Ant Colony Optimization Algorithm Enhanced with an Genetic Algorithm for Capacitated Vehicle Routing Problem

Gabriela Marissa Mosquera Orellana
Tecnológico de Monterrey
campus Querétaro
a01666191@tec.mx

Felipe de Jesús Damián Rodríguez
Tecnológico de Monterrey
campus Querétaro
a01707246@tec.mx

Abstract – This work presents a hybrid metaheuristic approach for optimizing last-mile delivery routes in the city of Querétaro, Mexico. The problem is formulated as a Capacitated Vehicle Routing Problem (CVRP). The proposed solution integrates a sequential combination of Genetic Algorithm (GA) and Ant Colony Optimization (ACO), where the GA performs global exploration to identify the best route structures, and the ACO refines those solutions through local exploitation.

Experimental results using 18 delivery points and three vehicles demonstrate that the hybrid GA-ACO model achieves faster convergence and lower total travel cost compared to using either algorithm independently. The proposed methodology provides an effective and flexible framework for route optimization in urban logistics and last-mile distribution systems.

Keywords— *last-mile delivery, global exploration, route structures, urban logistic*

I. Introduction

The vehicle routing problem (VRP) concerns the transport of items between depots and customers by means of a fleet of vehicles without exceeding the capacity constraints of each vehicle at minimum cost. It was initially formulated as "The Truck Dispatch Problem" and is considered a more broader and generalized version of the Travelling Salesman Problem (TSP). Is considered a very complicated combinatorial optimization problems and is classified as NP-hard, where the number of feasible solutions for the problem increases exponentially with the number of customers increasing, since the customer combination is not restricted to the selection of vehicle routes (Bin et al., 2009). It becomes harder and harder to obtain an exact solution for it in a reasonable amount of time (Rizzoli et al., 2007).

Even the most advanced solution impose particular constraints on the problem instance, which are often violated when dealing with real-world vehicle routing problems (Rizzoli et al., 2007). Solving real-world route planning requires methods capable of handling both inherent complexity and uncertainty. Optimizing vehicle distribution scheme can yield substantial cost reductions in specific business domains where efficient transportation adds value to the product (Toth & Vigo, 2014). Playing a pivotal role in realizing economic benefits in more open markets. The VRP aims to satisfy specific constraints and achieve goals of cost minimization and efficiency maximization by optimizing vehicle distribution scheme.

The VRP is of central importance in distribution management and has therefore been worked on with optimization algorithms based on metaheuristics, such as simulated annealing (SA) (Chiang and Russell, 1996; Koulamas et al., 1994; Osman, 1993; Tavakkoli-Moghaddam et al., 2006), genetic algorithms (GAs) (Baker and Ayechew, 2003; Osman et al., 2005; Thangiah et al., 1994; Prins, 2004), tabu search (TS) (Gendreau et al., 1999; Semet and Taillard, 1993; Renaud et al., 1996; Brandao and Mercer, 1997; Osman, 1993) and ant colony optimization (ACO) (Doringo et al., 1996; Doerner et al., 2002; Reimann et al., 2002; Peng et al., 2005; Mazzeo and Loiseau, 2004; Bullnheimer et al., 1999; Doerner et al., 2004). Further research indicates that combining basic metaheuristic algorithms with exact algorithms or constructive heuristic methods can effectively alleviate the limitations caused by convergence to local optima, thereby improving overall algorithm performance (Chen et al., 2025).

Given these factors, this study proposes an hybrid between Genetic and Ant Colony Optimization algorithm based on the algoritm used for solving the VRPTW, hybrid Improved Genetic Ant Colony Optimimzation (IGA-ACO) proposed by Chen et al, 2025. They integrated a GA with Variable Neighbourhood Search (VNS) and an ACO algorithm, to accelerate convergence and optimizing rout plaining to meet the time window constraints and the vehicles capacities. They use the VNS to avoid local optima and premature convergence. Aditionally, they introduce a dual-population structure to enhance global search efficiency. Our proposed algorithm performs a broad exploration of the solution space by performing a GA for a limited number of generations. The solution discovered is used as a high-quality seed for the ACO algorithm.

