A linear optimization approach for increasing sustainability in vegetable crop production

Lana Mara R. dos Santos[1], Marcos N. Arenales[2], Alysson M. Costa[2], Ricardo H. S. Santos[3]

[1]*Departamento de Matemática – Universidade Federal de Viçosa, Brazil.*
[2]*Instituto de Ciências Matemáticas e de Computação – Universidade de São Paulo, Brazil.*
[3]*Departamento de Fitotecnia – Universidade Federal de Viçosa, Brazil.*

ABSTRACT

This chapter is concerned with a set of optimization problems associated to crop rotation scheduling in the context of vegetable crop production according to some ecological criteria: no crop of the same botanic family is planted in sequence, green manure and fallow periods must be present in any schedule. A core mathematical model called the crop rotation scheduling model is proposed to represent these ecological criteria together with specific technical constraints associated to the growing of vegetable crops. Three optimization problems based on crop rotation schedules are written in detail in this chapter. For each problem, we present a general modeling framework and a solution methodology based on a technique known as column generation, which iteratively builds crop rotation plans for a number of plots. Some extensions are also presented, with the aim of incorporating additional characteristics found in production field conditions. This chapter ends with a brief discussion on a set of computational experiments and some suggestions for future research.

INTRODUCTION

Large monoculture-based production systems have been present in Brazil since sugar cane was grown for sugar production in the 16th – 18th centuries. Furthermore, it became more important in the sixties and seventies. Although these systems are able to produce food at low prices, they are also associated with many social and environmental problems. Indeed, besides depending on extensive use of highly soluble mineral fertilizers, pesticides and fossil energy, large monocultures need a small number of employees and usually exclude small family growers due to the dependence on large amounts of capital. This has raised concerns about the economical, environmental and social sustainability of the modern food production system. In response to these concerns, alternative production models such as 'Natural', 'Organic', and 'Biodynamic' agricultures have attracted attention. Implementing these models has been boosted by private initiatives, such as the promotion of specific fairs and distribution strategies, as well as governmental incentives in the form of regulations and guidelines for organic production in the country.

The main cornerstone of more sustainable models in crop production is crop diversification and the introduction of biological processes in the system. Various agricultural and ecological studies demonstrate that the increase in crop species diversity leads to beneficial effects, such as better exploitation of productive resources, less pest damage, a higher control of spontaneous weeds and an increase in production stability (Altieri, 1995; Gliessman, 2000).

Biological processes are fundamental for maintaining ecological-based crop production. Among the important processes that should be stimulated, we can cite nutrient cycling, nitrogen biological fixation and the processes associated with the thriving of spontaneous vegetation at specific times and places. Crop diversification associated to biological processes in cropping systems allows for the conservation of

productive resources, as well as less dependence on industrial inputs, such as fertilizers and pesticides. Moreover, they might reduce the production costs and result in social benefits, such as a decrease in grower's vulnerability to the market prices and a better distribution of grower's income and food for the population throughout the year. One of the most widespread strategies for enhancing crop diversification over time is crop rotation. In these production systems, a sequence of different crops is cultivated in the same field in a finite time interval and this production sequence is repeated indefinitely. Besides the mentioned potential benefits, crop rotation helps to protect soil from eroding and makes it possible to use the land in another period during the year.

Among the ecological criteria and biological methods that can be used to define rotation plans, one might list the following (Gliessman, 2000):

(a) avoid the growth of a crop immediately after a crop from the same botanical family has been harvested as many herbivores and pathogens only attack crops from a specific botanical family;
(b) include green manure crops (usually a leguminous) for biological nitrogen fixation; and
(c) include fallow periods, in which the spontaneous vegetation is left to thrive for a given period of time, thus contributing to the biological control of insects and improving soil biological, chemical and physical characteristics.

In this chapter, we focus on the scheduling of vegetable crop rotation. In this context, one must consider that vegetable crops embed many different species and cultivars, they present very different cropping times and a high specific cropping season due to their sensibility to climatic conditions. Vegetable crops are also more susceptible to pest attacks. Moreover, it is common practice to cultivate many different crops in the same area throughout the year. These characteristics make the rotation planning using ecological criteria (a)–(c) a complex task, which is why the use of support software is very useful, if not essential. In addition, for commercial reasons, vegetable crop growers must ensure a constant supply of products, sometimes to meet known demands, which is a characteristic that further complicates production planning.

BACKGROUND

The idea of obtaining crop rotation plans with the aid of mathematical tools is not new to operation researchers. Indeed, as early as 1939, Kantorovich had already mentioned this problem as one that could be dealt with by linear programming (Kantorovich, 1960 – translated from the Russian original, dated 1939). The author proposes a mathematical model which selects the optimal partition of land in order to maximize production. Hildreth and Reiter (1951), in turn, in one of the first conferences on applications of operations research, stated that crop rotation can be beneficial as it can increase crop yields (due to the fact that different rotations produce distinct effects on the soil). Moreover, the authors had already mentioned that crop diversification provides some sort of security against the failure of production of a given crop in a given year in addition to the fact that by dividing the available land among different rotations, one might reduce the need for resources (such as water, labor, etc) as different crops also have distinct needs throughout the year. The authors use a list of pre-determined crop rotations, which greatly simplifies the problem.

Since these two seminal papers, crop rotation schedules have been obtained with the aid of (integer) linear optimization models such as those proposed by El-Nazer and McCarl (1986), Clarke (1989), Haneveld and Stegeman (2005), Detlefsen and Jensen (2007), Santos et al. (2008; 2010) and Costa et al. (2009). Other manuscripts on crop rotation concentrate on the description of decision-aid tools which somehow evaluate the benefits of a given crop rotation plan (Jones et al., 2003; Stöckle et al., 2003, Bachinger and Zander, 2006). Among this list of contributions, we are briefly going to describe those which have made use of some optimization approach in order to obtain solutions which maximize or minimize a given goal.

El-Nazer and McCarl (1986) assume that the yield of a crop depends only on the crops grown on the same piece of land in the previous years. Using this simplification, they are able to eliminate the need for pre-determining all possible feasible crop rotation schedules. Haneveld and Stegeman (2005) use the idea of crop succession requirements, which are expressed in terms of an infeasible sequence of crops. Detlefsen and Jensen (2007), in turn, take into account the fact that crop rotation influences the needs for nitrogen and the yield of the fields. Their model considers that the amount of land to be planted with each crop is given, for each year, which enables the formulation of the crop rotation problem as a transportation problem (Appa, 1973). In this and in the previous mentioned articles, the crop production times are assumed to be equal to one year, the available cropping area is homogeneous and the authors explicitly forbid infeasible crop

sequences. Clarke (1989) considers crops with different production times and planting seasons of the year. However, the proposed model does not allow constraints concerning the feasible sequence of crops.

Sustainability constraints of the type described in the introduction have been used by Santos et al. (2008; 2010) and by Costa et al. (2009), who have dealt with crop rotation problems focusing on the sustainable production of vegetable crops. Production planning of vegetable crops is particularly complex as it generally involves a large number of crop species having different specific limitations regarding the planting seasons and highly varied production times and productivity. Santos et al. (2008) have considered the scheduling of crop rotations in order to avoid the spatial propagation of pests: crops of the same botanical family are not allowed to grow simultaneously in adjacent plots. The goal was to maximize production but no demand aspects were considered. Santos et al. (2010) have introduced the idea of a "supply problem" with the inclusion of known demands. Costa et al. (2009) have extended this previous work to include uncertainty considerations and to study the effect of allowing products to be stocked. Due to the perishable nature of the products, stocking was only allowed for a limited number of periods and with associated losses.

MAIN FOCUS OF THE CHAPTER

In this chapter, we present a general modeling and solution framework for the three optimization problems associated with vegetable crop rotation scheduling mentioned in the last paragraph:

(1) The crop rotation scheduling problem with known demands (Santos et al., 2010);
(2) The crop rotation scheduling problem with perishable stocks (Costa et al., 2009);
(3) The crop rotation scheduling problem with adjacency constraints (Santos et al., 2008).

Our idea here is to give a general framework for these studies in order to facilitate the task of modeling new similar problems and developing solution methodologies. We concentrate on the deterministic case. The reader who is interested in uncertainty considerations is refereed to (Costa et al., 2009).

The remainder of this chapter is organized as follows. Next, we present the concept of crop rotation in more detail and develop a crop rotation scheduling model, which respects ecologically-based criteria (a)-(c) and specific growing technical constraints. We then discuss each one of the above problems (1)-(3) and show how the crop rotation scheduling model can be seen as a core of the mathematical formulations developed in all of these cases (and, certainly, of many other similar variants). Using the same modeling framework allows us to develop a similar solution methodology for the three cases. This methodology, which is based on the iterative generation of crop rotation schedules respecting both ecological and technical constraints, will also be described in detail. This chapter ends with a brief discussion on some computational results that indicate the efficacy of the proposed strategy, as well as some conclusions and suggestions for future research.

THE CROP ROTATION SCHEDULING MODEL

In an agricultural production system, a crop rotation can be defined as the sequence of crops that should be planted, one after another, in a given area during a time interval. This time interval is called the length of the rotation and the idea of *rotation* comes from the fact that the same sequence is repeated indefinitely. For instance, the 3 year crop rotation corn-potatoes-potatoes indicate that one should grow corn, then one should grow potatoes twice, after which the rotation should be restarted with the growing of corn and so on.

Most of the articles mentioned earlier deal with grain crops and/or high latitude environments and, in these cases, each crop has a specific growing season and production time. However, in the context of vegetable crop production, the definition of interesting crop rotation plans has some particularities. Vegetable crops embed many species and cultivars which have different cropping times and usually have specific cropping seasons, due to their sensibility to climatic conditions. Thus the vegetable crop schedule must consider the different growing and production times of each crop, which greatly complicates the problem. In the case of the vegetable crop rotation schedule problems presented, we first define the time interval in which the sequence of crops will be repeated, say T years, which is divided into M periods. A crop rotation schedule is a planting calendar for each crop in the rotation.

Moreover, ecologically-based production systems impose additional constraints on the crop rotations. Among many possible sustainability practices, we concentrate on the following constraints that result from the ecological criteria (a)-(c) mentioned in the previous section:

(i) *Botanic family*: crops belonging to the same botanic family can not be planted in sequence.
(ii) *Green manure*: crops for green manuring must be planted in each crop rotation.
(iii) *Fallow*: a minimum number of fallow periods must be present in each rotation.

As an example, let us consider a one year- crop rotation schedule, divided into twelve one month-periods (M = 12). There are three crops, X, Y and Z with production times of 5, 4 and 2 months, respectively. Crops X and Y belong to the same botanic family while Z is a crop for green manuring. Crop X can be planted between January (month 1) and July (month 7), while crops Y and Z can be planted all year round. Finally, let us assume that one green manure and one month of fallow (F) are mandatory per schedule. According to (i)-(iii) constraints, a feasible crop rotation schedule is depicted in Fig. 1.

IMAGE GOES HERE

Figure 1. *A feasible crop rotation schedule.*

In the crop rotation schedule presented in Fig. 1, crop X is planted in month 3 and occupies the land until period 7 (as its production time is 5 months). In period 8, crop Z is planted, staying on the land until period 9, when it is harvested. In period 10, crop Y is planted and its final harvesting occurs in period 1 (of the next production year). In period 2, there is one month fallow (F). Each crop respects its planting times and crops X and Y (which belong to the same botanic family) are not planted in sequence. Moreover, as desired, in each crop rotation a green manure crop (Z) is cultivated and there is a period of fallow (F).

As the planting of a crop in a given period implies one or more subsequent harvesting periods, with known productivities, one might easily convert the crop rotation schedule into a crop production schedule, which is a harvesting calendar indicating the amount (in mass, volume, units or other appropriate measure) of each crop which is being produced in each period.

