

# Improving unloading time prediction for Vehicle Routing Problem based on GPS data

Emir Žunić<sup>1,2</sup>, Amar Kurić<sup>3,1</sup>, Sead Delalić<sup>3,1</sup>

<sup>1</sup>Info Studio d.o.o., Sarajevo, Bosnia and Herzegovina

<sup>2</sup>Faculty of Electrical Engineering at University of Sarajevo, Bosnia and Herzegovina

<sup>3</sup>Faculty of Science at University of Sarajevo, Bosnia and Herzegovina

emir.zunic@infostudio.ba, amar.kuric@pmf.unsa.ba, delalic.sead@pmf.unsa.ba

**Abstract**—The problem of transport optimization is of great importance for the successful operation of distribution companies. To successfully find routes, it is necessary to provide accurate input data on orders, customer location, vehicle fleet, depots, and delivery restrictions. Most of the input data can be provided through the order creation process or the use of various online services. One of the most important inputs is an estimate of the unloading time of the goods for each customer. The number of customers that the vehicle serves during the day directly depends on the time of unloading. This estimate depends on the number of items, weight and volume of orders, but also on the specifics of customers, such as the proximity of parking or crowds at the unloading location. Customers repeat over time, and unloading time can be calculated from GPS data history. The paper describes the innovative application of machine learning techniques and delivery history obtained through a GPS vehicle tracking system for a more accurate estimate of unloading time. The application of techniques gave quality results and significantly improved the accuracy of unloading time data by 83.27% compared to previously used methods. The proposed method has been implemented for some of the largest distribution companies in Bosnia and Herzegovina.

**Keywords**— GPS tracking, GPS data analysis, Vehicle Routing Problem, Machine Learning

## I. INTRODUCTION

THE problem of transport routes optimization and optimal utilization of the transport fleet is a researched problem which is constantly and continuously researched due to its importance. The Vehicle Routing Problem (VRP) is a class of problems in which it is necessary to find the optimal route by which a vehicle from an available vehicle fleet visit the number of customers (delivery points), starting from the central depot and returning to the same location after the completion of customer service. The optimal route is primarily the one with the minimum cost of serving all customers [1]. These optimization problems become extremely complex if a large number of customers need to be served. If many real limitations are added, such as customer time windows, goods unloading times, ways of packing goods in vehicles, predefined capacity, working hours and diverse vehicle fleet, fixed and variable vehicle costs, these problems become a real challenge to solve. These limitations drastically reduce the number of available approaches, models and algorithms that could be applied to a complex set of input data.

With the progress of logistics processes in the early 1950s [1], there has been a large amount of research focusing on their

various applications. The importance of logistics management has grown significantly in recent years with the globalization of this process. Logistics tries to optimize existing distribution processes. One of the most important elements in logistics chains is the transport system. According to numerous studies, transport accounts for a third of the total logistics cost, and transport systems significantly affect the performance of the complete logistics system. Transport is necessary in the complete process of production of goods, from production to delivery of goods to end customers. Only in the case of good coordination between all components it is possible to get the maximum benefit for distributors and manufacturers. Without a well-developed transport system, logistics planning cannot reach its full potential. Therefore, it is indisputable that quality transport systems can increase efficiency, reduce operating costs, and increase the quality of service. Success in solving the problem of vehicle routing can significantly improve the processes in the transport part of each company's business.

In order for any system for optimization of transport routes to be usable by a certain company, it is necessary to determine the appropriate input parameters of the algorithm (as accurately as possible), such as parameters for customers, vehicles, restrictions, etc. One of the most important parameters that is very difficult to determine is the time of service (unloading of goods) for each customer. For these purposes, an approximate method of determining the time of unloading was proposed in [2] [3], which is based on a relation that depends on the number of ordered items, weight and volume of the customer's order. However, a real example of a distribution company has shown that this parameter can vary significantly, and that it can be determined more accurately based on the analysis of available historical GPS data and modern machine learning methods. The time of unloading is determined more realistically for each customer, whereby several modern machine learning algorithms have been implemented. After that, the voting system selects the results with the highest reliability. The resulting unloading time then becomes the part of the implemented transport optimization system, and the final transport routes are fully feasible in a real environment, which is the most important fact for any company whose transport is an integral segment. These facts were the motivation to implement several algorithms for more accurate determination

of unloading time, and compare the results. On the other hand, in most scientific research in the field of solving VRP problems, this parameter is taken as known in advance, which is not a common case in the real-world application.