II. Description of VRP

The VRP is represented by a set of identical vehicles denoted by V, and a directed graph $\mathcal{G} = (C, A)$, which consist of a set of clients, C. The nodes 0 and $n+1$ represent the depot. The set of n vertices denoting customers is denoted by N. The arc set A denotes all possible connections between the nodes, including the origin (depot nodes), where all routes start at 0 and end at $n+1$. There is a cost C_{ij} associate with each arc $(i, j) \in A$ of the routing network. Each vehicle has a capacity limit q, each customer i has a demand d_i, where $i \in C$, and the demand d_i of each customer i is not greater than the maximum load capacity Q of the vehicle k. If we were to make it a time window problem, then we must have added each customer their respective opening and closing time. Vehicles must also leave the depot within the depot time window and must return before or at time b_{n+1}.

For each arc (i, j) where $i \neq j$, $i \neq n+1$, $j \neq 0$, and each vehicle k. The decision variable x_{ijk} is equal to 1 if vehicle k drives from vertex i to vertex j, and 0 otherwise. The decision variable y_{ik} is equal to 1 if i's task is performed by vehicle k, and 0 otherwise. The objective of the VRP is to service all the C customers using the V vehicles such that the following objectives are met and the following constraints are satisfied:

Objective

Minimize the total delivery path while ensuring that the service demand at all customers points is met while achieving the maximum benefit at the minimum cost.

Constraints

- The total demand at the customer point for each route is less than or equal to the vehicle's maximum capacity Q
- Each customer is serviced exactly once for distribution.
- Each vehicle route starts at vertex 0 and ends at vertex $n+1$ after completing their tasks
- All delivery vehicles have the same speed and load

The mathematical model is developed with an objective function:

$$\text{Min} \quad F = \sum_{i=1}^{n}\sum_{j=i}^{n}\sum_{k=1}^{v} x_{ijk} \cdot d_{ij}, \quad \forall i \neq j \tag{1}$$

Where the function corresponds to the distance cost. The corresponding constraints are as follows:

$$\sum_{i=1}^{n} y_{ik} \cdot d_i \leq K, \quad \forall v \in \{1, 2, \ldots, m\} \tag{2}$$

$$\sum_{i=1}^{n} x_{0ik} = \sum_{j=1}^{n} x_{j0k} = 1, \quad \forall v \in \{1, 2, \ldots, m\} \tag{3}$$

$$\sum_{k=1}^{v} y_{ik} = 1, \quad \forall i \in \{1, 2, \ldots, n\} \tag{4}$$

$$\tag{5}$$

III. Enhanced algorithm

A. Genetic Algorithm

Diversity is a key factor in the effectiveness of population-based algorithms. However, randomly generated initial populations can result in insufficient diversification. Therefore, customers were clustered as groups, taking into consideration the constraint of vehicle capacity. To get initial feasible solutions the clients are group by K-means clustering algorithm in the initial population generation phase. The customers are divided into feasible clusters, then feasible routes are constructed for each cluster. The solutions given are taken as the initial population. K-means algorithm partitions the set of objects into k clusters.

The routing process begins with the inclusion of the depot in the clusters. The customer who is nearest to the depot in one cluster is assigned to the route. A new route is formed, when there is a lack of a feasible location. The evolution of the algorithm gravitates towards the computation of the fitness function, that penalize those solutions that violate the constraints. The selection process utilizes two approaches the roulette-wheel method and elitism. Elitism guarantees the survival of the fittest individuals. In each generation, we identify the best individuals in the population, which are copied without modification, directly into the next population. This ensures that the best-found solution is monotonically non-degrading. The remaining individuals, where N is the total population size, are generated through selection and reproduction.

The Roulette-Wheel mechanism ensures that fitter individuals have a higher chance of reproduction, while still allowing less fit individuals a non-zero probability, therefore maintaining genetic diversity. Because chromosomes are vehicle subroutes, the parental solutions undergo a crossver by removing the exact customers that it's about to add, it ensures the final offspring still contains every customer exactly once, just in a new configuration.

B. Ant Colony Algorithm

The second stage of the proposed hybrid model employs an Ant Colony Optimization (ACO) algorithm to perform local exploitation and refinement of the solutions previously explored by the Genetic Algorithm (GA). The ACO stage focuses on minimizing the total distribution cost.

Each ant represents a complete feasible solution consisting of one or more vehicle routes separated by a depot marker. The heuristic information is inversely proportional to the distance, $\eta_{ij} = 1/d_{ij}$, while the pheromone matrix is initialized using a `defaultdict` structure that assigns a value of 1.0 to undefined edges. Distances are stored bidirectionally, ensuring that $d_{ij} = d_{ji}$.

