Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2015, Article ID 713043, 14 pages
http://dx.doi.org/10.1155/2015/713043

Research Article

Reducing the Size of Combinatorial Optimization Problems Using the Operator Vaccine by Fuzzy Selector with Adaptive Heuristics

Oscar Montiel and Francisco Javier Díaz Delgadillo

Instituto Politécnico Nacional-CITEDI, Avenue del Parque No. 1310, Mesa de Otay, 22510 Tijuana, BCN, Mexico

Correspondence should be addressed to Oscar Montiel; oross@ipn.mx

Received 10 March 2015; Revised 5 June 2015; Accepted 8 June 2015

Academic Editor: David Bigaud

Copyright © 2015 O. Montiel and F. J. Díaz Delgadillo. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Nowadays, solving optimally combinatorial problems is an open problem. Determining the best arrangement of elements proves being a very complex task that becomes critical when the problem size increases. Researchers have proposed various algorithms for solving Combinatorial Optimization Problems (COPs) that take into account the scalability; however, issues are still presented with larger COPs concerning hardware limitations such as memory and CPU speed. It has been shown that the Reduce-Optimize-Expand (ROE) method can solve COPs faster with the same resources; in this methodology, the reduction step is the most important procedure since inappropriate reductions, applied to the problem, will produce suboptimal results on the subsequent stages. In this work, an algorithm to improve the reduction step is proposed. It is based on a fuzzy inference system to classify portions of the problem and remove them, allowing COPs solving algorithms to utilize better the hardware resources by dealing with smaller problem sizes, and the use of metadata and adaptive heuristics. The Travelling Salesman Problem has been used as a case of study; instances that range from 343 to 3056 cities were used to prove that the fuzzy logic approach produces a higher percentage of successful reductions.

1. Introduction

In the field of mathematics and computer science, one of the most challenging problems is searching for optimal arrangements of elements within hundreds of thousands of possibilities. Collectively, these problems are known as Combinatorial Optimization Problems (COPs), and their importance lies in the large diversity of real-life problems that can be solved in the industry and engineering sectors by studying them. They become a challenge because using a direct approach of calculating all the possibilities and choosing the best one becomes an unfeasible task, because the number of possible combinations grows quickly compared to checking modestly sized instances, which is beyond the capabilities of even the most powerful today's supercomputers; therefore, a great deal of research has been invested in developing faster and better algorithms for solving COPs [1].

One of the most well-known COPs is the Traveling Salesman Problem (TSP) whose popularity and importance can be attributed to its simple definition but high complexity to solve it, making it an ideal research test problem. Additionally, there are many scientific, real-life industrial and commercial applications that can be analyzed in an analogous way. The TSP consists in finding the shortest tour that a traveling salesman must take between a finite amount of cities, starting and ending in the same city. The TSP is known to be NP-hard [2] and the term was coined by W. R. Hamilton and Thomas Kirkman in the 1800s [3]; it was first formulated as a mathematical problem in 1930 by Karl Menger. Numerous approaches to solving it have been published [4, 5], from which two main categories can be identified: *exact solutions* and *approximate approaches* [6].

The most straightforward *exact solution* algorithm would be to utilize brute force by creating every possible outcome and choosing the best one. This approach is not feasible for larger TSP instances and, thus, better performing algorithms were proposed. One of the best proposals comes under the published name "implementing the Dantzig-Fulkerson-Johnson algorithm for large traveling salesman problems" [7, 8], which was an early description of the computer program solver for the symmetric TSP named Concorde [9]. A modified Concorde's algorithm was used for solving instances up to 85,900 cities optimally [10]. Other approaches such as "Branch-and-Bound/Cut" method described in [11] in 1958 was applied to solve the TSP in 1963 in [12].

The search for exact solutions is not always the best course of action when trying to solve a problem, and this is where *approximate approaches* take the lead. Based on heuristics, these types of algorithms do not guarantee that the optimum solution will be found; however, they provide suboptimal results that can be "good enough" for the task at hand. One of the best known approximate algorithms was published as "Polynomial Time Approximation Schemes for Euclidean Traveling Salesman and Other Geometric Problems" [13]. Some other heuristics examples as well as their adaptation to solve the TSP are as follows: Lin-Kernighan [14, 15], Tabu Search [16, 17], Evolutionary Algorithms [18–20], Ant Colony Optimization [21–23], Bee Colony [24], Neural Network Algorithms [25, 26], Memetic Approaches [27, 28], and hybrid strategies such as Neural Networks [29] and Memetic Algorithms [30] plus Lin-Kernighan local optimization.

