of a quarter of ha. The number of soil samples to cover an ha may change according to each farmer's requirements.

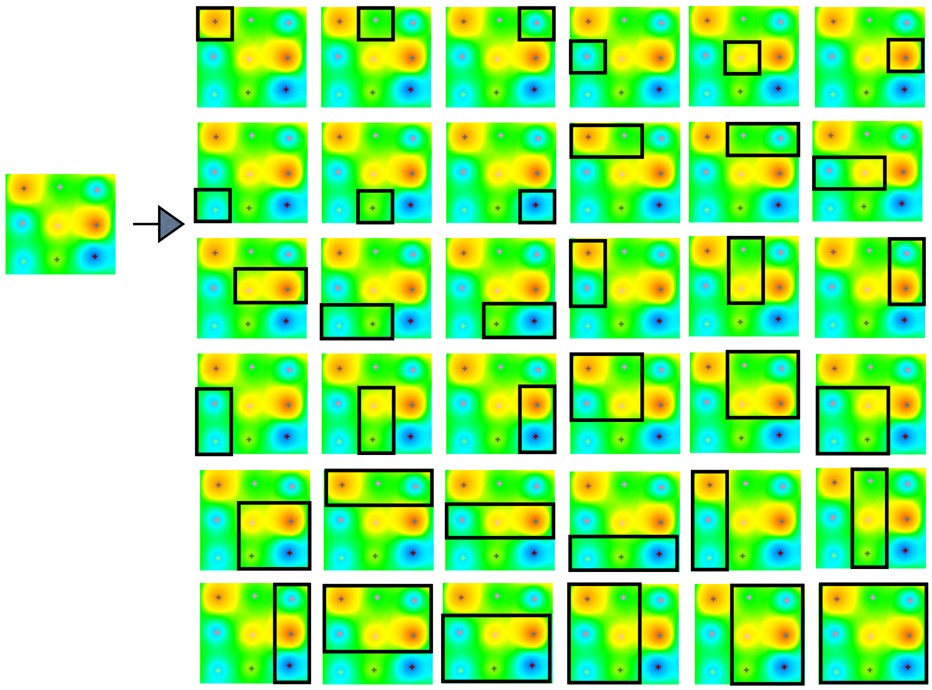


Figure 2.3: The 36 quarters of a field.

The total number of potential zones $|Q|$ can be computed by the following formula:

$$|Q| = \left(\sum_{i=1}^{WidthF - MinWidthQ + 1} i \right) \left(\sum_{j=1}^{LengthF - MinLengthQ + 1} j \right).$$

Indeed, this determination of all potential zones is not hard since there are only a polynomial number of them. This manner, some of the computations are *done* outside the mathematical model of step b), making it more tractable to solve.

The determination of all possible quarters is implemented by Algorithm 1. The input of this algorithm is the soil samples data (as in Table 2.1), the number of samples in the width of the field ($WidthF$), the number of samples in the length of the field ($LengthF$), the minimum quantity of samples the width of a quarter (zone) must contain ($MinWidthQ$), and the minimum quantity of samples the length of a quarter must contain ($MinLengthQ$). The algorithm starts creating the smallest quarters width wise. Then it checks if there is still some width to cover. After, it checks the length.

Algorithm 1 Quarters generation of a field.

```
INPUT: WidthF, LengthF, MinWidthQ, MinLengthQ, soil samples
for j = MinWidthQ To WidthF do
  for l = 0 To (WidthF − 1) do
    if (j + l) ≤ WidthF then
      for i = MinLengthQ To LengthF do
        for k = 0 To (LengthF − 1) do
          if (k + i) ≤ LengthF then
            creation of a new quarter
          end if
        end for
      end for
    end if
  end for
end for
```

With this algorithm we can create the correspondence matrix $C = \{c_{ij}\}$. If $c_{ij} = 1$, then quarter i covers sample point j, $c_{ij} = 0$ otherwise. Once all the potential quarters are enumerated, we also compute for each one of them its variance, i.e., we compute the variance of a particular soil property for the set of the samples included in a potential quarter. The correspondence matrix of the field of Figure 2.3 appears in Table 2.2. The rows are the quarters, the columns are the sample points except for the last column that corresponds to the variance of the quarter. Notice quarter 1 only covers soil sample 1 and therefore, its variance is 0. Quarter 6 covers soil samples 1, 2, and 3, while quarter 36 covers all soil samples (i.e., there is only one zone that is equal to the field).

Most of the fields are not initially rectangular (see the example of Figure 2.3). In this case, the ILPMZ method inserts dummy soil samples to fill a rectangle where the field can be contained. This dummy samples are like the real ones: equidistant one from each other. Nevertheless, their data about the properties is very high

Table 2.2: Correspondence matrix C for the field of example of Figure 2.3 with the variance of each quarter i.

		Sample point j									σ_i^2
		1	2	3	4	5	6	7	8	9	
Potential quarter i	1	1	0	0	0	0	0	0	0	0	0
	2	0	1	0	0	0	0	0	0	0	0
	3	0	0	1	0	0	0	0	0	0	0
	4	1	1	0	0	0	0	0	0	0	0.21
	5	0	1	1	0	0	0	0	0	0	0.57
	6	1	1	1	0	0	0	0	0	0	0.57
	7	0	0	0	1	0	0	0	0	0	0.00
	8	0	0	0	0	1	0	0	0	0	0.00
	9	0	0	0	0	0	1	0	0	0	0.00
	10	0	0	0	1	1	0	0	0	0	2.40
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
	36	1	1	1	1	1	1	1	1	1	2.07

with respect to the real samples. This manner, the mathematical model make this dummy soil samples to be alone or with other dummy samples which facilitates their elimination in the visualization stage of the ILPMZ method.

Next stage of the ILPMZ algorithm is the Mathematical Model. It requires the correspondence matrix of the potential quarters together with their variances.

Mathematical model

Once we have the data from the samples transformed into a correspondence matrix, we proceed to run an integer linear programming model. This model minimizes the sum of the variance of the potential quarters such that they cover the whole field and that comply with a given relative variance that guarantees the homogeneity of the management zones. This manner, the field will have delineated homogeneous rectangular management zones.

Let I be the set of potential quarters and J the set of soil samples of the field. Each quarter i has n_i soil sample points. The total number of soil samples points is N. Farmers do not wish to have tiny management zones because of their machinery, so let LS be the maximum number of zones in the field while LI is the minimum one. Now we can state the decision variables of our model:

$$x_i = \begin{cases} 1 & \text{if quarter } i \text{ is chosen,} \\ 0 & \text{otherwise.} \end{cases}$$

Our proposed integer linear programming is named ILP and is as follows.

$$\min \quad \sum_{i\in I} \sigma_i x_i \tag{2.1}$$

$$\textit{subject to:} \tag{2.2}$$

$$\sum_{i\in I} c_{ij} x_i = 1 \quad \forall j \in J \tag{2.3}$$

$$\sum_{i\in I} x_i \leq LS \tag{2.4}$$

$$\sum_{i\in I} x_i \geq LI \tag{2.5}$$

$$x_i \in \{0,1\} \qquad \forall i \in I$$

Objective function (2.1) minimizes the sum of the variance of each chosen zone (or potential management zone). Restriction (2.3) ensures that every point sample j is covered by only one zone, i.e., the whole filed is partitioned into non overlapping zones. Constraints (2.4) and (2.5) limit the number of zones in which the field will be partitioned.

To guarantee a homogeneous zoning delineation we propose to introduce to the ILP model the relative variance since it has been proved to be a high quality criterion to measure the efficiency of a zoning method Ortega y Santibáñez (2007). Suppose a set of quarters Q satisfy restrictions (2.3)–(2.5), i.e., quarters in Q cover all the field and satisfy the minimum and maximum number of allowed zones, then the relative variance of Q is as follows:

$$RV(Q) = 1 - \frac{\sum_{i\in Q} \sigma_{w_i}^2}{\sigma_T^2},$$

where σ_T^2 is the total variance of all the field and sum of the $\sigma_{w_i}^2$ for $i \in Q$ is the variance within defined as follows:

$$\sum_{i \in Q} \sigma_{w_i}^2 = \frac{\sum_{i \in Q}(n_i - 1)\sigma_i^2}{N - |Q|}. \tag{2.6}$$

Numerator in (2.6) considers the number of samples n_i in quarter i (minus one degree of freedom) as a weight and the denominator takes into account the number of selected quarters (total number N minus the number of quarters $|Q|$). Therefore, we introduce the following restriction to ILP:

$$\left(1 - \frac{\sum_{i \in I}(n_i - 1)\sigma_i^2 x_i}{\sigma_T^2 \left[N - \sum_{i \in I} x_i\right]}\right) \geq \alpha. \tag{2.7}$$

Restriction (2.7) implies that the relative variance of all the chosen quarters is at least α which is a parameter that has to be greater than 0.5 (value given by the experts) to guarantee an homogeneous behaviour of the zoning method. This restriction can be easily put into a linear form:

$$(1 - \alpha)\, \sigma_T^2 \left[N - \sum_{i \in I} x_i\right] \geq \sum_{i \in I}(n_i - k)\sigma_i^2 x_i. \tag{2.8}$$

Model ILP enhanced with restriction (2.8) delineates a field into rectangular and most homogeneous management zones. ILP is an NP-hard problem[1] that can be solved with a branch and bound method[2] for a field with 30 × 30 soil samples (see Section 2.1.3). This efficiency is mainly due to the elegance of ILP: few restrictions but sufficiently close to the convex hull of the solution space. Therefore, there is no need of extra valid inequalities and even less, there is no need of an heuristic algorithm which would provide solutions without optimality guarantee.

To the best of our knowledge, there is not a field in Chile with more than 30×30 soil samples. Although, there could be larger instances where a different solution methodology would be required since ILP is NP-hard. The key point of determining the potential zones at the instance generation stage makes the ILP mathematical model simpler but still NP-hard.

[1]A knapsack problem or a general assignment problem can be easily reduced to ILP.

[2]A branch and bound is an enumeration algorithm. Therefore, when it ends it gives the optimal solution (more details in Wolsey (1998)).

VISUALIZATION

The solution of ILP, as it is given by standard optimization software is not friendly for a farmer. A visualization stage must translate the solution of ILP into a map that is useful for the farmer. This map must indicate the characteristics of each one of the final zones so that the farmer knows, for example, where to fertilize. Notice that the dummy soil samples that we inserted in the Instance Generation stage must be deleted at this point since they will be naturally discarded by the method (see more detail in Section 2.1.3).

A summary of the ILPMZ method is presented in Figure 2.4. Initially we only have the soil samples from the field we want to delineate into rectangular homogeneous management zones. If the field is not rectangular, then we must add dummy samples that will be then deleted. Then, the method proceeds with the Instance Generation stage where the thematic maps of the different soil properties are presented, the potential quarters are generated together with their variance, and finally, the correspondence matrix is computed. This information is the input of the mathematical model ILP which is solved by a branch and bound method. If there is not a feasible solution then one must adjust either the value of LS (maximum number of zones in the field) or reduce parameter α which is equal to relax the homogenization of the zones.

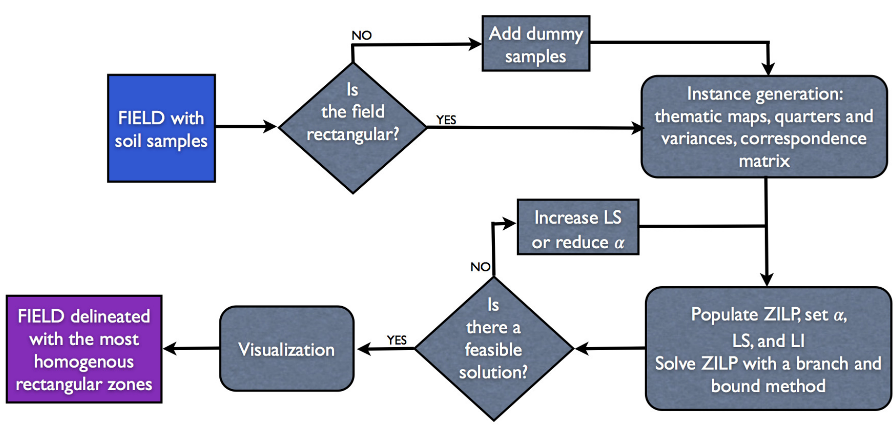


Figure 2.4: Summary of the ILPMZ method.

Indeed, a strict bound on LS or in the shape of the quarters may lead to the unfeasibility of ILP. Then, the method must be flexible in one of these parameters with respect to the farmers wishes. More details are presented in Section 2.1.3.

2.1.3 Experimental Results

In this section we validate the ILPMZ method in eleven different fields:

- Real field: vineyard close to Santiago, Chile, with 256 meters width and 305.6 meters long (around 7.82 ha). Tables 2.1 and 2.2 of Section 2.1.2 describe this instance.
- Large fields: 10 fictitious fields based on generated data (realistic parameters) with at most 900 soil samples (30 $\times$ 30 samples). This fields are to test the efficiency of the ILPMZ method.

Two codes need to be executed for the ILPMZ method on a standard personal PC[3]:

- Quarters generation: Algorithm 1 creates all the possible quarters that could be a management zone. This algorithm was coded in Visual Basic for Applications for Excel.
- B&B: The branch and bound algorithm to solve mixed integer linear programs was implemented by GAMS/CPLEX 12.2 using default options, except for the optimal criterion fixed at 0.

Table 2.3 present the experimental results, for OM, P, pH and, SB soil properties of an vineyard field close to Santiago, Chile, that has a total of 42 soil samples (6 soil samples width by 7 length) with a total number of quarters $|Q| = 588$ (generated by Algorithm 1 of Section 2.1.2). For this field the minimum shape of a management zone, $w \times l = 1 \times 1$ means the minimum zone is w soil samples width for l soil

[3]PC Intel Core 2 Duo of 2.0 GHz, and 4 GB of RAM.

samples long (one can translate soil samples into distance easily). The homogenization parameter α is set to 0.5. First column of Table 2.3 is the maximum number of management zones the farmer wishes for this field. Second column presents the solution of the objective function of ILP for the total variance where“–” means that no feasible solution could be found (parameters α must be reduced). Last column corresponds to the resulting number of management zones used to partition the field.

Table 2.3: ILPMZ method applied to a 7.82 *ha* field close to Santiago, Chile, for the MO, SB, pH, and P soil properties, with the homogenization parameter $\alpha = 0.5$, and total size of quarters $|Q| = 588$.

OM

LS	σ^2	Zones
42	0.00	42
20	3.12	20
15	4.73	15
10	9.43	10
9	11.52	9
8	–	–

SB

LS	σ^2	Zones
42	0.00	42
20	0.93	20
15	1.34	13
10	1.64	10
7	3.59	7
6	–	–

pH

LS	σ^2	Zones
42	0.000	42
20	0.002	19
15	0.004	12
10	0.006	10
5	0.014	5
4	0.021	4
3	–	–

P

LS	σ^2	Zones
42	0.00	42
20	1.25	18
15	1.95	15
10	3.46	10
5	4.24	5
3	6.21	3
2	–	–

Table 2.3 shows that the ILPMZ with the homogeneity parameter $\alpha = 0.5$ leads to high quality solutions. The maximum number of management zones LS plays an

important role since in most of the solutions this bound is reached. The smaller the management zones, the most homogeneous they are going to be. Nevertheless, a farmer does not wish to have tiny management zones. When LS is too restrictive, the ILP may not find feasible solutions that satisfy $\alpha = 0.5$. Therefore, one could relax this parameter by reducing it until the ILP finds a solution as presented in Scheme 2.4.

The relationship between variance and quarters for the OM and P soil properties with α equal to 0.5 is showed in Figure 2.5. As expected, the number of quarters increases when the value of the variance decreases. Hence, the maximum number of zones given by the farmer to partition the field is an important aspect to consider since the value of the variance within the field depends on it.

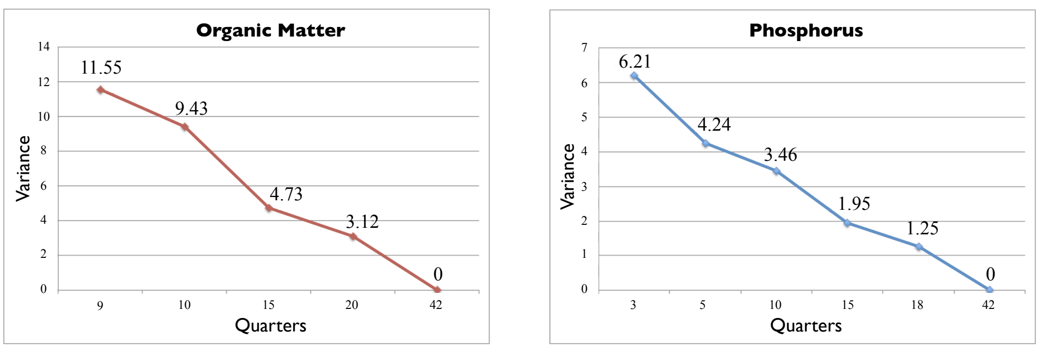


Figure 2.5: Relationship between Variance and Quarters applied to a 7.82 *ha* field close to Santiago, Chile, for the OM and P soil properties for $\alpha = 0.5$.

Figure 2.6 is the visualization of the ILPMZ applied to the real instance related to the SB soil property. The left side has a minimum size of the management zones of $w \times l = 1 \times 1$ while the right side is set to $w \times l = 2 \times 1$. As mentioned, the result of the zoning drastically vary depending on the minimum size of the quarter and the homogeneity parameter α.

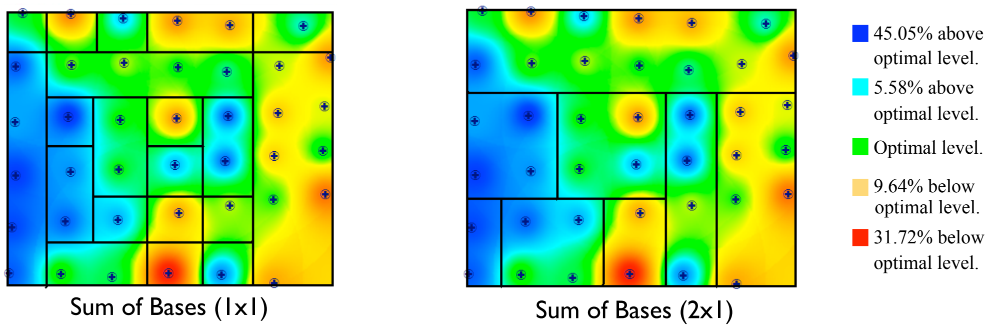


Figure 2.6: ILPMZ applied to a real field close to Santiago, Chile, related to the SB soil property with minimum size of the management zones $w \times l = 1 \times 1$ and $w \times l = 2 \times 1$, respectively.

Table 2.4: ILPMZ method applied to the real instance for the OM and SB soil properties.

OM

$w \times l$	α	LS	$\lvert Q \rvert$	σ^2	Zones
1×2	0.1	42	441	17.35	6
1×2	0.1	20	441	17.35	6
1×2	0.1	5	441	22.18	5
1×2	0.1	4	441	–	–
2×1	0.4	42	420	19.50	11
2×1	0.4	20	420	19.50	11
2×1	0.4	9	420	21.49	9
2×1	0.4	8	420	–	–
2×2	0.0	42	315	4.61	1
2×2	0.0	20	315	4.61	1
2×2	0.0	10	315	4.61	1
2×2	0.0	5	315	4.61	1
3×3	0.0	42	150	4.61	1
3×3	0.0	20	150	4.61	1
3×3	0.0	10	150	4.61	1
3×3	0.0	5	150	4.61	1

SB

$w \times l$	α	LS	$\lvert Q \rvert$	σ^2	Zones
1×2	0.4	42	441	7.24	7
1×2	0.4	20	441	7.24	7
1×2	0.4	7	441	7.24	7
1×2	0.4	6	441	–	–
2×1	0.5	42	420	7.75	8
2×1	0.5	20	420	7.75	8
2×1	0.5	7	420	7.76	7
2×1	0.5	6	420	–	–
2×2	0.3	42	315	5.09	4
2×2	0.3	20	315	5.09	4
2×2	0.3	4	315	5.09	4
2×2	0.3	3	315	–	–
3×3	0.2	42	150	3.17	2
3×3	0.2	20	150	3.17	2
3×3	0.2	2	150	3.17	2
3×3	0.2	1	150	–	–

In Tables 2.4 and 2.5 are showed the results obtained by ILPMZ when varying the minimum size of a zone ($w \times l$) in the Real Instance (first column). Second column is related to parameter α. It is first set to 0.5, but if no solution can be found (ILP is unfeasible) then it is reduced by 0.1. In the table we show the first α for which a feasible solution could be found. Third column LS is the maximum number of zones desired by the farmer. Once we have an α that makes ILP to be feasible, then we decrease LS. Fourth column represent the total number of potential zones $|Q|$ (taking into account $w \times l$). Column σ^2 is the objective function of the ILP. Last column is the number of management zones delineated by ILPMZ. Each one of the tests took no more than 0.63 seconds in a personal PC.

Table 2.5: ILPMZ method applied to the real instance for the pH and P soil properties.

pH						P					
$w \times l$	α	LS	$\|Q\|$	σ^2	Zones	$w \times l$	α	LS	$\|Q\|$	σ^2	Zones
1×2	0.4	42	441	0.01	4	1×2	0.5	42	441	6.21	3
1×2	0.4	20	441	0.01	4	1×2	0.5	20	441	6.21	3
1×2	0.4	4	441	0.01	4	1×2	0.5	3	441	6.21	3
1×2	0.4	3	441	–	–	1×2	0.5	2	441	–	–
2×1	0.4	42	420	0.05	9	2×1	0.3	42	420	102.6	3
2×1	0.4	20	420	0.05	9	2×1	0.3	20	420	102.6	3
2×1	0.4	8	420	0.06	8	2×1	0.3	3	420	102.6	3
2×1	0.4	7	420	–	–	2×1	0.3	2	420	–	–
2×2	0.2	42	315	0.07	5	2×2	0.3	42	315	102.6	3
2×2	0.2	20	315	0.07	5	2×2	0.3	20	315	102.6	3
2×2	0.2	5	315	0.07	5	2×2	0.3	3	315	102.6	3
2×2	0.2	4	315	–	4	2×2	0.3	2	315	–	–
3×3	0.1	42	150	0.02	2	3×3	0.2	42	150	53.07	3
3×3	0.1	20	150	0.02	2	3×3	0.2	20	150	53.07	3
3×3	0.1	2	150	0.02	2	3×3	0.2	3	150	53.07	3
3×3	0.1	1	150	–	–	3×3	0.2	2	150	–	–

From Tables 2.4 and 2.5 we remark that if the minimal size of a zone drastically affects the delineation of the zones. Indeed, if $w \times l$ is 3×3 for the Real Instance, then it would be difficult for ILPMZ to find homogeneous zones for this field. The option is either to reduce this minimum size of the zones or reduce the wished homogeneity within the zones. Experts suggest that best delineations of management zones are achieved with a homogeneity parameter α greater than 0.5. Therefore, it would be better to reduce the minimum size of the wished zones than reducing parameter α. Notice that the number of zones obtained by ILPMZ (last column) is not always equal to LS which is the maximum number of zones that a farmer would want.

Table 2.6 presents the solution for the Large instance set. The main purpose is to test the ILPMZ method on intances that have different sizes. The columns of this table are as follows: number of the instance (first column), number of soil samples in the width ($WidthF$) and length ($LengthF$) of the field, total number of soil samples (fourth column), total number of quarters generated (fifth column), total time in minutes for generating all the quarters ("Q time" column), loading time of the model in seconds ("Loading time" column), and B&B time in seconds (last column). For all instances the B&B is totally executed until the optimal solution is found. We

Table 2.6: ILPMZ applied to the set of large instances.

Instance	$WidthF$	$LengthF$	# Samples	$\|Q\|$	Q time	Loading time	B&B time
1	6	7	42	588	0.03	0.63	0.33
2	10	10	100	3025	0.40	1.26	0.46
3	15	10	150	6600	1.33	4.71	2.67
4	15	15	225	14400	4.25	8.65	3.40
5	15	20	300	25200	18.30	31.15	19.95
6	20	20	400	44100	22.78	48.11	22.10
7	20	25	500	68250	54.40	92.51	41.26
8	25	25	625	105625	84.88	227.35	121.27
9	25	30	750	151125	174.05	415.82	223.95
10	30	30	900	216225	250.96	1852.54	193.66

can say that the ILP model is very efficient since for the large instances it does not take more than four minutes. The main issue is the loading time of the model which contains all the information about the quarters of the field. Nevertheless, since this ILPMZ is only needed at the beginning of the planning period, the total time is reasonable. Notice that instance 9 is smaller in size than 10 but it is harder to solve by the branch and bound (although, the magnitude order remains the same). This behavior is frequent for integer linear programmings since an instance is not only harder because of its size but also because of the parameters of the variables and resources.

2.1.4 Conclusions of Section

Dividing the field into site-specific management zones is an interesting manner to face within-field variability. Classical zoning methods, based on soil properties, have a disadvantage: the zones have oval shapes which are not practical for the fertilization machinery which is most of the times towed by a tractor.

In this work, we present a new zoning method that optimally delineates rectangular homogeneous management zones, using relative variance to guarantee the homogeneity within the zones. Our ILPMZ method based on an integer linear programming model can be efficiently solved even for a large fields.

Experimental results show that the ILPMZ method is efficient and practical so it could be embedded in any decision system.

Chapter 2

Agricultural Planning

2.2 A Crop Planning and Real Time Irrigation Method Based on Site-specific Management Zones and Linear Programming

This research produced the article: Cid-Garcia, N. M., Bravo-Lozano, A. G., & Rios-Solis, Y. A. (2014). "A crop planning and real-time irrigation method based on site-specific management zones and linear programmng". *Computers and electronics in agriculture*, 107, 20-28.

Abstract

The spatial variability of the physical and chemical soil properties directly affects the agricultural production planning. In this study, we present two mathematical models that consider the physical and chemical site-specific management zones within the parcels. The first model is for the crop planning problem. At the beginning of the production cycle, it assesses the chemical and physical management zones to determine the optimal crop pattern for maximizing the farmer's expected profit. The second model is a real-time irrigation method that takes the solution of the crop planning problem as input. Then, at each irrigation period, it considers the physical management zones and the humidity level of each parcel to determine in real-time the optimal amount of water for each crop irrigation. This is especially important in regions where droughts are frequent. We empirically show that our methodologies are efficient on instances based on real data.

2.2.1 Introduction

At the beginning of the production cycle, farmers must decide which crops they are going to plant in each one of their parcels. It is one of the most complex decisions since it is impacted by the spatial variability of the physical and chemical soil properties within each parcel. Soil variability directly affects crop pattern choice because it has a great impact on water balance, nutrient dynamics, and response to the application of inputs (seeds and fertilizers). For example, if a parcel is rich in nitrogen and phosphorus then planting maize or tomatoes could lead to higher yields without using much fertilizer. However, if half of the parcel is poor in nitrogen then the farmers may decide to fertilize only half of the parcel instead of fertilizing it entirely.

Moreover, the crop planning decisions must also consider several factors as the expected prices of crops yielded, the expected amount of available water for the production cycle, the cost of irrigating a parcel (some parcels may be far away and their irrigation may consume more electricity), the phenological stages of the crops, the number of hectares in each parcel, the expected amount of resources, and so on.

In this paper we propose a methodology to help the farmers to consider all these attributes to determine the optimal crop pattern to maximize the farmer's profit; we call this problem as the *Crop Planning Problem* (CPP). For this, the farmers must first delineate chemical and physical management zones within their parcels. A management zone is a sub-region of a parcel, which expresses a relatively homogeneous combination of yield limiting factors, for which a single rate of a specific crop input is appropriate (Ortega y Santibáñez, 2007; Al-Karadsheh *et al.*, 2002; McBratney *et al.*, 2005). Managing fields as zones helps to reduce input costs (Moore y Wolcott, 2000; Doerge, 1999). To facilitate the use of agricultural machinery, these management zones should have a rectangular shape. Farmers can then use the ILPMZ method proposed by Cid-Garcia *et al.* (2013) that efficiently creates rectangular shape homogeneous management zones.

The main chemical soil properties that a farmer might consider are: pH, organic matter rate, phosphorus, and nitrogen[4]. The main physical soil properties are: field capacity or permanent wilting point, which is based on the texture, structure, and porosity of the field, and influences the movement and retention of water; air and solutes in the soil, which impact plant growth and organism activity. All these properties may be altered by management practices that are usually expensive. It is therefore imperative to consider which zones need theses practices. Figure 2.7 shows an example of the chemical and physical management zones (at the right hand side of the figure) of three parcels (at the left hand side of the figure) determined by the ILPMZ method. Notice that the number and size of the physical and chemical zones are different for each parcel. In Chile, Ortega *et al.* (2002) have demonstrated that the use of management zones based on soil properties, produce a positive impact on vineyards and traditional crops.

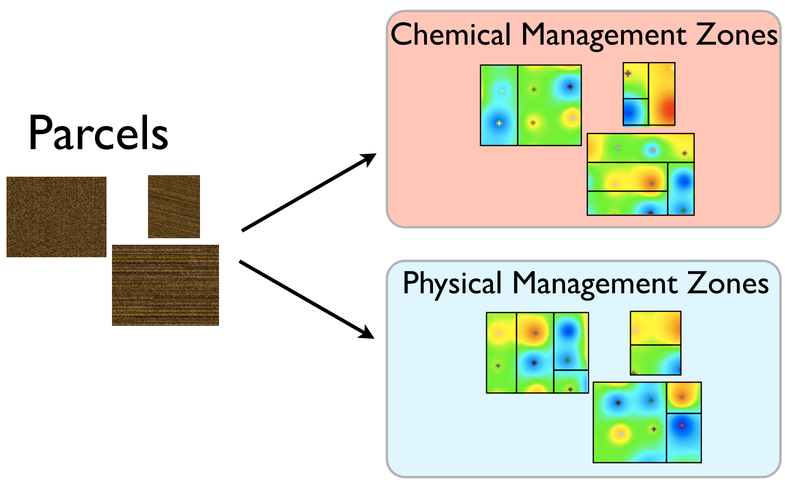


Figure 2.7: Chemical and physical management zones of three parcels.

An important contribution of this study is to solve the CPP that considers the chemical and physical management zones of the parcels to decide the optimal crop pattern. Then at the beginning of each irrigation period, in order to decide the optimal water irrigation level for each parcel, farmers need to consider a number of

[4]The properties suitable for a specific crop in Mexico are available in SAGARPA or in INIFAP. In other countries they can be found in FAO.

factors: the physical management zones of their parcels; the crops planted in each one; the phenological stages of the crops; the real-time humidity of the soil determined by sensors, and the available water. These decisions are crucial, especially in arid and semiarid regions, since farmers seek to maximize total profit. We call this the *Real-Time Irrigation Problem* (RTIP), and it is also a key contribution of this paper.

Control strategies that locally modify the irrigation volumes must be adaptive in order to use scarce resources more effectively (McCarthy *et al.*, 2008; Smith *et al.*, 2009). The linear programming model we present for RTIP considers the yield response to water shortages, the climate conditions, the evapotranspiration of the crops in a specific geographical region, and the phenological stage of the crops. In this manner, with RTIP we use only the necessary amount of water or, if there is a water shortage, RTIP determines which parcels to fully irrigate, which ones must be under deficit irrigation, and which ones will be lost, to optimize the allocation of water resources.

This article is structured as follows. In the rest of this section we make a brief literature review. Then in Section 2.2.2, CPP is presented together with its mathematical model. The output of this problem serves as input for the model of RTIP (Section 2.2.2). In Section 2.2.3, we show that our new approaches are efficient with instances based on real data. Section 2.2.4 concludes this work. A summary of the mathematical notation can be found in C and D.

Related literature

Sarker *et al.* (1997) propose a linear programming model to solve the CPP that considers land type, alternative crops, crop patterns, input requirement, investment, and output. Later, Sarker y Ray (2009) formulate a CPP as a multiobjective optimization model. A major result of their work is that their algorithm delivers superior solutions to the nonlinear version.

Mainuddin *et al.* (1997) propose a crop planning model for an existing groundwater irrigation. Adeyemo y Otieno (2010) present an evolutionary algorithm to solve the multiobjective crop planning model: minimize the total irrigation water, to maximize both the total net income from farming and the total agricultural output. Itoh *et al.* (2003) consider crop planning under uncertainty.

In Casadesús *et al.* (2012); Xu *et al.* (2011); Hedley y Yule (2009); Hassanli *et al.* (2009) the authors propose heuristics for scheduling irrigation plans according to weather conditions, crop development, and other factors. A work that is closely related to ours is Alminana *et al.* (2010), where they present models and algorithms to determine water irrigation scheduling by taking into account the irrigation network topology, water volume, technical conditions, and logistical operations. McCarthy *et al.* (2013) review the existing literature of advanced control process in irrigation.

Few works deal with both crop and irrigation problems as we do in this research. Ortega Álvarez *et al.* (2004) propose a non-linear model solved by genetic algorithms to identify production plans, and water irrigation management strategies. They estimate crop yield, production and gross margin as a function of the irrigation depth. Sahoo *et al.* (2006) propose fuzzy multiobjective linear programming models for land-water-crop system planning. Reddy y Kumar (2008) present a multiobjective approach for the optimal crop pattern and operation policies for a multi-crop irrigation reservoir system.

As we see from the literature review, there are several studies concerning CPP or RTIP which use real-time information in their methodology from a great diversify of technological devices, as humidity sensors, as we do in this work. However, few approaches use this real-time information to feed mathematical models and execute them in real-time too. This is an important characteristic because we can give to farmer a response in real-time from the current conditions (weather, water, seeds, etc.). Moreover, for both mathematical models, CPP and RTIP, we propose exact solutions (optimal solutions) in an efficient period of time, instead of approximations. Many of the previous methods consider management zones but, to the best of our knowledge, this is the first approach which uses rectangular and homogeneous physical and chemical management zones with mathematical programming.

2.2.2 Materials and Methods

In this section we present our methodologies for the crop planning problem (Section 2.2.2) and real-time irrigation problem (Section 2.2.2).

We use the term field for the whole land that can be irrigated by a water well or a dam. This field is made up of different parcels. In each parcel j a single crop i is going to be planted. Each parcel j is subdivided into z management zones (see Figure 2.8). All the management zones z of a parcel j must be planted with the same crop i. This is by farmer requirements, as they prefer not to have more than one crop in the same parcel. We could easily modify the model to plant a different crop in each management zone if necessary.

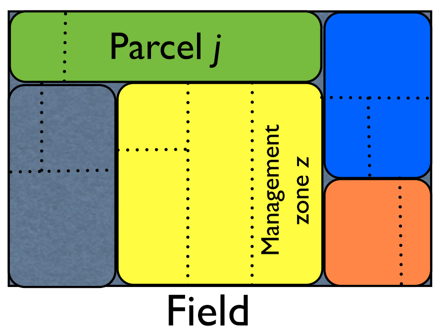


Figure 2.8: Terms used in this article: field, parcel, and management zone.

Crop Planning Problem, CPP

The CPP problem decides which crops i to plant in different parcels j considering the chemical and physical management zones of these parcels. The farmer's objective is to maximize the total expected profit given a limited expected water availability for the production cycle.

We propose a mixed integer linear programming (MIP) to solve CPP (see Section A.5). The parameters of the MIP model for CPP are as follows. Let I be the set of the different crops a farmer could plant, J the set of parcels of the farmer's field, $Z^{Ch}(j)$ the set of chemical management zones within parcel j and $Z^{Ph}(j)$ the set of physical management zones within parcel j (see Figure 2.7) .

Below are the costs and benefits needed by CPP.

- G_i is the expected benefit of selling a tonne (tn) of crop i at the end of the production cycle. This value varies each year and changes with respect to the market place where the farmer sells.
- $Cirr_{jz}$ cost of irrigate one cubic meter (m^3) of water in parcel j and physical zone $z \in Z^{Ph}(j)$. Indeed, there could be a higher cost of irrigating a distant parcel or a zone with poor physical properties.
- $Cseed_i$ is the cost of buying a kilogram (kg) of seeds of crop i.
- $Cplant_{ijz}$ is the cost of planting an hectare (ha) of crop i in parcel j and chemical zone $z \in Z^{Ch}(j)$.

The information known to the farmer about the field and crop characteristics is as follows.

- ha_j is the number of ha of parcel j.
- hac_{jz} and hap_{jz} correspond to the number of ha in chemical management zone $z \in Z^{Ch}(j)$ of parcel j and physical management zone $z \in Z^{Ph}(j)$ of parcel j, respectively.
- $Iseed_i$ is the quantity of seeds in kg of crop i in the farmer's stock.
- $Seed_i$ is the quantity of seeds in kg needed to plant a ha of crop i.
- Y_i is the expected yield in tn by ha of crop i at the end of the production cycle.

- D_i is the demand in *tn* of crop $i \in I_0 \subsetneq I$ where I_0 is a subset of the crop set I. This is to model situations where a farmer is paid in advance for some yields of a specific crop.

The MIP model for CPP has two parameters related to the water:

- W is the expected total amount of water in m^3 for the whole production cycle.
- W_{ijz} is the expected amount of water in m^3 needed for irrigating a *ha* planted with crop i in parcel j and physical management zone $z \in Z^{Ph}(j)$.

To obtain parameter W_{ijz} it is first necessary to calculate the total expected amount of water, in m^3 by *ha*, consumed by crop i planted in parcel j during the whole production cycle. This information can be obtained from historic data or by using the Penman-Monteith equation (Allen *et al.*, 2006):

$$ETc_{ij}^{v} = ETo^{v} \cdot Kc_{ij}^{v} \tag{2.9}$$

where ETc_{ij}^{v} is the crop evapotranspiration that represents the amount of water (in *mm*) required by the crop i at phenological stage v for parcel j, ETo^{v} is the reference crop evapotranspiration that expresses the evaporating power of the atmosphere (in *mm*) during phenological stage v. Kc_{ij}^{v} is known as the crop coefficient and its value change from crop to crop, phenological stage of the crop v, and geographic location j.