Twenty ants are used to construct solutions over 50 iterations. When the number of vehicles is not explicitly specified, the algorithm dynamically allocates vehicles according to remaining customer demands. If a vehicle exceeds its capacity, the feasibility function forces a return to the depot.

The probability that ant k located at node i moves to node j is defined by:

$$p_{ij}^{k} = \frac{[\tau_{ij}]^{\alpha}[\eta_{ij}]^{\beta}}{\sum\limits_{l \in \text{allowed}_k} [\tau_{il}]^{\alpha}[\eta_{il}]^{\beta}}, \tag{6}$$

where α and β control the relative influence of the pheromone trail and the heuristic distance, respectively. The exploration–exploitation balance is governed by parameter q_0 if a random value $q < q_0$, the next node is chosen greedily according to $\max(\tau_{ij}\eta_{ij}^{\beta})$; otherwise, the node is selected probabilistically following Eq.(6).

After all ants have completed their routes, the pheromone matrix is updated using the evaporation rule:

$$\tau_{ij} \leftarrow (1-\rho)\tau_{ij} + \sum_{k=1}^{m} \Delta\tau_{ij}^{k}, \tag{7}$$

where ρ is the pheromone evaporation rate, and $\Delta\tau_{ij}^k$ represents the contribution of ant k, defined as:

$$\Delta\tau_{ij}^k = \begin{cases} \dfrac{Q}{f_k}, & \text{if ant } k \text{ used edge } (i,j), \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

Here, f_k denotes the total distance of the route constructed by ant k, and Q is a constant related to the intensity of pheromone deposition.

IV. Results

While making the code, we encountered a fundamental challenge in integrating the ACO algorithm with the GA in a coupled manner. Instead, we put into comparison the ACO, GA, and a GA with K-means, to enrich our understanding of the scope of the different algorithms and what can we do to enhanced their potential.

This section describes computational experiments carried out to investigate the performance of the ACO, the GA and the proposed GA with K-means. The algorithm was coded in Python. Our experimental results used the coordinates provided by a drug distributor that delivers to hotels and pharmacies. The demand for each client was chosen randomly between values of 5 - 20 kg. The results presented below are based on the following parameters:

ACO Parameters	
$\alpha = 1$	(pheromone importance)
$\beta = 3$	(distance importance)
$\rho = 0.3$	(pheromone evaporation rate)
$Q = 1/d$	(pheromone reinforcement proportional to inverse distance)
$\tau_0 = 1$	(initial pheromone)
$m = 20$	(number of ants)
$q_0 = 0.4$	(balance between exploration and exploitation)
Iterations	50

Genetic Parameters	
Generations	150
Population size	90
Number of vehicles	3
Vehicle capacity (Q)	150 kg
Average demand per client	20 kg
Mutation rate	5%
Crossover	Single-point (route-based)
Selection	Tournament method
Seed	42 (reproducibility)

GA + K-Means (Hybrid Approach)

- Same parameters as the GA above
- Initial clustering of delivery points using K-Means ($k = 3$)
- Each cluster optimized independently before global integration

