Computers & Industrial Engineering 145 (2020) 106497

Contents lists available at ScienceDirect

Computers & Industrial Engineering

journal homepage: www.elsevier.com/locate/caie

A multi-objective approach for supporting wine grape harvest operations

Mauricio Varas[a,*], Franco Basso[b,e], Sergio Maturana[c], David Osorio[d], Raúl Pezoa[d]

[a] *Centro de Investigación en Sustentabilidad y Gestión Estratégica de Recursos, Facultad de Ingeniería, Universidad del Desarrollo, Santiago, Chile*
[b] *School of Industrial Engineering, Pontificia Universidad Católica de Valparaíso, Valparaíso, Chile*
[c] *Industrial and Systems Engineering Department, Pontificia Universidad Católica de Chile, Santiago, Chile*
[d] *Escuela de Ingeniería Industrial, Universidad Diego Portales, Santiago, Chile*
[e] *Instituto Sistemas Complejos de Ingeniería (ISCI), Santiago, Chile*

ARTICLE INFO

Keywords:
Wine industry
Grape harvest
Multi-objective optimization
Interactive methods

ABSTRACT

In this paper, we present a novel multi-objective mixed-integer linear programming model to support wine grape harvesting. The proposed model considers the opposing nature of operational cost minimization and grape quality maximization, subject to several constraints, such as grape requirements and routing decisions. Based on the operations of a winery we worked with, we develop a negotiation protocol that can lead to an agreed final harvest schedule. The protocol includes an initial Pareto optimal solution obtained through the augmented weighted Tchebycheff method. Then, the solutions are presented to the two decision-makers and, if no agreement is reached, we conduct an iterative process, which includes finding Pareto optimal solutions in a neighborhood using the augmented ϵ-constraint method. Finally, we choose, within this set, the solution following a substitution rate criteria. We illustrate our procedure using an educational example.

1. Introduction

The Chilean wine industry has greatly developed in the last few decades. It currently exports over 1800 million US dollars and plays an important role in the Chilean economy (Mora, 2019). Although the Chilean wine industry only contributes about 1% of the Chilean gross domestic product, it has a significant role in positioning Chile as a brand around the world (Egan & Bell, 2002). In fact, in the last decades, Chilean wineries have turned from local-consumption-focused companies to export-focused companies, entering into more extensive and more competitive markets (Overton & Murray, 2011). Thus, local wineries need to improve their efficiency and productivity along the whole wine supply chain to remain competitive in the global market.

This paper focuses on the wine grape harvest stage, which is the first one of the wine supply chain. The grape harvest in Chile is generally carried out from the end of February until the end of April of each year (Lima, 2015). Wine grape harvest planning consists of deciding how, when, where, and how much to harvest. For managerial reasons, the soil is divided into blocks, which are portions of land with similar characteristics in terms of composition and quality of grapes. In order to generate a good harvesting schedule, both the oenologist and the field manager must participate since they tend to have quite different perspectives.

In this paper, we address the opposing objectives of the oenologist and the field manager in the grape harvest scheduling problem. On the one hand, the oenologist seeks to maximize the quality of the harvested grapes, while on the other, the field manager attempts to minimize the operational costs of the harvest. Moreover, these two opposing objectives must consider operational constraints, such as resource availability. For example, each block has a feasible harvest time window around an optimal specific day. The more the harvest deviates from the optimal day, the more the grapes lose quality. In this context, the oenologist has incentives to propose a harvesting plan in which every block is harvested on its optimal day, but this may be infeasible or prohibitively expensive considering a limited number of workers or machinery.

The contribution of this paper is twofold. First, we develop a novel multi-objective mixed-integer linear programming model to support wine grapes harvesting. The model considers the opposing nature of the minimization of operational costs and the maximization of grape quality, subject to several constraints, such as requirements for the grapes, winery cellar reception capacities, and routing decisions within a period. Second, based on the real operations of one of the three largest wineries in Chile, we develop a negotiation protocol that can help decision-makers to reach an agreement on the final harvest schedule. The negotiation protocol uses the augmented weighted Tchebycheff method

* Corresponding author.
E-mail addresses: mavaras@udd.cl (M. Varas), francobasso@gmail.com (F. Basso), smaturan@ing.puc.cl (S. Maturana), david.osoriom@mail.udp.cl (D. Osorio), raul.pezoa@udp.cl (R. Pezoa).

https://doi.org/10.1016/j.cie.2020.106497
Received 30 May 2019; Received in revised form 17 January 2020; Accepted 20 April 2020
Available online 11 May 2020

to compute a first optimal Pareto solution close to an ideal solution. This solution is presented to the two decision-makers and, if no agreement is reached, we conduct an iterative process that includes finding Pareto optimal solutions in a neighborhood using the augmented ϵ-constraint method. In each iteration, we choose within these Pareto optimal solutions, the one with the substitution rate closest to one.

The rest of this paper is organized as follows. Section 2 reviews relevant literature. Section 3 describes the multi-objective optimization methodology. Section 4 presents the model for the problem at hand. Section 5 describes the negotiation protocol and Section 6 presents an illustrative example. Finally, Section 7 presents the conclusions.

2. Literature review

Since the 1940s, operations research has been successfully used to improve productivity and efficiency in various industrial problems. In particular, it has helped improve decisions in the management of natural resources; for example, in agriculture (Souza & Gomes, 2015); in aquaculture (Mesquita, Murta, Paias, & Wise, 2017); in mining (Moreno, Rezakhah, Newman, & Ferreira, 2017); and in the forest industry (Alvarez, Espinoza, Maturana, & Vera, 2020; Maturana, Pizani, & Vera, 2010; Troncoso, D'Amours, Flisberg, Rönnqvist, & Weintraub, 2015). The wine industry has also incorporated the use of advanced analytical methods. However, its adoption has been slower than in other sectors due mainly to two factors. First, as stated by Morande and Maturana (2010), winemakers tend to see themselves more as artists than technicians, so they are usually reluctant to use quantitative methods to support decision making. Second, since advanced analytical models require both mathematical and computational knowledge for the use of these tools, many practitioners tend to resist their application for decision making (Garcia et al., 1990).

Notwithstanding these obstacles, in recent decades, there has been significant development of mathematical tools to increase efficiency and modernize the wine industry due, mainly, to the increase in complexity of the decisions involved as a result of the globalization of all the business areas in this industry (Hussain, Cholette, & Castaldi, 2008).

2.1. Wine supply chain

According to Basso, Guajardo, and Varas (2020), the wine supply chain, shown in Fig. 1, consists essentially of four stages: (i) wine grapes growth and harvest, (ii) wine manufacturing, (iii) bottling, labeling and packaging, and (iv) distribution. For a detailed description of each stage, we refer the reader to (Petti et al., 2006).

The wine supply chain has been researched from different perspectives. For example, Garcia, Marchetta, Camargo, Morel, and Forradellas (2012) propose a logistics benchmarking framework for the wine industry, which was applied to several wineries from Mendoza in Argentina. Ting, Tse, Ho, Chung, and Pang (2014) propose a decision support system of supply chain quality sustainability to support managers in food manufacturing firms for better planning their logistics in order to maintain the quality and safety of food products. They also conducted a case study of a Hong Kong red wine company.

Environmental issues have generated much research on how to make the wine supply chain cleaner. For example, Valenzuela and Maturana (2016) propose a three-dimensional performance measurement system (SMD3D) that encompasses three key dimensions: sustainable, temporal, and spatial. The in-depth interviews conducted with managers of 50 wine companies in Chile confirmed the importance they assign to sustainability, having formally defined it in their strategic plan. More recently, Harris, Rodrigues, Pettit, Beresford, and Liashko (2018) examine the impact of different wine distribution alternatives on carbon and sulfate emissions. On the other hand, Ponstein, Ghinoi, and Steiner (2019) use a life cycle assessment methodology to investigate greenhouse gas emissions in the wine supply chain in Finland.

2.2. Operation research models in the wine industry

There have been relevant contributions on the use of operations research tools in the different stages of the wine supply chain (Moccia, 2013). At the grape harvest stage, which is the subject of our research, Ferrer, Mac Cawley, Maturana, Toloza, and Vera (2008) propose a mixed-integer linear optimization model (MILP) that incorporates route decisions and the location of workers for scheduling the grape harvest. The authors incorporate a novel loss function that represents the decrease in quality by advancing or delaying the harvest with respect to its optimal day. Arnaout and Maatouk (2010) modify the model proposed by Ferrer et al. (2008), adding a new constraint that forces no inactive days between the start and the end of the harvest of a block. The authors also develop a novel heuristic to solve large-sized instances that significantly outperforms the Branch-and-Bound algorithm used in Ferrer et al. (2008). Finally, Bohle, Maturana, and Vera (2010) present an extension of the model proposed by Ferrer et al. (2008), addressing the uncertainty of crop productivity by using the robust optimization approach of Bertsimas and Sim (2004).

The manufacturing stage has been the subject of less research than the other stages, due to its complexity. However, Cakici et al. (2006)

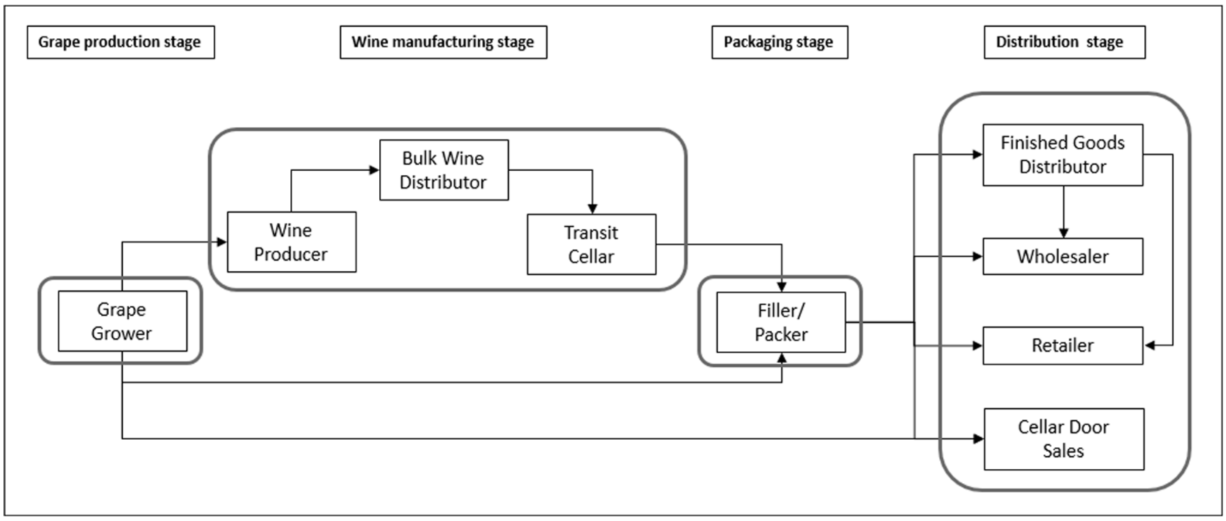

Fig. 1. Wine industry supply chain (Basso et al., 2020).

propose a MILP model to route flows through a network of tanks and pipes at E. & J. Gallo Winery, seeking to minimize wine damage and optimize resources. Due to the size of the piping networks, the authors propose a heuristic procedure to solve full-scale problems. Also, Palmowski and Sidorowicz (2018) consider a dynamic programming optimization approach to model the assignation of grapes to pressing tanks in order to obtain the highest possible income from produced wine. Tests on real data show that the proposed algorithm generates a higher payoff than the manual assignment method.

