Revista Internacional de Investigación e Innovación Tecnológica

Página principal: www.riiit.com.mx

Reducing the complexity of implementing the vehicle routing problem using evolutionary strategies and fuzzy logic

Reducción de la complejidad de implementar el problema de ruteo de vehículos usando estrategias evolutivas y lógica difusa

Montes-Dorantes, P.N.[a*], Ireta-Sanchez, P.H.[b], Méndez, G.M.[c]

[a] Universidad Autónoma del Noreste. División de Estudios de posgrado e investigación. Blvd. José Musa de León y General Medardo de la Peña S/N Col. Los Pinos, 25100, Saltillo, Coahuila, México.
[b] Tecnológico Nacional de México/ Instituto Tecnológico de Saltillo. Blvd. V. Carranza No. 2400, Col. Tecnológico, CP. 25280 Saltillo, Coah. México.
[c] Tecnológico Nacional de México/ Instituto Tecnológico de Nuevo León, Av. Eloy Cavazos 2001, Guadalupe Nuevo León, México, Posgrado en Ingeniería Mecatrónica.
pascualresearch@gmail.com*; pireta@itsaltillo.edu.mx; gmm_paper@yahoo.com.mx

Technological innovation: Optimization of distribution routes in logistics.

Industrial application area: Distribution and logistics.

Received: february 26th, 2022
Accepted: november 13th, 2022

Resumen

El problema de ruteo de vehículos (VRP) clásico, también llamado problema del agente viajero (TSP) es uno de los casos más estudiados de problemas polinomiales no determinísticos (NP) -completos. Para resolver un problema NP-completo en un escenario del mundo real (problema práctico), es necesario generar una solución inicial seguida de una optimización para reducir el costo en el consumo de recursos. En este trabajo se propone un algoritmo híbrido que combina una estrategia evolutiva (ES) para la generación de rutas que representan la entrada para un sistema difuso (FLS) que posteriormente es optimizado por la misma ES una vez resuelto por el sistema difuso. Esta técnica de optimización heurística resuelve el VRP encontrando la ruta que minimiza los costos como objetivo. La unión de las ES con la lógica difusa reduce la complejidad de un sistema difuso puro ya que sólo necesita la distancia en las conexiones entre los vértices para obtener la solución de una función que define el sistema, en comparación con la combinación factorial utilizada en algunas de las metaheurísticas. Esa propuesta pone a prueba el cómputo suave

(SC) y su capacidad para resolver problemas combinatorios con mejor precisión que un enfoque metaheurístico como el recocido simulado (SA) y el problema del camino más corto (SP) o modelo de Dijkstra. Los resultados muestran una precisión mayor al 80% que es habitual en las redes neuronales artificiales (ANN), adicionalmente este enfoque dedica sólo el 10% del tiempo computacional requerido por el SA y el SP para obtener una solución factible.

Palabras clave: Cómputo evolutivo, estrategias evolutivas, lógica difusa tipo 1, modelo híbrido, problema de ruteo de vehículos.

Abstract

The classical vehicle routing problem (VRP) also called travelling salesman problem (TSP) is one of the most studied cases of nondeterministic polynomial (NP)-complete problems. To solve the NP-complete problem in a real-world scenario (practical problem), it is necessary to generate an initial solution followed by an optimization to reduce the cost in resource consumption. In this paper we propose a hybrid algorithm that combines an evolutionary strategy (ES) for the generation of routes that represent the inputs to a fuzzy logic system (FLS) that is subsequently optimized by the same ES once solved by the fuzzy system. This heuristic optimization technique solves the VRP by finding the route that minimizes the costs as a goal. The union of ES with FLS reduces the complexity of a pure FLS as it only needs the distance in the connections between vertexes to obtain the solution of a system-defining function in comparison to the factorial combination used in some metaheuristics. This proposal tests soft computing (SC) and its capacity to solve combinatorial problems with better precision than a metaheuristic approach such as simulated annealing (SA) and shortest path problem (SP) or Dijkstra model. The results show an accuracy higher than 80% that is usual in the artificial neural networks (ANN); additionally, this approach spends only the 10% of computational time required by SA and SP to obtain a feasible solution.

Keywords: Evolutionary computing, evolutionary strategies, hybrid model, Type-1 singleton fuzzy logic system, vehicle routing problem.

1. Introduction

The VRP was systematically studied by Dantzig and Ramser in [1] when they studied the fuel distribution for gas stations. Since then, many variants of the VRP problem have been presented. Every variant added constraints to the studied model. In [2] the following classification is presented: Capacitated VRP (CVRP), VRP with time windows (VRPTW), Stochastic (SVRP), VRP with pick-up and deliveries (VRPPD), multiple depot VRP (MDVRP), time-dependent travel time VRP (TD-VRP), heterogeneous fleet (HFVRP), mixed fleet VRP (MFVRP), among others. The basics of modeling this class of problems is by the establishing Hamiltonian paths [3].

Studies presented in the literature shows that there are a few approaches to solve these problems with hybrid models [4-5]. Existing approaches provide a solution via meta-heuristics (SA, tabu search, genetic algorithms, ant colony and GRASP), exact methods and hybrid methods. These hybrid combinations are produced by metaheuristic and exact methods; however, soft computing approaches are not mentioned. On the other hand, in [2, 6] these methods are defined as differential evolution, including a soft computing branch of evolutionary computation that contains several metaheuristics (ant colony optimization, bee colony, partial swarm, genetic algorithm). Evolutionary computing also contains bio inspired algorithms (ant colony, particle swarm, bee algorithm, bacterial foraging optimization, memetic algorithms, cultural algorithms, harmonic search and artificial immune systems, among others). Currently, the use of bio inspired algorithms to solve combinatorial problems is a common tool that provides solutions close to the optimal value. For example, a hybrid Bat algorithm added by: Kassem et al in [7]. Although all the mentioned algorithms provide satisfactory results, accuracy and runtime represent a challenge for the mentioned methods.

Today, emergent economies produce around 50% of the goods distributed in a global economy. This is important because the composition of these economies is mainly based on small and medium-sized enterprises (SMES) that operate on small budgets and require the implementation of scientific and technological tools. SMES require the use of tools that allow them to provide quick and low-cost solutions without the complications of more complex mathematical models that lead to loss of profit. The SMES regularly operate with a single vehicle or a couple of vehicles to generate their distribution routes. Because of this, it is necessary to simplify the mathematical models and algorithms to reduce the cost of development and the requirements for specialized software and expensive hardware. A developed solution was proposed in [8]. This tool is widely used by SMES due to its easy use and maintenance. Recently, logistics has gained importance because the costs associated with the distribution of goods represent up to 40% of the total product cost [9].

On one hand, no instances were found for the classical VRP model, the existent instances are either capacitated or CVRP. On the other hand, different approaches show different instances sizes as corroborated by literature. Classical instances as used in [10] to evaluate several routing problems are not for classical VRP approaches. Table 1 shows that there is no uniformity or criteria to decide what is the adequate number of clients for an instance. Also, small routes (less than 30 clients) are missing and classic approach of VRP and reality show that with more than 50 clients as example, a large vehicle is required that is not suitable for the small enterprises. It is an opportunity to develop new approaches to solve problems with small sizes. The actual literature also shows small sizes as demonstrated by this proposal in [11]:

> “We generated a number of test instances based on real data from industry partners and it was shown that both the proposed heuristics and the MIP solver are able to find high quality solutions in reasonable time for small problem instances with less than 20 orders. However, for larger instances with 20 to 60 orders…”

In [12] small instances are used to test the proposal, that consists in a non-deterministic demand using fuzzy systems and subsequently optimized using a genetic algorithm. In [13] small instances are used to test the model using instances of size 30 to 80 obtained from [10]. Also, it is mentioned that “due the combinatorial nature of these problems, the use of exact resolution approaches is often limited by the size of the problem”. In [14] the instances are small from 25, 50 and 100.

Table 1. Size of instances for CVRP. Adapted from [10].

Author	Size of instance
Augerat, 1965	32-80
Augerat, 1995	31-78
Chistofides and Eilon, 1969	45, 72, 135
Fisher, 1994	45, 72, 135
Chistofides, Mingozzi and Toth, 1979	101-200
Augerat, 1995	16-101
Chistofides, Mingozzi and Toth, 1979	50-199
Rochat and Taillard, 1995	75-385
Golden et a. 1998	240-480
Li et al, 2005	560-1200
Uchoa et al, 2014	100-1000
Arnold, Gendreau and Sörensen, 2017	6000-30000
DIMACS, 2021	400-1000
Solomon, 1987	100
Homemberg and Gehring, 1999	200-1000

Literature review and related work

In [12] it is mentioned that the fuzzy logic is used to avoid a non-deterministic demand in a multi depot and multi facility and later optimized by a genetic algorithm in a CVRP. In [13] the fuzzy numbers are used to evaluate and optimize: travel time, service time, or customer demands. A scholar bus route is designed in [14] and the fuzzy model is used to optimize and determine the optimal number of stops in relation to the required walking distance for students. Also, the number of clients of the instances is small to medium from 25 to 100. In [15] a model that uses fuzzy logic to determinate the costs (energy, service time, travel time) in electric vehicles is proposed but it is a TWCVRP called electric vehicle routing problem with time windows (EVRPTW). Is remarkable that fuzzy logic is only used to evaluate the fuzziness of times, demands or services but not of the routes.

It is important to mention that fuzzy logic in particular type-1 fuzzy numbers, that are used in several VRP applications are not capable to handle uncertainty as is mentioned by Mendel in [16]. These values can be treated as fuzzy numbers, but the uncertainty in this class of problems cannot be treated.

Ant colony optimization is another model used to optimize the VRP models. In [17] fuzzy logic is used to evaluate the evaporation of the pheromone parameters. The research in [18] presents a case for route cost optimization in CVRP and other variants in the same work. In [19] a VRP with fuzzy demands at a minimum cost is presented using trapezoidal functions, travel times, stochastic demands and processing times, while in [20] they are processed by fuzzy logic, but the trapezoidal functions maintain a vertex of the same value in most of their forms. It is remarkable that all previous works using fuzzy numbers as triangular membership functions, while this proposal presents a novelty with the use of Gaussian membership functions.

Cost, speed and distance are hierarchized in [21] with fuzzy. In [22] the VRP problem is oriented to the postman problem, it considers that the distance between nodes is not crisp and the travel times are uncertain. The food elaboration and delivery is treated as a VRP using fuzzy to avoid the uncertainty in production [23]. The milk delivery VRP is used to design a distribution network using fuzzy to maximize the satisfaction and reduce the time window of delivery [24].