Consider the crop rotation schedule in Fig. 1. Assume that crop X has its first harvest three months after planting and other harvests occur monthly, until the final harvest, five months after the time of planting. Suppose the productions in months 3, 4 and 5 after planting are 1, 2 and 1 kg/m^2. Crop Y has no partial harvestings and its production is estimated at 3 kg/m^2 with the harvesting occurring in the last period of its production time, i.e., four months after planting. Having these data, and assuming the planting calendar in Fig. 1 has been used in an area of 2 m^2, we can express the harvesting calendar (or production schedule) in Fig. 2. The numbers in italics represent the partial harvests of each crop: 2, 4 and 2 kg of X in months 5, 6 and 7, respectively, and 6 kg of Y in month 1. Crop Z is a green manure.

IMAGE GOES HERE

Figure 2. *Production schedule for a 2 m^2 cropping area.*

In the remainder of this section, we present a linear model to describe all feasible crop rotation schedules according to the ecological criteria (i)-(iii). In order to do this, we introduce the following notation:

M number of periods in each crop rotation (in a predefined unit of time);
C set of crops that can be selected for planting, except the ones associated to green manuring;
G set of crops that can be selected for green manuring;
N cardinality of $C \cup G$;
NF number of the botanic families;
$F(p)$ set of crops of botanic family p, $p = 1, \ldots, NF$;
t_i production time of crop i, including the time estimated for preparing the soil and harvesting;
I_i set of periods in which crop i can be planted;
$N+1$ hypothetical crop for fallow. We adopt $I_{N+1} = \{1,\ldots, M\}$.

We also define the decision variables:

$x_{ij} = 1$, if crop i is planted in period j, 0 otherwise, $i = 1, \ldots, N+1, j \in I_i$.

The model that describes all feasible crop rotation schedules (CRS model) can thus be written as:

$$\sum_{i=1}^{N+1}\sum_{r=0}^{t_i-1} x_{i,j-r} \le 1,\ j=1,...,M \tag{1}$$

$$\sum_{i\in F(p)}\sum_{r=0}^{t_i} x_{i,j-r} \le 1,\ p=1,...,NF, j=1,...,M \tag{2}$$

$$\sum_{i\in G}\sum_{j\in I_i} x_{ij} = 1 \tag{3}$$

$$\sum_{j=1}^{M} x_{N+1,j} = 1 \tag{4}$$

$$x_{ij} \in \{0,1\},\ i=1,...,N+1, j\in I_i \tag{5}$$

where, in constraints (1)–(2), $j-r$ is replaced by $j-r+M$, whenever it is less or equal to zero.

Constraints (1) ensure that at most one crop is cultivated at each period while constraints (2) forbid crops belonging to the same botanic family from being cultivated one immediately after the other. Constraints (3) and (4) force the planting of green manure and the existence of fallow. The model above allows for solutions in which a given period has no associated crop. This is equivalent to allowing more times of fallow in each schedule. Finally, constraints (5) define the used variables.

In model (1)–(5), the numbers of mandatory green manures and fallows have been set at one. Depending on the crop rotation length (T), these numbers might need to be increased. In this case, consider g the number of crops in G that must be planted per crop rotation, and f, the number of fallows per crop rotation. Constraints (3) and (4) can then be written as:

$$\sum_{i\in G}\sum_{j\in I_i} x_{ij} = g \tag{6}$$

$$\sum_{j=1}^{M} x_{N+1,j} = f \tag{7}$$

One might also need to ensure a minimum time delay between the planting of two green manures or two fallows. This can be done with the following additional constraints:

$$\sum_{i\in G}\sum_{r=0}^{tg} x_{i,j-r} \le 1,\ j=1,...,M \tag{8}$$

$$\sum_{r=0}^{tf} x_{N+1,j-r} \le 1,\ j=1,...,M \tag{9}$$

where tg and tf are the minimum time delays for green manures and fallows.

SOME CROP ROTATION OPTIMIZATION PROBLEMS

Many optimization problems are associated with obtaining crop rotation. In this section, we describe three of them and show how, in each of the cases, constraints (1)–(5) appear as a core part of the model.

The crop rotation scheduling problem with known demands (CRS-D problem)

There is increasing interest in the study of vegetable crop production as it represents an important alternative income for family farmers. These farmers usually own small plots of land and, therefore, have small associated productions. These productions are usually discontinuous, mainly due to climatic conditions,

which restrict the growth of some species in some regions or lead to significant reductions in the yields. These small and discontinuous productions do not justify investments in cleaning, packing and logistic structures necessary to make production available to customers.

In order to cope with this problem, an increasing number of farmers are gathering together in collective movements, such as cooperatives, so that the combined productions justify the investments in the necessary infrastructure. These cooperatives have the role of deciding the amount of land to be planted for each crop so that the production supplies the demand. In addition, the cooperatives must decide when such productions should occur, by taking into consideration differences between soil and climatic conditions for each one of the farmers' land, which might influence the crop yields and even forbid farmers to grow some crops.

Santos et al. (2010) deal with this problem with the aim of obtaining crop rotation plans that meet demands for vegetable crops, given a set of heterogeneous (in respect to soil and climatic conditions) pieces of land. Because of this, the set of growing crops and the associated productivity might differ from one cropping area to another. This problem is written as a linear optimization model, in which each variable is associated with a plot where a crop rotation schedule is given by constraints (1)–(5).

Mathematical model

Let C be a set of crops with a positive demand and $k = 1,..., L$ be the available cropping areas. Due to possible heterogeneities in climate and soil conditions, the set of ordered crops and the associated productivity might differ from one cropping area to another. Taking this into account, for each cropping area k, we have a specific crop rotation schedule model (1)–(5), based on the set C_k of crops that can be planted in cropping area k together with crops for green manure available to area k (G_k).

In order to write a mathematical model for the crop rotation scheduling problems with known demands (CRS-D problem), consider the following parameters:

- d_{ij} the demand of crop i in period j, given in appropriate units (kg, pack etc.);
- A_k the size of cropping area k, $k = 1,..., L$;
- S_k the number of feasible crop schedules associated to cropping area k, i.e., the number of all possible solutions ($\hat{x}^s_{ijk}$) of (1)–(5), restrict to $i \in C_k \cup G_k$;
- r_{ks} income per m^2 when crop rotation schedule s is used in area k.

A crop rotation schedule ($\hat{x}^s_{ijk}$) defines which and when crops are planted in area k. Once a planting schedule is fixed, one can determine the crop productions. For this, the following additional parameters are defined:

- o_i the number of periods between planting and the first harvesting of crop i;
- p_{irk} the production (units according to the demand) per m^2 of the r^{th} harvesting of crop i in the area k, for $r = 1,..., t_i - o_i - 1$.

Therefore, $p_{irk}x^s_{ijk}$ is the outcome per m^2 of the r^{th} harvesting of crop i in area k at period $j + o_i + r$ due to planting according to the crop rotation schedule s. Similarly, $p_{irk}\hat{x}^s_{i,j-o_i-r,k}$ is the outcome per m^2 of the r^{th} harvesting of crop i in area k at period j. Despite the fact that we assume that o_i and t_i are independent of area k, one can easily extend the model for this case.

Defining variables:

- λ_{ks} size of the plot planted according to crop rotation schedule s in area k.

We can write the CRS–D problem as:

$$\max \sum_{k=1}^{L} \sum_{s=1}^{S_k} r_{ks} \lambda_{ks} \tag{10}$$

subject to

$$\sum_{k=1}^{L} \sum_{s=1}^{S_k} \sum_{r=0}^{t_i - o_i - 1} p_{irk} \hat{x}^s_{i,j-o_i-r,k} \lambda_{ks} \geq d_{ij}, \; i \in C, \; j = 1, ..., M, \tag{11}$$

$$\sum_{s=1}^{S_k} \lambda_{ks} \leq A_k, \; k = 1, ..., L, \tag{12}$$

$$\lambda_{ks} \geq 0, \; s = 1, ..., S_k, \; k = 1, ..., L. \tag{13}$$

where $j - o_i - r$ is replaced by $j - o_i - r + M$, whenever it is less or equal to zero.

The aim of model (10)–(13) is to maximize the return obtained by the crop rotation schedules for the L areas. Given c_{ij}, the unit price (per kg, pack etc) of crop i at period j, the income r_{ks} can be expressed as:

$$r_{ks} = \sum_{i \in C_k} \sum_{j=1}^{M} \sum_{r=1}^{t_i - o_i - 1} c_{ij} p_{irk} \hat{x}^s_{i,j-o_i-r,k} \tag{14}$$

Constraints (11) guarantee that the demands for the crops at all the periods are met while constraints (12) ensure that only the available lands are used.

In the section labeled *Solution Methodology*, we describe a procedure based on column generation to solve model (10)–(13).

Example 1

Consider a one-year crop rotation schedule divided into twelve one month-periods (M = 12) and 8 hypothetical crops, whose data are listed in Table 1. For each crop i, Table 1 presents the botanic family (F_i) to which it belongs, the set of possible planting periods (I_i), the production time (t_i), the number of periods before partial harvesting (o_i) and partial harvesting outcomes, p_{ir}. For example, crop 1 belongs to botanic family 1, it can be planted from month 1 to month 7, and it consumes 4 months from planting to final harvest. As o_1 = 1 and $t_1 - o_1 - 1 = 2$, crop 1 has 1 period after the planting period and before the first harvesting period, and 2 periods of harvests. Table 1 also gives the demands for crops 1–6 at each period. Crops 7 and 8 are for green manuring and have no demand.

Table 1. *Problem data: crop data and demands for* 8 *hypothetical crops.*

Crop	*Crop data*							*Demand per period*											
	F_i	I_i	t_i	o_i	p_{i1}	p_{i2}	p_{i3}	1	2	3	4	5	6	7	8	9	10	11	12
1	1	1–7	4	1	1	1				1	1								
2	1	1–9	3	1	1						1								
3	2	3–11	3	0	1	1		2			1	1			1	1			
4	2	2–11	4	0	1	2	1	4				8	6	8			2	8	10
5	3	7–12; 1–5	3	0	1	1				1	1								
6	3	3–10	3	0	2	1					2	2	2	1	1	2	2	2	2
7	4	2–12	3	1	1														
8	4	8–12; 1–6	2	0	1														

Consider 3 cropping areas with sizes $A_1 = 6$, $A_2 = 4$ and $A_3 = 5$ m^2 and assume that crop 1 cannot be planted in area 1, crops 2, 3 and 5 are not planted in area 2 while in area 3 one can not grow crops 2 and 4. In terms of our notation: $C_1 = \{2, ..., 8\}$, $C_2 = \{1, 4, 6, 7, 8\}$ and $C_3 = \{1, 3, 5, ..., 8\}$. Moreover, due to the different soil quality and climatic conditions, we assume that the productivity in areas 1, 2 and 3 are 110%, 100%, and 80% of the basis values presented in Table 1, respectively, in such a way that the productivities of crop 1 in area 1 are $p_{111} = 1.1$, $p_{121} = 1.1$, $p_{131} = 0$; in area 2, $p_{112} = 1$, $p_{122} = 1$, $p_{132} = 0$ and, in area 3, $p_{113} = 0.8$, $p_{123} = 0.8$ and $p_{133} = 0$. Of course, when dealing with real-word data, some pieces of land might be more productive for some crops and less productive for others.

Forcing one period of fallow (represented as crop 9) the solution presented in Fig. 3 is the one that maximizes the total non-weighted production ($c_{ij} = 1$, for all i and j). Fig. 3 uses the notation already introduced in Fig. 2 where the italic figures correspond to the partial harvests.

In the optimal solution, the sizes of cropping areas 1 and 2 have been divided into two plots with different crop rotation schedules while the size of area 3 has been divided into three plots. The values of variables are: $\lambda_{11} = 1$, $\lambda_{12} = 5$, $\lambda_{21} = 1$, $\lambda_{22} = 3$, $\lambda_{31} = 2.5$, $\lambda_{32} = \lambda_{33} = 1.25$. Note that additional fallow periods may arise, as in the second schedule (plot of 5 m^2) for area 1 in period 3.