The packaging stage, on the other hand, has been much more researched. For example, Cholette (2009) uses a two-stage stochastic linear program with fixed recourse to study the problem of product misallocation. The model maximizes the expected profit over a distribution of demand scenarios. Basso and Varas (2017) propose a MILP model for the bottling scheduling problem. The resulting formulation considers specific operational constraints, but excessive computational times precludes its use in real instances. The authors also propose a greedy heuristic solution approach that can handle real-case scenarios in a reasonable amount of time. Varas, Maturana, Cholette, Mac Cawley, and Basso (2018) extend the analysis of Cholette (2009) by proposing a multi-stage stochastic programming model of the same problem. The authors analyze the impact on the performance of wine production planning when the labeling of bottled wine is delayed. Varas, Basso, Lüer-Villagra, Mac Cawley, and Maturana (2019) propose a $(s-1, s)$ strategy to manage the inventory of premium wines, which minimizes the steady-state expected values of work in process, overage, and underage costs. They developed a heuristic that solves a newsvendor type of problem for each end product. Basso et al. (2020) propose a horizontal collaborative approach for the wine bottling scheduling problem formulated in Basso and Varas (2017). The methodology analyzes the impact of sharing production lines among several wineries and concludes that delays can be reduced up to 56.9%.

Finally, at the distribution stage, Cholette (2007) develops a MIP model to qualify partnerships between wineries and distributors. The author applies this model in a real setting, recommending 31 winery-distributor matches out of 675 possibilities. Mac Cawley (2014) discusses the relationship between quality degradation and the shipping temperature of the wine. The author shows that the risk of extreme temperature exposure for the shipments is minimized during the Northern hemisphere winter. The author also notes that the transshipment phase is the most dangerous one of the distribution chain.

2.3. Multiobjective optimization applications

Multiobjective optimization has been used to help decision-makers to deal with many different problems. For example, Salamati-Hormozi, Zhang, Zarei, and Ramezanian (2018) propose a generalized mixed-integer linear production planning problem with a multi-period and multi-item specification in a make-to-order manufacturing system. This model seeks to assign the customers' orders to its subsidiary companies so that it minimizes the total cost as well as the maximal production utilization, leading to a fair allocation of production loads. Due to the difficulty of finding the set of Pareto solutions, an ϵ-constraint method is used for small-sized problems, and three metaheuristic algorithms are used for large-sized problems. Computational experiments show the feasibility of the approach.

Rezaei-Malek, Tavakkoli-Moghaddam, Zahiri, and Bozorgi-Amiri (2016) propose an integrated model that determines the optimal location-allocation and distribution plan, along with the best ordering policy for renewing the stocked perishable commodities that are prepositioned in a pre-disaster phase. Since it is impossible to know beforehand when a disaster will strike, and stocked perishable commodities need to be renewed, the model seeks to minimize two objectives: the average weighted response times and the total operational cost at the pre-disaster phase, and the penalty costs of unmet demand and unused commodities at a post-disaster phase. The reservation level Tchebycheff procedure (RLTP) is used to provide good solutions to decision-makers interactively. A case study in Iran is presented.

Bilir, Ekici, and Ulengin (2017) propose a multi-objective supply chain network optimization model to analyze the impact on customer demand by supply chain decisions. The model considers three objectives: profit maximization, sales maximization, and supply chain risk minimization. The model is applied to a real-world problem, which is solved as single and multi-objective models. The results are compared, and a sensitivity analysis is conducted to test the applicability of the model.

The application of multi-objective optimization can also be found in various natural resources industries. For example, in the forestry industry, Palma and Vergara (2016) develop an optimization model considering three objectives: production costs, waste, and overproduction. The authors solve problems considering uncertainty in the preferences of each objective. In this same industry, Palma and Nelson (2010) consider the model proposed by Johnson and Scheurman (1977), taking two objectives into account, and adding uncertainty to the importance of each of them. More recently, Vafaeenezhad, Tavakkoli-Moghaddam, and Cheikhrouhou (2019) describe a multi-objective linear programming model they developed for a multi-echelon, multi-product, multi-period supply chain planning problem. This model considers many dimensions of sustainable development simultaneously by using six objective functions. For solving this model, an improved version of the augmented-constraint method (AUGMECON2) is used. The authors also provide computational results using the proposed model. Finally, an application to the paper industry is presented and discussed.

In the food industry, Bottani, Murino, Schiavo, and Akkerman (2019) proposed a bi-objective mixed-integer programming formulation for the Resilient Food Supply Chain Design (RFSCD) problem, which is the problem of designing a food supply chain that is resilient enough to ensure business operations continuity in the event of risks or disruptions. The objectives were to maximize the total profit over one year and to minimize the total lead time of the product along the supply chain. They solve the model using an Ant Colony Optimization (ACO) algorithm.

In the wine industry, Varsei and Polyakovskiy (2017) propose a multi-objective optimization model for designing the supply chain considering three objectives: costs, CO_2 emissions, and social impact. To the best of our knowledge, this is the only application of the multi-objective optimization approach in the wine industry.

Finally, environmental issues have also been incorporated in multi-objective models. For example, Miranda-Ackerman, Azzaro-Pantel, and Aguilar-Lasserre (2017) present a methodology based on life cycle assessment, multi-objective optimization solved using genetic algorithms, and multiple-criteria decision-making tools for helping design food supply chain networks. The approach is illustrated and validated on an orange juice supply chain case study.

Our paper is related to the work presented in Bohle et al. (2010) with a significant difference: in our work, we focus on the inherent conflict between the oenologist and the field manager that arise when developing a harvest schedule. The underlying hypothesis of our work is that the improper management of these conflicting objectives could lead to inferior solutions in terms of both quality and costs. In this context, the use of a single-objective function that integrates the concerns of both decision-makers requires knowing the cost of grape quality degradation when deviating from the optimal harvesting day. In practice, however, this cost is quite difficult to estimate because the final price of a bottle depends not only on the grape quality, but also on several other factors that are subject to multiple sources of uncertainty, e.g., promotions and demand level, and unexpected events (Mac Cawley, 2014). Moreover, a single-objective solution approach does not include human interaction, which could be unrealistic in this industry. So, to tackle all the above drawbacks, in this paper, we propose a multi-objective formulation, a ϵ-constraint solution approach, and a negotiation protocol to cope with the conflicting objectives.

3. A multi-objective optimization framework: some building blocks

Multi-objective optimization can be defined as an optimization problem with at least two objective functions. Consider $m > 1$ objective functions $f_1: \chi \to \mathbb{R}, \dots, f_m: \chi \to \mathbb{R}$ which maps the decision space χ to the set of real numbers $\mathbb{R}$. A multi-objective optimization problem can be written as follows (1).

$$\begin{aligned} &\text{minimize} \quad (f_1(\mathbf{x}), \dots, f_m(\mathbf{x})) \\ &\text{s. t.} \quad \mathbf{x} \in \chi \end{aligned} \tag{1}$$

In a multi-objective framework, the goal is to find a solution on which the decision-maker can agree, and that is optimal in some sense (Emmerich & Deutz, 2018). On this, the main difficulty with this type of problem is the subjectivity of the optimal solution (Baesler & Palma, 2014). In multi-objective optimization, the notion of an optimal solution is changed to non-dominated solutions. A non-dominated solution, or Pareto optimal solution, is a feasible solution for which there is no other feasible solution that improves some objective function without worsening another (Mavrotas, 2009). The set of all non-dominated solutions is called the Pareto set, which reveals the trade-off between the objectives. Therefore, computing the Pareto set and choose a solution within it is the main topic in multi-objective optimization.

Regarding the methods of resolution, Hwang and Masud (2012) classify them into three categories: *a priori*, interactive, and *a posteriori*, which differ based on the stage at which decision-makers expresses their preferences. In the *a priori* methods, decision-makers reveal their preferences before the computing stage, for which they have to have extensive knowledge of the process to obtain satisfactory solutions (Mavrotas & Diakoulaki, 1998). In interactive methods, decision-makers add preferences during the calculation stage. Interactive methods are suitable for large problems because they can restrict the search area to focus only on the part of interest of the set, guided by the preferences of the decision-maker (Mavrotas & Diakoulaki, 1998). Finally, *a posteriori* methods seek to generate the complete Pareto set, and then show it to the decision-maker who makes the final decision (Mavrotas & Florios, 2013). The *a posteriori* methods are the most computationally demanding, making them less popular (Mavrotas, 2009).

There are different techniques to compute the Pareto set. We now briefly describe three of them, one for each category. First, the weighted sum is an *a priori* method that consists of assigning a weight to each objective function, transforming the problem from a multi-objective optimization problem to a classical single-objective optimization problem. Consider a weight vector $w \in \mathbb{R}^m_{\succ 0}$. The linear scalarization of (1) is given by (2). The limitation of this method is that not all non-dominated solutions can be found (Ehrgott, 2006). For a further description of *a priori* methods, we refer the reader to Marler and Arora (2004).

$$\begin{aligned} &\text{minimize} \quad \sum_{i=1}^{m} w_i f_i(\mathbf{x}) \\ &\text{s. t.} \quad \mathbf{x} \in \chi \end{aligned} \tag{2}$$

Second, STEM is an interactive method that seeks to determine solutions by minimizing the Tchebycheff distance with respect to an ideal solution (Benayoun, De Montgolfier, Tergny, & Laritchev, 1971). Consider a weight vector $\lambda \in \mathbb{R}^m_{\succ 0}$. The Tchebycheff scalarization of problem (1) is given by (3), where z_i^*, the components of the ideal point, are defined as $z_i^* = \inf_{\mathbf{x} \in \chi} f_i(\mathbf{x})$ with $i \in \{1, \dots, m\}$. For a more extensive description of interactive techniques, we refer the reader to Antunes, Alves, and Clímaco (2016).

$$\begin{aligned} &\text{minimize} \quad \max_{i \in \{1,\dots,m\}} \lambda_i |f_i(\mathbf{x}) - z_i^*| \\ &\text{s. t.} \quad \mathbf{x} \in \chi \end{aligned} \tag{3}$$

Finally, the ϵ-constraint method is an *a posteriori* method that solves one of the objective functions using the others as constraints (Chankong & Haimes, 1983). Consider $m-1$ constants $\epsilon_1 \in \mathbb{R}, \dots, \epsilon_{m-1} \in \mathbb{R}$. The ϵ-constraint scalarization of (1) is given by (4), where $f_1, g_1, \dots, g_{m-1}$ constitute the m component of the objective functions vector. This method was strongly improved by Mavrotas (2009), who proposed the augmented ϵ-constraint method that avoids redundant calculations and provides only non-weak solutions. A non-weak non-dominated solution is a feasible solution for which there is no other feasible solution that strictly improves all the objectives simultaneously. An improved version of this method is described in Mavrotas and Florios (2013).

$$\begin{aligned} &\text{minimize} \quad f_1(\mathbf{x}) \\ &\text{s. t.} \quad g_i(\mathbf{x}) \leqslant \epsilon_i, \ i \in \{1, \dots, m-1\} \\ &\qquad \mathbf{x} \in \chi \end{aligned} \tag{4}$$

The methodology proposed in this paper minimizes the Tchebycheff distance to the ideal solution to find and initial Pareto optimal point. It then uses the ϵ-constraint method to iteratively find other Pareto optimal solutions. Further details are provided in Section 5.