However, each physical management zone $z \in Z^{Ph}(j)$ has different amounts of stored water (rain or previous irrigation). Therefore, the total expected amount of water consumed by crop i in parcel j and physical management zone $z \in Z^{Ph}(j)$ throughout the production cycle (W_{ijz}) is equal to the sum of the total expected amount of water required by crop i in its vegetative cycle v ($\sum_v ETc_{ij}^{v}$) minus the sum of the total stored water in parcel j in physical zone $z \in Z^{Ph}(j)$ in the vegetative cycle v ($\sum_v SW_{jz}^{v}$), see equation (2.10). The value of $\sum_v SW_{jz}^{v}$ can be obtained from moisture sensors (in m^3 by *ha*) but the units of $\sum_v ETc_{ij}^{v}$ are in *mm*. Therefore, to get the amount of water in m^3 by ha, we divide $\sum_v ETc_{ij}^v$ by 1,000 to convert its value to meters, and finally multiply it by 10,000 (surface of a ha) which is equal to $10 \cdot \sum_v ETc_{ij}^v$:

$$W_{ijz} = \left(10 \cdot \sum_v ETc_{ij}^v\right) - \sum_v SW_{jz}^v. \tag{2.10}$$

Notice that it is necessary to know the duration of the crop production cycle (days), the duration of each one of the phenological stages, and the sowing date (month) to get the value of ETc_{ij}^v.

The main variables of the CPP integer linear programming model are $\boldsymbol{x_{ij}}$. These are equal to one if crop i is planted in parcel j, zero otherwise.

CPP requires another set of decision variable $\boldsymbol{s_i}$ related to the amount of seeds that the farmer could buy of crop i in kg, $i \in I$. The ILP model for CPP is as follows:

$$\max \sum_{i\in I}\sum_{j\in J}\Bigg[\boldsymbol{x_{ij}} \cdot G_i \cdot Y_i \cdot ha_j - \boldsymbol{x_{ij}} \sum_{z\in Z^{Ch}(j)} Cplant_{ijz} \cdot hac_{jz}$$

$$- \boldsymbol{x_{ij}} \sum_{z\in Z^{Ph}(j)} Cirr_{jz} \cdot W_{ijz} \cdot hap_{jz}\Bigg] - \sum_{i\in I} \boldsymbol{s_i} \cdot Cseed_i \tag{2.11}$$

subject to: (2.12)

$$\sum_{i\in I} \boldsymbol{x_{ij}} \leq 1 \qquad j \in J \tag{2.13}$$

$$\sum_{j\in J}\sum_{z\in Z^{Ch}(j)} Y_i \cdot hac_{jz} \cdot \boldsymbol{x_{ij}} \geq D_i \qquad i \in I_0 \tag{2.14}$$

$$\sum_{j\in J}\sum_{z\in Z^{Ch}(j)} Seed_i \cdot hac_{jz} \cdot \boldsymbol{x_{ij}} \leq Iseed_i + \boldsymbol{s_i} \qquad i \in I \tag{2.15}$$

$$\sum_{i\in I}\sum_{j\in J}\left[\sum_{z\in Z^{Ph}(j)} W_{ijz} \cdot hap_{jz}\right] \boldsymbol{x_{ij}} \leq W \tag{2.16}$$

$$\boldsymbol{s_i} \geq 0, \boldsymbol{x_{ij}} \in \{0,1\} \qquad i \in I, j \in J$$

The first term in the objective function (2.11) represents the benefits of selling the expected yields of each crop planted in each parcel. The second term corresponds to the cost of planting the crops in each chemical management zone of each parcel (it includes fertilizers and pesticides: approximately between 20% and 30% of the production costs). The third term is about the irrigation costs per parcel and per physical management zone of the parcel. Finally, we have the cost of buying seeds. We can say that (2.11) maximizes the total expected profit. Restrictions (2.13) ensure a unique assignment of crop i to each parcel j. Restrictions (2.14) specify that there are some crops i that must be planted in order to satisfy a certain demand D_i negotiated by the farmer before planting. Notice this is only for few crops in I_0. Restrictions (2.15) determine the amount of seeds needed and also the amount of seeds that must be bought. Restrictions (2.16) establish that the expected amount of water needed to irrigate the crops must be sufficient for the whole production cycle.

The CPP is a NP-hard problem since a reduction of the Knapsack problem (Garey y Johnson, 1979) is straight forward. However, in Section 2.2.3 we show that this model is elegant enough to optimally solve real size instances by a branch-and-bound algorithm in less than one second.

Real-Time Irrigation Problem, RTIP

The production cycle is divided into irrigation periods. At the beginning of each irrigation period the farmer must take some decisions (specially if there is a drought): which crops need to be irrigated optimally, which crops should be in deficit irrigation, and which crops should not be irrigated (up until the point of allowing a crop to die) in order to maximize the profit at the end of the production cycle. The aim of RTIP is to determine the amount of water to be supplied to crops at each irrigation period, according to their current water requirements, the climate, physical management zones, phenological stage of the crops, the amount of water available at the irrigation period, and the moisture content stored on the soil (humidity sensors are placed in each one of the management zones of the parcels).

Before presenting the linear programming model (LP) for RTIP (see Section A.2), we discuss the main parameters and matters concerning about the water requirements and humidity sensors.

Yield response to water

The crop water production function we use is based on the one given by the FAO (Doorenbos *et al.*, 1979):

$$\frac{\boldsymbol{Ya^p_{ijz}}}{Ya^{p-1}_{ijz}} = 1 - Ky^p_i \left(1 - \frac{ETa^p_{ijz}}{W^p_{ijz} \cdot ha_{jz}}\right). \tag{2.17}$$

We can specify each one of the factors of this equation, such that the only unknown parameter is $\boldsymbol{Ya^p_{ijz}}$, which corresponds to the real yields of crop i in parcel j, in physical zone $z \in Z^{Ph}(J)$ at period p, i.e., the yield reached by crop at the current irrigation period. To obtain this value, is necessary to know the yield of crop in the last irrigation period. This parameter is specified by Ya^{p-1}_{ijz}, which represents the yield reached by crop i, in parcel j, in physical zone $z \in Z^{Ph}(J)$ in the last irrigation period p. Only when p is the first irrigation period, Ya^{p-1}_{ijz} is the harvested yield of crop i under an optimal growing environment, i.e., the crop yield is not limited by water, nutrients, pests, nor diseases.

W^p_{ijz} represents the maximum water requirements of a crop i in parcel j and physical zone $z \in Z^{Ph}(J)$ at period p. This value can be computed by equation (2.10), using only the vegetative stages up to the current irrigation period p.

ETa^p_{ijz} represents the current amount of water in crop i of parcel j in physical zone $z \in Z^{Ph}(J)$ at period p. This parameter corresponds to stored water in the soil, SW^p_{jz}, by rain or previous irrigation periods, plus the amount of water supplied in the current irrigation period, $\boldsymbol{w^p_{jz}}$. The water level stored in each one of the physical management zones of the parcels, before the current irrigation, is determined by a moisture sensor in real-time (in this research we used WATERMARK 200SS-V. The sensors are located in each physical management zone of the field, next to the roots of the crops, and with their data it is easy to determinate the amount of stored water, in m^3 by ha.

Finally, response factor Ky^p_i represents the relationship between water and yield for crop i at irrigation period p. This value is crop specific and has different values at each irrigation period p. In combination with the previous parameters, the water production function (2.17) can be computed for each crop (more information in Geerts y Raes (2009) or in Barnes *et al.* (2000)).

Figure 2.9 is a linear approximation of the expected yield of a crop with deficit irrigation during its production cycle. The expected yield decreases with respect to the crop coefficient, its vegetative stage, and the available water. For example, in the firsts periods P1 and P2, the crop is optimally irrigated, therefore the expected yield is still 100%. Nevertheless, in P3 and P5 the crop is not optimally irrigated (there are water shortages) so the expected yield decreases and it will never recovered to 100% even if in later periods the crop is watered to optimal level. In this case, the yield reached at the end of the production cycle will be 70% of the expected yield at the beginning of the production cycle.

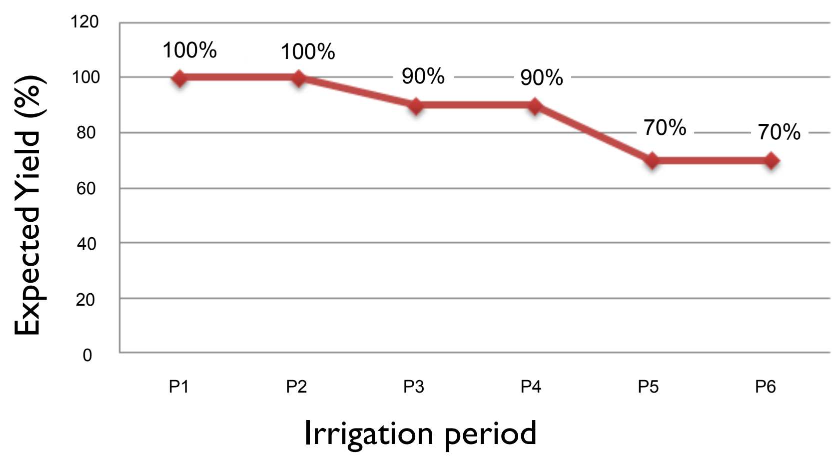


Figure 2.9: Expected yield behavior of a crop with deficit irrigation.

Mathematical model for RTIP

At the beginning of each irrigation period, the farmer knows the volume of available water. This volume may not be the expected one, therefore some crops will not be optimally irrigated. With the RTIP the farmer would know which crops to irrigate in order to obtain the best profit from the total crop yields. The additional parameters needed to formulate the LP are listed below:

- G_i^p expected benefit of selling a tn of crop i at the end of the production cycle given that we are at period p. Indeed, this value is known at the beginning of each irrigation period p but may vary from period to period.

- $\gamma(j) = i$ is a function that indicates that crop i is sown in parcel j (this is obtained from the solution of CPP).

- W_{ijz}^p corresponds to the real amount of water in m^3/ha that crop i needs in physical management zone $z \in Z^{Ph}(J)$ of parcel j at period p. It is trivial to compute which vegetative stages v corresponds to crop i at period p, so we omit cumbersome notation.

- SW_{jz}^p is the amount of water in m^3/ha that already exists in physical management zone $z \in Z^{Ph}(J)$ of parcel j at the beginning of period p. This information is retrieved from the humidity sensors in m^3/ha.

- W^p represents the amount of available water in m^3 at period p.

- Ky_i^p is the yield response factor of crop i at period p.

- Ya_{ijz}^{p-1} is the crop yield in tn/ha in physical management zone $z \in Z^{Ph}(J)$ of parcel j computed at the previous irrigation period p.

The variables of the LP model are $\boldsymbol{w_{jz}^p}$ which corresponds to the amount of irrigated water in m^3 in physical management zone z of parcel j at period p, and $\boldsymbol{Ya_{ijz}^p}$ which corresponds to the current expected total crop yield in tn/ha achieved in management zone z of parcel j computed after irrigation period p.

For each irrigation period p of the production cycle, the following LP must be solved:

$$\max \quad \sum_{i \in I} G_i^p \left(\sum_{\{j|\gamma(j)=i\}} \sum_{z \in Z^{Ph}(j)} hap_{jz} \cdot \boldsymbol{Ya_{ijz}^p} \right)$$

subject to: (2.18)

$$\boldsymbol{Ya_{ijz}^p} = Ya_{ijz}^{p-1} \left(1 - Ky_{\gamma(j)}^p \left(1 - \frac{\boldsymbol{w_{jz}^p} + SW_{jz}^p \cdot hap_{jz}}{W_{ijz}^p \cdot hap_{jz}} \right) \right)$$

$$i \in I, \{j \mid \gamma(j) = i\}, z \in Z^{Ph}(j) \quad (2.19)$$

$$\sum_{j \in J} \sum_{z \in Z^{Ph}(j)} \boldsymbol{w^p_{jz}} \leq W^p \quad (2.20)$$

$$\boldsymbol{w^p_{jz}} + (SW^p_{jz} \cdot hap_{jz}) \leq W^p_{ijz} \cdot hap_{jz}$$

$$i \in I, \{j \mid \gamma(j) = i\}, z \in Z^{Ph}(j) \quad (2.21)$$

$$\sum_{\{j \mid \gamma(j)=i\}} \sum_{z \in Z^{Ph}(j)} \boldsymbol{Ya^p_{ijz}} \cdot hap^p_{jz} \geq D_i \qquad i \in I_0 \quad (2.22)$$

$$\boldsymbol{w^p_{jz}}, \boldsymbol{Ya^p_{ijz}} \geq 0 \qquad j \in J, z \in Z^{Ph}(j)$$

The objective function reflects the fact that we are maximizing the total yield revenues: the price per tn times the total number of ha times the current crop yield (tn/ha), for all crops planted in the different parcels of the field. Restrictions (2.19) correspond to the yield response to the water function of Eq.(2.17). Restriction (2.20) limits the amount of water that the farmer can use for this period. Restrictions (2.21) indicate that the real amount of water ETa^p_{ijz} cannot exceed the maximum (or optimal) amount of water W^p_{ijz} required by crop i in parcel j in physical zone z at irrigation period p. Restrictions (2.22) determine that the current yield computed at period p of crop $i \in I_0$. It must satisfy the demand negotiated beforehand by the farmer.

RTIP's linear programming has continuous variables, and therefore belongs to the polynomial complexity class (Garey y Johnson, 1979). This means that it is possible to solve this problem efficiently, as we show in Section 2.2.3.

2.2.3 Experimental Results

In this section we empirically show that the models for CPP (Section 2.2.3) and RTIP (Section 2.2.3) are valid and efficient for problems of realistic magnitude.

Mathematical models for CPP and RTIP were solved using the General Algebraic Modeling System (GAMS) with the optimizer CPLEX 12.2 of IBM (with default options, except that the optimal criterion fixed at 0). The experiments were executed on a Virtual Machine with Windows 7 fitted with 1 GB of RAM and a processor Intel Core 2 Duo of 3.06 GHz running on a iMac equipped with the same processor and 4 GB of RAM. Optimal solutions were reached in less than 1 second for all instances of CPP or RTIP.

Crop Planning Problem

In this section we assesses the CPP model performance through four groups of instances that differ in the number of parcels. *Small* instances have 10 parcels, *Medium* have 50, *Large* have 100, and *ELarge* have between 1,000 and 20,000. For each one of these sets, 10 different instances were generated based on real data. Each instance has a set I of 19 possible crops to sow except for the *ELarge* instances that have 114. The real sizes of the fields in Mexico are on average between the *Small* and *Medium* instances. However, a few fields exist in Mexico with the size of the *Large* instances. The only purpose of the *ELarge* instances is to assess the scalability of our approach.

The data relates to 19 crops grown during the spring-summer production cycle in the state of Michoacán, Mexico in 2008, is presented in Table 2.7. The first and second columns refers to the identification number (ID) and name of the crop i, respectively. The third column shows the expected yield Y_i of the crop i in tn/ha at the end of the production cycle. This parameter is updated yearly by SIACON[5], INFOSIAP[6], or SNIIM[7]. It can be obtained for each state, crop, and type of irrigation system. The fourth column is the amount of seeds $Seed_i$ in $unit/ha$ needed to sown crop i. The term *unit* represents *kg*, plants, or packages of each crop seeds needed. Sowing costs $Cplant_i$ in $\$/ha$ (obtained from INFOSIAP) and seed costs $Cseed_i$ in $\$/unit$ of crop i (obtained from SNIIM or SIACON) are presented in the fifth and sixth column, respectively. The expected benefit G_i of selling a *tn* of the crop i at the end of the production cycle is showed in the seventh column. In Mexico this value

[5]The agri-food consultation information system (SIACON) about agriculture, livestock, and fisheries; data are averages of last year.

[6]Agri-food and fisheries information system (INFOSIAP); data are averages of last year.

[7]National information system and market integration (SNIIM) of the Mexican ministry of economy.

can be obtained from SIACON or from SNIIM. For other countries the information can be obtained from FAOSTAT[8]. We assume that the demand D_i to be satisfied by the farmer is equal to zero for all crops, and the stock of seeds $Iseed_i$, is also 0.

Table 2.7: Crop data from spring-summer cycle in the state of Michoacán, Mexico.

ID	**Crop** ($i \in I$)	**Expected Yield** (Y_i) tn/ha	**Seed Amount** ($Seed_i$) $unit/ha$	**Sowing Cost** ($Cplant_i$) $\$/ha$	**Seed Cost** ($Cseed_i$) $\$/unit$	**Expected Benefit** (Gi) $\$/tn$
1	Sesame TCS	0.60	4.00	2,318.00	10.00	13,681.80
2	Sesame TMF	0.50	4.00	8,117.91	10.00	13,681.80
3	Onion BMF	40.60	12,500.00	69,251.08	0.15	3,381.17
4	Green pepper BMF	24.70	12,500.00	106,121.91	0.15	4,923.99
5	Strawberry BMF	20.40	85,228,00	74,543.27	0.11	3,943.97
6	Strawberry GMF	20.40	85,228.00	48,533.07	0.11	3,493.97
7	Corn grain BCF	4.85	25.00	10,273.02	17.10	4,373.49
8	Corn grain BMF	5.38	25.00	10,013.49	17.10	4,373.49
9	Corn grain GCF	4.85	25.00	9,693.02	17.10	4,373.49
10	Corn grain GMF	5.38	25.00	10,668.41	17.10	4,373.49
11	Corn grain TCF	2.41	25.00	10,512.40	17.10	4,373.49
12	Sorghum grain BMF	8.31	3,24.00	12,022.65	1.50	3,491.25
13	Sorghum grain GMF	8.31	3,24.00	7,891.88	1.50	3,491.25
14	Sorghum grain TMF	4.71	3,24.00	6,674.43	1.50	3,491.25
15	Red Tomato BMF	38.10	12,500.00	75,259.74	0.56	2,171.99
16	Red Tomato GMF	38.10	12,500.00	74,440.68	0.56	2,171.99
17	Green Tomato BCF	16.20	12,300.00	50,574.63	0.15	3,416.17
18	Green Tomato BMF	17.80	12,300.00	41,867.90	0.15	3,416.17
19	Green Tomato TCF	2.70	12,300.00	34,056.14	0.15	3,416.17

The maximum number of chemical and physical management zones z per parcel j in all instances is a random number between 1 and 5 (this range has been given by an agricultural expert that corresponds to farmer finding it impracticable to have many zones in a parcel). The number of ha per chemical and physical management

[8]System of the statistics division of the food and agriculture organization (FAO).

zone of each parcel hac_{jz} and hap_{jz} is a random number between 1 and 7. The irrigation costs $Cirr_{jz}$ per physical zone for each one of the parcels j is uniformly chosen between 1.8 and 2.2 (these parameters are based on real data from SIACON or SNIIM). The parameters to compute Eqs. (2.9) and (2.10) were obtained from FAO and INIFAP[9].

Table 2.8 presents the experimental results of *Small*, *Medium*, and *Large* instances for CPP and their comparison with the *traditional method* (TM) most often used in practice (at least in Mexico), which is: plant the crop that generates the maximum benefit such that the same crop cannot be cultivated in a parcel for more than three consecutive cycles. The first column is the instance type. The number of management zones in the field is in the second column, while the field surface (FS) is in the third one. The fourth column is the expected amount of water for the whole production cycle, established by the upper bound allowed by CONAGUA[10]: no more than 6,000 m^3 of water, per cycle, per *ha*. The "CPP" column shows the solution of the ILP model and "TM" column shows the solution of the traditional method (total profit in Mexican pesos MXN \$). The seventh column is the percentage increase on profit by using CPP instead of TM. Finally, the last column is the time (in seconds) that the branch-and-bound algorithm needs to solve the ILP of CPP.

In Table 2.8, we see that farmers can increase their profits in at least 5.27% using CPP, as compared to the traditional method. Considering that the total expected benefits are around 520,000 dollars for the *Small* instances, this percentage is substantial for farmers. This profit arises from CPP's site-specific application of nutrients in each management zone of the parcel. Moreover, with the traditional method, a farmer cannot guarantee that the expected water will be sufficient while in CPP we have already taken it into account.

[9]The Mexican national institute for forestry, agriculture and livestock (INIFAP).
[10]The Mexican national water commission (CONAGUA).

Table 2.8: Experimental results of Crop Planning Problem (CPP) versus the Traditional Method (TM).

Instance	Zones	FS	Water	CPP	TM	Increase	Time
		ha	m^3	$	$	%	*s*
Small_1	38	150	900,000	8,771,266.70	8,191,359.79	6.61	0.171
Small_2	22	81	486,000	4,791,148.75	4,424,095.86	7.66	0.163
Small_3	21	82	492,000	4,846,297.81	4,590,747.87	5.27	0.168
Small_4	24	85	510,000	4,995,785.01	4,641,084.74	7.09	0.151
Small_5	35	155	930,000	9,094,239.54	8,449,174.77	7.09	0.162
Small_6	44	184	1,104,000	10,775,424.29	10,035,631.24	6.86	0.168
Small_7	39	177	1,062,000	10,417,072.47	9,678,911.01	7.08	0.156
Small_8	29	109	654,000	6,407,156.99	5,945,176.13	7.21	0.166
Small_9	31	148	888,000	8,725,180.10	8,084,161.63	7.34	0.173
Small_10	38	157	942,000	9,202,489.92	8,569,720.69	6.87	0.174
Medium_1	142	559	3,354,000	32,810,677.55	30,269,585.22	7.74	0.217
Medium_2	166	605	3,630,000	35,594,702.11	30,269,585.22	8.21	0.169
Medium_3	148	603	3,618,000	35,414,119.28	32,573,101.79	8.02	0.152
Medium_4	155	630	3,780,000	36,912,518.04	34,050,041.84	7.75	0.181
Medium_5	149	602	3,612,000	35,450,662.58	32,554,719.11	8.16	0.166
Medium_6	147	577	3,462,000	33,945,798.52	31,254,250.19	7.93	0.176
Medium_7	165	684	4,104,000	40,201,116.23	36,976,538.02	8.02	0.164
Medium_8	151	617	3,702,000	36,354,771.08	33,388,917.28	8.15	0.176
Medium_9	149	574	3,444,000	33,741,334.80	31,048,444.68	7.98	0.181
Medium_10	161	665	3,990,000	39,056,761.62	35,926,757.51	8.01	0.224
Large_1	283	1,107	6,642,000	65,025,451.21	59,920,420.35	7.85	0.177
Large_2	301	1,186	7,116,000	69,830,754.70	64,307,858.00	7.90	0.186
Large_3	316	1,250	7,500,000	73,526,772.16	67,701,296.51	7.92	0.385
Large_4	296	1,153	6,918,000	67,693,577.61	62,382,047.15	7.84	0.21
Large_5	318	1,298	7,788,000	76,199,527.44	70,347,748.19	7.67	0.182
Large_6	318	1,269	7,614,000	74,489,320.40	68,663,542.51	7.82	0.253
Large_7	279	1,152	6,912,000	67,614,304.96	62,327,357.54	7.81	0.185
Large_8	289	1,106	6,636,000	64,939,139.48	59,807,787.54	7.90	0.331
Large_9	301	1,124	6,744,000	66,004,727.06	60,896,244.00	7.73	0.186
Large_10	301	1,225	7,350,000	71,915,787.31	66,265,903.94	7.85	0.186

We tested *ELarge* instances with CPP. We found that the resolution times are never longer than one second. Therefore, even if we were crop planning for a whole state or region, CPP would be able to offer exact solutions in a reasonable time. We have not compared the solutions of CPP with TM for the *ELarge* instances because not all of their parameters are based on real data.

In Tables 2.9 and 2.10, we modified the instance *Small_2* to observe the model's behavior when the water parameters change and when there is a demand on certain crop's yields. Table 2.9 shows the behavior of CPP when water is progressively reduced, and the farmer has not negotiated production for any crop. The first column indicates the instance label. The second one shows the reduction made to the total amount of water (originally set to 486,000 m^3). The third column is the maximum revenue and last column is the time (in seconds) of the branch-and-bound algorithm takes to find a solution.

Table 2.9: Experimental results of CPP, based on instance *Small_2*, to see the effect on farmer's profit when a progressive reduction of water availability exist. In this case, the farmer has not negotiated production for any crop.

Instance	**Water reduction** %	**Max revenues** $	**Time** *s*
Small_2_1	0	4,791,148.75	0.103
Small_2_2	10	4,791,148.75	0.101
Small_2_3	20	4,791,148.75	0.106
Small_2_4	30	4,791,148.75	0.111
Small_2_5	40	4,612,531.00	0.081
Small_2_6	50	3,904,648.80	0.020
Small_2_7	60	3,140,433.65	0.060
Small_2_8	70	2,371,206.90	0.081
Small_2_9	80	1,603,112.41	0.035
Small_2_10	90	771,504.10	0.027
Small_2_11	100	0.00	0.000

In Table 2.9 we observe that even if we reduce the amount of water by 30% with respect to water available at the beginning of the production cycle, and we do not have demand for any crop, the solution remains the same as in Table 2.8. It means that the optimal assignment of crops to parcels would not use all the available water. But, if we reduce the amount of water by more than or equal to 40% the water available at the beginning of the production cycle, we can see the impact on the farmer's profit.

Table 2.10 shows the behavior of CPP when water is progressively reduced and the farmer has negotiated production for a crop. In this case the farmer has negotiated 30 *tn* for crops 1, 4, 7, and 16 (sesame, green pepper, corn, and red tomato), presented in Table 2.7. The first column indicates the label of the instance. The second one shows the reduction made to the total amount of water (originally set the by CONAGUA to 486,000 m^3). Third column is the maximum revenue and last column is the time (seconds) of the branch-and-bound algorithm.

Table 2.10: Experimental results of CPP, based on instance *Small_2*, to see the effect on the farmer's profit when a progressive reduction of water availability exist. In this case, the farmer has negotiated 30 *tn* of production for sesame, green pepper, corn, and red tomato.

Instance	**Water reduction** %	**Max revenues** $	**Time** *s*
Small_2_12	0	877,690.95	0.105
Small_2_13	10	877,690.95	0.119
Small_2_14	20	877,690.95	0.111
Small_2_15	30	-125,012.11	0.081
Small_2_16	40	-125,012.11	0.061
Small_2_17	50	No solution	0.031

Instance *Small_2_12* shows the sizeable impact that previously negotiated demands can have on the revenues. Notice that the second and third rows show that the total amount of water is not the key constraint, since it is reduced by 10% and by 20% without any impact on revenues. However, the fourth and fifth rows show that reducing the amount of water between 30% and 40% where there is previously negotiated demand, may result in negative benefits. This is due several factors: the costs of producing the crops are high (production costs, seed costs, irrigation costs) or, crops with previously negotiated demand are very expensive to produce and do not offer enough benefits. This negative value shows it is not profitable for the farmer to produce these crops under these conditions. The last row indicates that if available water is reduced by greater than or equal to 50% versus the water at the beginning of the production cycle, the model indicates there is no solution. This is because there is neither enough water nor resources to produce any crops.

Real-Time Irrigation Problem

The results of CPP are used as parameters to validate RTIP's performance. So, in this stage it is already known which crop i has been sown in parcel j. Now the farmer must decide the amount of water to be supplied to each crop during each irrigation period p to maintain its total expected yields at their maximum.

To illustrate how RTIP works, we will use the solution of instance *Small_2_12*. In this instance the solution of CPP is to sow sesame TCS in parcels 1, 3, 6, 7, 8, and 10; to sow onion TMF in parcel 2; to sow corn grain BCF in parcel 4; to sow red tomato GMF in parcel 5; and to sow green pepper BMF in parcel 9, to obtain an expected income of $877,690.95 at the end of the production cycle.

Table 2.11 presents the parameters needed for RTIP in instance *Small_2_12* at irrigation period 1. The first row is the yield response factor $Ky^p_{\gamma(j)}$ of crop $\gamma(j)$ sown in parcel j at irrigation period 1. This parameter is obtained from FAO. The last row is the maximum expected yield Ya^{p-1}_{ijz} in tn/ha in each physical management zone $z \in Z^{Ph}(j)$ of parcel j (this value is provided by INIFAP, SIAP, SNIIM, or FAO). This value is only used during the first irrigation period, in future periods this value is updated with the model solution, Ya^p_{ijz}.

In instance *Small_2_12* we evaluate six irrigation periods. We simulate a drought during periods 3-5. The values of the existing water in the soil SW^p_{jz},

Table 2.11: Parameters of RTIP at irrigation period 1 for instance *Small_2_12*.

	Sesame	Onion	Corn	Red Tomato	Green Pepper
$Ky^{p}_{\gamma(j)}$	0.30	0.45	0.40	0.40	1.10
Ya^{p-1}_{ijz}	0.60	40.60	4.85	38.10	24.70

are given by the humidity sensors. The amount of water required by crop i in each physical management zone $z \in Z^{Ph}(j)$ of parcel j at current irrigation period p, ETc^{p}_{ijz}, is calculated with Eqs. (2.9) and (2.10) using information from INIFAP.

Table 2.12 shows the final expected yields for each crop (columns 2-6) of instance *Small_2_12* after each irrigation period (first column), the percentage of irrigation water (column 7) with respect to the total available water (81,000 m^3 for each period), and farmer's expected final profit (column 8) after each irrigation period.

Table 2.12: Expected yields by crop, % of water used for irrigation, and farmer's expected profit after each irrigation period.

Irrigation Period	**Expected Yields (tn)**					**Water**	**Profit**
	Sesame TCS	Green pepper BMF	Corn grain BCF	Red Tomato GMF	Onion TMF	%	$
1	30.00	98.80	33.95	114.30	690.20	53.40	3,627,366.19
2	30.00	98.80	33.95	114.30	690.20	53.40	3,627,366.19
3	30.00	98.80	30.00	114.30	464.60	100.00	2,847,269.31
4	30.00	98.80	30.00	114.30	329.80	100.00	2,391,531.03
5	30.00	36.39	30.00	41.65	291.52	100.00	1,796,976.97
6	30.00	36.39	30.00	41.65	291.52	61.10	1,796,976.97

RTIP complies with the established demand in the CPP (sesame, green pepper, corn grain and red tomato) at the end of the production cycle. We can see that crops with no initial demand are the most affected when there is not enough water, even if these crops would have given more benefits to the producer, such as the onion. Notice that during periods 1, 2, and 6 the farmer saved a significant percentage of the available water, making the simulated drought less severe. In the first two periods the crops were irrigated at the optimal level, so the farmer's profit remained at 100%. However, periods 3, 4, and 5, required more water in each irrigation period. This situation drove a decrease of 50.06% in the farmer's final profit.

Fig. 2.10 shows the expected yield of each physical management zone after each irrigation period. The X axis represents the parcels while the Y axis is the expected profit expressed as a percentage. Periods 2 and 6 are not shown because they remained the same as the previous period.

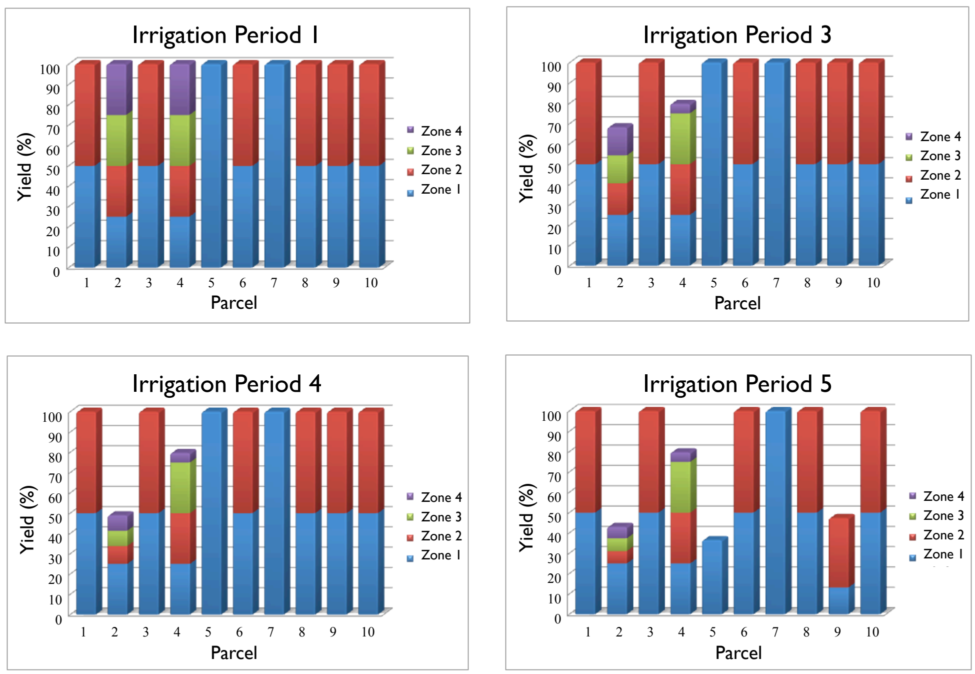


Figure 2.10: Expected yield reached in each management zone after each irrigation period.

During periods 1 and 2, the crops are in their initial phenological stages, meaning all the crops can be irrigated at the optimal level. In period 3 there is not enough water to irrigate all crops at the optimal level, so the expected yields of parcel 2 in zones 2, 3, and 4, and the expected yields of parcel 4 in zone 4 decrease considerably. Since there is not enough water to irrigate the crops at the optimal level on their flowering and yield formation stages, the expected yields of parcel 2 in zones 2, 3, and 4 decrease again at period 4. At period 5 the expected yield of parcel 2 in zones 2, 3 and 4, the expected yields of parcel 5 in zone 1, and the expected yields of parcel 9 in zones 1 and 2 decrease considerably. At period 6 the crops are in their final phenological stage and do not consume much water, meaning all of them can be irrigated at the optimal level.

RTIP guarantees to supply only the amount of water needed to satisfy the crop water requirements and helps the farmer when a severe drought arises by taking into account the physical soil properties of the parcel, prices in the market, the current climate, and the phenological stages of the crops.

2.2.4 Conclusions of Section

Soil diversity within agricultural production parcels is an important characteristic that affects both the agricultural planning and water management processes. Dividing the field into site-specific management zones, that take chemical and physical soil properties into consideration, is an interesting critical way to face within-field variability and improve the agricultural practices.

In this study, we present a new methodology based on mathematical models of linear programming and site-specific management zones (chemical and physical) to determine the optimal crop pattern and to decide in real-time the amount of water to be used in crop irrigation at each irrigation period.

The use of site-specific management zones is a strategy that helps to determine the crop pattern and the use of water in agricultural fields considering the crop requirements in real-time. However, this activity represent an extra-cost (in effort and investment) because it is necessary to have soil samples and to analyze them to determine the soil properties of each parcel and their site-specific management zones. The number of management zones in a parcel can be determined by several factors as the homogeneity of parcel and the investment capital, just to name a few. In this study, we assume that the site-specific management zones of each parcel has been previously delineated. Also, it is nec- essary to invest in technology, as humidity sensors, to monitor in real-time the crop requirements in each one of the management zones. All the previous factors represent an extra effort to farmer but, the profit of the farmer at the end of the production cycle can increase considerably with respect to the traditional method and, the costs of using site-specific management zones can be negligible.

In our models we use real-time information about crops and environment (temperature, moisture level, solar radiation, wind, phenological stage of crops, etc.), and they are solved optimally in less than a second with low computational requirements.

Experimental results show that the new methodology is efficient and practical for used in a decision support system to improve agricultural planning processes.

2.2.5 Acknowledgements

We thank INIFAP Zacatecas for information related to evapotranspiration parameters.

Chapter 2

Agricultural Planning

2.3 A Hierarchical Planning Scheme Based on Precision Agriculture

This research produced the chapter: Albornoz V. M., Cid-Garcia, N. M., Ortega R., & Rios-Solis, Y. A. (2015). A hierarchical planning scheme baed on precision agriculture. In *Handbook of Operations Research in Agriculture and the Agri-Food Industry*. (pp. 129-162). Springer New York.

Abstract

The process for agriculture planning starts by delineating the field into site-specific rectangular management zones to face within-field variability. We propose a bi-objective model that minimizes the number of these zones and maximizes their homogeneity with respect to a soil property. Then we use a method to assign the crops to the different plots to obtain the best profit at the end of the production cycle subject to water forecasts for the period, humidity sensors, and the chemical and physical properties of the zones within the plot. With this crop planning model we can identify the best management zones of the previous bi-objective model. Finally, we show an efficient approach that at each irrigation turn decides how much and which plots must be watered such to maximize the total final yields. This is a critical decision in countries where water shortages are frequent. In this study we integrate these stages in a hierarchical process for the agriculture planning and empirically prove its efficiency.

2.3.1 Introduction

Precision agriculture has modified the decision making tools used by the farmers in order to plan their production cycle as well as their daily operation. Investing in precision agriculture goods such as humidity sensors or soil samples is interesting when farmers get not only tools for monitoring their fields but get a powerful hierarchical system that optimizes the benefits of the yields as the one we present in this study.

One of the main aspects of precision agriculture is to provide farming management methods to respond to within-field variability. Precision agriculture permits the application in a site-specific manner of agronomic practices such as fertilization, weed and pest control, as a function of the information compiled from collected field data.

Physical and chemical soil properties make the soil suitable for agricultural practices. Texture, structure, and porosity influence the movement and retention of water, air and solutes in the soil, which subsequently affect plant growth and organism activity. Chemical soil properties affect nutrient availability and growing conditions (McCauley *et al.*, 2005). All this properties may be altered by management practices that are usually expensive so it is imperative to determine which zones of the plots need theses practices.