Differently from the above proposals, in this work, the goal is not to provide new solving algorithms but to allow that current proposals can perform better by providing an operator that performs a systematic selection, removal, optimization, and reconstruction of TSP instances to reduce the problem size. The ROE method was conceived in [31], and, here, we will contribute to the method by providing better selection strategies (operator) than the existing ones, which includes the use of adaptive heuristics. This will produce a higher percentage of reductions that are part of the global optimum. With the Reduce-Optimize-Expand (ROE) method, we can produce faster and higher quality solutions and, in some cases, provide solutions which was previously impossible due to algorithm or hardware limitations; now, using the fuzzy operator with adaptive heuristics, it will be possible to improve the quality of solutions. We present a series of comparative experiments from the well-known TSPLIB that showcase the improvement of the fuzzy selection strategy to those that were previously published.

The main contributions of this work can be summarized as follows: a general method to reduce the size of TSP instances that uses a rule-based Fuzzy Inference System (FIS) and a procedure to adequate TSP instances for future evaluation by the FIS; this includes the creation of custom made linguistic variables based on TSP instance metadata and the logical rules that drive the FIS. The above contributions improve the ROE methodology [31] providing a higher quantity of reductions that are part of the global optimum.

This paper is organized as follows. In Section 2 a summary of the state of the art of this extensive topic, focusing on TSP, is presented. Section 3 provides a mathematical overview of the TSP. Section 4 is concerned with the proposed methodology and its inclusion in the ROE method, the fuzzy logic classifier with adaptive heuristics, which is the core of this work, and the use of advanced metadata is explained. In Section 5, a detailed case of study through the use of plots is explained. Section 6 provides experimental results for ten TSP instances from the TSPLIB. Finally, Section 7 provides the conclusions and future work.

2. Related Work

Research has been conducted to solve the scalability issues associated with COPs in different ways. One of the most relevant trend that have proved to be very effective is to utilize modern hardware architectures to provide an additional degree of scalability through high levels of parallelism. With the increased access to such technologies, mainly by the introduction to mainstream multicore CPU computing and more recently GPU oriented computing, a significant amount of effort has been shifted to develop algorithms that properly use these new paradigms. Researchers have proven that revisiting classic algorithms and adding parallelism to them is a very effective strategy for solving COPs. Such is the work in [32] where the authors revisited the design and implementation of Branch-and-Bound algorithms for solving large COPs on GPU-enhanced multicore machines, and in [33] a high-performance GPU implementation of the 2-opt and 3-opt local search algorithm was presented. Other metaheuristics that were enhanced using GPU are Ant Colony Optimization [34, 35] and Genetic Algorithms [36, 37].

The systematic breakdown of COPs has also proven to be an effective strategy. Utilizing the divide and conquer strategies, the algorithms are capable of producing smaller problems that can be solved, and at a later stage they can be interconnected. The task at hand is providing proper dividing methodologies and the algorithms that will interconnect them. Works such as [38] describe the effectiveness of this approach and a discussion of multiple algorithms for undertaking the task is presented. In [29], a hybrid Neural Network, with local search via modified Lin-Kernighan algorithm that tries to solve a million city TSP, was introduced, although the work proves to be very successful in allowing for a higher degree of division of this huge TSP instance, thus enabling more scalability; the next inherent issue is presented in their results: *faster solution with lower quality*.

In the ROE method [31], a proposal to reduce the size of TSPs was presented; the main concept is to apply systematic reductions to the TSP instance in order to obtain a new instance that is representative of the original but with the added benefit that the TSP solving algorithms would require fewer resources to produce solutions. After it has been solved, the last stage of the ROE methodology is to reconstruct the original instance, giving us the final solution in the form of the expanded route. In [39] source code to implement the method is provided. Experimental results reported a 30% to 55% reduction in computational time, and the final solutions' length remained between 6.48% and 15.30% from the global optimum.

An exhaustive literature review in the Scopus database, ACM Digital Library, IEEE Xplore, Springer link, and Google Scholar was performed, and there was no methodology found that works similarly to what is proposed in this work, that is, a generic methodology that aims to remove elements from the given problem using a Fuzzy Inference System based on metadata and adaptive heuristics. To compare the ROE method using the proposed fuzzy operator, we have surveyed the current state of clustering algorithms, taking a special interest in clustering by fuzzy strategies. ROE can be denoted as a clustering algorithm because it performs the task of grouping via classification, in such a way that the elements share fuzzy characteristics. The idea of using fuzzy classification for clustering has been also considered by other researchers; however, they do not use advanced metadata based on adaptive heuristics, which provides a good degree of intelligence in the grouping process.