4. Mathematical formulation

In this section, we formulate the grape harvest planning as a mixed-integer linear multi-objective mathematical programming problem. This model tries to capture the operational complexity that underlies the harvesting process in the wine industry.

For the sake of exposition, we state the relevant definitions and the main assumptions in SubSection 4.1. In SubSection 4.2, the sets and parameters are described. The decision variables of the model are defined and described in SubSection 4.3, while the objective functions are established and explained in SubSection 4.4. In SubSection 4.5, the constraints of the model are stated. Finally, the constraints and variables necessary to linearize the model are added in SubSection 4.6, and consequently, one of the objective functions is modified according to the new formulation.

4.1. Definitions and assumptions

It may be useful for a better understanding of the model to give some definitions and assumptions that are fairly standard in the related literature (see for example both Ferrer et al. (2008) and Bohle et al. (2010)) and in the wine industry of several countries, such as Argentina, Australia, Chile, and the United States. There may be, of course, some vineyards of those countries that operate differently.

Consider the following definitions:

- A *block* is a particular portion of land with similar characteristics of soil, grape variety, and quality.
- A *winery* is a place of destination where the grapes arrive after being harvested in the blocks. Here is where wine production begins.
- A *mode* is the harvesting method, which can be mechanical or manual.
- A *harvest window* corresponds to a feasible set of consecutive harvest periods.
- A *tour* is an ordering of blocks to be harvested in a period.
- The *loss function* provides the decrease of grapes quality as a function of the deviation from the optimal day of harvest. An example is shown in Fig. 2.

Consider the following assumptions:

- For each block, the harvest window corresponds to a strict subset of the planning horizon.
- When the harvest of a block starts, it cannot be stopped.
- The harvested grapes in each block may go to multiple wineries.

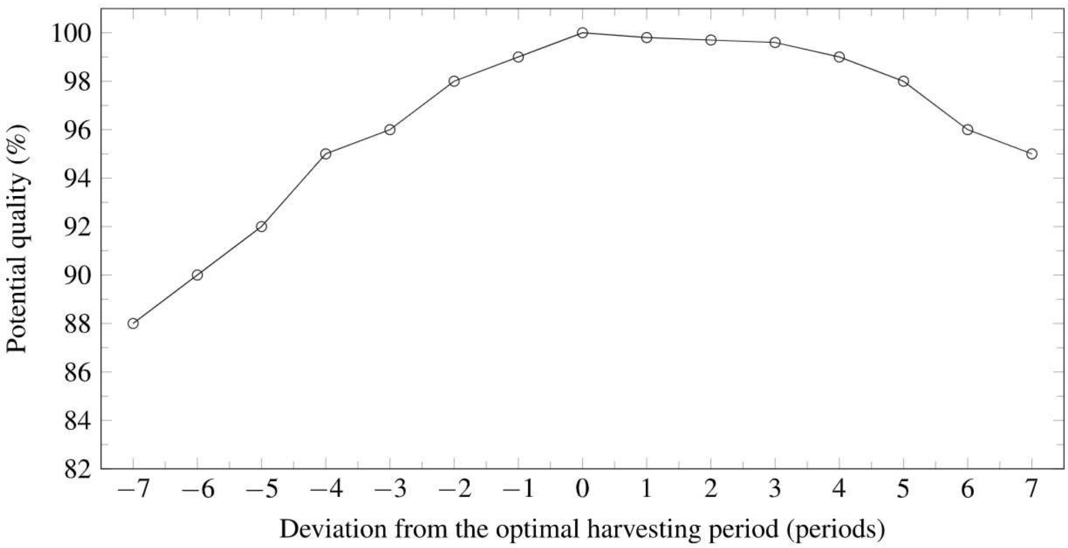

Fig. 2. Quality loss function example. Based on Ferrer et al. (2008).

Each winery has a maximum reception capacity.

- Wineries might require grapes from some particular block and a specific harvest mode. For example, in the case of premium wines, the harvest mode is usually by hand, and the grapes come from the blocks with the best grapes.
- Multiple blocks can be harvested in a period.
- Each block may be harvested using multiple modes in each period.
- There is a headquarters (depot), denoted AG, where the work begins and ends every period. So, in each period, it is necessary to define which blocks and in what sequence they will be harvested. Thus, the existence of intra-period routing is allowed. This last is a distinctive feature of our modeling approach compared to Ferrer et al. (2008).
- If a block is harvested, there is a minimum amount of grapes that should be harvested in each period. This value depends on the mode.
- There is a minimum number of workers required for harvesting a block.
- There is a maximum number of hours available for the mechanical harvest mode.
- All available grapes must be harvested in a block. If this does not happen, the residual non-harvested grapes will be detrimental for next year's plantation.
- For each harvest mode, one block is harvested at a time.
- There are fixed hiring and firing costs. Also, the workers are paid each period.
- A predefined loss function gives the reduction in grape quality due to harvesting before or after the optimal date.

4.2. Sets and Parameters

Consider the following sets:

- J_1: Set of blocks that can be harvested mechanically (indexed by j_1).
- J_2: Set of blocks that can be harvested manually (indexed by j_2).
- J: Equals to $J_1 \cup J_2 \cup \{AG\}$. For the sake of exposition, we denote AG by 0.
- K: Set of harvesting modes (indexed by k). Specifically, $k = 1$ represents the mechanical harvest mode and $k = 2$ the manual mode.
- T: Planning horizon (indexed by t).
- B: Set of wineries (indexed by b).

Consider the following parameters:

- $D_{ij} \in \mathbb{R}^+$: Distance between blocks i and j (km).
- $Q_{jt} \in \mathbb{R}^+$: Quality factor for block j in period t (%). This parameter is equal to zero for periods outside the harvest window, and within, its values are given by the loss function.
- $H \in \mathbb{R}^+$: Hiring cost ($\frac{\text{"\$"}}{worker}$).
- $F \in \mathbb{R}^+$: Firing cost ($\frac{\text{"\$"}}{worker}$).
- $C_k \in \mathbb{R}^+$: Unit cost of a productive resource of mode k. For $k = 1$, this parameter corresponds to the amount paid for using a machine one hour ($\frac{\text{"\$"}}{machinery\ hours}$). For $k = 2$, this parameter corresponds to the amount paid to each worker every period ($\frac{\text{"\$"}}{worker}$).
- $P_{kj} \in \mathbb{R}^+$: Unit productivity of mode k to harvest block j ($\frac{kg}{period\cdot worker\ or\ machine}$).
- $G_{jkb} \in \mathbb{R}^+$: Quantity of grapes in block j harvested with mode k required by winery b (kg).
- $W_{jt} \in \{0, 1\}$: It takes value 1 if block j can be harvested in period t, and 0 otherwise.
- $L_{kbt} \in \mathbb{R}^+$: Reception capacity of winery b in period t with mode k ($\frac{kg}{period}$).
- $A_t \in \mathbb{R}^+$: Available hours for the use of machinery in period t ($hours$).
- $M \in \mathbb{R}^+$: A sufficiently large number.
- $R_k \in \mathbb{R}^+$: Transportation cost for moving the whole mode k operation. ($\frac{\text{"\$"}}{Km}$).
- $V_k \in \mathbb{R}^+$: Minimum amount of grapes to be harvested if harvesting takes place in a block with mode k (kg).
- $N \in \mathbb{Z}^+$: Minimum number of workers needed to harvest one block ($workers$).

4.3. Variables

The main decisions of the problem are determining how, when, where, and how much to harvest.

- $x_{jtkb} \in \mathbb{R}$: Quantity of grapes harvested in block j, in period t, with mode k, en route to winery b (kg).
- $y_{jtkb} \in \{0, 1\}$: Binary variable that takes the value 1 if for block j, the harvest starts in period t, with mode k, en route to winery b, and 0 otherwise.
- $z_{ijtbk} \in \{0, 1\}$: Binary variable that takes the value 1 if the operations are moved from block i to block j, in period t, using mode k, en route to winery b, and 0 otherwise.
- $\tau_{jtbk} \in \mathbb{N}$: Variable used to eliminate the subtours in MTZ formulation. It represents the position in the cycle of block j, in period t,

with mode k, en route to winery b.

- $wh_t \in \mathbb{Z}^+$: Number of workers hired in period t (workers).
- $wf_t \in \mathbb{Z}^+$: Number of workers fired in period t (workers).
- $u_{jtk} \in \mathbb{Z}^+$: Quantity of productive resources used in block j, in period t, with mode k (workers or machine hours).
- $v_{jtkb} \in \{0, 1\}$: Binary variable that takes the value 1 if block j is harvested in period t, with mode k, en route to winery b, and 0 otherwise.

4.4. Objective functions

In general terms, optimizing the harvest planning implies minimizing the operational costs (5), and maximizing the quality of the harvested grapes (6). Both objectives are defined as follows:

$$MinF_1 = \overbrace{\sum_{j\in J,t\in T,k\in K} C_k\cdot u_{jtk}}^{(a)} + \overbrace{\sum_{t\in T} H\cdot wh_t}^{(b)} + \overbrace{\sum_{t\in T} F\cdot wf_t}^{(c)} + \overbrace{\sum_{i\in J,j\in J,t\in T,b\in B,k\in K:i\neq j} R\cdot D_{ij}\cdot z_{ijtbk}}^{(d)} \tag{5}$$

$$MaxF_2 = \sum_{j\in J,t\in T,k\in K,b\in B} Q_{jt}\cdot x_{jtkb} \tag{6}$$

The objective function F_1 seeks to minimize the harvest operational costs. In particular, (a) represents the resources (machine hours or workers) cost; (b) represents hiring cost; (c) represents the firing cost; and (d) represents the transportation cost. The objective function F_2, on the other hand, seeks to maximize the quality of the harvested grapes.

4.5. Constraints

We group the constraints in three categories: capacity and harvest operations constraints, harvesting modes constraints, and MTZ constraints. We refer the reader to Miller, Tucker, and Zemlin (1960) for the MTZ formulation of traveling salesman problems.