The paper consists of five sections. In this first section, an introduction is presented with a defined problem and motivation for solving it. Section 2 presents a detailed review of the research literature. In the third section, the implementation and used techniques are presented and described in detail, while in the fourth section, the obtained results are presented. In the last section, the conclusions of the paper are presented, as well as guidelines for future research.

## II. RELATED WORK

The Vehicle Routing Problem belongs to a class of NP-hard problems. This means that no deterministic algorithm will provide an optimal solution in real time. Over the years, many heuristic algorithms have been developed for different variations of the problem. The paper [4] describes the results of research over 50 years in the field of vehicle routing. In [5], different metaheuristic approaches for solving the VRP have been described. In papers [6] and [7], recent progress in the field is described. Many different approaches and algorithms have been widely used, such as: Simulated annealing [8], Tabu search [9], Genetic algorithm [10], Bat algorithm [11], Firefly algorithm [12] etc. When solving the vehicle routing problem, it is necessary to adjust a number of parameters and input data for the algorithm, which can be a difficult problem for real application [3].

In the paper [13], the concept of a smart warehouse management system is described, which is based on a series of optimization algorithms and the application of historical data to improve business. However, for the successful operation of distribution companies, it is necessary to optimize the operation of all segments, which especially refers to transport as one of the most expensive operations.

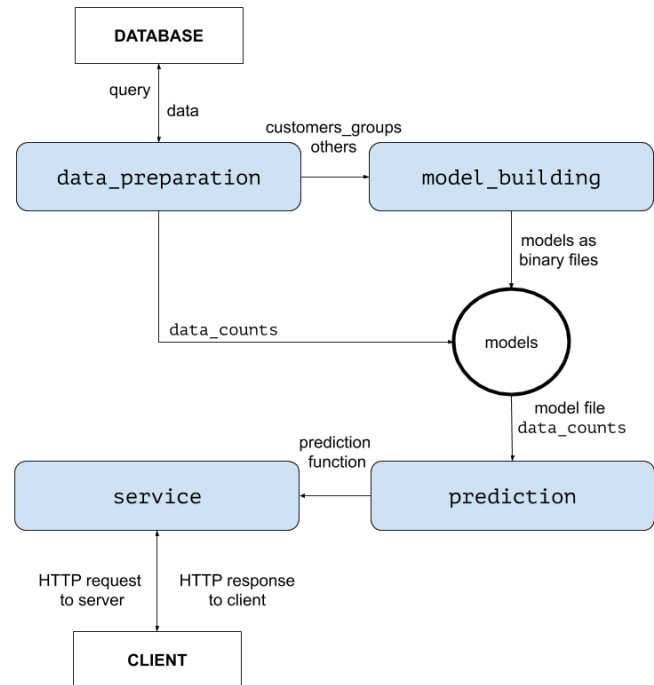
The idea of using GPS and geographic data from history for optimization processes permeates other scientific and industrial fields, such as the music and event planning industry [14], [15], social event detection [16], or even coronavirus tracking [17] etc.

In [18], the methods for improving the performance of the vehicle routing algorithms based on the GPS data have been described. In the same paper, the algorithm for using the GPS data to detect deliveries is described. When using the GPS data, it is necessary to ensure the accuracy. In [19], the innovative algorithm for anomaly detection in GPS data is described. The algorithm is inspired by the QRS complex detection algorithms in ECG signals.

## III. CASE STUDY

The desired functionality is realised through four stages, each of which is delegated to a separate module: `data_preparation`, `model_building`, `prediction`, `service`. The figure 1 contains a diagram describing the means by which these components interact

with one another as well as with the database and the client utilizing the service. What follows afterwards is an overview of their implementation and their respective roles in the system.



**Figure 1:** Schematic depiction of application workflow

The `data_preparation` step is mainly responsible for fetching required data from the database. Each fetched record contains the following for a particular delivery: customer's numeric ID, number of articles, total weight, total volume and time of unloading. A delivery fulfills a customer's order. Orders are also present in the database. Every order contains one or more articles each having a weight and a volume. The total weight of a delivery is the sum of the individual weights of all articles within the corresponding order. The total volume is obtained in the same manner. The unloading time is calculated based on the GPS history data. The algorithm calculates the length of all stops in customer's range. If one stop belongs to more than one customer, the unloading time is proportionally divided by the estimated unloading time for each customer in range. The detailed algorithm for unloading time detection from GPS data is described in [18].

Since inspection of data uncovered a number of undoubtedly incorrect records, they are removed from the pool using a criterion of either an impossibly low weight-to-volume ratio or an impossibly high time of unloading. After this, the number of records for each respective customer is extracted and saved as `data_counts`. In addition, this module defines the minimum number of data points a customer must have available for a separate model to be built for them. All customers with less data points are aggregated in the `others` object, so a single model can be built. This choice was made to

prevent building models on too few data points. The grouping of the original query result by customer ID is stored in `customers_groups`.

