A Model for Planning TELCO Work-Field Activities Enabled by Genetic and Ant Colony Algorithms

João Henriques, Filipe Caldeira *

Informatics Department, Polytechnic of Viseu, 3504-510 Viseu (Portugal)
Department of Informatics Engineering, University of Coimbra, 3030-290 Coimbra (Portugal)
CISeD—Research Centre in Digital Services, Polytechnic of Viseu, 3504-510 Viseu (Portugal)

Received 5 April 2022 | Accepted 4 July 2022 | Early Access 19 August 2021

Abstract

Telecommunication Company's (TELCO) are continuously delivering their efforts on the effectiveness of their daily work. Planning the activities for their workers is a crucial sensitive, and time-consuming task usually taken by experts. This plan aims to find an optimized solution maximizing the number of activities assigned to workers and minimizing the inherent costs (e.g., labor from workers, fuel, and other transportation costs). This paper proposes a model that allows computing a maximized plan for the activities assigned to their workers, allowing to alleviate the burden of the existing experts, even if supported by software implementing rule-based heuristic models. The proposed model is inspired by nature and relies on two stages supported by Genetic and Ant Colony evolutionary algorithms. At the first stage, a Genetic Algorithms (GA) identifies the optimal set of activities to be assigned to workers as the way to maximize the revenues. At a second step, an Ant Colony algorithm searches for an efficient path among the activities to minimize the costs. The conducted experimental work validates the effectiveness of the proposed model in the optimization of the planning TELCO work-field activities in comparison to a rule-based heuristic model.

Keywords

Ant Colony, Genetic Algorithms, Route Optimization, TELCO.

DOI: 10.9781/ijimai.2022.08.011

I. Introduction

The TELCO are putting significant efforts into the optimization of their current operations in order to strengthen business competitiveness. One key aspect that should be addressed is optimizing the activities assigned to workers to be executed at remote locations. Preparing an optimized work plan is a challenging and complex task, even for specialists, maximizing the number of activities to be assigned to workers while keeping low as possible the use of resources, including labor, fuel, and vehicles. Despite this, computing an optimized plan is a time-consuming task taken by experts in a time-consuming iterative trial and error process, even if supported by software implementing rule-based heuristic models.

Workers are usually assigned to geographical areas, departing from their base stations to execute the planned activities within an expected period. Typically the base location is the location they return to at the end of the day. In order to assign activities to a worker, the skills required by the activities and the worker skills should match.

Thus, the availability of work at different locations is a crucial resource that organizations should carefully manage. They set up plans to optimize and maximize the number of activities for every workday. A solution must consider the time to run the foreseen activities, the duration and kilometers of the journey to and from each activity location, and the labor, fuel, and kilometers in maintenance.

* Corresponding author.

E-mail addresses: joaohenriques@estgv.ipv.pt (J. Henriques), caldeira@estgv.ipv.pt (F. Caldeira).

The quality of a candidate solution depends on the number of activities as revenue, while costs result from labor and the distance between different locations. Thus, a solution composed of revenue and cost means that it has a monetary value. Thus, it will be possible to compare the different solutions from other models despite their different nature and structure.

This work takes inspiration from nature to propose an optimization model for planning the activities of TELCO. The model incorporates heuristics as key knowledge retrieved from businesses to schedule workers' activities. The application of Evolutionary Algorithms (EA) is explored as an alternative to the common use of rule-based heuristic models supported in two stages. In the first stage, the best set of activities assigned to workers supported by the use of GA is selected. The second stage defines the order by which each activity should be executed according to their different locations by optimizing the distance the workers should run, supported by the use of Ant Colony Optimization (ACO). This optimization requires computing the best route to the different activities locations.

Beyond this section, section II presents the background and the key concepts. Section III presents the related work. Section IV, describes the implemented model. Section V presents the experimental work. Section VI discusses the achieved results. Section VII concludes the paper.

II. Background

This section provides the background on the adopted methodology to optimize the selection of the activities and the order to execute them. For that purpose, it is explored the use of Genetic Algorithms and Ant Colony algorithms.

A. Genetic Algorithm

The use of GA represents an alternative to the rule-based model by exploring the space of solution containing the optimized set of activities to be assigned to workers. Such activities result from the scheduling process that minimizes the duration of their activities.

The GA takes individuals containing the encoding solutions in chromosomes by evaluating the fitness function to create a new offspring supported by the crossover operator [1], [2]. The operation performs a random mutation when creating a new offspring. The chromosomes are commonly encoded as strings containing binary, real-valued, integer, octal, or hexadecimal numbers. Initially, a random population is created, providing an ample search space for potential solutions.

A fitness function scores the individuals in the current population, determining their survival for the next generation. The individuals returning the higher scores are the ones who are likely to be selected for the next generation. The individual score depends on how well chromosomes can solve the problem at hand, mostly done using probabilistic methods supported by evolutionary computing research, such as *roulette wheel, rank selection* and *tournament*.