Figure 3. Crop production schedules for the three cropping areas.

Model discussion and extensions

In this subsection, we discuss some practical issues for the CRS–D problem and how they could be integrated into the model.

Unmet demand and upper bounds on demands

The presence of ecological-based and technical constraints makes it harder to predict whether a feasible crop rotation schedule that supplies a given demand exists or not. In practice, however, one might be interested in solutions that minimize the unmet demand or obtain solutions presenting a trade-off between the gains associated with the harvests and the unmet demands. This multi-objective facet of the problem can be somehow incorporated into model (10)–(13) by changing the objective function and by using auxiliary variables θ_{ij}^- representing the unmet demand of crop i at period j. To limit the amount of the unmet demand, one must bound the value of these variables. Model (15)-(18) formalizes these concepts:

$$\max \sum_{k=1}^{L} \sum_{s=1}^{S_k} r_{ks} \lambda_{ks} - \sum_{i \in C} \sum_{j=1}^{M} q_{ij} \theta_{ij}^- \tag{15}$$

subject to

$$\sum_{k=1}^{L} \sum_{s=1}^{S_k} \sum_{r=0}^{t_i - o_i - 1} p_{irk} \hat{x}_{i,j-o_i-r,k}^{s} \lambda_{ks} \geq d_{ij} - \theta_{ij}^-, \; i \in C, \; j = 1, ..., M, \tag{16}$$

$$\sum_{s=1}^{S_k} \lambda_{ks} \leq A_k, \; k = 1, ..., L, \tag{17}$$

$$\lambda_{ks} \geq 0, \; \theta_{ij}^- \geq 0, \; s = 1, ..., S_k, \; k = 1, ..., L. \tag{18}$$

where θ_{ij}^- is the amount of unmet demand of crop i at period j, and q_{ij} is an associated penalty.

Besides allowing part of the demand to be unmet, it might be convenient to establish an upper bound on the exceeding production. The motivation for this would be to limit the production to levels that can be absorbed by the farmers' clients. This can easily be done by imposing upper bounds to the slack variables of constraints (16). In the computational results section, we analyze some computational experiments based on real-word data for vegetable crops with the model (15)–(18).

Considerations on the number of plots

Model (10)–(13) imposes no restriction on the number of plots that a cropping area might be divided into, allowing for solutions with a very large number of plots. In order to reduce the occurrence of these solutions, a fixed cost, say z_k, can be given for $\lambda_{ks} > 0$, i.e., whenever a new plot is considered. The original objective function is then replaced by:

$$\max \sum_{k=1}^{L} \sum_{s=1}^{S_k} (r_{ks}\lambda_{ks} - z_k\delta_{ks}) \tag{19}$$

and the following constraint is included:

$$\lambda_{ks} \leq A_k\delta_{ks},\ \delta_{ks} = \{0,1\}, s = 1,...,S_k,\ k = 1, ..., L \tag{20}$$

The fixed cost z_k is selected according to the desired trade-off between the conflictive goals of minimizing the number of plots in the solution and maximizing income. Formulation (19), subject to constraints (11)–(13) and (20) is no longer a linear programming model but a more complex mixed-integer problem. The reader is referred to Santos et al. (2010) for a discussion on this subject.

The crop rotation scheduling problem with stocks (CRS-S problem)

Vegetable crops are usually sent to packing houses after harvesting. In the packing houses, the vegetable crops are cleaned, packed and stored in cold chambers until delivery. Due to the high perishableness of many vegetable crops, however, they can only remain in stock for a limited amount of time and, even then, with losses. The possibility of stocking defines a more general and, therefore, more complex problem. We call this problem, which clearly is a general case of the CRS-D presented earlier, the crop rotation scheduling problem with is stocks (CRS-S problem). In this new situation, the demand is no longer only supplied by fresh-harvested crops.

Formulation (10)–(13) and its variants do not model the possibility of keeping some of the harvested crops in stock in order to meet future demands. In order to overcome this limitation, we define the following variables:

- I_{ij}^{w} stock of crop i available at period j which has been produced w periods previously, $w = 0,...,G_i$;
- η_{ij}^{w} quantity of crop i delivered at period j using stocks I_{ij}^{w}, $w = 0,...,G_i$;
- θ_{ij}^{+} quantity of crop i delivered at period j exceeding demand d_{ij};
- θ_{ij}^{-} unmet demand d_{ij}.

In both sets of variables I_{ij}^{w} and η_{ij}^{w}, index w indicates the age of the product. When $w = 0$, it indicates a product that has just been harvested. In each period, the available stocks (of different ages) can either be used to meet the current demand or be held in stock for one additional period. To model the fact that each product can only be stocked for a limited amount of time, it suffices to limit index w to appropriate values. For example, if a crop can be stocked for 3 periods, only the variables with indexes $w = 0,\ldots,4$ should exist, index 4 being associated with a spoiled product (final losses). In the general case, parameter G_i represents the number of periods a given crop i can be held in stock.

The total harvested quantities of crop i, at period j, is modeled by variables I_{ij}^0 (Production balance). This quantity can either be used to supply the demand of period j, by means of variables η_{ij}^0 or be stocked, in which case it will appear (discounted the losses) as variable $I_{i,j+1}^1$ in the next period. The same is valid for values of w other than zero (Stock balance). The quantities η_{ij}^w, for $w = 0..G_i$, are used to supply the current demand (Demand supply). Fig. 4 illustrates this general idea of the production balance.

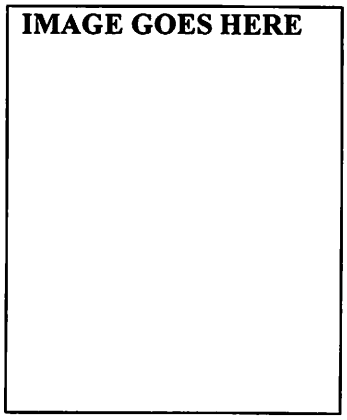


Figure 4. Relations among variables for crop i at period j.

Variables θ_{ij}^+ and θ_{ij}^- model the unmet demand or the quantity over demand, respectively. With the following additional notation:

- C set of crops, not including those associated with green manure neither fallow;
- S the number of feasible crop rotation schedules for the cropping area;
- A the size of the cropping area;
- c_{ij} unit income of selling ordered by demand d_{ij} of the crop i at period j;
- c_{ij}^+ unit income of selling non ordered by demand d_{ij} of the crop i at period j ;
- r_i fraction lost of the quantity already in stock of crop i, if it is held for one more period.

For the sake of presentation simplicity, we now consider a single cropping area. Taking this into account, index k in parameters p_{irk} and variables λ_{sk} are no longer needed. The generalization for the case of many heterogeneous cropping areas is straightforward.

A model for the CRS-S problem can be written as:

$$\max \sum_{i \in C} \sum_{j=1}^{M} c_{ij}(d_{ij} - \theta_{ij}^-) + c_{ij}^+ \theta_{ij}^+ \tag{21}$$

subject to

$$\sum_{s=1}^{S} \sum_{r=0}^{t_i - o_i - 1} p_{ir} \hat{x}_{i,j-o_i-r}^s \lambda_s = I_{ij}^0, i \in C, j = 1, ..., M, \tag{22}$$

$$\sum_{s=1}^{S} \lambda_s \leq A, \tag{23}$$

$$(1 - r_i)(I_{ij}^w - \eta_{ij}^w) = I_{i,j+1}^{w+1}, \; i \in C \; w = 0,...,G_i, j = 1, ..., M, \tag{24}$$

$$\sum_{w=0}^{G_i} \eta_{ij}^w = d_{ij} - \theta_{ij}^- + \theta_{ij}^+, \; i \in C, j = 1, ..., M, \tag{25}$$

$$\lambda_s \geq 0,\ s = 1..S, \tag{26}$$

$$\theta_{ij}^{+} \geq 0,\ \theta_{ij}^{-} \geq 0,\ \eta_{ij}^{w} \geq 0,\ I_{ij}^{w} \geq 0,\ I_{ij}^{G_i+1} \geq 0,\ i \in C, j = 1, ..., M,\ w = 0,...,G_i. \tag{27}$$

where, $j - o_i - r$ is replaced by $j - o_i - r + M$, whenever it is less or equal to zero.

The objective function (21) maximizes incomes, which can be obtained by selling products to meet the ordered demand or by selling the extra production (usually at a lower price). Constraints (22) define I_{ij}^{0} as the harvested quantities of each crop i in period j. Constraint (23) makes sure that only the available land is used.

Constraints (24) are the stock balance constraints and include a loss-factor for every additional period a product is kept in stock. These constraints can be read as follows: the quantity of stocked production of crop i in period j produced w periods previously (I_{ij}^{w}) can either be used to meet the demand of period j (η_{ij}^{w}) or remain in stock in the following period ($I_{i,j+1}^{w+1}$). For $w = G_i$, the variable $I_{ij}^{G_i+1}$ indicates losses (all remaining stock of crop i after G_i periods is considered spoiled). Constraints (25) control the demand supply: the demand of production of crop i at period j, d_{ij}, can be met from fresh harvested products (η_{ij}^{0}) or from stocked products with $w \leq G_i$ periods of age, ($\eta_{ij}^{w}, w \geq 1$).

In constraints (24) the fraction lost at the end of a period is independent of the period itself. This assumption can be easily generalized by considering the fraction of losses as a parameter dependent on the current period w, i.e., define r_{iw} as the fraction lost in product i, at the end of period w, if the product is kept one period longer in stock.

Example 2

In order to illustrate the impact of stocks, consider A = 200m^2 (total cropping area) and 5 hypothetical crops (crop 5 is used as green manure). Again we use a 1 year crop rotation scheduling, divided into 12 months and one fallow period (denoted by crop 6) per schedule. Table 2 presents the technical data and known demands for each one of the crops, including the number of periods in which crop i can be held in stock (G_i) and the percentage loss (r_i) incurring in crop i if it is stocked for one more period. For example, in the row associated with crop 1, we can read that it can be stocked for at most three periods ($G_1 = 3$) but for each period in stock, 20% of the production stocked will be lost ($r_1 = 0.2$).

Table 2. Problem data: crop data and demands for 5 hypothetical crops.

crop	Crop data									Demand per period											
	F_i	I_i	t_i	o_i	p_{i1}	p_{i2}	p_{i3}	r_i	G_i	1	2	3	4	5	6	7	8	9	10	11	12
1	1	11-12;1-7	4	1	1	1		0.2	3					10	10	10	10				
2	1	1-9	3	1	1			0.1	2					20	20	20	40	40	40		
3	2	4-12	3	0	1	1		0.2	1	10	10	20	20	20							
4	2	1-11	4	0	1	2	1	0.4	2	10				10	10	10				10	10
5	3	1-12	2	0	1																

We consider two cases: (a) no stock is allowed, and (b) stocking (under the conditions described in Table 2) is allowed. In both cases, we assumed $c_{ij} = 1$ and $c_{ij}^{+} = 0$, for all i and j.

In case (a), 8 plots are in the optimal solution with sizes $\lambda_1 = \lambda_2 = 10$, $\lambda_5 = \lambda_6 = \lambda_8 = 20$, $\lambda_3 = \lambda_4 = \lambda_7 = 40$m^2. The percentage of unmet demand is 1.9% and production over demand is 5.7%. In case (b), 5 plots are in the optimal solution, with sizes $\lambda_1 = 10$, $\lambda_2 = 35$, $\lambda_3 = \lambda_5 = 40$ and $\lambda_4 = 46.9$ m^2, and corresponding crop rotation schedules are depicted in Fig.5. The percentage of unmet demand is 0.95% and the extra production is 6.1%.

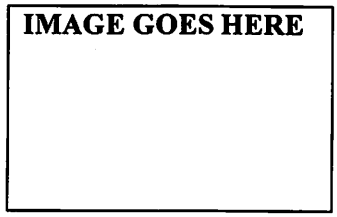


Figure 5. Optimal crop rotation schedules for case (b):stock allowed.