Clustering algorithms are usually unable to locate global optima because of an underlying data structure that is difficult to propose optimally. However, important contributions in clustering have been reported, and, similarly to this new approach, their aim is to improve performance. Fuzzy classification has been employed as a powerful tool in clustering algorithms; two important advantages are as follows:

(1) The use of linguistic variables to describe important characteristics and concepts about the elements in the problem.

(2) Classification based on knowledge represented through fuzzy rules extracted from the human expertise.

By surveying fuzzy logic applied to COPs, we found a clear tendency of using C-means fuzzy clustering techniques as the main form of grouping. From the year 2004 to the present, the consulted digital databases contain around 40 articles that fall within the search terms regarding fuzzy clustering and applications for TSP. The keywords chosen were TSP, COP, fuzzy clustering, fuzzy sets, and classification, among different variations of them. A selection of the surveyed articles was carried out. The following works are of special interest because they are the most fitting for comparison with the proposed methodology; however, they are still being substantially different because the grouping method and final purpose is dissimilar.

Focusing only on clustering based algorithms, the other methodologies usually group all the elements that fall within certain decision criteria and then they calculate the centroid of that group this being the first step in the optimization process. Differently, ROE using the proposed fuzzy operator classifies segments of the problem instance, and a centroid is calculated for those segments that are chosen to be removed; the segments may or may not be close to each other. This concept of classification and grouping of elements is one of the main contributions of this work.

Now, focusing on three-step methodologies for solving the TSP problem, the work presented in [40] proposes using a Genetic Algorithm (GA) based on unsupervised fuzzy clustering. In the first step, the cities in the problem are divided into several subtours, using a clustering algorithm. In the second step, each partition of cities is considered a smaller size TSP and it is solved using a GA, obtaining optimal subtours of the cities for each partition. In the third step, the subtours are connected in an appropriate way to obtain an optimized tour. Then a fourth step is required since the final tour needs to be improved by the GA. On the other hand, the ROE method, with the fuzzy operator, has three well-defined steps, and, differently from the above method, it does not depend on a particular method to achieve the optimization. Instead of working with independent subtours, which inevitably most of the times guide the optimization process to suboptimal solutions, the ROE works with the whole problem at the same time; in the first step the problem's original size is reduced to a convenient size using a fuzzy classifier based on advanced metadata and heuristics. In the second step, the reduced size problem can be optimized by any state-of-the-art method. In the third step, the optimized tour is methodology adapted to the original size problem, producing better solutions.

In [41] a hybrid metaheuristics for solving the TSP is presented; this is a fused method that combines a GA and an adaptive fuzzy greedy search operator. In [42], the individuals of the population of a GA are subject to be grouped using a fuzzy clustering approach; hence the reduction and the optimization stages are also fused. Contrary to these approaches, in ROE, the optimization process is not blended with the method, providing it with a big flexibility since it does not require modifying the inner workings of the selected optimization algorithm.

Fuzzy c-means clustering is the most popular technique of all. Twenty of the surveyed articles use this grouping method. In [43], a hybrid evolutionary fuzzy learning algorithm that automatically determines the near optimal traveling path in large-scale traveling salesman problems is presented. It described the steps required to identify, group, solve, and connect clusters of cities. The key difference in the aforementioned work is that the cities of the TSP are never removed; thus the problem size remains the same. Other works that apply fuzzy clustering in a similar matter can be in [42, 44, 45].

Hybrid approaches besides Evolutionary Algorithms exist. Although these types of approaches do not share as many characteristics as the ones discussed previously, it is important to note that fuzzy clustering can be applied in different ways. In [46], a fuzzy self-organizing map for an artificial Neural Network is presented.

Other approaches come in the form of fixing edges [47] and they provide significant improvements to the solving of the TSP instances as well. However, these approaches require additional local optimization as a preprocess.

3. Mathematical Formulation of the TSP

The TSP is a highly studied problem for testing and benchmarking algorithms. Mathematically, the TSP can be expressed as an edge-weighted, directed graph $G = (V, E)$, where V is the set of $n = |V|$ vertices (cities), $E \subseteq V \times V$, the set of (directed) edges (path between two cities), and a *distance function* $d : E \longrightarrow \mathbb{R}^+$ assigning each edge $e \in E$ to a distance $d(e)$ [10, 48].