B. Ant Colony Optimization Algorithm

Ant species can find the shortest path between a food source and the nest. The ACO simulates the behavior of ants as agents collectively searching the space for food while sharing information among them to achieve reasonable solutions [3]. Because they drop pheromones every time they bring food, shorter paths are more likely to have more significant amounts of pheromones, hence optimizing the solution. Ants select the next location to follow, depending on the distance and the amount of pheromone in the path. Ants have some properties like memory and sight. Their memory helps them to, among others, save the locations they visited, the distance they traveled, and the shortest path they saw. Sight allows them to know the possible end locations j and the distance d_{ij} to travel from their location at a given point i, with $j \neq i$. Ants are forced to make complete tours by maintaining the information about the previously visited locations in a tabu search [4].

Traveling Salesman Problem (TSP) aims to compute the optimized path considering the locations to visit. Due to this complex combination problem, a meta-heuristic approach optimization is used to find the shortest route from the nest to the food source.

The equation 2 provides the mathematical representation of the TSP problem, where n is the number of cities. The optimal solution of π with index nodes 1, 2, ..., n, such as the length of π is minimal and d is the distance between those nodes index, The $\Pi\{1, 2, ..., n\}$ indicates all the permutations 1, 2, ..., n [1]

$$\text{minimizef } (\pi) = \sum_{i=1}^{n=1} d_{\pi_i \pi(i+1)} + d_{\pi(n)\pi(1)}$$
$$\pi \in \Pi\{1,2,\ldots,n\} \tag{1}$$

The GA helps to replace the old ant's generation with the new one. Equation 2 describes the probability that ant k, located at node i, moves to node j, τ_{ij} is the pheromone level of edge (i, j), all taken at iteration t, and N_i is the set of one step neighbors of node i. While traversing an edge (i, j), the ant puts some pheromone on it, and the pheromone level of edge (i, j) is updated according to the following rule: $\tau_{ij}(t) \leftarrow \tau_{ij} + \Delta\tau$ where τ_{ij} is the iteration counter and $\Delta\tau$ is the constant amount of pheromone deposited by the ant.

$$P_{ij}^{k}(t) = \begin{cases} \dfrac{\tau_{ij}(t)}{\sum_{j \in N_i} \tau_{ij}(t)} \\ 0 \end{cases} \tag{2}$$

The best-expected solution should come in line with the short path length, driven by the process of releasing pheromones. Because the pheromone evaporates with time, links in longer routes will eventually contain much less pheromone linking the shorter tours.

Suppose n cities, with path p_{ij} between every pair of cities i, j. The length of p_{ij} is the distance between i and j, d_{ij}. The goal is to find path $P = p_{i_n i_1}$ where $(i_{1 \ldots n})$ is a permutation of (1; :::; n) to find out the shortest path to visit the cities exactly once time.

III. Related Work

This section presents some relevant literature in the area. Harada et. al. [5] surveyed the works in the domain of parallel genetic algorithms.

Li et. al. [6] proposed a multiobjective evacuation route assignment model to plan an optimal egress route set for the individual evacuees to minimize the total evacuation time, minimize the total travel distance of all the evacuees and minimize the congestion during the evacuation process.

Wang et. al. [7] used a GPU-adapted Parallel Genetic Algorithm to solve the problem of generating daily activity plans for individual and household agents.

Yang et. al. [8] proposed an Electric Vehicle (EV) route model considering the fast charging and regular charging under the time-of-use price in the electricity market. The proposed model aims to minimize the total distribution costs of the EV route while satisfying the constraints of battery capacity, charging time and delivery/pickup demands, and the impact of vehicle loading on the unit electricity consumption per mile.

Yang et. al. [9] proposed cooperative scheduling rules and defined the overlapping time between the accelerating and braking trains for a peak-hours scenario and an off-peak-hours scenario, respectively. They also formulate a programming model to maximize the overlapping time with the headway time and dwell time control. They designed a GA with binary encoding to solve the optimal timetable. In [10], they formulated a two-objective integer programming model with headway time and dwell time control. Second, we design a genetic algorithm with binary encoding to find the optimal solution.

Tsai et. al. [11] presented an algorithm for reducing the computation time of GA and its variants using the traveling salesman problem.

Wang et. al. [12] introduced the untwist operator to improve the performance of GA to shorten the length of the route and quicken the convergent speed.

IV. Proposed Model

This section presents the proposed model, including its structure and parameters, and describes the algorithm driving it.

The proposed model is applied in two stages, exploring the space of solutions to produce two different outputs. The first stage identifies a set of activities to be assigned to workers, supported by the GA algorithm.