The `model_building` module builds required models based on data processed in the previous stage. Algorithm 1 outlines the procedure.

---

**Algorithm 1:** Model-building
 

---

```

estimators=estimators to test;
params=parameters to test for each estimator;
foreach customer ∈ customers_iter do
  cv=appropriate cross-validation method;
  scoring=neg_root_mean_square_error;
  best_estimator=None;
  best_score=-∞;
  scaler=None;
  foreach est ∈ estimators do
    current_scaler=None;
    if est needs scaling then
      current_scaler=scaler object;
      scale inputs using current_scaler;
      find parameter from params[est] which
        maximises scoring metric;
      current_best=score for found parameter;
      if current_best>best_score then
        best_estimator=est;
        best_score=current_best;
        scaler=current_scaler;
    end
  save best_estimator, scaler, best_score to file in
  "models" directory;
end

```

---

At the time of writing, estimators being considered are linear regression, ridge regression, lasso regression,  $K$  nearest neighbors regression, random forest regression and SVM regression (using linear kernel).

Cross-validation method is selected based on the number of data points for a given customer: leave-one-out is used if less than 60 are available and 10-fold cross-validation otherwise. Root mean square error is chosen as a criterion to compare models for two reasons. First, the objective is to maximise predictive power. Second,  $r^2$ , a frequently used scoring metric, is a measure of the reduction in variance accounted for by the independent variables and is, as such, extremely sensitive to outliers at low numbers of data points. Moreover, given that certain models require all input variables to be equally scaled, it is necessary to perform normalization prior to model evaluation. Several known normalization methods were tested during development leading to a conclusion that this selection has no significant impact on model performance. A method which utilizes quartiles was chosen due to its resilience to outliers. The reason for not fully removing outliers is the difficulty of detecting them caused by the absence of sufficient data points per customer necessary to assess the distribution

of input variables. The large quantity of customers in the database prompt the model-building process to be a lengthy one. Hence, it is scheduled to automatically execute on a weekly basis, and the resulting models are stored as binary files. The accompanying normalizing method, as well as the root mean squared error and the  $r^2$  score, are stored along with the best-performing model for each customer.

The `prediction` module implements a prediction function which, for a given customer ID, number of articles, total weight and volume of delivery, returns the predicted value of unloading time, the number of data points the prediction is based on and a 1-10 score quantifying the confidence in the prediction. Number of data points is obtained from the `data_counts` file. This is done since records generated later than the most recent model building do not need to be considered. The corresponding model object is obtained from its binary file. Normalization is performed on passed arguments if demanded by the model. Confidence score is calculated based on the  $r^2$  and root mean squared error also contained in the model object in the following manner. The intervals  $[0, 1]$  and  $[0, 12]$  are divided linearly into 10 segments. The  $r^2$  score is assigned a 1-10 score on the basis of the segment it belongs to. Similarly, the root mean squared error is assigned a 1-10 score as well. In both cases, a greater score corresponds to a better performing model. These two scores are then averaged (at the time of writing, both are assigned an equal weight of 0.5) which produces the final confidence score. Confidence is established by the use of two different evaluation metrics since inspection of results discovered that, while both are indicative of predictive power, they do not necessarily correlate with one another.

The final component (`service`) provides an interface for services which other components can use. A HTTP request is made containing the customer ID, number of articles, total weight and volume of delivery as parameters. The response contains the result of the prediction function in JSON format. This service can be used as a part of the VRP solver.

#### IV. RESULTS AND DISCUSSION

Slow accumulation of data points per customer proved to be the main difficulty for every aspect of development, result collection and analysis included. For this purpose, several model files generated by a previous execution of the model-building module were saved. At a later time, the database was queried for the data required for predictions. Therefore, the data contains records that were not present at the time of model-building execution. This section will briefly discuss the results of applying the saved models to (at the time of writing) up to date data for two particular customers.

Figures 2 and 3 show scatter plots depicting the relationship of each input feature (number of articles, total weight and total volume) to the dependent variable (time of unloading) for each customer separately. In the case of Customer 1, a clear and fairly strong linear relationship can be recognized. The algorithm selected linear regression as the best-performing model. The absence of a linear relationship regarding Customer 2 is

also reflected in the algorithm's model choice - in this case, random forest regression was selected. In both instances, the models yielded satisfactory results: root mean squared error of 6.12 and 4.05 minutes for customers 1 and 2 respectively. These values are not significantly higher than the respective errors for data sets the models were built on (as a matter of fact, the mean error for customer 2 decreased with the addition of new records), indicating that the models were not overfit.