A tour π is *Hamiltonian cycle* that consists of a single unique permutation of all $N+1$ vertices in V where the first and last elements of the permutation are the same, and W is the set of all tours π. By interpreting $\pi(j)$ as the city visited after city j, for $j = 1, \ldots, N$, the cost of a tour can be written as

$$c : \pi \longmapsto \sum_{j=1}^{N} d_{j\pi(j)}. \tag{1}$$

Thus, the TSP is defined as a set $v_1, v_2, \ldots, v_N$ cities where, for each pair (v_i, v_j) of distinct cities, there exists a distance $d(v_i, v_j)$; the goal is to find the ordering π of cities that minimize the quantity:

$$\sum_{i=1}^{N-1} d\left(v_{\pi(i)}, v_{\pi(i+1)}\right) + d\left(v_{\pi(N)}, v_{\pi(1)}\right). \tag{2}$$

This quantity is referred to as the tour length since it is the length of the tour that a salesman would make when visiting the cities in the order specified by the permutation, returning at the end of the initial city. For two-dimensional problems, the vertices are points $P_i = (v_{x_i}, v_{y_j})$ in the plane, and d_{ij} is the Euclidian distance [31] given by

$$\left\|\vec{d}\right\|_2 = d_{ij} = \sqrt{\left(v_{x_i} - v_{x_j}\right)^2 + \left(v_{y_i} - v_{y_i}\right)^2}. \tag{3}$$

Finally, solving the TSP consists in producing a tour which comprises a closed route where there are no repeated nodes, except for the first and last node as they are the same and the length of the tour is minimal. This task is not easy to satisfy, specifically as the amount of cities increases, in which obtaining a tour with minimal cost for a large number of cities is classified as an NP-hard problem; these kinds of problems are considered unsolvable by polynomial algorithms. For this very reason, there is a special interest in developing and using methodologies that limit the search space and then utilize a metaheuristic.

As it was explained, many different variations of the TSP exist in literature [6]; however, three of them stand out as the most widely studied and used: the symmetric Traveling Salesman Problem (sTSP) where the distances must always satisfy $d(v_i, v_j) = d(v_j, v_i)$, the asymmetric Traveling Salesman Problem (aTSP) in which at least one distance is $d(v_i, v_j) \neq d(v_j, v_i)$, for both cases $1 \leq i, j \leq N$, and finally, the Multitraveling Salesman Problem (mTSP) where m salesmen are deployed from the starting city with the objective of finding a tour for each salesman and each city is visited once by only one salesman, and the cost of the tour is minimized.

Each variation of the TSP requires a different mathematical formulation and consideration. Being the sTSP the most cited work [7], with specialized WEB pages such as the TSPLIB where the most recent advances and optimally obtained results for different instances are summarized, we focused this proposal on this variation of TSP, with the aim of having many examples to achieve comparisons with the latest results obtained by the state-of-the-art algorithms. From this point forward, when speaking of the TSP, we are referring to the sTSP.

4. The VFS with Adaptive Heuristics in the ROE Method

The ROE method [31] produces a temporally smaller-sized COP, representing the original problem. The purpose is helping COP solving algorithms to perform faster providing high-quality solutions. For huge sized problems, where no other existing methods can provide a solution, the ROE method allows doing it whether the problem can be effectively reduced to a manageable size for optimizing algorithms. The first reduction operators of ROE were inspired by Artificial Immune Systems (AIS) [49–51], specifically in artificial vaccination [52]; for this reason, the reduction operators in this method are called vaccines. Two main operators for selection were reported in [31], and they are Vaccination by Random Selector (VRS) and Vaccination by Elitist Selector (VES). In this paper, the operator Vaccination by Fuzzy Selector with adaptive heuristics (VFSah) is proposed.

The ROE method consists of three steps, and they are as follows:

(i) *Reduction*. In this first step, the TSP instance is analyzed to select the nodes to be removed; it is the most important and researched step. It has many considerations to guarantee that the reductions made to the instance actually form part of the optimal solution, and they remain representing the original TSP instance. The reduction step has most significance on the outcome of the whole methodology. A selection or decision criteria to determine which candidates are chosen to be removed is required. In our program, a .tsp file was created to save the new reduced TSP instance. This stage requires achieving the next following actions:

(1) *Loading the TSP Instance and Representation*. The instance data is loaded into memory. Each city is represented in (x, y) coordinates. An example of this representation is shown in Figure 1(a).