The second stage optimizes the route for the set of activities selected in the first stage, supported by the use of the ACO algorithm. This work is inspired by observing the foraging behavior of ant colonies [13].

Several experiments evaluate the model's effectiveness and explore alternative configurations to produce the best results in the selection stage. Three experiments in the first stage try to find out the best selection operator from the set including *roulette*, *remainder*, *uniform*, *tournament* and *stochunif*.

The final experiment evaluates the outcomes from the second stage and the performance of the overall model. For that purpose, the different solutions are evaluated according to the number of activities assigned to workers and the number of kilometers to execute them.

A. Experimental Setup

This section presents the experimental setup with its data structures, datasets, and parameters.

Several structures, including matrices and vectors, helped explore and validate the model's behavior with short datasets generalizing their application to larger datasets. The matrix "CoordinatesWorkOrders" maintains the coordinates for activities to be scheduled. "Workers" is the matrix with the reference for workers and coordinates of worker homes.

The "activities" is the vector with references for all the activities to be scheduled by the model.

The parameters drive the GA algorithm at the first stage, aiming to identify the optimized set of activities and the ACO optimizing the path to execute those activities at the second stage.

The first stage of the model gathers the required parameters by the GA to implement some of the business rules. In the case of `ACTIVITIES_TO_EVAL` constrains the maximum number of activities to be executed by workers while computing feasible solutions. The number of genes `NUMBER_GENES` corresponds to the number of activities to be encoded by in matrix "coordenatesWorkOrders". Four genes were used to support the encoding until a maximum number of 16 different activities (1111). One of the critical issues to cope with the model is the travel time by road between two locations. The average speed parameter with kilometers per hour (`AVERAGE_SPEED`) is used for that purpose. The `COEFFICIENT_TOUR` parameter normalizes the Euclidean distance between Global Positioning System (GPS) coordinates in order to have an approximation of the distance by road. Attending the fact that distance by road is not euclidean, a coefficient can help to reduce the error. The `PENALTY` is the core parameter driving the score of the fitness function of different other parameters aiming to implement other heuristics. The `MIN_WORK_TIME` is the parameter controlling the minimum work time for workers and impacts negatively the fitness score by `PENALTY/10`. The working hours per day should fit in the range from `MIN_WORK_TIME` to `MAX_WORK_TIME`. In order to have control over the total number of daily activities assigned to a given worker in a single workday, the fitness function penalizes the solutions according to `MAX_WORK_HOURS`, set equal to `PENALTY/100`. The parameter `ACTIVITY_TIME` sets the duration of all the different activities in the model. The `MIN_ACTIVITIES` parameter penalizes over the minimum number of activities or locations set with penalty `PENALTY/20`. A solution including repeated activities should be classified as not valid and penalized according to `PENALTY`. Parameter `MIGRATION_FRACTION` sets the percentage of individuals returning the best scores to migrate to the next generation. The crossover fraction, defined by the parameter `CROSSOVER_FRACTION`, corresponds to the fraction of genes swapped between individuals. Parameter `ELITE_COUNT` sets the percentage of the best individuals surviving for the next generation without any change. The elite count is computed according to the product `ELITE_COUNT` with the size of the population.

The second stage of the model tries to find out the best path over the activities by using the ACO. The `MAX_I_TIME` parameter sets the maximum execution time in minutes. The `NUMBER_ANTS` parameter sets the number of ants included in the simulation and corresponds to the number of activities.

The configuration of the ACO parameters sets generally adopted values: alpha (pheromone influence factor) equals 1 , beta (heuristic information importance) equals 5 , and rho (pheromone evaporation coefficient) equals 0.65 . The parameter PLOT enables the graphical presentation of the progress of the model, including the GA and ACO algorithms.

Table I gathers the parameters and their settings in experiments.

TABLE I. Parameters Settings

Configuration	Value
ACTIVITY_TIME	1.5
ACTIVITIES_TO_EVAL	15
AVERAGE_SPEED	60
COEFFICIENT_TOUR	1.3
CROSSOVER_FRACTION	0.1
ELITE_COUNT	0.05
GENERATIONS	100
MAX_I_TIME	1000
MAX_WORK_TIME	9
MIGRATION_FRACTION	0.1
MIN_ACTIVITIES	3
MIN_WORK_TIME	8
NUMBER_ANTS	1000
NUMBER_GENES	4
PENALTY	10000
POPULATION_SIZE	1000
PLOT	1

B. Algorithm

The following algorithm defines the steps to lookup for a optimized solution S^2 including the activities $a_i \in A$ assigned to workers $w_i \in W$, according to the following steps:

Algorithm 1: Model (P_{ga}, P_{aco}, $W, A, F, N, \Phi, \gamma, \alpha, \beta$)

INPUT:

P_{ga}, GA Parameters
P_{aco}, ACO Parameters
W, Workers
A, Activities
F, GA Fitness Function
N, Number of activities
Φ, Sort the activities by distance
γ, Extracts N activities
α, Extracts location
β, Computes Distance

for all $w_i \in W$ **do**
  for all $a_i \in A$ **do**
    $L_{w_i} \leftarrow \alpha(w_i)$
    $L_{a_i} \leftarrow \alpha(a_i)$
    $D_{w_i,a_i} \leftarrow \beta(L_{w_i}, L_{a_i})$
  end for
  $O^1_{w_i} \leftarrow \Phi(D)$
  $O^0_{w_i} \leftarrow \gamma(O^1_{w_i}, N)$
  for all $a_i \in O^0_{w_i}$ **do**
    $S^0_{w_i} \leftarrow S^0_{w_i} \cup a_i$
  end for
  $S^1_{w_i} \leftarrow GA(S^0_{w_i}, F, P_{ga})$
  $S^2_{w_i} \leftarrow ACO(S^1_{w_i}, P_{aco})$

OUTPUT: S^2, List of activities assigned to workers

C. GA Solution Encoding, Decoding and Fitness Function

This work adopted a binary scheme to encode individuals in the population into binary chromosomes (`POPULATION_SIZE`). Each individual denotes a candidate solution, carrying out a set of chromosomes as the set of activities assigned to workers. Each individual represents a activities assigned to workers. Each individual represents a the worker.

An individual gathers 60 chromosomes to encode a set of 15 (`ACTIVITIES_TO_EVAL`) different activities while encoding each requires four chromosomes. Thus, a solution gathering the activities identified by indexes 2, 6, 4, 1, 3 and 8 can be encoded as an individual as follows: [2 6 4 1 3 0 0 0 0 0 0 0 0 0 8] is codified in binary as [[0010] [0110] [0100] [0001] [0011] [0000] [0000] [0000] [0000] [0000] [0000] [0000] [0000] [0000] [0111]]. Is also important to realize that alternative solutions such as [2 6 4 1 3 0 0 0 0 0 0 0 0 0 8], [3 4 0 0 0 8 0 6 0 1 0 0 2 0] and [2 4 0 0 0 8 0 6 0 1 0 0 0 3] have the same score. In such encoding, scheme value 0 has no meaning.

V. Experimental Work

This section presents the experimental work comprising a set of experiments evaluating the effectiveness of the proposed model and describes the structure of the solution and datasets. It discusses the implementation aspects, the involved datasets, and the settings.

A. Experimental Setup

MATLAB provided the package tools supporting the implementation of the proposed model with Genetic and Ant Colony algorithms. The quality of the solutions provided by individuals is scored by the GA fitness function of its Global Optimization Toolbox to lookup for a global minimum.

Table II presents the list of implemented MATLAB modules and describes their role.

TABLE II. MATLAB Model Modules

Module	Role
TELCO	Core application
FitnessTELCO	Fitness function
ACOTELCO	ACO implementation
Lldistkm	Distances in kilometres
DecodeWorkOrder	Decodes a binary solution
Assesstour	Validates a solution

The parameters settings adopted along the several experiments are defined according to the following Table I. Several experiments have been taken to explore these settings returning, including the maximum number of work hours" and "maximum activities to visit". The Fitness function $F()$ scores the best solutions as the ones not including repeated activities.

Fig. 1 depicts the progress with the scores of the fitness function along 100 generations and the information regarding the stall able to stop the algorithm's execution.

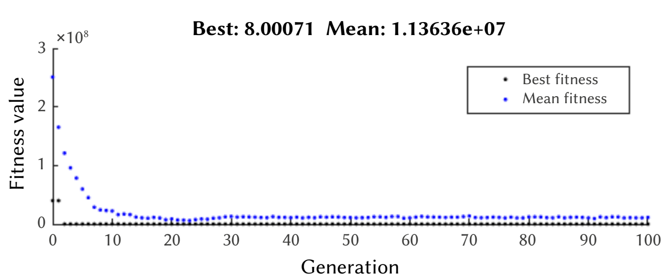

Fig. 1. Genetic Algorithm Evolution.

B. Datasets

Datasets for workers and activities are inputs to the model to compute the solutions. They include, respectively, the locations for worker base stations and locations where activities take place as GPS coordinates.

To support the following four experiments were used two different datasets. The purpose of the first dataset was to explore the effectiveness of the settings for the parameters containing 133 activities and 13 workers. To that aim, three experiments explored the most suitable GA operators in the selection of the activities along the first stage of the model.

A second larger dataset contains 436 activities and 191 workers. The fourth experiment uses this dataset to compare the quality of the solution produced by the proposed model with those produced by rule-based heuristic algorithms.

C. Experiments

The implementation of the proposed model is supported by the MATLAB toolbox environment providing EA. In order to demonstrate the capabilities for solving either complex or significant problems in the second stage of the model supported by the use of an ACO algorithm, a solution was computed from the second dataset according to Fig. 2.