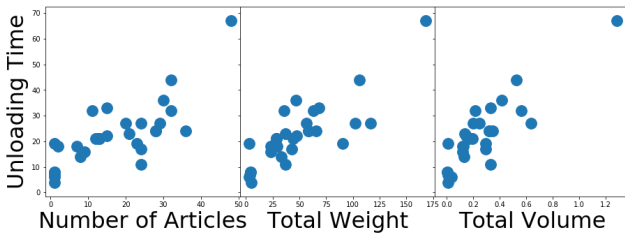


Figure 2: Scatter plots for customer 1

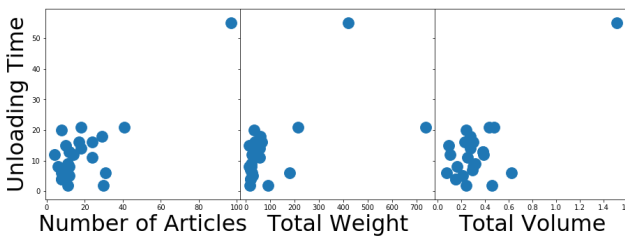


Figure 3: Scatter plots for customer 2

Table I summarises the results of a larger-scale test. It was conducted in a similar manner to two previously discussed customer-specific tests. Models built during development for the 33 customers with the most records were saved and applied to up to date data. A prediction is deemed accurate if it differed from the correct value by either no more than 3 minutes or no more than 20%. Percentage of improved results refers to the percentage of instances in which the algorithm yielded a better result than the previously used prediction formula described in [20]. That formula is obtained on the experimental way and depends of several parameters for each customer: (i) Number of articles, (ii) Total ordered volume, (iii) Total ordered weight, and (iv) Predefined unloading time constant. The correction factor is added after the calculation of the unloading time, and it is determined based on the previously available historical data for the appropriate customer. In the cases where the algorithm was outperformed by formula, the differences in predictive power were significantly smaller than those in cases where the algorithm performed better. It is expected that the accuracy will increase with time, so it can be stated that this approach gave significantly better results compared to the previous one.

It is worth noting that the process of estimating unloading time from GPS data itself uses the predictions: when a vehicle makes a single stop in close proximity to multiple customers and serves them, the total time expended is divided among

TABLE I: RESULTS

Property	Value
Number of models built	33
Min. number of data points	19
Max. number of data points	38
Total number of data points	807
<b>Percentage of improved results</b>	<b>83.27%</b>
Percentage of accurate predictions	63.57%

these customers in proportion to their predicted unloading times. There is thus reason to believe that further improvement of predictions will lead to more accurate estimates in the database.

## V. CONCLUSION AND FUTURE WORK

The paper describes an innovative way to improve the vehicle routing process. Improvement is achieved by analyzing GPS data and earlier deliveries. The obtained data were used with modern machine learning techniques, which improved the estimation of unloading time at customers. Based on the customer data, weight, volume and number of items in the order, an estimate of unloading time is created which is used as input data to resolve the VRP.

The described technique is used as part of the transport management system for some of the largest distribution companies in Bosnia and Herzegovina, and improvements over the standard statistical estimate are noticeable. Therefore, research in this area can significantly improve the vehicle fleet routing process.

In the future, it is planned to implement additional parameters to create the model, such as dispatcher and driver feedback for the estimated time, and a number of others improvements.