In related work, the use of the SC to evaluate non-deterministic demands, uses fuzzy triangular membership functions to collect the quantities demanded by the clients, however there are uncertainties on the loads to generate the final solution [25]. See (Fig. 1).

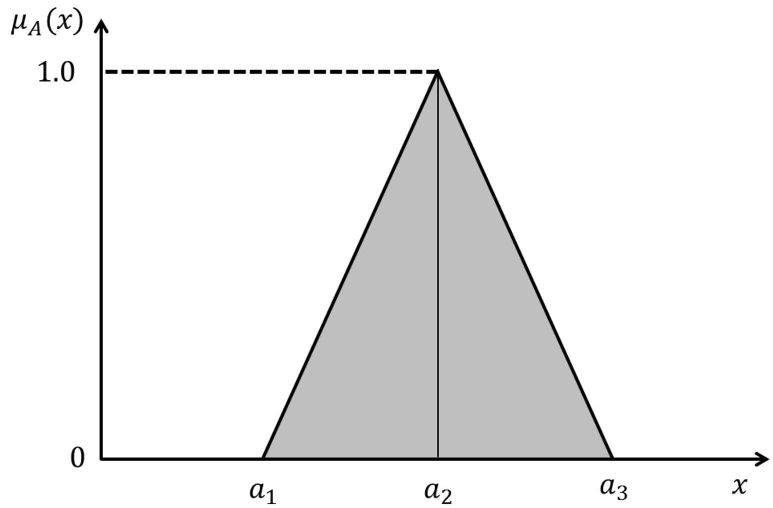


Figure 1. Triangular Fuzzy set. Adapted from: [25].

In a multi objective model, the fuzzy logic is used to generate a prediction of the demand in order to determine the class of vehicle used, and the defined route, both based on the amount of goods required by the client and by his fuzzy demands. In this sense [26] presents a CVRP-based model to produce a priori optimal route with stochastic demands. In this work the FLS is used to generate a response time evaluation for each client or node. In [26] the time window serves as a complementary parameter to evaluate the system [27]. The multi-attribute case is composed by several techniques: heuristics, metaheuristics and soft computing techniques such as GA and ANN. The GA method is used to generate the necessary knowledge, but they have memory loss, it is to say that only the latest results are available in [28] and their computational process could be more than a SC model as shown in [3, 8]. Green vehicle routing uses fuzzy distances to try to reduce gas emissions by route optimization [29]. Intelligent approaches are used to add a safety factor to the model as the VRP is invaded by uncertainty. This factor helps to consider the traffic risks, accidents, deadlines, delays, etc. But, these approaches are not used to create the routes. In addition, the fuzzy models are used to generate a numerical scale to evaluate the factors mentioned above that could be complex to establish as crisps values.

The real demands and routes necessary to satisfy the problems of the industry are committed by smaller route sizes, i.e. less than 50. In [11] the routes of a real industrial problem as in this work are used, for less than 20 orders and from 20 to 60 orders or clients. In [13] again small instances (30-80) are used and in [14] the instances are small of 25, 50 and 100.

2. Equipment, materials and methods

2.1 Equipment

The application of the proposal was carried out in MatLab R2009a using a laptop with intel® CORE™ i5 processor @ 2.5 GHz, 4

MB RAM and windows® 7 Professional as the operating system.

2.2 Materials

To test the materials, a database, extracted from the dissertation in [30] is used as the distance matrix to test the proposal. This database is asymmetric and the information was adapted using the upper part to produce a symmetric matrix with no capacity limit. The database provides the distance matrix necessary to generate the routes and obtain the costs to be optimized.

2.3 Methods

To solve this problem, it is necessary to find the route that implies the lowest cost. One of the proposed approaches is use of the graph theory, which can be implemented in the form of polygons where the routes can be calculated as a result of the union of two points or vertexes (clients). Examples of this are the Hamilton´s puzzle, Hamiltonian cycle or Hamiltonian path. Hamiltonian path is restricted to visiting every vertex only once. This path is equivalent a route as shown in (Fig. 2). Although the polygon has more connections (six) than vertexes (four). The number of combinations is four, and the number of permutations is 1296 [3], which is given by (Eq. 1),

$$c^{v} \qquad \textbf{(Eq. 1)}$$

v= {1, 2… n}

Where: c is the number of connections in the graph and v are the set of vertices of the Hamiltonian path.

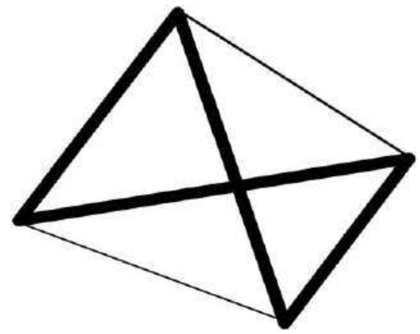


Figure 2. Hamiltonian path. Adapted from: [3].

From the total number of permutations, several of them are eliminated because they are not viable. These permutations or states do not meet the constraints of the model that yields (Eq. 2),

$$\min \sum_{\{i,j\}\in n} c_{ij} y_{ij} \qquad \textbf{(Eq. 2)}$$

Where: C_{ij}= transport cost of the connection between two nodes; i is the client or start node for $i \in \{1, 2, \ldots, n\}$, j is the end node for $j = n$, n is the total number of clients, and y is a binary variable that assigns the path given by (Eq. 3),

$$y_{ij} \begin{cases} 1 \; if\ the\ path\ from\ node\ i\ to\ node\ j\ it's\ taken \\ 0 \; otherwise \end{cases}$$

(Eq. 3)

Subject to:

$$\sum_{k\in K} x_{ij}^{k} = y_{ij};\ \forall \{i,j\} \in n \quad \textbf{(Eq. 4)}$$

Where: x defines the assignation of a vehicle to the route between node i and j (Eq. 5), and k is the number of vehicles used in a fleet to distribute the goods for, *k ε{1, 2, …, k}.*

$$X_{ij}^{k} \begin{cases} 1 \; if\ the\ vehicle\ k\ is\ asigned\ to\ traverse\ the\ arch\ from\ node\ i\ to\ node\ j \\ 0 \; otherwise \end{cases}$$

(Eq. 5)

The displacement between nodes is given by (Eq. 6, 7), these equations are formulated to ensure transit between the rows and columns of the cost matrix (represented economically or distance).

$$\sum_{j\in n} y_{ij} = 1;\ \forall \{i\} \in n \qquad \textbf{(Eq. 6)}$$

$$\sum_{i\in n} y_{ij} = 1;\ \forall \{j\} \in n \qquad \textbf{(Eq. 7)}$$

One of the main constraints of the VRP problem is that the route begins at the depot and ends at the same point, assured by (Eq. 8, 9).

$$\sum_{j\in n} y_{oj} = k \qquad \textbf{(Eq. 8)}$$

$$\sum_{i\in n} y_{io} = k \qquad \textbf{(Eq. 9)}$$

The *o* index means that all routes start at the depot and returns to the it once finished. The capacity of the vehicle must be taken into account and it is given by (Eq. 10),

$$\sum_{i\in n}\ \sum_{j\in n} d_i x_{ij}^k \leq u;\ \ \forall\{k\} \in K \ \ \textbf{(Eq. 10)}$$

Where: d_j is the demand of the j node; u is the capacity of the vehicle k

The optimized route is verified by (Eq. 11) that assures that the route is not defined by a complete sequence, e.g. *1, 2, .., n* that does not satisfy the constraint because the solution is exactly Q and not a subset of Q.

$$\sum_{i\in Q}\ \sum_{j\in Q} y_{ij} \leq |Q| - 1;\ \ \forall\subseteq Q\ \{1, 2, \ldots, n\} \qquad \textbf{(Eq. 11)}$$

The possible values for the variable y are given by Eq. (Eq. 12, 13), and defined in (Eq. 3).

$$y_{ij} \in \{0, 1\};\ \ \forall\ \{i, j\} \in n \qquad \textbf{(Eq. 12)}$$

$$x_{ij}^k \in \{0, 1\};\ \ \forall\ \{i, j\} \in n, \forall\ k\ \in K \ \textbf{(Eq. 13)}$$

From the constraints presented, some of the permutations do not start or end at the depot and do not fulfill the constraints of the VRP model (Eq. 8) and (Eq. 9). From these restrictions the total permutations that could meet the criteria presented in the case of a polygon of Fig.1 and Eq.1 are only six.

Due to the complexities of the previous mathematical model, their solutions require hard mathematical work in the modeling phase, multiple mathematical calculations to obtain an evaluation that meet the constraints as (Eq. 3-13). The use of unconventional techniques is convenient when large instances or real-world problems are being solved. The enterprises that need these models do not have the mathematical knowledge or the mathematical qualifications to create and solve the algorithms presented in the literature. As is the case here, a medium sized enterprise in emergent markets does not have the necessary knowledge, time and budgets to develop or to purchase these kinds of classical systems. The use of heuristics or metaheuristics represents a challenge for enterprises to obtain feasible solutions in terms of time and quality among other aspects. Good solutions in a lot of cases are systematically adjusted by trial and error until a specific solution that meets the enterprise policies is obtained, regardless of whether it is the best solution or not.

2.3.1 Evolutionary strategies

As in nature, ES are methods used to finding solutions to several combinatorial problems [31]. Similar to the genetic algorithms the evolutionary strategies are mainly based on the studies of Charles Darwin, who stated that the strongest individuals survive and some characteristics remain over time [32]. The work of Gregor Mendel [33] affirms that some characteristics pass or conserved through generations, few characteristics are obtained from the father and the rest of the mother. The ES is a black-box optimization algorithm. Basically, the concept of the ES consists in comparing an individual with another that contains the same parameters, but in different combinations or in order to obtain the best individual that fits with an objective function. It uses repetitive test to obtain the best combination (optimal or near to optimal) to solve the objective function.

2.3.2 Evolutionary strategy algorithm

There is not a standardized model for ES, but it can be adapted from the literature forms [3]. The basic evolutionary strategy algorithm in its several forms consists of the following steps:

A. Initialize a parent population of size μ (Eq. 14),

$$P_{\mu} = \{a_1, a_2, \dots, a_{\mu}\} \quad \textbf{(Eq. 14)}$$

Where: P is the population matrix and μ=1, 2…, n marks the number of parents. a_i= {x_1, x_2,..., x_{μ}}; x_{μ} represents a variable.