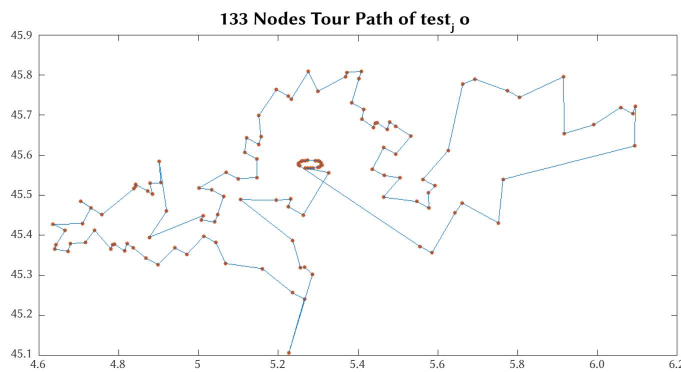

Fig. 2. ACO Scheduling Solution.

1. Experiment One

This experiment investigated the settings regarding the first stage of the proposed model. It relies on the use of the GA, including the genetic operator, the size of the population (`POPULATION_SIZE`), and the number of generations `GENERATIONS`. To that aim, the first dataset helped to explore the effectiveness of five different genetic selection operators.

The time to complete this experiment was 29338.619718 seconds (almost 8 hours). Table III summarizes the statistical results for the best GA scores for operators. From these results, it was possible to conclude that the tournament operator provided the best scores while reducing the number of individuals has a significant impact on the running time.

TABLE III. Experiment One Statistics

GA Operator	Mean	Min	Max	Std Dev.
Roulette	9.7229	8.0451	19.7617	3.4804
Remainder	10.6456	8.0411	23.8486	4.5726
Uniform	17.9256	8.0411	79.7528	16.2499
Tournament	9.4802	8.0414	79.7528	3.2803
Stochunif	9.8770	8.0576	19.9788	3.6922

Reducing the population (`POPULATION_SIZE`) from 1000 to 100 and the generations from 100 to 20 (parameter GENERATIONS significantly reduced the running time to 3825.188944 seconds (almost 1 hour).

2. Experiment Two

Departing from the already identified Tournament operator in the first experiment, this second one explored the effectiveness of its use with the GA. The number of generations gENERATIONS was set to 100 , and the first dataset was kept.

So, as a result, the score for the average value increased to 10.1522 while the computing time took 25838.210946 seconds (almost 7 hours). Table IV summarizes the achieved results.

TABLE IV. Experiment Two Statistics

GA Operator	Mean	Min	Max	Std Dev.
Tournament	10.1522	8.0406	28.2251	4.2976

3. Experiment Three

A third experiment explored the ability of the model to produce an effective plan in the shortest possible time. Thus, the first dataset (133 activities and 13 workers) and Tournament operator was selected, while the size of the population (`POPULATION_SIZE`) was reduced from 1000 to 100 individuals.

This experiment took 858.045974 seconds to run. Table V summarizes the statistical results from this experiment.

One of the key aspects regarding the quality of the solution comprises the number of activities assigned to workers. In this case, the number of assigned activities was 78 , and the non-assigned activities were 55 . Therefore, the ratio for activities assigned per worker was 6.

TABLE V. Experiment Three Statistics

GA Operator	Mean	Min	Max	Std Dev.
Tournament	8.0041	8.0000	8.6392	0.1753

4. Experiment Four

This last experiment explored a larger space of solutions and compared the results of this model with the ones from the heuristic rule-based model. For that purpose, a second dataset was used while the configurations from the experiment were maintained. As already stated, the second dataset includes 436 activities and 191 workers.

Computing the results took 3658.5085 seconds (almost 1 hour) to assign all the activities to 71 workers (100%), resulting in 6.1408 activities per worker. The kilometers needed for executing the activities assigned to the workers was 33859, and the average of kilometers per worker was 476.8858. The scoring average was 179.7771, which is worst compared to the optimal value of 8, achieved in the previous experiment.

In the case of the rule-based model, the assigned activities were 95 for 21 workers, while the non-assigned activities were 341 (21.79%). The activities per worker were 4.52, and the kilometers to execute the activities was 3315 (184.17 kilometers per worker).

VI. Results

Fig. 3 denotes the evolution of the fitness function $F()$ along several generations. The high scores at the beginning rapidly decrease and converge to a global minimum. Therefore, this global minimum corresponds to the available working hours, denoted by parameter `MIN_WORK_TIME`. Fig. 4 depicts the scores for the individuals contained in a given generation. Fig. 5 depicts the standard normal density function from the function operator tournament. The observed median value was 9.4802 while the standard deviation was 3.2803. Fig. 6 depicts the GA fitness scores for 13 different workers. Fig. 7 summarizes the online analysis for the best, worst and mean scores. Fig. 8 presents the computed path from the ACO algorithm with the set of activities assigned to a worker.

