Ann Oper Res (2018) 267:153–177
https://doi.org/10.1007/s10479-017-2568-2

MULTIPLE OBJECTIVE OPTIMIZATION

A multiple objective methodology for sugarcane harvest management with varying maturation periods

Helenice de Oliveira Florentino[1] · Chandra Irawan[2] · Angelo Filho Aliano[3] · Dylan F. Jones[2] · Daniela Renata Cantane[1] · Jonis Jecks Nervis[4]

Published online: 30 June 2017

Abstract This paper addresses the management of a sugarcane harvest over a multi-year planning period. A methodology to assist the harvest planning of the sugarcane is proposed in order to improve the production of POL (a measure of the amount of sucrose contained in a sugar solution) and the quality of the raw material, considering the constraints imposed by the mill such as the demand per period. An extended goal programming model is proposed for optimizing the harvest plan of the sugarcane so the harvesting point is as close as possible to the ideal, considering the constrained nature of the problem. A genetic algorithm (GA) is developed to tackle the problem in order to solve realistically large problems within an appropriate computational time. A comparative analysis between the GA and an exact method for small instances is also given in order to validate the performance of the developed model and methods. Computational results for medium and large farm instances using GA are also presented in order to demonstrate the capability of the developed method. The computational results illustrate the trade-off between satisfying the conflicting goals of harvesting as closely as possible to the ideal and making optimum use of harvesting equipment with a minimum of movement between farms. They also demonstrate that, whilst harvesting plans for small scale farms can be generated by the exact method, a meta-heuristic GA method is currently required in order to devise plans for medium and large farms.

Keywords Multiple objective optimization · Goal programming · Genetic algorithm · Sugarcane harvest planning

✉ Helenice de Oliveira Florentino
helenice@ibb.unesp.br

1 Department of Biostatistics, UNESP - Univ Estadual Paulista, Botucatu, SP, Brazil

2 Department of Mathematics, Centre for Operational Research and Logistics, University of Portsmouth, Portsmouth, UK

3 Academic Department of Mathematics, Federal Technology University of Paraná, Apucarana, PR, Brazil

4 Energy in Agriculture, FCA, UNESP - Univ Estadual Paulista, Botucatu, SP, Brazil

1 Introduction

In recent years, the increased production of sugarcane in tropical countries has led to a corresponding increase in the size and complexity of the decision problems associated with sugarcane mills. The challenges caused by this accelerated growth have caused difficulties for managers of companies in this sector. Thus, any tool to support decision making, to optimize managerial plans and to obtain estimations of the quality of the harvest will be of benefit to the sector. As a particular country example, Brazil has prominence in the world market for sugar and alcohol. According to the United States Department of Agriculture, USDA (2015), Brazil is the world's largest producer and exporter of sugar; is the second largest producer of ethanol in the world (Worldwatch institute 2015); and is the world's largest sugarcane producer (Conab 2016). The sugar-alcohol sector contributed 1.85% of the Brazilian GDP and 29% of the Brazilian agricultural GDP in 2015, and employs approximately 4.4 million people (Florentino et al. 2015).

Based on Brazilian Ministry of Agriculture and Livestock statistics, in the 2015/2016 season Brazil produced around 658 million tonnes of sugarcane. Ninety-three percent of this production came from the Brazilian Center-South region (Conab 2016). This region produces 93% of Brazilian total ethanol and sugar (Conab 2016).

As sugarcane makes a significant contribution to the Brazilian economy, several studies have been undertaken to improve the quality of the sugarcane and to assist in understanding its production cycle (Florentino et al. 2015; Higgins and Postma 2004). In contrast to many crops, the production cycle of the sugarcane starts with its planting in the first year. Annual harvesting of the sugarcane can in principle take place at least four times before it needs to be replanted (renewal). However, there is no guarantee that good quality sugarcane will be produced using a plantation that has already been harvested multiple times (Calija et al. 2001; Magalhães and Braunbeck 2014).

The period when the sugarcane should can be harvested is known as the period of industrial utilization (PIU). Generally, in Brazil the PIU starts from 2 months before the maximum sugarcane maturation point and finishes 2 months after. The sugarcane should be harvested as closely as possible to this maturation date, taking into account the technical limitations and the ongoing demands of the mill. However, the dimensions and the complexity of the current sugarcane fields make the achievement of the above goal very difficult. This is in part, due to the limited amount of machinery for harvesting, processing and transporting the sugarcane and in part due to the sheer size of the operation in terms of land area and hence sugar to be harvested. Therefore, optimal harvest planning is one of the most important tasks if a good production of sugarcane is to be achieved. To assist decision makers in determining the optimal harvesting plan, in this paper we propose a model and an appropriate solution method to optimize the sugarcane harvesting plan.

The optimized planning of a sugarcane crop should improve agricultural and industrial practices so that all of the relevant stakeholders (the farm owners, employees and the onward supply chain) gain maximal benefit from the process. The sugarcane should be harvested when it reaches the maximum content of sucrose (pol% cane), which occurs in the peak period of maturation. This period is dependent on the system of cultivation adopted, the sugarcane variety, the region and other factors that influence the quality of the raw material obtained (López-Milán and Plà-Aragonés 2014; Ramesch and Mahadevaswamy 2000).

In Brazil a further climatic restriction is that the recommended period for harvesting sugarcane is from April to December (Yirsaw et al. 2000). According to Florentino and Pato (2014) and López-Milán and Plà-Aragonés (2013), several kinds of adversities could potentially occur (e.g., climate related, administrative, social, or economic problems), but the planning process should incorporate mitigation actions or sufficient flexibility in order to prevent serious deviations from the goal of harvesting at the peak of the sugarcane maturation.

In da Silva et al. (2015), a goal programming model is proposed for sugarcane harvest planning which aims to simulate several scenarios that involve uncertain parameters and hence minimize agro-industrial costs. The authors in Paiva and Morabito (2008) present an optimization model to support decision making in the aggregated production planning of sugar and ethanol companies based on industrial process selection and production lot-sizing models. Their model aims to select industrial processes used to produce sugar, ethanol and molasses and hence determine an optimal logistical configuration. A linear optimization model for sugarcane cultivation and harvest planning is proposed in Supsomboon and Niemsakul (2014) in order to maximize commercially recoverable sugar content by set of Thai farms.

In Sharma et al. (2003), the optimal mix of sugarcane fertilizer is found using lexicographic goal programming with a quadratic distance measure. A case study arising from Indian sugarcane farms is used to illustrate the methodology. The majority of other recent works that use goal programming for harvest planning are related to the forestry sector. Bagdon et al. (2016), Demirci and Bettinger (2015), Gómez et al. (2011), Martins et al. (2014), Weintraub and Murray (2006) and Zengin et al. (2015) all fall into this category and contain a range of goals relating to the sustainability and effective management of forests. In Baraku et al. (2015), production planning across a set of eight agricultural farms is optimized via goal programming. In Prišenk and Turk (2015) and Prišenk et al. (2014), the weighted goal programming is used to treat the crop rotation problem in organic farms in Slovakia.

Given the above successful track record of goal programming in modeling harvest planning problems, together with the goal based nature of the requirement to harvest as closely as possible to maturation, a goal programming methodology is chosen to model the sugar cane harvesting problem in this paper. Furthermore, as the balance between the average and worst case deviations from the maturation goals amongst the set of plots to be harvested is also of interest, the extended goal programming variant is chosen for this purpose.

The above discussion demonstrates that whilst there are literature examples relating to the optimal planning and harvesting of sugarcane, the literature focusses on cost reduction, mill capacity planning and transportation logistics. It is hence concluded that a work aimed at sugarcane harvest planning considering the quality of the cane harvested, operational constraints and mill demands would provide a novel and relevant contribution to the literature. Hence, this paper proposes to develop:

(i) a mathematical model to obtain an optimal sugarcane harvest plan using goal programming in order to maximize the sucrose and sugarcane production whilst respecting the constraints imposed by the mill, and
(ii) an efficient solution method for solving the above model. This will utilize genetic algorithm (GA) methodology as the model is relatively hard to solve for the large-scale problems occurring in modern farms.

The remainder of this article is divided into five sections. In Sect. 2, we present a discussion of the factors relevant to the planting and harvesting of sugarcane that will inform the model built in this paper. In Sect. 3, we formulate a new goal programming based model to optimize the harvest schedule in order to minimize the sum of deviations from the maturation period for each lot as well as to minimize the movement of machines between farms. In this way, the harvest is always carried out close to the sugarcane maturation. In Sect. 4, a metaheuristic is proposed—a genetic algorithm which includes four novel specialized heuristics—specifically developed to solve the large size instances that occur in practice. The computational results from using an exact method (for small scale instances) and the GA (for all instances) are presented in Sect. 5. In Sect. 6, some conclusions and future perspectives are detailed.