The fewer amount of unmet demands and the increase in extra production shows the greater flexibility introduced by the presence of stocks. Once product holdings are allowed, part of the production in some periods is stocked to meet the demand of the following periods. An illustrative example is the case of crop 1 in the solution shown in Fig. 5. In fact, crop 1 is grown in plot 2 (of 35 m^2) in period 3, and its harvestings are forecasted to be 35 kg in period 5 and 35 kg in period 6. The demand of crop 1, as shown in Table 2, is of 10 kg for each one of periods 5 to 8.

To meet demands, one could proceed as follows: from the 35 kg harvested in period 5, 10 kg are immediately used to supply the demand of the period, 21.1 kg are sold as extra demand and 3.90 kg are kept in stock. Due to losses, only 80% of this amount is available at period 6 (3.125 kg), which associated with the harvest of the crop in the period, gives a total availability of 38.125 kg (35 kg with age zero and 3.125 kg one-period old). From this amount, 10 kg are used to supply the demand of the period (priority is given to the older product) and 28.125 kg remain in stock yielding an availability of 22.5 kg of one-period old product at period 7. The demand of 10 kg is therefore supplied and 12.5 kg of product is held in stock and appears as 10 kg of a two-period old product that is used to supply the demand of period 8.

The crop rotation scheduling problem with adjacency constraints (CRS-A problem)

Consider a cropping area divided beforehand in plots. Since many herbivores and pathogens attack crops of the same botanic family, the first ecological criterion (a) listed in the Introduction Section can be seen in a spatial manner and one might therefore be interested in avoiding the simultaneous growth of crops belonging to the same botanic family in adjacent plots. This gives rise to the following constraint:

(iv) *Spatial neighborhood:* crops belonging to the same botanic family are not cultivated simultaneously in adjacent plots.

In CRS–D and CRS–S problems, adjacencies among plots are not taken into account, since they are not known *a priori* but rather defined during the optimization process. For these problems, decisions are made so that the yields meet the expected demand. In the crop rotation scheduling problem with adjacency constraints (CRS-A problem), the goal is to maximize land occupation while respecting ecological criteria (i)–(iv). In this case, it is necessary to know the neighborhood of each plot in the cropping area.

A cropping area divided into L plots can be represented by a planar graph G(V, A), in which a node in V corresponds to a plot and an edge (u, v) belongs to A if and only if plots u and v are adjacent. Fig. 6(a) depicts a cropping area divided into 5 plots and Fig. 6(b) shows the associated graph G(V, A).



Figure 6.(a): Cropping area divided into five plots, and (b): the associated planar graph G(V, A).

The mathematical model

In order to present a mathematical model for the CRS–A problem, consider in addition to the parameters previously defined for the crop rotation schedule for one plot, the connected planar graph G(V, A) associated to the cropping area and the following decision variables:

x_{ijk} = 1, if crop i is planted at period j in plot k, 0 otherwise, $i = 1, ...,N+1$, $j \in I_i$, $k = 1, ..., L$.

For each plot k, the decisions of planting crop i in period j are restricted to constraints (i)-(iii), that is, the decisions are restricted to the crop rotation scheduling problem (1)–(5). Moreover, the crop rotation schedules for adjacent plots need in addition to ensure ecological constraint (iv) for adjacent plots.

A model for the CRS–A problem can be thus written as:

$$\max \sum_{k=1}^{L} \sum_{i \in C} \sum_{j \in I_i} t_i x_{ijk} \tag{28}$$

subject to

$$\sum_{i=1}^{N+1} \sum_{r=0}^{t_i-1} x_{i,j-r,k} \le 1,\ j = 1,...,M, k = 1,...,L \tag{29}$$

$$\sum_{i \in F(p)} \sum_{r=0}^{t_i} x_{i,j-r,k} \le 1,\ p = 1..NF, j = 1,...,M, k = 1,...,L \tag{30}$$

$$\sum_{i \in G} \sum_{j \in I_i} x_{ijk} = 1, k = 1,...,L \tag{31}$$

$$\sum_{j=1}^{M} x_{njk} = 1, k = 1,...,L \tag{32}$$

$$\sum_{i \in F(p)} \sum_{r=0}^{t_i-1} (x_{i,j-r,u} + x_{i,j-r,v}) \le 1,\ p = 1,...,NF, j = 1,...,M, (u,v) \in A \tag{33}$$

$$x_{ijk} \in \{0,1\},\ i = 1, ..., N+1, j \in I_i,\ k = 1,...,L. \tag{34}$$

where, as usual, in (29), (30) and (33), $j - r$ is replaced by $j - r + M$, whenever it is less or equal to zero.

The objective function (28) gives the total occupation of the plots in each crop rotation schedule. The variables related to green manuring crops and fallow periods were excluded from the objective function. Constraints (29)–(32) are analogous to constraints (1)–(5), which define a feasible crop rotation schedule, now for a plot k known a priori: constraints (29) ensure that two crops cannot be grown simultaneously, constraints (30) prevent crops from the same botanic family being planted in sequence, and constraints (31)-(32) force the planting of a green manure crop and the presence of one fallow in each crop rotation. . Constraints (33) prohibit crops of the same family being grown at the same time in adjacent plots.

Example 3

To illustrate the CRS-A problem, we consider a cropping area divided into 5 plots with a spatial configuration as illustrated in Fig. 6. Each rotation is two years long divided into ten days (M = 72 periods). In this case, we used real-world data for the set of 28 crops from 10 botanical families obtained in a non-commercial organic vegetable garden in Viçosa - Brazil. The data for the 28 crops can be found in Table A.1 in the Appendix. A fallow period of 3 periods (1 month), represented as crop 29, was established for each crop rotation schedule. Fig. 7 shows an optimal crop rotation schedule for the 5 plots over 24 months. In the scheduling for each plot, the bold number represents the crop number (for example, crop 1 is lettuce) and indicates its planting date.

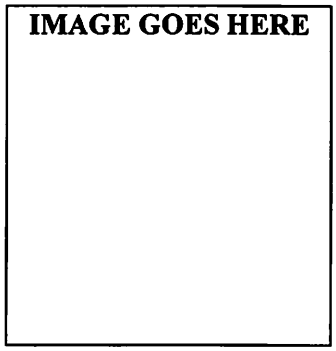


Figure 7. Optimal crop rotation schedule for 5 plots.

It can be seen in Fig. 7 that constraints (i)–(iv) are fulfilled: for each plot, the corresponding crop rotation schedule associated with the core problem (1)–(5) is respected. Moreover, no crops of the same botanic family are simultaneously cultivated in adjacent plots.

Cropping area with special spatial configuration

For the CRS–A problem, there are some characteristics of the spatial configuration of the cropping area that can be explored in order to obtain feasible or optimal solutions more easily. Consider G(V, A) the connected planar graph of the cropping area divided into L plots. Suppose V is partitioned into n independent subsets (i.e., there is no edge linking vertices in an independent subset). For example, set V of graph G given in Fig. 6(b) can be partitioned into three independent sets: {1}, {2, 4} and {3, 5}.

If a feasible solution for the CRS–A problem exists for a complete graph K_n (a graph with n vertices and one edge for all pair of nodes), then a feasible solution for G can be constructed from this solution as follows. As any two vertices of an independent subset of V are not adjacent to each other, the same schedule can be assigned to all plots of this subset without violating adjacency constraints (33). We obtain a feasible solution to graph G, assigning a schedule of one of the vertices of K_n to all the plots of each independent subset of V so that no schedule is attributed to more than one subset.

For instance, consider any feasible crop rotation scheduling to graph K_3 and graph G in Fig. 6(b). We can assign the schedule of plot 1 in K_3 to plot 1 in G; the schedule of plot 2 in K_3 to plot 2 and 4 of G; and finally, the schedule of plot 3 in K_3 to plot 3 and 5 of G. However, even though we have the optimal scheduling for K_3, this solution may not be the optimal to G (Santos, 2009). Moreover, the non-existence of a solution for K_3 does not imply that there is no solution for graph G. Nevertheless, in the next section, we present some situations where this strategy produces an optimal solution to the original problem.

Cropping areas divided into parallel plots or checkerboard patterns

A simple and common situation found in practice is where the cropping area is divided into parallel plots or checkerboard patterns (see Fig. 8(a)). In these cases, the plots can be divided into just two independent subsets. Santos et al. (2008) prove it is then possible to construct an optimal solution from the solution for only two adjacent plots. Consider, for example, a cropping area with a checkerboard pattern, as shown in Fig. 8, and an optimal solution for two adjacent plots (that is, for graph K_2). An optimal crop rotation scheduling for the 9 plots is obtained by assigning one of the schedules of K_2 to plots 1, 3, 5, 7 and 9 (colored in Fig. 8(b)), and the other schedule of K_2 to the plots 2, 4, 6 and 8.

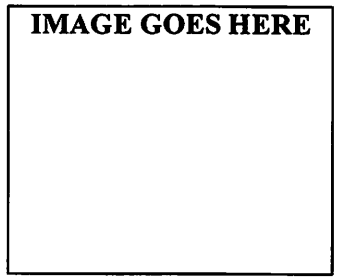


Figure 8. The cropping area divided into L = 9 plots

Model Reformulation

The mathematical formulation (28)–(34) for the CRS-A problem has, in practice, a very large number of constraints and variables which limit the use of exact solution methods. However, the special structure found in (29)–(33) can be exploited. Constraints (29) to (32) are associated with independent plots and are only linked by constraints (33), suggesting the use of decomposition methods.

The problem of finding a crop rotation schedule for L plots can also be seen as one of choosing, among all feasible schedules for a plot, L schedules satisfying plot adjacency constraints (33). Using this decomposition, Santos et al. (2008) reformulated the model (28)–(34) as follows, and devised exact and heuristic methods to solve it.

Consider all feasible crop rotation schedules for one plot, $(\hat{x}_{ij}^{s})$, $s = 1, ..., S$, defined by constraints (1)–(5), and let the binary decision variables be:

$\lambda_{sk} = 1$, if, plot k is planted according to s^{th} feasible crop rotation, 0 otherwise, for $s = 1, ..., S$.

Consequently, any feasible crop rotation schedule to plot k, (x_{ijk}), can be written as:

$$x_{ijk} = \sum_{s=1}^{S} \hat{x}_{ij}^{s} \lambda_{sk}, \ \sum_{s=1}^{S} \lambda_{sk} = 1, \ \lambda_{sk} \in \{0,1\} \tag{35}$$

Therefore, important combinatorial aspects of the CRS-A problem (to obtain crop rotation schedules for a plot) are supposed to be already solved and the decision now consists of which schedule should be used for each plot, taking into account adjacency constraints among the plots. Therefore, model (28)-(34) is rewritten by substituting (35) by (28)-(33), which leads to its reformulation called the *master problem*, given by:

$$\max \sum_{k=1}^{L} \sum_{i \in C} \sum_{j \in I_i} \sum_{s=1}^{S} t_i \hat{x}_{ij}^{s} \lambda_{sk} \tag{36}$$

Subject to:

$$\sum_{i \in F(p)} \sum_{\substack{r=0 \\ j-r \in I_i}}^{t_i-1} \sum_{s=1}^{S} (\hat{x}_{i,j-r}^{s} \lambda_{su} + \hat{x}_{i,j-r}^{s} \lambda_{sv}) \leq 1, \ p = 1,...,NF, \ j = 1,...,M, \ (u,v) \in A \tag{37}$$

$$\sum_{s=1}^{S} \lambda_{sk} = 1, \ k = 1,...,L \tag{38}$$

$$\lambda_{sk} \in \{0,1\}, \ s = 1, ..., S, \ k = 1, ..., L. \tag{39}$$

As a final comment to this section, we observe that uncertainty can be of importance when dealing, for example, with the yields of the crops and selling prices. For a discussion on this issue and an extension of model (21)-(27) to incorporate uncertainty, the reader is refereed to Costa et al. (2009).