The first problem farmers face (see Figure 2.11) is how to delineate management zones within the plots before planting the crops to improve the overall yield. More precisely, a management zone is a sub-region of a plot that is relative homogeneous with respect to soil parameters, and for which a specific rate of agricultural inputs is needed (Roudier *et al.*, 2008). For this, soil samples are taken and then analyzed. One of the main contributions of this work is a model that takes as input the soil samples and delineates the minimum number of rectangular management zones such that the homogeneity within the obtained zones is maximized. Indeed, tiny management zones even if they are rectangular are difficult to operate by agriculture machinery. Therefore, a main issue that is solved in this work is to have the largest management zones that are the most homogeneous possible. We name our methodology as Minimization of **Rectangular and Homogeneous Management Zones (R&H-MZ)**. Two types of management zones are obtained depending of the soil property used during the delineation method: physical and chemical management zones.

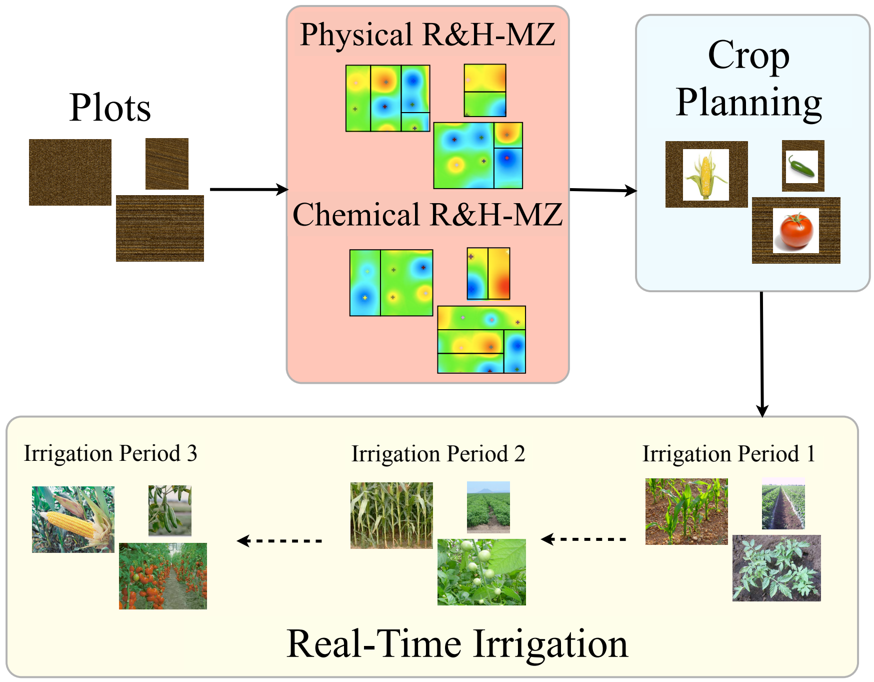


Figure 2.11: Hierarchical agriculture planning method HAP.

The second problem encountered by farmers is to select the crops that they are going to sow into their plots considering the previously delineated management zones. For example, if a plot has several management zones with high amount of phosphorus and nitrogen then probably it would be better to plant tomatoes or maize because they would save in fertilizers. This problem is known as the **Crop Planning Problem (CPP)**. It becomes quickly a hard problem since there are many parameters to take into account.

Once the CPP problem has been solved and the selected crops are already planted, the decisions the farmers must take are mainly about the optimal amount of water that has to be irrigated to the plots at each irrigation period (see Figure 2.11). This is another hard problem, named as the **Real-Time Irrigation Problem (RTIP)**, and it considers the phenological stages of the crops, the soil properties of the management zones, the data from the humidity sensors, the evapotranspiration factors, and the previous irrigation decisions. When a drought arises and the total amount of water is not sufficient to irrigate all the crops to optimality the RTIP decides which crops must be under deficit irrigation or even without irrigation in order to maximize the total final benefits at the end of the production cycle.

In this work we propose an approach we name as **Hierarchical Agriculture Planning (HAP)** for helping the decision makers (the farmers) to plan and operate their plots avoiding wastage and maximizing their benefits. The importance of an hierarchical approach resides on the fact that one stage needs as input the results of the previous one.

HAP is composed by a new bi-objective mathematical model that improves the method proposed by Cid-Garcia *et al.* (2013) to solve R&H-MZ methodology since it considers both the minimum number of management zones and the maximum homogeneity within these zones. One of the contributions of this work is to use the CPP methodology of Cid-Garcia *et al.* (2014) as a criterion to help the farmer chose between the set of proposals that arise from R&H-MZ. As mentioned, with CPP the farmer obtains an optimal crop pattern. After the crops have been planted on the parcels, HAP takes hand of the real-time irrigation method proposed RTIP by Cid-Garcia *et al.* (2014) (see Figure 2.11).

As mentioned by Bitran y Hax (1977), to provide effective managerial support for decisions related to production planning, it is useful to partition the set of decisions into a hierarchical framework as we propose in the HAP. Indeed, strategical higher level decisions (management zones) impose constraints on tactic lower level actions (crop planning and real-time irrigation), and lower level decisions provide the necessary feed-back to evaluate higher level actions for future production cycles.

The rest of the work is organized as follows. Section 2.3.1 is devoted to related scientific literature. In Section 2.3.2 we present the R&H-MZ methodology. In Section 2.3.3 we present the Crop Planning Problem and how to use it as a discriminator between the set of solutions given by R&H-MZ. In Section 2.3.4 is summarized the Real-Time Irrigation Problem. In Section 2.3.5 we empirically test the R&H-MZ methodology together with the HAP approach on a real instance. Finally, Section 2.3.6 concludes the study.

Literature Review and Terminology

Most of the approaches in literature for determining management zones are based on clustering algorithms. Many of them are based on soil samples information like in our case. For example, Fraisse *et al.* (2001) and Schepers *et al.* (2004) use soil and relief information, Carr *et al.* (1991) base their zoning on topographic maps while methods of Bhatti *et al.* (1991), Mortensen *et al.* (2003), or Mulla (1991) need soil sampling. Other approaches are based on yield maps, combining data from several seasons like in Blackmore (2000), Diker *et al.* (2004), and Pedroso *et al.* (2010). Some other clustering methods combine soil samples and yield maps: Franzen y Nanna (2003), Hornung *et al.* (2006), Hornung *et al.* (2003), and Whelan *et al.* (2003). In Roudier *et al.* (2008) they use a watershed segmentation algorithm where the user can introduce morphologies of the desired zones.

Usually, K-means or Fuzzy K-means methods are used for the classification like in Jiang *et al.* (2011), Li *et al.* (2005), and Ortega *et al.* (2002), or principal component analysis with a cluster method (Ortega y Flores, 1999). Nevertheless, the choice of the data layers processed by the clustering is an issue as mentioned by Jaynes *et al.* (2005). Moreover, the resulting fragmentation of the oval shaped zones due to clustering methods is not an appealing solution for farmers as pointed out by Frogbrook y Oliver (2007), Li *et al.* (2005), and Simbahan y Dobermann (2006).

Indeed, to the best of our knowledge, Cid-Garcia *et al.* (2013) are the first to propose a management zone delineation method that directly gives rectangular shape zones. This is important since most of the fertilizing agricultural machinery is towed by tractors. Moreover, most of the irrigation systems are designed in a rectangular pattern. In this work, we improve the results of Cid-Garcia *et al.* (2013) since instead of minimize the variance between the fields, we minimize the number of zones. Additionally, we maximize the homogeneity within the management zones. This gives a bi-objective model that offers more practical solutions for the farmers.

In terms of crop planning, Sarker *et al.* (1997) propose a linear programming model considering land type, alternative crops, crop patterns, input requirement, investment, and output. Nevertheless, they do not take into account that the water is a restriction as we do in this work. Later, Sarker y Ray (2009) formulate a crop planning problem as a multiobjective optimization model. Mainuddin *et al.* (1997) propose a crop planning model for an existing groundwater irrigation. Nevertheless, they do not consider the use of humidity sensors as we do in this work. Adeyemo y Otieno (2010) present evolutionary algorithm to solve the multiobjective crop planning model: minimize the total irrigation water, to maximize both the total net income from farming and the total agricultural output. Contrary to our research, water availability is not a restriction.

Ortega Álvarez *et al.* (2004) propose a non-linear model solved by genetic algorithms to identify production plans, and water irrigation management strategies. They estimate crop yield, production and gross margin as a function of the irrigation depth. In our work, the yields also depend of the irrigation depth but we manage to have linear restrictions. Moreover, we use real time data for the irrigation stage. Sahoo *et al.* (2006) propose some fuzzy multiobjective linear programming models for land-water-crop system planning. Reddy y Kumar (2008) present a multiobjective approach for the optimal cropping pattern and operation policies for a multi-crop irrigation reservoir system. These authors do not consider water shortages since they try to maximize the yields and to minimize the water.

In Casadesús *et al.* (2012), Hassanli *et al.* (2009), Hedley y Yule (2009), and Xu *et al.* (2011) authors propose schedule irrigation plans according to weather conditions, crop development, and other factors. In this work we propose a mathematical model instead of heuristics. In Alminana *et al.* (2010) they present models and algorithms to determine water irrigation scheduling by taking into account the irrigation network topology, the water volume, technical conditions, and the logistical operations. Their models do not use the real time information of humidity sensors like we do in this research.

We use the term field for the whole of land that can be irrigated by a water well or a dam. This field is made up of different plots. In each plot a single crop is to be planted. Each plot is subdivided into physical and chemical management zones (see Figure 2.12). Notice that all the management zones of a plot must be planted with the same crop.

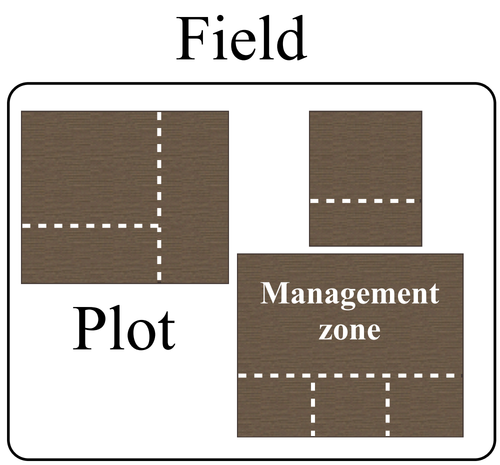


Figure 2.12: Terms used in this article: field, plot, and management zone.

2.3.2 Rectangular and Homogeneous Management Zones

At the begin of the production cycle two delineations of rectangular and homogeneous management zones are made for each one of the plots. The first one uses chemical soil properties and the second one physical soil properties.

The delineation with chemical soil properties is used to determine the expected amount of nutrients (fertilizers, pesticides, etc.) that the crops require in the whole production cycle, while the delineation with physical soil properties is used to determine the expected amount of water required by the crops in the whole production cycle and the amount of water required by the crops during each irrigation period, respectively.

The R&H-MZ methodology proposed in this work improves the one of Cid-Garcia *et al.* (2013) since we present a bi-objective problem where the number of rectangular management zones is minimized and the homogeneity within the zones is maximized. R&H-MZ methodology consists of two main stages:

a) **Instance generation.** In this stage, we process the information from the soil samples that have been taken from the field. These soil samples are approximately equidistant in the field (a GPS detects their position). Then, they are labeled and their positions are translated into the first quadrant of the Cartesian map. Next, the information about each soil property is registered (pH, organic matter rate (OM), amount of phosphorus (P), sand, field capacity, permanent wilting point, etc.). Soil texture is considered among the most important physical properties and it corresponds to the proportion of three mineral particles (sand, silt, and clay). Then, all the quarters (or potential zones) are computed together with their variances (more details about this stage are given below).

b) **Mathematical model.** With the input of stage a), we propose a bi-objective Integer Linear Programming (BILP) (see Section A.6). The aim of the BILP is to find the minimum set of rectangular management zones such that they cover the whole field and at the same time this set is the one that maximizes the homogeneity within the selected management zones. BILP has a set of optimal solutions that are a trade-off between the two objectives. This set (or Pareto front) is exactly obtained by an ϵ-constraint algorithm.

We now describe these two stages in more details. Soil diversity in the field can be observed from a thematic map of a certain property (here we use MapInfo with the default grid and the inverse distance weighting interpolator).

In Figure 2.13 is presented the thematic map of a real plot using phosphorus as chemical soil property and the label of each soil sample (in this case we have 40 soil samples). In the thematic map we can see the regions at optimal level.

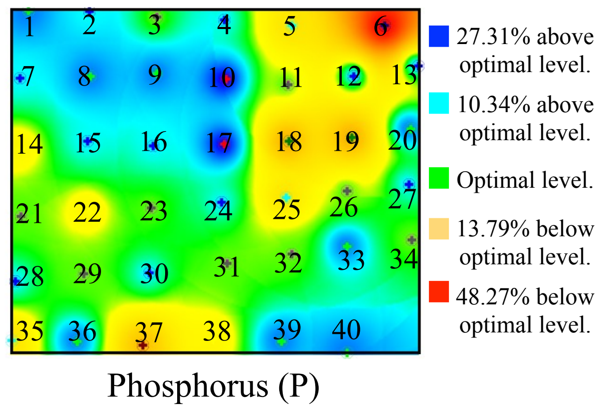


Figure 2.13: Thematic map of phosphorus of a real plot. The numbers indicate the label of each soil sample.

The thematic map help us to determine the smaller management zone allowed, $MinWidthQ \times MinLengthQ$, where $MinWitdthQ$ is the number of samples in the width of the smaller zone and $MinLengthQ$ is the number of samples in its length. If the diversity of the soil property is high then the zones should be relatively small (one sample width by one sample length in the worst case).

Once the minimum size of a zone is set, we enumerate all the possible management zones (or quarters) that could be created in this plot. Notice that the soil samples included inside of each potential zone is know. The search of potential zones can be done in $\Omega(WidthF \times LengthF)$ where $WidthF$ is the number of samples in the width of the field while $LengthF$ is the number of samples in its length.

An illustration is given in Figure 2.14. The left hand side of this figure shows a plot with nine samples (each one with its label). On the right side, all potential zones are labeled. For this example, we have a total of 36 rectangular quarters (generated by Algorithm 2 presented below). Each quarter shows which samples are included on it, e.g. quarter 1 include only the sample 1 but quarter 30 include the samples 4 to 9.

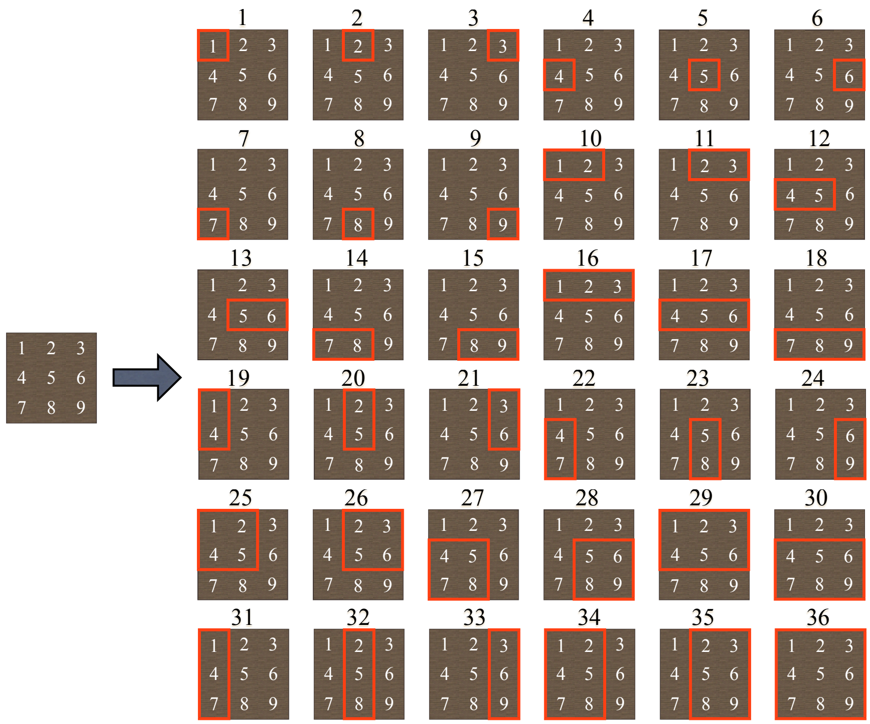


Figure 2.14: The 36 potential management zones or quarters of a plot.

The soil samples are almost equidistant, in our example four soil samples (two width for two long) are needed to cover an *ha* but this number can change according to the farmer's requirements. The total number of potential zones $|Z|$ can be computed by the following formula:

$$|Z| = \left(\sum_{i=1}^{WidthF - MinWidthQ + 1} i \right) \left(\sum_{j=1}^{LengthF - MinLengthQ + 1} j \right).$$

The determination of all possible management zones is implemented by Algorithm 2 from Cid-Garcia *et al.* (2013). The input of this algorithm is the soil samples data, the number of samples in the width of the plot ($WidthF$), the number of samples in the length of the plot ($LengthF$), the minimum quantity of samples the width of a quarter must contain ($MinWidthQ$), and the minimum quantity of samples the length of a quarter must contain ($MinLengthQ$). The algorithm starts creating the smallest quarters width wise. Then it checks if there is still some width to cover. After, it checks the length.

Algorithm 2 Quarters generation of a plot.

```
INPUT: WidthF, LengthF, MinWidthQ, MinLengthQ, soil samples
for j = MinWidthQ To WidthF do
  for l = 0 To (WidthF − 1) do
    if (j + l) ≤ WidthF then
      for i = MinLengthQ To LengthF do
        for k = 0 To (LengthF − 1) do
          if (k + i) ≤ LengthF then
            creation of a new quarter
          end if
        end for
      end for
    end if
  end for
end for
```

The result of Algorithm 2 is a correspondence matrix $C = \{c_{zs}\}$, where $c_{zs} = 1$ if quarter with label z covers sample point with label s, $c_{zs} = 0$ otherwise. Once all the potential quarters are determined, we compute the variance of a particular soil property for the set of the samples included in each potential quarter. For example, a quarter that only covers a soil sample would have a variance of 0. A quarter that covers three soil samples would have the variance computed from these three samples. Also, there would be a quarter that covers all soil samples (i.e., there is only one zone that is equal to the plot).

For an example about this instance generation stage see Section 2.3.5. Next stage is the mathematical model that requires the correspondence matrix of the potential quarters together with their variances. For this purpose, let Z be the set of potential quarters and S the set of soil samples of the field ($|S| = N$). Each quarter z has n_z soil sample points. Farmers do not wish to have tiny management zones because of their machinery, so let LS be the maximum number of zones in the field while LI is the minimum one.

The set of decision variables of the BILP model is:

$$q_z = \begin{cases} 1 & \text{if quarter with label } z \text{ is chosen,} \\ 0 & \text{otherwise.} \end{cases}$$

The main idea is to cover the plot by a set of non overlapping quarters. To guarantee a homogeneous zoning delineation we use the relative variance since it has been proved to be a high quality criterion to measure the efficiency of a zoning method (Ortega y Santibáñez, 2007; Cid-Garcia *et al.*, 2013). Suppose a set of quarters $Q \subset Z$ is already selected, then the relative variance of Q is $RV(Q) = 1 - \frac{\sum_{z\in Q} \sigma^2_{w_z}}{\sigma^2_T}$, where σ^2_T is the total variance of all the field and sum of the $\sigma^2_{w_z}$ is the variance within each $z \in Q$ defined as follows:

$$\sum_{z\in Q} \sigma^2_{w_z} = \frac{\sum_{z\in Q}(n_z - 1)\sigma^2_z}{N - |Q|}. \tag{2.23}$$

Numerator in (2.23) considers the number of samples n_z in quarter z (minus one degree of freedom) as a weight and the denominator takes into account the number of selected quarters (total number N minus the number of quarters $|Q|$). Therefore, we have the following equation where variable $\alpha \in [0, 1]$ must be maximized to have the highest homogeneity within the management zones:

$$\left(1 - \frac{\sum_{z\in Z}(n_z - 1)\sigma^2_z q_z}{\sigma^2_T \left[N - \sum_{z\in Z} q_z\right]}\right) \geq \alpha. \tag{2.24}$$

The BILP model to determine the best management zones for a plot is as follows:

$$\left\{\min \sum_{z\in Z} q_z, \max \alpha\right\} \tag{2.25}$$

$$\text{s.t.} \sum_{z \in Z} c_{zs} q_z = 1 \quad \forall s \in S \tag{2.26}$$

$$\sum_{z \in Z} q_z \leq LS \tag{2.27}$$

$$\sum_{z \in Z} q_z \geq LI \tag{2.28}$$

$$\left(1 - \frac{\sum_{z \in Z} (n_z - 1)\sigma_z^2 q_z}{\sigma_T^2 \left[N - \sum_{z \in Z} q_z\right]}\right) \geq \alpha \tag{2.29}$$

$$\alpha \in [0,1], q_z \in \{0,1\} \quad \forall i \in I$$

Bi-objective function (2.25) minimizes the sum of the chosen zones (or potential management zones) and maximizes the homogeneity within each management zones (value of α). Restrictions (2.26) ensure that every point sample s is covered by only one zone, i.e., the whole filed is partitioned into non overlapping zones. Constraints (2.27) and (2.28) limit the number of zones in which the plot will be partitioned. Restriction (2.29), which can be easily linearized, corresponds to the relative variance.

In Section 2.3.5 we prove that the objective functions considered by BILP are conflicting. For the kind of bi-objective problems as BILP, there are many solution that optimize both objectives. The set of non-dominated solutions (for non-dominated solution there are not other solutions that improve an objective without worsening the other one) represents the trade-off set satisfying both objectives. This trade-off curve is known as the Pareto front and we compute it using the ϵ-*constraint* method (Marler y Arora, 2004a; Ehrgott, 2005). This ϵ-*constraint* method optimizes one of the objective functions using the other one as constraint of the model. In our case we have if we apply the ϵ-*constraint* method we get the following problem for an α that is fixed and not anymore a decision variable.

$$\begin{aligned}
\min \quad & \textstyle\sum_{z \in Z} q_z \\
\textit{subject to:} \quad & (2.26) - (2.28) \\
& \left(1 - \frac{\sum_{z \in Z} (n_z - 1)\sigma_z^2 q_z}{\sigma_T^2 \left[N - \sum_{z \in Z} q_z\right]}\right) \geq \alpha \\
& q_z \in \{0,1\} \qquad \forall i \in I
\end{aligned} \tag{2.30}$$

By using a parametrical variation of the values of α the efficient solutions of the problem can be obtained. Indeed, in our case α acts as the ϵ of the method.

Once that we have the Pareto front, the next step is to choose a solution from it that satisfies the farmer's requirements and that guarantees the wished homogeneity within each zone. Experimental results of this bi-objective model in HAP are presented in Section 2.3.5.

2.3.3 Crop Planning Problem and Selection of the Best Management Zones

We first summarize the CPP problem presented in Cid-Garcia *et al.* (2014) and then, in Section 2.3.3, we show how to use it in order to select the best solution among the Pareto front obtained by R&H-MZ.

Crop Planning Problem

After the chemical and physical management zones have been obtained by R&H-MZ, the second decision in HAP is which crops i to plant in the different plots j by taking into account the soil properties of the physical and chemical management zones that were previously delineated. In this section we present a mixed integer linear programming (MIP) to solve CPP (see Section A.4) which improves the model presented in Cid-Garcia *et al.* (2014) since it introduces the chemical and physical management zones of the plots.

Let I be the set of the different crops a farmer could plant, J the set of plots of the farmer's field, $Z^{Ph}(j)$ the set of physical management zones within plot j while $Z^{Ch}(j)$ the set of chemical management zones of j. The data set we use for the MIP mathematical model is described in the following list.

- G_i is the expected benefit of selling a ton (tn) of crop i at the end of the production cycle.

- $Cirr_{jz}$ is the cost of irrigating one cubic meter (m^3) of water in plot j and physical zone $z \in Z^{Ph}(j)$.
- $Cseed_i$ is the cost of buying a kilogram (kg) of seed of crop i.
- $Cplant_{ijz}$ is the cost of planting an hectare (ha) of crop i in plot j, and chemical management zone $z \in Z^{Ch}(j)$.
- ha_j is the number of ha of plot j.
- hac_{jz} corresponds to the number of ha in chemical management zone $z \in Z^{Ch}(j)$ of plot j.
- hap_{jz} corresponds to the number of ha in physical management zone $z \in Z^{Ph}(j)$ of plot j.
- $Iseed_i$ is the quantity of seeds in kg of crop i in the farmer's stock.
- $Seed_i$ is the quantity of seeds in kg needed to plant a ha of crop i.
- Y_i is the expected yield in tn by ha of crop i at the end of the production cycle.
- D_i is the demand in tn of crop $i \in \bar{I} \subsetneq I$ where $\bar{I}$ is a subset of the crop set I. This is to model situations where a farmer is payed in advance for some yields of a specific crop.
- W is the expected total amount of water in m^3 for all the production cycle.
- W_{ijz} is the expected amount of water in m^3 needed for irrigating a ha planted with crop i in plot j in physical management zone $z \in Z^{Ph}(j)$.

Parameter W_{ijz} can be obtained either by historic data or by deriving it from the Penman-Monteith equation (Allen *et al.*, 2006):

$$ETc_{ij}^v = ETo^v \cdot Kc_{ij}^v \tag{2.31}$$

where ETc_{ij}^v is the crop evapotranspiration that represents the amount of water (in mm) required by crop i at phenological stage v for plot j, ETo^v is the reference crop evapotranspiration that expresses the evaporating power of the atmosphere (in mm) during phenological stage v. The crop coefficient Kc_{ij}^{v} values change from crop to crop, phenological stage of the crop v, and geographic location j. Then, total expected amount of water consumed by crop i planted in plot j and situated in physical management zone $z \in Z^{Ph}(j)$ throughout the production cycle (W_{ijz}) is the sum of all the expected amount of water required by crop i planted in plot j for each vegetative stage v (ETc_{ij}^{v}) minus the sum of all the stored water in plot j in physical management zone $z \in Z^{Ph}(j)$ at each vegetative stage v (SW_{jz}^{v}):

$$W_{ijz} = \left(10 \sum_{v} ETc_{ij}^{v}\right) - \sum_{v} SW_{jz}^{v}. \tag{2.32}$$

The assignment variables for the CPP integer linear programming model are:

$$\boldsymbol{x_{ij}} = \begin{cases} 1 & \text{if crop } i \text{ is planted in plot } j, \\ 0 & \text{otherwise.} \end{cases}$$

Finally, variables $\boldsymbol{s_i}$ correspond to the amount of seeds the farmer must buy of crop i in *kg*, $i \in I$.

$$\max \sum_{i \in I} \sum_{j \in J} \left[\boldsymbol{x_{ij}} \cdot G_i \cdot Y_i \cdot ha_j - \boldsymbol{x_{ij}} \sum_{z \in Z^{Ch}(j)} Cplant_{ijz} \cdot hac_{jz} \right.$$
$$\left. - \boldsymbol{x_{ij}} \sum_{z \in Z^{Ph}(j)} Cirr_{jz} \cdot W_{ijz} \cdot hap_{jz} \right] - \sum_{i \in I} \boldsymbol{s_i} \cdot Cseed_i \tag{2.33}$$

subject to:

$$\sum_{i \in I} \boldsymbol{x_{ij}} \leq 1 \qquad j \in J \tag{2.34}$$

$$\sum_{j \in J} \sum_{z \in Z^{Ch}(j)} Y_i \cdot hac_{jz} \cdot \boldsymbol{x_{ij}} \geq D_i \qquad i \in \bar{I} \tag{2.35}$$

$$\sum_{j \in J} \sum_{z \in Z^{Ch}(j)} Seed_i \cdot hac_{jz} \cdot \boldsymbol{x_{ij}} \leq Iseed_i + \boldsymbol{s_i} \qquad i \in I \tag{2.36}$$

$$\sum_{i \in I} \sum_{j \in J} \left[\sum_{z \in Z^{Ph}(j)} W_{ijz} \cdot hap_{jz} \right] \boldsymbol{x_{ij}} \leq W \tag{2.37}$$

$$\boldsymbol{s_i} \geq 0, \boldsymbol{x_{ij}} \in \{0, 1\} \qquad i \in I, j \in J$$

Objective function (2.33) maximizes the total expected benefits: first term represents the benefits of selling the expected yields of each crop planted in each plot, second one corresponds to the cost of planting the crops in each one of the plots (it includes fertilizers and pesticides that each chemical management zone would require), third term is about the irrigation costs per plot and per physical management zone, finally, we have the cost of buying seeds.

Restrictions (2.34) guarantee unique assignment of a crop i to each plot j. Restrictions (2.35) specify that there are some crops i that must be planted in order to satisfy a certain demand D_i but only for crops in $\bar{I} \subset I$. Restrictions (2.36) determine the amount of seeds needed and also the amount of seeds that must be bought. Restrictions (2.37) establish that the expected amount of water needed to irrigate the crops must be sufficient for the whole production cycle.

Notice that we are supposing that any crop i can be planted on any plot j. In the case where some crops could not be planted in a specific plot (due to soil cycles, or experience) we could easily introduce the notation $J(i)$ corresponding to the set of plots where i can be planted and analogously, $I(j)$ would be the set of crops that can be planted on plot j.

The CPP is a NP-hard problem which has a MIP that is elegant enough to optimally solve real size instances by a branch-and-bound algorithm in less than one second as it can be seen in Section 2.3.5.

Selection of the Best Management Zones

After the Pareto front has been obtained by R&H-MZ, the next step is to select the best delineation of chemical and physical management zones in each plot of the farmer's field, that is, the delineation that gives the best profit at the end of the production cycle.

It is difficult for the farmer to establish a criterion to chose between a delineation with $\alpha = 0.7$ or $\alpha = 0.5$ (Ehrgott, 2005). Therefore, we execute CPP for each solution of the Pareto front of R&H-MZ and selected the management zones that gives the best farmer profit.

Figure 2.15 shows an example about this procedure. We show the delineation resulting of R&H-MZ for a plot using α values of 0.5, 0.7, and 0.9. After executing CPP, we obtain that the best delineation is obtained with alpha value of 0.5. This manner, the farmer does not have to specify a way of discriminating the solutions in the Pareto front, the HAP procedure does it for the farmer.

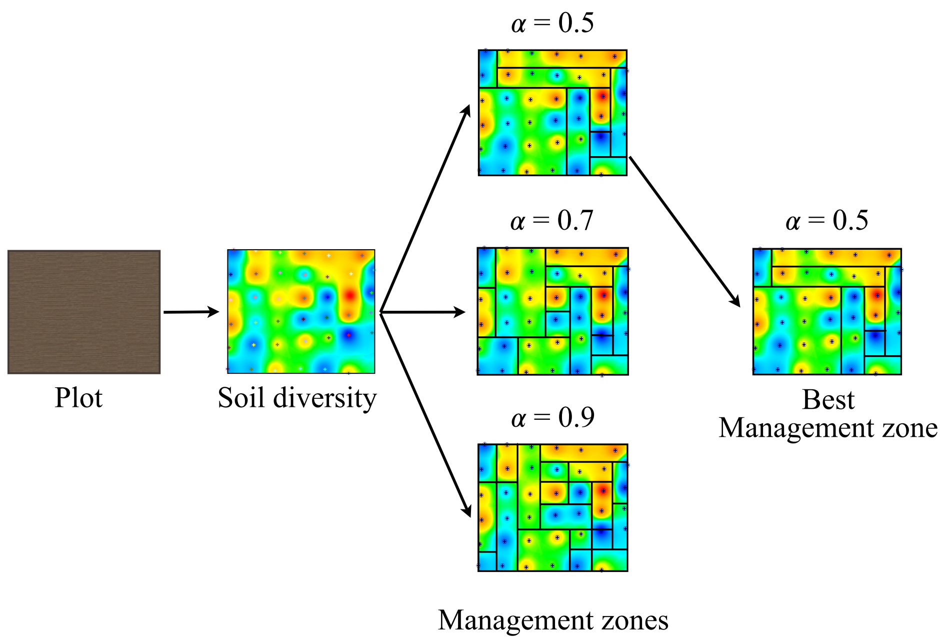


Figure 2.15: Selection of the best management zone.

2.3.4 Real-Time Irrigation Problem

Suppose that a drought arises once the crops have already been planted in the plots. How to choose the crops that need to be irrigated to optimality, the crops that would be in deficit irrigation, and the crops that would not be irrigated at all (until the point to let a crop die) in order to maximize the benefits at the end of the production cycle?

The production cycle of each crop is divided in different irrigation periods, then, at the beginning of each irrigation period the farmer must decide how much water must be assigned to each plot according to their water requirements in real time, to maintain the maximum crop yield at the end of the production cycle. For this, we use the information of water sensors placed in each one of the physical management zones of the plots. Notice that the aim is to have the maximum possible yields at the end of the production cycle considering that each crop has different vegetative cycles and different needs of water. If there are no water shortages, the crops must be irrigated to optimality. If there are water shortages, then the farmer needs to decide which crops is better to put under deficit irrigation.

The crop water production function (Doorenbos *et al.*, 1979) is given by

$$\frac{Ya_{ij}^p}{Ym_{ij}^p} = 1 - Ky_i^p \left(1 - \sum_{z \in Z^{Ph}(J)} \frac{ETa_{ijz}^p}{ETc_{ijz}^p}\right) \tag{2.38}$$

where the only unknown parameter is Ya_{ij}^p that corresponds to the real yields of crop i planted in plot j at period p. Ym_{ij}^p is the maximum yield reached by crop i in parcel j at last irrigation period p. When p is the first irrigation period Ym_{ij}^p is the harvested yield of crop i under an optimal growing environment, i.e., the yield of the crop is not limited by water, nutrients, pests, nor diseases.

ETc_{ijz}^p represents the maximum water requirements of crop i in plot j and physical management zone $z \in Z^{Ph}(j)$ at period p. This is expressed by the sum of all the rates of evapotranspiration in mm per phenological stage v since the last irrigation period minus the sum of all the amount of stored water in plot j in physical management zone $z \in Z^{Ph}(j)$of each vegetative stage v since the last irrigation period p (Allen *et al.*, 2006). This value can be computed by equation (2.31) presented in Section 2.3.3. Based on equation (2.32), the amount of water in m^3/ha needed by crop i planted in physical management zone $z \in Z^{Ph}(j)$ of plot j for irrigation period p is

$$ETc_{ijz}^p = \left(10 \sum_{v(p)} ETc_{ij}^v\right) - \sum_{v(p)} SW_{jz}^v. \tag{2.39}$$

where $v(p)$ represents all the vegetative stages in irrigation period p.

ETa_{ijz}^p represents the amount of stored water in the soil of plot j in physical management zone $z \in Z^{Ph}(j)$ at current irrigation period p plus the amount of water supplied at current irrigation period. The water level stored in each one of the physical management zones of the plots, before the current irrigation, is determined by a moisture sensor in real time.

Response factor Ky_i^p represents the relationship between water and yield for crop i. These values are crop specific and have different values at each irrigation period p.

At the beginning of each irrigation period, the farmer knows the volume of available water. This volume may not be the expected one, therefore some crops are not going to be optimally irrigated. For each irrigation period p a Linear Programming (LP) must be solved to obtain the amount of water to be irrigated in each physical management zone to maximize the expected total benefit (see Section A.2). The parameters needed to formulate the LP model for RTIP are listed below:

- G_i^p expected benefit of selling a tn of crop i at the end of the production cycle given that we are at period p. Indeed, this value is known at the beginning of each irrigation period p but it can vary from period to period.

- $\gamma(j) = i$ is a function that indicates that crop i is sown in plot j (this is obtained from the solution of CPP).

- ETc_{ijz}^p corresponds to the real amount of water in m^3/ha that crop i needs in physical management zone $z \in Z^{Ph}(j)$ of plot j at period p. It is easy to compute which vegetative stages v corresponds to crop i at period p, so we omit cumbersome notation.

- hap_{jz} is the number of hectares in physical management zone $z \in Z^{Ph}(j)$ of plot j.

- SW_{jz}^p is the amount of water that already exists in physical management zone $z \in Z^{Ph}(j)$ of plot j at the beginning of irrigation period p. This information is retrieved from the humidity sensors in m^3/ha.

- W^p represents the amount of available water for irrigation period p.

- Ky_i^p is the yield response factor of crop i corresponding at period p.
- $\overline{\mathcal{Y}}_{jz}^{p-1}$ is the maximum crop yield in tn/ha in physical management zone $z \in Z^{Ph}(j)$ of plot j reached at previous irrigation period p.

The variables used in the formulation of the RTIP model are listed below:

- $\boldsymbol{w_{jz}^p}$ is a variable representing the amount in m^3 of irrigated water in physical management zone $z \in Z^{Ph}(j)$ of plot j at period p.
- $\boldsymbol{\mathcal{Y}_{jz}^p}$ represents the current total crop yield in tn/ha reached in physical management zone $z \in Z^{Ph}(j)$ of plot j computed after current irrigation period p.

The LP for RTIP is as follows:

$$\max \quad \sum_{i \in I} G_i^p \left(\sum_{\{j \mid \gamma(j)=i\}} \sum_{z \in Z^{Ph}(j)} hap_{jz} \cdot \boldsymbol{\mathcal{Y}_{jz}^p} \right) \tag{2.40}$$

$$\textit{subject to:} \quad \boldsymbol{\mathcal{Y}_{jz}^p} = \overline{\mathcal{Y}}_{jz}^{p-1} \left(1 - Ky_{\gamma(j)}^p \left(1 - \frac{\boldsymbol{w_{jz}^p} + (SW_{jz}^p \cdot hap_{jz})}{ETc_{ijz}^p \cdot hap_{jz}} \right) \right)$$

$$i \in I, \{j \mid \gamma(j) = i\}, z \in Z^{Ph}(j) \tag{2.41}$$

$$\sum_{j \in J} \sum_{z \in Z^{Ph}(j)} \boldsymbol{w_{jz}^p} \leq W^p \tag{2.42}$$

$$\boldsymbol{w_{jz}^p} + (SW_{jz}^p \cdot hap_{jz}) \leq ETc_{ijz}^p \cdot hap_{jz}$$

$$i \in I, \{j \mid \gamma(j) = i\}, z \in Z^{Ph}(j) \tag{2.43}$$

$$\sum_{\{j \mid \gamma(j)=i\}} \sum_{z \in Z^{Ph}(j)} \boldsymbol{\mathcal{Y}_{jz}^p} \cdot hap_{jz}^p \geq D_i \qquad i \in \bar{I} \tag{2.44}$$

$$\boldsymbol{w_{jz}^p}, \boldsymbol{\mathcal{Y}_{jz}^p} \geq 0 \qquad j \in J, z \in Z^{Ph}(j)$$

Objective function (2.40) maximizes the expected total yield revenues: the price per tn times the total number of ha times the current crop yield (tn/ha), for all crops planted in the different plots of the field.