B. Generate λ offspring $\tilde{a}$ forming the offspring population, e.g. (Eq. 15), where each offspring of $\tilde{a}$ is generated by:

$$\tilde{P}_{\lambda} = \{\tilde{a}_1, \tilde{a}_2, \dots, \tilde{a}_{\lambda}\} \qquad \textbf{(Eq. 15)}$$

1. Select ρ parents from (Eq. 14). If ρ= μ take all parental individuals instead.
2. Recombine the ρ selected parents a to create a recombinant individual r, $\tilde{a}_i$=*1, 2…, n.*
From
$$\rho = \{\tilde{a}_1, \tilde{a}_2, \dots, \tilde{a}_i\}$$
to
$$\rho = \{\tilde{a}_1, \tilde{a}_6, \dots, \tilde{a}_i\}$$
Where: $\tilde{a}_1 = \{x_1, x_2, \dots, x_i\}$ *becomes* $r_1 = \{x_7,\ x_1,\ \dots,\ x_i\}$

3. Mutate the strategy parameter, set a_i to the recombinant r_1.
4. Mutate the objective parameter, set a_i of the recombination r_1, and then use the mutated strategy parameter set to control the statistical properties.

C. Select new parent population (using deterministic truncation selection) from either:

1. The offspring population $\tilde{P}_{\lambda}$ (this is referred to as *the comma - selection*, usually denoted as "(μ, -selection").
2. The offspring $\tilde{P}\lambda$ and parent $P\mu$ population (this referred to as *plus*-selection, usually denoted as "(μ+λ, -selection").

D. Go to B, until a stop criterion fulfilled.

2.3.3 1 + 1 Evolutionary strategy

The ES is assembled by a vector that contains characteristics of an individual that fulfill all variables of a function. The parent generates a couple of matrices by recombination of the original parent (vector *r*). The pair of matrices represent the parents and the children. Both matrices have the same size. A pair of individuals (one of every matrix) is evaluated by a Hamming distance [34] with a fitness function. Because the evolutionary strategy has no stopping criterion, it is necessary generate multiple options as a solution of a function. The most used stop criterion is the restriction of the number of generations. See Fig. 3.

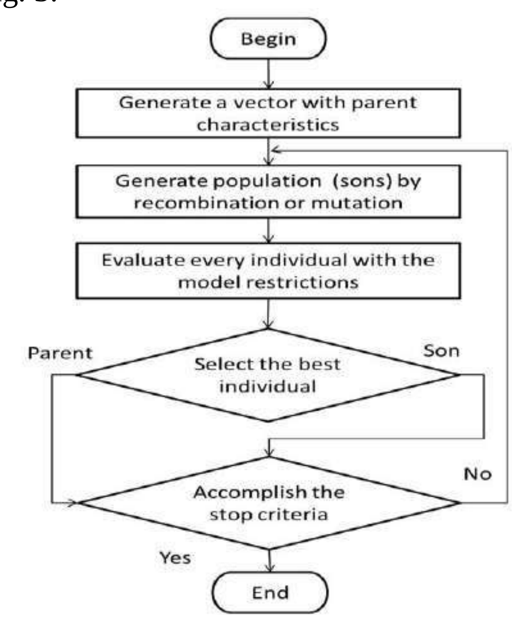


Figure 3. Block diagram of evolutionary strategy.

2.3.4 Fuzzy logic

The FLS's are proposed to improve and control some processes as can be seen in [35].

Fuzzy logic type-1 owes its growth to [36] who developed fuzzy logic controller (FLC) that is the basis of all FLS's. This control is assembled in four modules as a function of the variables. It contains a fuzzifier, a fuzzy rule base in the form (Eq. 16). It is used for generating the inputs and an evaluation inference module that has a defuzzifier used as a translator. The type-1 FLS's are still in use. Some industrial applications of type-1 in various fields are shown in [3, 10, 35, 37, 39].

$$Rule\ n: IF\ X_1\ is\ A_1\ and\ \ldots\ and\ X_n\ is\ A_n, Then\ Y_n = B_n \quad \textbf{(Eq. 16)}$$

Where: X_n represents the fuzzy input variables, A_n represents the inputs, Y_n represents the fuzzy output and B_n represents the crisp output.

2.3.4.1 Fuzzy implications

The fuzzy implication is a combination of *N* rules that generates an output answer, the inference engine is operated by T-norm, for example, the maximum T-norm operator is given by (Eq. 17) or the minimum operator S-norm yields (Eq. 18). The T-norm and/or the S-norm operators represent the "if" part of the fuzzy rules, however in the literature there are a number of operators that can be applied as is shown in [37],

$$T[\mu_A(x), \mu_B(x)] = max_{\mu_{A\cap B}}(x) \quad \textbf{(Eq. 17)}$$

Where: $\mu_A(x), \mu_B(x)$represents the membership of x_i, respectively to their fuzzy set x_i, *S* represents the *S*-norm and $\mu_A(x), \mu_B(x)$ both of which, yields (Eq. 19),

$$S[\mu_A(x), \mu_B(x)] = min_{\mu_{A\cup B}}(x) \quad \textbf{(Eq. 18)}$$

$$\mu_{x_i} = -\frac{e^{-(x_i-\overline{x}_i)^2}}{2\sigma^2} \quad \textbf{(Eq. 19)}$$

Where: μ_{x_i} represents the membership value of x_i, $\overline{x}_i$ represents the mean of the fuzzy set, x_i' represents the input and σ represents the standard deviation or the spread of the fuzzy set.

The fuzzy value for the output or the firing strength for every rule is generated by Sup★star composition in according to the equation (Eq. 20), extracted from [17] and [37]. This relation produced the fuzzy output for the whole system in the "if" part of the product of all variables present in the rule.

$$\mu_{R\circ S}(x,z) = sup_{y\in v}[\mu_R(x,y) * \mu_S(y,z)] \quad \textbf{(Eq. 20)}$$

Where: * denotes product, μ_R, μ_S represents the universe of the input variables and their relations.

2.3.4.2 Fuzzy basis function

Reported in the literature [35] uses the IBI (Individual Base Inference) model that is the basis for the proposed approach (See section 3), where each rule is performed using all inputs and gives a value as a consequent. Each value is a partial rule and each rule is defined by a fuzzy set. The Fuzzy basis function (FBF) combines all fuzzy rules outputs that are activated as shown in the equation (Eq. 20). This is defined in [17], [37] by:

Definition 1: The set of fuzzy systems with a singleton fuzzifier, product inference, centroid defuzzifier, and Gaussian membership function consist of all fuzzy basis functions (FBF) of the form (Eq. 21),

$$FBF = \frac{\sum_{j=1}^{M} \bar{z}^j (\prod_{i=1}^{n} \mu_i^j(x_i))}{\sum_{j=1}^{M} (\prod_{i=1}^{n} \mu_i^j(x_i))} \quad \textbf{(Eq. 21)}$$

Where: $\bar{z}^j$ is the output for a point in the space R, $\mu_i^j(x_i)$ represents the output of the rule: *Ri*, *i=1, …, j*.

3. Experimental method

The proposal uses classic VRP with one vehicle and unlimited capacity. Uncertainty is not present due to only is required the optimization of the route.

The proposal uses 1+1 ES to generate routes provided to a fuzzy system and then the best routes are selected by the 1+1 ES, so we referred to it as Evolutionary Strategy of fuzzy logic vehicle routing problem (ESFLSVRP). The fuzzy system evaluates the possible routes and later uses the 1+1 ES to select the best option. The methodology consists in the following steps:

- Determine the number of clients.
- Make a row vector from 1 to n, (Eq. 22),

$$r = (x_1, x_2, \dots, x_n) \quad \textbf{(Eq. 22)}$$

Where: x is a client and n represent the number of clients, r represents a route in the form of a vector that is a route that forms part of the population that requires optimization to reduce costs.

- From (Eq. 22) is obtained a vector m that is recombined to generate the one that represents the possible routes to be evaluated. The matrix generated ($n{\times}m$; $n{=}m$) is symmetric (Eq. 23).

$$\begin{vmatrix} 0 & 4 & 5 & 8 \\ 4 & 0 & 3 & 7 \\ 5 & 3 & 0 & 1 \\ 8 & 7 & 1 & 0 \end{vmatrix} \quad \textbf{(Eq. 23)}$$

- Arrange a coordinate matrix with the vertices of the connection to obtain an address (x, y) as shown in (Table 2). e. g. Coordinate (1, 1) generates “11” as an address.

Table 2. Coordinate matrix.

Client	0	1	2	3
0	00	01	02	03
1	10	11	12	13
2	20	21	22	23
3	30	31	32	33

- Transform the coordinate matrix (Table 2) to a vector for every client with the addresses (Table 3).

Table 3. Address vector.

Address															
0	1	2	3	10	11	12	13	20	21	22	23	30	31	32	33

- Calculate the distance matrix (Table 4); this is obtained with the values of the arches or connections between two vertexes. E.g. (Eq. 23) for n=3. The first column represents the distance between the depot and the client i.

Table 4. Cost matrix.

Client	0	1	2	3
0	-	4	5	8
1	4	-	3	7
2	5	3	-	1
3	8	7	1	-

- Convert the cost matrix (Table 5) to a column vector (Table 6). The conversion is done when the columns, two to n are placed in order over the preceding column.
- Table 4 is generated using the data of the generated route, i.e. Distance between the client 1 and client 6.
- The connections are obtained using the distance from client to client. E.g. From client 10 to depot, depot to client 2, etc. obtaining all distances between client to client, depot to client and client to the depot.

Table 5. Costs vector (Transposed).

Distance															
0	4	5	8	4	0	3	7	5	3	0	1	8	7	1	0

- Assign to each coordinating distance variable. (Table 5).

Table 6. Relation matrix (Address/distance).

Address	**Distance**
0	0
1	4
2	5
3	8
10	4
11	0
12	3
13	7
20	5
21	3
22	0
23	1
30	8
31	7
32	1
33	0

- Create a parent with a randomly ordered vector; do this to obtain the set number of parents.
- Do the ES steps.
  - Fix: number of parents and children, clients, and generations.
  - Create input matrices (Parents and children) of size (Eq. 24), e.g. Parent (Eq. 25), children (Eq. 26).

$$M = n \times m \qquad \textbf{(Eq. 24)}$$

$$\begin{vmatrix} x_1 & x_2 & x_3 \\ x_1 & x_3 & x_2 \\ x_3 & x_1 & x_2 \end{vmatrix} \qquad \textbf{(Eq. 25)}$$

$$\begin{vmatrix} x_1 & x_3 & x_2 \\ x_2 & x_3 & x_1 \\ x_3 & x_1 & x_2 \end{vmatrix} \qquad \textbf{(Eq. 26)}$$

- Assemble the route (Eq. 27)

There are some computational constraints that require unconventional solutions such as: the use of *0* (Zero) at the beginning of a number in order to avoid the use of zero, the solution is generated using numbers >1, then a x_0 is added at the beginning and the end of the route vector (Eq. 28), which represents the arches that connect the first and the last client to the depot, this fulfills the constraints (Eq. 8 and Eq. 9).

$$R_i = \{x_1, x_2, x_3\} \qquad \textbf{(Eq. 27)}$$

$$R_i = \{x_0, x_1, x_2, x_3, x_0\} \qquad \textbf{(Eq. 28)}$$

- Create with (Table 6) the fuzzy rule base.