SOLUTION METHODOLOGY

In the previous sections, we have described and presented mathematical models for three crop rotation problems and extensions focusing on vegetable crop production. Now, we discuss some methodological approaches, both exact and heuristic, that have been used to solve them. First, we present a brief review of some of the available methodologies, which is followed by a discussion of the specific strategies that have been proposed for each particular case.

The problems discussed have been modeled as linear programs (CRS–D and CRS–S problems) or as linear programs with integer variables (CRS-A problem). This enables the use of a vast arsenal of methods that have been proposed over the years since the development of the simplex algorithm by Dantzig (1947

[Nemhauser and Wolsey, 1988)]). The simplex algorithm exploits the property that in a linear program, an optimal solution, if it exists, can always be found in a vertex of the polyhedron defined by the constraints of the problem. The method works by moving between neighbor solutions (vertices) until the one with the best objective function value has been found. Other efficient strategies do exist for solving linear programs. In particular, interior point methods (Vanderbei, 2008) have received a great deal of attention in the literature. Nevertheless, the simplex algorithm is still the most popular approach in part because of its good performance in practice, in spite of the fact that it has been shown for some instances it visits 2^n vertices before finding the optimal solution, where n is the number of the problem variables (Vanderbei, 2008).

One of the advantages of the simplex algorithm is its ability to deal with some of the variables in an implicit manner. Indeed, one can start the method with only a few variables (columns) and iteratively include promising variables generated by an auxiliary problem. This methodology, known as column generation, was firstly proposed in the 60's and it has been successfully applied since then in many different optimization linear problems (Lübbecke and Desrosiers, 2005). Procedures based on column generation were used to solve the three problems presented in this chapter. In all of them, columns are generated by the auxiliary problem (also called subproblem or pricing problem), which is the crop rotation scheduling problem (1)-(5).

In the case where some variables need to be integers, the simplex algorithm can only supply a bound on the value of the objective function. This bound is associated with the solution obtained when the integrality constraints are relaxed. There are many techniques to round the solution obtained by the simplex algorithm to its closest integer, but there is usually no guarantee that a good (or even feasible) integer solution will be obtained. For this reason, more elaborate methods have been proposed, two of which emerged almost simultaneously in the sixties: the branch and bound algorithm (Land and Doig, 1960) and the cutting planes method (Gomory, 1958). Both of these methods rely on a number of calls to the simplex algorithm but with different strategies: while the branch and bound approaches do an implicit tree enumeration of all solutions (using the bound given by the simplex algorithm to cut non-promising branches), the cutting plane methods alternate between the solution of the simplex algorithm and the generation (and subsequent addition to the problem) of a valid inequality which cuts the current simplex optimal solution without eliminating any feasible integer solution, until the solution given by the simplex is, itself, integer. Nowadays, probably one of the most used strategies is a combination of these two methods, called branch-and-cut, which implicitly enumerates solutions as in the branch and bound, but also uses the idea of the valid inequalities to improve the value of the bound obtained at each call of the simplex algorithm (Johnson et al., 2000). An extension of the branch-and-bound algorithm, which is able to cope with a large number of variables in an implicit manner (as was done by the column generation algorithm in the case of pure linear program), also exists and is known in the literature as the branch-and-price (Barnhart et al., 1998). In branch-and-cut-price algorithms, cutting planes are added to accelerate the branch-and-price.

Commercial mixed-integer LP solver packages, such as CPLEX and XPRESS, and non-commercial, such as lp_solve and SIMPHONY, usually rely on sophisticated versions of the branch-and-cut algorithms, both because of its efficiency and because of its generality (Linderoth and Ralphs, 2005; Atamtürk and Savelsbergh, 2005). In spite of the ability of the algorithms implemented in these packages to obtain solutions to many practical problems, they can fail to converge or even to obtain a single feasible solution to many others. The alternative is then to rely on heuristic methodologies that do not ensure optimality, but that can be rather efficient in obtaining – usually quickly – good quality solutions. These methods can vary from very intuitive strategies based on a sequential (and often myopic) construction of the solution, to local descent methodologies which try to find the best solution in a given neighborhood of the search space, to the so-called metaheuristics which are approaches that rely on different mechanisms to try to avoid premature convergence to low quality local optima. Examples of metaheuristics are abundant in the literature. Among the most used we find tabu search, grasp, simulated annealing and genetic algorithms, to mention a few (Blum and Roli, 2003).

A more recent trend that can be found in the literature is the combination of exact and heuristic algorithms yielding the so-called hybrid approaches. Numerous strategies have been developed and can be roughly classified into two groups. In the first group, exact and heuristic algorithms are applied sequentially, each one of them finding part of the solution, often in an iterative process. In a second group of algorithms, there is a more hierarchical structure with the exact approach making successive calls to the heuristic method or vice-versa (Raidl and Puchinger, 2008).

The key technique behind the solution methods for the three crop rotation optimization problems presented earlier is the column generation approach. The column-generation methodology is devised to solve a master problem containing a large number of columns, which cannot be explicitly determined within reasonable computational times. These columns are obtained from solutions of a core model, which is, for

the three presented optimization problems, the crop rotation scheduling model (1)–(5). In the context of this chapter, this model is solved by a conventional branch-and-bound algorithm. The column generation technique works iteratively with a sequence of restricted master programs, which contain only a small subset of variables (or columns) of the master problem. At each iteration of the algorithm, a new restricted master problem is obtained by adding new columns in order to improve the current solution. When no improving column can be found, the optimal solution of the current restricted master program is also an optimal solution of the original master problem (Lübbecke and Desrosiers, 2005).

We briefly discuss in the following section how the column generation technique can be applied to each one of the presented problems. For the reader who is interested in a more thorough presentation, please refer to Santos et al (2010), Costa et al. (2009) and Santos (2009). For more details on the column generation technique, see Desaulniers et al. (2005).

Solving the CRS–D problem by the column generation technique

In model (10)–(13) for the CRS–D problem, columns are built from crop rotation schedules for the L cropping areas. Let $(\hat{x}^s_{ijk})$ be the s^{th} feasible crop rotation schedule for cropping area k. The coefficient of the variable λ_{ks} in constraint (11) for a given i and j is:

$$a^s_{ijk} = \sum_{r=0}^{t_i-o_i-1} p_{irk}\hat{x}^s_{i,j-o_i-r,k}$$

indicating the production of the r^{th} harvesting of crop i at period j per m^2 cultivated according to the s^{th} crop rotation schedule for area k.

Therefore, column $\mathbf{a}_{sk}$ in the CRS-D model can be written as:

$$\mathbf{a}_{sk} = \begin{bmatrix} (a^s_{ijk}) \\ \mathbf{e}_k \end{bmatrix} \begin{matrix} \rightarrow \text{constraints (11)} \\ \rightarrow \text{constraints (12)} \end{matrix}$$

where $\mathbf{e}_k$ is an L–dimensional vector with $e_{kk} = 1$ and $e_{kj} = 0$ for $j \neq k$.

By associating dual variables π_{ij} to constraints (11) and dual variables α_k to constraints (12) the column that gives the largest reduced cost for area k can be obtained by solving the following pricing problem P_k:

$$\max \sum_{i \in C_k} \sum_{j=1}^{M} \sum_{r=1}^{t_i-o_i-1} (c_{ij} - \pi_{ij}) p_{irk} x_{i,j-o_i-r,k} - \alpha_k$$

subject to that (x_{ijk}) satisfies (1)–(5) where only crops in $C_k \cup G_k$ are considered.

Based on the above we present a column generation algorithm for the CRS-D problem:

Algorithm 1: column generation for the CRS-D problem

1. Choose an initial set of columns for the restricted master problem.
2. Solve the restricted master problem and obtain (π, α), the dual variables associated to constraints (11) and (12), respectively.
3. Solve the sub problem P_k, for $k = 1, ..., L$, and obtain the reduced costs $\grave{a}_{ks}$ and column $\mathbf{a}_{sk}$.
4. If $\grave{a}_{ks} = 0$, for $k = 1, ..., L$, then STOP. The current solution is optimal. Otherwise, add column $\mathbf{a}_{sk}$ with relative costs $\grave{a}_{ks} > 0$ in the restricted master problem and go to step 2.

In order to solve model (15)–(18) for the case where unmet demands are allowed, other types of columns associated to variables θ_{ij}^{-} are considered. In this case, the initial columns for the restricted master problem can be assumed to be those associated with these variables. If no initial set of columns is given for problem (10)–(13), these variables can serve as artificial variables, where q_{ij} is a large penalty.

Solving the CRS–S problem by the column generation technique

In model (21)–(27) for the CRS-S problem, we consider just one cropping area, allowing index k to be omitted. Let $(\hat{x}_{ij}^{s})$ be the s^{th} feasible crop rotation schedule. Then the coefficient of the variable λ_s in constraint (22) for given crop i at period j is $a_{ij}^{s} = \sum_{r=0}^{t_i-o_i-1} p_{ir}\hat{x}_{i,j-o_i-r}^{s}$ and column $\mathbf{a}_s$ in the CRS-S model can be written as:

$$\mathbf{a}^{s} = \begin{bmatrix} (a_{ij}^{s}) \\ 1 \\ \mathbf{0} \end{bmatrix} \begin{matrix} \rightarrow \text{ constraints (22)} \\ \rightarrow \text{ constraint (23)} \\ \rightarrow \text{constraints (24)-(25)} \end{matrix}$$

This model also contains a number of additional columns associated to the different existing sets of variables: I_{ij}^{w}, η_{ij}^{w}, θ_{ij}^{+}, θ_{ij}^{-}. Each one of these columns has its own particular form. For instance, the column associated to variable I_{ij}^{0} is given by:

$$\begin{bmatrix} -e_{ij} \\ 0 \\ (1-r_i)e_{ij}^{w} \\ \mathbf{0} \end{bmatrix} \begin{matrix} \rightarrow \text{ constraint (22)} \\ \rightarrow \text{ constraint (23)} \\ \rightarrow \text{constraint (24)} \\ \rightarrow \text{constraints (25)} \end{matrix}$$

where e_{ij} and e_{ij}^{w} are appropriate columns of identity matrices.

In order to solve model (21)–(27), the initial restricted master problem can be formed with the columns associated to variables I_{ij}^{w}, η_{ij}^{w}, θ_{ij}^{+} and θ_{ij}^{-} and then, promising columns $\mathbf{a}^{s}$ are iteratively added into the restricted master problem until optimality is achieved. By associating dual variables π_{ij} with constraints (22) and α with constraint (23), the column with the largest reduced cost can be obtained by solving the following pricing problem P:

$$\max \sum_{i \in C} \sum_{j=1}^{M} \sum_{r=1}^{t_i-o_i-1} -\pi_{ij} p_{ir} x_{i,j-o_i-r} - \alpha$$

subject to that (x_{ij}) satisfies (1)–(5).

Algorithm 2 summarizes this column generation procedure:

Algorithm 2: column generation for the CRS-S problem

1. Consider the initial restricted master problem (21)–(27) consisting of columns associated to variables I_{ij}^{w}, η_{ij}^{w}, θ_{ij}^{+} and θ_{ij}^{-}.
2. Solve the restricted master problem and obtain the dual variables (π, α) associated to constraints (22)–(23).

3. Solve the pricing problem (P) and obtain the reduced cost $\grave{a}_s$ and column $\mathbf{a}^s$.
4. If $\grave{a}_s = 0$, then STOP. The current solution is optimal. Otherwise, add column $\mathbf{a}^s$ to the restricted master problem (21)–(27) and go to step 2.

Solving the CRS-A problem

Earlier in this chapter, we present a 0-1 linear optimization model (28)–(34) for the CRS-A problem. Traditional branch-and-bound techniques are only able to solve small instances of this model and, therefore, the development of new strategies is in order. With the goal of reducing the problem's dimension, Santos (2009) proposed a decomposition strategy, in which several sub problems, each one associated with one plot, are solved in sequence, in such a way that adjacency constraints (33) are fulfilled. Analogously to what was done for the CRS-D and CRS-S problems, the sub problems are solved by branch-and-bound algorithms. In this section, we present three algorithms that use this strategy (Santos, 2009). The first one is a constructive heuristic (*plot-to-plot*) which can be used to quickly provide feasible solutions for the other two algorithms which are based on the column generation technique.