Restrictions (2.41) corresponds to the current yield reached after irrigation period p. It is based on the crop water production function (2.38) where term ETa_{ijz}^p is equal to $\boldsymbol{w_{jz}^p} + (SW_{jz}^p \cdot hap_{jz})$, i.e., the amount of water irrigated at period p plus the already existing water that is indicated by the humidity sensor. By equation

(2.39), the maximum water requirements are $ETc^p_{ijz} \cdot ha_{jz}$. Restriction (2.42) is about the available water the farmer can use for the irrigation of its plots during period p. Restrictions (2.43) indicate that the real amount of water ETa^p_{ijz} can not exceed the maximum (or optimal) amount of water ETc^p_{ijz} required by crop i in plot j in physical zone $z \in Z^{Ph}(j)$ at irrigation period p. Restrictions (2.44) determine that current yield reached by the crop $i \in I_0$ at period p must satisfy the demand negotiated beforehand by the farmer.

RTIP is a linear programming that can be solved in an efficient way as we show in Section 2.3.5.

2.3.5 Experimental Results

In this section we empirically show that the HAP methodology is valid and efficient for a real size instance. For R&H-MZ we use the data from a plot called "Quilaco" presented by Cid-Garcia *et al.* (2013). CPP and RTIP use crops data from Cid-Garcia *et al.* (2014) where we use an instance of a field constituted by a set J of 10 plots and a set I of 19 possible crops to be sowed.

The HAP was executed on a Virtual machine with Windows 7 fitted with 1 GB of RAM and a processor Intel Core 2 Duo of 3.06 GHz running on a IMAC equipped with the same processor and 4 GB of RAM. For R&H-MZ and CPP we used the linear integer branch-and-bound algorithm of GAMS/CPLEX 12.2 using default options, except for the optimal criterion fixed at 0. For RTIP we use the linear programming solver of GAMS/CPLEX 12.2 with default parameters. Specific parameters for each stage of the HAP methodology are presented in the following.

Rectangular & Homogeneous Management Zones

In this section two delineations of rectangular and homogeneous management zones are made for each one of the plots. The delineation with chemical soil properties is used in CPP to determine the expected amount of nutrients (fertilizers, pesticides, etc.) that the crops require in the whole production cycle, while the delineation with physical soil properties is used in CPP and RTIP to determine the expected amount of water required by the crops in the whole production cycle and the amount of water required by the crops during each irrigation period, respectively.

The procedure performed in the delineation of management zones is similar for both physical and chemical soil properties. Then, we only present an example for “Quilaco” using as specific chemical soil property the organic matter (OM).

Table 2.13: Coordinates and values of the soil properties for each sample of “Quilaco”. OM is in %, P in $mg\ kg^{-1}$ and SB in $Cmol(+)\ kg^{-1}$.

Sample	Coordinates	Soil properties				Sample	Coordinates	Soil properties			
	(x, y)	pH	OM	P	SB		(x, y)	pH	OM	P	SB
1	0.00, 9.14	5.2	11.8	8.0	5.89	21	297.68, 166.36	5.6	10.4	4.0	8.26
2	48.97, 8.46	5.5	12.8	4.0	7.97	22	253.87, 160.20	5.4	18.7	11.0	8.88
3	97.52, 5.57	5.2	14.9	10.0	7.63	23	206.99, 157.26	5.6	10.5	11.0	6.03
4	150.52, 9.42	5.4	14.0	7.0	11.44	24	158.29, 155.16	5.5	16.8	3.0	9.48
5	201.07, 8.25	5.5	11.2	4.0	6.36	25	105.27, 153.53	5.4	14.8	5.0	7.85
6	250.24, 0.00	5.4	14.7	4.0	9.31	26	56.47, 156.87	5.5	12.6	5.0	5.38
7	298.57, 84.00	5.6	12.5	6.0	10.03	27	6.15, 151.48	5.4	15.1	7.0	6.50
8	249.94, 78.89	5.6	9.6	4.0	7.99	28	6.33, 204.03	5.4	11.7	5.0	5.88
9	208.71, 73.33	5.5	14.3	6.0	8.20	29	58.83, 205.57	5.5	16.0	4.0	8.09
10	160.73, 66.20	5.5	15.0	6.0	9.23	30	108.59, 207.64	5.4	13.8	4.0	8.18
11	102.69, 59.51	5.4	14.5	5.0	6.64	31	159.65, 203.22	5.6	12.6	3.0	7.95
12	53.66, 58.30	5.4	11.1	6.0	6.00	32	206.04, 199.18	5.4	14.4	6.0	7.50
13	2.81, 52.71	5.3	14.1	5.0	5.67	33	255.23, 205.16	5.4	15.4	5.0	8.23
14	6.93, 101.13	5.3	16.3	6.0	5.51	34	303.14, 212.73	5.7	11.2	5.0	9.51
15	58.25, 105.04	5.4	12.7	7.0	6.36	35	278.06, 242.75	5.2	16.6	22.0	7.30
16	104.05, 107.24	5.4	14.2	6.0	7.80	36	208.60, 243.31	5.5	15.6	8.0	9.21
17	156.53, 111.44	5.5	11.4	5.0	6.72	37	158.68, 247.47	5.5	16.1	5.0	9.51
18	204.49, 114.91	5.5	11.5	8.0	6.11	38	108.00, 249.65	5.4	13.9	6.0	6.90
19	250.37, 119.77	5.4	16.7	6.0	8.75	39	58.16, 253.69	5.5	15.4	5.0	9.69
20	296.17, 124.74	5.5	13.5	5.0	7.81	40	12.72, 254.37	5.4	10.7	4.0	7.71

Table 2.13 shows the soil samples of “Quilaco”. This field has 256 meters width

and 305.6 meters long (around 7.82 ha). There have been taken 40 soil samples that are approximately spaced by 50 meters one from each other, so four soil samples are needed to cover an ha. Each soil sample is labeled (first and fourth column of the table) and their positions are translated into a Cartesian map, coordinates (x, y), (second and fifth column of the table). Finally, the information about each chemical soil property is presented: pH, organic matter (OM), phosphorus (P), and sum of bases (SB) determined by the CH3COONH4 method of INIA (2006). A similar table could be presented for the physical soil properties such as field capacity, water holding capacity, and permanent wilting point.

From the thematic map for "Quilaco" of the OM property we determine that the minimum size of a quarter contains a single sample width ($MinWidthQ = 1$) per one sample of length ($MinLengthQ = 1$) since there is a lot of diversity.

After, the possible quarters are generated and labeled by Algorithm 2, in this case we have 588 quarters. Then, Table 2.14 allows to see the structure of the correspondence matrix of "Quilaco" for organic matter, except by the last column that corresponds to the variance of the different soil samples that are contained in quarter with label z.

Most of the fields are not initially rectangular, so the R&H-MZ method inserts dummy soil samples to fill a rectangle where the field can be contained. This is the reason why Table 2.14 is composed of 42 samples. The dummy samples are also equidistant with respect to the others. Nevertheless, their data about the properties is very high with respect to the real samples. This manner, the mathematical model puts these dummy soil samples alone in a zone or with other dummy samples which facilitates their elimination afterwards.

Table 2.14: Correspondence matrix of "Quilaco" for organic matter soil property with the variance of each quarter z.

		Sample point s																	σ_z^2
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	...	42	
Potential quarter z	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	...	0	0.00
	2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	...	0	0.00
	3	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	...	0	0.00
	4	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	...	0	0.00
	5	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	...	0	0.00
	6	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	...	0	0.00
	7	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	...	0	0.00
	8	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	...	0	11.04
	9	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	...	0	1.12
	10	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	...	0	2.42
	11	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	...	0	0.12
	12	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	...	0	0.50
	13	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	...	0	0.00
	14	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	...	0	5.76
	15	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	...	0	4.61
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	...	⋮	⋮
	588	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	...	1	2.07

Table 2.15 presents the experimental results of ϵ-*constraint* method applied to the R&H-MZ for the "Quilaco" instance. First column is the alpha parameter (ϵ value) that determines the homogeneity level in each selected quarter. The higher the α the more homogeneous the management zones. Second column is the number of quarters (zones) used to partitioning the plot (we want to minimize the number of management zones). Last column is the solution time in seconds required by the solver to obtain the optimal solution computed by the branch-and-bound algorithm of GAMS/CPLEX 12.2.

Table 2.15: Experimental results for R&H-MZ.

α	Quarters	Time
0	1	0.296
0.1	3	0.227
0.2	5	0.237
0.3	6	0.291
0.4	7	0.229
0.5	9	0.211
0.6	11	0.241
0.7	14	0.251
0.8	17	0.241
0.9	20	0.231
1	40	0.225

With Table 2.15 we empirically prove that minimizing the number of management zones and maximizing the homogeneity within each zone are conflicting objectives. We can also notice that the computing times are negligible. This implies that we can compute the exact Pareto front in an efficient way which is a remarkable characteristic.

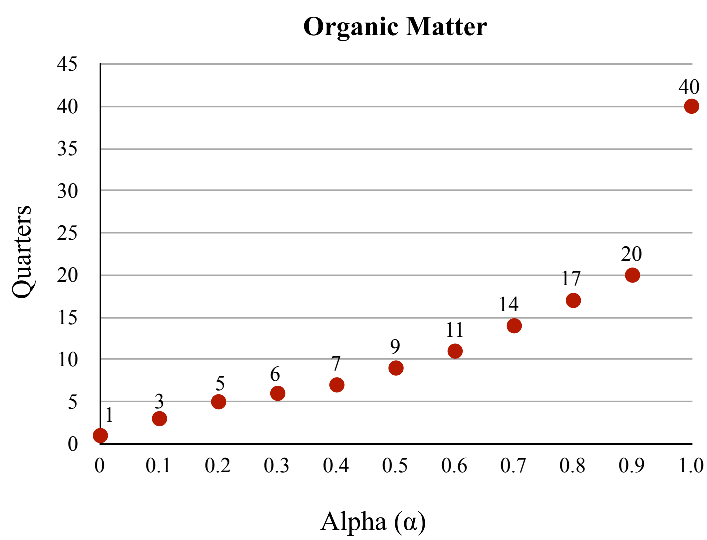


Figure 2.16: Pareto front for "Quilaco" using organic matter (OM) as chemical soil property.

Figure 2.16 shows the exact Pareto front for “Quilaco” using organic matter as chemical soil property. On the x-axis is represented the value of α while on the y-axis corresponds to the number of zones used to partitioning the plot. Here we partition the α rank $[0, 1]$ in subintervals of 0.1. We could easily do a more dense partition.

Once that we have the Pareto front, the next step is to choose the solution from this front that satisfies the farmer's requirements and guarantees homogeneity in each selected quarter. In this case, only solutions with α greater or equal than 0.5 guarantee homogeneity in the selected zones (this value is given by an agricultural expert). In Figure 2.17 is presented the chemical management zones resulting after partitioning the field “Quilaco” using organic matter as soil property and α values of 0.5, 0.7 and 0.9 (left to right maps). We can observe that if α increases then the number of quarters increases too.

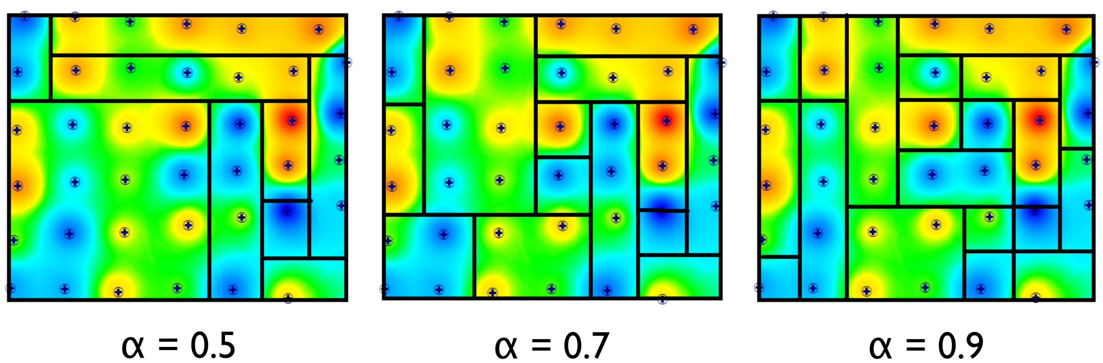


Figure 2.17: Management zones for “Quilaco” using organic matter as chemical soil property and alpha values of 0.5, 0.7, and 0.9.

With R&H-MZ method we have the chemical and physical management zones of all the plots of the field. Then, with this information, HAP decides which crops to plant in each one of the plots.

Crop Planning Problem

In this section we first selected the best delineation of chemical and physical management zones, in each plot of the farmer's field, executing CPP model in each solution of the Pareto front given by R&H-MZ (see Figure 2.16). Once we have the best chemical and physical management zones CPP uses this delineation to generate the optimal crop pattern that maximizes the farmer's profit at the end of the production cycle. In this stage, we use the same crops than Cid-Garcia *et al.* (2014) and 10 plots with their respective best physical and chemical management zones obtained by R&H-MZ.

The total number of ha is 81 and the total expected amount of available water for the whole production cycle is 486,000 m^3 (we assume that we have the maximum limit of water per ha, established by CONAGUA[11]: 6,000 m^3 per ha.) The irrigation costs $Cirr_{jz}$ for each one of the plots j is chosen at random between 1.8 and 2.2 per m^3 (these parameters are based on real data).

General information of plots and the number of their physical and chemical management zones is showed in Table 2.16. First column is the label of the plot,

Table 2.16: General information of plots.

Plot	$Z^{Ph}(j)$	$Z^{Ch}(j)$	ha_j
1	2	4	10
2	4	3	17
3	2	2	4
4	4	1	7
5	1	2	3
6	2	4	9
7	1	2	6
8	2	3	11
9	2	1	4
10	2	4	10

[11] The Mexican national water commission.

second and third column is the number of physical and chemical management zones in each plot and last column is the total number of *ha* in each plot.

Table 2.17: Crop data from spring-summer cycle in the state of Michoacán, Mexico. The 19 crops and their related information were obtained from SAGARPA for the year 2008.

ID	**Crop** i	**Expected Yield** Y_i	**Seed Amount** $Seed_i$	**Sowing Cost** $Cplant_{ijz}$	**Seed Cost** $Cseed_i$	**Expected Benefit** G_i
1	Sesame TCS	0.60	4	2,318.00	10.00	13,681.80
2	Sesame TMF	0.50	4	8,117.91	10.00	13,681.80
3	Onion BMF	40.60	12,500	69,251.08	0.15	3,381.17
4	Green pepper BMF	24.70	12,500	106,121.91	0.15	4,923.99
5	Strawberry BMF	20.40	85,228	74,543.27	0.11	3,943.97
6	Strawberry GMF	20.40	85,228	48,533.07	0.11	3,493.97
7	Corn grain BCF	4.85	25	10,273.02	17.10	4,373.49
8	Corn grain BMF	5.38	25	10,013.49	17.10	4,373.49
9	Corn grain GCF	4.85	25	9,693.02	17.10	4,373.49
10	Corn grain GMF	5.38	25	10,668.41	17.10	4,373.49
11	Corn grain TCF	2.41	25	10,512.40	17.10	4,373.49
12	Sorghum grain BMF	8.31	324	12,022.65	1.50	3,491.25
13	Sorghum grain GMF	8.31	324	7,891.88	1.50	3,491.25
14	Sorghum grain TMF	4.71	324	6,674.43	1.50	3,491.25
15	Red Tomato BMF	38.10	12,500	75,259.74	0.56	2,171.99
16	Red Tomato GMF	38.10	12,500	74,440.68	0.56	2,171.99
17	Green Tomato BCF	16.20	12,300	50,574.63	0.15	3,416.17
18	Green Tomato BMF	17.80	12,300	41,867.90	0.15	3,416.17
19	Green Tomato TCF	2.70	12,300	34,056.14	0.15	3,416.17

In Table 2.17 are presented the crops used in the instance. They correspond to the crops that can be sown in Michoacán, Mexico, during the production cycle of spring-summer (data of 2008 from SAGARPA[12]). First and second columns are

[12]Mexican ministry of agriculture, livestock, rural development, fisheries, and food.

the identification number (ID) and name of the crop i. Third column shows the expected yield Y_i of the crop i in tn/ha at the end of the production cycle. Fourth column is the amount of seeds $Seed_i$ in $unit/ha$ needed to sown crop i. The term *unit* represents *kg*, plants or packages. Sowing costs $Cplant_{ijz}$ in plot j within chemical management zone $z \in Z^{Ch}(j)$ in $\$/ha$ and seed costs $Cseed_i$ in $\$/unit$ of crop i are presented in the fifth and sixth column, respectively. The expected benefit G_i of selling a *tn* of crop i at the end of the production cycle is showed in last column. We assume that the demands that should be satisfied by the farmer of onion BMF, green pepper BMF, corn grain BCF, and red tomato GMF (crops 1, 4, 7, and 16) are all equal to 30. Finally, the stock of seeds $Iseed_i$ for all crops is equal to zero.

When HAP has previous information about the sowing costs $Cplant_{ijz}$ of each crop i then this costs is specific for each chemical zone $z \in Z^{Ch}(j)$ of each plot j. In this research we take the same sowing costs of crops i for all the chemical zones $z \in Z^{Ch}(j)$ of all the plots j.

To calculate the total expected amount of water supplied W_{ijz} to crop i in each plot j in each physical management zone $z \in Z^{Ph}(j)$ during the whole production cycle we use equations (2.31) and (2.32). The parameters to compute these equations were obtained from FAO and INIFAP[13]. Crop coefficient values Kc^v_{ij} of crop i in plot j for phenological stage v at irrigation period p and the duration of the vegetative cycle of the crops were collected from FAO (Allen *et al.*, 2006). Values for crop reference evapotranspiration ETo^v at phenological stage v and for the amount of water stored SW^v_{jz} in plot j in physical management zone $z \in Z^{Ph}(j)$ at phenological stage v were obtained from INIFAP located in Zacatecas, Mexico. These values correspond to averages of previous years.

Table 2.18 shows the total expected amount of water W_{ijz} needed by crop i in each physical management zone $z \in Z^{Ph}(j)$ of plot j after computing equations

[13]The Mexican national institute for forestry, agriculture and livestock. There is a research center INIFAP at every state therefore, producers can get specific information depending on the geographic location of their fields.

(2.31) and (2.32) (data of ETo^v and SW^v_{jz} are averages of the last five years). In this table we only present the first ten crops of Table 2.17 related to two plots with four physical management zones each one. First column is the plot j and second column is the physical management zone $z \in Z^{Ph}(j)$. The surface hap_{jz} in ha of each physical management zone $z \in Z^{Ph}(j)$ is showed in the third column. The total expected amount of water W_{ijz} in needed by crop i during its production cycle in the plot j in each physical management zone $z \in Z^{Ph}(j)$ is presented in the rest of the columns. The instances are generated such that each crop i consumes the same amount of water in the physical management zones $z \in Z^{Ph}(j)$ with the same ID. For example, crop number 3 (onion) consumes 3,418 m^3 in the physical management zone 1 of plots 1 and 2, while the same crop consumes 4,081 m^3 in the physical management zone 4 of plots 1 and 2.

Table 2.18: Total expected amount of water in m^3 supplied by the first 10 crops in each management zone of two plots.

Plot	**Zone**	hap_{jz}	**Crop (ID)**									
j	z		**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	1	4	3,845	3,845	3,418	7,372	7,050	7,050	7,418	7,418	7,418	7,418
1	2	5	3,819	3,819	3,871	7,448	6,710	6,710	8,100	8,100	8,100	8,100
1	3	3	2,389	2,389	3,664	5,190	5,232	5,232	5,200	5,200	5,200	5,200
1	4	3	4,220	4,220	4,081	8,136	7,597	7,597	8,442	8,442	8,442	8,442
2	1	1	3,845	3,845	3,418	7,372	7,050	7,050	7,418	7,418	7,418	7,418
2	2	6	3,819	3,819	3,871	7,448	6,710	6,710	8,100	8,100	8,100	8,100
2	3	6	2,389	2,389	3,664	5,190	5,232	5,232	5,200	5,200	5,200	5,200
2	4	5	4,220	4,220	4,081	8,136	7,597	7,597	8,442	8,442	8,442	8,442

The result of CPP for this instance is an expected income of $877,690.90 at the end of the production cycle. The crops that should be sown in each plot are presented in Table 2.19. First column indicates the plot while second column the crop sown on it. Notice that all the zones of each plot are planted with the same crop but the decision about which crop to plant strongly depends on the chemical and physical characteristics of the zones of the plots.

Table 2.19: Results of CPP. This output is used as input for RTIP.

Plot	Crop
1	Sesame TCS
2	Onion TMF
3	Sesame TCS
4	Corn grain BCF
5	Red Tomato GMF
6	Sesame TCS
7	Sesame TCS
8	Sesame TCS
9	Green pepper BMF
10	Sesame TCS

Recall that CPP is used as a method to chose between the different management zones delineations proposed by R&H-MZ. Here we have shown the CPP with the best management zones, that is, the one that gives the best profits in CPP. With this result, we now consider the operational plan of the irrigation decisions.

Real-Time Irrigation Problem

In this stage it is already known which crops i have been sown in each one of plots j (Table 2.19). The RTIP step of HAP decides the amount of water to be supplied on each plot j during each irrigation period p to maintain the yields as high as possible at the end of the production cycle.

Table 2.20 presents the parameters needed for RTIP at irrigation period $p = 1$. First and second columns correspond to the plot j and the physical management zone $z \in Z^{Ph}(j)$. Third column is the yield response factor $Ky^p_{\gamma(j)}$ of crop i sown in plot j at irrigation period p (plots with the same crops and planted at the same time have the same factor), this parameter is obtained from FAO. Fourth column is the amount of stored water SW^p_{jz} in m^3/ha in each physical management zone $z \in Z^{Ph}(j)$ of plot j before irrigating at period p. This information is given by the humidity sensors (in this research we used WATERMARK 200SS-V sensors). Fifth column is the amount of water ETc^{p}_{ijz} in m^3/ha needed by crop i in each physical management zone $z \in Z^{Ph}(j)$ of plot j at current irrigation period p. This data is calculated with equations (2.31) and (2.32) using information from INIFAP. Last column is the maximum yield $\overline{\mathcal{Y}}^{p-1}_{jz}$ in tn/ha reached in each physical management zone $z \in Z^{Ph}(j)$ of plot j in the previous irrigation period $p-1$. When p is equal to zero, $\overline{\mathcal{Y}}^{p-1}_{jz}$ takes the value of the expected harvest yield of crop under an optimal growing environment (this value is given by INIFAP or FAO).

Table 2.20: Parameters of RTIP at irrigation period 1.

Plot	**Zone**	**Yield response factor**	**Stored Water**	**Required Water**	**Maximum Yield**
j	z	$Ky^{p}_{\gamma(j)}$	SW^{p}_{jz}	ETc^{p}_{ijz}	$\overline{\mathcal{Y}}^{p-1}_{jz}$
1	1	0.3	100	393.4	0.6
1	2	0.3	100	424.9	0.6
2	1	0.45	100	1040.2	40.6
2	2	0.45	100	1123.5	40.6
2	3	0.45	100	1010.8	40.6
2	4	0.45	100	1250.2	40.6
3	1	0.3	100	393.4	0.6
3	2	0.3	100	424.9	0.6
4	1	0.4	100	786.8	4.85
4	2	0.4	100	849.8	4.85
4	3	0.4	100	884.1	4.85
4	4	0.4	100	972.5	4.85
5	1	0.4	100	1219.2	38.1
6	1	0.3	100	393.4	0.6
6	2	0.3	100	424.9	0.6
7	1	0.3	100	393.4	0.6
8	1	0.3	100	393.4	0.6
8	2	0.3	100	424.9	0.6
9	1	1.1	100	706.4	24.7
9	2	1.1	100	789.9	24.7
10	1	0.3	100	393.4	0.6
10	2	0.3	100	424.9	0.6

We consider six irrigation periods for our experimental instance. Table 2.21 shows the experimental results for RTIP at period $p = 1$ with available water of 81,000 m^3 which corresponds to a sixth of the total expected amount of water by the production cycle (486,000 m^3). First and second columns represent the plot and the physical management zone. Third column is the amount of water supplied to the crop in m^3 at current irrigation period. Fourth column indicates whether the crop was irrigated at optimal level or not. Finally, last column is the current expected

Table 2.21: Experimental results of RTIP at irrigation period 1.

Plot j	**Zone** z	**Supplied Water** $\boldsymbol{w^p_{jz}}$	**Irrigation Level**	**Current Yield** $\boldsymbol{\mathcal{Y}^p_{jz}}$
1	1	880.20	Optimal	0.60
1	2	2274.30	Optimal	0.60
2	1	3760.80	Optimal	40.60
2	2	40940	Optimal	40.60
2	3	6375.60	Optimal	40.60
2	4	2300.40	Optimal	40.60
3	1	586.80	Optimal	0.60
3	2	649.80	Optimal	0.60
4	1	686.80	Optimal	4.85
4	2	749.80	Optimal	4.85
4	3	3136.40	Optimal	4.85
4	4	872.50	Optimal	4.85
5	1	3357.60	Optimal	38.10
6	1	1760.40	Optimal	0.60
6	2	974.70	Optimal	0.60
7	1	1760.40	Optimal	0.60
8	1	14670	Optimal	0.60
8	2	1949.40	Optimal	0.60
9	1	1819.20	Optimal	24.70
9	2	689.90	Optimal	24.70
10	1	1173.60	Optimal	0.60
10	2	1949.00	Optimal	0.60

yield in tn/ha reached after irrigating the plot. Optimal solutions were computed in less than 1 second.

At this period, the total amount of water is enough for irrigating all the crops at optimal level. The total amount of water supplied to irrigate all the crops is only 43,268.6 m^3 which corresponds to 53.4% of the total available water of the period. Therefore, savings on water are made.

Table 2.22 presents the experimental results of the RTIP throughout the whole production cycle. First column indicates the plot j and the second column the physical management zone $z \in Z^{Ph}(j)$. Third column is the expected harvest yield of crop i planted in plot j under a growing environment, this is the parameter of maximum yield by crop i of plot j only for the first period ($Ym^1_{\gamma(j)}$). Fourth column indicates if the irrigation level at period 1 (IL^1) was optimal or not ("–" means that the crop was not irrigated to optimal level). Fifth column shows the current yield of crop i of plot j ($Ya^1_{\gamma(j)}$) reached after irrigate at period 1, $Ya^1_{\gamma(j)}$ is the parameter $Ym^2_{\gamma(j)}$ for the second period. Columns 6 and 7 are the same as above but for period 2, and so on until period 6. At periods 1 and 2 the crops are in their initial phenological stages so they do not consume too much water. All the crops are irrigated at optimal level and reach their maximum expected yield at the end of these periods. At period 3 there is not enough water to irrigate all crops to optimum level and the current expected yield of plot 2 in zones 2, 3, and 4, together with the current yield of plot 4 in zone 4, decrease considerably with respect to the maximum yield.

Since there is not enough water to irrigate the crops to optimal level on their flowering and yield formation stages, the current expected yield of plot 2 in zones 2, 3, and 4 decrease again at period 4 with respect to maximum yield of period 3. At period 5, the current yield of plot 2 in zones 2, 3 and 4, the current yield of plot 5 in zone 1, and the current yield of plot 9 in zones 1 and 2 decrease considerably. At period 6 the crops are in their final phenological stage so they do not consume too much water and all of them are irrigated again to optimal level.

Table 2.22: Experimental results of RTIP for the six irrigation periods of the production cycle.

Plot	Zone	$Ym^1_{\gamma(j)}$	p1		p2		p3		p4		p5		p6	
j	z		IL^1	$Ya^1_{\gamma(j)}$	IL^2	$Ya^2_{\gamma(j)}$	IL^3	$Ya^3_{\gamma(j)}$	IL^4	$Ya^4_{\gamma(j)}$	IL^5	$Ya^5_{\gamma(j)}$	IL^6	$Ya^6_{\gamma(j)}$
1	1	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6
1	2	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6
2	1	40.6	Opt	40.6	Opt	40.6	Opt	40.6	Opt	40.6	Opt	40.6	Opt	40.6
2	2	40.6	Opt	40.6	Opt	40.6	–	25.3	–	14.21	–	9.86	Opt	9.86
2	3	40.6	Opt	40.6	Opt	40.6	–	22.33	–	12.28	–	10.23	Opt	10.23
2	4	40.6	Opt	40.6	Opt	40.6	–	22.33	–	12.28	–	9.08	Opt	9.08
3	1	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6
3	2	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6
4	1	4.85	Opt	4.85	Opt	4.85	Opt	4.85	Opt	4.85	Opt	4.85	Opt	4.85
4	2	4.85	Opt	4.85	Opt	4.85	Opt	4.85	Opt	4.85	Opt	4.85	Opt	4.85
4	3	4.85	Opt	4.85	Opt	4.85	Opt	4.85	Opt	4.85	Opt	4.85	Opt	4.85
4	4	4.85	Opt	4.85	Opt	4.85	–	0.9	Opt	0.9	Opt	0.9	Opt	0.9
5	1	38.1	Opt	38.1	Opt	38.1	Opt	38.1	Opt	38.1	–	13.88	Opt	13.88
6	1	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6
6	2	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6
7	1	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6
8	1	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6
8	2	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6
9	1	24.7	Opt	24.7	Opt	24.7	Opt	24.7	Opt	24.7	–	6.512	Opt	6.512
9	2	24.7	Opt	24.7	Opt	24.7	Opt	24.7	Opt	24.7	–	16.854	Opt	16.854
10	1	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6
10	2	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6	Opt	0.6

The model must comply with the established demand in the CPP (sesame, green pepper, corn grain and red tomato), so these crops have priority over the others. The model can let die crops that do not have a fixed demand even if those crops would generate more profit for the farmer. Table 2.23 shows the final yield reached by each crop after each irrigation period. It is verified that the demand established in the CPP is satisfied for each crop at the end of the production cycle (last irrigation period).

Table 2.23: Final total yield reached by the crops after each irrigation period.

Irrigation Period	Final Yield (tn)				
	Sesame TCS	Green pepper BMF	Corn grain BCF	Red Tomato GMF	Onion TMF
1	30	98.80	33.95	114.30	690.20
2	30	98.80	33.95	114.30	690.20
3	30	98.80	30.00	114.30	464.60
4	30	98.80	30.00	114.30	329.80
5	30	36.39	30.00	41.65	291.52
6	30	36.39	30.00	41.65	291.52

Finally, in Table 2.24 is presented the expected profit achieved by the farmer at each irrigation period after watering the crops (notice that the sowing costs are not considered here). First column indicates the irrigation period. Second column is the amount of available water in m^3 for irrigating crops (AW), and third column is the real amount of irrigated water in m^3 on crops (IW). Fourth column is the percentage (%) of irrigated water (IW) with respect to the total available, and the last column is the expected profit in $ achieved in the period. In the first two periods the crops were irrigated at optimal level therefore the farmer's expected profit remained at 100%. However, in periods 3, 4 and 5, there is a greater need of water with respect to the total amount of available water in each irrigation period.

Table 2.24: Final expected profit reached by the farmer at each irrigation period.

Period	AW (m^3)	IW (m^3)	IW (%)	Profit ($)
1	81,000	43,269	53.4	3,627,366.19
2	81,000	43,269	53.4	3,627,366.19
3	81,000	81,000	100	2,847,269.31
4	81,000	81,000	100	2,391,531.03
5	81,000	81,000	100	1,796,976.97
6	81,000	49,449	61.1	1,796,976.97

Water needed by the crops was not 100% satisfied causing a decrease of 50.46% in the farmer's profit that would never be recovered in despite that in the period 6 all crops were irrigated at 100% (see Figure 2.18). So, at the end of the production cycle the profit is only 49.54% with respect to the total expected profit at the begin of the production cycle.

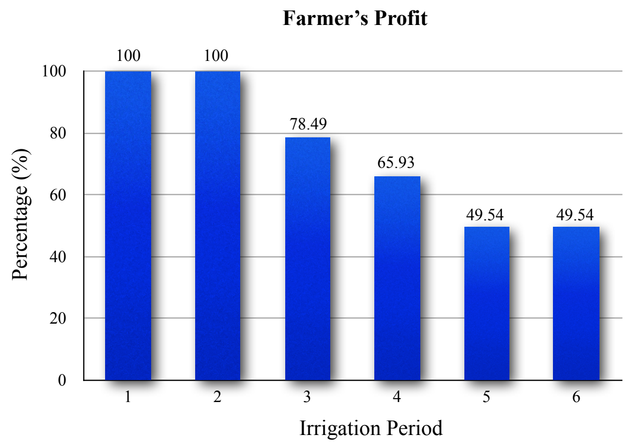


Figure 2.18: Percentage of farmer's profit after each irrigation period.

In Figure 2.19 is showed the yield reached in each physical management zone $z \in Z^{Ph}(j)$ of plot j after each irrigation period p (periods 1, 3, 4 and 6).

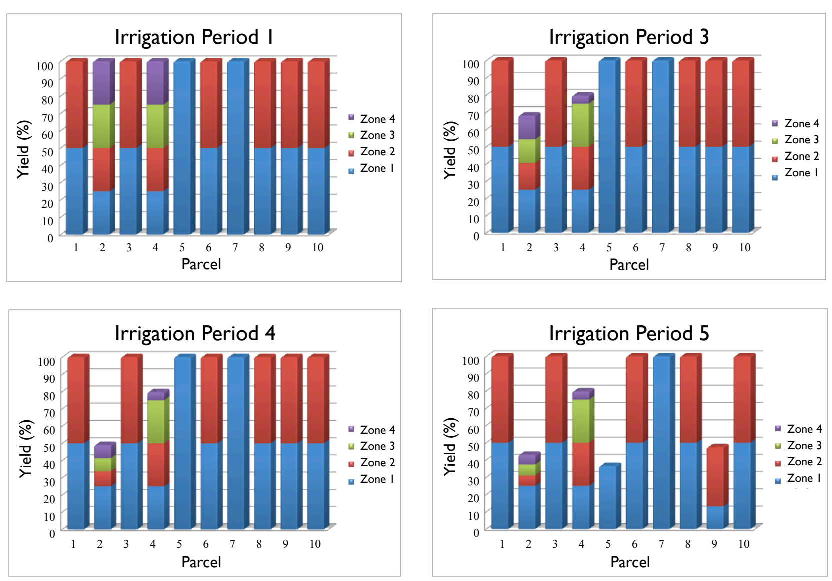


Figure 2.19: Yield reached in each plot after each irrigation period (periods 1, 3, 4 and 6).

RTIP guarantees to supply only the amount of water needed to satisfy the water requirements of crops and avoid wastage. Thus, water can be stored to future irrigation periods. Moreover, the farmer now has a decision tool that is relevant when water shortages arise.

2.3.6 Conclusions of Section

Physical and chemical soil properties existing in agricultural production plots are an important characteristic that should be considered in the agricultural planning process. Chemical soil properties affect on the application of inputs (fertilizers, pesticides, etc.), while physical soil properties are related to the water use.

In this work we propose a new approach we name as Hierarchical Agriculture Planning (HAP) for helping the decision makers (the farmers) to plan and operate their plots in order to avoid wastage and to maximize their benefits considering the soil diversity. In this hierarchical approach the farmers start by delineating the field into rectangular and homogeneous site-specific chemical and physical management zones to face within-field variability. Then the farmers assign a crop to the different plots to obtain the best profit at the end of the production cycle (CPP). Finally, in each irrigation period the farmer must decide how much and which plots must be watered such to maximize the total final yields (RTIP).

Experimental results show that the new hierarchical approach is efficient and practical since optimal solutions are obtained in seconds.

2.3.7 Acknowledgements

We are grateful to INIFAP for much of the data that we used in this study.

Chapter 3

Two-Dimensional Packing Problems

3.1 Positions and Covering: A New Two-stage Methodology to Obtain Exact Solutions for the 2D-Bin Packing Problem

This research produced the article: Cid-Garcia, N. M., & Rios-Solis, Y. A. (2015). Positions and Covering: A two-stage methodology to obtain exact solutions for the 2d-bin packing problem, *working paper.*

Abstract

In this section is presented an exact methodology to solve the two-dimensional bin packing problem. In this classical combinatorial NP-hard problem, a given set of small rectangular items has to be packed into a set of rectangular bins with the objective of minimizing the number of bins used to pack all the items. The methodology is based on a two-stage procedure where, first is generated in a pseudo-polynomial way a set of valid positions in which each item can be packed into a bin. Then, is used a set-covering formulation to select the optimal non-overlapping configuration of items for each bin. The best algorithms in the literature are heuristic, thus the importance of this research.

3.1.1 Introduction

The two-dimensional bin packing problem (2D-BPP) consists in packing, without overlapping, a set of two-dimensional rectangular items into the minimum number of two-dimensional rectangular bins (Gonçalves y Resende, 2011, 2013; Lodi *et al.*, 2002a). All the bins are identical with width W and height H, and each item i has its specific width w_i, height h_i, and demand d_i for $i = 1, \ldots, n$. We assume that all input data are positive integers and that $w_i \leq W$ and $h_i \leq H$ for $i = 1, \ldots, n$.

Figure 3.1 shows the optimal configuration of two bins for an instance from Martello y Vigo (1998), with more than 50 items that had to be packed. The white spaces are lefts-overs.