2 Factors in the timing of the sugarcane planting and harvesting lifecycle

The sugarcane can be used to produce ethanol in a sugar mill which is supplied by several sugarcane farms. The number of sugarcane farms that supply a mill depends on the size and demand of the company. In addition, it is also affected by the maximum amount of raw material that can be harvested. In Brazil, the number of farms that serve a mill generally varies between 1 and 40 with an average of 35 farms for large a company. Each farm is divided into a set of smaller areas called plots. A flat plot is preferred with canes planted in long lines to avoid a lot of machine maneuvers. In general, sugarcane fields are subdivided according to soil topography and homogeneity where each field has an average of 10 to 20 hectares.

In tropical countries such as Brazil, when the sugarcane is planted in months from January to April, it should be harvested 18 months after planting. This is termed year-and-half sugarcane, ($t^* = 18$, PIU period is $t_0 + 18 \pm 2$). This sugarcane presents a minimal growth rate between May and September, when the weather is relatively cold. The next development phase of the sugarcane occurs from October to April with December being the best period for the sugarcane due to higher rainfall, longer daylight hours and a higher average temperature. When the sugarcane is planted in September and October, it should be harvested 12 months after planting. This is termed year sugarcane ($t^* = 12$, PIU period is $t_0 + 12 \pm 2$). The next development phase of the sugarcane occurs from November to April, when the growth of the sugarcane starts to reduce due to the weather conditions characterized by a lack of rain and lower average temperatures. Sugarcane planted from May to August is called winter sugarcane, where irrigation is needed and the harvest also takes place 12 months after it has been planted (Picoli et al. 2014; Rudorff et al. 2010). In general, the period (in months) for harvesting (t_1) is calculated by $t_1 = t_0 + t^* \pm d$, where t_0 is the month in which the sugarcane was planted, t^* is the number of periods (months) required for the sugarcane to mature (which is dependent on t_0) and d is a deviation between the ideal and the actual harvesting points. In other words, if $d = 0$, the sugarcane is harvested at the point of maximum maturation. If $d \in [-2, 2]$ the sugarcane is in the PIU.

The setting of the time for renewal of a sugarcane plantation is related to the sugarcane productivity due to the age of the crop. At some stage renewal needs to be considered in order to increase the productivity at the expense of a larger initial cost. The sugarcane after the first cut is called ratoon sugarcane. After the cut, the sprouting of stumps and the beginning of a new stage of cutting occur. With the increase of the number of stages of cutting, a gradual loss occurs in agricultural productivity (Higgins 1999). The cutting stages of the ratoon sugarcane are repeated yearly until the crop is no longer economically profitable. When this happens the culture needs to be reformed and the cycle restarts with the planting of new seedlings (Landers 2007). The productivity of a year-and-half sugarcane appears to be higher than its counterpart, year sugarcane, due to the longer time that the sugarcane remains in the field. The productivity of the first cutting of the year sugarcane is approximately equal to the productivity of the second cutting of the year-and-half sugarcane (Higgins 1999).

In Brazil the sugarcane is harvested from April to December (Hofsetz and Silva 2012). More specifically, in the Brazilian South West region, the sugarcane maturation period occurs from April or May to its peak in September due to the climatic conditions prevailing in this period. The gradual decrease in the temperature and the decrease in rainfall are crucial for the maturation process (Cardozo and Sentelhas 2013; López-Milán and Plà-Aragonés 2013; Vianna and Sentelhas 2014) in the different production environment of the Center-South region of Brazil.

The determination of the maturation of sugarcane is directly linked to the sucrose content, presence of flowering, genetics, climate, soil, management, age of the sugarcane and other factors. A further important factor is the variety of sugarcane used.

Sugarcane varieties are classified as early variety, when they have a POL content above 13% (at the beginning of May), intermediate variety when they reach maturity in July, and late variety when the peak of maturation occurs in August or September, assuming the same date of planting or cutting for each variety (Landers 2007).

3 Mathematical model

3.1 Notations and assumptions

In this section, a mathematical model is developed to optimize the sugarcane harvesting plan in an area containing different varieties with different maturation periods. An agricultural area consists of F farms where each farm is divided into several plots. In total there are k plots, and each plot is planted with one sugarcane variety.

There are n different possible sugarcane varieties to select from each plot. It is assumed that the variety planted for each plot (j) is known, and the date (t_{0_j}) when this variety was planted is also fixed $j = 1, \ldots, k$. The problem is to determine the harvesting plan of this sugarcane during the planning horizon in order to satisfy all demand (D_i) in established months (T_i) and to harvest the sugarcane for each month (t_j) in the PIU, $(t_j = t_{0_j} + t^* + d_j)$. The preferred harvest time is in the period as close as possible to the maximum maturation period $(t_{0_j} + t^*)$ of the sugarcane. The pol constraints demand imposed by the mill, $i = 1, \ldots, m$; $j = 1, \ldots, k$, should also be considered.

There are multiple objectives to be considered in this problem. The first one aims to minimize the sum of deviations from the optimal maturation in all lots to be harvested. Due to the high cost of machinery, we also want to minimize the number of farms being harvested in the same period. However, these objectives are conflicting, i.e., the optimization of one leads a worsening of the other, and vice-versa, because if we try the minimize the deviations from the optimal maturity, then the model chooses to harvest several farms in the same period. On the other hand, if the machinery is limited to a lower number of farms in the same period, then the tendency of generating delays in the sugarcane harvesting is evident. The two conflicting objectives have different preference structures. The harvest plan must be achieved as closely as possible, considering both the average and worst case deviations, whereas the number of farms visited should be kept within a reasonable level. Hence, a plan which harvests as closely as possible to the ideal, whilst keeping the number of farms visited to a reasonable level, should be devised.

Hence, a new mathematical model is developed to tackle the harvest problem in the presence of multiple conflicting goals and the need to balance deviations as follows.

Consider k plots and F farms (Farm 1 with r_1 plots, farm 2 with r_2 plots,…, farm F with r_F plots), where the sets of plots within farm f $(f = 1, \ldots, F)$, denoted by J_f, are defined as $J_1 = \{1, \ldots, r_1\}$, $J_2 = \{r_1 + 1, \ldots, r_1 + r_2\}$,…, $J_F = \{r_{F-1} + 1, \ldots, r_{F-1} + r_F\}$ and $r_1 + \ldots + r_F = k$, and are illustrated in Fig. 1.

The following indices, parameters and variables will be used in the optimization model:

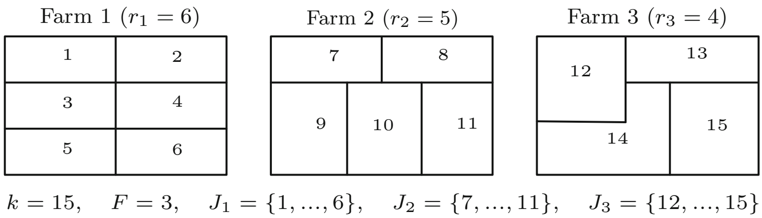

Fig. 1 Illustration of data with 3 farms consisting of 15 plots

Indices:

- i is associated with the period (months) to harvest and to satisfy the demand;
- j is associated with the plots;
- f is associated with the farms.

Parameters:

- k is the number of the plots that can be harvested;
- m is the number of the months for harvesting sugarcane;
- F is the number of farms;
- T_i is the i-th demand period (in month);
- t_{0_j} is the month when the planting or last harvesting of the sugarcane has occurred in plot j;
- t_j^* is set equal to 12 if the sugarcane planted in plot j is a year-and-half sugarcane and 18 otherwise;
- α is the parameter that controls the mix of objective weights, $0 \leq \alpha \leq 1$;
- P_j is the productivity of the sugarcane planted in plot j;
- L_j is the size of plot j;
- D_i is the demand in the i-th month;
- J_f is the set of plots within farm f, where $J_1 = \{1, \ldots, r_1\}$, $J_2 = \{r_1 + 1, \ldots, r_1 + r_2\}$, ..., $J_F = \{r_{F-1} + 1, \ldots, r_{F-1} + r_F\}$ with $r_{F-1} + r_F = k$.

Decision variables:

- x_{ij} binary integer (= 1, if there exists some plot of the farm f that is harvested in month i, and 0 otherwise) for all $i = 1, 2, \ldots, m$; $j = 1, 2, \ldots, k$;
- y_{if} binary integer (= 1, if there exists some plot of the farm f that is harvested in month i, and 0 otherwise) for all $i = 1, 2, \ldots, m$; $f = 1, 2, \ldots, F$;
- N_i is related to the farms harvested in month i;
- t_j is the decision variable associated with the best month for the harvesting the sugarcane in plot j;
- d_j^+ is the deviational variable associated with positive deviation in plot j;
- d_j^- is the deviational variable associated with negative deviation in plot j;
- θ is the maximum deviation among all plots.