A *plot-to-plot* constructive heuristic

A simple decomposition strategy can be devised in order to solve the CRS-A problem by first determining a feasible schedule for one plot and then finding a feasible schedule for a neighboring plot while forbidding the growth of crops which violate adjacency constraints (33). This is done by setting appropriate variables at zero which would violate the adjacency constraints. The strategy is repeated iteratively until a schedule has been assigned to all plots. Algorithm 3 formalizes the *plot-to-plot* constructive heuristic.

Algorithm 3: The *plot-to-plot* constructive heuristic

1. Find feasible crop rotation schedules for the L plots by considering only green manure crops and fallow mandatory periods. If there are not L feasible schedules, then STOP. The CRS–A problem has no solution. Otherwise, set $k = 0$ and sort the L plots.
2. If $k = L$, then STOP. An overall crop rotation schedule for the L plots has been obtained. Otherwise, for plot $k + 1$, with some variables already fixed in step 1 for green manure and fallow period, and taking into account the crops already fixed to plots 1,..., k, set to zero appropriate variables in order to ensure:
   (a) constraints (29) – at most one crop is cultivated in each period, in the same plot,
   (b) constraints (30) – crops of the same botanic family are not cultivated one immediately after the other in the same plot, and
   (c) constraints (33) – crops of the same botanic family are not cultivated simultaneously in adjacent plots.
3. Find a feasible (or optimal) crop rotation schedule for plot $k + 1$. Set $k = k + 1$ and go to step 2.

The performance of this heuristic strongly depends on the sequence in which the plots are treated. In our implementations, it was found to be useful to travel from the plot with the greatest number of neighbors to the one with the least number of them.

A column generation procedure with *plot-to-plot* heuristic (CGH)

Many exact and heuristic methods based on the column generation technique can be devised to solve the master problem (36)-(39). In the column generation procedure CGH, besides the columns with the largest reduced costs, other columns are also added to the restricted master problem in order to find a feasible solution to the CRS-A problem (Santos et al.; 2008; Santos, 2009). In model (36)–(39), each column is associated with the periods of growth of the crops determined by a crop rotation schedule.

Given the s^{th} schedule ($\hat{x}^s_{ijk}$) for plot k, let q^s_{pjk} be the parameter which has value 1, if a crop i belonging to botanic family p is growing in period j in cropping area k, and has value 0, otherwise. Therefore:

$$q^s_{pjk} = \sum_{i \in F(p)} \sum_{r=0}^{t_i - 1} \hat{x}^s_{i,j-r,k} .$$

Consider now the vector q_{uk} where its components are (q^s_{ijk}), if $(u, k) \in$ A, and $q_{uk} = 0$, otherwise. The column $\mathbf{q}^s_k$ associated to variable λ_{ks} in model (36) – (39) can be defined as:

$$\mathbf{q}^s_k = \begin{bmatrix} (q_{uk}) \\ \mathbf{e}_k \end{bmatrix} \begin{matrix} \rightarrow \text{ constraints (37)} \\ \rightarrow \text{ constraints (38)} \end{matrix}$$

where vector $\mathbf{e}_k$ is an L–dimensional vector with $e_{kk} = 1$ and $e_{kj} = 0$ for $j \neq k$.

Model (36)-(39) is an integer optimization problem. In order to solve the linear relaxation of this problem by column generation methodology, consider the dual variables π^{ku}_{pj} referring to constraints (37) for adjacent plots k and u, and dual variables α_k referring to constraints (38) for plot k. The column $\mathbf{q}^s_k$ associated with plot k that has the largest reduced cost is determined by solving the following pricing problem P_k:

$$\max \sum_{i=1}^{N+1} \sum_{j=1}^{M} t_i x_{ijk} - \sum_{j=1}^{M} \sum_{p=1}^{NF} \sum_{i \in F(p)} \sum_{r=0}^{t_i - 1} \sum_{(u,k) \in A} \pi^{ku}_{p,j+r} x_{ijk} - \alpha_k$$

subject to that (x_{ijk}) satisfies (1)–(5).

The *plot-to-plot* heuristic is used in order to insert new columns during the iterations of the CGH procedure. All columns obtained in this way with positive reduced costs are added to the linear relaxation of the master problem, otherwise the above pricing problem is solved. The aim of this strategy is to find feasible solutions for the CRS-A problem while solving the linear relaxation of the master problem (36)–(39). The columns obtained by a conventional column generation procedure might not be enough to provide a feasible integer solution for the problem, because the pricing problems that generate them do not take explicitly into account the adjacency constraints.

Algorithm 4 briefly presents this procedure.

Algorithm 4: A column generation procedure with the plot-to-plot heuristic (CGH)

1. Sort the L plots and apply the *plot-to-plot* heuristic in order to find an initial set of columns for the first restricted master problem (RMP). Keep the solution obtained by the *plot-to-plot* heuristic as incumbent.
2. Solve the RMP and obtain the dual variables π^{ku}_{pj} associated with constraints (37), and the dual variable α_k associated with the convexity constraints (38).
3. Set $col = 1$. Solve the pricing problem P_1 and obtain $\hat{x}_1 = (\hat{x}_{ij1})$ and the reduced cost $\hat{c}_1$.
   If $\hat{c}_1 > 0$ then add the column associated to $\hat{x}_1$ in RPM and set $col = 2$.
4. For $k = 2, ..., L$, do:
   a) If $col = k$, then
      i. solve P_k but fixing to zero appropriate variables that would violate adjacency constraints (33), taking into account the crop rotation scheduling $\hat{x}_1, ..., \hat{x}_k$ already determined to plots 1, ..., $k - 1$ (strategy analogous to the *plot-to-plot* heuristic).

ii. If P_k (fixing variables) is feasible then $col = col + 1$.
Else, solve P_k without fixing variables.

b) Let and $\hat{c}_k$ be the obtained solution from P_k and its reduced cost. If $\hat{c}_k > 0$ then add the column associated to $\hat{x}_k$ into RMP and go to step 5.

5. If $col = L$, a feasible solution $(\hat{x}_1, ..., \hat{x}_L)$ of CRS–A problem is obtained. If this solution is better than the incumbent one, keep the new solution as incumbent.
6. If $col = 0$ (i.e., no column with relative cost is found) then STOP. The linear relaxation of the master problem solved. Return the incumbent solution. Otherwise, go to step 2.

Adding new columns by the *plot-to-plot* heuristic allows finding feasible solutions to the CRS–A (lower bounds). This strategy also accelerates the convergence in solving the linear relaxation of the master problem (upper bound). For a computational study of the impact of adding these new columns see Santos (2009).

Branch-and-price with CGH procedure

The CGH procedure always finds a feasible solution to the CRS–A problem. In the worst case, it provides the solution obtained by the *plot–to–plot* heuristic in step 1. However, there is no guarantee that the obtained solution is optimal. If proven optimality is needed then a branch–and–price algorithm can be devised to solve the (integer) master problem.

In both branch-and-price and branch-and-cut-price algorithms the linear programs which arise in the nodes of the tree of enumeration are solved by the column generation technique.

The adopted branching rules and the proper use of valid inequalities are crucial issues for successful implementation of branch-and-price algorithms (Barnhart et al., 1998). When the master problem originates from a reformulation of an original 0-1 linear program (such as the master problem (36)–(39) which is a reformulation of problem (28)–(34)), the branching constraints can be built on the original variables and by fixing these variables (to zero or one) in the pricing problems. Santos (2009) solves model (36)–(39) by a branch-and-price algorithm using branching in the original variables and the CGH procedure to solve the linear relaxation of the master problem in each node of the enumeration tree. The performance of the branch-and-price algorithm with the CGH procedure is much better than the performance obtained by the column generation procedure without the *plot-to-plot* heuristic. This can be explained by the fact that the CGH procedure produces an upper bound more quickly in addition to the fact that it can obtain good quality lower bounds (Santos, 2009).

In the following section, we present some computational results in order to assess the efficiency of the proposed methods.

COMPUTATIONAL RESULTS

In Solution Methodology Section, we presented some procedures based on the column generation technique to solve the CRS–D, CRS–S and CRS–A problems. In this section, we briefly discuss computational experiments run in order to analyze the performance of algorithms (For more details, see Costa et al., 2009 Santos, 2009 and Santos et al., 2010). The algorithms have been implemented in C++ and the linear (integer) models solved by the commercial package CPLEX 11.1 (ILOG, 2006). The experiments were carried out on a computer with 2.33 GHz and 4 Gb of RAM, under Linux. For all the problems, we used 2-year crop rotation lengths, and 1-month fallow in the scheduling of each crop rotation.

For the CRS–D and CRS–S problems, the 2-year rotation length was divided into weeks. Data in Appendix B comes from an ecologically based production unit situated in the city of Barbacena, Brazil (21° 13' 30" S and 43° 46' 25" W, 1165 meters above sea level). The demands for the crops were randomly generated taking into account planting and harvesting periods of each crop.

For the CRS–A problem, we consider 10-day periods. Data in Appendix A was obtained in an organic vegetable garden of the Federal University of Viçosa, Brazil (20° 45' 14" S and 42° 52' 53" W, 648 meters above sea level).

Computational experiments for the CRS-D problem

In this section, we briefly present and discuss a number of computational experiments run in order to analyze the performance of the column-generation algorithm devised for the model (15)–(18) for the CRS–D problem. Different parameters were varied such as the number of crops (N = 12, 16, 20, 24), the number of cropping areas (L = 1, 3, 5) and their sizes (the total area was kept constant at 10,000 m^2), We consider the case without upper bounds on extra production ($d\infty$) and the case where there is a limit on the extra production equal to 200% of the given demand ($d200$). In order to mimic the heterogeneity of lands found in practice, we defined, for each cropping area, a set of crops that could be grown and set their productivities by making random changes to the original data (with a maximum variation of 30% regarding the data in Table B.2). The combination of the different characteristics presented above gives rise to 24 different instance classes.

Table 3 presents a summary of the obtained results. For each class of instances, the table provides the algorithm's running times in seconds (*time*), the number of plots in the optimal solution (*# plots*), and the percentage of residual unmet demand (*%resid*). Equal costs were used to penalize the unmet demands of the different crops. The algorithm was allowed to run for 1 hour for each instance.

Table 3: Average results for the 24 classes of instances.

		$L=1$			$L=3$			$L=5$		
crop	***demand***	***time***	***# plots***	***%resid***	***time***	***# plots***	***%resid***	***time***	***# plots***	***%resid***
12	*d200*	250.20	224.30	0.73	257.37	266.10	0.51	129.83	249.60	0.68
	d∞	757.25	119.80	0.00	646.62	116.89	0.00	708.97	135.60	0.00
16	*d200*	910.35	395.40	0.47	811.76	335.40	0.63	766.70	394.50	0.48
	d∞	2331.47	237.00	0.00	1283.94	244.50	0.00	2722.03	241.50	0.00
20	*d200*	3283.76	494.70	0.50	2630.80	497.90	0.49	2062.26	502.40	0.67
	d∞	3075.58	426.30	0.00	3430.26	457.56	0.00	2947.01	424.80	0.00
24	*d200*	3230.28	549.40	0.60	3230.28	549.40	0.60	3168.13	546.67	0.60
	d∞	3601.36	528.22	0.00	3596.96	594.50	0.00	3612.39	635.89	0.00

As shown in Table 3, the algorithm was able to obtain solutions within the allowed time limit for most of the instances. The average running times reached 1h only for the class with N = 24 and $d\infty$, indicating that the algorithm was usually unable to prove the optimality of the obtained solutions for these instances in the allowed time limit. The running times increased with the number of crops but showed no clear dependency on the number of cropping areas. Moreover, they are usually larger for the cases where no limit is imposed on the unmet demand. This is particularly true for the classes with fewer crops. However, the average percentage of unmet demand is smaller. The number of cropping areas has no clear effect on the number of plots in the optimal solutions, but these numbers are heavily dependent on the number of crops, reaching around 600 different plots for the instances with 24 crops. Santos et al. (2010) and Santos (2009) present some heuristic procedures to reduce the number of plots.

