2023

Optimised Routing of the Blueberry Cultivating Unmanned Ground Vehicle

Obrenović Nikola, Lalić Maksim, Stefanović Dimitrije, Panić Marko, Brdar Sanja, Crnojević Vladimir, Marko Oskar

OPTIMISED ROUTING OF THE BLUEBERRY CULTIVATING UNMANNED GROUND VEHICLE

Nikola Obrenović, Maksim Lalić, Dimitrije Stefanović, Marko Panić, Sanja Brdar, Vladimir Crnojević and **Oskar Marko**
University of Novi Sad, BioSense Institute
Dr Zorana Đinđića bb, Novi Sad, Serbia
E-mail: {nikola.obrenovic, lamaksim, dimitrije.stefanovic, panic, sanja.brdar, crnojevic, oskar.marko}@biosense.rs

Abstract: Precision treatment of blueberries has recently gained high importance due to the value of the crop. Such treatment is intended to be performed by unmanned ground vehicles (UGVs). However, the autonomy of UGVs is constrained by technological limitations, requiring their tasks to be executed in optimal routes. This paper presents an ALNS-based approach for the optimisation of UGV routes and its evaluation on problem instances based on an actual blueberry field. We also identify possible alternative solution methodologies that will be evaluated in the future research.

Keywords: Precision agriculture, UGV, blueberries, optimal routing, ALNS, TSP.

1 INTRODUCTION

Blueberries are a high-value crop which requires a careful and dedicated treatment. To address this need, the BioSense Institute is developing an unmanned ground vehicle (UGV) specialized in precision monitoring, soil sampling, and spraying in blueberry plantations. The UGV autonomy is of major importance as it determines the efficiency of UGV deployment. Hence, determining the near-/optimal UGV route for tasks execution, considering the observed blueberry field, its rows and characteristics, the requested task locations within the field, and the starting point of UGV's execution, is essential.

We particularly analyse the problems of precision spraying and soil sampling performed by the UGV. In both cases, the UGV is required to visit certain points in the blueberry field and perform spraying or soil sampling. To reach those points, the UGV can only move between blueberry rows, without crossing them, as shown in

Figure 1. Changing rows is only permitted at the ends of each row. Furthermore, there is a drainage trench, located adjacent to each field row, preventing the UGV from approaching the row from that side. The trench is represented with a zigzag line in
Figure 1. Finally, the UGV's camera for precise positioning is looking on the left-hand side, with respect to the UGV movement direction. The camera must face the targeted blueberry row to ensure accurate UGV positioning. Therefore, we aim to approach each spraying/sampling point (SSP) so that the camera is looking in its direction, thereby avoiding UGV rotation and additional power consumption.

Figure 2 illustrates a desired, optimised UGV path. The green points and purple diamonds denote the allowed and visited projections of SSPs to inter-row corridors. At the former, the UGV approaches the SPP with the camera facing it, while the latter denote locations where the UGV must rotate to achieve precise positioning. The dark green lines denote the travelled path. The black X marks represent the SSPs projections that are not allowed due to the presence of the drainage trench. The red points and dashed lines denote the inter-row corridors that are not utilised.

Since the UGV should visit every SSP in the near-/optimal manner, with respect to the abovementioned physical characteristics of blueberry fields and utilised UGVs, this problem represents a customised version of the travelling salesman problem (TSP). In this paper, we provide the details of the extended problem and our solution approach.

The remaining of the paper is organised as follows. In Section 2, we analyse the related work. Sections 3 shows the details of problem modelling and solution algorithm. Section 4 presents the results and evaluation of our solution approach. Finally, Section 5 gives the conclusions and our intentions for the further research.

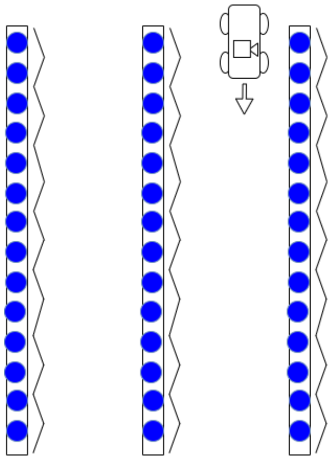

Figure 1: The appropriate UGV movement within the field rows.