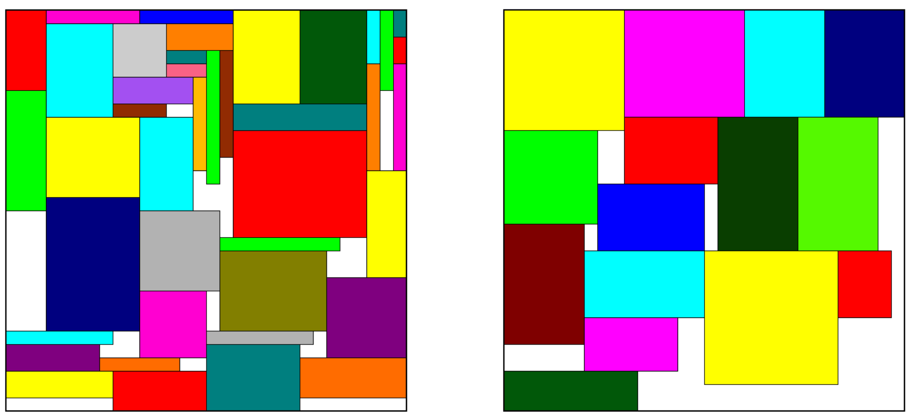


Figure 3.1: Example of two bins with more than 50 items. Instance from Martello y Vigo (1998).

The 2D-BPP is strongly NP-hard since it generalizes the one-dimensional bin packing problem (Lodi *et al.*, 2002a; Martello y Vigo, 1998), and according to the typology for cutting and packing problems proposed by Wäscher *et al.* (2007), it belongs to the class of 2D-Single Bin-Size Bin Packing Problems.

This problem has great significance for many industrial applications, where rectangular figures are cut from larger rectangular sheets of textiles, glass, steel, wood, or paper (Hopper y Turton, 2001b; Lodi *et al.*, 2002a). Moreover, recent applications in the delineation of rectangular management zones in agricultural fields have led to some of the preliminary ideas of the methodology presented here (Cid-Garcia *et al.*, 2013; Albornoz *et al.*, 2015).

Bin packing problems are similar to cutting stock problems (Haessler y Sweeney, 1991). Indeed, in bin packing problems the item set is strongly heterogeneous, that is, there are many types of items with small demands. In the worst case, all the items are different and their demand is equal to one. In cutting stock problems, the item set is weakly heterogeneous, that is, there are a few types of items and their demands are larger (Wäscher *et al.*, 2007; Silva *et al.*, 2010). Excellent surveys of packing and cutting problems are presented by Haessler y Sweeney (1991); Dowsland y Dowsland (1992); Lodi *et al.* (2002a), and Lodi *et al.* (2002b).

Most of the literature refers to the case where the items have an orientation constraint, so that one is not allowed to rotate them. However, the more general case where 90° rotations are allowed has not been broadly studied (Gonçalves y Resende, 2013; Boschetti y Mingozzi, 2003b). In the present paper, we consider both cases.

Our two-stage methodology for solving the 2D-BPP is called Positions and Covering (P&C for short). Given an instance of 2D-BPP, we establish all possible positions where each item could be placed into a bin. This *preprocessing* is the key point of the P&C. Then, the total number of bins K is fixed to a lower bound and P&C solves a set-covering model to solve a decision version of the 2D-BPP: is there a non-overlapping packing of the n items into the K bins? If there is a feasible solution, then K is the optimal value of 2D-BPP, otherwise, K is increased by one and P&C iterates.

There are some studies that seek optimal solutions as we do. Christofides y Whitlock (1977) propose a Lagrangian relaxation to solve an integer linear programming model based on a discrete representation of the geometric space. Martello y Vigo (1998) introduce combinatorial lower bounds and present an exact branch-and-bound approach. In Fekete y Schepers (1997, 2004); Fekete *et al.* (2007), there is defined a graph describing the overlaps of the items in the container from the projection of the items on each orthogonal axis.

Another paper that is close to ours is from Beasley (1985b) where the author develops a Lagrangian relaxation of a binary integer programming formulation to obtain an exact solution for the two-dimensional non-guillotine cutting problem. A guillotine cut splits a block into two smaller blocks, where the slice plane is parallel to one side of the initial block. In this paper we do not consider the guillotine cut constraint. In Beasley (1985a), the author presents different algorithms based on dynamic programming for unconstrained two-dimensional guillotine cutting.

The best algorithms in the literature are heuristic, thus the importance of our work consists in validating that the best solutions found by these approaches, for some classes of the instance benchmarks in the literature, are indeed optimal. In Monaci y Toth (2006), the authors present a two-stage heuristic method where column generation and Lagrangian methods for a set-covering model are used. A Guided Local Search (GLS) is proposed by Faroe *et al.* (2003). It starts with a greedy heuristic to obtain an upper bound on the number of bins. Then, the algorithm iteratively decreases the number of bins, each time searching for a feasible packing of the boxes using GLS. Boschetti y Mingozzi (2003a,b) propose a heuristic procedure to obtain new lower and upper bounds for the 2D-BPP. Lodi *et al.* (2004) implement a Tabu Search proposed by Lodi *et al.* (1999a,b,c) for general multi-dimensional bin packing problems. Parreño *et al.* (2010) developed a Greedy Randomized Adaptive Search Procedure. Gonçalves y Resende (2011, 2013) have developed genetic algorithms for the 2D-BPP. To the best of our knowledge these algorithms yield the best results so far in terms of quality.

The rest of this paper is organized as follows. In Section 3.1.2 we present our two-stage methodology P&C to give exact solutions for the 2D-BPP. In Section 3.1.3 we validate that the grid used by P&C yields indeed the optimal solution. In Section 3.1.4 we extend the P&C methodology for the case where 90° rotation of the items is allowed. In Section 3.1.5 we experimentally validate the P&C methodology on a set of instances that has been broadly used in the literature so we are able to compare P&C with the best algorithms known so far. Finally, in Section 3.1.6 we make some concluding remarks.

3.1.2 Materials and Methods

The Positions and Covering Methodology for the 2D-BPP

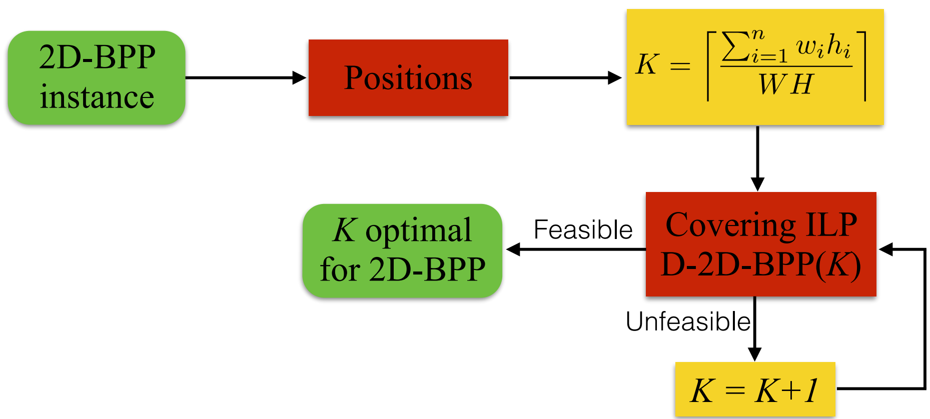


Figure 3.2: Scheme of the P&C methodology for the 2D-BPP.

Figure 3.2 schemes the P&C methodology. Given an instance of the 2D-BPP, the P&C establishes all the possible *positions* where each item could be placed into a bin (more details in Section 3.1.2, where we will prove that the total number of positions is of pseudo-polynomial size). Then, the total number of bins K is fixed:

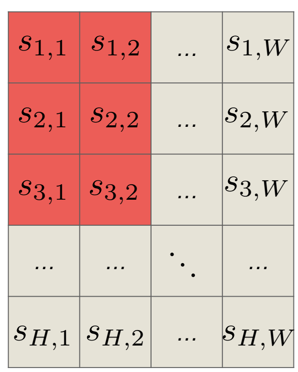
$$K = \left\lceil \frac{\sum_{i=1}^{n} w_i h_i}{WH} \right\rceil,$$

that corresponds to the total area of the items divided by the area of a bin, that is, we assume that the items are perfectly packed (although the best lower bound for each instance can be used). The P&C considers the decision version of the 2D-BPP, that we name as D-2D-BPP(K): is there a non-overlapping packing of the n items into K bins? The P&C exactly solves a covering integer linear programming for D-2D-BPP(K) (see Section 3.1.2). If D-2D-BPP(K) is feasible, then K is the optimal value for 2D-BPP. Otherwise K is increased by one and we solve the covering model again. Notice that $K \leq n$, since in the worst case we need n bins to pack all the items.

Naturally, better bounds and other ways of exploring the feasible set of the number of bins could improve the number of iterations of P&C. Some papers related to this task are Lodi *et al.* (1999a); Boschetti y Mingozzi (2003a,b). In the rest of this section we describe in detail the different stages of the P&C methodology for the 2D-BPP.

The Position stage of P&C

The main idea of the Positions stage is to generate the dominant set of *valid positions* where an item can be placed into a bin. Indeed, from the infinite set of all the positions that an item can take in a bin, we only construct a finite set that guarantees the optimality of the solution. Notice that we are not enumerating or forming the different patterns of the bins.



$$\gamma(i_{1,1}) = \{s_{1,1}, s_{1,2}, s_{2,1}, s_{2,2}, s_{3,1}, s_{3,2}\}$$

Figure 3.3: Item i of size 2×3 in position $i_{1,1}$.

Let us consider for the moment a single $H \times W$ bin. The first step in Positions is to delineate a Cartesian grid inside the bin, that is, a regular tessellation of the 2-dimensional Euclidean space by congruent unit squares whose coordinates are integer. The ordered pair (1,1) describes a unit square placed at the top left corner of the bin. For each $l = 1, \ldots, H$ and $p = 1, \ldots, W$, $s_{l,p}$ is the unit square placed at

(l, p) (see Figure 3.3). Moreover, let $i_{h,w}$ be the position of item i when placed with its top left corner on the unit square $s_{h,w}$, and let $\gamma(i_{h,w})$ indicate the unit squares of the grid covered by $i_{h,w}$, that is, $\gamma(i_{h,w}) = \{s_{l,p} | l = h, \cdots, h + h_i - 1 \text{ and } p = w, \cdots, w + w_i - 1\}$, for $h \leq H - h_i + 1$ and $w \leq W - w_i + 1$. In Figure 3.3 we have in red the item i of size 2×3 placed on $s_{1,1}$, so $\gamma(i_{1,1}) = \{s_{1,1}, s_{1,2}, s_{2,1}, s_{2,2}, s_{3,1}, s_{3,2}\}$.

In Section 3.1.3 we show that the set of valid positions determined by the unitary grid delineated on a bin is sufficient to obtain the optimal solution.

With *Positions*(i) shown in Algorithm 3, we determine all the different valid positions where a specific item i could be placed in the bin and the unit squares of the bin's grid it covers. We start by positioning the top left corner of item i on the unit square $s_{1,1}$. In step 3 we set the number of unit squares that are covered by this item in the current position. Let C be a correspondence matrix with its rows representing the unit squares $s_{l,p}$ for $l = 1, \cdots, H$, $p = 1, \cdots, W$ and its columns being the valid positions $i_{w,h}$ for every i, $h \leq H - h_i + 1$, and $w \leq W - w_i + 1$. An entry of C, $c^i_{h,w}(s_{l,p}) = 1$ if item i in position $i_{h,w}$ covers the unit square $s_{l,p}$. It is 0 otherwise. Then, we shift the item to the next $s_{h,w}$ and analyze its position.

Algorithm 3 *Positions(i)*

1: **for** $h \leq H - h_i + 1$ **do**
2:     **for** $w \leq W - w_i + 1$ **do**
3:         compute $\gamma(i_{h,w})$ for item i in position $i_{h,w}$
4:         **for all** unit square $s_{l,p}$ **do**
5:             **if** $s_{l,p} \in \gamma(i_{h,w})$ **then**
6:                 $c^i_{h,w}(s_{l,p}) = 1$
7:             **else**
8:                 $c^i_{h,w}(s_{l,p}) = 0$
9:             **end if**
10:         **end for**
11:     **end for**
12: **end for**

In this manner, we obtain a set of valid positions that an item i can take in a bin, as shown in Figure 3.4 for a bin of size 5×4 and an item of size 3×2.

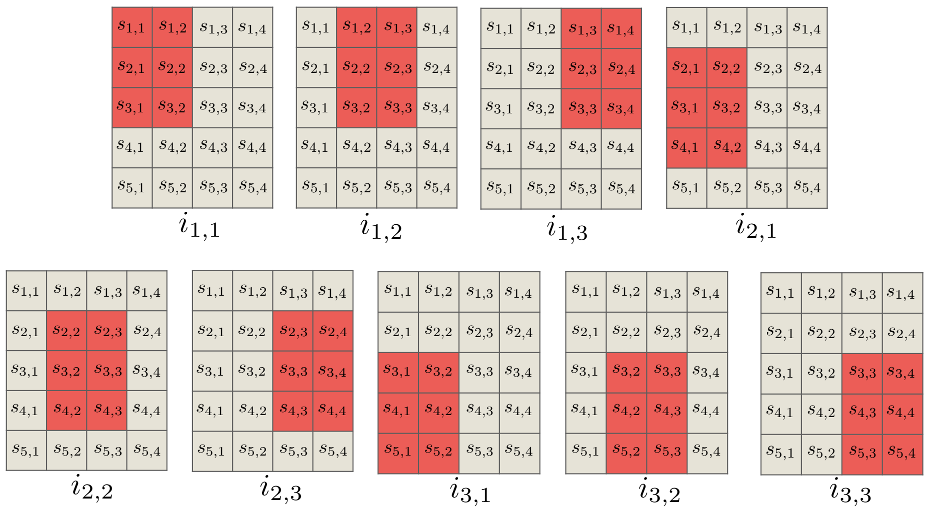


Figure 3.4: All the different positions that an item of size 3×2 can take in a bin of size 5×4.

In Table 3.1 we show the correspondence matrix C related to Figure 3.4. Notice that $i_{1,4}$ is not a valid position for an item of size 3×2 because it would exceed the edge of the bin if it is placed with its left top corner on unit square $s_{1,4}$. Thus, column $i_{1,4}$ is not part of C for this item.

Table 3.1: Correspondence matrix C: columns are the unit squares $s_{l,p}$ and rows are the valid positions $i_{w,h}$. An entry $c^i_{h,w}(s_{l,p}) = 1$ if item i in valid position $i_{h,w}$ covers unit square $s_{l,p}$, 0 otherwise.

	$s_{1,1}$	$s_{1,2}$	$s_{1,3}$	$s_{1,4}$	$s_{2,1}$	$s_{2,2}$	$s_{2,3}$	$s_{2,4}$	$s_{3,1}$	$s_{3,2}$	$s_{3,3}$	$s_{3,4}$	$s_{4,1}$	$s_{4,2}$	$s_{4,3}$	$s_{4,4}$	$s_{5,1}$	$s_{5,2}$	$s_{5,3}$	$s_{5,4}$
$i_{1,1}$	1	1	0	0	1	1	0	0	1	1	0	0	0	0	0	0	0	0	0	0
$i_{1,2}$	0	1	1	0	0	1	1	0	0	1	1	0	0	0	0	0	0	0	0	0
$i_{1,3}$	0	0	1	1	0	0	1	1	0	0	1	1	0	0	0	0	0	0	0	0
$i_{2,1}$	0	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	0	0	0	0
$i_{2,2}$	0	0	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	0	0	0
$i_{2,3}$	0	0	0	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	0	0
$i_{3,1}$	0	0	0	0	0	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0
$i_{3,2}$	0	0	0	0	0	0	0	0	0	1	1	0	0	1	1	0	0	1	1	0
$i_{3,3}$	0	0	0	0	0	0	0	0	0	0	1	1	0	0	1	1	0	0	1	1

The next theorem shows that the number of valid positions of the items in a bin is a pseudo-polynomial number.

Theorem 1. *The number of different valid positions that all the items can take in the Cartesian grid of a bin with size $H \times W$ is bounded by*

$$\frac{W(W+1)H(H+1)}{2}.$$

Proof. Recall that all input data is integer. Since there are n different items, we have that the number of different positions in a bin is

$$\sum_{i=1}^{n}(W - w_i + 1)(H - h_i + 1).$$

In the worst case, we have all possible different sizes of widths and heights, that is,

$$\begin{aligned}\sum_{w=1}^{W}\sum_{h=1}^{H} \quad & H(W - w + 1)(H - h + 1)\\ = \quad & (W^2 - \tfrac{W(W+1)}{2} + W)(H^2 - \tfrac{H(H+1)}{2} + H)\\ = \quad & \left(\tfrac{W(W+1)}{2}\right)\left(\tfrac{H(H+1)}{2}\right).\end{aligned}$$

Note that we use the partial sum of the first integers given by the triangular number.

□

The Position stage must be only computed once in the P&C methodology.

The Covering stage of P&C

Now that we have the correspondence matrix C obtained by the Positions stage, we aim to find a feasible covering of the bins with the different positions that the items can take in the K bins, that is, we aim to solve the decision problem D-2D-BPP(K).

Lets now consider a fixed number K of bins. Since all the bins are equal, then we have a unit square $s_{l,p}$ in each bin k, for $k = 1, \cdots, K$, and every item i has the same set of specific positions in each bin k since all the bins have the same size.

We state the following decision variables:

$$x_{h,w}^{i,k} = \begin{cases} 1 & \text{if item } i \text{ in valid position } i_{h,w} \text{ is placed on bin } k, \\ 0 & \text{otherwise.} \end{cases}$$

The following covering problem solves the decision problem D-2D-BPP(K):

$$\sum_{i=1}^{n} \sum_{h=1}^{H-h_i+1} \sum_{w=1}^{W-w_i+1} x_{h,w}^{i,k} c_{h,w}^{i}(s_{l,p}) \leq 1 \quad \forall l = 1, \ldots, H, \forall p = 1, \ldots, W, \quad \forall k = 1, \ldots, K \tag{3.1}$$

$$\sum_{k=1}^{K} \sum_{h=1}^{H-h_i+1} \sum_{w=1}^{W-w_i+1} x_{h,w}^{i,k} \geq d_i \quad \forall i = 1, \ldots, n \tag{3.2}$$

$$x_{h,w}^{i,k} \in \{0,1\} \quad \forall h = 1, \ldots, H, \forall w = 1, \ldots, W, \quad \forall k = 1, \ldots, K, \forall i = 1, \ldots, n.$$

Notice we do not force any objective function: indeed, any feasible solution that places all n items in the K bins is desirable. The restrictions (3.1) ensure that each unit square of each bin is covered by only one item. In this way, overlapping of the items is avoided. The constraints (3.2) ensure that the demand for every item i is satisfied.

If there is no feasible solution for the previous covering model, the P&C methodology increases K by one and iterates again until a feasible solution is found, that is, until all items can be placed in the bins, which corresponds to the optimal solution.

3.1.3 Could a denser grid yield a better solution?

An important question arises: could a better solution be found if the grid delineation in the bins is denser? Indeed, an item could not only be placed with its top corner on the integer points of the Cartesian grid, as we have proposed. By Theorem 2 and Corollary 1 hereafter there is no other grid that could yield a better solution to the 2D-BPP since the packing in this non-Cartesian grid can be transformed into a packing in a bin with a Cartesian grid. In the right bin of Figure 3.5 we have the packing with the Cartesian grid obtained from the one in the left that has a grid of half unit squares.

Theorem 2. *Any packing of the items in a bin can be transformed into a packing where all vertices of the items coincide with the Cartesian grid delineated in the bin.*

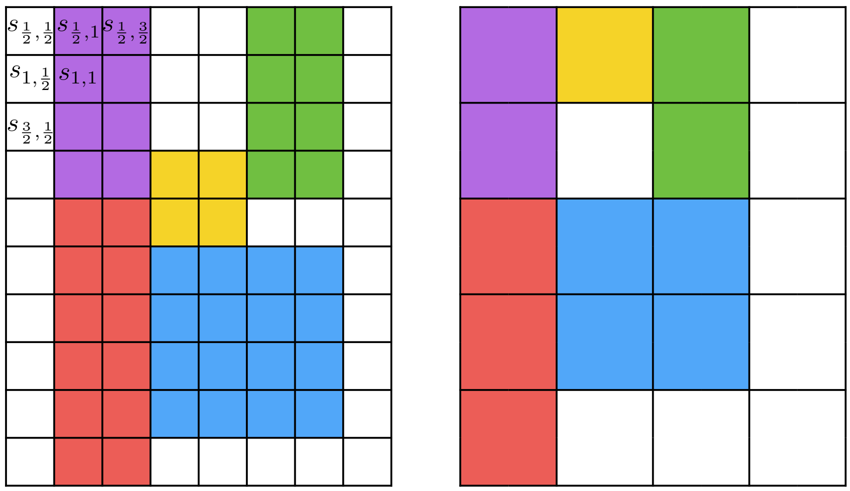


Figure 3.5: Left bin has a half unit square grid where five items are packed. The right bin has the same items but they are now placed on the Cartesian grid with the help of Algorithm 4.

Proof. Recall that W, H, w_i, and h_i for all items i are integer numbers. Let $ItemList$ be a non-decreasing ordered list of the items with respect to their position, so items $i_{l,p}$ are first arranged with respect to l and then to p. In Figure 3.5 we can see on the left side a grid that is not a Cartesian one and the items placed on the grid: P (purple), G (green), Y (yellow), R (red), and B (blue). Therefore, $ItemList = \{P_{\frac{1}{2},1}, G_{\frac{1}{2},3}, Y_{2,2}, R_{\frac{5}{2},1}, B_{3,2}\}$.

With Algorithm 4, we can transform a packing in a bin with a non-Cartesian grid to a packing in the same bin but with a Cartesian grid. In steps 2 and 5, $first(ItemList)$ represents the first item of the ordered list and $last(ItemList)$ its last one. In step 3, each element of $ItemList$ is moved until it reaches the top of the bin or until it is next to another item. Finally, in step 6, each item is moved to the left of the bin up to the edge or next to another item.

Algorithm 4 Packing on the Cartesian grid

1: INPUT: Ordered list $ItemList$
2: **for all** $i = first(ItemList)$ to $n = last(ItemList)$ **do**
3: Move i vertically to the upper border of the bin or to the bottom of an item
4: **end for**
5: **for all** $i = first(ItemList)$ to $n = last(ItemList)$ **do**
6: Move i horizontally to the left border of the bin or to the right side of an item
7: **end for**

Algorithm 4 is clearly finite and its correctness is based on the fact that all the widths and highs of the items are integers, as are the dimension of the bins as well. □

Corollary 1. *The number of bins needed to pack all the items obtained by P&C is the optimal one.*

Proof. By Theorem 2, any configuration that does not use a Cartesian grid can be transformed into one that does. So, if there is a solution that has fewer bins than P&C, it is because it uses a different grid for the bins and this leads to a contradiction. □

3.1.4 2D-BPP WITH POSSIBLE ROTATION OF THE ITEMS BY 90

In the 2D-BPP, the items cannot be rotated, that is, an item of size 3×5 is different from one of 5×3. Nevertheless, in many applications these two items would be exactly the same. The P&C methodology can be adapted to the case when rotations of the items by 90^o is allowed.

First, the Positions stage of P&C must be extended. Recall that $i_{h,w}$ is the valid position of item i when placed with its top left corner on unit square $s_{h,w}$.

Now, $\overline{i_{h,w}}$ will indicate the valid position of the item i rotated by 90^o with its top left corner on unit square $s_{h,w}$. In this way, Algorithm 3, which enumerates all the positions of i, can be easily adapted. Notice that the incidence matrix C can almost double the number of its columns. Let $\overline{c^i_{h,w}}$ correspond to the new columns of matrix C related to position $\overline{i_{h,w}}$ of item i. Figure 3.6 shows the extended positions for an item of size 3×2 in a bin of size 5×4 when rotation is allowed.

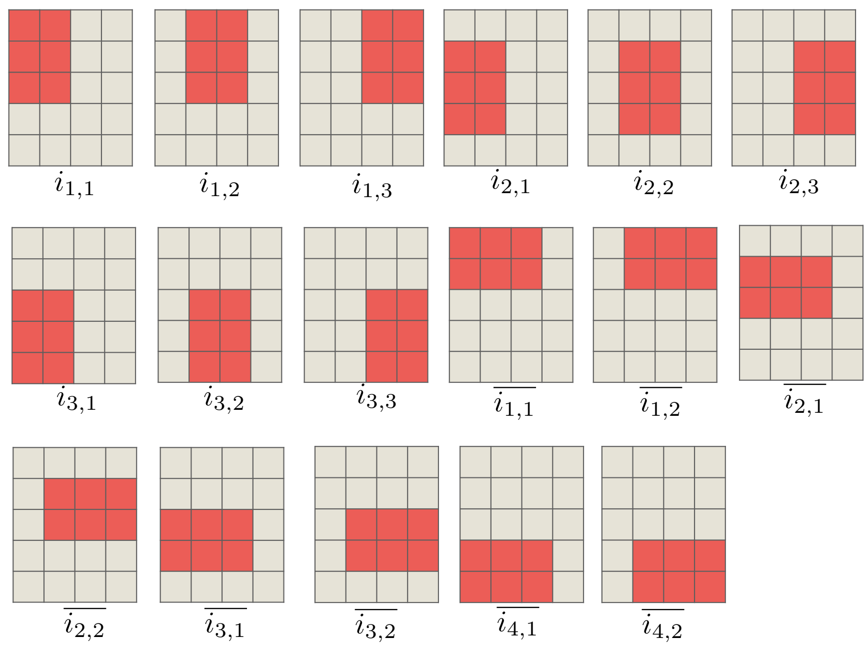


Figure 3.6: All the different positions that an item of size 3×2 can take in a bin of size 5×4 when rotation is allowed.

The Set-Covering formulation must also be adapted for the case of rotations. To this end, we must add the following variables:

$$y^{i,k}_{h,w} = \begin{cases} 1 & \text{if item } i \text{ in valid position } \overline{i_{h,w}} \text{ is placed on bin } k, \\ 0 & \text{otherwise.} \end{cases}$$

The following covering problem solves the decision problem D-2D-BPP(K) when rotation of the items is allowed:

$$\sum_{i=1}^{n} \left[\sum_{h=1}^{H-h_i+1} \sum_{w=1}^{W-w_i+1} x^{i,k}_{h,w} c^i_{h,w}(s_{l,p}) + \sum_{h=1}^{H-w_i+1} \sum_{w=1}^{W-h_i+1} y^{i,k}_{h,w} \overline{c^i_{h,w}}(s_{l,p}) \right] \leq 1$$
$$\forall l = 1, \ldots, H, \forall p = 1, \ldots, W, \forall k = 1, \ldots, K \qquad (3.3)$$

$$\sum_{k=1}^{K}\left[\sum_{h=1}^{H-h_i+1}\sum_{w=1}^{W-w_i+1} x_{h,w}^{i,k} + \sum_{h=1}^{H-w_i+1}\sum_{w=1}^{W-h_i+1} y_{h,w}^{i,k}\right] \geq d_i \qquad \forall i=1,\ldots,n \tag{3.4}$$

$$x_{h,w}^{i,k}, y_{h,w}^{i,k} \in \{0,1\} \qquad \forall h=1,\ldots,H, \forall w=1,\ldots,W,$$
$$\forall k=1,\ldots,K, \forall i=1,\ldots,n.$$

The restrictions (3.3) ensure that each unit square of each bin is covered by only one item, whether rotated or not. The constraints (3.4) ensure that the demand of every item i is satisfied by the rotated and non-rotated items taken together.

As before, if there is no feasible solution for the previous covering model, the P&C methodology increases K by one and iterates again until a feasible solution is found and it will correspond to the optimal solution.

3.1.5 Experimental Results

In this section, we present the experimental results to validate our P&C methodology by testing the classical benchmark instances for 2D-BPP (Gonçalves y Resende, 2013), which are as follows:

bwmv: instances of Berkey y Wang (1987) and Martello y Vigo (1998)[1]. These instances are divided into classes: each class comprises 50 instances, 10 for each value of $n \in \{20, 40, 60, 80, 100\}$.

cgcut: instances proposed by Christofides y Whitlock (1977)[2].

ngcut: 12 instances proposed by Beasley (1985b)[2].

beng: 10 instances generated by Bengtsson (1982)[3].

[1]All instances, and the corresponding averages of the best known values, are available at `http://www.or.deis.unibo.it/research_pages/ORinstances/2BP.html`.

[2]Available from the OR-library: `http://people.brunel.ac.uk/~mastjjb/jeb/info.html`

[3]Available from PackLib2 (Fekete *et al.*, 2007), `http://www.ibr.cs.tu-bs.de/alg/packlib/index.shtml`

The *cgcut*, *gcut*, and *ngcut* instances are test problems for two-dimensional cutting problems, which were transformed to 2D-bin packing instances accordingly to Martello y Vigo (1998). There is another set of instances known as *gcut*, proposed by Beasley (1985a)[2]. We did not include these instances in our comparative benchmarks since they are large in terms of their size, so our exact methodology is not able to solve them in a reasonable time.

We compare the P&C methodology with six approaches that have proved to be the most effective in the literature:

TS3: Tabu search based on the constructive procedures of Lodi *et al.* (1999a).

HBP: A constructive heuristic that assigns a score to each item (Boschetti y Mingozzi, 2003b).

GLS: A Guided Local Search heuristic based on the iterative solution of constraint satisfaction problems (Faroe *et al.*, 2003).

SCH: The set-covering-based heuristic approach of Monaci y Toth (2006).

GVND: The hybrid GRASP/VND algorithm of Parreño *et al.* (2010).

BRKGA: The Biased Random Key Genetic Algorithm of Gonçalves y Resende (2013). Only this algorithm allows 90° rotation of the items.

Notice that all these algorithms are heuristic, so most of the best solutions obtained so far are not guaranteed to be optimal. Since P&C is an exact solution method, it verifies whether the best solutions obtained by these algorithms are optimal.

For the P&C algorithm, the Positions stage was coded in C++ and executed on a MAC Pro equipped with an Intel Core 2 Duo processor of 3.06 GHz, and 4GB of RAM. For the set-covering model, we used the integer linear solver (B&B) of CPLEX 12.6 with its default options except for the gap, which was set to 0, and the time limit, which was fixed at 24 hours. The B&B was executed on a MAC Pro equipped with two processors: Quad-Core Intel Xeon of 2.4 GHz, and 20GB of

RAM. To reduce the number of indices in the mathematical models, we have merged h and w into a single label.

Table 3.2 presents the results for the *bwmv* instances for the 2D-BPP without rotation of the items. The first column indicates the instance class of the *bwmv* group. The second column indicates the bin size $W \times H$. The third column indicates the number of items that should be packed into the bins. The fourth column indicates the lower bound reported by Monaci y Toth (2006), computed by applying all the lower-bounding procedures from the literature and an exact algorithm for a long computing time. The fifth column indicates the optimal number of bins obtained by P&C. Entries with "*T.O.*" mean that P&C could not find a feasible solution within the time limit. The rest of the columns indicate the results of the other approaches: BRKGA, GVND, SCH, GLS, TS3, and HBP. Each row indicates the average values for the 10 instances of each class-size. The entries in bold are the best results.

Table 3.2: Experimental results for *bwmv* instances without rotation of the items.

Class	$W \times H$	n	LB*	P&C	BRKGA	GVND	SCH	GLS	TS3	HBP
1	10×10	20	7.1	**7.1**	**7.1**	**7.1**	**7.1**	**7.1**	**7.1**	**7.1**
		40	13.4	**13.4**	**13.4**	**13.4**	**13.4**	**13.4**	13.5	**13.4**
		60	19.7	**20**	**20**	**20**	**20**	20.1	20.1	20.1
		80	27.4	**27.5**	**27.5**	**27.5**	**27.5**	**27.5**	28.2	**27.5**
		100	31.7	**31.7**	**31.7**	**31.7**	**31.7**	32.1	32.6	31.8
2	30×30	20	1.0	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**	**1.0**
		40	1.9	**1.9**	**1.9**	**1.9**	**1.9**	**1.9**	2.0	**1.9**
		60	2.5	**2.5**	**2.5**	**2.5**	**2.5**	**2.5**	2.7	**2.5**
		80	3.1	**3.1**	**3.1**	**3.1**	**3.1**	**3.1**	3.3	**3.1**
		100	3.9	**3.9**	**3.9**	**3.9**	**3.9**	**3.9**	4.0	**3.9**
3	40×40	20	5.1	**5.1**	**5.1**	**5.1**	**5.1**	**5.1**	**5.5**	**5.1**
		40	9.2	9/10	**9.4**	**9.4**	**9.4**	**9.4**	9.7	9.5
		60	13.6	1/10	**13.9**	**13.9**	**13.9**	14.0	14.0	14.0
		80	18.7	*T.O.*	**18.9**	**18.9**	**18.9**	19.1	19.8	19.1
		100	22.1	*T.O.*	**22.3**	**22.3**	**22.3**	22.6	23.6	22.6

P&C is able to obtain the exact solutions for instances of class 3 with 20 items. For class 3 with 40 items, P&C solved 9 instances out of 10, and for 60 items, P&C solved 1 instance out of 10. We cannot compare ourselves with the literature in this specific type of instances since we only have their average of bins of the 10 instances. The P&C has a particular behavior in its set-covering phase since the B&B cannot easily find dual relaxations. Therefore, most of the time, the B&B evolves without a relative gap and suddenly, either the optimum is found or the infeasibility of the instance is exhibited. In the cases where the optimum was not found, the relative gap is more than 100%. Recall that P&C starts with a fixed number K of bins, and if there is no a possible packing in these bins, then K is augmented and another iteration is executed until there is a feasible solution which corresponds to the optimal one.

The P&C certifies the optimality of the value of the instances of 10×10 with 60 and 80 items. In the other cases, the lower bound was equal to the best solution values, so the optimum was only corroborated by the P&C.

Table 3.3 shows the results for the instance set *bwmv* when rotations of the items are allowed. The structure of the table is similar to that of Table 3.2 but we only compare ourselves to the BRKGA of Gonçalves y Resende (2013) since they are the only ones that present solutions when the items can be rotated by 90^o.

After the results of Table 3.3, we can say that whenever rotation is allowed, it should be used since it yields fewer bins for the packing than in the case without rotations (compared with the results of Table 3.2). The P&C could solve the instances up to class 3 with 20 items. For class 3 with 40 items, P&C solved 9 instances out of 10. Recall that the number of positions of the items and therefore the number of variables in the P&C set-covering formulation can almost double when rotation is allowed. Thus, these instances are harder to solve by our exact methodology.

Table 3.3: Experimental results for *bwmv* instances with rotation of the items.

Class	$W \times H$	n	P&C	BRKGA
1	10×10	20	**6.6**	**6.6**
		40	**12.8**	**12.8**
		60	**19.5**	**19.5**
		80	**27**	**27**
		100	**31.3**	**31.3**
2	30×30	20	**1.0**	**1.0**
		40	**1.9**	**1.9**
		60	**2.5**	**2.5**
		80	**3.1**	**3.1**
		100	**3.9**	**3.9**
3	40×40	20	**4.7**	**4.7**
		40	9/10	**9.2**
		60	*T.O.*	**13.4**
		80	*T.O.*	**18.2**
		100	*T.O.*	**22**

The experimental results for the instances *cgcut*, *ngcut*, and *beng* are shown in Table 3.4 when rotations are not allowed. The first column indicates the class of instances, the second one indicates the number of items that should be packed into the bins. The third column indicates the number of instances contained in each class. The fourth column indicates the best known average lower bound. The rest of the columns indicate the average number of bins obtained by P&C, BRKGA, GVND, SCH, GLS, and TS3.

P&C could find optimal solutions for instances that could not be dealt with metaheuristics like the SCH, GLS or TS3. The BRKGA and the GVND already were at the optimum value since they matched the lower bound. Nevertheless, the P&C shows a good performance for these instances.

Table 3.4: Experimental results for the instances *cgcut*, *ngcut*, and *beng* when rotations are not allowed.

Class	n	# inst	LB*	P&C	BRKGA	GVND	SCH	GLS	TS3
cgcut	16-62	3	9	**9**	**9**	**9**	**9**	**9**	**9**
ngcut	7-22	12	2.67	**2.67**	**2.67**	**2.67**	**2.67**	**2.67**	3
beng 1-8	20-120	8	6.75	**6.75**	**6.75**	**6.75**	6.88		
beng 9-10	160-200	2	6.5	**6.5**	**6.5**	**6.5**			

Table 3.5 is similar to the previous table and shows the experimental results for *cgcut*, *ngcut*, and *beng* instances when rotations are allowed. Again, we only compare P&C to the BRKGA. For class *beng1-8*, P&C only solved 7 instances out of 8.

Table 3.5: Experimental results for *cgcut*, *ngcut*, and *beng* instances when rotations are allowed.

Class	n	# inst	P&C	BRKGA
cgcut	16-62	3	**7.67**	7.67
ngcut	7-22	12	**2.5**	2.5
beng 1-8	20-120	8	7/8	6.75
beng 9-10	160-200	6.5	**6.5**	6.5

We can again verify that allowing the items to rotate decreases the number of bins needed to pack all the items. The P&C methodology shows its value by obtaining the optimal solutions for almost all these instances.

Normally, one does not have the LB* value at hand since it is the best lower bound of many procedures from the literature and an exact algorithm for a long computing time. So the P&C methodology is very valuable for guaranteeing the optimal values.

3.1.6 Conclusions

In this study, we present a new two-stage methodology, called Positions and Covering (P&C), to obtain exact solutions for the two-dimensional bin packing problem (2D-BPP). The key point of this work is the first stage where P&C generates, in a pseudo-polynomial way, all the positions in which each item can be packed into the bin. In the second stage, a set-covering model is solved to select the optimal packing in each bin. The P&C iteratively increases the number of bins needed to pack the items.