Computational experiments for the CRS-S problem

The computational experiments presented in this section try to analyze the performance of the column-generation algorithm in solving the CRS–S problem and the impact of allowing stocks (under the conditions described in Table B.2 in Appendix). In these tests, the instances were divided into classes according to the total cropping area (A = 500, 1000, 2000 and 4000 m^2) and the total number of crops (N = 10, 15, 20). For the sake of simplicity, a single cropping area is considered. The combination of these parameters yields 12 classes, each one containing 10 instances. All instances were solved for both the situations with stocks allowed and with stocks not allowed. For all the experiments, we fixed c_{ij} = 1 and c_{ij}^{+} = 0, for all i and j.

Table 4 presents the average results obtained, in terms of the algorithm running times in seconds (*time*), the number of plots in the optimal solution (*# plots*), the percentage of unmet demand (*%resid*), and the percentage of production exceeding nominal demand (*%extra*). Run times were limited on 1 hour for each instance.

Table 4: Average results for the 12 classes of instances when no stock is allowed and when stock is allowed according to the conditions presented in Table B.2.

		no stock is allowed				*stock is allowed*			
A	*crop*	*time*	*#plots*	*%resid*	*%extra*	*time*	*#plots*	*%resid*	*%extra*
500	**10**	53.40	62.40	18.16	69.51	52.6	52.6	17.9	77.5
	15	96.30	93.80	43.08	94.86	99.7	90.4	43.0	97.3
	20	404.10	112.50	66.63	96.39	406.8	110.4	66.6	97.6
1000	**10**	15.30	69.80	3.57	56.29	11.6	38.7	3.5	67.3
	15	184.40	200.70	16.10	89.85	272.4	184.7	15.9	93.2
	20	181.89	121.00	46.55	99.21	220.7	120.2	46.5	99.9
2000	**10**	2.70	40.00	0.13	39.11	2.5	21.4	0.1	49.4
	15	94.40	189.20	0.88	72.21	154.4	155.4	0.6	75.4
	20	392.70	349.10	3.39	91.14	918.3	313.4	2.8	93.5
4000	**10**	1.10	25.60	0.00	25.61	1.1	15.8	0.0	40.8
	15	19.90	109.50	0.00	55.14	9.0	49.8	0.0	61.9
	20	79.70	218.40	0.00	68.66	61.1	105.4	0.0	72.8

The computational experiments show that the proposed algorithm is able to find optimal solutions within quite reasonable running times for both the situations with and without stocks. In general terms, the instances become harder as the planting areas and the number of crops increase. From Table 4, one can see that by enabling stocks extra production increases and unmet demand is reduced. In average, a reduction of 5.35% was obtained for the unmet demand with an increase of 8.97% in extra production.

Table 4 also shows that solution allowing stocks present a smaller numbers of plots, up to 50% when A = 4000m2. This is a particularly desired characteristic since solutions with a very high number of plots can be infeasible in real-life situations (Santos et al., 2010). Considering all area sizes, the average reduction was 21.20%.

Computational experiments for the CRS-A problem

When dealing with the Crop rotation scheduling problem with adjacency constraints, we presented two integer optimization models for the CRS–A problem and some of the solution procedures which were proposed by Santos et al. (2008) and Santos (2009). We will now discuss some results obtained with a heuristic based on column generation (CGH) and branch-and-price (B&P) algorithms.

As mentioned in the Solution Methodology Section, most optimization linear models with integer variables belong to the NP class, which means that no efficient algorithms are known that can obtain provable optimal solutions. This is the case of the CRS-A problem. In these situations, it is common to rely on heuristic methods, which provide feasible solutions. Evaluating the quality of solutions obtained is an important issue to be addressed. The objective value of a feasible solution provides a lower bound (LB) to the optimal value for maximizing problems. If we have an upper bound (UB) to the objective function, then we have an interval that the optimal value belongs to. Therefore, we can define a gap value that gives a maximum deviation of the feasible solution value to the optimal value, as follows.

$$gap = 100\,\frac{UB - LB}{|LB| + 10^{-10}}$$

Typically, UB are obtained by relaxing the master problem. A particular relaxation of (36)-(39) consists of relaxing the integrality conditions and solving it by the column generation method described in the Solution Methodology Section.

For the computational tests, the instances were divided into classes according to the total number of plots in the cropping area (L = 4, 8, 12, 16, 20) and the total number of crops (N = 6, 16, 26). Each class consists of 10 crops randomly chosen from the 28 crops in Appendix A. Spatial configurations for cropping areas were also randomly generated. By combining these different parameters, 15 classes were obtained.

Table 5 shows results comparing CGH and B&P algorithms. For each procedure, we present the average results for the running times in seconds (*time*), the percentage of instances solved to optimality (*%opt*), and the gap (*gap*). Run times were limited to 30 minutes for each instance.

Table 5: Average results for 15 classes of instances.

		CGH			B&P		
L	*N*	*time*	*%opt*	*gap*	*time*	*%opt*	*gap*
4	6	214.92	0.00	3.62	1482.30	20.00	1.97
	16	3.08	80.00	0.17	3.65	100.00	0.00
	26	2.61	100.00	0.00	2.52	100.00	0.00
8	6	121.83	0.00	4.58	1817.93	0.00	3.12
	16	4.87	60.00	0.33	8.91	100.00	0.00
	26	5.48	100.00	0.00	5.44	100.00	0.00
12	6	59.89	0.00	5.07	1718.15	10.00	3.37
	16	7.13	80.00	0.06	12.23	100.00	0.00
	26	8.79	90.00	0.07	9.96	100.00	0.00
16	6	76.94	0.00	3.82	1834.29	0.00	3.09
	16	12.24	70.00	0.08	29.02	100.00	0.00
	26	14.17	80.00	0.07	15.75	100.00	0.00
20	6	110.54	0.00	10.31	1884.43	0.00	6.37
	16	16.99	40.00	0.21	68.50	100.00	0.00
	26	17.56	80.00	0.17	19.40	100.00	0.00

The CGH algorithm was able to prove optimality of about 55% of the instances. The average running time was 45.14 seconds and the average gap was 1.9%. The algorithm B&P was able to prove optimality for all instances with $N = 16$ and $N = 26$, within 50.55 seconds. In general, the run time slightly increases with the number of plots. However, the performance of the algorithm seems to be very dependent on the number of crops. For the classes with $N = 6$ crops, it proves optimality within 30 minutes for only 6% of the instances, and a gap of 3.59%. The fact that the algorithm behaves better for instances with a larger number of crops is interesting since this situation is more common in practice.

As a final note, we mentioned that the default branch-and-cut algorithm in CPLEX package performs poorly. Indeed, it was only able to prove optimality within 30 minutes for 40% of the instances with $L = 4$. For the other classes, convergence was not obtained within the time limit and, more importantly, the solver had trouble even finding feasible solutions. Indeed, no feasible solution was obtained for 33.7% of the instances with $L = 16$ and for 93.6% of the instances with $L = 20$. For the whole set of instances, the average gap of the solutions obtained was 13.26% and optimality was proved for only 8% of instances.

CONCLUSIONS AND FUTURE RESEARCH

In this chapter, we presented three problems associated to crop rotation scheduling in the context of vegetable crop production according to ecologically-based criteria. The idea of gathering these problems was to somehow develop a framework that could be used to tackle similar situations. Indeed, the same core structure has been used to model each one of the problems as linear or integer linear programs and similar solution methodology strategies have been devised. A brief analysis of the results suggests that the proposed methodologies are adequate and can be extended to deal with other variants of the problems.

Probably the main assumption to develop the framework proposed in this chapter was related to the definition of a feasible crop rotation schedule. Our choice has been to use general guidelines imposing, for each crop rotation, some desirable features such as the presence of fallow and green manure crops, as well as the prohibition of growing crops of the same botanic family one immediately after another. Although this is a very particular choice, we believe that the proposed models are quite general and can be adapted to include other criteria. One important feature of the proposed methodology is the generality with which it copes with the crop rotation constraints, without the need to explicitly list feasible or infeasible crop sequences.

We foresee many extensions and future perspectives for the study presented here. We can cite the possibility of considering the economical aspect of production in a more explicit manner in our models, the inclusion of additional constraints specific for some growers (such as the climatic limitations to some crops or the options for few specific crops in certain seasons of the year) and the development of visual interfaces that can help the adoption of the methods in cooperatives, to mention a few. Moreover, one interesting research area is the adoption of a more flexible model by occasionally relaxing some of the ecologically-based constraints. We hope that this chapter will encourage research efforts and prove to be a helpful source of ideas and references.

REFERENCES

Altieri, M. A. (1995). *Agroecology: the science of sustainable agriculture*. Boulder: Westview press.

Appa, G. M. (1973). The transportation problem and its variants. *Operational Research Quarterly*, 24:79-99.

Atamtürk, A. & Savelsbergh, M. (2005). Integer-programming software systems. *Annals of Operations Research*, 140:67–124.

Bachinger, J., & Zander P. (2006). ROTOR, a tool for generating and evaluating crop rotations for organic farming systems. *European Journal Agronomy*, 26, 130–143.

Barnhart C., Johnson E. L., Nemhauser G. L., Savelsbergh M. W. P. & Vance P.H. (1998). Branch and price: Column generation for solving huge integer programs. *Operations Research*, 46(3), 316–329.

Blum, C. & Roli, A. (2003). Metaheuristics in combinatorial optimization: Overview and conceptual comparison. ACM Computing Surveys 35(3), 268–308.

Campanhola, C., & Valarini, P. J. (2001). A agricultura orgânica e seu potencial para o pequeno agricultor. *Cadernos de Ciência & Tecnologia*, 18(3), 69-101.

Castellazzi, M. S., Wood, G. A., Burgess, P. J., Morris, J., Conrad, K. F., & Perry, J. N. (2008). A systematic representation of crop rotations. *Agricultural Systems*, 97, 26–33.

Clarke, H. R. (1989). Combinatorial aspects of cropping pattern selection in agriculture. *European Journal of Operational Research*, 40, 70-77.

Costa, A. M., Santos, L. M. R., Alem, D. J., & Santos, R. H. S. (2009, July). Sustainable Vegetable Crop Supply Problem with Perishable Stocks. Paper presented at the EURO Summer Institute, Lleida, Spain.

Dantzig, G. B., & Wolfe, P. (1960). Decomposition principle for linear programs. *Operations Research*, 8, 101-111.

Desaulniers, G., Desrosiers, J. & Solomon, M. M. (Eds) (2005). *Column generation*. Springer, Berlin.

Detlefsen, N., & Jensen, A. L. (2007). Modelling optimal crop sequences using network flows. *Agricultural Systems*, 94, 566–572.

Dogliotti, S., Rossing, W. A. H., & Van Ittersum, M. K. (2003). ROTAT, a tool for systematically generating crop rotations. *European Journal of Agronomy*, 19, 239–250.

El-Nazer, T., & McCarl, B. A. (1986). The choice of crop rotation: A modeling approach and case study. *American Journal of Agricultural Economics*, 68(1), 127–136.

Gliessman, S. R. (2000). *Agroecology: ecological processes in sustainable agriculture*. Chelsea: Ann Arbor.

Gomory, R.E. (1958) Outline of an algorithm for integer solutions to linear programs. Bulletin of the American Mathematical Society 64, 275-278.

Haneveld W. K. & Stegeman A. W. (2005). Crop succession requirements in agricultural production planning. *European Journal of Operations Research*, 166, 406–429.

Hildreth, C. G., & Reiter, S. (1951). On the choice of a crop rotation plan. In Activity Analysis of Production and Allocation, T.C. Koopmans (Ed.), *Proceedings of the Conference on Linear Programming*, 177-188.