  Make the fuzzy system to obtain the costs of routes (Eq. 29).

$$\sum_{i=1}^{n} x_i \, , \forall \; x_i \in \; R_i \qquad \textbf{(Eq. 29)}$$

Use the tournament selection to select the minimum cost route with ES.

3.1 Procedure

The simplest way to define the proposal is in a multiple input, single output (MISO) system (Fig. 4), which consists in *n* inputs provided by ES and one output given by fuzzy model and optimized by ES.

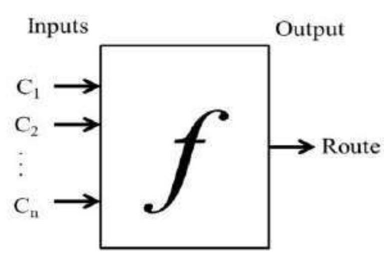


Figure 4. Multiple input single output system. Adapted from: [38].

The simplest way to evaluate is by testing all possible solutions to obtain an optimal route with the minimum cost, but spending a lot of resources. The ESFLSVRP uses graph theory on Platonic solids (Fig. 5) to generate bigger problems with every solid simulating a scenario or instance to make decisions. This approach expends only seconds in execution time for emulate the routes. The ESFLSVRP arises as new metaheuristic to the most commonly used models.

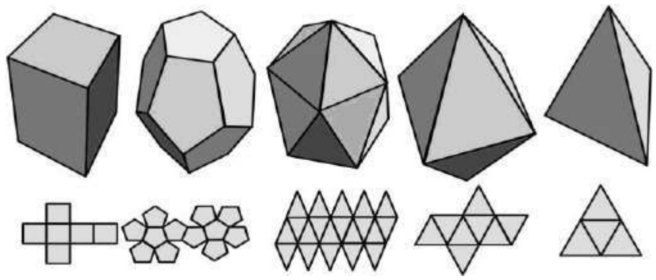


Figure 5. Platonic solids. Adapted from: [39].

Experimental tests were implemented in MatLab R2009a whit a 1+1 ES and type-1 singleton fuzzy logic system. The experiments were conducted as follows: every experiment was tested with 2, 5, 10, 25, 50 and 200 generations with 10 individuals per generation, 4 times for networks with more than 10 individuals.

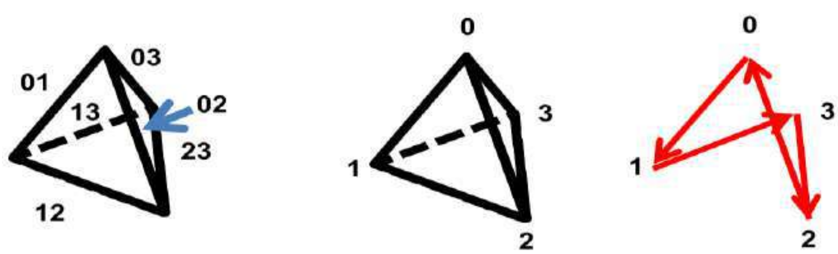
The model was tested using platonic solids (PS) for creating a priori route. The database used was adapted from [40] to meet the requirements of the platonic solids. The icosahedron and the 30 client were extracted from [30] because the database of [40] has only 18 clients plus the depot. In the experiment the depot is represented as a client and the matrix is symmetric. The experimented cases are: They are described in the next paragraphs.

3.2 Platonic solids

The platonic solids are regular polyhedral [41] used to graphically establish by the Hamiltonian paths and are a reference in graph theory and routing problems. In this case they are used as instances to compare the performance of the proposed model against the SA from [42, 43], the shortest path (SP) or Dijkstra algorithm [44], and are tested and compared with the exact Branch and Bound method (B&B) [45].

3.2.1 Tetrahedron

The tetrahedron has 6 connections (01, 02, 03, 12, 13, and 23) as shown in (Fig. 6a) and 4 vertices described graphically in (Fig. 6b), client 0 is the depot, and the rest represent a regular client. Routes for the three clients should be made (Fig. 6 c).

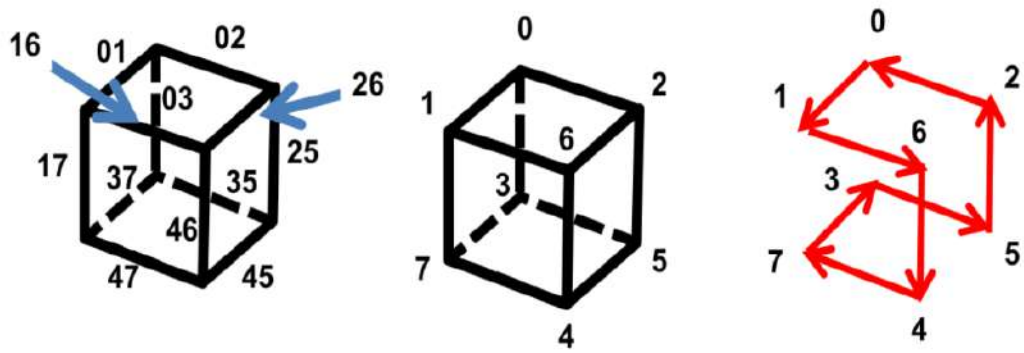


a) b) c)

Figure 6. Tetrahedron. a) Arches or connections, b) Vertexes, c) Sample route. Source: authors.

3.2.2 Cube

The cube, due to its geometry, presents more complexity because of the number of connections; the final count has 16 connections including internal diagonals, (See Fig. 7a) and 8 vertexes (Fig. 7b) to make a route of 7 clients (Fig.7c).

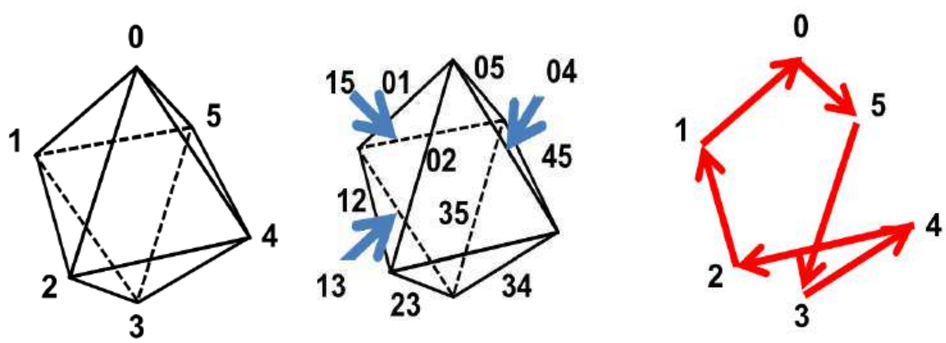


a) b) c)

Figure 7. Cube. a) Arches or connections b) Vertexes, c) Sample route. Source: authors.

3.2.3 Octahedron

The octahedron with its number of faces and vertexes (Fig. 8a and Fig. 8b) has only one route for five clients (Fig. 7c). In this case the number of vertexes restricts the number of clients without worrying of the connections.

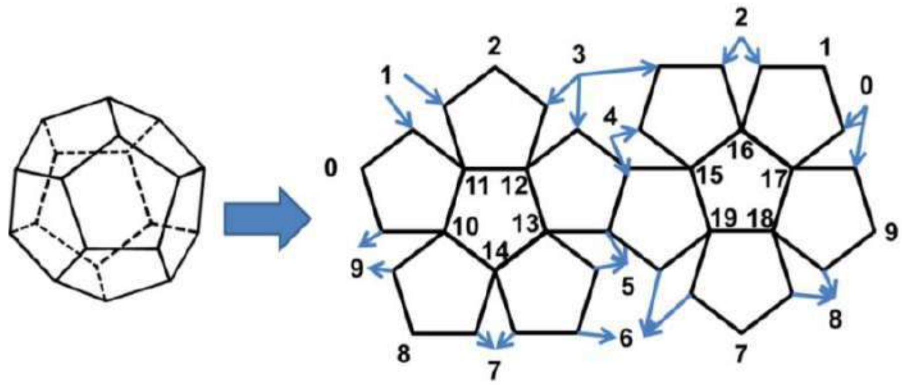


a) b) c)

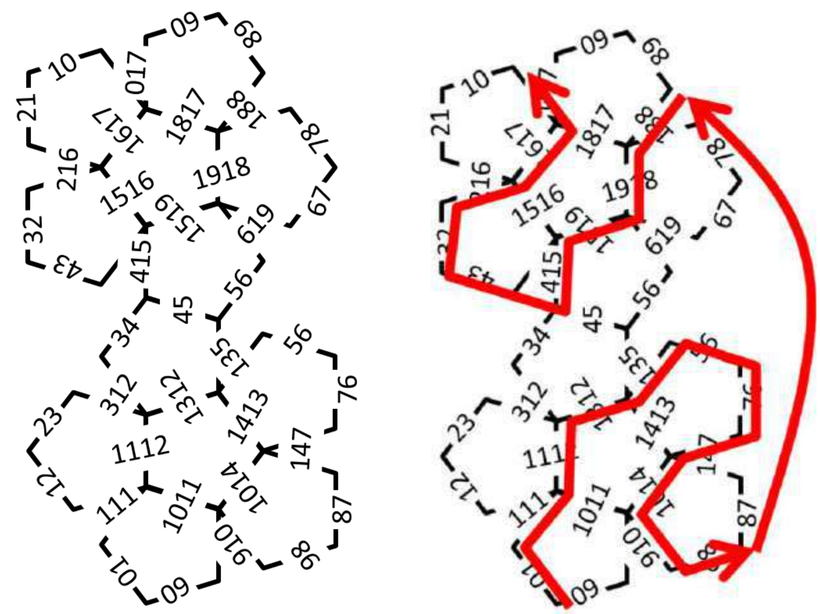
Figure 8. Octahedron. a) Vertexes, b) Arches or connections, c) Sample route. Source: authors.

3.2.4 Dodecahedron

The dodecahedron creates 30 connections (Fig. 9) and 19 clients by their number of vertexes. In this case the number of vertexes restricts the number of clients without worrying of the connections.



a)



b) c)

Figure 9. Dodecahedron. a) Vertexes, b) Connections, c) Sample route. Source: authors.