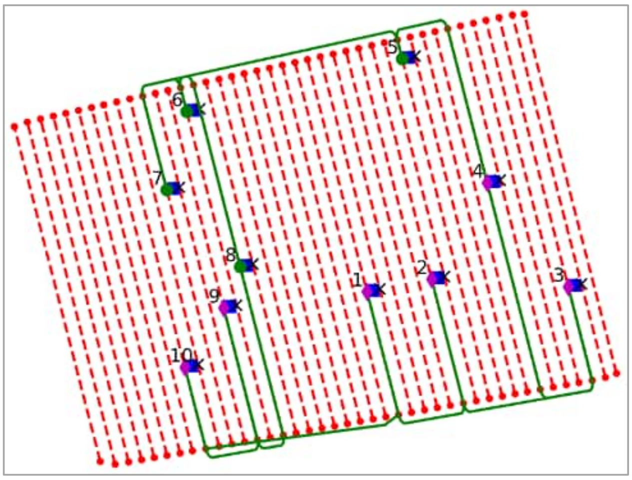

Figure 2: An optimised UGV route.

2 RELATED WORK

One of the most common approaches for TSP solution is the Lin-Kernighan's improvement heuristic, which belongs to the class of local search algorithms [3]. Here, the search starts from an arbitrarily selected tour, and, at each state, the algorithm tries to remove k tour arcs and insert k new ones to improve the solution cost. Such move is known as the *k-opt* move [3].

In general, metaheuristic algorithms such as simulated annealing (SA, [1]), tabu search [4], and genetic algorithms (GA, [2]) are also commonly employed to solve the TSP. Recently, adaptive large neighbourhood search (ALNS), a metaheuristic algorithm, has been used to solve the TSP with good results [5, 6]. This metaheuristic builds upon the SA framework and utilises a set of heuristic moves, called destroy and repair operators, to generate candidate solutions. The purpose of a destroy operator is to create a significantly different solution and hopefully push the search process into an unexplored part of the search space. Then, the repair operator is responsible for restoring the feasibility of the new solution, if needed. In each iteration, a pair of destroy-repair operators is randomly selected driven by their success rate up to that point, i.e., more successful operators have higher chances of being selected. If the operators create a better solution, their success rate is increased, otherwise, it is penalised.

Based on the benchmark tests, Laporte, Ropke, and Vidal suggest several other solution methods, such as GA, memetic algorithms, parallelised local search algorithms, and hybrid approaches combining GA and neighbourhood search [5]. These methods are located on the Pareto front representing the compromise between the average optimality gap and solution time. Consequently, all of them are considered good candidates for the solution of our problem.

3 SOLUTION METHODOLOGY

To account for the constraints additional to the original TSP, we perform specific data preparation. Each sampling point is projected onto a corresponding point in the inter-row corridor, on the opposite side of the drainage trench. The graph utilised in the TSP solving consists solely of these projection points. With that, we avoid approaching the rows from the side of the trench.

Rather than explicitly implementing the constraint that the precise positioning camera must look towards the sampled row, we solve the relaxed problem. Each rotation incurs a substantial penalising cost, which drives the search algorithm to avoid such situations. The benefit of such a solution approach is allowing the algorithm to investigate states, worse than the current solution, and escape local minima. Finally, the objective function accounts for an additional rotation cost at each location where the camera is faced opposite to the sampling point.

Thus, the mathematical model of our UGV routing problem is defined by extending the objective function of the original TSP model [1] with the cost of UGV rotations. The cost of travelled distance and UGV rotations are both expressed in the amount of utilised energy, in the same units, and combined in a single objective function with a plain sum. The model constraints are the same as in the original TSP model [1]. Due to limitations in paper length, we omit the mathematical formulation of the model, which is available upon request.

One of the simplest solution approaches for the TSP is the greedy algorithm [1]. In our case, the greedy algorithm starts at the SSP closest to the UGV's starting position. At each SSP, it selects the nearest unvisited SSP as the next destination. This process is repeated until all SSPs have been visited. Although it is quite naive, the greedy algorithm is capable of quickly finding the solutions that outperform than the man-made ones. In our solution approach, we utilise the greedy algorithm to generate an initial solution, which will be further improved.

Finally, we apply the heuristic based on the ALNS framework, starting from the initial solution. Following the findings of [3], we combine the pairs of destroy and repair operators into unified operators based on *k-opt* moves. In the current state of our research, we select *2-opt*, *3-opt*, and *4-opt* operators, while the *larger-k-opt* operators induce either a longer execution time without yielding better solutions, or even the algorithm being trapped in solution spaces distant from the best-found solution. We have also tested shuffle operators, which randomly permute subtours. However, these did not lead to any result improvements.