Capacity and harvest operations constraints:

$$\sum_{j\in J} x_{jtkb} \leqslant L_{kbt} \quad \forall\, k\in K,\, b\in B,\, t\in T \tag{7}$$

$$\sum_{t\in T} x_{jtkb} = G_{jkb} \quad \forall\, k\in K,\, b\in B,\, j\in J \tag{8}$$

$$x_{jtkb} \leqslant G_{jkb}\cdot v_{jtkb} \quad \forall\, k\in K,\, b\in B,\, j\in J,\, t\in T \tag{9}$$

$$V_k\cdot v_{jtkb} \leqslant x_{jtkb} \quad \forall\, k\in K,\, b\in B,\, j\in J,\, t\in T,\, G_{jkb}\neq 0 \tag{10}$$

$$\sum_{b\in B} x_{jtkb} \leqslant P_{kj}\cdot u_{jtk} \quad \forall\, k\in K,\, j\in J,\, t\in T \tag{11}$$

$$v_{jtkb} \leqslant W_{jt} \quad \forall\, j\in J,\, t\in T,\, k\in K,\, b\in B \tag{12}$$

$$1 \leqslant \sum_{k\in K,t\in T,b\in B:G_{jkb}\neq 0} v_{jtkb} \quad \forall\, j\in J,\, j\neq 0 \tag{13}$$

$$\sum_{t\in T:W_{jt}=1} y_{jtkb} \leqslant 1 \quad \forall\, b\in B,\, k\in K,\, j\in J \tag{14}$$

$$y_{jtkb} \leqslant v_{jtkb} \quad \forall\, k\in K,\, j\in J,\, t\in T,\, b\in B \tag{15}$$

$$v_{jt+1kb} - v_{jtkb} \leqslant y_{jt+1kb} \quad \forall\, k\in K,\, j\in J,\, t\in T-1,\, b\in B,\, j\neq 0,\, G_{jkb}\neq 0 \tag{16}$$

$$v_{jtkb} \leqslant \sum_{s\leqslant t:W_{js}=1} y_{jskb} \quad \forall\, k\in K,\, j\in J,\, t\in T,\, b\in B,\, W_{jt}=1,\, G_{jkb}\neq 0 \tag{17}$$

$$y_{jtkb} \leqslant \sum_{s\geqslant t:W_{js}=1} v_{jskb} \quad \forall\, k\in K,\, j\in J,\, t\in T,\, b\in B,\, W_{jt}=1,\, G_{jkb}\neq 0 \tag{18}$$

Harvesting modes constraints:

$$\sum_{j_2\in J_2} u_{j_2 t2} = \sum_{j_2\in J_2} u_{j_2 (t-1)2} + wh_t - wf_t \quad \forall\, t\geqslant 2 \tag{19}$$

$$\sum_{j_1\in J_1} u_{j_1 t1} \leqslant A_t \quad \forall\, t\in T \tag{20}$$

$$N\cdot v_{j_2 t2b} \leqslant u_{j_2 t2} \quad \forall\, t\in T,\, j_2\in J_2,\, b\in B \tag{21}$$

MTZ constraints:

$$\sum_{j\in J} z_{0jtbk} = 1 \quad \forall\, b\in B,\, t\in T,\, k\in K \tag{22}$$

$$\sum_{i\in J} z_{i0tbk} = 1 \quad \forall\, b\in B,\, t\in T,\, k\in K \tag{23}$$

$$\sum_{j\in J:j\neq i} z_{ijtbk} = v_{itkb} \quad \forall\, b\in B,\, i\in J-\{0\},\, t\in T,\, k\in K \tag{24}$$

$$\sum_{j\in J:j\neq i} z_{jitbk} = v_{itkb} \quad \forall\, b\in B,\, i\in J-\{0\},\, t\in T,\, k\in K \tag{25}$$

$$\tau_{0tbk} = 1 \quad \forall\, b\in B,\, t\in T,\, k\in K \tag{26}$$

$$2\cdot v_{jtkb} \leqslant \tau_{jtbk} \quad \forall\, j\in J-\{0\},\, b\in B,\, t\in T,\, k\in K \tag{27}$$

$$\tau_{jtbk} \leqslant \sum_{i\in J-\{0\}} v_{itkb} + 1 \quad \forall\, j\in J-\{0\},\, b\in B,\, t\in T,\, k\in K \tag{28}$$

$$\tau_{itbk} - \tau_{jtbk} + 1 \leqslant \left(\sum_{l\in J-\{0\}} v_{ltkb} - 1\right)\left(1 - z_{ijtbk}\right) \quad \forall\, b\in B,\, j,\, i\in J-\{0\},\, t\in T,\, k\in K,\, i\neq j \tag{29}$$

Eq. (7) is the wineries reception capacity constraint. Constraint (8) ensures that the wineries requirements are fulfilled. Constraint (9) forces v_{jtkb} to take the value 1 if grapes are harvested. Constraint (10) imposes a lower bound on the quantity of the harvested grapes, while (11) imposes a resource constraint. Constraint (12) allows harvesting only within the time window. Constraint (13) requires harvesting every block at least once. Constraint (14) imposes starting the harvest at most once, while constraints (15)–(18) ensure that there are no interruptions once the harvest starts. Constraint (19) relates the number of workers in consecutive periods. Constraint (20) limits the maximum number of machine-hours used, while (21) provides a lower bound on the number of workers used in a harvested block. Finally, Eqs. (22)–(29) are MTZ subtour elimination constraints.

4.6. Model linearization

Note that constraint (29) is non-linear because of the product between the variables z_{ijtbk} and v_{ltkb} in the right-hand side of the equation. This is a drawback when trying to solve the problem multiple times, as is the case of the ϵ-constraint method. To linearize the model, constraint (29) and the objective function of costs (5) must be modified to (34) and (35), respectively. Moreover, two new variables and four auxiliary constraints are added.

Auxiliary Variables:

$\delta_{lijtbk} \in \{0, 1\}$: Corresponds to the product between z_{ijtbk} and v_{ltkb} with $l, i, j \in J - \{0\}$.

$\beta_{jtbk} \in \{0, 1\}$: Binary variable that takes the value 1 if $j \in J - \{0\}$ belongs to the tour in period t, with method k, with winery b as destination.

Auxiliary constraints:

$$\delta_{lijtbk} \leqslant z_{ijtbk} \quad \forall\, b \in B, j, i, l \in J - \{0\}, t \in T, k \in K, i \neq j \tag{30}$$

$$\delta_{lijtbk} \leqslant v_{ltkb} \quad \forall\, b \in B, j, i, l \in J - \{0\}, t \in T, k \in K, i \neq j \tag{31}$$

$$v_{ltkb} + z_{ijtbk} - 1 \leqslant \delta_{lijtbk} \quad \forall\, b \in B, j, i, l \in J - \{0\}, t \in T, k \in K, i \neq j \tag{32}$$

$$\tau_{jtbk} \leqslant M \cdot \beta_{jtbk} \quad \forall\, b \in B, j \in J - \{0\}, k \in K, t \in T \tag{33}$$

$$\tau_{itbk} - \tau_{jtbk} + 1 \leqslant \sum_{l \in J-\{0\}} v_{ltkb} - \sum_{l \in J-\{0\}} \delta_{lijtbk} + \left(2 - \beta_{itbk} - \beta_{jtbk}\right) \cdot M \quad \forall\, b \in B, j, i \in J - \{0\}, t \in T, k \in K, i \neq j \tag{34}$$

New operational costs objective function:

$$MinF_1{}' = \overbrace{\sum\nolimits_{j \in J, t \in T, k \in K} C_k \cdot u_{jtk}}^{(a)} + \overbrace{\sum\nolimits_{t \in T} H \cdot wh_t}^{(b)} + \overbrace{\sum\nolimits_{t \in T} F \cdot wf_t}^{(c)} + \overbrace{\sum\nolimits_{i \in J, j \in J, t \in T, b \in B, k \in K: i \neq j} R \cdot D_{ij} \cdot z_{ijtbk}}^{(d)} + \overbrace{\sum\nolimits_{i \in J-\{0\}, t \in T, b \in B, k \in K} \lambda \cdot \beta_{itbk}}^{(e)} \tag{35}$$

Constraints (30)–(32), force δ_{lijtbk} to be the product between z_{ijtbk} and v_{ltkb} keeping linearity. Constraint (33), and the term (e) of the new objective function $F_1{}'$, force τ_{jtbk} to be zero if and only if β_{jtbk} is zero, where $\lambda > 0$ is small enough.

5. The negotiation protocol

This section describes in detail the negotiation protocol. The required algorithms are described in each subsection. The protocol is summarized in SubSection 5.4.

5.1. Computing Pareto optimal solutions: the augmented ϵ-constraint method

As mentioned earlier, in a multi-objective optimization framework, it is uncommon that a single solution optimizes all the objective functions simultaneously. Therefore, the concept of optimality is replaced by Pareto optimality or non-dominated solutions. Pareto optimal solutions are those such that any objective function cannot be improved without worsening the value of at least one of the others. The Pareto set groups all of the Pareto optimal solutions. In multi-objective optimization problems, the decision-makers must choose the solution they prefer the most within this set.

The proposed negotiation protocol needs to generate Pareto optimal solutions iteratively. In this paper, we use the augmented ϵ-constraint method of Mavrotas (2009) for generating the Pareto optimal solutions. This method constitutes an improved implementation of the ϵ-constraint method of Chankong and Haimes (1983) in which one goal is optimized while the others are added as constraints.

The augmented ϵ-constraint method has several steps. The first one is to compute the Payoff Table in lexicographic order, which is shown in Algorithm 1. In this case, z_1^* corresponds to the optimal value of maximizing $-F_1$, while n_2 corresponds to the maximum value of F_2 having $-F_1(X) \geqslant z_1^*$ as a constraint. The same goes for the values of z_2^* and n_1. Then, $[n_1, z_1^*]$ and $[n_2, z_2^*]$ are ranges of variation of F_1 and F_2, respectively.

Algorithm 1. Payoff Table in lexicographic order

input(F_1, F_2) $z_1^* \leftarrow \max -F_1(X)$ s.t constraints (3)–(24), (26)–(30)
add $-F_1(X) \geqslant z_1^*$ as constraint (29)
$n_2 \leftarrow \max F_2(X)$ s.t constraints (3)–(24), (26)–(29)
Remove constraint (29)
$z_2^* \leftarrow \max F_2(X)$ s.t constraints (3)–(24), (26)–(30)
add $F_2(X) \geqslant z_2^*$ as constraint (29)
$n_1 \leftarrow \max -F_1(X)$ s.t constraints (3)–(24), (26)–(29)
Remove constraint (29)
return $(n_1, z_1^*), (n_2, z_2^*)$

Algorithm 2 describes how this method is used to solve the multi-objective grape harvest planning problem. We divide the range of variation $[n_2, z_2^*]$ into g equal intervals of length Δ, using $g - 1$ intermediate equidistant grid points. We use the lower bounds of such g intervals, as lower bound constraints of F_2 to solve auxiliary optimization problems. Mavrotas (2009) shows that if constraints in the form $F_2(X) = n_2 + i \cdot \Delta$ are added, then the auxiliary optimization problems could lead to weak non-dominated solutions. To tackle this drawback, we modify the constraint to $F_2(X) \geqslant n_2 + i \cdot \Delta$, as proposed by Mavrotas (2009), and we add λ as a non-negative surplus variable, turning the constraint to $F_2(X) - \lambda = n_2 + i \cdot \Delta$. To ensure that the solution is Pareto optimal, we include λ in the objective function.