3.2 Multiobjective model

We propose a new multiobjective model presented below, where the objective (1) is to minimize the sum of deviations from t_j, ($t_j = t_{0_j} + t_j^* + d_j^+ - d_j^-$), for harvesting the sugarcane in each plot j ($j = 1, \ldots, k$) that satisfies the i-th demand of the mill (D_i).

$$\text{minimize} \quad z_1 = \sum_{j=1}^{k} d_j^+ + d_j^- \tag{1}$$

$$\text{minimize} \quad z_2 = \theta \tag{2}$$

$$\text{minimize} \quad z_3 = \sum_{i=1}^{m} N_i \tag{3}$$

$$\text{subject to} \quad t_j - t_{0_j} - t_j^* - d_j^+ + d_j^- = 0, \qquad j = 1, \ldots, k, \tag{4}$$

$$t_j = \sum_{i=1}^{m} T_i \cdot x_{ij}, \qquad j = 1, \ldots, k, \tag{5}$$

$$\sum_{i=1}^{m} x_{ij} = 1, \qquad j = 1, \ldots, k, \tag{6}$$

$$\sum_{j=1}^{k} P_j \cdot L_j \cdot x_{ij} \geq D_i, \qquad i = 1, \ldots, m, \tag{7}$$

$$d_j^+ + d_j^- \leq \theta, \qquad j = 1, \ldots, k, \tag{8}$$

$$x_{ij} \leq y_{if}, \qquad i = 1, \ldots, m, \; j \in J_f, \; f = 1, \ldots, F, \tag{9}$$

$$N_i = \sum_{f \in J_f} y_{if}, \qquad i = 1, \ldots, m, \tag{10}$$

$$y_{if} \in \{0, 1\}, \; x_{ij} \in \{0, 1\}, \quad d_j^+ \geq 0, \quad d_j^- \geq 0, \qquad i = 1, \ldots, m, \quad j = 1, \ldots, k, \quad f = 1, \ldots, F. \tag{11}$$

This period t_j should be chosen as close as possible to the period of the maximum maturation ($t_{0_j} + t^*$), i.e., the objective is to minimize the sum of the deviations from this value across all plots. The objective (2) minimizes the maximal deviation from amongst the set of deviations of all plots. The objective (3) minimizes the total number of different farms to be harvested in the planning horizon, in order to avoid excessive movements of harvesting machinery, with will hence minimize subsequent soil compaction and machine travel costs.

The goal set (4) defines the period for harvesting sugarcane. Equation set (5) ensures that the harvesting is made within the demand period. Equation set (6) imposes the constraint that each plot is only harvested once. Equation set (7) guarantees that the all demands are met. Constraints (8) impose an upper bound on the deviations. The equation set (9) links variables x_{ij} and y_{if}. Equation set (10) defines the number of the farms harvested in month i. Sign restriction set (11) defines the binary and non-negative variables.

In order to solve the binary linear multiobjective model (1)–(11) an achievement (scalarization) function and objective bound set are proposed by Eqs. (12) and (13) respectively. The objective in (12) is composed of objectives (1) and (2):

$$\text{minimize} \quad z_4 = \alpha \cdot \sum_{j=1}^{k} \left(d_j^+ + d_j^- \right) + (1 - \alpha) \cdot \theta, \tag{12}$$

where $\alpha \in [0, 1]$. In fact, the objective (12) and the constraints (4)–(11) form an extended goal programming model according to González-Pachón and Romero (2001) and Romero (2004). The constraints (13) considers the feasible upper bound G, where G is the maximum number of farms to be harvested in each month. This leads to the following replacement of objective (3) by the upper objective bound set (13), thus reducing the tri-objective model

(1)–(11) to a more pragmatic extended goal programming model, (4)–(13) that is also in accord with the preferential reasoning of the mill owner to achieve the set of harvesting goals as closely as possible whilst limiting the movement between farms to a reasonable level,

$$N_i \leq G, \quad i = 1, \ldots, m. \tag{13}$$

In Sect. 4 a genetic algorithm to solve the model (1)–(11) is proposed.

4 A genetic algorithm

In order to solve this problem for the large instances that occur in practice, a metaheuristic method based on genetic algorithms (GA) is developed to obtain good quality solutions within a reasonable computing time. The use of GA is justified because an exact method (in this study the CPLEX solver using state-of-the-art integer programming solution techniques) is not able to solve large instances of the problem in reasonable time. This will be demonstrated by the computational results, where CPLEX was not able to solve instances with more than 50 lots for objective (3), which in reality corresponds to the smallest mill sizes. The choice of GA is linked with its simplicity of implementation, low computational cost, and good results solving in combinatorial multiobjective problems according to Deb (2001) and Jones et al. (2002), because it works with a set of solutions instead of a single one.

The steps of this method are described in the following subsections.

4.1 Codification

A solution for the harvest problem is treated as an individual, which is defined as a vector $X \in \mathbb{N}^k$, where each component $x_j \in \{1, \ldots, m\}$ denotes the period in which plot j is harvested. This encoding has the advantage of simplicity and providing all the information needed for the proposed problem.

4.2 Initial population

The initial population of the GA is carefully generated in order to ensure the required level of variability and feasibility in the population so that the process will be able to sufficiently explore the search space. This particular way of generating the initial population, with different characteristics via multiple procedures is bespoke for the sugarcane harvesting model considered in this paper, but hopefully has sufficient generic aspects to be considered a contribution to the wider multiobjective GA initial population construction literature. The well-established genetic principle behind the process is based on the fact that a heterogeneous and high genetic variability population has a greater chance to develop and generate more promising and distinct descendants.

This population is constructed by four constructive algorithms defined below. This is necessary because the deviations and demand constraints compete in opposite directions. A heuristic solution that satisfies the demand has high deviations, whereas, a low deviation solution tends not to satisfy the demand.

The n individuals in the population were created as follows[1]:

- $\frac{n}{3}$ individuals by the **Procedure 1**.

[1] A non-uniform distribution of each algorithm was used, because **Procedures 1** and **3** have a high computational cost.

- $\frac{n}{6}$ individuals by the **Procedure 2**.
- $\frac{n}{3}$ individuals by the **Procedure 3**.
- $\frac{n}{6}$ individuals by the **Procedure 4**.

The four procedures, each with different constructive characteristics, are defined in the following subsections.

4.2.1 Procedure 1

This procedure constructs vector X by assigning a random number between 1 and m for each component j, with a normal distribution with mean $t_{0_j} + t^*$ and variance generated between 0.1 and 5. The idea of this procedure is to build a harvest calendar where lot j is harvested as close to its optimum maturation period so a smaller variance will be generated. The advantages of this algorithm include its simplicity, variability of solutions and the relatively low sum of deviations; whereas the drawback is that the solutions may not be feasible with respect to the demand constraints.

The pseudocode of this algorithm is shown below.

Algorithm 4.1 Procedure 1

```
Input: data of the problem
X = ∅
for j = 1, ..., k do
    Generate randomly a value for variance σ² ∈ [0.1, 5]
    Pick randomly value x_j in between 1 and m using normal distribution with mean t0_j + t*_j and variance σ²
    X = X ∪ {x_j}
end for
Output: X
```

4.2.2 Procedure 2

This procedure generates a feasible solution with respect to the demand constraints, without taking the deviations into account. Initially, the **Procedure 1** is called to build a solution to the problem. Let X be the solution. Then, we calculate a residue vector R whose component i formulated as follows:

$$R_i = \sum_{j:X_j=i} P_j \cdot L_j - D_i, \quad i = 1, \ldots, m.$$

If $R_i \geq 0$, in period i the demand is satisfied, otherwise i is not. Set $\mathcal{I} = \{i : R_i < 0\}$. If $\mathcal{I} = \emptyset$, then the generated solution is feasible with respect to the required demand in all periods, otherwise it is infeasible. When the solution is infeasible, the following procedure will transform the solution into a feasible solution. Analyze each element j of X in position, whose period already satisfies the demand. The idea is to put into this position j the amount that the period lacks in demand. By making this change, the residual associated with this new solution is analyzed. If it remains positive in the position where it was excluded from that period, then the exchange is continued until the demand of period i is satisfied. Otherwise, the change is undone and a new permutation of lots to be analyzed is performed. The process ends when all components of the set $\mathcal{I}$ are checked.

Example 1 Consider the following data: $m = 3$ periods, $j = 4$ lots, $P = (110, 120, 140, 160)^T$, $L = (20, 17, 16, 14)^T$ and $D = (2000, 2300, 2200)^T$. Suppose the following solution has been obtained by the **Procedure 1**: $X = (3, 1, 2, 1)^T$, indicating that the lots $j = 1, 2, 3, 4$ are harvested in periods 3,1,2,1 respectively.

Suppose that the order of the lots to be harvested is 1, 3, 4 and 2. This scheme gives a residue $R = (2280, -60, 0)^T$, indicating that in period $i = 2$ there is a lack of 60 units of sugar. To obtain a feasible solution, assign some component of X to period $i = 2$ while satisfying the demand in periods 1 and 3.

- Starting with $j = 1$, assign the harvest period in this lot to period $i = 2$. The new solution will be $X' = (2, 1, 2, 1)$, where the residue $R = (2280, 2140, -2200)^T$, meaning that the new solution is still infeasible and the original solution will still be used.
- For the second iteration, analyze the third lot. The harvest period in this lot can not be changed since $x_3 = 2$ already, which signifies a shortfall in the production period.
- The next lot to be analyzed is $j = 4$, the new solution $X' = (3, 1, 2, 2)^T$. Its residue is $R = (40, 2180, 0)^T$, indicating that this solution is feasible. Therefore, the procedure terminates.