4 EXPERIMENT RESULTS

The ALNS parameter tuning included selecting the initial and final probabilities for accepting a worse solution, the strategy for temperature drop, the number of temperature drops, and the number of intra-temperature iterations. Figure 3 illustrates the achieved algorithm behaviour, demonstrating desired exploration and diversification at the beginning, followed by intensified local search in the most promising search region. The green line denotes the current solution, while the blue line represents the overall best-found solution.

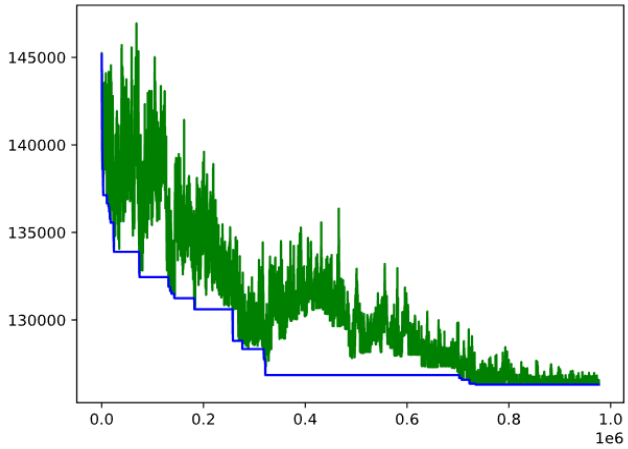

Figure 3: The desired algorithm execution. The x-axis denotes the iteration number, while the y-axis represents the objective function value.

For the algorithm evaluation, we have solved the three problem instances based on the model of a real blueberry field located in Babe, Serbia. To thoroughly assess algorithm's performance, the tested instances contain 10, 50, and 100 randomly generated SSPs. They are denoted UGV10, UGV50, and UGV100, respectively. Each problem instance is solved 50 times with randomly selected starting UGV locations around the field. Table 1 shows the statistics regarding the cost improvement and execution time for each instance. The tests were conducted on Intel(R) Core(TM) i5-9400 CPU @ 2.90GHz with 16 GB RAM.

Table 1. Developed algorithm solution statistics, i.e., average, minimal, and maximal solution improvement and time are shown for each problem instance.

Instance	*Average impr. [%]*	*Max. impr. [%]*	*Min. impr. [%]*	*Avg. sol. time [s]*	*Max. sol. time [s]*	*Min. sol. time [s]*
UGV10	16.47	22.58	12.52	81.43	90.91	79.67
UGV50	12.47	18.63	4.79	395.39	405.53	387.67
UGV100	7.76	11.22	4.26	759.68	776.95	749.06

5 CONCLUSIONS AND FUTURE RESEARCH STEPS

The evaluation of the current algorithm has demonstrated its suitability for the intended purpose of UGV routing optimisation. With the smaller number of SSPs, the cost improvement is significant while the running time is satisfactory. Moreover, the experiment results indicate that the running time increases linearly with the problem size, which is the same time complexity as obtained by the *k-opt* implementation by Helsgaun [3]. Furthermore, the current algorithm implementation uses only one CPU thread, which leaves space for further performance improvement.

In our future research, we intend to estimate the optimality gaps obtained by the current solution algorithm. Additionally, we aim to develop algorithms based on GA and hybrid approaches, as suggested in [5], and compare their performance with the presented algorithm in the observed case of UGV routing optimisation.

Acknowledgement

This work has been carried out in the scope of the H2020 FlexiGroBots project, which has been funded by the European Commission in the scope of its H2020 programme (contract number 101017111, https://flexigrobots-h2020.eu/).

References

[1] Bierlaire, M. 2015. Optimization: principles and algorithms. Lausanne, Switzerland: EPFL Press. ISBN 978-1482203455.

[2] Chatterjee, S., Carrera, C., Lynch, L. A. 1996. Genetic algorithms and traveling salesman problems. European Journal of Operational Research, 93(3): 490-510.

[3] Helsgaun. K. G. 2009. General k-opt submoves for the Lin–Kernighan TSP heuristic. Mathematical Programming Computation, 1: 119–163.

[4] Knox, J. 1994. Tabu search performance on the symmetric traveling salesman problem. Computers & Operations Research, 21(8): 867-876.

[5] Laporte, G., Ropke, S., Vidal, T. 2014. Heuristics for the Vehicle Routing Problem. In Toth, P., Vigo, D. (Eds.). Vehicle routing (pp. 87-116). Philadelphia, USA: Society for Industrial and Applied Mathematics.

[6] Smith, S. L., Imeson, F. 2017. GLNS: An effective large neighbourhood search heuristic for the Generalized Traveling Salesman Problem. Computers & Operations Research, 87: 1-19.