The P&C methodology was tested using the literature benchmark for the 2D-BPP. Due to the combinatorial complexity of the problem it was expected that P&C could not solve large instances. Nevertheless, for the small and medium instances, we were able to verify that the solutions proposed by other approaches were indeed the optimal.

We generalize the P&C to consider the case where the items can be rotated of 90°. The set-covering integer linear programming related to this case can have almost the double of variables. In the experimental setting we observed that these instances take longer to solve than the ones that do not allow the rotation. Nevertheless, the generalized P&C is one of the few exacts algorithms to solve the 2D-bin packing with rotation.

A natural research line is to develop a decomposition method for P&C like a column generation or branch-and-price. This will allow to consider larger instances and to reduce the computational times. We observed that the set-covering model of the P&C has poor quality lower bounds. It can be worth to investigate either new formulations, lower bounds, or cuts to strengthener the model. Finally, it will be interesting to consider the 3D-bin packing problem.

3.1.7 Acknowledgments

N.M. Cid-García wishes to acknowledge a graduate scholarship from the Mexican National Council for Sciences and Technology (CONACYT). This work has been supported by grant I0002 PDCPN2013-01/213630 of CONACYT.

Chapter 3

Two-Dimensional Packing Problems

3.2 Positions and Covering (P&C), an adaptive approach to obtain optimal solutions for the 2D-Strip Packing Problem

Abstract

We present an adaptation of the *Positions and Covering* (P&C) methodology to obtain exact solutions for the two-dimensional, non-guillotine restricted, strip packing problem (2SP). In this problem, a given set of rectangular items has to be packed into a strip of fixed weight and infinite height. The objective is minimizing the height of the strip used to pack all the items. P&C is based on a two-stage procedure where first is generated in a pseudo polynomial way, a set of valid positions in which an item can be packed into the strip. Then, is solved a Set-Covering formulation to select the best configuration of items into the strip. Experimental results, based in the literature benchmark, validate the quality of the solutions and the effectiveness of the method.

3.2.1 Introduction

The *Two-Dimensional Strip Packing Problem* (2SP) is composed by a given set of n rectangular items, each one with specific width w_i and height h_i for all $i = 1, \ldots, n$, and a strip of width W and infinite height. The aim is to orthogonally place all the items into the strip, without overlapping, to minimizing the overall height of the strip (Martello *et al.*, 2003; Lesh *et al.*, 2004). We assume that all input data w_i, h_i, and W are positive integers and that $w_i \leq W$ for all items $i = 1, \ldots, n$. We consider the case when the items have a fixed orientation and the guillotine cut constraint is not restricted.

The 2SP is NP-hard in the strong sense since it can be reduced to the one-dimensional bin-packing problem (Baker *et al.*, 1980; Hochbaum y Maass, 1985; Martello *et al.*, 2003) and, according to the typology proposed by Wäscher *et al.* (2007), the 2SP belongs to the class of cutting and packing problems and is classified as *two-dimensional, open dimension problem (2D-ODP).*

Figure 3.7 shows the optimal configuration for an instance proposed by Jakobs (1996) with 50 items, and a strip of width $W = 40$. In this case, there is no wasted parts of the strip, that is, we have a *perfect packing*.

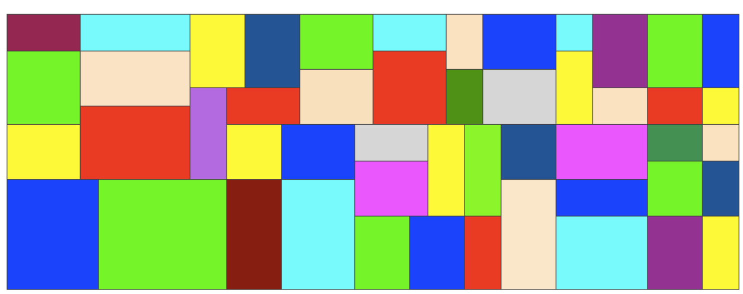


Figure 3.7: The optimal configuration of an instance proposed by Jakobs (1996) with 50 items, and a strip of width $W = 40$.

Many real-world applications of this problem can be found in the paper, textile, glass, steel, and wood industries, where rectangular items are cut from large rectangular sheets of material, which can be considered of infinite height (Gilmore y Gomory, 1965; Hopper y Turton, 2001b). 2SP also appears in scheduling problems in which tasks require a contiguous subset of identical resources (Augustine *et al.*, 2006).

The 2SP is closely related to the two dimensional bin packing (2D-BPP) problem where a set of items with specific width and height must be placed without overlapping into the minimum number of bins of given size. The main difference is that for 2SP we must determine the size of the strip while in the 2D-BPP we must determine the number of bins. This difference impede us to use the same exact methodology developed for the 2D-BPP, named P&C, (Cid-Garcia y Rios-Solis, 2015) which is the best method to find optimal solutions, in the best of our knowledge.

The main contribution of this work is to revisit the P&C methodology for the 2D-BPP and propose some structural modifications that allow us to use it for the 2SP. Thus, the obtained methodology is able to solve instances from the classical benchmarks, in particular, P&C solved to optimality the instances `dagli-01` and `dagli-03` proposed by Dagli y Poshyanonda (1997) where the optimal solutions were not known before of this research.

Due to the combinatorial complexity of the 2SP, the attempts to solve it can be roughly divided into exact and metaheuristics methods. In terms of exact methods, there have been recent combinatorial branch-and-bound (B&B) algorithms that build solutions by packing items one at a time in the strip like the ones of Martello *et al.* (2003), Alvarez-Valdés *et al.* (2009), Lesh *et al.* (2004), Kenmochi *et al.* (2009), Boschetti y Montaletti (2010), and Arahori *et al.* (2012).

Two exact algorithms and an approximate algorithm have been proposed by Hifi (1998) to solve a variant of the strip cutting problem. The algorithms are based upon branch-and-bound and dynamic programming procedures.

In Martello *et al.* (2003) is introduced a new relaxation that produces good lower bounds and gives information to obtain effective heuristic algorithms. Then, these results were used in a branch-and-bound algorithm, which was able to solve test instances from the literature. The most important contribution is the way to compute the lower bound for the branch and bound procedure.

A branch and bound algorithm is showed in Lesh *et al.* (2004) to solve 2D rectangular packing problems. A special case of the 2D strip packing problem. The branch-and-bound algorithm is enhanced with a dynamic programming mechanism for determining if gaps can be filled that proves surprisingly effective on benchmark problems.

Kenmochi *et al.* (2009) proposed two algorithms for the 2SP with and without 90 degrees rotations. These algorithms are called STAIRCASE and G-STAIRCASE algorithms. The firs one uses branching operations based on the staircase placement, and the second one uses branching operations based on the generalized staircase placement.

Inside of the heuristics and metaheuristics methods to solve the 2SP, we can found the work proposed by Baker *et al.* (1980), where is introduced the bottom-left (BL) heuristic. Another work which improved the previous one was proposed by Chazelle (1983). Some approaches which used some variants or implementations of the bottom-left strategy to solve packing problems are: Jakobs (1996); Liu y Teng (1999); Hopper y Turton (2001a); Lesh *et al.* (2005); Gonçalves y Resende (2013).

Another works which implemented metaheuristics methods as tabu search, simulated annealing, and genetic algorithms are the following: Iori *et al.* (2003); Bortfeldt (2006); Alvarez-Valdés *et al.* (2008). In many of the previous studies, was used a version of the BL algorithm and modified the order of the rectangles.

Reviews about some methodologies for the 2SP are presented in Hopper y Turton (2001b); Ntene y van Vuuren (2009). A revision of approaches for strip packing problems are showed in Riff *et al.* (2009), and some surveys for packing and cutting problems are presented in Haessler y Sweeney (1991); Dowsland y Dowsland

(1992); Lodi *et al.* (2002a,b).

In this study, we propose an adaptation of the "Positions and Covering (P&C)" methodology, used by Cid-Garcia y Rios-Solis (2015) to obtain optimal solutions for the two-dimensional bin packing problem, to solve the 2SP. P&C was originally designed as a zoning method to delineate rectangular and homogeneous management zones in agricultural fields (see Cid-Garcia *et al.* (2013); Albornoz *et al.* (2015)) but, according to its characteristics, it has been modified to solve packing problems. More details in Section 3.2.2.

The results presented in this paper are referred as exact solutions for the 2SP considering the case when items have a fixed orientation. In this case, we are not considering the guillotine cut constraint. We use instances from the literature benchmark to validate the solution quality and method effectiveness.

The rest of paper is organized as follows. In Section 3.2.2 we describe the methodology proposed, we present briefly the Positions and Covering methodology, and we show the adaptation of Positions and Covering for the 2SP. Section 3.2.3 shows the experimental results based on literature benchmark. Finally, in Section 3.2.4 we make some concluding remarks.

3.2.2 Materials and methods

The P&C methodology consist on a two-stage procedure where in the first stage is generated a set of valid positions for each item into the strip. Then, in the second stage, is solved to optimality a Set-Covering formulation based on an integer programming model. A brief description of P&C and its adaptation to 2SP is given next.

"Positions and Covering" methodology

Figure 3.8 shows the P&C methodology used to solve the 2D-BPP. Given an instance of the 2D-BPP, the first step is to generate, for each item, a set of valid positions where each item can be placed in the bin. Then, it is computed the number of K bins used to pack all the items, in this case the number of bins is fixed assuming that exists a perfect packing (the sum of the total area of items divided by the area of bin). Although, this value can be fixed used the best lower bound known in the literature.

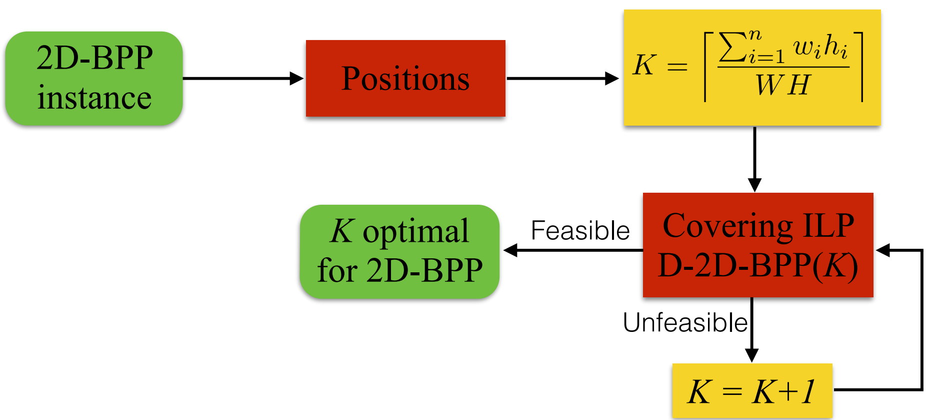


Figure 3.8: Scheme of the P&C methodology for the 2D-BPP.

The next step is to solve a covering model where is taken the decision version of the 2D-BPP (D-2D-BPP(K)). If the covering model is feasible then the optimal solution for the 2D-BPP is obtained, else, the number of bins is increased by one and the covering model is solved again. The procedure ends when a feasible solution for the D-2D-BPP(K) is founded.

Adaptation of "Positions and Covering" to 2SP

To implement the P&C methodology in the strip packing problem it is necessary to make some modifications on it. This modifications are showed in the Figure 3.9.

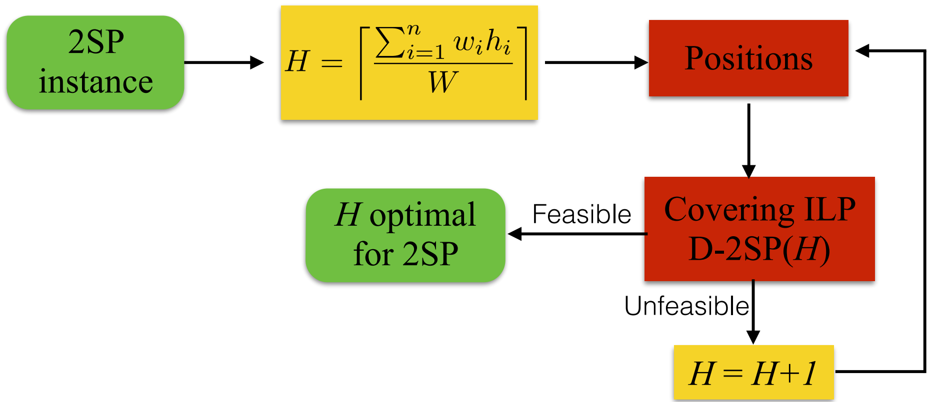


Figure 3.9: Scheme of the P&C methodology for the 2SP.

Given an instance of the 2SP the first step is to compute the *height* of the strip (H) assuming that exists a perfect packing. Then, the set of valid positions for each item inside of the strip is generated, and a covering model, using the decision version of the 2SP (D-2SP(H)), is solved. If the covering model is feasible then the optimal solution for the 2SP is founded, else, the H-value is increased by one, the new positions for the items are generated, and the covering model is solved again. The procedure ends when a feasible solution for the D-2SP(H) is founded.

POSITIONS STAGE

The objective of this stage is to generate, for each item, a set of valid positions where the item can be placed into the strip. As the 2D-BPP, in this stage we are not generating patterns. We are giving a set of valid positions for each item where the Set-Covering formulation will choose the optimal configuration for the strip.

Let us to consider a single strip with known width W and unknown height H, and a set of n items $i = 1, \ldots, n$, each one with specific width w_i and height h_i. The first step is to compute the height of the strip assuming that exist a perfect packing:

$$H = \left\lceil \frac{\sum_{i=1}^{n} w_i h_i}{W} \right\rceil,$$

that corresponds to the total area of the items, divided by the width of the strip (see Figure 3.10).

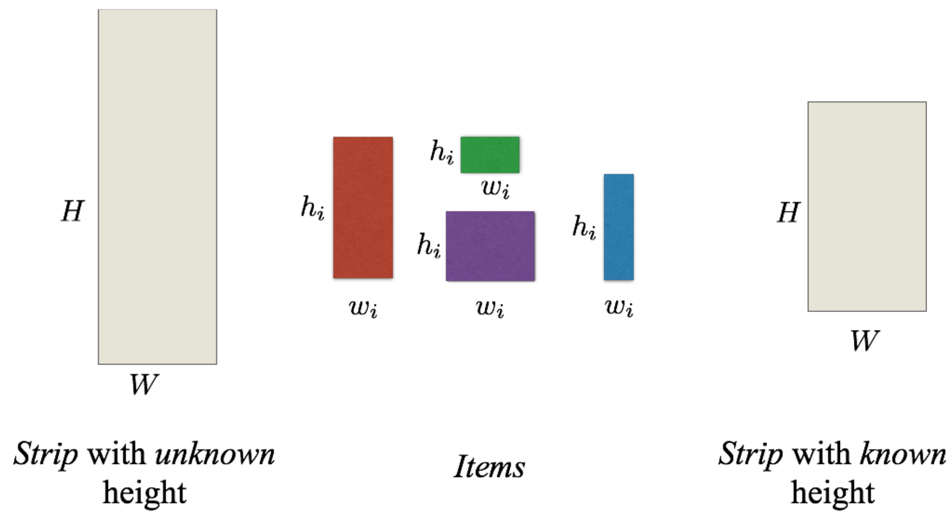


Figure 3.10: Determination of the height of the strip.

Once it is defined the *height* of the strip, the next step is to generate a grid inside of the strip, that is, a regular tessellation of the 2-dimensional Euclidean space by congruent unit squares, where each square has a particular Id. The enumeration start at the top left corner square and, end at the bottom right square (see Figure 3.11).

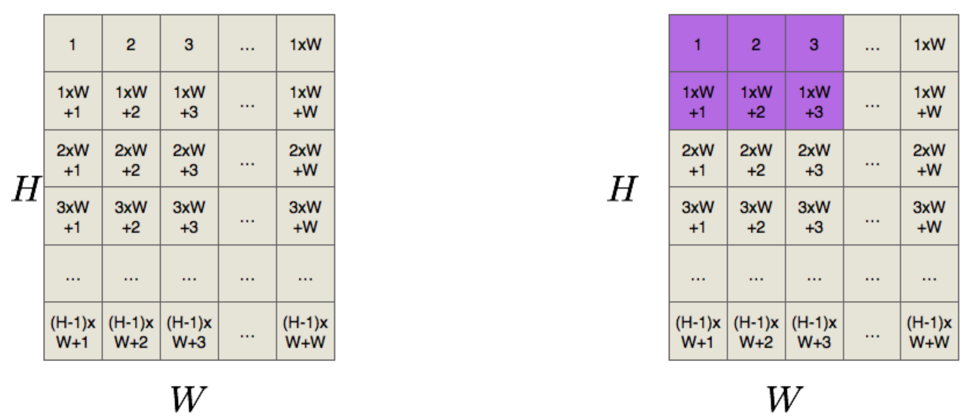


Figure 3.11: Grid inside of the strip.

For each item, a valid position is going to be created if its top left corner is placed in the square with ID=Id and its dimensions of width and height do not exceed the dimensions of the strip. Also, the valid position must be labeled to differentiate one with each other. In Figure 3.11 is showed the valid position for an item of dimensions 2×3 with top left corner at square 1. Notice that the position with start point at square 1×W is not a valid position for this item.

The set of valid positions for all the items is pseudo-polynomial and it can be generated using the Algorithm 5 proposed by Cid-Garcia *et al.* (2013). Some variations of the original algorithm were made to adapt it to 2SP. The input of the Algorithm 5 is the number of points into the strip, the number of points in the width of the strip ($WidthS$), the number of points in the length of the strip ($LengthS$), the number of points in the width of each item ($WidthI$), and the number of points in the length of each item ($LengthI$). The algorithm starts creating the positions width wise. Then it checks if there is still some width to cover. After, it checks the length. The Algorithm 5 is executed for each item.

Algorithm 5 Generation of possible positions of a item into the strip.

```
INPUT: WidthS, LengthS, WidthI, LengthI, strip points
for j = WidthI To WidthS do
  for l = 0 To (WidthS − 1) do
    if (j + l) ≤ WidthS then
      for i = LengthI To LengthS do
        for k = 0 To (LengthS − 1) do
          if (k + i) ≤ LengthS then
            creation of a valid position
          end if
        end for
      end for
    end if
  end for
end for
```

In Figure 3.12 is showed the set of valid positions for an item of 2×3 and an item of 5×3 in a strip with dimensions 6×4. In this case only 14 valid positions are generated. The set of positions since 1 to 10 is for the first item while the set since 11 to 14 is for the second item.

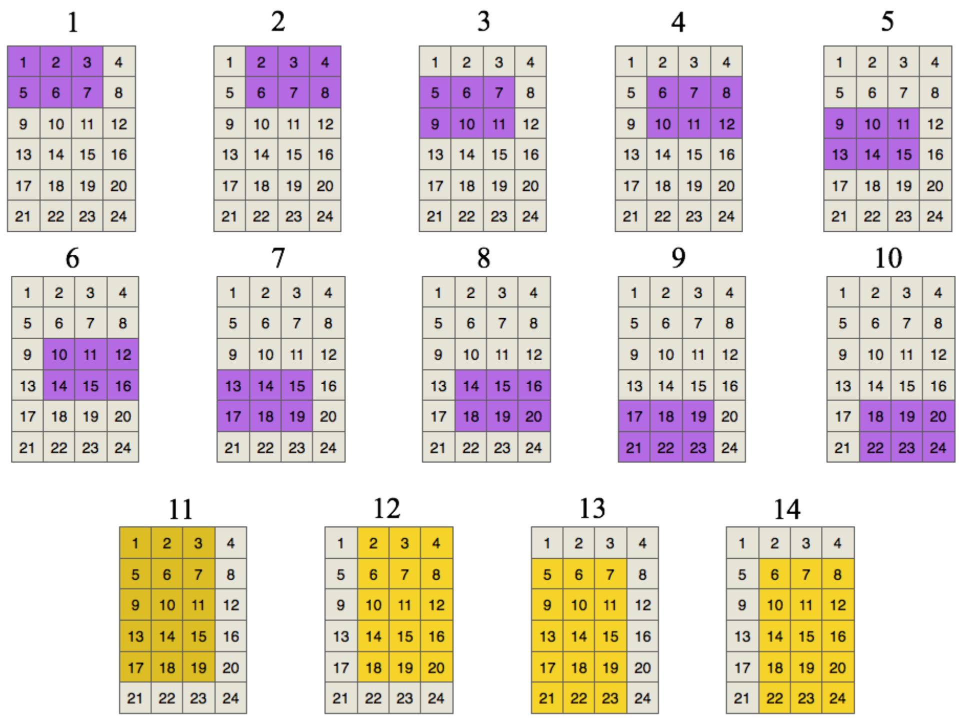


Figure 3.12: Valid positions for an item of 2×3 in a strip of 6×4.

Each valid position obtained by Algorithm 5 is unique, therefore, it has a specific label and an unrepeatable set of points. For example, the position 1 contains the points: 1, 2, 3, 5, 6, 7 while, the position 9 contains the points: 17, 18, 19, 21, 22, and 23.

The result of Algorithm 5 is a correspondence matrix $C = \{c_{jp}\}$ where rows represents the valid positions for all the items, and columns are the number of points into the strip. $C = \{c_{jp}\}$ is composed of 1's and 0's, where $c_{jp} = 1$ if valid position j covers point p, $c_{jp} = 0$ otherwise. The correspondence matrix of Figure 3.12 appears in Table 3.6.

Table 3.6: Correspondence matrix C for Figure 3.12.

		Points of strip p									
		1	2	3	4	5	6	7	8	...	24
Potential position j	1	1	1	1	0	1	1	1	0	...	0
	2	0	1	1	1	0	1	1	1	...	0
	3	0	0	0	0	1	1	1	0	...	0
	4	0	0	0	0	0	1	1	1	...	0
	5	0	0	0	0	0	0	0	0	...	0
	6	0	0	0	0	0	0	0	0	...	0
	7	0	0	0	0	0	0	0	0	...	0
	8	0	0	0	0	0	0	0	0	...	0
	9	0	0	0	0	0	0	0	0	...	0
	10	0	0	0	0	0	0	0	0	...	1
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	
	14	0	0	0	0	0	1	1	1	...	1

Each row in the correspondence matrix is the respective labeled position of Figure 3.12, i.e., the row 1 of matrix have ones in points 1, 2, 3, 5, 6, and 7. If you check the figure, you can verify that inside the position 1 only points 1, 2, 3, 5, 6, and 7 are covered. This can be made for each row of matrix with its respective position.

If the model is feasible using this set of valid positions then the procedure ends, else, the height of the strip is increased by one (or using the best lower bound, if this exist), and the new set of valid positions is generated for each item.

Set-Covering formulation

After we have the set of valid positions transformed into a correspondence matrix, we proceed to execute a Set-Covering formulation based in an integer linear programming model. This model solve the decision problem D-2SP(H)

A particular characteristic of the model is that the initial height of the strip (H) is fixed. If the model is feasible then the optimal solution is founded else we increase the height of the strip by one, the new positions for the items are generated and the model is solved again. The process ends when the model is feasible.

Next, the parameters used for the model are given:

- I: set of items.
- J: set of valid positions.
- $T(i)$: subset of valid positions for item i where $T(i) \in J$.
- P: set of points inside of the strip.

The decision variables of the model:

$$x_j = \begin{cases} 1 & \text{if position with label } j \text{ is chosen,} \\ 0 & \text{otherwise.} \end{cases}$$

The Set-Covering formulation to solve the decision problem D-2SP(H) is as follows:

$$\sum_j c_{jp} x_j \leq 1 \quad \forall p \in P \tag{3.5}$$

$$\sum_{j \in T(i)} x_j \geq d_i \quad \forall i \in I \tag{3.6}$$

$$x_j \in \{0, 1\} \quad \forall j \in J \tag{3.7}$$

In the previous mathematical formulation do not exist an objective function, this means we are only searching feasibility in the model. Restrictions (3.5) ensure that each point $p \in P$ of the strip is covered by only one item. With this, the overlapping is avoided. Constraints (3.6) ensure that demand of item $i \in I$ be satisfied. Notice that the demand of each item i is satisfied taking into account only the corresponding valid positions for each item. The nature of variables is declared in 3.7.

3.2.3 Experimental Results

In this section, we present the experimental results to validate the P&C methodology using the classical benchmark instances for 2SP.

Test problem instances

The P&C methodology was tested using the original benchmark for the strip packing and the benchmark for other two-dimensional cutting problems. The description of each group of instances is given next:

Original instances for the strip packing

This set of instances refers to the original instances of the strip packing problem.

- 2 instances proposed by Jakobs (1996) and known as `jack01-jack02` instances.
- 4 instances proposed by Dagli y Poshyanonda (1997); Ratanapan y Dagli (1997, 1998); and known as `dagli01-dagli04` instances. The optimum for instances 01, 03 and, 04 is not known.
- 25 instances proposed by Hifi (1998) and known as `hifi01-hifi25` instances.
- 12 instances proposed by Hopper y Turton (2001a) and known as `ht01-ht18` instances.

Instances for other two-dimensional cutting problems

This set of instances was originally introduced for other two-dimensional cutting problems, and were transformed into strip packing instances by using the item sizes and bin width.

- 3 instances proposed by Christofides y Whitlock (1977), and known as *cgcut* instances.

- 12 instances proposed by Beasley (1985b), and known as *ngcut* instances. The *cgcut*, and *ngcut* instances are test problems for 2D cutting problems, which were transformed to 2D bin packing instances according to Martello y Vigo (1998). This instances are available from the ORLIB library.

- 10 instances generated by Bengtsson (1982). Available in PackLib2 (Fekete *et al.* (2007)), `http://www.ibr.cs.tu-bs.de/alg/packlib/index.shtml`.

- 3 instances proposed by Burke *et al.* (2004) and known as *N1–N3* instances.

Computational results

To solve the instances we use the branch & bound algorithm (B&B) of CPLEX 12.6 using default options. It was executed on a MAC Pro equipped with two processors Quad-Core Intel Xeon with 8 cores and, running at 2.4 GHz. The RAM was fitted with 20 GB. The limit time for CPLEX execution in each instance was fixed to one day.

The computational results for *the original instances for the strip packing* are shown in Tables 3.7–3.10. Table 3.7 shows the experimental results for the instances from Jakobs (1996). The first three columns describe the instance. The fourth column is the optimum value known for each instance. The fifth column is the result of P&C, numbers in bold represent the optimal solution. From sixth to tenth columns are showed the results for other approaches: Jakobs (1996), Liu y Teng (1999), Mumford-Valenzuela *et al.* (2004) and, Bortfeldt (2006), respectively. P&C is the only approach with optimal solutions for both instances.

Table 3.7: Results for instances proposed by *Jakobs (1996)*.

Instance	Problem size		Opt	P&C	Jakobs	Liu and	Mumford–	Bortfeldt	
	W	n				Teng	Valenzuela	Average	Best
01	40	25	15	**15**	17	16	16	16	16
02	40	50	15	**15**	17	16	16	**15**	**15**

Table 3.8 shows the experimental results for the instances proposed by Dagli y Poshyanonda (1997). The first three columns describe the instance. The fourth column is the best lower bound known for each instance. The fifth column is the result of P&C, numbers in bold represent the optimal solution. From sixth to tenth columns are showed the results for other approaches: Ratanapan y Dagli (1997, 1998), Dagli y Poshyanonda (1997), Poshyanonda y Dagli (2004) and, Bortfeldt (2006), respectively. In this case, P&C was able to solve 3 of 4 instances to optimality. In the fourth instance we cannot guarantee the optimality because P&C cannot solve it with value of 208 and we cannot determine if the optimal value is 208 or 209.

Table 3.8: Results for instances proposed by *Dagli y Poshyanonda (1997)*.

Instance	Problem size		LB	P&C	Ratanapan and Dagli	Dagli and Poshyanonda	Poshyanonda and Dagli	Bortfeldt	
	W	n						Average	Best
01	60	31	45	**46**	91.88	–	–	93.9	95.67
02	60	21	40	**40**	92.50	–	–	96.6	97.56
03	30	37	112	**126**	94.41	–	96.03	98.5	98.58
04	20	37	161	209	–	97.15	–	97.6	97.62

Table 3.9 shows the experimental results for the instances from Hifi (1998). The first three columns describe the characteristics of each instance. The fourth column is the best lower bound known for each instance. The fifth column is the result of P&C. The sixth column shows the results of Hifi (1998). For this instances set, P&C was able to solve all the instances to optimality.

In Table 3.10 are presented the results for 12 of 21 instances proposed by Hopper y Turton (2001a). The first four columns describe the characteristics of each instance. The fifth column is the optimal value known for each instance. The result of P&C is showed in the sixth column, MO means the computer memory was not enough to solve the instance. The optimal solutions were reached only for the first 9 instances. For the other instances P&C cannot obtain the optimal solutions due to the size of the correspondence matrix was too large and the memory of the computer was not enough to solve them. The column 7 corresponds to the solutions obtained by the hybrid algorithm by Iori *et al.* (2003). The columns 8–11 show the results obtained by the best-fit algorithm from Burke *et al.* (2004) and its enhancements adding tabu search (TS), simulated annealing (SA) and a genetic algorithm (GA) from Burke *et al.* (2006). In columns 12 and 13 is presented the average and best solutions obtained by Bortfeldt (2006). Finally, the last two columns show the average and best results obtained by the GRASP proposed by Alvarez-Valdés *et al.* (2008).

To make a comparison of our methodology with respect to other approaches, we use the average percentages from optimum proposed by Alvarez-Valdés *et al.* (2008). This averages is calculated as (sol–opt)/opt. Table 3.10 shows that our algorithm obtains the best results for the first 3 class, obtaining 9 of 9 optimal solutions, improving the best results of the GRASP proposed by Alvarez-Valdés *et al.* (2008). However, for the other instances of classes 4 to 7, P&C cannot obtain the optimal solutions due to the size of the instance.

From Tables 3.7 and 3.9 we can see that all the instances proposed by Jakobs (1996), Dagli y Poshyanonda (1997); Ratanapan y Dagli (1997, 1998), and Hifi (1998) were solved using the P&C methodology. For the instances proposed by Hopper y Turton (2001a) P&C only solved 9 of 12 instances. This is due to the size of the items and of the height of the strip. The set of valid positions is too large, and the memory is not enough to solve this problems.

It is important to make a special emphasis in the instances proposed by Dagli y Poshyanonda (1997); Ratanapan y Dagli (1997, 1998) because, the optimal solution for the instances `dagli-01`, `dagli-03` and, `dagli-04` it was not known before of this study. With this study we can guarantee the optimal solutions for the instances `dagli-01` and `dagli-03`.

Table 3.9: Results for instances proposed by *Hifi (1998)*.

Instance	Problem size		LB	P&C	Hifi
	W	n			
01	5	10	13	**13**	**13**
02	4	11	40	**40**	**40**
03	6	15	14	**14**	**14**
04	6	11	19	**20**	**20**
05	20	8	20	**20**	**20**
06	30	7	38	**38**	**38**
07	15	8	14	**14**	**14**
08	15	12	17	**17**	**17**
09	27	12	68	**68**	**68**
10	50	8	80	**80**	**80**
11	27	10	48	**48**	**48**
12	81	18	34	**34**	**34**
13	70	7	50	**50**	**50**
14	100	10	60	**69**	**69**
15	45	14	34	**34**	**34**
16	6	14	32	**33**	**33**
17	42	9	34	**34**	39
18	70	10	89	**100**	101
19	5	12	25	**25**	26
20	15	10	19	**20**	21
21	30	11	140	**140**	145
22	90	22	34	**34**	**34**
23	15	12	34	**34**	35
24	50	10	103	**109**	114
25	25	15	35	**35**	36

Table 3.10: Experimental results for instances proposed by *Hopper y Turton (2001a)*.

Class	Instance	Problem size		Opt	P&C	Iori	Bestfit	Burke			Bortfeldt		GRASP	
		W	*n*					BF+TS	BF+SA	BF+GA	Average	Best	Average	Best
C1	01	20	16	20	**20**	**20**	21	**20**	**20**	**20**			**20**	**20**
	02	20	17	20	**20**	21	22	21	**20**	21			**20**	**20**
	03	20	16	20	**20**	**20**	24	**20**	**20**	**20**			**20**	**20**
Average percentage deviation from optimum					**0**	1.59	10.17	1.59	**0**	1.59	1.59	1.59	**0**	**0**
C2	04	40	25	15	**15**	**15**	16	16	16	16			**15**	**15**
	05	40	25	15	**15**	16	16	16	16	16			**15**	**15**
	06	40	25	15	**15**	**15**	16	16	16	16			**15**	**15**
Average percentage deviation from optimum					**0**	2.08	6.25	6.25	6.25	6.25	3.33	2.08	0	0
C3	07	60	28	30	**30**	31	32	31	31	31			**30**	**30**
	08	60	29	30	**30**	31	34	32	31	32			31	31
	09	60	28	30	**30**	**30**	33	31	31	31			**30**	**30**
Average percentage deviation from optimum					**0**	2.15	9.04	4.23	3.23	4.23	3.16	3.16	1.08	1.08
C4	10	60	49	60	MO	64	63	62	61	62			61	61
	11	60	49	60	MO	63	62	62	61	62			61	61
	12	60	49	60	MO	62	62	61	61	62			61	61
Average percentage deviation from optimum					–	4.75	3.74	2.70	1.64	3.23	3.52	2.70	1.64	1.64
Number of optimal solutions					9/12	5/12	0/12	2/12	3/12	2/12			8/12	8/12

Tables 3.11 and 3.12 show the experimental results for instances *of other two-dimensional cutting problems*. In Table 3.11 are presented the results of Burke *et al.* (2004), the first three columns describe the characteristics of each instance. The fourth column is the optimal value known for each instance. The result of P&C is showed in the fifth column. The optimal solutions (numbers in bold) were reached only for the first 3 instances, for other instances P&C cannot obtain the optimal solutions due to the size of the instance. The columns 6–9 show the results obtained by the best-fit algorithm from Burke *et al.* (2004) and its enhancements adding tabu search (TS), simulated annealing (SA) and a genetic algorithm (GA) from Burke *et al.* (2006). Finally, the last two columns show the average and best results obtained for the GRASP proposed by Alvarez-Valdés *et al.* (2008).

Table 3.11: Results for instances proposed by Burke *et al.* (2004)

Instance	Problem size		Opt	P&C	Best-fit	Burke solutions			GRASP solutions	
	W	n				BF+TS	BF+SA	BF+GA	Average	Best
N1	40	10	40	**40**	45	**40**	**40**	**40**	**40**	**40**
N2	30	20	50	**50**	53	**50**	**50**	**50**	**50**	**50**
N3	30	30	50	**50**	52	51	51	52	51	51

Table 3.12 shows the instances proposed by Beasley (1985b), Bengtsson (1982) and, Christofides y Whitlock (1977). The first column represents the source of the problem. Columns 2–4 describe the characteristics of each instance. The fifth column is the best known lower bound. Column 6 is the optimal value obtained for P&C, MO means that the computer memory was not enough to solve the instance. The seventh column is the result of Iori *et al.* (2003). The last two columns show the average and best results obtained for the GRASP proposed by Alvarez-Valdés *et al.* (2008).

Table 3.12: Results for instances of *other two-dimensional cutting problems*

Source of problem	Instance	Problem size		LB	P&C	Iori	GRASP	
		W	n				Average	Best
Beasley *ngcut*	01	10	10	23	**23**	**23**	**23**	**23**
	02	10	17	30	**30**	**30**	**30**	**30**
	03	10	21	28	**28**	**28**	**28**	**28**
	04	10	7	20	**20**	**20**	**20**	**20**
	05	10	14	36	**36**	**36**	**36**	**36**
	06	10	15	29	**31**	**31**	**31**	**31**
	07	20	8	20	**20**	**20**	**20**	**20**
	08	20	13	32	**33**	**33**	**33**	**33**
	09	20	18	49	**50**	**50**	**50**	**50**
	10	30	13	80	**80**	**80**	**80**	**80**
	11	30	15	50	**52**	**52**	**52**	**52**
	12	30	22	87	**87**	**87**	**87**	**87**
Bengtsson *beng*	01	25	20	30	**30**	31	**30**	**30**
	02	25	40	57	**57**	58	**57**	**57**
	03	25	60	84	**84**	86	**84**	**84**
	04	25	80	107	**107**	110	**107**	**107**
	05	25	100	134	MO	136	**134**	**134**
	06	40	40	36	**36**	37	**36**	**36**
	07	40	80	67	**67**	69	**67**	**67**
	08	40	120	101	MO	–	**101**	**101**
	09	40	160	126	MO	–	**126**	**126**
	10	40	200	156	MO	–	**156**	**156**
Christofides *cgcut*	1	10	16	23	**23**	**23**	**23**	**23**
	2	70	23	63	MO	65	65	65
	3	70	62	636	MO	676	661	661
Number of proven optimal solutions (P&C & matching LB)					19/25	13/25	23/25	23/25

From Table 3.12 we can see that the best results were obtained for the GRAPS of Alvarez-Valdés *et al.* (2008). However, our contribution here, is that we have increased the number of optimal solutions obtained for the GRASP from 19/25 to 23/25.

From Tables 3.7 and 3.9 we can see that all the instances proposed by Jakobs (1996), Dagli y Poshyanonda (1997); Ratanapan y Dagli (1997, 1998), and Hifi (1998) were solved to optimality using the P&C methodology. From the instances proposed by Hopper y Turton (2001a) P&C only solved 9/12 instances. This is due to the size of the items and of the height of the strip. The set of valid positions is too large, and the memory it was not sufficient to solve this problems.

We make a special emphasis in the instances proposed by Dagli y Poshyanonda (1997); Ratanapan y Dagli (1997, 1998) because, the optimal solutions for the instances `dagli-01` and `dagli-03`, it was not known before of this study.

Big instances could not be solved using P&C methodology due to the combinatorial complexity of the problem. In this case we propose to apply alternative exact methodologies, as columns generation, to give solutions to this kind of instances.

With respect to the computational times, this are not showed because we cannot make comparisons between an exact method and a heuristic method. Beside, the instances were implemented and tested on computers with different characteristics. Also, the main purpose of this study is to give optimal solutions for instances of 2SP and, not the quick solution of them.