Algorithm 2. Augmented $\epsilon -$ *constraint*

Input(F_1, F_2, g)
$(n_1, z_1^*), (n_2, z_2^*) \leftarrow$ **Algorithm 1**: Lexicographic Payoff Table(F_1, F_2)
Define $r \leftarrow z_2^* - n_2$ as the length of the range of variation of F_2
Define $\epsilon > 0$ as an adequately small number
Calculate step values $\Delta = \frac{r}{g}$
for $i = 0, \ldots, g - 1$ **do**
$\left(X_i^*, \lambda_i^*\right) \leftarrow \underset{X, \lambda}{\mathrm{argmax}}\left(-F_1(X) + \epsilon \cdot \frac{\lambda}{r}\right)$ s.t. constraints (3)–(24), (26)–(30); $F_2(X) - \lambda = n_2 + i \cdot \Delta; \lambda \geqslant 0$
end for
return $\{X_1^*, \ldots, X_g^*\}$

We now turn to the problem of selecting a solution within the Pareto optimal set. For instance, Mavrotas (2009) proposes an interactive method that uses the augmented ϵ-constraint to iteratively find five Pareto optimal solutions. These solutions are presented to a single decision-maker who must choose his/her preferred solution. This poses a key problem when we consider two decision-makers with conflicting objectives. Indeed, they would obviously choose different points within these five solutions, maximizing his/her objective function. To deal with this problem, we propose in the next subsection a negotiation method that can lead to an agreed final harvest schedule.

5.2. Obtaining a first Pareto optimal solution: minimizing the augmented weighted Tchebycheff distance

Using Algorithm 1, we obtain the range of variation for both objective functions. According to Benayoun et al. (1971), the vector (z_1^*, z_2^*) is called the *ideal solution* and corresponds to the optimal objective functions values. In general, there is no feasible $\widehat{X}$ such that

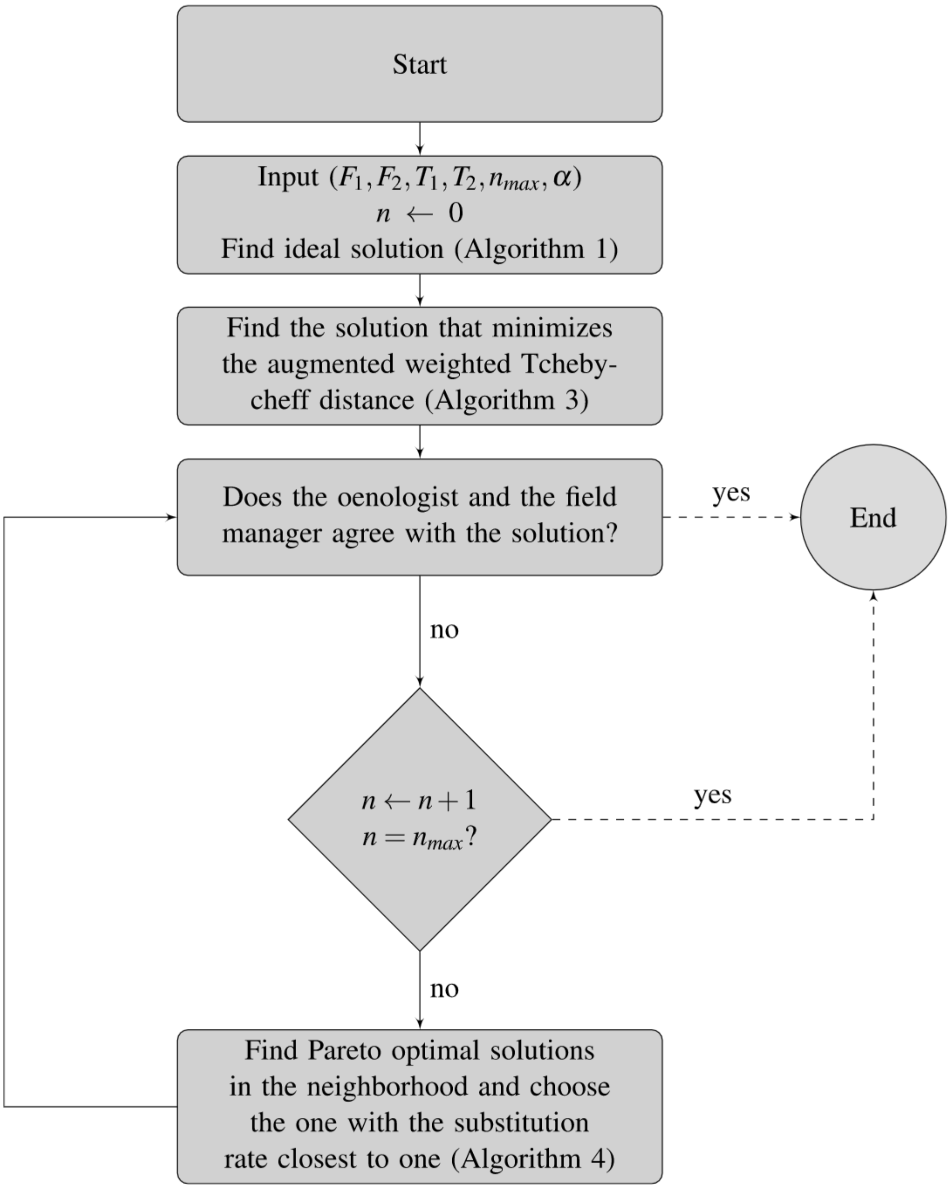

Fig. 3. Flow chart of the negotiation protocol.

Table 1
Payoff table with lexicographical order.

	$-F_1$	F_2
$\max -F_1$	−2,599,500	55,000
$\max F_2$	−2,600,344	78,000

$-F_1(\widehat{X}) = z_1^*$ and $F_2(\widehat{X}) = z_2^*$. To find a starting point for the negotiation, one would like to have a feasible and Pareto optimal solution near this ideal solution.

In this paper, we use Solanki (1991) approach to find an initial Pareto optimal solution by minimizing the augmented weighted Tchebycheff distance. A pseudo-code of this procedure is provided in Algorithm 3. In line 5, the first term $\max\{\alpha_1(z_1^* + F_1(X)), \alpha_2(z_2^* - F_2(X))\}$ corresponds to the augmented weighted Tchebycheff distance between (z_1^*, z_2^*) and the Pareto boundary. We follow Benayoun et al. (1971) for the computation of the weights α_1, α_2. The second term $\rho_1 \cdot F_1(X) - \rho_2 \cdot F_2(X)$ ensures that the solution of the optimization problem is non-weak, where ρ_1 and ρ_2 are small positive scalars (Antunes et al., 2016).

Table 2
Time and number of Pareto optimal solutions as a function of g.

g	Time (sec)	Number of Pareto optimal solutions
10	120	9
100	1800	13
1000	15,840	14
10,000	259,920	14

Algorithm 3. Augmented weighted Tchebycheff

1: input(F_1, F_2)
2: $(n_1, z_1^*), (n_2, z_2^*) \leftarrow$ Payoff Table(F_1, F_2)
3: $\beta_1 \leftarrow \frac{n_1 - z_1^*}{n_1 \parallel \nabla F_1 \parallel_{L_2}}$; $\beta_2 \leftarrow \frac{z_2^* - n_2}{z_2^* \parallel \nabla F_2 \parallel_{L_2}}$
4: $\alpha_1 \leftarrow \frac{\beta_1}{\beta_1 + \beta_2}$; $\alpha_2 \leftarrow \frac{\beta_2}{\beta_1 + \beta_2}$
5: **return** X^*

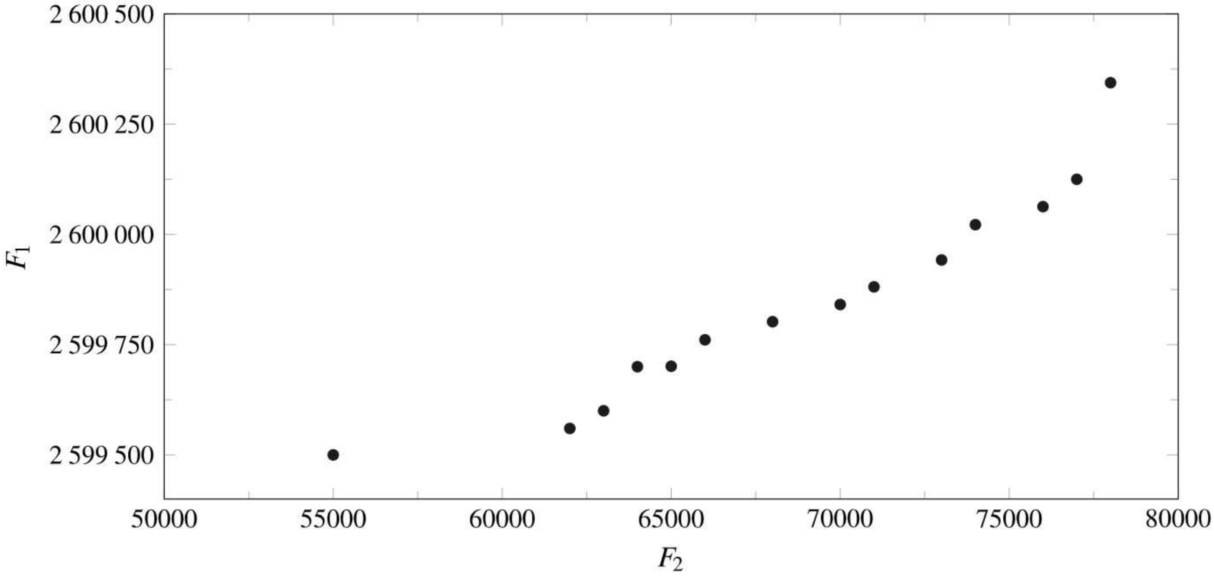

Fig. 4. Pareto optimal solutions for $g = 10, 000$.

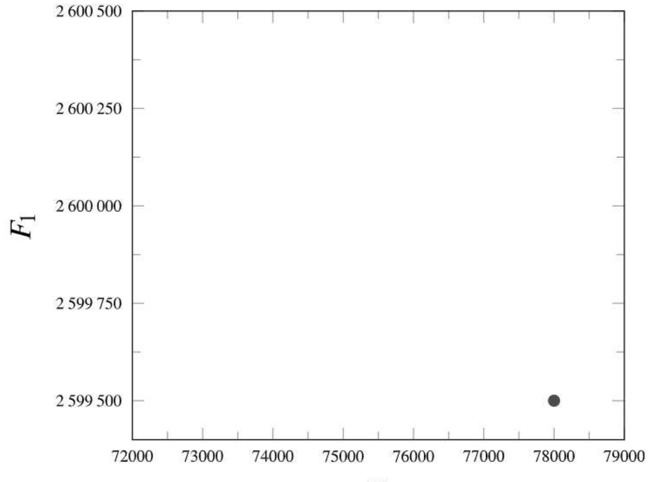

Fig. 5. Ideal solution.