The pseudocode for this procedure is given below.

Algorithm 4.2 Procedure 2

```
Input: data of the problem
Build a solution X by the Procedure 1
Calculate R_i = \sum_{j:X_j=i} P_j · L_j − D_i for all i = 1, ..., m
Calculate I = {i : R_i < 0}
for i ∈ I do
  Let p a random permutation of the {1, ..., k}
  for j ∈ p do
    if p_j ≠ i then
      x_{p_j} ← i
      Calculate R_{p_j} and R_{p_i}
      if R_{p_j} < 0 then
        Undo the change of periods in the position p_j
      end if
      if R_{p_i} ≥ 0 then
        BREAK
      end if
    end if
  end for
end for
Output: optimized solution X
```

4.2.3 Procedure 3

Note that **Procedure 2** only considers the feasibility of the solution which may generate a harvest schedule with relatively high deviations. This procedure seeks a feasible solution with minimal deviations without violating the demand constraints which is described as follows. First, compute vector d deviations of the solution X by using the following expression:

$$d = |T_X - \left(t_0 + t^*\right)|,$$

where $T_X = T_{x_j}, j = 1, \ldots, k$ is the harvest period of lot j. Then we analyze all indexes $\mathcal{J} = \{j : d_j > 0\}$ to examine the possibility of changing the harvest periods of each lot to reduce the corresponding deviation without violating the demand constraints. For each lot $j \in \mathcal{J}$, we calculate the production $P_j \cdot L_j$ and the residue in the harvest period which is allocated for this lot, i.e., R_{x_j}. If $P_j \cdot L_j \leq R_{x_j}$ and meets the demand constraint, then the zero deviation period can be attributed to this lot, which can be written as $x_j = \max\{t_{0_j} + t_j^* - \left(\min_j\{T_j\} - 1\right), 1\}$. Otherwise, any change in the period for this lot will make the residue smaller than 0. The vector R is then updated, and the procedure continues for the other components of $\mathcal{J}$. Upon completion, it is expected to produce a feasible solution with respect to the demand constraints, with a reasonably low sum of deviations.

Example 2 Consider the same data given in Example 4.1 and $t_0 = (8, 9, 7, 5)^T$, $t^* = (12, 12, 18, 18)^T$ and $T = (22, 23, 24)^T$. The deviation of the feasible solution $X = (3, 1, 2, 2)^T$ is $d = (4, 1, 2, 2)^T$, so $\mathcal{J} = \{1, 2, 3, 4\}$ and $R = (40, 2280, 0)^T$.

- $j = 1$. $P_1 \cdot L_1 = 2200 > 0 = R_{x_1}$, then changing to this lot is not allowed.
- $j = 2$. $P_2 \cdot L_2 = 2040 > 40 = R_{x_2}$, then changing of this lot is not allowed.
- $j = 3$. $P_3 \cdot L_3 = 2240 < 2280 = R_{x_3}$, then changing to this lot is allowed. The period allocated for this lot is the one which generates the lowest possible deviation, i.e., $x_3 = 3$. Then, $X = (3, 1, 3, 2)^T$ and $R = (40, 40, 2140)$.
- $j = 4$. $P_4 \cdot L_4 = 2240 > 40 = R_{x_4}$, then changing to this lot is not allowed.

Thus, the new feasible solution produces deviations whose sum is $\sum_i d_i = 8$.

The pseudocode of this procedure is given below.

Algorithm 4.3 Procedure 3

```
1: Input: data problem and a feasible solution X
2: Calculate d = |T_X − (t_0 + t*)|
3: Calculate J = {j : d_j > 0}
4: for j ∈ J do
5:    if P_j · L_j ≤ R_{x_j} then
6:       x_j ← max{t_{0_j} + t*_j − (min{T} − 1) , 1}
7:       Update R
8:    end if
9: end for
10: Output: solution X
```

4.2.4 Procedure 4

The procedure developed in this subsection is a matheuristic (hybridization of an exact method and heuristic algorithm) which aims to build, deterministically, a feasible solution to the problem. In the first step, a heuristic technique is used where a feasible solution is heuristically generated that satisfies only a subset $\mathcal{I} \subset \{1, \ldots, m\}$ of the demand constraints. For each element $i \in \mathcal{I}$, a lot is selected to meet the demand and provide the smallest deviation. From this initial stage, there is a solution X, an undefined harvest period for each lot in set $\mathcal{J}$. A mathematical model which can be solved by an exact method is proposed to obtain the harvest period for each lot in order to minimize the total sum of deviations. The formulation of the model is given as follows:

$$\text{minimize} \quad z_1 = \sum_{j \in \mathcal{J}} d_j^+ + d_j^- \tag{14}$$

$$\text{subject to} \quad t_j - t_{0_j} - t_j^* - d_j^+ + d_j^- = 0, \qquad j \in \mathcal{J}, \tag{15}$$

$$t_j = \sum_{i=1}^{m} T_j \cdot x_{ij}, \qquad j \in \mathcal{J}, \tag{16}$$

$$\sum_{i=1}^{m} x_{ij} = 1, \qquad j \in \mathcal{J}, \tag{17}$$

$$\sum_{j \in \mathcal{J}} P_j \cdot L_j \cdot x_{ij} \geq D_i, \qquad i \in \{1, \ldots, m\} - \mathcal{I}, \tag{18}$$

$$x_{ij} \in \{0, 1\}, \quad d_j^+ \geq 0, \quad d_j^- \geq 0, \qquad i = \{1, \ldots, m\} - \mathcal{I}, \quad j \in \mathcal{J}. \tag{19}$$

The idea of this procedure is to generate a partial solution heuristically in order to satisfy the demand, then the exact method is used to obtain a feasible solution with a minimum total deviation. As the cardinality of $\mathcal{I}$ increases, the problem (14)–(19) has fewer variables and constraints, and does not require as much computational effort, since in its formulation only includes the variables x_{ij}. The variability of solutions is achieved by assigning different $\mathcal{I}$. Then, the resulting solution will be the union of the heuristic and exact steps.

The pseudocode for this algorithm is given below.

Algorithm 4.4 Procedure 4

1: Input: data of the problem and $\mathcal{I}$, with $|\mathcal{I}| < m$
  %Step 1
2: **for** $i \in \mathcal{I}$ **do**
3:   **while** Demand for the period i is not satisfied **do**
4:     Determine the set lots $\mathcal{L}_i$, in ascending order of deviation and who have not had their defined harvests, to be harvested in the period i
5:     $x_{\mathcal{L}_i} \leftarrow i$
6:     Update set $\mathcal{L}_i$
7:   **end while**
8: **end for**
  %Step 2
9: Determine $\mathcal{J}$, the lots that have not yet been scheduled
10: Solve the problem (14–19)
11: Allocate in X in the positions $j \in \mathcal{J}$ the periods determined by the Step 2
12: output: solution X

When the cardinality $\mathcal{I}$ increases, the problem of minimizing the deviations is easier to solve, however, the final solution is found to have a higher deviation. On the other hand, when $|\mathcal{I}|$ is small, smaller total deviations are obtained but this require more effort to optimize the problem (14–19). In order to maintain a compromise between these goals, $|\mathcal{I}|$ is set to the value 2 in Step 1.

4.3 Fitness

The fitness (evaluation) of each solution X in the population, is given by z, defined as

$$z = z_i + \beta_1 \cdot v_1 + \beta_2 \cdot v_2, \tag{20}$$

where z_i is the i-th objective being minimized ($i = 1, \ldots, 4$), β_1 and β_2 are constants that penalize the violations v_1 and v_2 with respect to demand constraints (7) and maximum number of farms in period (13), respectively, which are calculated by

$$v_1 = -\sum_{i=1}^{m} \min\{0, R_i\} \tag{21}$$

and

$$v_2 = -\sum_{i=1}^{m} \min\left\{0, G - \sum_{f=1}^{F} \phi_{if}\right\}, \tag{22}$$

where

$$\phi_{if} = \begin{cases} 1, & \text{if } \sum_{j=r_{f-1}+1}^{r_f} Y_{jf} > 0 \\ 0, & \text{otherwise,} \end{cases}$$

$r_0 = 0$ and $Y_{jf} = 1$ if the farm f is harvested in plot j or 0 otherwise. If a solution is feasible, the values of v_1 and v_2 are zero and the fitness is given by the objective function value of the solution.

4.4 Selection

The process of selecting $\lambda_1 \cdot n$ (where λ_1 is the selection rate) individuals to perform the remaining steps of the GA is conducted by tournament selection, i.e., two different individuals are selected and the one that has a better fitness is chosen and is introduced to be in the crossover process which is the next operator of GA.