3.2.4 Conclusions

In this study, we present an adaptation of the "Positions and Covering (P&C)" methodology to obtain exact solutions for the two-dimensional strip packing problem (2SP). The methodology is based on a two-stage procedure where first is generated in a pseudo-polynomial way a set of valid positions where each item can be allocated inside of the strip. The height of the strip is computed assuming that exist a perfect packing. Then, a set covering formulation, based on integer linear programming, is solved to determine if the height of the strip is optimal or not. The configuration of the strip is also given by the set covering formulation.

The P&C methodology was tested using the benchmark for 2SP, but P&C only was able of solving small and medium instances, where we were able to verify that the solutions proposed by other approaches were the optimal solutions.

A main contribution of this study is that we have optimal solutions for instances proposed by Dagli y Poshyanonda (1997); Ratanapan y Dagli (1997, 1998), where the optimum for instances dagli-01 and dagli-03 it was not known before of this study.

According to the combinatorial complexity of the problem, a decomposition approach is necessary to complement P&C to give optiml solutions for large instances.

Chapter 4

Conclusions

In the present dissertation we propose a new methodology to obtain exact solutions for the agricultural planning, and for two packing problems.

The first part of this dissertation, focuses on the agricultural planning problems, shows a new methodology based on Precision Agriculture and mathematical models of linear programming for helping the decision makers (the farmers) to plan and operate their plots in order to avoid resources wastage and to maximize their profits.

An important characteristic of this study is that we have considered the soil variability existing in the agricultural plot, that is, the physical and chemical soil properties of the field. The chemical soil properties affect on the application of inputs (fertilizers, pesticides, etc.), while the physical soil properties are related to the water use.

Our methodology starts by delineating the field into rectangular and homogeneous site-specific chemical and physical management zones to face within-field variability. Then the farmers assign a crop to the different plots to obtain the best profit at the end of the production cycle. Finally, during each irrigation period, the farmer must decide how much and which plots must be watered such to maximize the total final yields.

In the models we use real-time information about crops and environment (temperature, moisture level, solar radiation, wind, phenological stage of crops, etc.), and they are solved optimally in reasonable time with low computational requirements. Experimental results show that the methodology is efficient and practical to be used in a decision support system to improve agricultural planning processes.

The second part of this dissertation is related to the Two-Dimensional Packing problems. In this case we are considering the Bin and Strip Packing problems.

In the Two-Dimensional Bin Packing Problem we have an unlimited number of identical rectangular bins with fixed width and height, and a set of rectangular items each one with specific width and height. The objective of this problem is to allocate the set of rectangular items using the minimum number of bins.

For the Two-Dimensional Strip Packing Problem we have a rectangular strip with fixed width but infinite height, and a set of rectangular items each one with specific width and height. In this case, the objective consist in allocate all the set of rectangular items into the strip using the minimum height of the strip.

Both problems, Strip and Bin Packing, are NP-Hard in the strong sense, since a reduction of these problems can be easily made for the one-dimensional bin packing problem which is strongly NP-Hard. Due to the combinatorial complexity of the problem, many studies have been focused to implement heuristics and metaheuristics methods to solve these problems, and just a few approaches have been focused in exact methods to give a solution for these problems.

In this study, we show a new methodology called Positions and Covering (P&C) to obtain exact solutions for the Two-Dimensional Bin and Strip Packing Problems. The methodology is based on a two-stage procedure where is solved a Set-Covering formulation of integer linear programming. The P&C methodology was tested using the literature benchmark for Bin and Strip Packing problems. P&C was able to solve small and medium instances, but for large instances P&C cannot solve them.

4.1 Conclusions

4.1.1 Agricultural Planning

It has been showed that a great amount of resources (seeds, costs, profits, water, etc.) affects directly in the agricultural production planning. Therefore, it is very important to analyze the amount of available resources during the whole production cycle to take a good decision, which maximizes the farmer's profit and minimizes the production cost at the end of the production cycle.

To delineate the field in management zones is one of the most important characteristics to improve the agricultural planning process and the efficient water management. A bad delineation of the agricultural field can imply several consequences as:

- The bad selection of the crop pattern, although the mathematical model is solved to optimality.
- The production cost can increase considerably because the amount of inputs supplied into the plots is greater than the necessary.
- The solution obtained by the crop planning and the real-time irrigation models is no the best option to maximize the farmer's profit.
- An unnecessary use of natural resources such as the water. This is an important characteristic because the water must to take special care in the production planning process.

For the crop planning problem we have considered the information of previous production cycles and different regions to execute the mathematical models. This is a big problem because the crops have different behavior according to the weather conditions of each region. It is not the same behavior when a crop is planted in a warm weather than in a cold weather, the weather conditions are different for each region and therefore they have different impact on each crop.

Unfortunately, almost all the farmers in Mexico do not have a database or a data record that show the performance, profit, production costs, or crop yield, just to name a few, of last seasons to execute the mathematical models. Instead of that, the information used in this research was taken from different research centers (as INIFAP and SGARPA) to make the experimentation and the analysis of the models. But, we think that if we use local information, specific for each region, then the selection of the crop pattern can be different and more adjusted to the real requirements of the farmer. In this sense, the information does not have a strong impact in the mathematical models performance, but in the selection of the crop pattern.

With respect to the irrigation problem, it is very important to consider the crop requirements in real time to select the optimal amount of water to be supplied in each crop. This is a very hard task due to not all the farmers have the appropriate technology to obtain the information required to calculate this value.

Inside of this technology we can find moisture sensors and weather stations which are used to obtain information such as: the solar radiation, the evapotranspiration level of the crop, the speed of wind, the crop coefficient, and the humidity level of the crop, just to name a few. With this information we can calculate the exact amount of water to be irrigated to the crop attending its real-time water requirements.

As in the crop planning problem, if this model is applied using information of other regions, then the results cannot be the appropriate for the crop and the yield can be affected considerably. Therefore, to apply this model, it is necessary to invest in technology to obtain the information locally.

With respect to the mathematical models, we conclude that these can be used as make-decision tools in the agricultural planning process because they work efficiently given exact solutions to the problem. Also, it is considered that the tools showed in this study can replace the traditional techniques, where it is used the improvisation and the inappropriate study of the soil properties, to improve the agricultural planning process.

4.1.2 Two-Dimensional Packing Problems

The P&C methodology showed good results for small and medium instances, however for large instances P&C cannot guarantee a good performance. This is due to the combinatorial complexity of the problems.

The pre-processing stage is a key point for the methodology. But, if we want to solve more instances (large and huge instances, mainly) then it is needed to make some modifications in this step and do not generate all the set of valid positions.

Instead of generate all the set of valid positions, we propose to study each instance in a particular way to generate just a small set of valid positions that guarantee the optimality of the solution.

It is interesting to analyze the particular elements of each instance such as the size of the small item, the demand for each item, the number of items to be packed and, the size of the bin/strip. All these elements determine the set of valid positions used in the set-covering formulation to solve the problem.

If the set of valid positions is huge, then it is very probably that the solver cannot obtain the optimal solution in a reasonable time; even the solver cannot obtain the solution. Therefore, the key point in the P&C methodology is the generation of the set of valid positions of each item. Thus, many efforts must be addressed to the intelligent creation of this kind of information.

Therefore, of this research we can conclude that we have develop a new methodology called Positions and Covering (P&C) that optimally solve small and medium instances of the bin and strip packing problems, however, to solve large instances using this methodology it is necessary to make some modifications on it taking into account all the previous considerations.

4.2 FUTURE WORK

4.2.1 AGRICULTURAL PLANNING

There exists a lot of work to do in the agricultural planning process. It is true that this problem has been tackled since several years ago, however, in the best of our knowledge, just a few amount of studies take into account the real-time crop requirements.

Currently, there exist many technological tools, like Precision Agriculture, which proportionate a lot of information that can be used to make a better agricultural production planning. The mathematical models developed in this research show some benefits of use this kind of information.

Next, we give a list about of new topics that can be used to improve the agricultural planning process.

- First, it is necessary to implement all the mathematical models using the information of a specific location. This is very complicated, because not all the farmers have the data of their crops, plots, cost, profits, and resources, just to name a few. Therefore, the first part is to make a compilation of the information where these tools are going to be used.

- Another important point is to define some strategies to improve the water use, i.e., it is necessary to assess the existing of this vital resource for future generations. Therefore, it is very important to decide if the production planning is going to be made using all the available resources (including the water) or, if this planning must be done considering a limiting amount of resources, specially the water. This last consideration can help to guarantee the saving water and make the agriculture planning a more sustainable practice.

- With respect to the mathematical models, the crop planning problem has been addressed using a deterministic approach. However, there are some parameters such as costs, profits, water, that have a stochastic nature. Therefore, a possible extension of this problem is considering the stochastic approach of this problem.

- For the delineation of rectangular and homogeneous management zones problem, it is required to make a delineation using several soil properties such as Nitrogenous (N), Phosphorous (P), Potassium (K), or another, to analyze which chemical soil property has more impact in the crop planning selection. The same procedure must be done for the physical soil properties.

- Another important think to do is the supply chain in the farming sector. This is very interesting because almost all the farmers have animals in their farms. Therefore, a very attractive extension of this work can be the incorporation of the livestock sector in the agricultural production planning.

  More specifically, to determine the number of animals in the farms (cows, horses, pigs, chickens); to determine the space and food for each specie; to define the crop pattern to satisfy the farm requirements; to decide when to sell/buy an animal; to satisfy the demand for each crop (if this exist); just to name some activities inside of this new project. All the previous activities must be done using the new technologies as Precision Agriculture in joint with optimization techniques.

4.2.2 Two-dimensional packing problems

For the classical two-dimensional bin and strip packing problems, there exists a lot of work to do yet. These problems have been tackled since several years ago using mainly heuristics or metaheuristics methods. But, in the best of our knowledge, just a few amount of works used the exact method to solve these problems.

With respect to our methodology and, according to the experimental results obtained by P&C, we have a lot of work to do. Also, we think this procedure can be implemented to solve another kind of problems. Next, we give a list of new topics where the P&C methodology can be implemented.

- To make some variants in the P&C methodology to try of solving large instances for the two-dimensional bin and strip packing problems using this procedure. The variants we propose are referred mainly to do not generate all the set of valid positions for each item. Instead of that, we propose make a special analysis of each instance and then generate a small set of valid positions, which guarantee the optimality of the solution.

- To implement a "Branch and Price" approach to obtain optimal solutions for large instances of the bin and strip packing problems. The main idea of this approach is to generate just a small set of valid positions and then, the model is going to decide if new positions for each item are required. In affirmative case, a sub-problem model generates the new positions and the process ends when no more positions are required or no more positions can be generated. Some advances of this research are presented in Appendix B

- Another possible extension is to implement the P&C methodology in the following problems:

  - *2D-BPP for irregular pieces.* This problem is similar to the classical 2D-BPP just that in this case the shape of the items to be packed is irregular. For this future research, we want to consider the case when the items have "L" and "T" shapes.

  - *2D-BPP for heterogeneous bins.* As the previous problem, this is similar to the classical 2D-BPP just that in this case it is considered that we have bins with different size. Due to this particular characteristic, the problem complexity increases considerably.

- *Heuristics for 2D-BPP.* Another research line is the implementation of metaheuristics to solve the 2D-BPP. The metaheuristics that we want propose are going to be based using the P&C methodology but using a "divide and conquer" approach, and a pre-classification of the items searching an incompatibility between them.

- *Packing 3D.* A more complicated but very exciting research line is the extension of the P&C methodology to solve 3D packing problems. This problem is similar to the 2D-BPP just that in this case are considered three-dimensional rectangular bins, which makes the problem strongly NP-Hard and extremely difficult to solve in practice.

- *Constraint Programming for packing problems.* The constraint programming paradigm is another strategy that can be used to solve the 2D packing problems. This approach has been implemented for solving several combinatorial search problems that draws on a wide range of techniques from artificial intelligence, computer science, databases, programming languages, and operations research. Constraint programming is currently applied with success to many domains, such as scheduling, planning, vehicle routing, configuration, networks, and bioinformatics. Therefore, constraint programming is another interesting research line to solve this kind of problems.

Appendix A

Optimization Concepts

In this appendix are present some basic concepts related to the optimization area, a short history about the operations research, and a brief description of some terms used in this Ph.D. dissertation.

A.1 General concepts

In Sarker y Newton (2007) the concept of "Optimization" is related to find the best solution (point or alternative) for a given problem. To find this solution, it is necessary to examine a set of alternatives and prove that the selected solution is the best. This best solution is known as the "optimal solution".

Conejo *et al.* (2006) refers to the optimization as "the science of the best" in the sense that it helps us to make not just a reasonable decision, but the best decision subject to observing certain constraints describing the domain within which the decision has to be made.

The optimization activities have been available for more than a century, but at the beginning, differential calculus was the basic tool applied for finding maxima or minima functions.

Around of 1942, during the World War II, when the British government introduced scientific groups to support the making decisions of military operations about logic, strategic and tactic problems. The objective of the scientists was maximizing the war effort using the limited amount of resources. The British community called this activity as Operational Research. Simultaneously, scientists of the United Sates of American did some similar activities for air military operations but they referred this activity as Operations Research. After the second worldwide war many people saw the results of these techniques in military operations and they applied them in the industrial planning as a making-decisions tool.

Currently, the optimization techniques have several applications in real-world problems. Some of these areas are: transportation and logistics, airline, rail industries, forestry industries, manufacturing planning and control, energy, telecommunications networks, molecular biology, water reservoir systems, agriculture, just to name a few (Pardalos y Resende, 2001; Ríos-Mercado, 2002).

Optimization also pertains to everyday decision making. For example, we try to buy the best car provided our budget is sufficient, and we look for the best college education for our children provided we can afford it. It also concerns minor decisions such as buying the best cup of coffee at a reasonable price. Therefore, optimization is part of our everyday activities.

To get a solution for an optimization problem it is required to solve a scientific model, typically a mathematical programming model, which is a representation of the problem. A solution that satisfies all the constraints of the model is known as a feasible solution, but if this solution also gives the best value for the objective function then this solution is the optimal solution (Bazaraa *et al.*, 2004; Sarker y Newton, 2007).

When the mathematical model is closer to the reality then the problem can be much harder to solve and the complexity to obtain the optimal solution can increase considerably (Taha, 2004; Winston *et al.*, 2005; Dantzig y Thapa, 1997; Hillier Frederick y Lieberman Gerald, 2010).

A mathematical model is composed by four elements: parameters, decision variables, constraints, and objective function, which are briefly defined next:

- The parameters are all the information known about of the problem. This information is constant during the model execution.

- The decision variables, as it is indicated by their name, are all the decisions that the optimizer wants to take for the problem. We can specify continuous, integer or binary variables.

- The constraints indicate the limits of the problem. They have tree components: a function that determine the amount of resources required to do an activity, process or service; a constant which represents the total amount of available resource; and an equality or inequality sign, to determine the function limits.

- The objective function represents the objective or goal of the mathematical model. According to the problem, the objective function can be maximized or minimized, and it is influenced directly by the constraints of the problem.

Basically, a mathematical model is represented as in Figure A.1.

Maximize or minimize the **Objective Function**

Subject to:

Constraints

Figure A.1: Structure of a mathematical model (Taha, 2004).

In mathematical terms, Figure A.1 can be represented as follows:

$$\textit{Maximize} \quad f(x) \tag{A.1}$$

Subject to:

$$g_i(x) \leq b_i \tag{A.2}$$

$$x \geq 0 \tag{A.3}$$

where the Equation A.1 is the objective function of the problem (in this case, maximized). The Equation A.2 represents the linear constraints of the problem. The Equation A.3 is the declaration of the variables (in this case, continuous variables).

An optimization problem can be classified according to: objective, kind of problem, kind of variables, or kind of objective function (see Figure A.2).

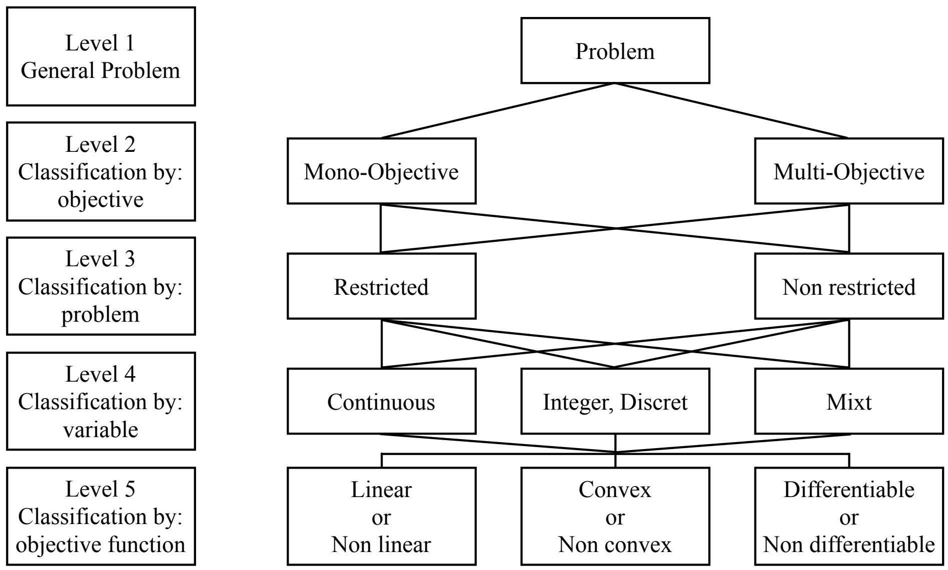


Figure A.2: Classification of a mathematical model (Sarker y Newton, 2007).

According to the variable, the mathematical models can be: Linear Programming (LP), Integer Programming (IP), Binary Integer Programming (BIP), and Mixed Integer Programming (MIP) (Chvatal, 1983).

In this dissertation, we present mono-objective models of: Linear Programming, Binary Integer Programming, and Mixed Integer Programming. A multi-objective model of Binary Integer Programming is showed too. A short description of these models is given in the following sections.

A.2 Linear Programming Model

In a linear programming model or linear program (LP) the objective function and the constraints are developing as linear functions. Also, the decision variables take continuous values greater than or equal to zero (Wolsey, 1998; Williams, 1999). The linear programming model can be represented as:

$$\begin{aligned} \textit{Maximize} \quad & cx \\ & Ax \leq b \\ & x \geq 0 \end{aligned}$$

where A is a matrix of m by n, c is a row vector of n dimensions, b is a column vector of m dimensions, and x is a vector of variables of n dimensions.

The "Simplex" algorithm proposed by Dantzig is the most used algorithm to solve this kind of problems (Dantzig y Thapa, 1997; Dantzig, 1998; Hillier Frederick y Lieberman Gerald, 2010). Although there exist another methods as the ellipsoid method proposed by (Khachiyan, 1980) and the interior-point method proposed by Karmarkar (1984) to solve linear optimization models.

A.3 Integer Programming Model

An integer programming model (IP) is an extension of the LP model just that in this case, the decision variables can take only integer values (Wolsey, 1998; Williams, 1999).

$$\begin{aligned} \textit{Maximize} \quad & cx \\ & Ax \leq b \\ & x \geq 0 \text{ and } \textit{integer} \end{aligned}$$

These models are very used in real-world applications where fractional values of the decision variables cannot be implemented. For example, you cannot buy the half of a car, or the half of a cow, this is illogical and impractical. The complexity of this kind of models is NP-Hard.

A.4 Binary Integer Programming Model

As the previous model, a binary integer programming model (BIP) is a variant of the LP model. In this case, the decision variables can take only binary values (0 or 1) (Wolsey, 1998; Williams, 1999).

$$\begin{aligned} Maximize \qquad & cx \\ & Ax \leq b \\ & x \in \{0,1\}^n \end{aligned}$$

These models are used in problems where the decisions variables only can take one of two possible values as "Yes" or "No". For example: if a product is elaborated or not, if a service is realized or nor, or if a center is opened or not, just to name a few. The complexity of this kind of models is NP-Hard.

A.5 Mixed Integer Programming Model

In a mixed integer program model (MIP) we can find a mix of variables: continuous, binaries or integer variables. This kind of problems is the most commonly used in real-life applications because, although it is necessary to take some decision, in many cases it is required to assign some amount of resources to products or activities (Wolsey, 1998; Williams, 1999).

$$\begin{aligned} Maximize \qquad & cx + hy \\ & Ax + Gy \leq b \\ & x \geq 0, y \geq 0 \text{ and } integer \end{aligned}$$

The MIP models are classified as NP-Hard problems, but there exist a set of exhaustive algorithms, as the Branch and Bound algorithm, which can proportionate a solution for the problem in considerable times periods.

A.6 Multi-objective Optimization

The purpose of the Multi-objective Optimization Problems (MOOPs) is solving problems that involve the simultaneous optimization of several objectives. Generally, these objectives are in conflict one with each other (Coello *et al.*, 2002; Figueira *et al.*, 2005; Ehrgott, 2006; Jaimes *et al.*, 2009).

For example, while a consumer wants to buy a house with low cost and high comfort, the seller wants to send the house with low cost or high comfort (see Figure A.3). Clearly, there exist a conflict of preferences between the consumer and the seller, and the consumer only can take one of these two preferences: low cost or high comfort, but not both (Burke y Kendall, 2005; Zopounidis y Pardalos, 2010).

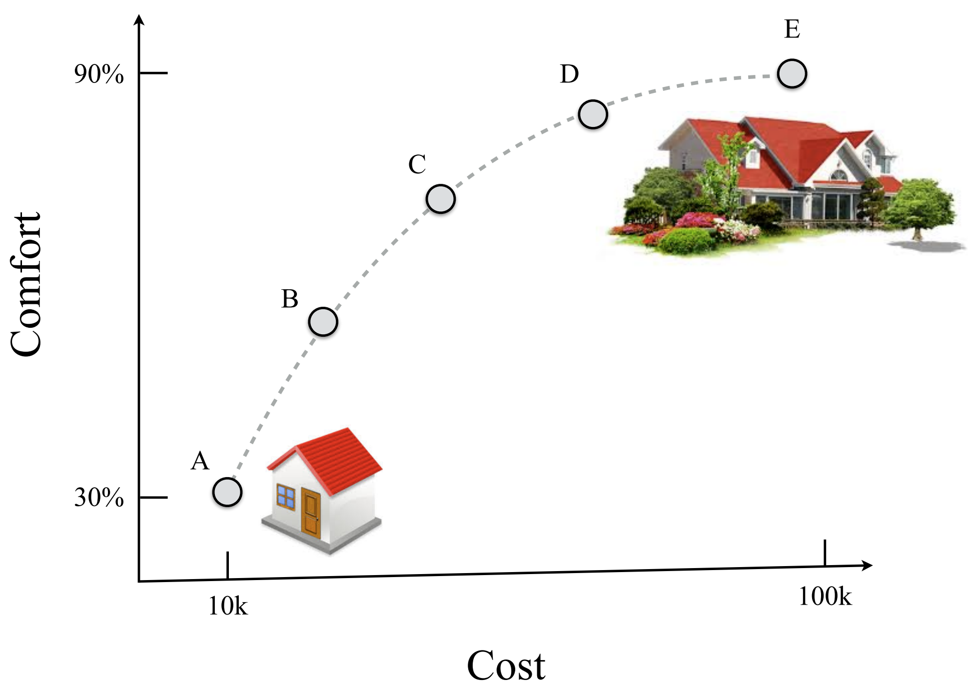


Figure A.3: Trade-off solutions for a house-buying decision making problem.

Jaimes *et al.* (2009) formally define a MOOP as:

$$\begin{aligned} &\textit{Minimize} \quad f(x) = [f_1(x), f_2(x), \ldots, f_k(x)]^T \\ &\textit{subject to:} \\ &\qquad\qquad x \in X \end{aligned}$$

where $x \in \mathbb{R}^n$ is a vector of n decisions variables representing the quantities for which values are to be chosen in the optimization problem. The vector function $f : \mathbb{R}^n \rightarrow \mathbb{R}^k$ is composed by k scalar objective functions $f_i : \mathbb{R}^n \rightarrow \mathbb{R}(i = 1, \ldots, k; k \geq 2)$. The sets $\mathbb{R}^n$ and $\mathbb{R}^k$ are known as decision variable space and objective function space, respectively. $X \subseteq \mathbb{R}^n$ is the feasible set that is determined by a set of equality and inequality constraints.

In multi-objective optimization we do not have a single solution as in the mono-objective optimization problems, instead of that we have a set of solutions, which represents the different trade-off among the objectives, see Figure A.3. In the figure, two houses are presented, the first one with a cost about of 10,000 dollars (option A) and the second house costing about of 100,000 dollars (option E).

If only one objective is considered, for example comfort, then, the optimal solution will be the option E. The figure shows that a cheaper house has a lower level of comfort. This is a two-objective optimization problem and the results are two extreme solutions. Between these two extreme solutions, there exist many other solutions, where a trade-off between cost and comfort exists. A number of such solutions (solutions B, C and D) with differing costs and comfort levels are also shown in the figure. Thus, between any two such solutions, one is better in terms of one objective, but this betterment comes only from a sacrifice on the other objective. In this sense, all such trade-off solutions are optimal solutions to a MOOP. Often, such trade-off solutions provide a clear front on an objective space plotted with the objective values. This front is called the Pareto-optimal front and all such trade-off solutions are called Pareto-optimal solutions. Mathematically, Pareto optimality can be defined as (Marler y Arora, 2004b; Chinchuluun y Pardalos, 2007):

Pareto Optimality: A point $x^* \in X$ with $f(x^*)$ is called (globally) Pareto optimal (or efficient or non-dominated, or non-inferior), if and only if there exists no point $x \in X$ such that $f_i(x) \leq f_i(x^*)$ for all $i = 1, 2, \ldots, k$ and $f_j(x) < f_j(x^*)$ for at least one index $j \in 1, 2, \ldots, k$.

Local Pareto Optimality: A point $x^* \in X$ with $f(x^*)$ is called locally Pareto optimal, if and only if there exists $\delta > 0$ such that x^* is Pareto optimal in $S \cap B(x^*, \delta)$. Here, $B(x^*, \delta)$ is the open ball of radius δ centered at point $x^* \in X$, that is, $B(x^*, \delta) = \{x \in R^n | \; \|x - x^*\| < \delta\}$. Note that every globally Pareto optimal solution is a locally Pareto optimal solution. However, the converse is not always true unless there are some assumptions in the problem.

Weak Pareto Optimality: A point $x^* \in X$ with $f(x^*)$ is called weakly Pareto optimal, if and only if there exists no point $x^* \in X$ such that $f_i(x) < f_i(x^*)$ for all $i = 1, 2, \ldots, k$. It is easy to see that every Pareto optimal solution is weakly Pareto optimal.

For multi-objective discrete optimization, the concept of Pareto optimality can be stated in the similar way as we defined in continuous optimization. Finding all Pareto optimal solutions is often computationally problematic since there are usually exponentially (or infinite) large Pareto optimal solutions. Furthermore, for even the simplest problems and even for two objectives, determining whether a point belongs to the Pareto optimal set is NP-hard (Papadimitriou y Yannakakis, 2000). One way to handle those problems is to introduce approximate Pareto solutions.

ϵ-Approximate Pareto Optimality: Given a scalar $\epsilon > 0$, an ϵ-approximate Pareto optimal set, denoted by P_ϵ, is a subset of X such that there is no other solution y such that $(1 + \epsilon) f_i(y) \geq f_i(x)$ for all $x \in P_\epsilon$ and for some i.

This definition says that every other solution is almost dominated by some solution in P_ϵ , i.e. there is a solution in P_ϵ that is within a factor of ϵ in all objectives.

There exist several techniques to solve multi-objective optimization problems as the scalarization technique, the ϵ-constraint method, goal programming, multilevel programming, just to name a few (Marler y Arora, 2004b; Chinchuluun y Pardalos, 2007; Jaimes *et al.*, 2009). Next, it is presented a short description of the ϵ-constraint method, because it was the method used to solve the bi-objective model of the Section 2.3.

The ϵ-constraint method was proposed by Vira y Haimes (1983). This method takes only one of the k objectives to be minimized (or maximized). The other objectives are used as constraints to be less than or equal to a given target values. In mathematical terms, if the first objective has been selected to be minimized we have:

$$\begin{aligned} &\textit{Minimize} \quad f_1(x) \\ &\textit{subject to:} \\ &\qquad f_i(x) \leq \epsilon_i \quad \forall i \in \{1, \ldots, k\} \setminus \{2\} \\ &\qquad x \in X \end{aligned}$$

One advantage of the ϵ-constraint method is that it is able to achieve efficient points in a non-convex Pareto curve. For instance, assume we have two objective functions and the objective function $f_1(x)$ is chosen to be minimized, i.e., then the problem is:

$$\begin{aligned} &\textit{Minimize} \quad f_1(x) \\ &\textit{subject to:} \\ &\qquad f_2(x) \leq \epsilon_2 \\ &\qquad x \in X \end{aligned}$$

The decision maker can vary the upper bounds ϵ_i to obtain weak Pareto optima. Clearly, this is also a drawback of this method, i.e., the decision maker has to choose appropriate upper bounds for the constraints, i.e., the ϵ_i values. Moreover, the method is not particularly efficient if the number of the objective functions is greater than two.

Appendix B

A Branch & Price Approach to Obtain Optimal Solutions for the 2D Bin Packing Problem

A branch and price approach is used for solving large-scale IPs. This method basically consists in a branch and bound method in which at each node of the search tree, several columns may be added to the LP relaxation.

At the start of the algorithm, sets of columns are excluded from the LP relaxation in order to reduce the computational and memory requirements and then columns are added back to the LP relaxation as needed.

The algorithm typically begins by using a reformulation, such as Dantzig-Wolfe decomposition, to form what is known as the **Master Problem**. The decomposition is performed to obtain a problem formulation that gives better bounds when the relaxation is solved than when the relaxation of the original formulation is solved.

The decomposition usually contains many variables and a modified version, called the **Restricted Master Problem**, that only considers a subset of columns.

Then, to check for optimality, a *subproblem* called the **Pricing Problem** is solved to find columns that can be added to the basis and reduce the objective function (for a minimization problem). This involves finding a column that has a negative reduced cost (see Figure B.1).

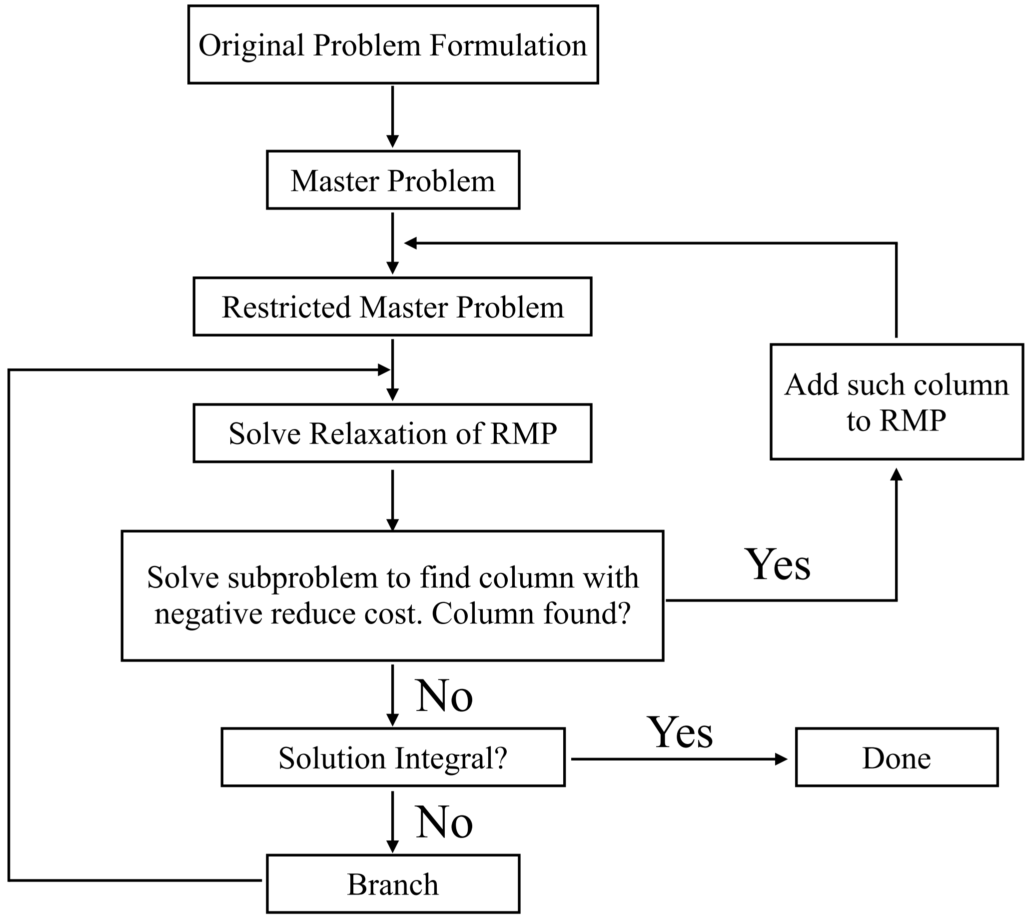


Figure B.1: A branch and price approach.

The implementation of the branch and price approach using the P & C methodology is given next.

B.1 Grid-based reformulation for the D-2D-BPP

B.1.1 Notations

Let I be the set of rectangular items to be packed, S be the set of points inside of the bins and K be the set of bins (unchanged).

Let L be the set of all positions for all items. It is possible to define constant parameters $a_l^i, \forall i \in I$ and $b_l^s, \forall s \in S$, that verify:

$$a_l^i = \begin{cases} 1 & \text{if position l belongs to item i, and} \\ 0 & \text{otherwise} \end{cases}$$

$$b_l^s = \begin{cases} 1 & \text{if point s belongs to position l, and} \\ 0 & \text{otherwise} \end{cases}$$

B.1.2 Potential reformulation

The proposed reformulation requires the following decision variables:

$$x_i^{sk} = \begin{cases} 1 & \text{if item i is placed on point s, and} \\ 0 & \text{otherwise} \end{cases}$$

$$y_l k = \begin{cases} 1 & \text{if position l is used in bin k, and} \\ 0 & \text{otherwise} \end{cases}$$

t_{sk}: continuous variable, to determine the number of items in point s of bin k.

The resulting model is:

$$\text{(EF)} \min \sum_s \sum_k t_{sk} \tag{B.1}$$

subject to:

$$\sum_{k \in K} \sum_{l \in L} a_l^i y_{lk} \geq d_i, \qquad \forall i \in I \tag{B.2}$$

$$t_{sk} = 1 - \sum_{l \in L} b_l^s y_{lk}, \qquad \forall s \in S, \forall k \in K \tag{B.3}$$

$$x_i^{s,k} = \sum_{l \in L} a_l^i b_l^s y_{lk}, \qquad \forall i \in I, \forall s \in S, \forall k \in K \tag{B.4}$$

$$x_i^{s,k} \in \{0,1\}, \qquad \forall i \in I, \forall s \in S, \forall k \in K$$

$$y_{lk} \in \{0,1\}, \qquad \forall l \in L, \forall k \in K$$

$$t_{sk} \in \geq 0, \qquad \forall s \in S, \forall k \in K$$

Objective (B.1) minimize the number of items in each point of each bin (minimizing the overlapping). Constraints (B.2) ensure that the demand of each item is satisfied. Constraints (B.3) establish the link between t_{sk} and y_{lk} variables. Finally, constraint (B.4) establish the link between $x_i^{s,k}$ and y_{lk} variables.

To allow the location of more than one item in each point of each bin (overlapping), Constraints (B.3) are changed to

$$t_{sk} + \sum_{l \in L} b_l^s y_{lk} \geq 1, \qquad \forall s \in S, \forall k \in K \tag{B.5}$$

B.1.3 Relaxing y_{lk} without losing optimality

Let (EF') be the problem obtained by replacing in (EF) the domain of variables y_{lk} with $\mathbb{R}^+$.

Theorem: (EF') and (EF) are equivalent.

Proof: It is based on the fact that any *optimal* solution of (EF') will be feasible for (EF).

B.2 Resulting Dantzig-Wolfe decomposition

B.2.1 Master Problem

The master problem is equivalent to (EF').

Restricted master problem: Primal linear relaxation

The restricted master problem (RMP) is obtained by replacing L with a subset $\overline{L} \subset L$ of the position sets of (EF'). Its linear relaxation (LRMP) is:

$$\text{(LRMP)} \min \sum_s \sum_k t_{sk} \tag{B.6}$$

subject to:

$$\sum_{k\in K}\sum_{l\in L} a_l^i y_{lk} \geq d_i, \qquad \forall i \in I \tag{B.7}$$

$$t_{sk} + \sum_{l\in L} b_l^s y_{lk} \geq 1, \qquad \forall s \in S, \forall k \in K \tag{B.8}$$

$$x_i^{s,k} - \sum_{l\in L} a_l^i b_l^s y_{lk} = 0, \qquad \forall i \in I, \forall s \in S, \forall k \in K \tag{B.9}$$

$$-x_i^{s,k} \geq -1, \qquad \forall i \in I, \forall s \in S, \forall k \in K \tag{B.10}$$

$$x_i^{s,k} \geq 0, \qquad \forall i \in I, \forall s \in S, \forall k \in K$$

$$y_{lk} \geq 0, \qquad \forall l \in L, \forall k \in K$$

$$t_{sk} \geq 0, \qquad \forall s \in S, \forall k \in K$$

Objective (B.6) minimize the number of items in each point of each bin. Constraints (B.7) ensure that the number demand for each item is ensured. Constraint (B.8) state that most one position set can be active on each point in each bin. Constraints (B.9) establish the link between $x_i^{s,k}$ and y_{lt} variables. Constraints (B.10) are required because variables x_{it} were binary before the linear relaxation.