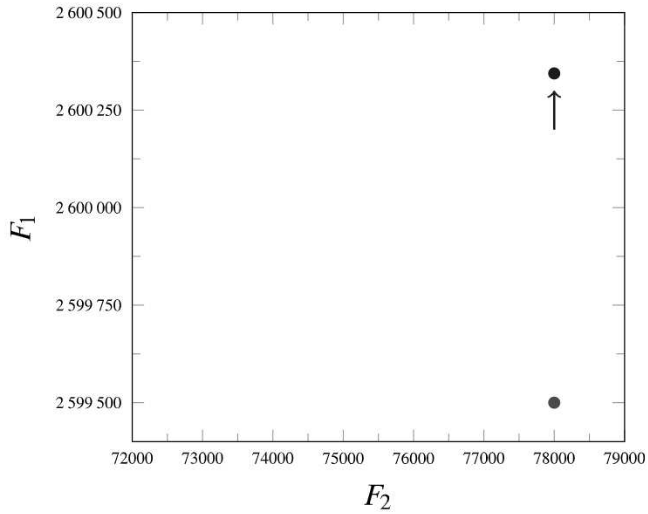

Fig. 6. Closest Pareto optimal solution.

5.3. Exploring the neighborhood and generation of a new solution

After obtaining the solution described in the previous subsection, it is presented to the two decision-makers, in our case, the oenologist and the field manager. If both agree with the solution, then the negotiation ends. Otherwise, it is necessary to find a new solution. To do so, we propose Algorithm 4.

Algorithm 4. local neighborhood

1: Input(F_1, F_2, X^n, α, g, W),
2: $r_{LB} \leftarrow (1-\alpha)F_2(X^n) + \alpha n_2$ $r_{UB} \leftarrow (1-\alpha)F_2(X^n) + \alpha z_2^*$
3: Define $r \leftarrow r_{UB} - r_{LB}$ as the new range for F_2
4: Define $\epsilon > 0$ as an adequately small number
5: Calculate step values $\Delta = \frac{r}{g}$
6: **for** $i = 0, \ldots, g-1$ **do**
7: $\left(X_i^*, \lambda_i^*\right) \leftarrow \underset{X,\lambda}{\operatorname{argmax}}\left(-F_1(X) + \epsilon\cdot\frac{\lambda}{r}\right)$ s.t constraints (1)–(22), (24)–(28);
$F_2(X) - \lambda = r_{LB} + i\cdot\Delta; \lambda \geqslant 0$
8: $\gamma_i \leftarrow \frac{|F_1(X^n) - F_1(X_i^*)|}{|F_2(X_n) - F_2(X_i^*)|}\cdot\frac{F_2(X^n)}{F_1(X^n)}$
9: **end for**
10: **if** $\{X_0^*, X_1^*, \ldots, X_{g-1}^*\} \subseteq W$ **then**
11: $X^{n+1} \leftarrow X^n$
12: **else**
13: $j \leftarrow \underset{\substack{i=0,\ldots,g-1 \\ X_i^* \notin W}}{\operatorname{argmin}} \left| 1 - \gamma_i \right|$
14: $X^{n+1} \leftarrow X_j^*$
15: **end if**
16: **return** X^{n+1}

This algorithm explores a neighborhood around the current solution. The radius of the neighborhood is controlled by the parameter α (line 2). To find Pareto optimal solutions within this neighborhood, we use the augmented ϵ-constraint method described in Algorithm 2 to generate a set of Pareto optimal points. Restricting this procedure to a neighborhood has the advantage of reducing the computation time. We consider only the solutions not previously presented to the decision-makers, using a tabu list W.

It is important to point out that several interactive methods present

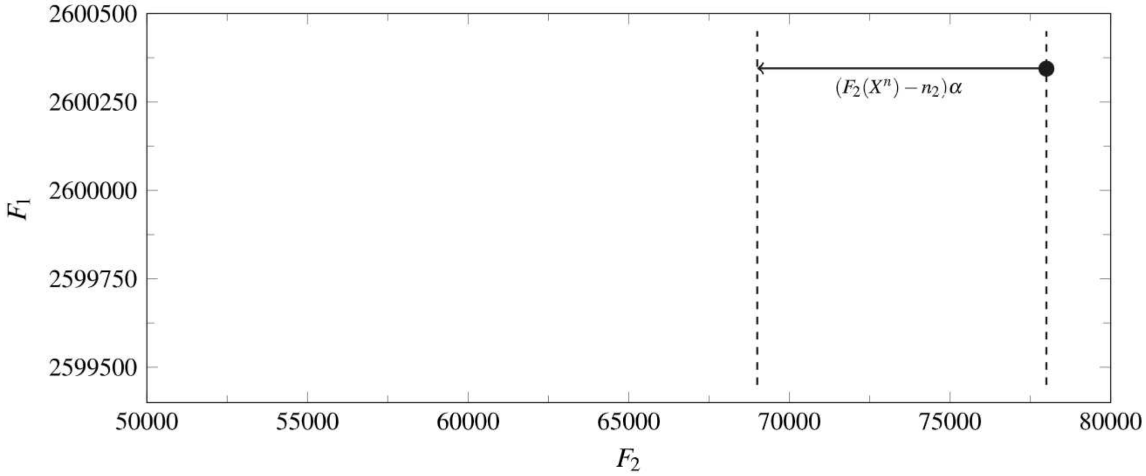

Fig. 7. Local Neighborhood of the first Pareto optimal solution.

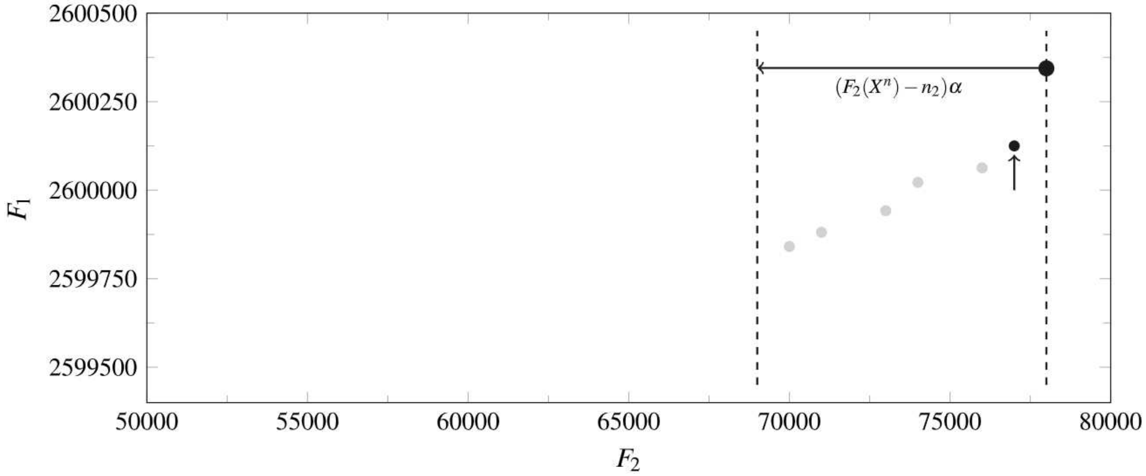

Fig. 8. Points of the Pareto Boundary in the neighborhood.

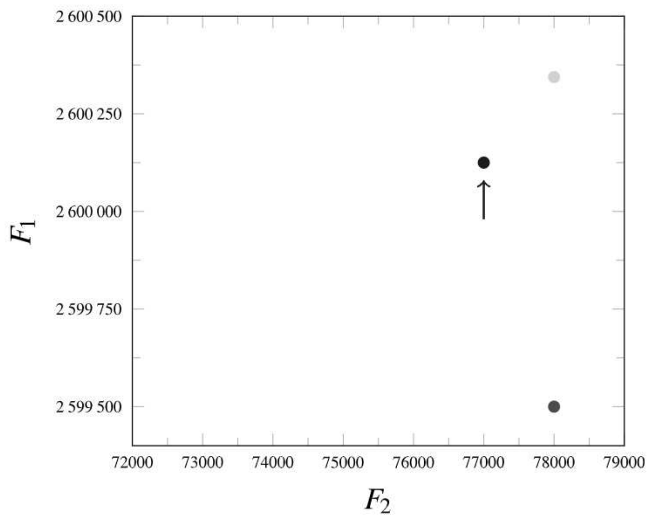

Fig. 9. Solution with $n = 1$.

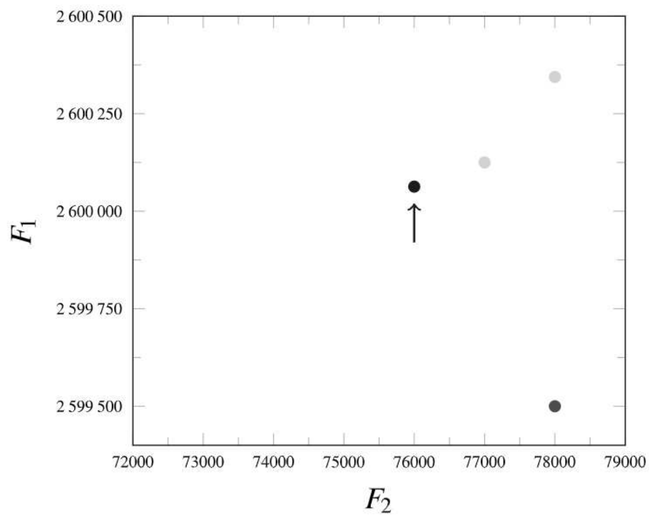

Fig. 10. Solution with $n = 2$.

more than one point to the decision-maker to guide his/her decision iteratively by following his/her preferred solution within these points (Mavrotas, 2009). In our context, however, there are two decision-makers with opposing objectives. Therefore, it is very likely that, if more than one Pareto optimal solution is presented to both decision-makers, they might choose different points, which would make it unclear how to continue the exploration. To avoid this problem, we offer a single Pareto optimal solution to both decision-makers.

To choose among the solutions provided by the augmented ϵ-constraint method, when exploring the neighborhood around the current solution (line 7), we propose to select the solution with the proportional change in F_1 more similar to the proportional change in F_2. More

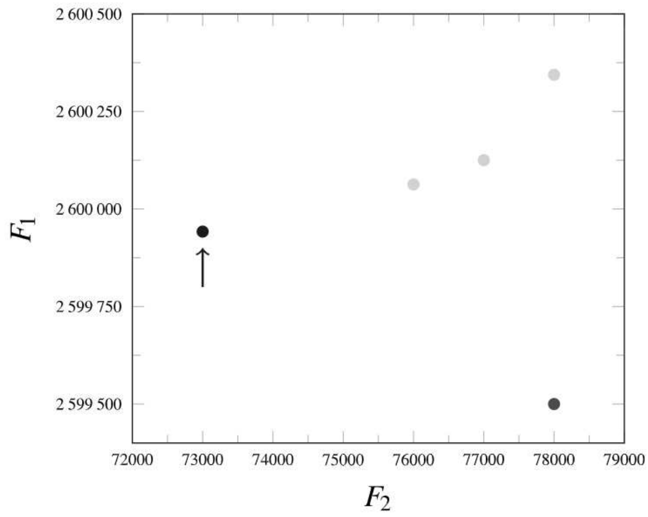

Fig. 11. Solution with $n = 3$.