VII. Discussion

The experimental work evaluated the proposed model aiming to optimize the plan of activities for TELCO sector supported by GA and ACO along their first and second stages, respectively. The proposed model explored the space of solutions to maximize earnings, trying to discover a large possible number of activities to be assigned to workers and minimizing costs in terms of distance to perform the activities.

In that regard, the GA algorithm always tries to assign the maximum number of activities to workers (earnings), even if the number of kilometers and time are high (costs). The best solution (high score) from the first stage, supported by the use of GA, was achieved within twenty generations (parameter GENERATIONS). The computation time increases linearly with the number of generations. The fitness function penalizes the candidate solutions when they include non-singular activities. The best solutions are the ones that score the minimum number of hours in a day of work (8).

The results from the first three experiments denote the GA selection

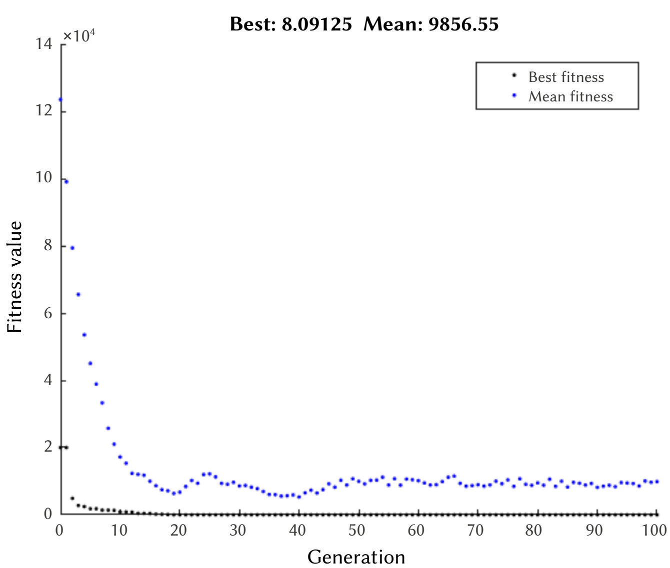

Fig. 3. Fitness Values Over Generations.

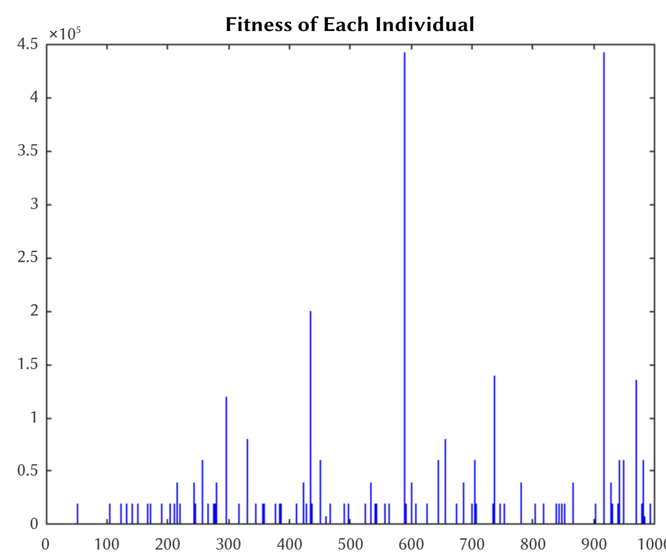

Fig. 4. Individual Fitness Function.

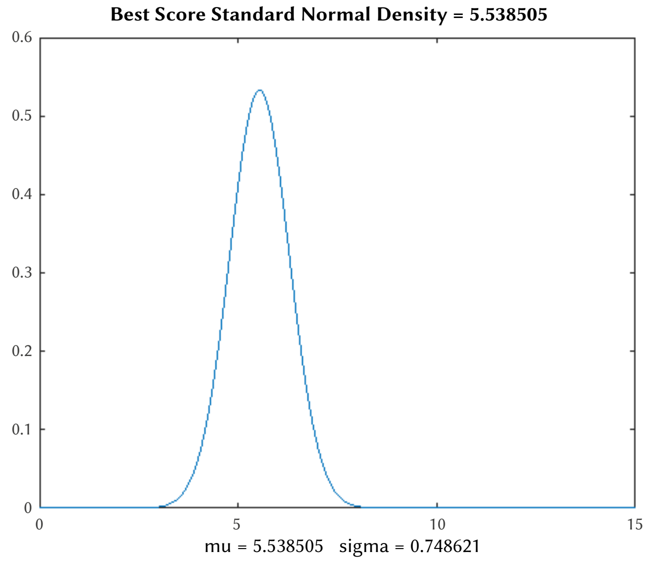

Fig. 5. Standard Normal Density Tournament Operator.

