Artificial Intelligence Research and Development
T. Alsinet et al. (Eds.)

doi:10.3233/FAIA240447

Optimization of Vineyard Task Planning: Constraint Programming Based Tool with Customizable Domain

Alex Pujol[a,1], Raimón Fabregat[a], Josep Pijuan[a], Marta Otero[a], Montse Torres[b], Antonio Dente[c], Fernando Alves[d]
[a] *Eurecat, Centre Tecnològic de Catalunya, Barcelona, Spain*
[b] *Familia Torres, Vilafranca del Penedès, Spain*
[c] *Mastroberardino, Atripalda (Avellino), Italy*
[d] *Symington Family Estates, Porto, Portugal*

Abstract. Producing grapes for wine is a complex and delicate process that requires meticulous execution of numerous tasks to reach the desired quality and yield. Wine producers dedicate many resources to specialized teams, from vine pruning in the early months of the year to harvesting when the fruit is ripe. Scheduling the available workforce and resources to complete these tasks optimally is of utmost importance. In this context, parting from an initial schedule could assist in the decision-making process and enhance the outcomes. With this objective, the Vitigeoss Business Service has been developed within the context of the Vitigeoss project. It is a generalized scheduling tool based on Constraint Programming, which assigns tasks to work teams on specific timeslots. For this, it uses First Fit Decreasing and Local Search as search algorithms. It also offers the capabilities of choosing between a set of constraints defined alongside wine producers, and focusing on one of three objective functions: time, cost, and environmental impact. In this document, a series of updates on the constraint decision process and customization of the domain are introduced, and its implications for the results obtained are analized. The Vitigeoss Business Service was available during the 2023 campaign, and its results have been validated with actual planning data from three use cases, showing improvements in the use of resources and finalization time.

Keywords. Optimization, Constraint Programming, Task scheduling, Cost Function

1. Introduction

In viticultural settings, the management of various field tasks poses significant challenges due to the diverse and time-sensitive nature of agricultural operations. From planting and soil management to harvesting and crop maintenance, the coordination of numerous tasks and work teams demands meticulous planning and execution in order to achieve cost-effective products. The complexity of these operations often leads to inefficiencies, including resource wastage, suboptimal utilization of labor, and increased production costs. Consequently, the need to optimize the scheduling of field tasks becomes paramount. By strategically allocating resources, prioritizing tasks, and minimizing non profitable time, optimized scheduling not only enhances productivity and resource utilization but also contributes to improved crop yields, reduced operational

[1] Corresponding Author: Alex Pujol Garcia, alex.pujol@eurecat.org.

costs, and overall sustainability in agricultural production. For this purpose, the Vitigeoss Business Service (VBS) has been developed [1] as an agriculture-focused task scheduler within the context of the VITIGEOSS project. This tool has been codesigned with winemakers from three European countries with different production objectives, processes, geographies, and weathers. Following this design, VBS has been developed, using the Optaplanner framework [2], with a set of base constraints, a set of optional constraints, and three cost function priorities, which offer winemakers the ability to customize the optimization problem to their own needs. The development of this tool has been based on assignment-based Constraint Programming (CP) [3] where, given a specific scenario, including tasks and work teams as well as a selection of constraints, the user obtains an optimized task schedule. Then, the process of selecting the best possible schedule is done using a combination of Construction Heuristics (CH) [4] and Local Search (LS) [5] algorithms. In this document, the latest updates of the VBS are presented; these include an improvement on the constraint definition process and some results regarding the 2023 campaign. The VITIGEOSS project counts three use cases related to the VBS for development and validation purposes. These use cases are harvesting, soil management, and shoot positioning, which provide a variety of characteristics different from each other.

2. Methods

The methodology employed for the implementation of the optimization tool revolves around the utilization of CP to generate a feasible schedule for a given scenario comprising a set of tasks and a set of work teams. The first step involves the formulation of the problem domain as a constraint satisfaction problem, wherein variables represent the assignment of tasks to specific timeslots and work teams. These variables are subjected to hard constraints encapsulating various aspects of the scheduling problem, such as task dependencies, team availability, and resource constraints. Furthermore, the methodology incorporates the definition of a cost function that quantifies the objective of the optimization process. The cost function encompasses the goals of the scheduling problem, which the user can choose to be either lower economical cost, lower environmental impact, or completion time. This cost function guides the solver searching for a solution that satisfies all constraints while optimizing the defined objective.

The VBS is designed to be a scheduling tool that can adapt to many viticulture-related processes, the variables that define the model are a set of working teams and their assignment to a set of tasks. To this end, a set of optional hard constraints has been defined to allow the customization of the problem domain. These constraints are defined in Table 1 and are divided into two categories: a) the base constraints that define the assignment of tasks to work teams on specific timeslots, and b) the optional constraints that allow the domain to be adapted to the problem by covering specific scenarios. The solution-searching process is divided into two steps: the first uses the First Fit Decreasing (FFD) construction heuristic to generate a non-random but unoptimized initial solution, and the second uses LS algorithms to obtain an optimized solution. The two LS algorithms used are Great Deluge (GD) and Tabu Search (TS) [5], the two LS algorithms are used in succession to balance their flaws and avoid falling into local optimal points. The number of iterations of this process depends on the run time selected by the user, if a long time is selected, the model has a higher chance of finding a better output. By following this methodology, the optimization tool leverages CP and a cost function to systematically generate schedules that allocate tasks to work teams within timeslots, thereby achieving good resource utilization and predefined scheduling objectives. The constraints added to the VBS tool since its initial version in 2022 are OC3, OC4 and OC5, these have been identified alongside the winemakers present on the project as important conditions for some of their problems.