4.5 Crossover

The aim of this operator is to construct subsequent generations with the good characteristics that the population has, through building mechanisms of new elements based on the original population. The crossover is performed between two distinct individuals (a father and a mother), and generates two distinct individuals (child 1 and child 2). Each couple is randomly chosen from the population where a vector of dimension m is generated with each element consisting of a 0 or 1 value. For the first child if the component of this vector is 0, the genetic information comes from the first parent, otherwise the second parent. For the second child the process works in the opposite way. This type of crossover is called uniform. It is relatively easy to implement and may attain different solutions in order to exploit the search space efficiently.

The Fig. 2 schematically illustrates this operator.

The crossing of two feasible solutions may produce in an infeasible solution (with respect to demand constraints). To avoid generating many infeasible solutions, each child is tested for its feasibility. If a child is infeasible, the repair algorithm **Procedure 2** is applied to transform it into a feasible one. This ensures the method is very efficient in finding the feasible solutions in the search space.

4.6 Mutation

The mutation takes the $\lambda_2 \cdot n$ (where λ_2 is the mutation rate) worst individuals in the population. This is done to preserve the best individuals and maintain the convergence of the algorithm.

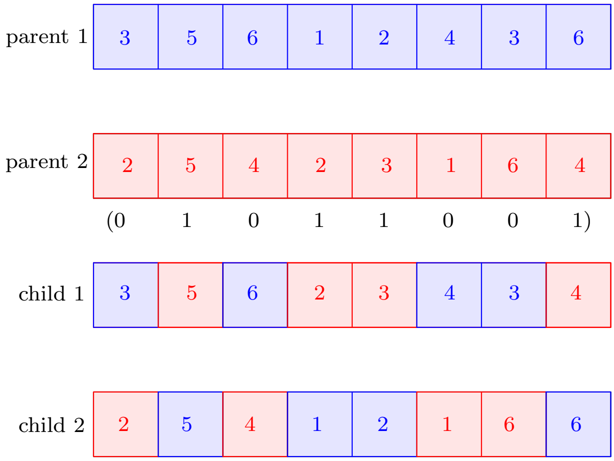

Fig. 2 Illustration of the uniform crossover

Each selected individual has a probability of 0.5 to alter its gene to its opposite value. However, this operator may remove the feasibility of a solution. In the case this happens, will be recovered by implementing repair algorithm **Procedure 2**.

The mutation occurs in the population with the following probability

$$\frac{1}{1 + e^{-10gen/g}} \tag{23}$$

where gen is the current generation. This means the probability of the mutation increases with the number of generations. In the early generations, there is little mutation, whereas at the end the probability to mutate will be close to 1. This is conduced in order to prevent the GA prematurely converging to poor quality local optima. This artificial mechanism is developed in order to ensure the most promising regions in the search space are explored.

4.7 Migration

Similar to the mutation process, the migration process aims to avoid premature convergence of the GA. An additional mechanism for inserting new elements in the population is proposed. This process is to address the trend of the search starting to stagnate at a specific location. In the migration process $\lambda_3 \cdot n$ (where λ_3 is the migration rate) randomly chosen individuals are replaced by the same number of individuals using **Procedure 4**. Here, the inserted solutions are always feasible solutions. Note that the migration only occurs in three generations, namely generations $0.5 \cdot g, 0.7 \cdot g$ and $0.9 \cdot g$, where g is the maximum allowed number of generations.

4.8 Updating and elitism

The update process is the stage where all solutions (parents$+$children) are evaluated based on their objective function values (20). The best n solutions are taken forward to the subsequent generation. The elitism is also applied to prevent the best solution $\mathcal{E}$ being alterd by the GA operators (selection, crossover, mutation and migration). Hence this solution is always transferred to the next generation. In this study, the stopping criteria are the maximum number of generations (g) generated and $0.5 \cdot g$ generations without improvement in the fitness of the $\mathcal{E}$.

Due to the computational complexity of the problem, and the desire of the decision makers (mill owners) to select from a small set of solutions, the presented algorithms aim to produce a limited number of representative Pareto efficient solutions rather than a detailed representation of the Pareto set. The setting of the harvesting goals at their ideal level ensures that the GA meta-heuristic will aim to find solutions that are close to the (unknown) exact Pareto efficient solutions via the underlying goal programming model (Jones and Tamiz 2010).

There are different ways to generate specific efficient solutions, such as Weighted Sum, Metric Tchebycheff (Bowman 1976), ε-Constrained (Ehrgott and Ruzika 2008; Haimes et al. 1971), Benson (1978), and specific algorithms for integer problems developed by Sylva and Crema (2004, 2007).

In the following section we discuss some computational results to assess the proposed solution methodology.

Algorithm 4.5 The proposed GA for the harvest plan problem

Input: problem data, $\lambda_1, \lambda_2, \lambda_3, \beta_1, \beta_2$ and g
Build $\mathcal{P}$, the initial population
$gen = 0$ and $h = 0$
while $gen \leq g \vee h \leq 0.5 \cdot g$ **do**
    Evaluate the individuals $\mathcal{P}$ and separate $\mathcal{E}$
    Apply the selection in $\mathcal{P} - \{\mathcal{E}\}$. Let $\mathcal{S}$ the $\lambda_1 \cdot n$ be selected elements
    Apply the crossover with the elements of $\mathcal{S}$. Let $\mathcal{F}$ the children. Apply **Procedure 2** to the infeasible elements of $\mathcal{F}$
    Evaluate $\mathcal{F}$ and separate the best child, $\bar{\mathcal{E}}$
    if If the fitness of the $\bar{\mathcal{E}}$ is better than fitness of the $\mathcal{E}$ **then**
        $\mathcal{E} \leftarrow \bar{\mathcal{E}}$
        $h = 0$
    else
        $h = h + 1$
    end if
    Apply mutation in $\lambda_2 \cdot n$ elements of the $(\mathcal{P} \cup \mathcal{F}) - \{\mathcal{E}\}$ with probability given by (23). If there was mutation, apply the **Procedure 2** in the mutated elements
    Apply migration in $\lambda_3 \cdot n$ elements of the $(\mathcal{P} \cup \mathcal{F}) - \{\mathcal{E}\}$ if $gen = \{0.5 \cdot g, 0.7 \cdot g, 0.9 \cdot g\}$
    If there was migration, evaluate the new individuals and rank them in the population
    Update $\mathcal{P}$ with the n best elements $\mathcal{P} \cup \mathcal{F}$
    $gen = gen + 1$
end while
Output: $\mathcal{E}$

5 Computational results

Computational experiments on this problem are performed, for smaller instances, using an exact method (via CPLEX) and, for all instances, the proposed GA. For smaller instances, the results obtained from the exact method will be used to assess the quality of solutions attained by the heuristic approach. The tests were run on a laptop with an Intel Core i7 with 8 GB of memory RAM. The GA algorithm was coded in the MATLAB software 2012 (MATLAB 2010).

In this paper, in line with the extended goal programming philosophy, we obtain a selection of points the Pareto frontier, representing a mixture from optimization to balance of the objectives. This is achieved by firstly optimizing singly the two meta-objectives (1), (2),

Table 1 Total area per instance (in ha.) demand of the sugarcane in each month for the instances

I-Plots-farms	Sugarcane demand (Ton.)				
	I-16-1	I-50-4	I-300-15	I-500-25	I-1000-35
April	2000	17,500	69,010	141,000	200,000
May	2000	11,200	96,110	149,000	290,000
June	10,000	12,845	76,216	128,000	190,000
July	6000	7000	58,700	100,000	269,005
August	7000	24,500	95,259	170,000	270,000
September	6000	11,200	77,350	159,000	260,000
October	10,000	31,500	78,268	131,000	300,000
November	2000	27,230	82,000	140,000	300,200
December	6000	18,500	79,100	120,000	290,000
Total area	332	1014	5987.8	9984.76	19,715.16

(3) and then by combining the meta-objectives (1) with (3) by using the equal weight point [$\alpha = 0.5$ in equation (12)]. Our intention is to compare these three solutions for each scenario.

Five instances (I-16-1, I-50-4, I-300-15, I-500-25, I-1000-35) are used to assess our solution method with the number of plots set to 16, 50, 300, 500 and 1000 plots respectively. Each instance has a different number of farms, representing small, medium and large mills. The details of the instances can be seen in Table 1.

The parameter values of the instances were randomly generated within a possible range. For example, the harvesting must be performed between April to December with the demand given by Table 1. Also, we provide the total area per instance (in ha.).

5.1 Experiments using the exact method (CPLEX)

Table 2 presents the computational results on all instances based on the proposed scenarios.

The optimal harvesting plans for each objective are shown in Figs. 3 and 4 (for instances I-16-1).

Figures 5, 6, 7 and 8, for instances I-50-4, show that for relatively small instances, the model is able to determine optimal harvest plan of the sugarcane and meet demand using various objectives. Interesting solutions are also found in the presence of different maturation stages of sugarcane and different number of plots.