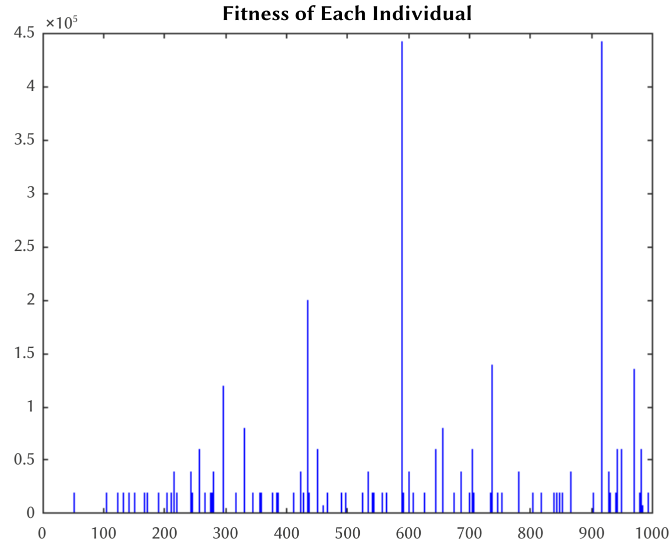

Fig. 6. GA Fitness Function Evolution.

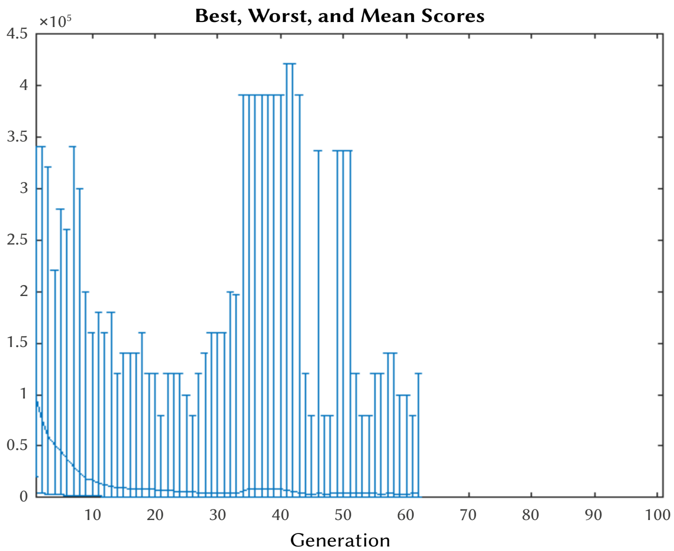

Fig. 7. GA Best Worst and Mean Values.

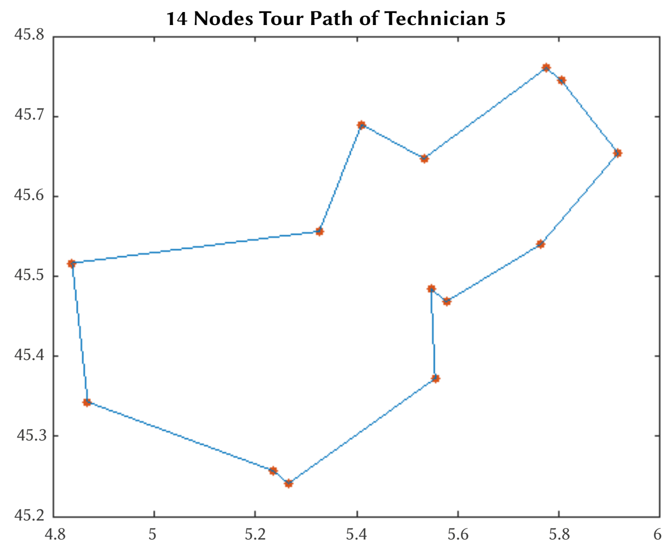

Fig. 8. ACO Activities Route.

operator tournament as the one providing the best scores. In addition, increasing the number of generations reduces the fitness function's average score. The best fitness score average was around 8 , according to the number of work hours in a single working day set by parameter `MIN_WORK_TIME`.

In the last experiment, a real dataset supports the comparison between the performance of the proposed model and the ruled-based model. All the activities were assigned to all the available workers, but at a cost. These results denote an acceptable solution despite the better performance of the ruled-based model regarding the ratio of activities per worker. It was also possible to depict a relevant number of workers assigned to a low number of activities. The cause is that the dataset for activities includes a significant number of activities distributed over vast regions.

From the analysis of experiment four, it was noticed that the distribution ratio per worker is significantly higher in comparison to the distribution provided by the rule-based model.

From the analysis of experiment four, it was noticed that the distribution ratio per worker is significantly higher in comparison to the distribution provided by the rule-based model.