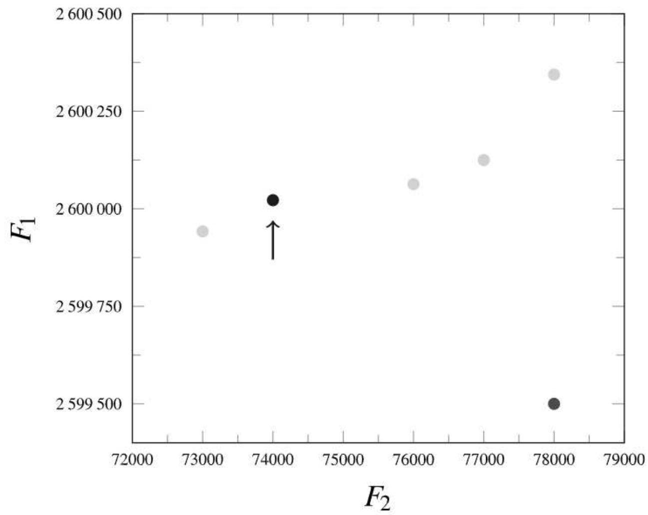

Fig. 12. Solution with $n = 4$.

specifically, we choose the Pareto optimal solution with the rate of substitution between F_1 and F_2 (line 8) that is closest to one (line 13). We seek to facilitate the negotiation between the two decision-makers by providing them with solutions that are potentially more acceptable and balanced. That is, we show them solutions that increase one goal in a similar percentage to the decrease of the second goal.

5.4. *The iterative process*

Algorithm 5, which summarizes the negotiation protocol, uses the previous algorithms in an iterative process that ends when an agreement is found (line 5 and 15). All the solutions found within a neighborhood belong to the tabu list (line 11), or the maximum number of iteration is reached (line 21). For simplicity, consider the non-parametric functions, $T_1(X)$ and $T_2(X)$, which represent the response given by the field manager and the winemaker, respectively, when faced with the solution *X*. These functions take value 1 if *X* is agreed upon, and 0 if not.

Algorithm 5. Main

1: input(F_1, F_2, T_1, T_2, n_{max}, α)
2: Define $W \leftarrow \emptyset$ as the tabu list
3: $X^0 \leftarrow$ Augmented weighted Tchebycheff (F_1, F_2)
4: **if** $T_1(X^0) \cdot T_2(X^0) = 1$ **then**
5:     **return** X^0
6: **else**
7:     $W \leftarrow W \cup \{X^0\}$
8: **end if**
9: **for** $n = 0, \ldots, n_{max} - 1$ **do**
10:     **if** X^n= local neighborhood(F_1, F_2, X^n, α, g, W) **then**
11:         **return** X^n
12:     **else**
13:         $X^{n+1} \leftarrow$ local neighbourhood (F_1, F_2, X^n, α, g, W)
14:         **if** $T_1(X^{n+1}) \cdot T_2(X^{n+1}) = 1$ **then**
15:             **return** X^{n+1}
16:         **else**
17:             $W \leftarrow W \cup \{X^{n+1}\}$
18:         **end if**
19: **end if**
20:     **end for**
21: **return** $X^{n_{max}}$

Finally, for the sake of the explanation, Fig. 3 presents a flow chart that also summarizes the proposed negotiation protocol.

6. An illustrative example

We use a synthetic educational example consisting of ten blocks and three periods to illustrate the negotiation protocol defined in the previous section (see Appendix for detailed data). The multi-objective harvest model and the augmented ϵ-constraint method are coded in Python 3.7. Each auxiliary optimization problem is solved using the Gurobi 8.0.1 solver on a computer with an Intel processor (R) Core (TM) i5-7300HQ CPU 2.50 GHz and 8 Gb of RAM.

The augmented ϵ-constraint method (SubSection 5.1) applied to our model can be found in Table 1, which shows the Payoff Table in lexicographic order. For $g \in \{10, 10^2, 10^3, 10^4\}$, Table 2 shows both the number of Pareto optimal solutions found and the corresponding computing time. Fig. 4 shows the 14 Pareto optimal solutions for $g = 10, 000$. At first glance, it seems that the number of Pareto optimal solutions converges to 14. Nevertheless, we cannot assure that there are no more Pareto optimal solutions to be found unless we analyze the case when $g \to \infty$, which implies that the computing time also goes to infinity. This, of course, is impractical, especially in the wine industry, where decision support systems for harvesting must be used on a daily basis.

To find the first Pareto optimal solution, we follow the procedure detailed in SubSection 5.2. Fig. 5 shows the ideal solution, while Fig. 6 shows the first Pareto optimal solution presented to both decision-makers.

To generate the next solution, we use the approach defined in SubSection 5.3. Fig. 7 shows the last solution with no agreement and the neighborhood around it. Note that in this case, the value of F_2 in the current solution equals the upper bound z_2^*. Thus, the neighborhood lies to only one side of that point. Fig. 8 shows the solutions found by the augmented ϵ-constraint method and the next candidate solution.

Finally, we apply the iterative process described in SubSection 5.4. Figs. 9–12 show the evolution of the Pareto optimal solutions presented to the decision-makers.

7. Concluding remarks

Based on the real operations of one of the three largest wineries in Chile, we study the conflicting objectives of the oenologist and the field manager when scheduling the grapes harvest. We propose a multi-objective mixed-integer linear programming considering two objectives, namely, to maximize the quality of the harvested grapes and to minimize the total operational costs. As far as we know, our solution approach is the first one that deals with the difficulty of dealing with two decision-makers simultaneously, in this type of problem.

To simplify the negotiation process, our protocol presents, in each iteration, a single non dominated solution to both decision-makers. This solution is obtained by examining the neighborhood of the previous solution and computing a set of Pareto optimal solutions through the augmented ϵ-constraint method. We choose within this set a new harvest schedule using a substitution rate criteria. We avoid visiting twice the same solution using a tabu list. The iterative process ends when an agreement is reached, or no further solutions could be provided. A small example is used to illustrate how the proposed procedure should be applied.

Even though our primary focus is the wine industry, we believe that the proposed procedure is flexible enough to be easily extendable to other contexts in which two or more decision-makers with conflicting objectives have to agree on a solution. Particularly, the idea of having optimal extracting/harvesting periods with a decreasing quality interval around it appears often in the natural resources industries, for example, in fishery, forestry, and agriculture. Thus, a modified version of our negotiation protocol could also be used in such contexts.

As future research, we would like to find a solution procedure that would allow us to solve larger instances of the proposed model in a reasonable amount of time. This would allow us to study how the proposed procedure could work with real decision-makers in the setting of a real problem. However, to implement the proposed methodology for larger wineries one should consider non-exact methods (Jha et al., 2019) since, even for our medium size instance, the Pareto solutions search scales poorly (see Table 2). Finally, we would also like to add a third objective function that minimizes the CO2 emissions resulting from the transportation of workers.

CRediT authorship contribution statement

Mauricio Varas: Conceptualization, Methodology, Software, Formal analysis, Investigation, Writing - original draft, Writing - review & editing, Supervision, Funding acquisition. **Franco Basso:** Formal analysis, Investigation, Writing - original draft, Writing - review & editing, Funding acquisition. **Sergio Maturana:** Investigation, Writing - original draft, Writing - review & editing. **David Osorio:** Conceptualization, Methodology, Software, Validation, Software. **Raul Pezoa:** Software, Validation, Writing - original draft, Funding acquisition.

Acknowledgments

Franco Basso gratefully acknowledges the financial support from the Complex Engineering Systems Institute, ISCI (grant CONICYT PIA/BASAL AFB180003). Mauricio Varas thanks a grant from Science, Technology, Knowledge, and Innovation Ministry of Chile (FONDECYT Project 11190892).

Appendix A

The data used in the illustrative example (see Section 5) are shown in Tables 3–9. In this case, the planning horizon is $T = 3$.

Table 3
Sets.

Set	Cardinality
J_1	10
J_2	10
J	11
K	2
B	2

Table 4
Parameters.

Parameter	Value
R_k	100
N	21
A	400
V_1	0
V_2	0

Table 5
Distance between blocks.

	0	1	2	3	4	5	6	7	8	9	10
0	0	14	17	9	8	5	13	9	6	4	5
1	14	0	7	15	14	17	23	19	16	14	15
2	17	7	0	8	7	10	20	22	19	17	18
3	9	15	8	0	7	10	18	14	11	9	10

(continued on next page)

Table 5 (*continued*)

	0	1	2	3	4	5	6	7	8	9	10
4	8	14	7	7	0	3	13	17	14	12	13
5	5	17	10	10	3	0	12	14	11	9	10
6	13	23	20	18	13	12	0	4	7	9	12
7	9	19	22	14	17	14	4	0	3	5	8
8	6	16	19	11	14	11	7	3	0	2	5
9	4	14	17	9	12	9	9	5	2	0	3
10	5	15	18	10	13	10	12	8	5	3	0

Table 6
Time windows and quality loss factor of blocks.

J	T	W	Q
0	1	1	0
0	2	1	0
0	3	1	0
1	1	1	0
1	2	1	2
1	3	1	0
2	1	1	2
2	2	1	0
2	3	1	1
3	1	1	2
3	2	1	0
3	3	1	1
4	1	1	0
4	2	1	2
4	3	1	0
5	1	1	0
5	2	1	2
5	3	1	0
6	1	1	0
6	2	1	0
6	3	1	2
7	1	1	8
7	2	1	6
7	3	1	0
8	1	1	0
8	2	1	0
8	3	1	0
9	1	1	0
9	2	1	0
9	3	1	0
10	1	1	0
10	2	1	0
10	3	1	0

Table 7
Productivity of resource in a block.

K	J	P_{kj}
1	0	1
1	1	4200
1	2	3540
1	3	4200
1	4	3540
1	5	3540
1	6	3540
1	7	4200
1	8	3540
1	9	3540
1	10	3540
2	0	1
2	1	3790
2	2	4980
2	3	4980
2	4	3790
2	5	3790
2	6	4980
2	7	3790
2	8	4980
2	9	6390
2	10	6390

Table 8
Estimated kg of grapes in block.

J	K	B	G_{jkb}
0	1	1	0
1	1	1	0
2	1	1	0
3	1	1	0
4	1	1	0
5	1	1	0
6	1	1	0
7	1	1	0
8	1	1	0
9	1	1	0
10	1	1	0
g	1	2	0
1	1	2	0
2	1	2	0
3	1	2	0
4	1	2	0
5	1	2	0
6	1	2	0
7	1	2	0
8	1	2	0
9	1	2	0
10	1	2	0
0	2	1	0
1	2	1	0
2	2	1	0
3	2	1	0
4	2	1	0
5	2	1	0
6	2	1	0
7	2	1	0
8	2	1	0
9	2	1	0
10	2	1	0
0	2	2	0
1	2	2	10,000
2	2	2	10,000
3	2	2	10,000
4	2	2	10,000
5	2	2	10,000
6	2	2	10,000
7	2	2	10,000
8	2	2	10,000
9	2	2	10,000
10	2	2	10,000

Table 9
Processing capacity of winery.