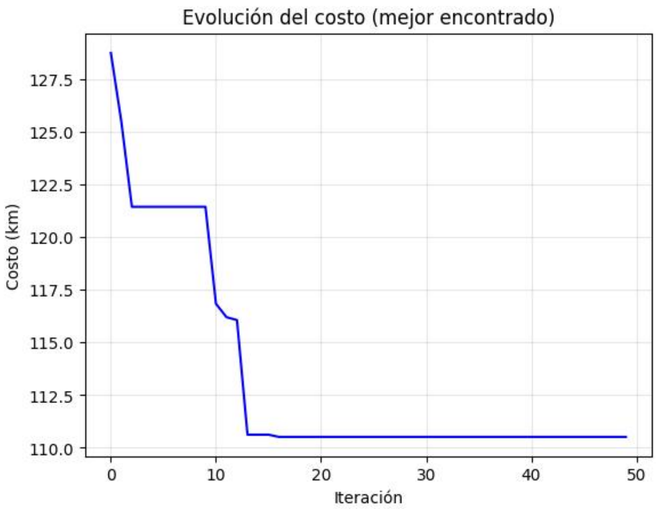

Figure I: Cost Evolution with ACO

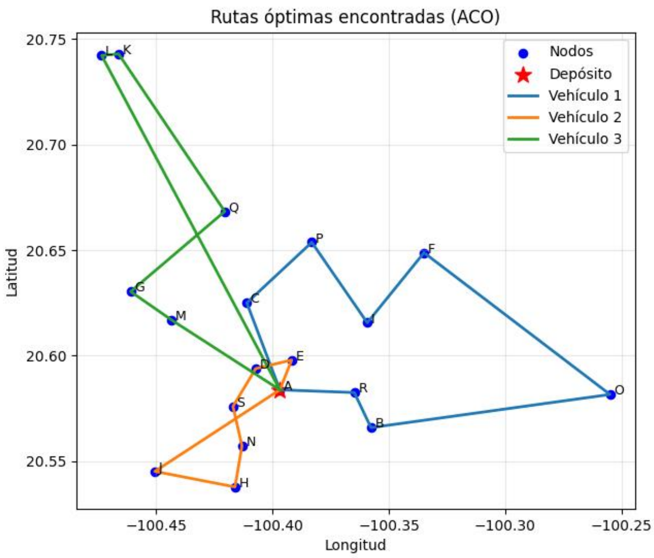

Figure II: Routes distribution for ACO

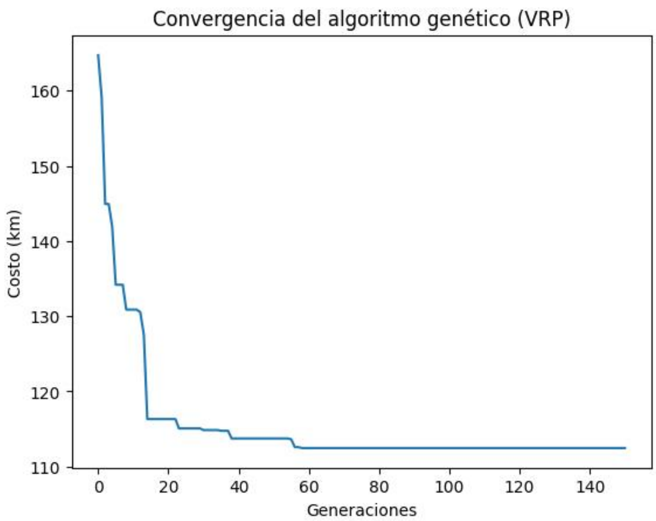

Figure III: Cost Evolution with Genetic

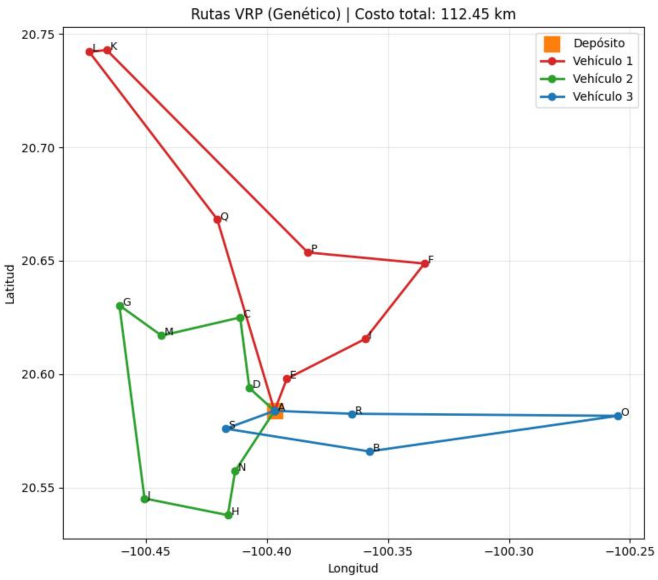

Figure IV: Routes distribution with Genetic

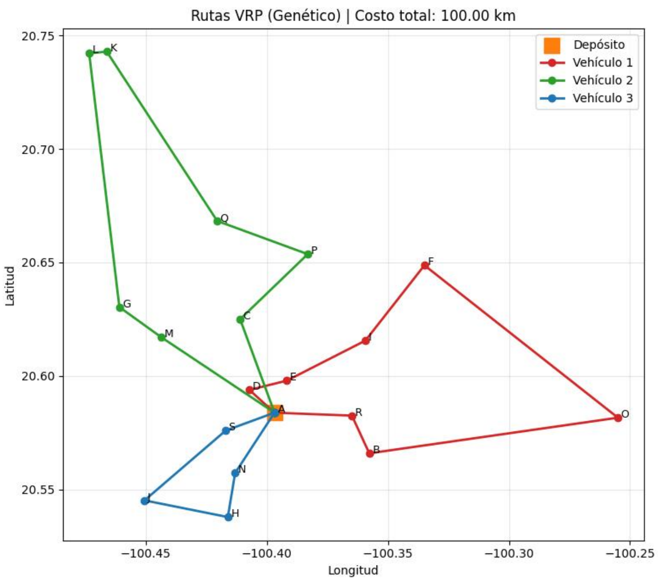

Figure V: Routes distribution with Genetic (K-means)