3.2.5 Icosahedron

Icosahedron creates a route for 11 clients with multiple connections. The proportion between connections and vertexes are 2.5 and despite having multiple connections per node (> 2), these are not relevant since only one is used to define the arc between vertices to make a route, so the clients are restricted only by the nodes without worrying the about connections.

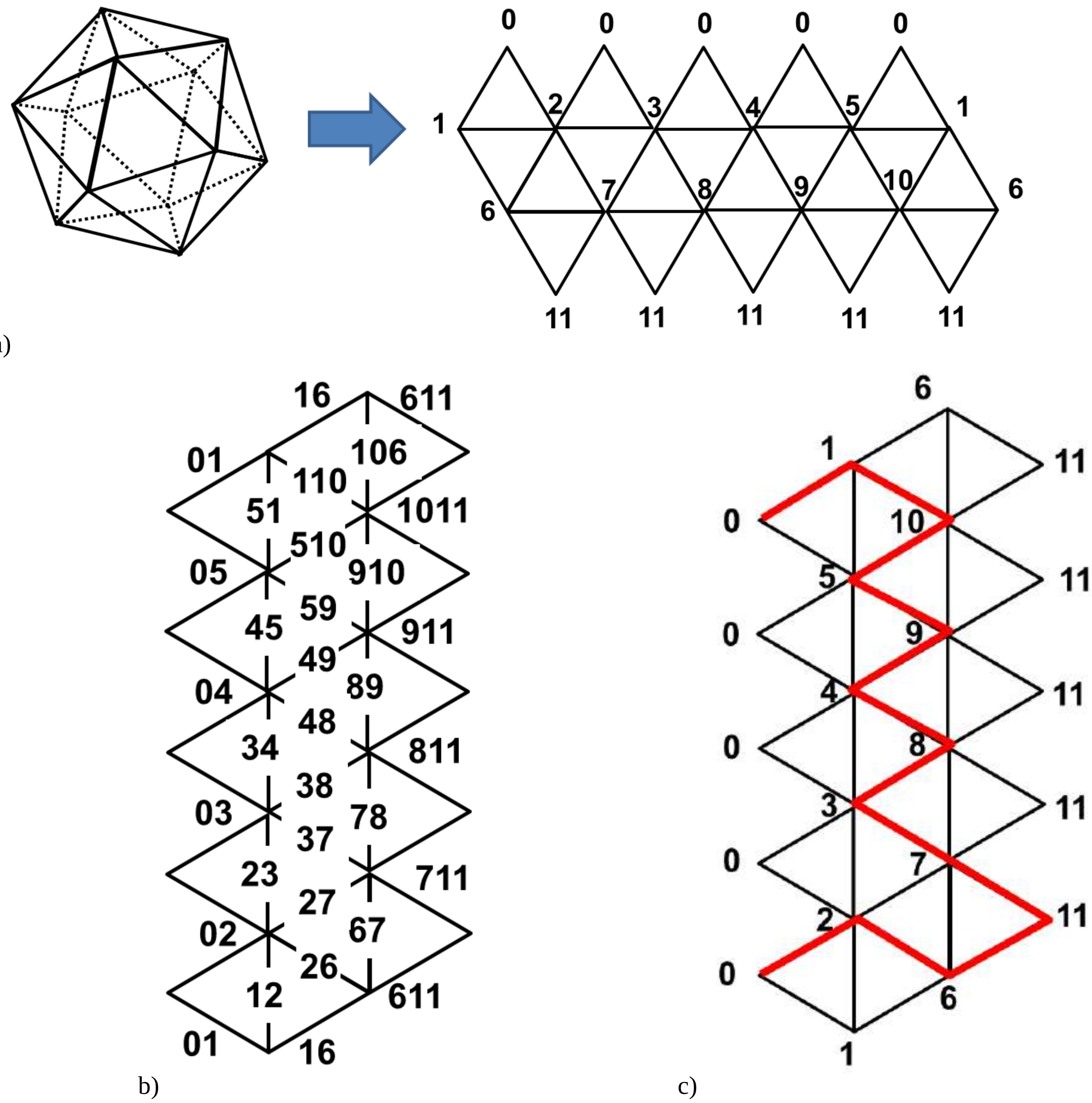


Figure 10. Icosahedron. a) Vertexes, b) Arches or connections, c) Sample route. Source: authors.

3.3. 30 Clients Route

To evaluate the proposal with a bigger problem, 10 and 200 generations with 10 individuals were tested. As benchmarking the SA metaheuristic was tested using an adaptation instance generating the first thirty data with lower triangular matrix data provided by [30] (Table 7).

The behavior of the larger instances can be observed in the results section. But the behavior of a bigger instances shows a tendency to reduce the error gap and maintain an error rate below 20% as is shown in (Tables 8 and 9). The value mentioned in last sentence is a normal error rate as shown in literature [13, 46].

Table 7. Instance matrix. Adapted from: [30].

Clients	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
0	0																													
1	37	0																												
2	35	36	0																											
3	36	34	37	0																										
4	34	40	35	36	0																									
5	36	37	39	38	37	0																								
6	36	42	40	38	35	38	0																							
7	35	42	40	37	38	37	36	0																						
8	41	47	40	42	37	38	34	37	0																					
9	41	48	41	46	35	42	34	39	32	0																				
10	38	45	39	41	34	37	34	37	34	37	0																			
11	36	39	36	38	39	35	36	34	37	41	38	0																		
12	44	45	41	38	41	43	47	43	49	52	50	41	0																	
13	36	33	39	35	37	39	45	41	44	48	43	39	44	0																
14	34	35	40	32	36	40	42	40	45	47	44	40	44	35	0															
15	39	39	41	40	44	43	46	44	49	51	46	44	46	37	39	0														
16	42	35	41	38	39	40	46	42	45	49	44	40	42	38	39	37	0													
17	38	44	46	43	38	50	50	46	51	53	49	44	42	40	46	38	37	0												
18	45	41	46	42	42	46	50	48	51	55	50	46	41	37	43	36	37	38	0											
19	44	44	47	43	41	47	48	46	50	54	50	47	44	39	43	36	39	40	38	0										
20	45	45	49	40	47	47	50	48	56	54	53	50	42	43	46	41	43	4	40	39	0									
21	46	42	45	39	47	47	46	44	49	51	48	46	46	40	39	43	46	46	45	39	37	0								
22	45	43	41	35	44	42	46	43	46	49	44	41	40	38	39	35	32	41	37	39	44	45	0							
23	40	35	41	37	46	43	38	39	49	51	48	44	45	38	38	37	40	44	45	39	40	39	40	0						
24	37	37	38	41	43	41	44	40	46	47	44	41	47	32	36	36	42	42	40	39	45	42	41	37	0					
25	42	42	47	39	39	43	48	46	50	50	50	47	48	37	41	37	40	45	44	39	43	40	41	41	37	0				
26	37	43	40	40	45	39	41	38	43	46	42	41	51	42	40	43	46	48	47	45	47	44	44	43	39	45	0			
27	39	43	41	41	39	37	40	37	42	44	41	39	50	42	40	42	46	45	47	45	47	44	44	44	39	46	32	0		
28	40	43	41	43	39	39	42	40	44	46	44	41	51	39	40	39	42	47	45	41	44	42	43	39	39	42	36	34	0	
29	41	47	44	39	37	41	43	40	46	48	44	43	52	43	43	43	46	51	49	44	43	40	47	40	43	42	38	37	36	0

4. Discussion and Results

The results obtained show that it is possible to generate a satisfactory route (Table 8) with a few individuals (10 individuals) per generation and a few generations (10 generations) to obtain an accuracy higher than 80% (Table 9). In most cases the generated approximation is satisfactory and it is not necessary to add more individuals or generations in the optimization process to obtain better solutions. The improvement represents a 5% in comparison with the 10 generations test and the route does not show a significant change. The results obtained by the recombination process and the new randomly generated individuals present patterns that conforms optimal sub-routes that are repetitive. For this cause the model cannot evolve to better solutions and requires many generations to evolve to optimal solutions which are not feasible in real life enterprises. The use of more than 10 generations does not provide a substantial gain when compared with the computational time spent in order to generate the route.

Table 8. Total distance (With depot as a client).

Method	Clients						
	4	6	8	12	18	20	30
SP	58	94.5	102	165.5	329	757	1117
ESFLSVRP 200 Generations	58	94.5	134	208.25	414	777	1194.5
ESFLSVRP 10 Generations	58	94.5	134	208.25	439.25	798	1228.5
SA	58	94.5	133	179	395.3	823	1298
Exact method (Branch and bound)	58	94.5	133	182.5	260	715	1073
ANFIS 10 Generations	58	95.5	150.5	250.2	435	765	1377
ANFIS 200 Generations	58	95.5	134	221.5	412	764.5	1302

Table 8 presents a comparative result using classical approaches such as: the exact method, SA, SP and ANFIS algorithms versus the ESFLSVRP its depiction is shown in (Fig. 11) for 10 generations and 10 individuals, and (Fig. 12) for 10 individuals and 200 generations of evolution using the proposed system and a neuro-fuzzy model as benchmark. The ESFLSVRP algorithm presents a better solution compared to the SA heuristic solution, also the ESFLSVRP reduces the total cost in distance of this particular case in an interval between 3 to 7% for small instances, 20 or 30 clients. The routes obtained to 4 and 6 clients converge to the same value in any of the used methods as shown in (Table 9). The solution for 18 clients with the ESFLSVRP model shows less accuracy when it is compared against the SA and the exact method as is shown in (Fig. 13) and (Fig. 14). This fact is a particular case that could be generated by the five connections on each node of the route as can be seen in (Fig. 9). Additionally, the obtained error rates are inside the interval determinate by [46] for an approximation of the optimal route. Also, the big error rates on 18 clients route are inside of interval [33.53% 62.75%] mentioned in [13] when comparing a method with a classical instance.

Table 9. Error of the model ESFLSVRP vs. Heuristics and ESFLSVRP vs. ANFIS (%).