One of the significant benefits of the model is that it does not require expert knowledge or a significant amount of effort in configuration activities to achieve an accurate solution. Thus, it is suitable to be applied at scale to larger datasets while requiring a minimum effort.

VIII. Conclusion

This work proposed a model inspired by nature to optimize the plan of the activities assigned to workers in the TELCO sector. A significant benefit of this approach comes from its effectiveness and reduced effort and time to compute a good solution, replacing the knowledge and work from experts. The model was supported by the use of Genetic Algorithms and Ant Colony Optimization Evolutionary Algorithms, and represents an alternative to the existing rule-based ones. The experimental results suggest that the model offers the foundation for its application in different use cases requiring the optimization of work-field services.

As the actual distance between the different locations is by road and not Euclidean, a regularization factor was introduced to diminuish the distance error, while future work can consider the use of distance. Future work will also pursue solutions for increasing the ratio of assigned activities per worker. Moreover, future work will seek for a fitness function returning the earnings in Euros allowing an improved comparison in terms of quality to the different models.

Acknowledgment

This work was partially funded by the European Social Fund, through the Regional Operational Program Centro 2020, within the scope of the projects UIDB/05583/2020 and CISUC UID/CEC/00326/2020. Furthermore, we would like to thank the Research Center in Digital Services (CISeD) and the Polytechnic of Viseu for their support

References

[1] T. R. Cunha A. G., A. C. H., "Manual de computação evolutiva e metaheurística," in *Imprensa da Universidade de Coimbra, pp. 87-105*.

[2] E. Bonabeau, G. Theraulaz, J.-L. Deneubourg, S. Aron, S. Camazine, "Self-organization in social insects," *Trends in Ecology & Evolution*, vol. 12, no. 5, pp. 188–193, 1997.

[3] T. R. Cunha A. G., . Antunes C. H., "How trail laying and trail following can solve foraging problems for ant colonies," in *Behavioural Mechanisms of Food Selection, NATO-ASI Series, G20, Springer-Verlag, pp. 661–678*.

[4] M. Dorigo, T. Stutzle, "Ant colony optimization," in *MIT Press*.

[5] T. Harada, E. Alba, "Parallel genetic algorithms: a useful survey," *ACM Computing Surveys (CSUR)*, vol. 53, no. 4, pp. 1–39, 2020.

[6] Q. Li, Z. Fang, Q. Li, X. Zong, "Multiobjective evacuation route assignment model based on genetic algorithm," in *2010 18th International Conference on Geoinformatics*, 2010, pp. 1–5, IEEE.

[7] K. Wang, Z. Shen, "A gpu-based parallel genetic algorithm for generating daily activity plans," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 3, pp. 1474–1480, 2012.

[8] H. Yang, S. Yang, Y. Xu, E. Cao, M. Lai, Z. Dong, "Electric vehicle route optimization considering time-of-use electricity price by learnable partheno- genetic algorithm," *IEEE Transactions on Smart Grid*, vol. 6, no. 2, pp. 657–666, 2015, doi: 10.1109/TSG.2014.2382684.

[9] X. Yang, X. Li, Z. Gao, H. Wang, T. Tang, "A cooperative scheduling model for timetable optimization in subway systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 1, pp. 438–447, 2012.

[10] X. Yang, B. Ning, X. Li, T. Tang, "A two-objective timetable optimization model in subway systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 5, pp. 1913–1921, 2014.

[11] C.-W. Tsai, S.-P. Tseng, M.-C. Chiang, C.-S. Yang, T.-P. Hong, "A high-performance genetic algorithm: using traveling salesman problem as a case," *The Scientific World Journal*, vol. 2014, 2014.

[12] L.-Y. Wang, J. Zhang, H. Li, "An improved genetic algorithm for tsp," in *2007 International Conference on Machine Learning and Cybernetics*, vol. 2, 2007, pp. 925–928, IEEE.

[13] Dorigo, C. D., "The ant colony optimization meta- heuristic," in *New Ideas in Optimization, McGraw-Hill, pp. 13–49*.

João Henriques

João Henriques is a PhD candidate in Science and Information Technology at the University of Coimbra (UC) and Adjunct Professor at the Department of Informatics Engineering at the Polytechnic of Viseu (IPV). His research interests at the Center for Informatics and Systems (CISUC) at C includes forensic and audit compliance for critical infrastructures protection. He also remains as Software Engineer in the private sector.

Filipe Caldeira

Filipe Caldeira is an Adjunct Professor at the Polytechnic Institute of Viseu, Portugal. He is a researcher at the CISeD research centre of the Polytechnic Institute of Viseu and at the Centre for Informatics and Systems of the University of Coimbra. His main research interests include ICT security, namely, trust and reputation systems, Smart Cities and Critical Infrastructure Protection. His research papers were published in various international conferences, journals and book chapters. He has been recently involved in some international and national research projects.