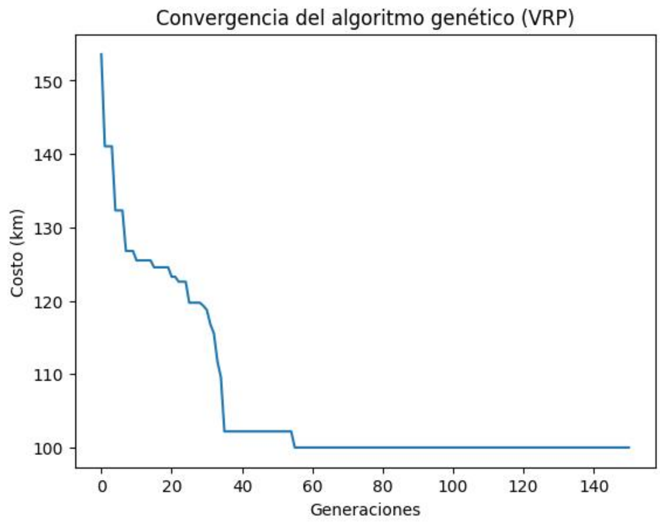

Figure VI: Cost Evolution with Genetic (K-means)

V. Discussion

The Ant Colony Optimization (ACO) algorithm achieved a final cost of around 110.9 km, showing a good performance, yet a moderate convergence stability. The pheromone mechanism helped the algorithm find feasible routes quickly, but its exploitation phase became dominant too early, reducing its exploration capacity and getting trapped in a local optima. This behaviour is typical of ACO when applied to problems with multiple vehicles, where balancing local reinforcement and global search becomes more complex. As a result, the solution was satisfactory but lacking optimality.

The basic Genetic Algorithm (GA) obtained a slightly higher cost of 112.45 km, but its convergence was smoother and more stable than ACO. Despite starting from random populations, GA efficiently evolved towards near-optimal routes using selection, crossover, and mutation. It distributed the delivery points evenly across vehicles, which made it more suitable for multi-route optimization. However, without a prior structure, the algorithm required more generations to converge.

When K-Means clustering was integrated into the initial population of the Genetic Algorithm, the total distance improved significantly to 100 km, marking the best performance among the three. Using clustering algorithms in the population generation step of GA have a positive effect on the results (Gocken & Yaktubay, 2019). That's why pre-clustering the clients helped the algorithm start with geographically coherent groups, accelerating convergence and reducing the search space. The hybrid approach combined the strengths of spatial clustering and evolutionary optimization, providing fast resolution, better total distance and effective competition with a decent amount of vehicles. Overall, K-Means clustering algorithm in generating the initial population proved to be the most efficient and scalable option for last-mile delivery optimization.

VI. Conclusions

The CVRP is a problem of drawing the routes from a depot to customers who need to be serviced. The minimization of the total distance and the capacity constraints are determined as the objective functions. In this study we compared the performance of the ACO, GA and a proposed GA with K-means in approaching the CVRP. The objectives might have conflicted each other, because trying to reduce one causes the increase of the other. The straightforward GA has that has been described here performs well. Incorporating simple types of clustering into the initial population produces a significant improvement. For other projects we might try to solve a number of variants of the basic vehicle routing problem, to find the most successful metaheuristcs for the VRP and its application to real-world problems.

VII. Reflexions

A. Felipe de Jesús Damián Rodríguez

Durante el desarrollo del proyecto, mi compañera y yo logramos mantener una dinámica de trabajo equilibrada y colaborativa. Desde el inicio, distribuimos las tareas de forma equitativa, lo que permitió que cada uno se enfocara en investigar, analizar los datos y proponer el enfoque más adecuada para resolver el problema. Tras discusiones y revisiones, llegamos al acuerdo de implementar un algoritmo genético y un algoritmo ACO, con el objetivo de integrar ambos y construir una versión híbrida que aprovechara las ventajas de cada método.