	Heuristic	**SA**	**SP**	**BB**	**ANFIS**	**SA**	**SP**	**BB**	**ANFIS**
	Generations	200	200	200	200	10	10	10	10
	4	0	0	0	0	0	0	0	0
	6	0	0	0	**1.04***	0	0	0	**1.04***
Clients	8	0.7	31	**0.75**	0	**0.7**	31	**0.75**	**10.96***
	12	**16**	25	**14.1**	**5.98***	**16**	25	**14.1**	**16.76***
	18	47.3	25.83	59.23	0.48	11.18	33.5	68.9	0.97
	20	**5.58***	**2**	**11.61**	1.63	**3.04***	**5**	**11.61**	**4.31**
	30	**7.97 ***	**6**	**14.49**	8.25*	**5.35***	**9**	**14.49**	**10.78***

*The * denotes an improvement in the optimization and presents the least cost on the route.*

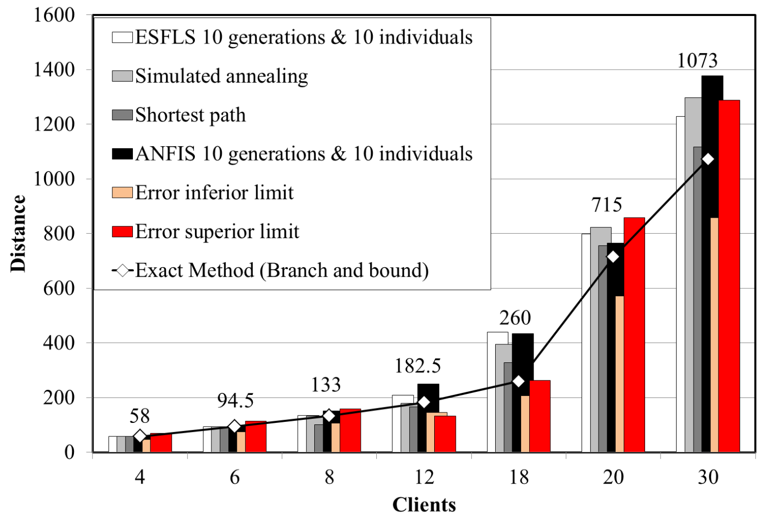


Figure 11. Experimental results of ESFLSVRP with 10 individuals and 10 generations of evolution versus metaheuristics and exact method Branch and Bound. In distance.

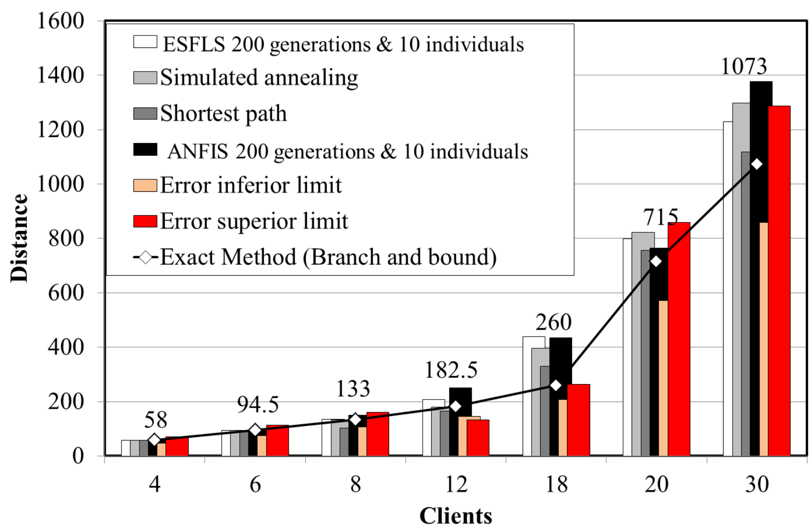


Figure 12. Experimental results of ESFLSVRP with 10 individuals and 200 generations of evolution versus metaheuristics and exact method Branch and Bound. In distance.

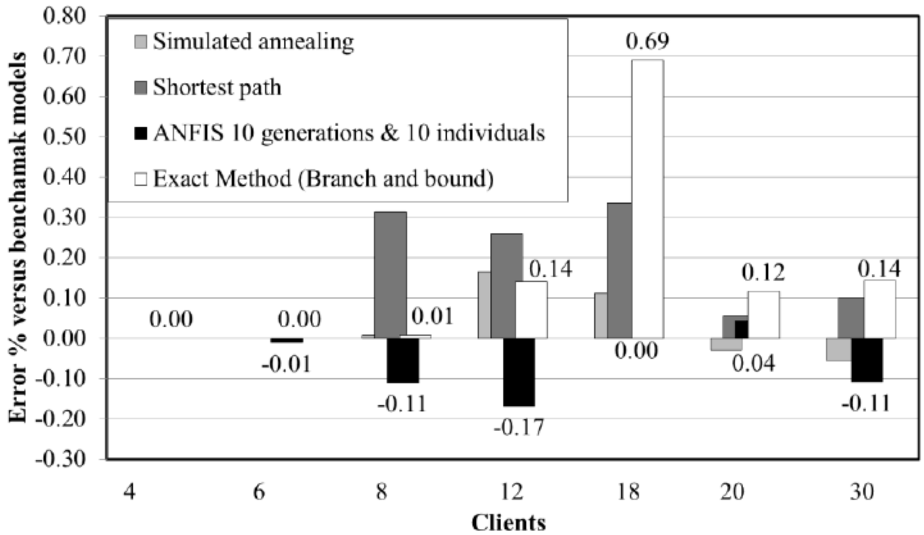


Figure 13. Error rates or enhancements (negative values) of ESFLSVRP with 10 individuals and 10 generations of evolution versus metaheuristics and exact method Branch and Bound. In Percentage.

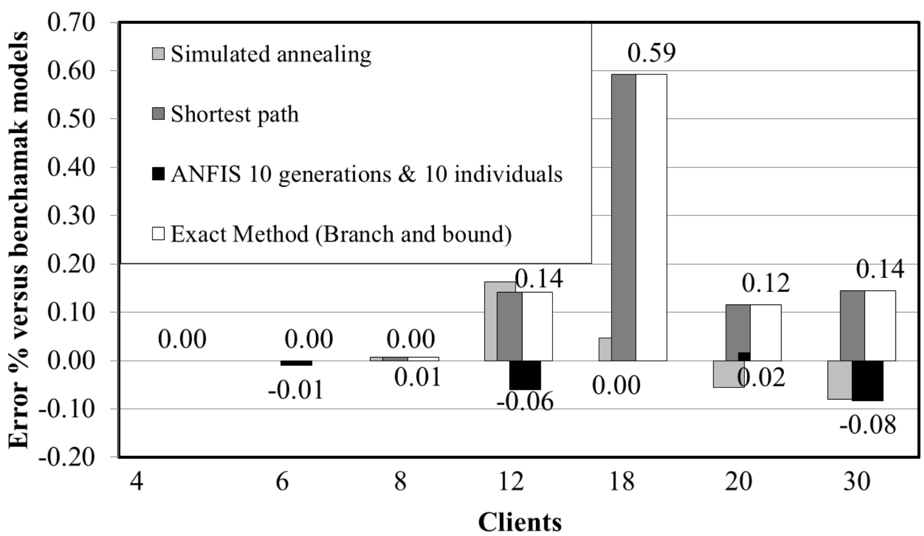


Figure 14. Error rates or enhancements (negative values) of ESFLSVRP with 10 individuals and 200 generations of evolution versus metaheuristics and exact method Branch and Bound. In Percentage.

The exposed results (Figs. 11 to 14) show that the gap between the SA approximations versus the ESFLSVRP model grows up when more clients are added and it represents an enhancement. The proposal shows better accuracy and fitting when it is compared with the SA and at least the same error of SP solution. It corroborates the statement of [47] that the ANN accuracy is less than or close to by not higher than 80%.

The results depicted in (Fig. 12 and 13), show the behavior of the proposed method that presents a trend to generate a better approximation than a metaheuristic method as in this case when it is compared to benchmark models. The report presented by [3] mentions that the behavior for an unknown data set solved by approximation presents an accuracy between 65-71% which shows a decrement in the accuracy of at least 15% versus the proposed model. (Fig. 13) presents the error percentages obtained with 200 generations of the ESFLSVRP model showing an error rates of less than 20%, that is reported in the literature as a normal gap in the approximation or the path calculation [11, 46].

Table 10 presents the computational time consumed by the ESFLSVRP to generate a route. The SA for small instances (20 or 30 clients) spends 140 seconds in execution time while the ESFLSVRP requires only 13.44 seconds. This represents a saving of 90% in execution time which is an enhancement in the creation of the routes: In-[13] it is shown that their approach requires 100 seconds tested five times for small instances. Furthermore, our approach presents a better accuracy when an optimal route is generated using only 10 generations (Fig. 15). The case of 200 generations requires more time as 20 minutes that is less than the one presented in [11] of one hour. Also [48] mentions that one hour is required to obtain a result with an exact method such as B&B. In [13] a gap interval of 33.53% to 62.75% on the error compared with a classic instance is mentioned. In this case it is a CVRP compared with the instances generated by Augerat [10]. An error of 30% on the route prediction is presented in the proposal of [18]. This quantity of time 1,203 seconds is generated by the nature or evolution of algorithm used by the generations require 20 individuals with 30 nodes obtaining 1,200 evaluations with the fuzzy model and 80 by the ES per generation and this is multiplied by 200 generations produce 256,000 mathematical operations that could be limited by the computer capacities but, with these restrictions are three times less than the exposed in literature [11 and 48].

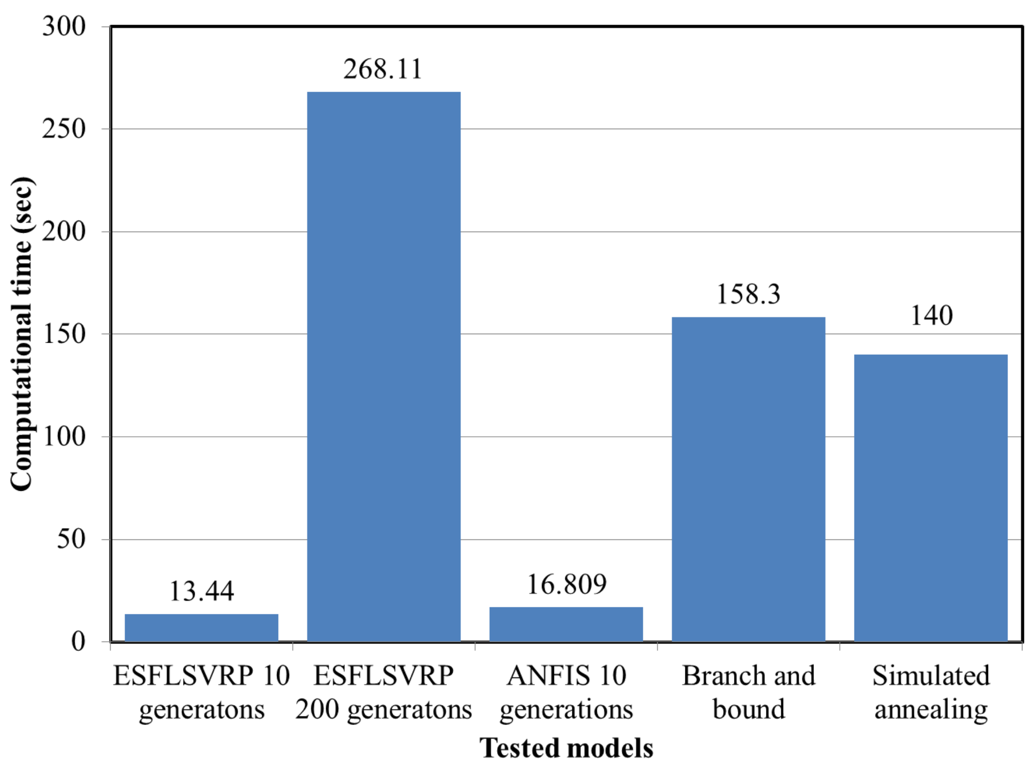


Figure 15. Approximation of the ESFLSVRP for 20 clients.