According to Table 2, minimizing objective (2) increases the sum of absolute deviations, however the harvest can be performed in the correct period (PIU) ($t_j = t_{0_j} + t^*$). In all plots, the deviation will be less than 3 ($d_j \leq 3$). When minimizing objective (3), a smaller number of different farms being harvested in the same month is obtained at the expense of a large deviation of the harvest period from the PIU in many plots. Moreover, a longer computational[2] time is needed when compared to other cases and the exact method is also not able to solve relatively large problems. Minimizing the combination of objectives (1) and (3) can reduce the number of the different farms being harvested in the same month, however some plots still have large deviations.

Table 3 shows the experimental results when the minimizing objective (12) problem is solved with the presence of constraint (13). It can be observed that a small number of different

[2] "–": CPLEX could solve the problem.

Table 2 The average of the absolute deviation, the maximum the of the absolute deviation, the number of the absolute deviation greater than 2; sum of the absolute deviation; the average of the number of the different farms being harvested per month and CPU time spent to solve the problem (1)–(11) using the objectives: (1), (2) and (3) for all instances in Table 1

Instances I-Plots farms	Area (ha)	Objective	Average |deviation|	Maximum |deviation|	% plots with |deviation| > 2	Sum of |deviation|	Average of the number of farms harvested per month	CPU time (s)
I-16-1	332.00	(1)	1.37	5	18.75%	22	1.0	0.11
		(2)	2.00	3	37.50%	32	1.0	0.27
I-50-4	1014.00	(1)	0.38	4	4.00%	19	2.5	0.33
		(2)	1.30	3	18.00%	65	2.9	0.97
		(3)	2.64	9	38.00%	132	1.1	10,177.75
		(1) + (3)	0.38	3	2.00%	19	2.3	1.26
I-300-15	5987.76	(1)	0.31	5	4.67%	115	13.0	1.12
		(2)	1.06	3	9.33%	317	12.8	0.64
		(3)	–	–	–	–	–	–
		(1) + (3)	0.44	5	3.67%	132	9.0	819.49
I-500-25	9984.79	(1)	0.17	4	1.20%	86	21.8	10.63
		(2)	0.97	3	5.80%	485	21.4	1.17
		(3)	–	–	–	–	–	–
		(1) + (3)	0.22	4	1.00%	108	16.6	4277.09
I-1000-35	19,715.76	(1)	0.22	5	3.00%	220	33.3	5.52
		(2)	1.00	3	6.60%	1001	33.5	3.68
		(3)	–	–	–	–	–	–
		(1) + (3)	–	–	–	–	–	–

Fig. 3 Optimal harvesting planning of the instance I-16-1 using the objective (1)

Harvest months								
Ap.	*May*	*Jun.*	*Jul.*	*Aug.*	*Sep.*	*Oct.*	*Nov.*	*Dec.*
10	13	4	6	9	1	3	11	2
	14	7	15	2		5		
		16				8		

Fig. 4 Optimal harvesting planning of the instance I-16-1 using the objective (2)

Harvest months								
Ap.	*May*	*Jun.*	*Jul.*	*Aug.*	*Sep.*	*Oct.*	*Nov.*	*Dec.*
7	10	14	6	9	3	2	11	1
		4	15	16	8	5		
		12						
		13						

Fig. 5 Optimal harvesting planning of the instance I-50-4 using the objective (1)

Farm 1	*Farm 2*	*Farm 3*	*Farm 4*
1 – 8	9 – 21	22 – 44	35 – 50

Harvest months								
Ap.	*May*	*Jun.*	*Jul.*	*Aug.*	*Sep.*	*Oct.*	*Nov.*	*Dec.*
6	14	4	16	9	25	1	17	2
7	30	31	33	18	32	11	21	3
10	36	35	38	27	39	12	23	5
13	37	45		34	41	20	26	8
15		47		46		22	40	19
43				48		24	42	
44						28	49	
						29		
						50		

farms being harvested in the same month is obtained. Based on the table, the harvesting is also conduced in the PIU or close to this period. However, for this scenario, the exact method is not able to deal with the large problems that represent medium to large Brazilian farms.

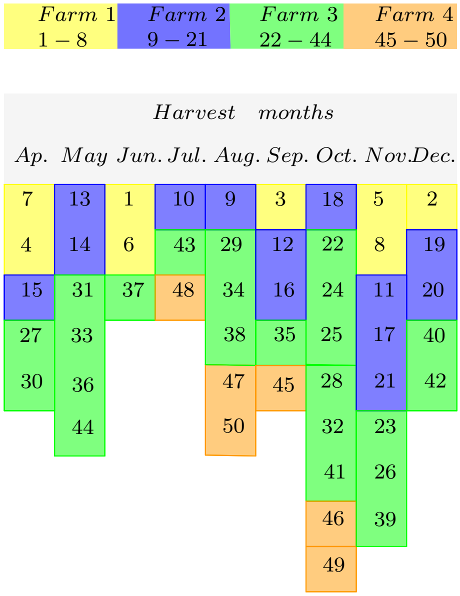

Fig. 6 Optimal harvesting planning of the instance I-50-4 using the objective (2)

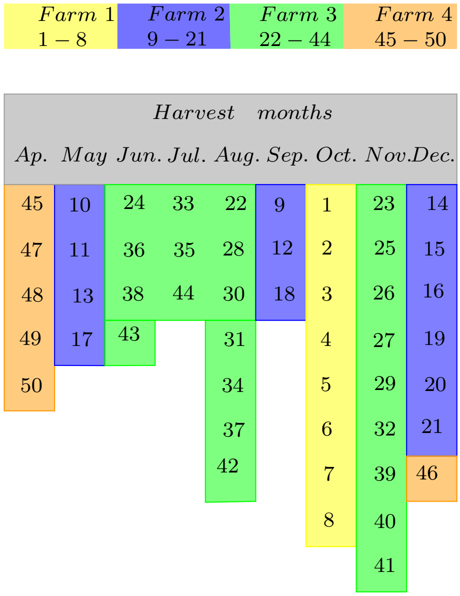

Fig. 7 Optimal harvesting planning of the instance I-50-4 using the objective (3)

5.2 Experiments using the GA

In previous experiments the exact method was used to generate an optimal harvest schedule for this problem. For minimizing objective (1), the exact method is able to solve all instances in a relatively short time. However, the exact method cannot solve minimizing the objective (3) problem due to memory issue. Therefore, the GA is proposed to overcome the limitations of the exact method. This section presents the experiments of the GA using the same instances

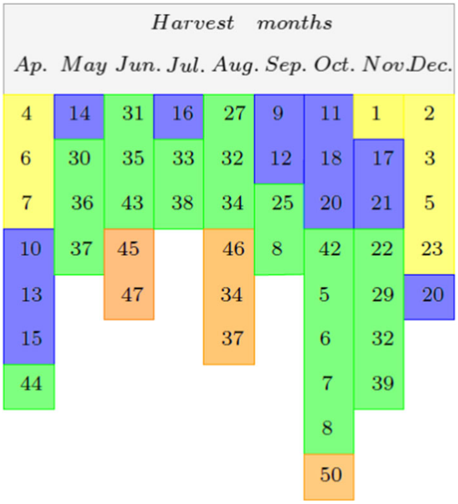

Fig. 8 Optimal harvesting planning of the instance I-50-4 using the objective (12)

Table 3 The absolute deviation, the maximum the of the absolute deviation, the number of the absolute deviation greater than 2; sum of the absolute deviation and CPU time spent to solve the proposed model, using the objective (12) and constraint (13)

Instances I-Plots farms	G value	Average |deviation|	Maximum |deviation|	% plots with |deviation| >2	Sum of |deviation|	Average of the number of farms harvested per month	CPU time (s)
I-16-1	1	1.5	3	12.50%	24	1	0.19
I-50-4	3	0.38	3	4.0%	19	2.4	0.61
I-300-15	8	–	–	–	–	–	–
I-500-25	15	–	–	–	–	–	–
I-1000-35	20	–	–	–	–	–	–

Table 4 Parameters used in GA

n	g	λ_1	λ_2	λ_3	β_1	β_2
120	100	0.80	0.05	0.20	100	100

used in previous experiments. The parameters used in the GA for all instances are presented by Table 4.

To assess the consistency of the proposed heuristic method, for each instance, the GA was executed 20 times with the average results are presented in Table 5. The structure of the table is similar to the one of Table 2

Based on the results, it can be noted that GA produces good solutions for all instances in an acceptable computational time. The computational time increases linearly with k. When k is set to 1000 lots (a large farm), the GA requires less than 20 min to solve the problem. On the other hand, the exact method runs faster than the GA in solving the problems solely minimizing the sum of deviations. However, the exact method experiences difficulties solving the minimizing objective z_2 problem. For $k = 50$, for example, the exact method took almost 3 h to solve the problem. Furthermore, the exact method was not able to solve instances with $k > 50$.