Table 1. A list of the hard constraints used in the optimization model is divided into base and optional.

Constraint			Description
Base Constraints	BC1	Teams conflict	Two work teams can't be assigned to the same task
	BC2	Tasks conflict	A work team can't be assigned to two tasks on the same timeslot
	BC3	Capacity limit	A work team can't work each day more than its capacity
	BC4	Pre-assignments	Previously assigned tasks can't be reassigned to other teams or timeslots
Optional Constraints	OC1	Availability conflict	A work team or task can't be assigned on a day it has no availability
	OC2	User task conflict	A work team can't be assigned to a group of tasks selected by the user
	OC3	Task type conflict	A work team can't be assigned to a specific type of task that it is not specialized for
	OC4	global capacity limit	The daily global capacity limit can't be surpassed
	OC5	Numerical priority limit	A task can't be assigned before another task with a higher numerical priority

3. Results and discussion

The presented results are focused on the application of VBS in the harvesting use case since it is one that showcases the latest features added to the constraint definition process of the tool [1]. The resulting schedule for a set of 40 tasks each one with its own work team of different capacity (Kg/day) can be viewed on Figure 1. Bellow the schedule, a plot detailing the daily input for both types of tasks can be found.

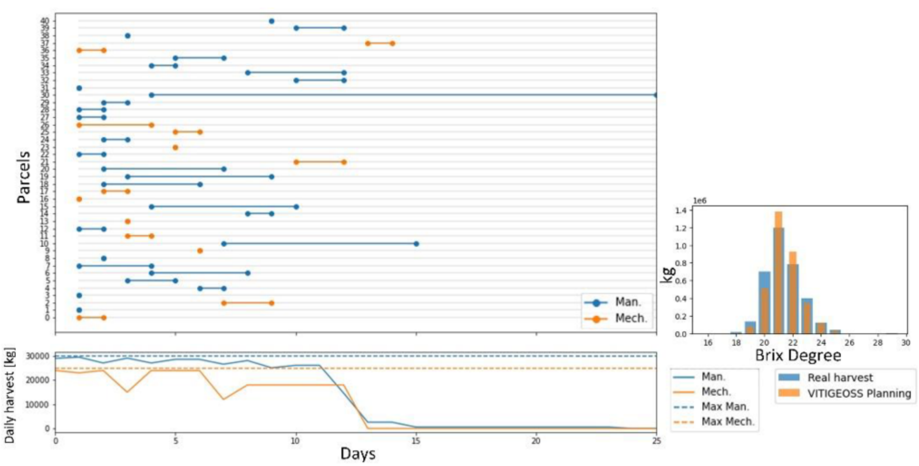

Figure 1. Harvest schedule for 40 tasks. Each task represents a parcel to harvest either by hand or with the use of machinery (top left), daily intake for each of the task types along with the daily max (bottom left), and brix degree comparison between real data (2023) and the degree simulated by VBS (right).

The first thing to notice from the resulting schedule is that the working capacity (kg/day) of each team and the magnitude (Kg) of the task determine the length of said task (BC3). An interesting example is task 30, which has a low-capacity team assigned and thus takes many days to complete. Besides the base constraints, the main factors that shape the resulting schedule is shaped by the daily maximum capacity (OC4) and the prioritization of those tasks with a higher priority (OC5), in this case the grape sugar content, brix degree. As shown in Figure 1, in the bottom left graph, the daily intake is close to the maximum permitted each day, and it decays over time as fewer products are available to harvest. Regarding the comparison between simulated and real data on the brix degree at which the product is collected, the results show that the schedule proposed by VBS can have more product collected around 21 degrees. More accurately, the simulated data has 16% more product collected in the range of [20 – 22] brix degrees, being 21 brix degrees the optimal harvesting point [6].

The resulting schedule offered by VBS is a simulation based on a model of a physical environment, which may have unpredictable factors unaccounted for that affect the schedule's application. Despite that, the results obtained provide the user with a feasible schedule that respects the constraints defined (brix degree) and is of good quality in terms of brix degree collection time. Also, the schedule provides information to the user on the flaws of their operation, allowing them to take preventive action beforehand.

4. Conclusion

This document summarizes the VBS tool's updates, including optional constraints that give users higher problem domain customization capabilities. It also contains a demonstration of the tool applied to a harvesting use case, in which the use of these optional constraints is displayed. Furthermore, the results obtained on the schedule regarding the harvest on the optimal brix degree show an improvement compared to the 2023 data.

Acknowledgments

The research leading to these results is funded by the European Union’s Horizon 2020 research and innovation programme under grant agreement No. 869565 – VITIGEOSS.

References

[1] Raimon Fabregat, et al. (2023). VITIGEOSS Business Service: Task scheduling optimization in vineyards. *IVES Conference Series, GiESCO 2023*.

[2] De Smet, G., & Contributors, O. S. (2006). Optaplanner user guide. Red Hat and the community. URL http://www. optaplanner. org. OptaPlanner is an open-source constraint satisfaction solver in Java.

[3] Rossi, F., et al. (2008). Constraint programming. *Foundations of Artificial Intelligence*, *3*, 181-211.

[4] Martí, R., Reinelt, G. (2022a). Heuristic Methods. In: Exact and Heuristic Methods in Combinatorial Optimization. Applied Mathematical Sciences, vol 175. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-662-64877-3_2

[5] Hentenryck, P. V., & Michel, L. (2009). *Constraint-based local search*. The MIT press.

[6] Sinton, T. H., et al. (1978). Grape juice indicators for prediction of potential wine quality. I. Relationship between crop level, juice and wine composition, and wine sensory ratings and scores. *American Journal of Enology and Viticulture*, *29*(4), 267-271.