The waiting time in the SA is produced because its algorithm requires a lot of time to generate the cooling factor that reduces the temperature in the order of 5 degrees per iteration. This is caused by the initial temperature that usually is fixed to 10,000 degrees. Also, the lead time is produced by the repetitive calculation of the route in every tested combination, the tests are produced by trial and error and the stopping criteria is zero degrees or close to zero. Additionally, the solution is saved if and only if it presents better results than the previous saved route.

Table 10. The computational time consumed.

	Time (Sec.)						
Clients	4	6	8	12	18	20	30
ESFLSVRP 10 gen.	0.96	1.37	1.7	1.7	16.6	13.44	62
ESFLSVRP 200 gen.	18.86	26.67	33.89	33.89	301.58	268.11	1203
ANFIS 10 gen.	0.77	1.16	1.55	2.28	14.8	16.6	1197.3

Table 9 presents the comparison between two approaches of the proposed hybrid intelligent system model (ESFLSVRP) with 10 generations of optimization versus 200 generations of optimization. The results show that the proposal provides an accuracy over 84% that is better than a common metric to evaluate intelligent systems as ANN as [3 and 47] mentioned, also the neural networks shows a chaotic behavior as shown in (Fig. 14) that shows the accuracy of the prediction model ESFLSVRP with 10 and 200 generations of optimization, the gap in the accuracy of the model with 10 generations versus 200 is too small in comparison with the computational time spend on the approximation. In this table could be observe that the errors are very similar to reported in the literature by [46] where it is mentioned “the average Gap of each set is more than 13.2% and up to 20.0%” it is practically equivalent to the error or gap obtained in the tests of 14.49% for the 10 generations and is equal with 200 generations in the worst case (excluding the case f 18 clients because it is a particular case due to their multiple connections and need to be studied in future works). These results are within the variations that appear in the literature as a gap between 13.2 to 20% in [46] and higher than 1,700% than the optimal cost as mentioned by [11], [13] corroborate this with the affirmation that the exact methods require a lot of time to generate a feasible solution with bigger instances (<50).

A similar behavior is presented in this case because of its random nature the optimal solution could not be found, and the stopping criterion established is the number of generations.

Table 9. Accuracy of the proposed model (%).

	Clients						
Model	4	6	8	12	18	20	30
ESFLSVRP 10 Generations	**100**	**100**	**99.248**	**85.89**	31.06	88.392	85.508
ESFLSVRP 200 Generations	100	100	99.020	84.441	40.77	91.810	**89.123**
ANFIS 10 Generations	100	98.94	52.45	48.82	**98.9**	**98.94**	76.72

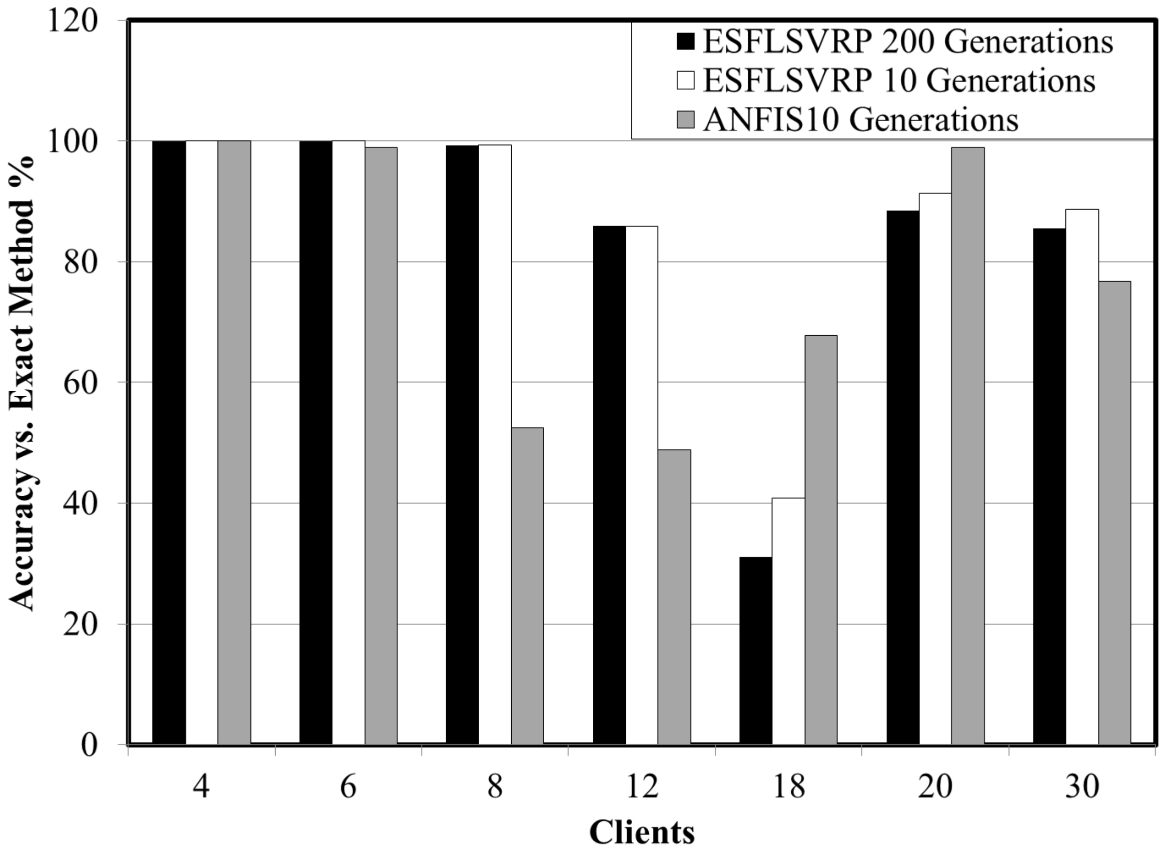


Figure 14. Accuracy of the proposed model.

5. Conclusion

As is shown in the literature, the classical VRP is only used in small and medium sized enterprises. Also, the literature does not report instances to evaluate new proposals and these instances need to be developed and tested with several methods to create an accurate evaluation.

The literature does not show the use of fuzzy models to solve the routes, they are only used to avoid uncertain parameters. In contrast as mentioned in Mendel [16], the type-1 fuzzy systems are not capable to manage the uncertainties. But can evaluate fuzzy numbers.

The instances used are small with less than 100 customers as shown in [11-14]. With less than 50 clients.

The random nature of the evolutionary strategy provides an option to explore all possibilities in the combinatorial production of different Hamiltonian cycles that provide a sample route to be evaluated by fuzzy logic. While the fuzzy logic in this approach provides the possibility to evaluate all combinations in a phase by marking one route without worrying about the complexity (size of the matrix) of the problem used on the fuzzy rule base.

The proposed approach generates two solutions because of its formulation that forms a Hamiltonian path starting and ending at the same point (Depot). The optimal route can be placed in inverse order without changing the total distance needed to complete it.

The computational times reported in the literature [13] are as much as six times longer than the obtained times for small instances.

The principal risk of implementing the ANN model is the problem of its premature convergence that means that it does not reach the optimum and gets trapped in a local optimum. Also, it spends a lot of time and only gets accuracy of less than 80% because it is trapped in a local optimum and repeats these cycles until the stopping criterion is met if it exists [47]. Because of this the pure ANN is not used for the model.

As future work is the use of the proposed model with pure neural networks to evaluate the convergence and precision generated as in [47]. Also, is the test of it using bigger instances.

References

[1] Dantzig, G., Ramser, J., (1956). "The truck dispatching problem." *Managements Science* 6(1), Pp. 80-91. https://doi.org/10.1287/mnsc.6.1.80

[2] Erdelić, T. & Carić, T., (2019). "A survey on the electric vehicle routing problem: variants and solution approaches". *Journal of Advanced Transportation*, 2019, Pp.1-48, https://doi.org/10.1155/2019/5075671

[3] Montes Dorantes P.N., Ireta Sánchez P.H., Velarde Cantú J.M., Liñán García E., Méndez G.M. (2018). "Design and optimization of Distribution Routes Using Evolutionary strategy and Type-1 Singleton Neuro-fuzzy systems". *IEEE LATIN AMERICA TRANSACTIONS*, 16(5), Pp. 1499-1507. DOI: 10.1109/TLA.2018.8408447.

[4] Goel R. & Maini R. (2017). "Vehicle routing problem and its solution methodologies: a survey". *International Journal Logistics Systems and Management*. 28(4), Pp. 419-435. https://doi.org/10.1504/IJLSM.2017.087786

[5] Han, M. & Wang, Y. (2018, December). "A survey for vehicle routing problems and its derivatives". *In IOP Conference Series: Materials Science and Engineering*. 452(4), Pp. 042024. IOP Publishing. https://doi: 10.1088/issn.1757-899X

[6] Bhuvaneswari, M., Eswaran, S., & Rajagopalan, S. P. (2018). "A survey of vehicle routing problem and its solutions using bio-inspired algorithms". *International Journal of Pure and Applied Mathematics*,118(9), Pp. 259-64.

[7] Kassem, S., Korayem, L., Khorshid, M., & Tharwat, A. (2019, October). "A hybrid bat algorithm to solve the capacitated vehicle routing problem". In 2019 Novel Intelligent and Leading Emerging Sciences Conference (NILES) (Vol. 1, pp. 222-225). IEEE.

[8] Ireta Sanchez P.H., Montes Dorantes P.N., Mendoza A. y Solis R. G. (2019). "Problema de trazo de rutas vehiculares con terminación diferente al depósito central". En: Sánchez Lara B, Torres Mendoza R.M., Gómez Maturano J., Cedillo Campos M.G. (eds) Innovación Logística en América Latina: Nuevas Perspectivas. Asociación Mexicana de logística y cadena de suministro A.C., Universidad Nacional Autónoma de México. ISBN 978-607-96403-1-6, pp. 82-88.