## REFERENCES

- [1] Dantzig, G. B., and Ramser, J. H. (1959). The truck dispatching problem. *Management science*, Vol. 6(1), 80-91. doi: 10.1287/mnsc.6.1.80
- [2] Žunić, E., Đonko, D., and Buza, E. (2020). An adaptive data-driven approach to solve real-world vehicle routing problems in logistics. *Complexity*. Hindawi. doi: 10.1155/2020/7386701
- [3] Žunić, E., and Đonko, D. (2019, September). Parameter setting problem in the case of practical vehicle routing problems with realistic constraints. In *2019 Federated Conference on Computer Science and Information Systems (FedCSIS)* (pp. 755-759). IEEE. doi: 10.15439/2019F194
- [4] Laporte, G. (2009). Fifty years of vehicle routing. *Transportation science*, 43(4), 408-416. doi: 10.1287/trsc.1090.0301
- [5] Gendreau, M., Laporte, G., and Potvin, J. Y. (2002). Metaheuristics for the capacitated VRP. In *The vehicle routing problem* (pp. 129-154). Society for Industrial and Applied Mathematics. doi: 10.1137/1.9780898718515.ch6
- [6] Braekers, K., Ramaekers, K., and Van Nieuwenhuysse, I. (2016). The vehicle routing problem: State of the art classification and review. *Computers Industrial Engineering*, 99, 300-313. doi: 10.1016/j.cie.2015.12.007
- [7] Osaba, E., Yang, X. S., and Del Ser, J. (2020). Is the Vehicle Routing Problem Dead? An Overview Through Bioinspired Perspective and a Prospect of Opportunities. In *Nature-Inspired Computation in Navigation and Routing Problems* (pp. 57-84). Springer, Singapore. doi: 10.1007/978-981-15-1842-3\_3
- [8] Chiang, W. C., and Russell, R. A. (1996). Simulated annealing metaheuristics for the vehicle routing problem with time windows. *Annals of Operations Research*, 63(1), 3-27. doi: 10.1007/BF02601637
- [9] Gendreau, M., Hertz, A., and Laporte, G. (1994). A tabu search heuristic for the vehicle routing problem. *Management science*, 40(10), 1276-1290. doi: 10.1287/mnsc.40.10.1276

- [10] Baker, B. M., and Ayechev, M. A. (2003). A genetic algorithm for the vehicle routing problem. *Computers Operations Research*, 30(5), 787-800. doi: 10.1109/ICECTECH.2010.5479956
- [11] Osaba, E., Yang, X. S., Fister Jr, I., Del Ser, J., Lopez-Garcia, P., and Vazquez-Pardavila, A. J. (2019). A discrete and improved bat algorithm for solving a medical goods distribution problem with pharmacological waste collection. *Swarm and evolutionary computation*, 44, 273-286. doi: 10.1016/j.swevo.2018.04.001
- [12] Osaba, E., Yang, X. S., Diaz, F., Onieva, E., Masegosa, A. D., and Perallos, A. (2017). A discrete firefly algorithm to solve a rich vehicle routing problem modelling a newspaper distribution system with recycling policy. *Soft Computing*, 21(18), 5295-5308. doi: 10.1007/s00500-016-2114-1
- [13] Žunić, E., Delalić, S., Hodžić, K., Beširević, A., and Hindija, H. (2018, November). Smart warehouse management system concept with implementation. In *2018 14th Symposium on Neural Networks and Applications (NEUREL)* (pp. 1-5). IEEE. doi: 10.1109/NEUREL.2018.8587004
- [14] Delalić, S., Chahin, M., and Alihodžić, A. (2019, October). Optimal City Selection and Concert Tour Planning Based on Heuristic Optimization Methods and the Use of Social Media Analytics. In *2019 XXVII International Conference on Information, Communication and Automation Technologies (ICAT)* (pp. 1-6). IEEE. doi: 10.1109/ICAT47117.2019.8939040
- [15] Delalić, S., Alihodžić, A., and Selmanović, E. (2019, October). Innovative Usage of Online Platforms Analytics on Event Planning based on the Genetic Algorithm with Greedy Approach. In *2019 XXVII International Conference on Information, Communication and Automation Technologies (ICAT)* (pp. 1-6). IEEE. doi: 10.1109/ICAT47117.2019.8938919
- [16] Zhou, X., and Chen, L. (2014). Event detection over twitter social media streams. *The VLDB journal*, 23(3), 381-400. doi: 10.1007/s00778-013-0320-3
- [17] Vaughan, A. (2020). Tracking down coronavirus. doi: 10.1016/S0262-4079(20)30834-4
- [18] Žunić, E., Hindija, H., Beširević, A., Hodžić, K., and Delalić, S. (2018, November). Improving Performance of Vehicle Routing Algorithms using GPS Data. In *2018 14th Symposium on Neural Networks and Applications (NEUREL)* (pp. 1-4). IEEE. doi: 10.1109/NEUREL.2018.8586982
- [19] Žunić, E., Delalić, S., Hodžić, K., and Tucaković, Z. (2019, July). Innovative GPS Data Anomaly Detection Algorithm inspired by QRS Complex Detection Algorithms in ECG Signals. In *IEEE EUROCON 2019-18th International Conference on Smart Technologies* (pp. 1-6). IEEE. doi: 10.1109/EUROCON.2019.8861619
- [20] Žunić, E., Delalić, S., and Đonko, Dž. (2020). Adaptive multi-phase approach for solving the realistic vehicle routing problems in logistics with innovative comparison method for evaluation based on real GPS data. *Transportation Letters*. doi: 10.1080/19427867.2020.1824311