Table 5 The average of the absolute deviation, the maximum the of the absolute deviation, the number of the absolute deviation greater than 2; sum of the absolute deviation; the average of the number of the different farms being harvested per month and CPU time spent to solve the problem (1)–(11) using the objectives: (1), (2) and (3) for all instances in Table 1 by using the genetic algorithm

Instances I-Plots farms	Objective	Average |deviation|	Maximum |deviation|	% plots with |deviation| > 2 (%)	Sum of |deviation|	Average of the number of farms harvested per month	CPU time (s)
I-16-1	(1)	1.42	5.2	18.95	22.2	1.0	5.89
	(2)	2.08	3.1	37.89	32.4	1.0	5.27
I-50-4	(1)	0.54	4.6	7.5	27.2	2.6	118.61
	(2)	0.73	4.0	15.2	36.4	2.7	117.76
	(3)	1.74	8.3	29.1	86.7	1.8	169.39
	(1) + (3)	0.47	4.6	4.2	23.5	2.5	159.53
I-300-15	(1)	0.43	5.5	6.2	130.9	12.3	233.7
	(2)	0.49	4.7	5.6	148.1	11.8	214.5
	(3)	0.57	5.7	4.4	172.1	9.4	296.7
	(1) + (3)	0.44	5.3	5.7	132.4	11.8	209.4
I-500-25	(1)	0.21	4.5	1.7	103.9	21.4	478.4
	(2)	0.24	4.5	2.1	120.5	20.9	332.9
	(3)	0.57	7.9	4.8	285.9	16.1	579.9
	(1) + (3)	0.21	4.0	1.4	107.0	20.3	568.7
I-1000-35	(1)	0.21	5.5	3.8	235.4	33.2	879.6
	(2)	0.43	4.0	1.7	395.0	27.1	1022.7
	(3)	0.52	5.5	2.5	484.5	23.1	1299.8
	(1) + (3)	0.21	5.2	3.7	239.1	32.7	1051.7

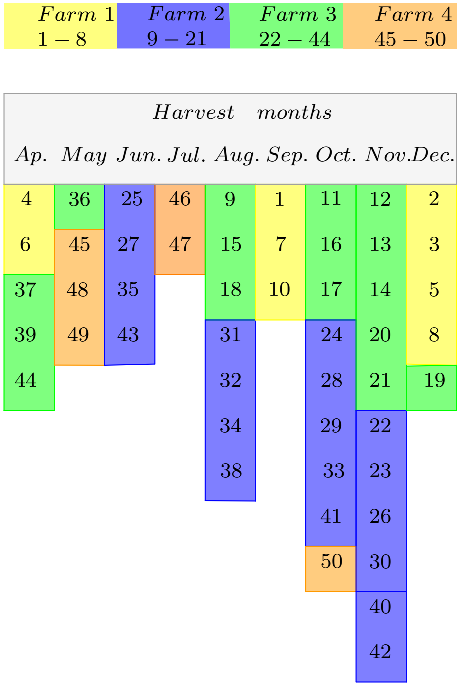

Fig. 9 Optimal harvesting planning of the instance I-50-4 using the GA and the the objective (3)

Another aspect to be highlighted is that a good quality of heuristic solutions is found, mainly due to the initial solution generated using the four constructive procedures. When only objective z_1 is taken into account, the GA yields an error of 0.90, 43.1, 13.8, 20.8 and 7.0% for instances with $k = 16$, 50, 300, 500 and 1000 plots respectively.

Based on the best solutions over the 20 runs, GA produces an error of 0.52, 10.1, 5.2, 6.9 and 4.1% for the same instances. In general the method provides good results and runs fast, thus demonstrating the value of a meta-heuristic for this type of hard to solve problem.

The GA algorithm was also able to provide feasible solutions to the problem of minimizing the movement of the machines for the instances with $k > 50$ in a reasonable computing time. In terms of the quality of the solutions, the solution obtained from the GA for $k = 50$ can be compared to the optimal one. In this case, the results of the GA are as follows. Based on the average results, 1.8 farms are harvested in a period with the sum of the deviations equal to 86.7, whereas based on the best results, 1.1 farms must be harvested in a month (as seen in Fig. 6) with the sum of deviations equal to 132.

This shows that the GA has a little difficulty in producing solutions with a small z_2 as the constructive heuristics focus on minimizing the sum o deviations. An example solution with a small value of z_2 obtained by GA is presented in Fig. 9, where the average number of farms to be harvested per period is 1.7 with the sum of deviations is equal to 69.

With respect to the problem of minimizing objective (12), good solutions are obtained using the heuristic method. It can be noted that the average solutions of the GA are relatively close to the optimal ones. For example, when $k = 50$, the average deviation obtained by the exact method and the GA is 0.38 and 0.49, respectively whereas the number of plots it deviations larger than 2 is 4 and 1.4% respectively. It is also highlighted that in the optimal solution for this instance, there are 2.4 farms harvested in the same period whereas the GA produces 2.5.

Table 6 The absolute deviation, the maximum the of the absolute deviation, the number of the absolute deviation greater than 2; sum of the absolute deviation and CPU time spent to solve the proposed model, using the objective (12) and constraint (13) by using the genetic algorithm

Instances I-Plots farms	G value	Average |deviation|	Maximum |deviation|	% plots with |deviation| > 2 (%)	Sum of |deviation|	Average of the number of farms harvested per month	CPU time (s)
I-16-1	1	1.5	3.1	12.6	24.2	1	5.3
I-50-4	3	0.49	4.0	5.4	24.8	2.5	161.2
I-300-15	8	0.62	5.6	4.2	169.8	7.5	286.4
I-500-25	15	0.20	4.2	1.4	104.6	14.6	502.9
I-1000-35	20	0.22	5.1	3.6	235.2	19.3	1085.1

For instances with $k = 300$, 500 and 1000 the GA produces a relatively small deviations, which are on average less than one month. The small percentage of the number of plots with deviations greater than 2 months is also obtained whilst satisfying all the constraints (13).

6 Conclusions and perspectives

This paper proposes a multiobjective sugarcane harvest scheduling model and solution algorithm that allows mill owners to effectively and efficiently manage their harvesting operations over a multi-year planning horizon. The methodology ensures at the same time that the harvest of each plot is as close as possible to its optimal maturation period and reduces the handling of machines. As noted, these goals are conflicting with each other, i.e., the enhancement of a goal entails a worsening of the other and vice-versa. These objectives can be balanced, and hence an intermediate solution for minimizing both goals can be achieved. This paper demostrates application of this model on real data, and indicates the current limitation of exact optimization techniques to small scale farms. This can be explained by the complex nature of the mathematical model for this problem that involves many binary variables and has a very loose linear relaxation (Table 6).

To overcome this drawback, and to solve the actual large size instances, a genetic algorithm based on four constructive heuristics is developed, implemented and compared with an exact method solution. The four constructive heuristics have different underlying philosophies of construction in order to enhance the subsequent search process over the generations. The results are quite favorable, since this procedure can obtain feasible solutions that are very close to the optimum problem and solve instances where it was not possible to determine any viable solution in a timely manner with the exact method. Furthermore, the algorithm has a very low computational cost, and can provide workable solutions for instances of 1000 lots in less than 20 min of computing time. In summary, the proposed model and solution method are applicable in realistic cases, hence helping farm managers in their decision making for this key agricultural product that has importance for the Brazilian economy.

For future research, it is worthwhile investigating other constructive heuristics to determine the Pareto frontier for this problem (e.g. Non-dominated Sorting Genetic Algorithm—NSGA). The enhancement of this model can also be considered by calculating the deviations based on the area where a plot is located. Moreover, applications to harvesting other crops may be performed by using the ideas and procedures presented in this work.

Acknowledgements The authors wish to thank the Brazilian foundations FAPESP (Grant Nos. 2014/01604-0 and 2014/04353-8), CNPq (Grant No. 303267/2011-9), PROEPE (UNESP) and FUNDUNESP. Also, to the Institute of Mathematics, Statistics and Scientific Computation belonging to UNICAMP and FAPESP (Grant 2013/06035-0), for their financial support. The authors also wish to thank the two anonymous referees whose comments helped shape the final version of this paper.

References

Bagdon, B. A., Huang, C. H., & Dewhurst, S. (2016). Managing for ecosystem services in northern arizona ponderosa pine forests using a novel simulation-to-optimization methodology. *Ecological Modelling*, *324*, 11–27. doi:10.1016/j.ecolmodel.2015.12.012. http://www.sciencedirect.com/science/article/pii/S0304380015005803.

Baraku, B., Shahu, E., & Mulliri, J. (2015). Goal programming as a method utilized in production planning at the farm level. *International Journal of Ecosystems and Ecology Science-IJEES*, *5*(3), 447–452.

Benson, H. (1978). Existence of efficient solutions for vector maximization problems. *Journal of Optimization Theory and Applications*, *26*(4), 569–580.

Bowman, V. J. (1976). On the relationship of the Tchebycheff norm and the efficient frontier of multiple-criteria objectives. In H. Thieriez (Ed.), *Multiple criteria decision making, lecture notes in economics and mathematical systems* (Vol. 130). Berlin: Springer.

Calija, V., Higgins, A. J., Jackson, P. A., Bielig, L. M., & Coomans, D. (2001). An operations research approach to the problem of the sugarcane selection. *Annals of Operations Research*, *108*(1), 123–142. doi:10.1023/A:1016054911470.