El proceso de hacer acuerdos y de interacciones fue constante. Cada avance nos lo compartiamos y comentabamos entre nosotros, revisando el código, señalando posibles errores o áreas de mejora, y aportando ideas para optimizar el rendimiento del modelo. La comunicación fue fluida, lo que facilitó el trabajo colaborativo.

En cuanto a la gestión de conflictos y la toma de decisiones, aplicamos el dialogo para resolver diferencias en criterios técnicos. Siempre todo con respeto y con la disposición a eschuchar las opiniones del otro. Finalmente, el trabajo conjunto reflejó nuestro compromiso con la generación de nuestra propuesta: logramos construir un proyecto funcional integrando herramientas de optimización y visualización como Shiny para mostrar los resultados. Si bien los coordenadas utilizadas no fueron el aspecto central, se contextualizó el caso como una distribuidora de medicinas en Querétaro que abastece farmacias y hospitales, lo cual le da un sentido práctico a nuestra propuesta.

B. Gabriela Marissa Mosquera Orellana

Siempre he trabajado con Felipe, entonces entiendo sus métodos de trabajo. Por lo que me siento muy cómoda trabajando con él. Habíamos quedado tras leer la literatura que íbamos a hacer un híbrido, entonces el comenzó haciendo el ACO, y yo por otro lado seguí leyendo para hacer del GA uno eficiente, además de que quería experimentar; pues, me llamó bastante la representación de los algoritmos bio-inpirados y el potencial de ellos. Siento que los resultados hubieran sido mejores, pero apunte muy alto y me caí, pero sé que es cuestión de tiempo.

Dentro de los conflictos que llegamos a tener durante el desarrollo del proyecto, fue la fusión de los códigos que hicimos. Por cuestión de tiempo, y demás factores, no logramos completar la idea que habíamos tenido sobre el híbrido que buscamos crear.

Como llevamos tiempo trabajando juntos, no existe como tal una gestión de conflictos. Pues, si uno se equivoca el otro entiende y lo intentamos arreglar entre los dos. Asimismo, en diferentes aspectos. En nuestras entregas se ve el compromiso que tenemos no solamente hacía las diferentes clases, sino a lo que podemos aprender a través de los entregables. Es el activo más importante, pues nos obliga a investigar, a comunicar lo que entendemos y a aplicarlo a problemas de la vida real.

Data Availability

The code supporting the findings of this study are available in the next link

References

[1] P. Toth & D. Vigo, *Vehicle Routing: Problems, Methods, and Applications.* Philadelphia, PA, USA: SIAM, 2014

[2] Gocken, T., & Yaktubay, M. (2019). *Comparison of different clustering algorithms via genetic algorithm for VRPTW.* International Journal of Simulation Modelling, 18(4), 574–585. https://doi.org/10.2507/ijsimm18(4)485

[3] Maroof, A., Ayvaz, B., & Naeem, K. (2024). *Logistics optimization using hybrid genetic algorithm (HGA): A solution to the vehicle routing problem with time windows (VRPTW).* IEEE Access: Practical Innovations, Open Solutions, 12, 36974–36989. https://doi.org/10.1109/access.2024.3373699

[4] Rizzoli, A. E., Montemanni, R., Lucibello, E., & Gambardella, L. M. (2007). *Ant colony optimization for real-world vehicle routing problems: From theory to applications.* Swarm Intelligence, 1(2), 135–151. https://doi.org/10.1007/s11721-007-0005-x

[5] Chen, G., Gao, J., & Chen, D. (2025). *Research on Vehicle Routing Problem with time windows based on improved genetic algorithm and ant colony algorithm.* Electronics, 14(4), 647. https://doi.org/10.3390/electronics14040647

[6] Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). *A fast and elitist multiobjective genetic algorithm: NSGA-II.* IEEE Transactions on Evolutionary Computation: A Publication of the IEEE Neural Networks Council, 6(2), 182–197. https://doi.org/10.1109/4235.996017

[7] Ombuki, B., Ross, B. J., & Hanshar, F. (2006). *Multi-objective genetic algorithms for vehicle routing problem with time windows.* Applied Intelligence, 24(1), 17–30. https://doi.org/10.1007/s10489-006-6926-z

[8] Gocken, Tolunay, Yaktubay, M., & Kilic, F. (2017). *Improvement of a genetic algorithm approach for the solution of vehicle routing problem with time windows.* 2017 International Artificial Intelligence and Data Processing Symposium (IDAP).