[9] Pérez-Salas G., González-Ramírez R.G., Cedillo-Campos M.G. (2014). "A holistic approach for measuring the international logistics costs". *2nd International*

Congress on Logistics & Supply Chain, (CiLOG2014) proceedings of, Universidad Anahuac. Huixquilucan, Estado de México. October 16-17. 2014.

[10] http://vrp.atd-lab.inf.puc-rio.br/index.php/en/

[11]Brekkå, I., Randøy, S., Fagerholt, K., Thun, K., & Vadseth, S. T. (2022). "The Fish Feed Production Routing Problem". *Computers & Operations Research*, 144, Pp. 105806. https://doi.org/10.1016/j.cor.2022.105806

[12] Aliahmadi, S. Z., Barzinpour, F., & Pishvaee, M. S. (2020). "A fuzzy optimization approach to the capacitated node-routing problem for municipal solid waste collection with multiple tours: A case study". *Waste Management & Research*, 38(3), Pp.279-290. https://doi.org/10.1177/0734242X19879754

[13] Tordecilla, R. D., Martins, L. D. C., Panadero, J., Copado, P. J., Perez-Bernabeu, E., & Juan, A. A. (2021). "Fuzzy simheuristics for optimizing transportation systems: Dealing with stochastic and fuzzy uncertainty". *Applied Sciences*, 11(17), Pp.7950. https://doi.org/10.3390/app11177950

[14] Sánchez-Ansola, E., Pérez-Pérez, A. C., & Rosete, A. (2020). "School Bus Routing Problem with Fuzzy Walking Distance". *Polibits*, 62, Pp.69-75. https://doi.org/10.17562/PB-62-8

[15] Zhang, S., Chen, M., Zhang, W., & Zhuang, X. (2020). "Fuzzy optimization model for electric vehicle routing problem with time windows and recharging stations". *Expert systems with applications*, 145, Pp.113123. https://doi.org/10.1016/j.eswa.2019.113123

[16] Mendel J. M. (2017). Uncertain rule-based fuzzy systems. Introduction and new directions, Springer.

[17] Goel, R. (2019). "Fuzzy based parameter adaptation in ACO for solving VRP". *International Journal of Operations Research and Information Systems (IJORIS)*, 10(2), Pp. 65-81. DOI: 10.4018/IJORIS.2019040104

[18] Kondratenko, Y., Kondratenko, G., Sidenko, I., & Taranov, M. (2020, July). "Fuzzy and evolutionary algorithms for transport logistics under uncertainty". *In International Conference on Intelligent and Fuzzy Systems* (pp. 1456-1463). Springer, Cham. https://doi.org/10.1007/978-3-030-51156-2_169

[19] Jin, C. X., Zhang, Y., & Li, F. C. (2021, June). "Research on Path Optimization Problem Based on Satisfaction Degree in Fuzzy Demand Environment". *In Journal of Physics: Conference Series* 1955(1), Pp. 012056. IOP Publishing. doi: 10.1088/issn.1742-6596

[20] Oliva, D., Copado, P., Hinojosa, S., Panadero, J., Riera, D., & Juan, A. A. (2020). "Fuzzy simheuristics: Solving optimization problems under stochastic and uncertainty scenarios". *Mathematics*, 8(12), Pp. 2240. https://doi.org/10.3390/math8122240

[21] Peker, I., Caybasi, G., Buyukozkan, G., & Gocer, F. (2019, July). "Evaluation of home health care vehicle routing methods by intuitionistic fuzzy AHP". *In International Conference on* Intelligent and Fuzzy Systems (pp. 607-615).

Springer, Cham. DOI: 10.1007/978-3-030-23756-1_74

[22] Yılmaz, M. (2021). “Hierarchical Chinese postman problem with fuzzy travel times”. *Iranian Journal of Fuzzy Systems,* 18(5), Pp. 87-105. DOI: 10.22111/IJFS.2021.6257

[23] Zheng, J., Wang, S., Wang, L., Chen, J. F., Wang, L., Hao, J.& Sun, Z. (2020, July). “A two-stage algorithm for fuzzy online order dispatching problem”. *In 2020 IEEE Congress on Evolutionary Computation (CEC).* Pp. 1-8. IEEE. DOI: 10.1109/CEC48606.2020.9185858

[24] Çakır, E., Ulukan, Z., Kahraman, C. (2022). “A Fuzzy Modeling for Time Constrained Vehicle Routing Problem”. In: Kahraman, C., Cebi, S., Cevik Onar, S., Oztaysi, B., Tolga, A.C., Sari, I.U. (eds) Intelligent and Fuzzy Techniques for Emerging Conditions and Digital Transformation. INFUS *2021*. *Lecture Notes in Networks and Systems,* 307. Pp.511-519. Springer, Cham. https://doi.org/10.1007/978-3-030-85626-7_60

[25] Zacharia, P., Drosos, C., Piromalis, D., & Papoutsidakis, M. (2021). “The vehicle routing problem with fuzzy payloads considering fuel consumption”. *Applied Artificial Intelligence*, 35(15), Pp. 1755-1776. https://doi.org/10.1080/08839514.2021.1992138

[26] Wang, K., Lan, S., & Zhao, Y. (2017). “A genetic-algorithm-based approach to the two-echelon capacitated vehicle routing problem with stochastic demands in logistics service”. *Journal of the Operational Research Society*, 68(11), Pp. 1409-1421. https://doi.org/10.1057/s41274-016-0170-7

[27] Zhang, S., Chen, M., Zhang, W., & Zhuang, X. (2020). “Fuzzy optimization model for electric vehicle routing problem with time windows and recharging stations”. *Expert systems with applications*, 145, 113123. https://doi.org/10.1016/j.eswa.2019.113123

[28] Karimi-Mamaghan, M., Mohammadi, M., Meyer, P., Karimi-Mamaghan, A. M., & Talbi, E. G. (2022). “Machine learning at the service of meta-heuristics for solving combinatorial optimization problems: A state-of-the-art”. *European Journal of Operational Research*, 296(2), Pp. 393-422. https://doi.org/10.1016/j.ejor.2021.04.032

[29] Gupta, P., Govindan, K., Mehlawat, M. K., & Khaitan, A. (2022). “Multiobjective capacitated green vehicle routing problem with fuzzy time-distances and demands split into bags”. *International Journal of Production Research,* 60(8), Pp. 2369-2385. https://doi.org/10.1016/j.ejor.2021.04.032

[30] Villalva, J. R., Monserrate, G. J., (2013). “*Optimización del plan de visitas de inspección en el servicio de mantenimiento de áreas verdes de la ciudad de Guayaquil*”. (Master dissertation, ESPOL).

[31] Rechenberg, I. (1973) Evolutionsstrategie, Optimierung technischer Systeme nach Prinzipien der biologischen Evolution. Stuttgart: Frommann Holzboog

[32] Darwin, C.R. (1859). On the origin of species by means of natural selection, or the preservation of favoured races in the

struggle for life. 1st edn. London: John Murray.

[33] Mendel, J. G. (1865) Versuche über Plflanzenhybriden. Verhandlungen des naturforschenden Vereines in Brünn, Bd. IV für das Jahr, Abhandlungen: P.p. 3-47.

[34] Droste, S., Jansen, T., Wegener, I. (2002). “On the analysis of the (1 + 1) evolutionary algorithm”. *Theoretical Computer Science*, 276, Pp.51–81, https://doi.org/10.1016/S0304-3975(01)00182-7.

[35] Montes-Dorantes, P. N., Santoyo, A. M., & Méndez, G. M. (2018). “Modeling Type-1 Singleton Fuzzy Logic Systems using statistical parameters in foundry temperature control application”. *Smart and Sustainable Manufacturing Systems*. 2(1), Pp. 180-20 .https://doi.org/10.1520/SSMS20180031

[36] Mandami, E. H., Assilian, S. (1975). “Experiment in Linguistic Synthesis with a Fuzzy Logic Controller”, *International Journal of Man-Machine Studies*, 7(1), Pp. 1-13, https://doi.org/10.1016/S0020-7373(75)80002-2.

[37] Mendel, J. M. (2001). Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions, Prentice.

[38] Joshaghan, M. A., Kamyad, A. V., Razavi, A. R., Norouzy, (2014). “A. Evaluation of the Nutritional Effects of Fasting on Cardiovascular Diseases, Using Fuzzy Data Mining.” *Journal of Fasting and Health*, 3(1), Pp. 14-21.

[39] Weisstein, E. W. Platonic Solid. MathWorld: A Wolfram Web Resource. [Online], [date of reference February 24th of 2022]. Available at: http://mathworld.wolfram.com/PlatonicSolid.html.

[40] Brito, J., Campos, C., Castro, J.P., Martínez, F.J., Melián, B., Moreno, J.A., Moreno, J.M. (2008). “Fuzzy Vehicle Routing Problem with time windows”, *Proceedings of IPMU'08*, Pp. 1266-1273.

[41] Slodowy, P. (1983). Platonic solids, Kleinian singularities, and Lie groups. In Algebraic geometry. Pp. 102-138. Springer Berlin Heidelberg.

[42] S. Kirkpatrick, C. C. D. Gelatt & M. P. Vecchi, (1983). “Optimization by Simulated Annealing. Science”. *Science, New Series*, 220(4598), pp. 671-680.

[43] V. Cerny, (1985). “Thermodinamical Approach to the Traveling Salesman Problem:An Efficient Simulation Algorithm”. *Journal of Optimization Theory and Applications*, 45(1), Pp. 41-51, 1985. https://doi.org/10.1007/BF00940812

[44] Huang, C. Y. R., Lai, C. Y., & Cheng, K. T. T. (2009). Fundamentals of algorithms. In Electronic design automation. Pp. 173-234. Morgan Kaufmann.

[45] Fox, B. L., Lenstra, J. K., Rinnooy Kan, A. H. G. and Schräge, L. E. (1978). “Branching from the largest upper bound: folklore and facts”. *European Journal of Operational Research.* 2, Pp. 191-194. https://doi.org/10.1016/0377-2217(78)90092-9

[46] Chevroton, H., Kergosien, Y., Berghman, L., & Billaut, J. C. (2021). “Solving an integrated scheduling and routing problem with inventory, routing and penalty costs”. *European Journal of Operational Research*, 294(2), Pp. 571-589. https://doi.org/10.1016/j.ejor.2021.02.012

[47] Anderson, J.A. Redes Neurales. (2007), Alfaomega.

[48] Forget, N., Gadegaard, S. L., Klamroth, K., Nielsen, L. R., & Przybylski, A. (2022). "Branch-and-bound and objective branching with three or more objectives". Computers & Operations Research, 148, Pp. 106012. https://doi.org/10.1016/j.cor.2022.106012