Cardozo, N. P., & Sentelhas, P. C. (2013). Climatic effects on sugarcane ripening under the influence of cultivars and crop age. *Scientia Agricola*, *70*, 449–456.

Conab: Companhia nacional de abstecimento. acompanhamento da safra brasileira. (2016). Cana de açúcar. oservatório agrícola. levantamento de agosto http://www.conab.gov/OalaCMS. Accessed 25 Nov 2016.

da Silva, A. F., Marins, F. A. S., & Dias, E. X. (2015). Addressing uncertainty in sugarcane harvest planning through a revised multi-choice goal programming model. *Applied Mathematical Modelling*, *39*(18), 5540–5558. doi:10.1016/j.apm.2015.01.007. http://www.sciencedirect.com/science/article/pii/S0307904X15000086.

Deb, K. (2001). *Multi-objective optimization using evolutionary algorithms. Wiley-interscience series in systems and optimization*. Chichester: Wiley.

Demirci, M., & Bettinger, P. (2015). Using mixed integer multi-objective goal programming for stand tending block designation: A case study from turkey. *Forest Policy and Economics*, *55*, 28–36. doi:10.1016/j.forpol.2015.03.007. http://www.sciencedirect.com/science/article/pii/S1389934115000568.

Ehrgott, M., & Ruzika, S. (2008). An improved ε-constraint method for multiobjective programming. *Journal of Optimization Theory and Applications, 138*, 375–396.

Florentino, H. O., & Pato, M. V. (2014). Bi-objective genetic approach for selection of sugarcane varieties. *Journal of the Operational Research Society*, *65*(6), 842–854. doi:10.1057/jors.2013.21.

Florentino, H. O., Pato, M. V., Jones, D., & Cantane, D. R. (2015). *Biomass production and uses, chap. Production and management of sugarcane biomass process optimization*. InTech.

Gómez, T., Hernández, M., Molina, J., León, M. A., Aldana, E., & Caballero, R. (2011). A multiobjective model for forest planning with adjacency constraints. *Annals of Operations Research*, *190*(1), 75–92. doi:10.1007/s10479-009-0525-4.

González-Pachón, J., & Romero, C. (2001). Aggregation of partial ordinal rankings: An interval goal programming approach. *Computers and Operations Research*, *28*(8), 827–834. doi:10.1016/S0305-0548(00)00010-1. http://www.sciencedirect.com/science/article/pii/S0305054800000101.

Haimes, Y. Y., Lasdon, L. S., & Wismer, D. A. (1971). On a bicriterion formulation of the problems of integrated system identification and system optimization. *IEEE Transactions on Systems*, *3*, 296.

Higgins, A. J. (1999). Optimizing cane supply decisions within a sugar mill region. *Journal of Scheduling*, *2*(5), 229–244. doi:10.1002/(SICI)1099-1425(199909/10)2:5<229::AID-JOS29>3.0.CO;2-L.

Higgins, A. J., & Postma, S. (2004). Australian sugar mills optimise siding rosters to increase profitability. *Annals of Operations Research*, *128*(1), 235–249. doi:10.1023/B:ANOR.0000019107.68291.36.

Hofsetz, K., & Silva, M. A. (2012). Brazilian sugarcane bagasse: Energy and non-energy consumption. *Biomass and Bioenergy*, *46*, 564–573. doi:10.1016/j.biombioe.2012.06.038. http://www.sciencedirect.com/science/article/pii/S096195341200284X. International Conference on Lignocellulosic ethanol.

Jones, D., Mirrazavi, S., & Tamiz, M. (2002). Multi-objective meta-heuristics: An overview of the current state-of-the-art. *European Journal of Operational Research*, *137*(1), 1–9. doi:10.1016/S0377-2217(01)00123-0. http://www.sciencedirect.com/science/article/pii/S0377221701001230.

Jones, D., & Tamiz, M. (2010). *Practical goal programming, international series in operations research and management science* (141st ed., Vol. 141). Berlin: Springer.

Landers, I. N. (2007). Tropical crop livestock systems in conservation agriculture: the Brazilian experience. Food and Agriculture Organization of the United Nations.

López-Milán, E., & Plà-Aragonés, L. M. (2013). A decision support system to manage the supply chain of sugar cane. *Annals of Operations Research*, *219*(1), 285–297. doi:10.1007/s10479-013-1361-0.

López-Milán, E., & Plà-Aragonés, L. M. (2014). A decision support system to manage the supply chain of sugar cane. *Annals of Operations Research*, *219*(1), 285–297. doi:10.1007/s10479-013-1361-0.

Magalhães, P., & Braunbeck, O. A. (2014). Trm: Agriculture component (pp. 897–908). doi:10.5151/BlucherOA-Sugarcane-SUGARCANEBIOETHANOL-75. URL openaccess.blucher.com.br/article-details/trm-agriculture-component-19294?articles/details/155

Martins, I., Ye, M., Constantino, M., da Conceição Fonseca, M., & Cadima, J. (2014). Modeling target volume flows in forest harvest scheduling subject to maximum area restrictions. *TOP*, *22*(1), 343–362. doi:10.1007/s11750-012-0260-x.

MATLAB: Version 7.10.0 (R2010a). The MathWorks Inc., Natick, Massachusetts (2010)

Paiva, R. P. O., & Morabito, R. (2008). An optimization model for the aggregate production planning of a Brazilian sugar and ethanol milling company. *Annals of Operations Research*, *169*(1), 117–130. doi:10.1007/s10479-008-0428-9.

Picoli, M. C. A., Lamparelli, R. A. C., Sano, E. E., & Rocha, J. V. (2014). The use of ALOS/PALSAR data for estimating sugarcane productivity. *Engenharia Agrícola*, *34*, 1245–1255.

Prišenk, J., & Turk, J. (2015). A multi-goal mathematical approach for the optimization of crop lanning on organic farms: A slovenian case study. *Pakistan Journal of Agricultural Sciences*, *4*, 971–979.

Prišenk, J., Turk, J., Rozman, Č., Borec, A., Zrakić, M., & Pažek, K. (2014). Advantages of combining linear programming and weighted goal programming for agriculture application. *Operational Research*, *14*(2), 253–260. doi:10.1007/s12351-014-0159-4.

Ramesch, P., & Mahadevaswamy, M. (2000). Effect of formative phase drought on different classes of shoots, shoot mortality, cane attributes, yield and quality of four sugarcane cultivars. *Journal of Agronomy and Crop Science*, *185*, 249–258.

Romero, C. (2004). A general structure of achievement function for a goal programming model. *European Journal of Operational Research*, *153*(3), 675–686. doi:10.1016/S0377-2217(02)00793-2. EURO Young Scientists.

Rudorff, B. F. T., Aguiar, D. A., Silva, W. F., Sugawara, L. M., Adami, M., & Moreira, M. A. (2010). Studies on the rapid expansion of sugarcane for ethanol production in Sao Paulo State (Brazil) using landsat data. *Remote Sensing*, *2*, 1057–1076.

Sharma, D. K., Ghosh, D., & Alade, J. A. (2003). Management decision-making for sugarcane fertilizer mix problems through goal programming. *Journal of Applied Mathematics and Computing*, *13*(1), 323–334. doi:10.1007/BF02936095.

Supsomboon, S., & Niemsakul, J. (2014). A linear programming for sugarcane cultivation and harvest planning with cane survival rate. *Agricultural Engineering International*, *16*(4), 207–216.

Sylva, J., & Crema, A. (2004). A method for finding the set of non-dominated vectors for multiple objective integer linear programs. *European Journal of Operational Research*, *158*, 46–55.

Sylva, J., & Crema, A. (2007). A method for finding well-dispersed subsets of non-dominated vectors for multiple mixed integer linear programs. *European Journal of Operational Research*, *180*, 1011–1027.

Vianna, M. D. S., & Sentelhas, P. C. (2014). Simulação do risco de deficit hídrico em regiões de expansão do cultivo de cana-de-açúcar no Brasil. *Pesquisa Agropecuária Brasileira*, *49*, 237–246.

Weintraub, A., & Murray, A. T. (2006). Review of combinatorial problems induced by spatial forest harvesting planning. *Discrete Applied Mathematics*, *154*(5), 867–879. doi:10.1016/j.dam.2005.05.025. http://www.sciencedirect.com/science/article/pii/S0166218X05003124. IV ALIO/EURO Workshop on Applied Combinatorial OptimizationIV ALIO/EURO Workshop on Applied Combinatorial Optimization.

Worldwatch institute: Vision for a Sustainable World. http://www.worldwatch.org/biofuels-transportation-selected-trends-and-facts/. Accessed: 18 Nov 2015.

Yirsaw, T., Woldetsadik, K., & Workneh, T. (2000). Effect of harvest time on quality of sugar cane cultivars. *Advanced Materials Research, 824*, 293–300.

Zengin, H., Asan, U., & Destan, S. (2015). Modeling harvest scheduling in multifunctional planning of forests for longterm water yield optimization. *Natural Resource Modeling*, *28*(1), 59–85.