ILOG CPLEX 11.1 Reference Manual. 10.1. Copyright by ILOG, France, 2006.

Johnson E. L., Nemhauser G. L. & Savelsbergh M. W. P. (2000). Progress in Linear Programming-Based Algorithms for Integer Programming: An Exposition. *INFORMS Journal on Computing*, 12(1), 2-23; 2000.

Jones, J., Hoogenboom, G., Porter, C., Boote, K., Batchelor, W., Hunt, L., Wilkens, P., Singh, U., Gijsman, A., & Ritchie J. (2003). The DSSAT cropping system model. *European Journal of Agronomy*, 18(3), 235-265.

Kantorovich, L. (1960). Mathematical methods of organizing and planning production (translated from a report in Russian, dated 1939). *Management Science*, 6, 366–422.

Land A. H. & Doig, A. G. (1960). An Automatic Method for Solving Discrete Programming Problems. *Econometrica*, 28, 497-520.

Linderoth J.T. & Ralphs T.K. (2005). Noncommercial Software for Mixed-Integer Linear Programming. In *Integer Programming: Theory and Practice* (pp. 253-303). CRC Press Operations Research Series.

Lübbecke, M. E., & Desrosiers, J. (2005). Selected topics in column generation. *Operations Research*, 53(6), 1007-1023.

Lyon, S. (2006). Evaluating fair trade consumption: politics, defetishization and producer participation. *International Journal of Consumers Studies*, 30, 452-464.

Makatouni, A. (2002). What motivates consumers to buy organic food in the UK? Results from a qualitative study. *British Food Journal*, 104, 345-352.

Nemhauser G. L. & Wolsey L. A. (1988). *Integer and Combinatorial Optimization*. Wiley-Interscience Publication.

Raidl, J. R. & Puchinger, J. (2008). Combining (Integer) Linear Programming Techniques and Metaheuristics for Combinatorial Optimization. Studies in Computational Intelligence (SCI) 114, 31–62.

Santos, L. M. R. (2009). *Programação de Rotação de culturas – modelos e métodos de solução. Doctoral dissertation*, ICMC/USP - University of São Paulo, Brazil.

Santos L. M. R., Michelon, P., Arenales, M. N. & Santos, R. H. S. (2008). Crop rotation scheduling with adjacency constraints. *Annals of Operations Research* (DOI: 10.1007/s10479-008-0478-z).

Santos L. M. R., Costa, A. M., Arenales, M. N., & Santos, R. H. S. (2010). Sustainable vegetable crop supply problem. *European Journal of Operations Research*, 204, 639-647.

Seyfang, G. (2006). Ecological citizenship and sustainable consumption: Examining local organic food networks. *Journal of Rural Studies*, 22, 383–395.

Stöckle C. O., Donatelli, M., & Nelson, R. (2003). CropSyst, a cropping systems simulation model. European *Journal of Operations Research*, 18, 289–307.

Tillman D., Cassman, K. G., Matson, P. A., Naylor, R., & Polask, S. (2002). Agricultural sustainability and intensive production practices. *Nature*, 418, 671–677.

Vanderbei, R. J. (2008) *Linear programming: foundations and extensions*. 3rd edition. Springer, New York.

APPENDIX A

For the computational experiments of the crop rotation scheduling problem with adjacency constraints (CRS-A problem), 28 crops which are cultivated at an organic farming unit in Viçosa, Brazil, were selected. Crops 25 to 28 are used for green manuring. The production time (in number of periods of 10 days) for each crop includes the estimated planting and harvesting periods.

Table A.1. Data for the 28 crops: name, botanic family, planting periods and production time.

CROP		BOTANIC FAMILY		PLANTING MONTH		PRODUCTION TIME (*10 days*)
				beginning	*end*	
1	Lettuce	1	*Asteraceae*	all year round		4
2	Watercress	2	*Brassicaceae*	all year round		6
3	Chinese cabbage	2	*Brassicaceae*	March	June	28
4	Broccoli	2	*Brassicaceae*	February	June	11
5	Cabbage	2	*Brassicaceae*	February	July	11
6	Cauliflower	2	*Brassicaceae*	March	June	10
7	Beet	3	*Chenopodiaceae*	March	July	7
8	Spinach	3	*Chenopodiaceae*	February	July	11
9	Zucchini	4	*Cucurbitaceae*	August	March	9
10	Pumpkin	4	*Cucurbitaceae*	September	January	15
11	Cucumber	4	*Cucurbitaceae*	all year round		8
12	Garlic	5	*Alliaceae*	March	April	15
13	Onion	5	*Alliaceae*	March	June	12
14	Okra	6	*Malvaceae*	August	March	23
15	Corn	7	*Gramineae*	August	April	10
16	Tomato	8	*Solanaceae*	all year round		18
17	Sweet pepper	8	*Solanaceae*	all year round		15
18	Eggplant	8	*Solanaceae*	September	February	16
19	Carrot	9	*Apiaceae*	March	July	19
20	Parsley	9	*Apiaceae*	September	March	14
21	Peruvian carrot	9	*Apiaceae*	March	August	19
22	Snap bean	10	*Leguminosae*	August	April	11
23	Pea	10	*Leguminosae*	March	April	9
24	Bean	10	*Leguminosae*	August	September	9
25	Sun hemp	10	*Leguminosae*	September	December	9
26	Jack bean	10	*Leguminosae*	September	December	10
27	Black velvet bean	10	*Leguminosae*	September	January	11
28	Vetch	10	*Leguminosae*	March	June	14

APPENDIX B

For the computational experiments of the crop rotation scheduling problem with known demands (CRS-D problem) and the crop rotation scheduling problem with perishable stocks (CRS-S problem) we select 23 crops cultivated at an organic farming unit in Barbacena, Brazil. Crops 20 to 23 are for green manuring. The production time (in number of periods of one week) for each crop includes the estimated planting and harvesting periods.

Table B.1: Data for the 23 crops: name, botanic family, planting periods and production time.

CROP		BOTANIC FAMILY		PLANTING PERIOD		PRODUCTION
				beginning	*end*	TIME (weeks)
1	Crisp head lettuce	1	*Asteraceae*	all year round		7
2	Loose leaf lettuce	1	*Asteraceae*	all year round		7
3	Butter head lettuce	1	*Asteraceae*	all year round		7
4	Endive	1	*Asteraceae*	all year round		9
5	Collards	2	*Brassicaceae*	February	September	32
6	Broccoli	2	*Brassicaceae*	February	October	20
7	Cauliflower	2	*Brassicaceae*	March	October	18
8	Beet	3	*Chenopodiaceae*	February	September	11
9	Spinach	3	*Chenopodiaceae*	February	September	20
10	Zucchini	4	*Cucurbitaceae*	October	February	14
11	Pumpkin	4	*Cucurbitaceae*	November	January	19
12	Cucumber	4	*Cucurbitaceae*	September	March	13
13	Garlic	5	*Alliaceae*	March	April	24
14	Onion	5	*Alliaceae*	March	July	24
15	Leek	5	*Alliaceae*	April	April	12
16	Okra	6	*Malvaceae*	November	January	27
17	Tomato	7	*Solanaceae*	all year round		24
18	Carrot	8	*Apiaceae*	all year round		16
19	Parsley	8	*Apiaceae*	October	February	21
20	Black velvet bean	9	*Leguminosae*	October	January	16
21	Jack bean	9	*Leguminosae*	October	February	12
22	Lupine	9	*Leguminosae*	March	July	18
23	Hairy vetch	9	*Leguminosae*	March	July	20

Table B.2: Data for the 19 crops with known demand: name, measuring unity, fraction lost (r_i) of the quantity already in stock of crop i, if it is held for one more period, number of periods(weeks) the crop can be held in stock (G_i), number of periods between planting and first partial harvesting (o_i), and expected harvests per square meter.

CROP	*measuring unity*	STOCK		HARVESTING																				
		r_i	G_i	o_i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1 Crisp head lettuce	*unities*	0.1	1	5	9	3																		
2 Loose leaf lettuce	*unities*	0.1	1	5	9	3																		
3 Butter head lettuce	*unities*	0.1	1	5	9	3																		
4 Endives	*unities*	0.1	1	7	9	3																		
5 Coolards	*pack*	0.1	1	12	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	1	1
6 Broccoli	*pack*	0.2	1	10	1	1	2	2	3	3	3	2	2	1										
7 Cauliflower	*unities*	0.1	2	16	1	3																		
8 Beet	*kg*	0.15	3	8	1	2	1																	
9 Spinach	*pack*	0.1	1	5	3	4	4	4	4	4	4	4	4	4	4	4	3	3	2					
10 Zucchini	*kg*	0.15	2	9	0.2	0.3	0.3	0.3	0.2															
11 Pumpkin	*kg*	0.05	9	16	0.5	0.5	0.5																	
12 Cucumber	*kg*	0.9	4	8	1	2	2	1	1															
13 Garlic	*kg*	0.02	24	23	0.3																			
14 Onion	*kg*	0.05	12	23	3																			
15 Leek	*kg*	0.15	2	8	3	3	3	3																
16 Okra	*kg*	0.1	2	14	0.2	0.3	0.3	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.2	0.2	0.2							
17 Tomato	*kg*	0.1	2	15	0.8	0.8	1	1	1.2	1.2	1	0.8	0.2											
18 Carrot	*kg*	0.1	2	13	1.5	2	1.5																	
19 Parsley	*pack*	0.1	1	8	14	16	16	16	16	16	16	16	16	16	16	14	10							

Sistema Administrativo da Pós-Graduação — Janus

Login

Docente
Alysson Machado Costa
Sair

Apresentação

- Apresentação
- Vídeos tutoriais

+ Acesso público

+ Senha / Email

+ Pessoa

- Aluno regular

- Ficha do aluno
- Pesquisa de alunos
- Envio de email

- PAE

- Consultas
  - Certificado para supervisor

- Matricula

- Aceitar matrícula de acompanhamento
- Avalização
- Deferimento do ministrante
- Cancelamento

- Docente/Orientador

- Carga Horária
- Lista de orientados
- Responsável por disciplinas

+ Disciplina

- Mapa do *site*
- E-mails automáticos

- Egressos USP

Lista de Orientados

Clique no nome de um aluno para selecioná-lo e visualizar a sua Ficha.

Preparar para imprimir ◯ Atuais ◯ Concluídos ◉ Ambos

Mestrandos atuais

1 aluno(s) encontrado(s).

Área	NUSP	Nome	Contagem de prazo Início	Contagem de prazo Término	Situação	Data da situação	Matriculado
Ciências de Computação e Matemática Computacional (55134)	5908961	Pedro Belin Castellucci	04/02/2013	04/02/2016	Matriculado	04/02/13	Sim

Doutorandos atuais

2 aluno(s) encontrado(s).

Área	NUSP	Nome	Contagem de prazo Início	Contagem de prazo Término	Situação	Data da situação	Matriculado
Ciências de Computação e Matemática Computacional (55134)	6714293	Felipe F. B. Araújo	25/05/2011	25/01/2016	Matrícula de Acomp.	15/02/13	Sim
Ciências de Computação e Matemática Computacional (55134)	6711776	Mayron C. de O. Moreira	15/03/2011	15/11/2015	Matrícula de Acomp.	15/02/13	Sim

Mestrados concluídos

3 aluno(s) encontrado(s).

Área	NUSP	Nome	Data da defesa
Ciências de Computação e Matemática Computacional (55134)	6714293	Felipe Francisco Bezerra Araújo	03/05/11
Ciências de Computação e Matemática Computacional (55134)	6711776	Mayron César de Oliveira Moreira	15/04/11
Ciências de Computação e Matemática Computacional (55134)	5110728	Pamela Michele Candida Cortez	09/03/12

Doutorados coorientados concluídos

1 aluno(s) encontrado(s).

Área	NUSP	Nome	Data da defesa
Ciências de Computação e Matemática Computacional (55134)	5308134	Claudia Fink	19/10/12