Restricted master problem: Dual of the linear relaxation (V-2)

The dual of the linear relaxation of the restricted master problem (DLRMP) requires the following decision dual variables:

- u_i: dual variables of constraints (B.7)
- v_{sk}: dual variables of constraints (B.8)
- w_i^{sk}: dual variables of constraints (B.9)
- z_i^{sk}: dual variables of constraints (B.10)

The resulting dual is :

$$(DLMRP) \max \sum_{i\in I} d_i u_i + \sum_{s\in S}\sum_{k\in K} v_{sk} - \sum_{i\in I}\sum_{s\in S}\sum_{k\in K} z_i^{sk} \tag{B.11}$$

subject to:

$$\sum_{i \in I} a_l^i u_i + \sum_{s \in S} b_l^s v_{sk} - \sum_{i \in I} \sum_{s \in S} a_l^i b_l^s w_{sk}^i \leq 0, \qquad \forall l \in L, \forall k \in K \tag{B.12}$$

$$w_{sk}^i - z_{sk} \leq 0, \qquad \forall i \in I, \forall s \in S, \forall k \in K \tag{B.13}$$

$$\sum_{s \in S} \sum_{k \in K} v_{sk} \leq 1, \tag{B.14}$$

$$u_i \geq 0, \qquad \forall i \in I \tag{B.15}$$

$$v_{sk} \geq 0, \qquad \forall s \in S, \forall k \in K \tag{B.16}$$

$$w_{sk}^i \in \mathbb{R}, \qquad \forall i \in I, \forall s \in S, \forall k \in K \tag{B.17}$$

$$z_i^{sk} \geq 0, \qquad \forall i \in I, \forall s \in S, \forall k \in K \tag{B.18}$$

Objective (B.11) is obtained from the right-hand-side of constraints (B.7)-(B.10). Constraints (B.12) correspond to the primal decision variable y_{lk}. Constraints (B.13) correspond to the primal decision variable x_i^{sk}. Constraints (B.15), (B.16) and (B.18) state that variables u_i, v_{sk} and z_{sk} are positive because their related primal constraints (B.7), (B.8) and (B.10) are of type "$\geq$". Constraints (B.17) state that variable w_i^{sk} is $\mathbb{R}$ because the related primal constraint (B.9) is of type "=".

B.2.2 Subproblems

Finding a variable of negative reduced cost for a given (LRMP) is equivalent to finding the missing violated constraint in the associated (DLRMP). Therefore, the goal of the subproblem is to identify the position $l' \in L$ that maximizes the violation of constraint (B.12).

Procedure to generate positions of negative reduced cost

The idea is that for a predefined bin k and a predefined item i, finding a position l that maximizes the left-hand-side of constraint (B.12) consists in finding the feasible position that minimizes $\sum_{s \in S} b_l^s v_{sk} - \sum_{i \in I} \sum_{s \in S} a_l^i b_l^s w_i^{sk}$, since $\sum_{i \in I} u_i$ is constant.

As a consequence, each point s is assigned a weight "$v_{sk} - w_i^{sk}$'. The goal is to identify the position that minimizes the sum of weight of points covered. The constraint to enforce, to ensure feasibility of the resulting position, is that the shape of the item must be respected.

Note: In the case where there are no rotation allowed, it is enough to identify the corner left of the object, or one predefined extremity, the rest of it can be deduced.

SUBPROBLEM USING AN EXACT FORMULATION

Let P_s be the weight of each point s and, let $\tau(s)$ be the set of all point such that if the left corner of item i is at there point, then s will be covered. $P_s =$ "$v_{sk} - w_i^{sk}$".

The decision variables of subproblem are:

$$\alpha_s = \begin{cases} 1 & \text{if point s is the corner left, and} \\ 0 & \text{otherwise} \end{cases} \qquad \beta_s = \begin{cases} 1 & \text{if point s is covered, and} \\ 0 & \text{otherwise} \end{cases}$$

.

The resulting subproblem is:

$$(SDLMRP) \min \sum_{s \in s} P_s \beta_s \tag{B.19}$$

subject to:

$$\sum_{s \in S} \alpha_s = 1, \tag{B.20}$$

$$\beta_s = \sum_{j \in \tau(s)} \alpha_j, \qquad \forall s \in S \tag{B.21}$$

$$\alpha_s \in \{0, 1\}, \qquad \forall s \in S \tag{B.22}$$

$$\beta_s \in \mathbb{R}, \qquad \forall s \in S \tag{B.23}$$

Objective (B.19) consist in minimizing the weight of all points corresponding to best position of item in bin. Constraint (B.20) select the point "s for the upper-left corner of the item. Constraints (B.21) activate all the points corresponding to position.

B.3 Experimental Results

Table B.1 shows the preliminary results for the 2D-BPP using the “Branch & Price” approach for the instances of Martello y Vigo (1998). The first column is the class of the instance. The second column is the bin size. The third column represents the number of items to be packed in each instance. The fourth column is the best lower bound known. The fifth column represents the results of the P&C mehtodology. The last column shows the the results of the P&C mehtodology using the “Branch and Price” approach.

Table B.1: Experimental results for the 2D-BPP using the “Branch & Price” approach.

Class	Bin size	n	LB*	P&C	P&C 2
1	10×10	20	7.1	**7.1**	**7.1**
		40	13.4	**13.4**	**13.4**
		60	19.7	**20**	**20**
		80	27.4	**27.5**	**27.5**
		100	31.7	**31.7**	**31.7**
2	30×30	20	1.0	**1.0**	**1.0**
		40	1.9	**1.9**	**1.9**
		60	2.5	**2.5**	**2.5**
		80	3.1	**3.1**	**3.1**
		100	3.9	**3.9**	**3.9**
3	40×40	20	5.1	**5.1**	**5.1**
		40	9.2	9/10	**9.4**
		60	13.6	1/10	*Exe*
		80	18.7	*T.O.*	*Exe*
		100	22.1	*T.O.*	*Exe*

From Table B.1 in Class 3 we can see that the “Branch & Price” approach have best results than the simple P&C. However, it is necessary to finish the experimental results stage to give final conclusions of the research.

Appendix C

Summary of Notation for CPP

- I set of the different crops a farmer could plant, J is the set of parcels in the farmer's field, $Z^{Ch}(j)$ is the set of chemical management zones within parcel j and $Z^{Ph}(j)$ is the set of physical management zones within parcel j.
- G_i is the expected benefit of selling a tn of crop i at the end of the production cycle.
- $Cirr_{jz}$ is the cost of irrigating a m^3 of water in parcel j and physical zone $z \in Z^{Ph}(j)$.
- $Cseed_i$ is the cost of buying a kg of seeds of crop i.
- ha_j is the number of ha of parcel j.
- hac_{jz} is the number of ha in chemical management zone $z \in Z^{Ch}(j)$ of parcel j.
- hap_{jz} is the number of ha in physical management zone $z \in Z^{Ph}(j)$ of parcel j.
- $Iseed_i$ is the quantity of seeds in kg of crop i in the farmer's stock.
- $Seed_i$ is the quantity of seeds in kg needed to plant a ha of crop i.
- Y_i is the expected yield in tn by ha of crop i at the end of the production cycle.
- D_i is the demand in tn of crop $i \in I_0 \subsetneq I$ where I_0 is a subset of the crop set I.
- W is the expected total amount of water in m^3 for the whole planning cycle.

- ETc_{ij}^{v} is the crop evapotranspiration that represents the amount of water in mm required by crop i at a specific time v for parcel j.
- ETo^{v} is the reference crop evapotranspiration in mm at a specific time v.
- Kc_{ij}^{v} is the crop coefficient at phenological stage v, and geographic location j.
- SW_{jz}^{p} is the expected amount of stored water in m^3 in parcel j and physical management zone $z \in Z^{Ph}(j)$ at period p.
- W_{ijz} is the expected amount of water in m^3 needed for irrigation a ha planted with crop i in parcel j and physical management zone $z \in Z^{Ph}(j)$.
- $\boldsymbol{x_{ij}}$ is a binary variable, which is equal to 1 if crop i is planted in parcel j, and 0 otherwise.
- $\boldsymbol{s_i}$ is a variable representing the kg of seeds the farmer could buy of crop $i \in I$.

Appendix D

Summary of Notation for RTIP

- $\boldsymbol{Ya^p_{ijz}}$ is a variable of the expected yield of crop i in parcel j and physical management zone $z \in Z^{Ph}(j)$ at period p.

- Ya^{p-1}_{ijz} is the maximum yield of a crop i in parcel j and physical management zone $z \in Z^{Ph}(j)$ at period $p-1$.

- SW^p_{jz} is the amount of stored water in m^3 in parcel j and physical management zone $z \in Z^{Ph}(j)$ at period p.

- W^p_{ijz} is the current amount of water in m^3/ha that crop i needs in parcel j and physical management zone $z \in Z^{Ph}(j)$ at period p.

- hap_{jz} is the number of ha in physical management zone $z \in Z^{Ph}(j)$ of parcel j.

- ETa^p_{ijz} represents the amount of water supplied to crop i of parcel j and zone z during the current irrigation period p.

- Ky^p_i is the response factor of crop i at period p.

- G^p_i is the expected benefit of selling a tonne of crop i at the end of the production cycle given that we are at period p.

- $\gamma(j) = i$ is a function that indicates that crop i is sown in parcel j (this is obtained from the solution of CPP).

- W^p represents the amount of water available at irrigation period p.

- $\boldsymbol{w_{jz}^{p}}$ is a variable representing the amount of irrigated water in m^3/ha in physical management zone $z \in Z^{Ph}(j)$ of parcel j at period p.

List of Figures

List of Tables

Bibliography

Adeyemo, J. y F. Otieno (2010), «Differential evolution algorithm for solving multi-objective crop planning model», *Agricultural Water Management*, **97**(6), págs. 848 – 856.

Al-Karadsheh, E., H. Sourell y R. Krause (2002), «Precision Irrigation: New strategy irrigation water management», en *Conference on International Agricultural Research for Development. Deutscher Tropentag 2002 Proc*, págs. 9–11.

Albornoz, V. M., N. M. Cid-García, R. Ortega y Y. A. Ríos-Solís (2015), «A Hierarchical Planning Scheme Based on Precision Agriculture», en *Handbook of Operations Research in Agriculture and the Agri-Food Industry*, Springer, págs. 129–162.

Allen, R., L. Pereira, D. Raes y M. Smith (2006), *Evapotranspiración del cultivo: guías para la determinación de los requerimientos de agua de los cultivos*, tomo 56, Food & Agriculture Org.

Alminana, M., L. Escudero, M. Landete, J. Monge, A. Rabasa y J. Sánchez-Soriano (2010), «WISCHE: A DSS for water irrigation scheduling», *Omega*, **38**(6), págs. 492 – 500.

Alvarez-Valdés, R., F. Parreño y J. Tamarit (2009), «A branch and bound algorithm for the strip packing problem», *OR spectrum*, **31**(2), págs. 431–459.

Alvarez-Valdés, R., F. Parreño y J. M. Tamarit (2008), «Reactive GRASP for the strip-packing problem», *Computers & Operations Research*, **35**(4), págs. 1065–1083.

ARAHORI, Y., T. IMAMICHI y H. NAGAMOCHI (2012), «An exact strip packing algorithm based on canonical forms», *Computers & Operations Research*, **39**(12), págs. 2991–3011.

AUGUSTINE, J., S. BANERJEE y S. IRANI (2006), «Strip packing with precedence constraints and strip packing with release times», en *Proceedings of the eighteenth annual ACM symposium on Parallelism in algorithms and architectures*, ACM, págs. 180–189.

BAKER, B. S., E. G. COFFMAN, JR y R. L. RIVEST (1980), «Orthogonal packings in two dimensions», *SIAM Journal on Computing*, **9**(4), págs. 846–855.

BARNES, E., P. PINTER JR, B. KIMBALL, D. HUNSAKER, G. WALL, R. LAMORTE *et al.* (2000), «Precision irrigation management using modeling and remote sensing approaches», en *National irrigation symposium. Proceedings of the 4th Decennial Symposium, Phoenix, Arizona, USA, November 14-16, 2000.*, American Society of Agricultural Engineers, págs. 332–337.

BAZARAA, M. S., J. J. JARVIS y H. D. SHERALI (2004), *Programación lineal y flujo en redes*, Limusa.

BEASLEY, J. (1985a), «Algorithms for unconstrained two-dimensional guillotine cutting», *Journal of the Operational Research Society*, págs. 297–306.

BEASLEY, J. (1985b), «An exact two-dimensional non-guillotine cutting tree search procedure», *Operations Research*, **33**(1), págs. 49–64.

BENGTSSON, B.-E. (1982), «Packing rectangular pieces – a heuristic approach», *The computer journal*, **25**(3), págs. 353–357.

BERKEY, J. O. y P. Y. WANG (1987), «Two-dimensional finite bin-packing algorithms», *Journal of the Operational Research Society*, **38**(5), págs. 423–429.

Bhatti, A. U., D. J. Mulla y B. E. Frazier (1991), «Estimation of soil properties and wheat yields on complex eroded hills using geostatistics and thematic mapper images», *Remote Sensing of Environment*, **37**(3), págs. 181–191.

Bitran, G. R. y A. C. Hax (1977), «On the design of hierarchical production planning systems», *Decision Sciences*, **8**(1), págs. 28–55.

Blackmore, S. (2000), «The interpretation of trends from multiple yield maps», *Computers and Electronics in Agriculture*, **26**(1), págs. 37–51.

Bortfeldt, A. (2006), «A genetic algorithm for the two-dimensional strip packing problem with rectangular pieces», *European Journal of Operational Research*, **172**(3), págs. 814–837.

Boschetti, M. A. y A. Mingozzi (2003a), «The two-dimensional finite bin packing problem. Part I: New lower bounds for the oriented case», *Quarterly Journal of the Belgian, French and Italian Operations Research Societies*, **1**(1), págs. 27–42.

Boschetti, M. A. y A. Mingozzi (2003b), «The two-dimensional finite bin packing problem. Part II: New lower and upper bounds», *Quarterly Journal of the Belgian, French and Italian Operations Research Societies*, **1**(2), págs. 135–147.

Boschetti, M. A. y L. Montaletti (2010), «An exact algorithm for the two-dimensional strip-packing problem», *Operations Research*, **58**(6), págs. 1774–1791.

Burke, E., G. Kendall y G. Whitwell (2006), «Metaheuristic enhancements of the best-fit heuristic for the orthogonal stock cutting problem», *Computer science technical report no. NOTTCS-TR-SUB-0605091028-4370, University of Nottingham.*

Burke, E. K. y G. Kendall (2005), *Search methodologies*, Springer.

Burke, E. K., G. Kendall y G. Whitwell (2004), «A new placement heuristic for the orthogonal stock-cutting problem», *Operations Research*, **52**(4), págs. 655–671.

Cambardella, C. A., T. B. Moorman, J. M. Novak, T. B. Parkin, D. L. Karlen, R. F. Turco y A. E. Konopka (1994), «Field-scale variability of soil properties in central Iowa soils», *Soil Science Society of America Journal*, **58**, págs. 1501–1511.

Carr, P. M., G. R. Carlson, J. S. Jacobsen, G. A. Nielsen y E. O. Skogley (1991), «Farming soils, not fields: A strategy for increasing fertilizer profitability», *Journal of Production Agriculture*, **4**(1), págs. 57–61.

Casadesús, J., M. Mata, J. Marsal y J. Girona (2012), «A general algorithm for automated scheduling of drip irrigation in tree crops», *Computers and Electronics in Agriculture*, **83**(0), págs. 11 – 20.

Chazelle, B. (1983), «The bottomn-left bin-packing heuristic: An efficient implementation», *Computers, IEEE Transactions on*, **100**(8), págs. 697–707.

Chinchuluun, A. y P. M. Pardalos (2007), «A survey of recent developments in multiobjective optimization», *Annals of Operations Research*, **154**(1), págs. 29–50.

Christofides, N. y C. Whitlock (1977), «An algorithm for two-dimensional cutting problems», *Operations Research*, **25**(1), págs. 30–44.

Chvatal, V. (1983), *Linear programming*, Macmillan.

Cid-Garcia, N. M., V. Albornoz, Y. A. Rios-Solis y R. Ortega (2013), «Rectangular shape management zone delineation using integer linear programming», *Computers and Electronics in Agriculture*, **93**(0), págs. 1–9.

Cid-Garcia, N. M., A. G. Bravo-Lozano y Y. A. Rios-Solis (2014), «A crop planning and real-time irrigation method based on site-specific management zones and linear programming», *Computers and Electronics in Agriculture*, **107**, págs. 20–28.

Cid-Garcia, N. M. y Y. A. Rios-Solis (2015), «Positions and Covering: A Two-stage Methodology to Obtain Exact Solutions for the 2D-Bin Packing Problem», Paper under revision. Submitted in: European Journal of Operations Research.

Coello, C. A. C., D. A. Van Veldhuizen y G. B. Lamont (2002), *Evolutionary algorithms for solving multi-objective problems*, tomo 242, Springer.

Conejo, A. J., E. Castillo, R. Minguez y R. Garcia-Bertrand (2006), *Decomposition techniques in mathematical programming: engineering and science applications*, Springer Science & Business Media.

Dagli, C. H. y P. Poshyanonda (1997), «New approaches to nesting rectangular patterns», *Journal of Intelligent Manufacturing*, **8**(3), págs. 177–190.

Dantzig, G. B. (1998), *Linear programming and extensions*, Princeton university press.

Dantzig, G. B. y M. N. Thapa (1997), *Linear programming 1: introduction*, Springer series in operations research.

Diker, K., D. F. Heermann y M. K. Brodahl (2004), «Frequency analysis of yield for delineating yield response zones», *Precision Agriculture*, **5**(5), págs. 435–444.

Doerge, T. (1999), «Management zone concepts», *Site specific management guidelines*, págs. SSMG–2, potash and Phosphate Institute.

Doorenbos, J., A. Kassam y C. Bentvelsen (1979), *Efectos del agua sobre el rendimiento de los cultivos*, 33, FAO.

Dowsland, K. A. y W. B. Dowsland (1992), «Packing problems», *European Journal of Operational Research*, **56**(1), págs. 2–14.

Ehrgott, M. (2005), *Multicriteria optimization*, tomo 2, Springer Berlin.

Ehrgott, M. (2006), *Multicriteria optimization*, Springer Science & Business Media.

Faroe, O., D. Pisinger y M. Zachariasen (2003), «Guided local search for the three-dimensional bin-packing problem», *INFORMS Journal on Computing*, **15**(3), págs. 267–283.

Fekete, S. P. y J. Schepers (1997), «A new exact algorithm for general orthogonal d-dimensional knapsack problems», en *Algorithms ESA'97*, Springer, págs. 144–156.

Fekete, S. P. y J. Schepers (2004), «A combinatorial characterization of higher-dimensional orthogonal packing», *Mathematics of Operations Research*, **29**(2), págs. 353–368.

Fekete, S. P., J. Schepers y J. C. Van der Veen (2007), «An exact algorithm for higher-dimensional orthogonal packing», *Operations Research*, **55**(3), págs. 569–587.

Figueira, J., S. Greco y M. Ehrgott (2005), *Multiple criteria decision analysis: state of the art surveys*, tomo 78, Springer Science & Business Media.

Fraisse, C. W., K. A. Sudduth y N. R. Kitchen (2001), «Delineation of site-specific management zones by unsupervised classification of topographic attributes and soil electrical conductivity», *Transactions of the ASAE*, **44**(1), págs. 155–166.

Franzen, D. W. y T. N. Nanna (2003), «Management zone delineation methods», en P. C. Robert (editor), *Proceedings of the 6th International Conference on Precision Agriculture and Other Precision Resources Management, Minneapolis, MN, USA, 14-17 July, 2002*, American Society of Agronomy, págs. 443–457.

Frogbrook, Z. L. y M. A. Oliver (2007), «Identifying management zones in agricultural fields using spatially constrained classification of soil and ancillary data», *Soil Use and Management*, **23**(1), págs. 40–51.

Garey, M. y D. Johnson (1979), *Computers and intractability*, tomo 174, Freeman San Francisco, CA.

GEERTS, S. y D. RAES (2009), «Deficit irrigation as an on-farm strategy to maximize crop water productivity in dry areas», *Agricultural Water Management*, **96**(9), págs. 1275–1284.

GILMORE, P. y R. E. GOMORY (1965), «Multistage cutting stock problems of two and more dimensions», *Operations research*, **13**(1), págs. 94–120.

GONÇALVES, J. F. y M. G. RESENDE (2011), «A parallel multi-population genetic algorithm for a constrained two-dimensional orthogonal packing problem», *Journal of Combinatorial Optimization*, **22**(2), págs. 180–201.

GONÇALVES, J. F. y M. G. RESENDE (2013), «A biased random key genetic algorithm for 2D and 3D bin packing problems», *International Journal of Production Economics*, **145**(2), págs. 500–510.

HAESSLER, R. W. y P. E. SWEENEY (1991), «Cutting stock problems and solution procedures», *European Journal of Operational Research*, **54**(2), págs. 141–150.

HASSANLI, A., M. EBRAHIMIZADEH y S. BEECHAM (2009), «The effects of irrigation methods with effluent and irrigation scheduling on water use efficiency and corn yields in an arid region», *Agricultural Water Management*, **96**(1), págs. 93–99.

HEDLEY, C. y I. YULE (2009), «A method for spatial prediction of daily soil water status for precise irrigation scheduling», *Agricultural Water Management*, **96**(12), págs. 1737 – 1745.

HIFI, M. (1998), «Exact algorithms for the guillotine strip cutting/packing problem», *Computers & Operations Research*, **25**(11), págs. 925–940.

HILLIER FREDERICK, S. y J. LIEBERMAN GERALD (2010), *Introducción a la investigación de operaciones*, Mc Graw Hill.

HOCHBAUM, D. S. y W. MAASS (1985), «Approximation schemes for covering and packing problems in image processing and VLSI», *Journal of the ACM (JACM)*, **32**(1), págs. 130–136.

Hopper, E. y B. C. Turton (2001a), «An empirical investigation of metaheuristic and heuristic algorithms for a 2D packing problem», *European Journal of Operational Research*, **128**(1), págs. 34–57.

Hopper, E. y B. C. Turton (2001b), «A review of the application of metaheuristic algorithms to 2D strip packing problems», *Artificial Intelligence Review*, **16**(4), págs. 257–300.

Hornung, A., R. Khosla, R. Reich, D. Inman y D. G. Westfall (2006), «Comparison of site-specific management zones: Soil-color-based and yield-based», *Agronomy Journal*, **98**(2), págs. 407–415.

Hornung, A., R. Khosla, R. Reich y D. G. Westfall (2003), «Evaluation of site-specific management zones: Grain yield and nitrogen use efficiency», en *Proceedings of the European Conference on Precision Agriculture, 4th. Wageningen Academic Publ., Wageningen, the Netherlands*, págs. 297–302.

INIA (2006), *Métodos de análisis recomendados para los suelos de Chile*, Instituto de Investigaciones Agropecuarias, Santiago de Chile.

Iori, M., S. Martello y M. Monaci (2003), «Metaheuristic algorithms for the strip packing problem», en *Optimization and industry: new frontiers*, Springer, págs. 159–179.

Itoh, T., H. Ishii y T. Nanseki (2003), «A model of crop planning under uncertainty in agricultural management», *International Journal of Production Economics*, **81–82**(0), págs. 555 – 558.

Jaimes, A. L., S. Z. Martinez y C. A. C. Coello (2009), «An Introduction to Multiobjective Optimization Techniques», *Optimization in Polymer Processing*, págs. 29–57.

Jakobs, S. (1996), «On genetic algorithms for the packing of polygons», *European journal of operational research*, **88**(1), págs. 165–181.

Jaynes, D. B., T. S. Colvin y T. C. Kaspar (2005), «Identifying potential soybean management zones from multi-year yield data», *Computers and Electronics in Agriculture*, **46**(1), págs. 309–327.

Jiang, Q., Q. Fu y Z. Wang (2011), «Study on Delineation of Irrigation Management Zones Based on Management Zone Analyst Software», en *Computer and Computing Technologies in Agriculture IV*, Springer, págs. 419–427.

Karmarkar, N. (1984), «A new polynomial-time algorithm for linear programming», en *Proceedings of the sixteenth annual ACM symposium on Theory of computing*, ACM, págs. 302–311.

Kenmochi, M., T. Imamichi, K. Nonobe, M. Yagiura y H. Nagamochi (2009), «Exact algorithms for the two-dimensional strip packing problem with and without rotations», *European Journal of Operational Research*, **198**(1), págs. 73–83.

Khachiyan, L. G. (1980), «Polynomial algorithms in linear programming», *USSR Computational Mathematics and Mathematical Physics*, **20**(1), págs. 53–72.

Lesh, N., J. Marks, A. McMahon y M. Mitzenmacher (2004), «Exhaustive approaches to 2D rectangular perfect packings», *Information Processing Letters*, **90**(1), págs. 7–14.

Lesh, N., J. Marks, A. McMahon y M. Mitzenmacher (2005), «New heuristic and interactive approaches to 2D rectangular strip packing», *Journal of Experimental Algorithmics (JEA)*, **10**, págs. 1–2.

Li, X., Y. Pan, C. Zhang, L. Liu y J. Wang (2005), «A new algorithm on delineation of management zone», en *Geoscience and Remote Sensing Symposium, 2005. IGARSS'05. Proceedings. 2005 IEEE International*, tomo 1, IEEE, págs. 4–pp.

Liu, D. y H. Teng (1999), «An improved BL-algorithm for genetic algorithm of the orthogonal packing of rectangles», *European Journal of Operational Research*, **112**(2), págs. 413–420.

Lodi, A., S. Martello y M. Monaci (2002a), «Two-dimensional packing problems: A survey», *European Journal of Operational Research*, **141**(2), págs. 241–252.

Lodi, A., S. Martello y D. Vigo (1999a), «Approximation algorithms for the oriented two-dimensional bin packing problem», *European Journal of Operational Research*, **112**(1), págs. 158–166.

Lodi, A., S. Martello y D. Vigo (1999b), «Heuristic and metaheuristic approaches for a class of two-dimensional bin packing problems», *INFORMS Journal on Computing*, **11**(4), págs. 345–357.

Lodi, A., S. Martello y D. Vigo (1999c), «Neighborhood search algorithm for the guillotine non-oriented two-dimensional bin packing problem», en *Meta-Heuristics*, Springer, págs. 125–139.

Lodi, A., S. Martello y D. Vigo (2002b), «Recent advances on two-dimensional bin packing problems», *Discrete Applied Mathematics*, **123**(1), págs. 379–396.

Lodi, A., S. Martello y D. Vigo (2004), «TSpack: a unified tabu search code for multi-dimensional bin packing problems», *Annals of Operations Research*, **131**(1-4), págs. 203–213.

Mainuddin, M., A. D. Gupta y P. R. Onta (1997), «Optimal crop planning model for an existing groundwater irrigation project in Thailand», *Agricultural Water Management*, **33**(1), págs. 43 – 62.

Marler, R. T. y J. S. Arora (2004a), «Survey of multi-objective optimization methods for engineering», *Structural and Multidisciplinary Optimization*, **26**(6), págs. 369–395.

Marler, R. T. y J. S. Arora (2004b), «Survey of multi-objective optimization methods for engineering», *Structural and multidisciplinary optimization*, **26**(6), págs. 369–395.

Martello, S., M. Monaci y D. Vigo (2003), «An exact approach to the strip-packing problem», *INFORMS Journal on Computing*, **15**(3), págs. 310–319.

Martello, S. y D. Vigo (1998), «Exact solution of the two-dimensional finite bin packing problem», *Management Science*, **44**(3), págs. 388–399.

McBratney, A., B. Whelan, T. Ancev y J. Bouma (2005), «Future directions of precision agriculture», *Precision Agriculture*, **6**(1), págs. 7–23.

McCarthy, A., N. Hancock y S. R. Raine (2008), «Towards evaluation of adaptive control systems for improved site-specific irrigation of cotton», en *Irrigation Australia 2008: Irrigation Association of Australia National Conference and Exhibition: Proceedings*, Irrigation Australia Ltd.

McCarthy, A. C., N. H. Hancock y S. R. Raine (2013), «Advanced process control of irrigation: the current state and an analysis to aid future development», *Irrigation Science*, págs. 1–10.

McCauley, A., C. Jones y J. Jacobsen (2005), «Soil and Water Management Module 1: Basic Soil Properties», *A self-study course from the msu extension service continuing education series*, Montana State University, URL http://landresources.montana.edu/SWM/PDF/Final_proof_SW1.pdf.

Monaci, M. y P. Toth (2006), «A set-covering-based heuristic approach for bin-packing problems», *INFORMS Journal on Computing*, **18**(1), págs. 71–85.

Moore, S. H. y M. C. Wolcott (2000), «Using yield maps to create management zones in field crops», *Louisiana Agriculture*, **43**(3), págs. 12–13.

Mortensen, D. A., C. K. Johnson, J. W. Doran, J. F. Shanahan y B. J. Wienhold (2003), «Site-specific management zones based on soil electrical conductivity in a semiarid cropping system», *Agronomy Journal*, **95**(2), págs. 303–315.

MULLA, D. J. (1991), «Using geostatistics and GIS to manage spatial patterns in soil fertility», en *Proceedings of the Automated Agricultural for the 21st Century*, American Society of Agricultural Engineers, págs. 336–345.

MUMFORD-VALENZUELA, C. L., J. VICK y P. Y. WANG (2004), «Heuristics for large strip packing problems with guillotine patterns: An empirical study», en *Metaheuristics: computer decision-making*, Springer, págs. 501–522.

NTENE, N. y J. H. VAN VUUREN (2009), «A survey and comparison of guillotine heuristics for the 2D oriented offline strip packing problem», *Discrete Optimization*, **6**(2), págs. 174–188.

ORTEGA, J. A., W. FOSTER y R. ORTEGA (2002), «Definición de sub-rodales para una silvicultura de precisión: una aplicación del método Fuzzy K-means», *Ciencia e Investigación Agraria*, **29**(1), págs. 35–44.

ORTEGA, R. y L. FLORES (1999), «Introducción al manejo sitio-específico», en R. Ortega y L. Flores (editores), *Agricultura de Precisión*, Ministerio de Agricultura, Instituto de Investigaciones Agropecuarias, Centro Regional de Investigación Quilamapu, págs. 13–46.

ORTEGA, R. A. y O. A. SANTIBÁÑEZ (2007), «Determination of management zones in corn (*Zea mays* L.) based on soil fertility», *Computers and Electronics in Agriculture*, **58**(1), págs. 49–59.

ORTEGA ÁLVAREZ, J. F., J. A. JUAN VALERO, J. M. TARJUELO MARTÍN-BENITO y E. LÓPEZ MATA (2004), «MOPECO: an economic optimization model for irrigation water management», *Irrigation Science*, **23**, págs. 61–75.

PAPADIMITRIOU, C. H. y M. YANNAKAKIS (2000), «On the approximability of trade-offs and optimal access of web sources», en *Foundations of Computer Science, 2000. Proceedings. 41st Annual Symposium on*, IEEE, págs. 86–92.

Pardalos, P. M. y M. G. Resende (2001), *Handbook of applied optimization*, Oxford University Press.

Parreño, F., R. Alvarez-Valdés, J. Oliveira y J. M. Tamarit (2010), «A hybrid GRASP/VND algorithm for two-and three-dimensional bin packing», *Annals of Operations Research*, **179**(1), págs. 203–220.

Pedroso, M., J. Taylor, B. Tisseyre, B. Charnomordic y S. Guillaume (2010), «A segmentation algorithm for the delineation of agricultural management zones», *Computers and Electronics in Agriculture*, **70**(1), págs. 199–208.

Poshyanonda, P. y C. H. Dagli (2004), «Genetic neuro-nester», *Journal of Intelligent Manufacturing*, **15**(2), págs. 201–218.

Ratanapan, K. y C. Dagli (1997), «An object-based evolutionary algorithm for solving irregular nesting problems», en *Proceedings Artificial Neural Networks in Engineering Conference*, págs. 383–388.

Ratanapan, K. y C. H. Dagli (1998), «An object-based evolutionary algorithm: the nesting solution», en *Evolutionary Computation Proceedings, 1998. IEEE World Congress on Computational Intelligence., The 1998 IEEE International Conference on*, IEEE, págs. 581–586.

Reddy, M. y D. Kumar (2008), «Evolving strategies for crop planning and operation of irrigation reservoir system using multi-objective differential evolution», *Irrigation Science*, **26**(2), págs. 177–190.

Riff, M. C., X. Bonnaire y B. Neveu (2009), «A revision of recent approaches for two-dimensional strip-packing problems», *Engineering Applications of Artificial Intelligence*, **22**(4), págs. 823–827.

Ríos-Mercado, R. Z. (2002), «Natural gas pipeline optimization», *Handbook of applied optimization*, págs. 813–826.

Roudier, P., B. Tisseyre, H. Poilvé y J. M. Roger (2008), «Management zone delineation using a modified watershed algorithm», *Precision Agriculture*, **9**(5), págs. 233–250.

Sahoo, B., A. Lohani y R. Sahu (2006), «Fuzzy Multiobjective and Linear Programming Based Management Models for Optimal Land-Water-Crop System Planning», *Water Resources Management*, **20**, págs. 931–948.

Sarker, R. y T. Ray (2009), «An improved evolutionary algorithm for solving multi-objective crop planning models», *Computers and Electronics in Agriculture*, **68**(2), págs. 191–199.

Sarker, R. A. y C. S. Newton (2007), *Optimization modelling: a practical approach*, CRC Press.

Sarker, R. A., S. Talukdar y A. Haque (1997), «Determination of optimum crop mix for crop cultivation in Bangladesh», *Applied Mathematical Modelling*, **21**(10), págs. 621 – 632.

Schepers, J. S., A. Luchiari, S. H. Johnson, M. A. Liebig, J. F. Shanahan y A. R. Schepers (2004), «Appropriateness of management zones for characterizing spatial variability of soil properties and irrigated corn yields across years», *Agronomy Journal*, **96**(1), págs. 195–203.

Silva, E., F. Alvelos y J. Valério de Carvalho (2010), «An integer programming model for two-and three-stage two-dimensional cutting stock problems», *European Journal of Operational Research*, **205**(3), págs. 699–708.

Simbahan, G. C. y A. Dobermann (2006), «An algorithm for spatially constrained classification of categorical and continuous soil properties», *Geoderma*, **136**(3), págs. 504–523.

Smith, R. J., S. R. Raine, A. C. McCarthy y N. H. Hancock (2009), «Managing Spatial and Temporal Variability in Irrigated Agriculture through Adaptive Control», *Australian Journal of Multi-disciplinary Engineering*, **7**(1), págs. 79–90.

Taha, H. A. (2004), *Investigación de operaciones*, Pearson Educación.

Vira, C. y Y. Y. Haimes (1983), *Multiobjective decision making: theory and methodology*, 8, North-Holland.

Wäscher, G., H. Haussner y H. Schumann (2007), «An improved typology of cutting and packing problems», *European Journal of Operational Research*, **183**(3), págs. 1109–1130.

Whelan, B. M., J. Cupitt y A. B. McBratney (2003), «Practical definition and interpretation of potential management zones in Australian dryland cropping», en P. C. Robert (editor), *Proceedings of the 6th International Conference on Precision Agriculture and Other Precision Resources Management, Minneapolis, MN, USA, 14-17 July, 2002*, American Society of Agronomy, págs. 395–409.

Williams, H. P. (1999), *Model building in mathematical programming*, John Wiley.

Winston, W. L., M. B. Anzures y F. S. Fragoso (2005), *Investigación de operaciones: aplicaciones y algoritmos*, Thomson.

Wolsey, L. A. (1998), *Integer programming*, Wiley-Interscience series in discrete mathematics and optimization, Wiley & Sons.

Xu, L., L. Chen, T. Chen y Y. Gao (2011), «SOA-based precision irrigation decision support system», *Mathematical and Computer Modelling*, **54**(3), págs. 944 – 949.

Zopounidis, C. y P. Pardalos (2010), *Handbook of multicriteria analysis*, tomo 103, Springer Science & Business Media.

Ficha autobiográfica

M.C. Néstor Miguel Cid García

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

Universidad Autónoma de Nuevo León

Facultad de Ingeniería Mecánica y Eléctrica

Thesis:

Exact Solutions for the Agricultural and the Two-Dimensional Packing Problems

Nestor was born on January 20, 1985 in the municipality of Morelos, Zacatecas, Mexico. He was the second of four children of Jaime Cid and Ana María García. He has a brother, Jaime, and two sisters, Viridiana y Fátima.

He was graduated with honors at the Autonomous University of Zacatecas obtaining the degree in Computer Engineering, generation 2003-2008.

From 2010 through 2012, he attended the Graduate Program in Systems Engineering of the Faculty of Mechanical and Electrical Engineering of the Autonomous University of Nuevo León. In May of 2012, Nestor got with honors his Master's degree, with a research developed under the direction of Ph.D. Yasmín Á. Ríos Solís.

In his Master's studies (October 2011 - January 2012), Nestor performed a research stay at the Technical University Federico Santa María, Campus Santiago, Vitacura, Chile, under the supervision of Ph.D. Victor Albornoz Sanhueza in collaboration with Ph.D. Rodrigo Ortega Blu.

His Ph.D. studies were carried out at the Graduate Program in Systems Engineering of the Autonomous University of Nuevo León, generation 2012-2015, developing this research project under the direction of Ph.D. Yasmín A. Ríos Solís.

From March to May 2015, Nestor performed a research stay at the Laboratory for Analysis and Architecture of Systems, Toulouse, France, under the supervision of Ph.D. Sandra Ulrich Ngueveu.

During his Ph.D. and Master's studies, Nestor attended to several national and international conferences, highlighting the ISMP at Pittsburgh, USA; the SEIO at Castellon, Spain; the ELAVIO at Valencia, Spain; the CLAIO at Monterrey, Mexico; four conferences of the SMIO at Guadalajara, Acapulco, Monterrey and Ciudad Juárez, Mexico; and, three conferences of the ENOAN at Cuernavaca, Tabasco and Saltillo, Mexico.