K	B	T	L_{kbt}
1	1	1	200,000
1	1	2	200,000
1	1	3	200,000
1	2	1	200,000
1	2	2	200,000
1	2	3	200,000
2	1	1	150,000
2	1	2	150,000
2	1	3	150,000
2	2	1	150,000
2	2	2	150,000
2	2	3	150,000

References

Alvarez, P. P., Espinoza, A., Maturana, S., & Vera, J. (2020). Improving consistency in hierarchical tactical and operational planning using robust optimization. *Computers & Industrial Engineering, 139*, 106112.

Antunes, C. H., Alves, M. J., & Clímaco, J. (2016). *Multiobjective linear and integer programming.* Springer.

Arnaout, J.-P. M., & Maatouk, M. (2010). Optimization of quality and operational costs through improved scheduling of harvest operations. *International Transactions in Operational Research, 17*(5), 595–605.

Baesler, F., & Palma, C. (2014). Multiobjective parallel machine scheduling in the sawmill industry using memetic algorithms. *The International Journal of Advanced Manufacturing Technology, 74*(5–8), 757–768.

Basso, F., Guajardo, M., & Varas, M. (2020). Collaborative job scheduling in the wine bottling process. *Omega, 91*, 102021.

Basso, F., & Varas, M. (2017). A mip formulation and a heuristic solution approach for the bottling scheduling problem in the wine industry. *Computers & Industrial Engineering, 105*, 136–145.
Benayoun, R., De Montgolfier, J., Tergny, J., & Laritchev, O. (1971). Linear programming with multiple objective functions: Step method (stem). *Mathematical Programming, 1*(1), 366–375.
Bertsimas, D., & Sim, M. (2004). The price of robustness. *Operations Research, 52*(1), 35–53.
Bilir, C., Ekici, S. O., & Ulengin, F. (2017). An integrated multi-objective supply chain network and competitive facility location model. *Computers & Industrial Engineering, 108*, 136–148.
Bohle, C., Maturana, S., & Vera, J. (2010). A robust optimization approach to wine grape harvesting scheduling. *European Journal of Operational Research, 200*(1), 245–252.
Bottani, E., Murino, T., Schiavo, M., & Akkerman, R. (2019). Resilient food supply chain design: Modelling framework and metaheuristic solution approach. *Computers & Industrial Engineering.*.
Cakici, E., Jia, J., Yu, P., Mason, S. J., Richard Cassady, C., Pohl, L., & Lachowsky, A. J. (2006). Cellar tank piping network analysis at e. & j. gallo winery. *Journal of Wine Research, 17*(3), 145–160.
Chankong, V., & Haimes, Y. Y. (1983). *Multiobjective decision making: theory and methodology.* Courier Dover Publications.
Cholette, S. (2007). A novel problem for a vintage technique: using mixed-integer programming to match wineries and distributors. *Interfaces, 37*(3), 231–239.
Cholette, S. (2009). Mitigating demand uncertainty across a winery's sales channels through postponement. *International Journal of Production Research, 47*(13), 3587–3609.
Egan, D., & Bell, A. (2002). Chilean wines: a successful image. *International Journal of Wine Marketing, 14*(2), 33–42.
Ehrgott, M. (2006). A discussion of scalarization techniques for multiple objective integer programming. *Annals of Operations Research, 147*(1), 343–360.
Emmerich, M. T., & Deutz, A. H. (2018). A tutorial on multiobjective optimization: fundamentals and evolutionary methods. *Natural Computing, 17*, 585–609.
Ferrer, J.-C., Mac Cawley, A., Maturana, S., Toloza, S., & Vera, J. (2008). An optimization approach for scheduling wine grape harvest operations. *International Journal of Production Economics, 112*(2), 985–999.
Garcia, O., et al. (1990). Linear programming and related approaches in forest planning. *New Zealand Journal of Forestry Science, 20*(3), 307–331.
Garcia, F. A., Marchetta, M. G., Camargo, M., Morel, L., & Forradellas, R. Q. (2012). A framework for measuring logistics performance in the wine industry. *International Journal of Production Economics, 135*(1), 284–298.
Harris, I., Rodrigues, V. S., Pettit, S., Beresford, A., & Liashko, R. (2018). The impact of alternative routeing and packaging scenarios on carbon and sulphate emissions in international wine distribution. *Transportation Research Part D: Transport and Environment, 58*, 261–279.
Hussain, M., Cholette, S., & Castaldi, R. M. (2008). An analysis of globalization forces in the wine industry: implications and recommendations for wineries. *Journal of Global Marketing, 21*(1), 33–47.
Hwang, C.-L., & Masud, A. S. M. (2012). *Multiple objective decision making—methods and applications: A state-of-the-art survey* (Vol. 164). Springer Science & Business Media.
Johnson, K. N., & Scheurman, H. L. (1977). Techniques for prescribing optimal timber harvest and investment under different objectives—discussion and synthesis. *Forest Science, 23*.
Lima, J. L. (2015). Estudio de caracterización de la cadena de producción y comercialización de la agroindustria vitivinícola: estructura, agentes y prácticas. Tech. rep. ODEPA, Ministerio de Agricultura.
Mac Cawley, A. F. (2014). *The international wine supply chain: challenges from bottling to the glass.* Ph.D. thesisGeorgia Institute of Technology.
Marler, R. T., & Arora, J. S. (2004). Survey of multi-objective optimization methods for engineering. *Structural and Multidisciplinary Optimization, 26*(6), 369–395.
Maturana, S., Pizani, E., & Vera, J. (2010). Scheduling production for a sawmill: A comparison of a mathematical model versus a heuristic. *Computers & Industrial Engineering, 59*(4), 667–674.
Mavrotas, G. (2009). Effective implementation of the ε-constraint method in multi-objective mathematical programming problems. *Applied Mathematics and Computation, 213*(2), 455–465.
Mavrotas, G., & Diakoulaki, D. (1998). A branch and bound algorithm for mixed zero-one multiple objective linear programming. *European Journal of Operational Research, 107*(3), 530–541.
Mavrotas, G., & Florios, K. (2013). An improved version of the augmented ε-constraint method (augmecon2) for finding the exact pareto set in multi-objective integer programming problems. *Applied Mathematics and Computation, 219*(18), 9652–9669.
Mesquita, M., Murta, A. G., Paias, A., & Wise, L. (2017). A metaheuristic approach to fisheries survey route planning. *International Transactions in Operational Research, 24*(3), 439–464.
Miller, C. E., Tucker, A. W., & Zemlin, R. A. (1960). Integer programming formulation of traveling salesman problems. *Journal of the ACM (JACM), 7*(4), 326–329.
Miranda-Ackerman, M. A., Azzaro-Pantel, C., & Aguilar-Lasserre, A. A. (2017). A green supply chain network design framework for the processed food industry: Application to the orange juice agrofood cluster. *Computers & Industrial Engineering, 109*, 369–389.
Moccia, L. (2013). Operational research in the wine supply chain. *INFOR: Information Systems and Operational Research, 51*(2), 53–63.
Mora, G. M. (2019). The chilean wine industry. *The Palgrave Handbook of Wine Industry Economics* (pp. 177–200). Springer.
Morande, H., & Maturana, S. (2010). Design and validation of a decision support system for oenologists. *Revista ICHIO, 1*(1), 46–58.
Moreno, E., Rezakhah, M., Newman, A., & Ferreira, F. (2017). Linear models for stockpiling in open-pit mine production scheduling problems. *European Journal of Operational Research, 260*(1), 212–221.
Overton, J., & Murray, W. E. (2011). Playing the scales: Regional transformations and the differentiation of rural space in the chilean wine industry. *Journal of Rural Studies, 27*(1), 63–72.
Palma, C. D., & Nelson, J. D. (2010). Bi-objective multi-period planning with uncertain weights: a robust optimization approach. *European Journal of Forest Research, 129*(6), 1081–1091.
Palma, C. D., & Vergara, F. P. (2016). A multiobjective model for the cutting pattern problem with unclear preferences. *Forest Science, 62*(2), 220–226.
Palmowski, Z., & Sidorowicz, A. (2018). Note on dynamic programming optimization for assigning pressing tanks at wineries. arXiv preprint arXiv: 1811.00469.
Petti, L., Raggi, A., De Camillis, C., Matteucci, P., Sára, B., & Pagliuca, G. (2006). Life cycle approach in an organic wine-making firm: an italian case-study. *Proceedings Fifth Australian Conference on Life Cycle Assessment, Melbourne, Australia* (pp. 22–24). .
Ponstein, H. J., Ghinoi, S., & Steiner, B. (2019). How to increase sustainability in the finnish wine supply chain? insights from a country of origin based greenhouse gas emissions analysis. *Journal of Cleaner Production, 226*, 768–780.
Rezaei-Malek, M., Tavakkoli-Moghaddam, R., Zahiri, B., & Bozorgi-Amiri, A. (2016). An interactive approach for designing a robust disaster relief logistics network with perishable commodities. *Computers & Industrial Engineering, 94*, 201–215.
Salamati-Hormozi, H., Zhang, Z.-H., Zarei, O., & Ramezanian, R. (2018). Trade-off between the costs and the fairness for a collaborative production planning problem in make-to-order manufacturing. *Computers & Industrial Engineering, 126*, 421–434.
Solanki, R. (1991). Generating the noninferior set in mixed integer biobjective linear programs: an application to a location problem. *Computers & Operations Research, 18*(1), 1–15.
Souza, G.d. S.e., & Gomes, E. G. (2015). Improving agricultural economic efficiency in brazil. *International Transactions in Operational Research, 22*(2), 329–337.
Ting, S., Tse, Y., Ho, G., Chung, S., & Pang, G. (2014). Mining logistics data to assure the quality in a sustainable food supply chain: A case in the red wine industry. *International Journal of Production Economics, 152*, 200–209.
Troncoso, J., D'Amours, S., Flisberg, P., Rönnqvist, M., & Weintraub, A. (2015). A mixed integer programming model to evaluate integrating strategies in the forest value chain—a case study in the chilean forest industry. *Canadian Journal of Forest Research, 45*(7), 937–949.
Vafaeenezhad, T., Tavakkoli-Moghaddam, R., & Cheikhrouhou, N. (2019). Multi-objective mathematical modeling for sustainable supply chain management in the paper industry. *Computers & Industrial Engineering.*
Valenzuela, L., & Maturana, S. (2016). Designing a three-dimensional performance measurement system (smd3d) for the wine industry: A chilean example. *Agricultural Systems, 142*, 112–121.
Varas, M., Basso, F., Lüer-Villagra, A., Mac Cawley, A., & Maturana, S. (2019). Managing premium wines using an (s-1, s) inventory policy: a heuristic solution approach. *Annals of Operations Research, 1–2*, 351–376.
Varas, M., Maturana, S., Cholette, S., Mac Cawley, A., & Basso, F. (2018). Assessing the benefits of labelling postponement in an export-focused winery. *International Journal of Production Research, 56*(12), 4132–4151.
Varsei, M., & Polyakovskiy, S. (2017). Sustainable supply chain network design: A case of the wine industry in australia. *Omega, 66*, 236–247.