(2) *Mesh Generation*. The mesh consists of a series of connections (edges) between nodes. Obtaining multiple paths from one node to the rest is desirable to promote options for the fuzzy classifier; however, a big amount of nodes can slow down algorithms due to memory and computing limitations. The simplest mesh can be created by calculating the distance matrix for the particular TSP instance; nevertheless this is not the best choice because we found the previously discussed problem. A better alternative is to create a distance matrix but only to the nearest nodes, and in this way the size is reduced

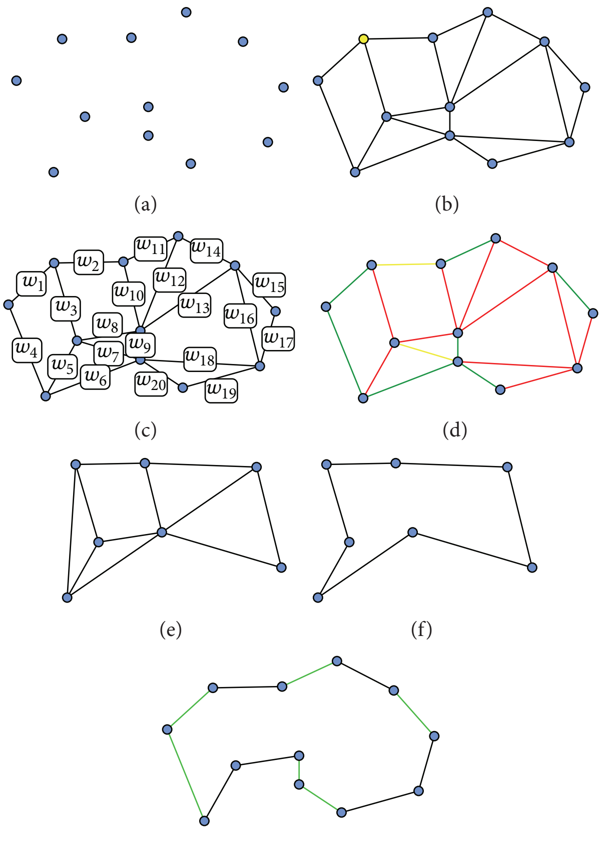


Figure 1: In this example, the operator Vaccine by Fuzzy Selector with adaptive heuristics (VFSah) was applied to reduce the TSP size, using the ROE method.

considerably. An example of a mesh is illustrated in Figure 1(b).

(3) *Advanced Metadata Generation*. This consists in calculating w_i for each edge; this value contains both the Proximity and Secludedness values that are part of the mesh. Figure 1(c) shows a representation of each edge with its assigned value.

(4) *Using the Operator VFSah*. The VFSah is used to classify each of the edges in the mesh as vaccines or not vaccine; that is, the idea is to identify which edges can be removed from the original TSP. This classification is represented using colors in Figure 1(d); in green color are the chosen edges to be removed (vaccines), in yellow are those that cannot be classified as vaccines, and in red are those that will remain as part of the instance. The resulting TSP reduced instance is shown in Figure 1(e).

(ii) *Optimize*. The new reduced instance is ready to be solved by any state-of-the-art specialized optimizing algorithm. This is one of the most important advantages of the ROE methodology, since we do not need to achieve any change to the optimizing algorithm. The resulting tour is going to be used in the next step. Figure 1(f) shows a reduced TSP instance that has been solved by any TSP solving algorithm as usual. It is expected that the TSP solving algorithm generates an output file in the form of a ".tour file." In this paper, we will not discuss TSP solving algorithms.

(iii) *Expand*. The expand step takes the solution from the previous step and reconstructs the tour by returning the removed nodes from the reduction step. Local optimization can be used in order to create a better reconstruction scenario when connecting the reduced nodes and paths. This step is critical because it provides the correct solution to the TSP instance. Figure 1(g) illustrates the last step; it consists in the expansion of the previously generated tour by including the original removed sections. This step requires the solution generated by the TSP solving algorithm, the original TSP instance data, and the reduced TSP instance; the final tour will be in terms of the original TSP instance, thus providing the real tour length.

Because the reduction step is the most critical one and the focus of this work, we will describe in detail its inner workings. First, we must consider the quality of the starting mesh to define possible candidates to be reduced. As most heuristics, the difference between a "good" and "bad" starting point changes the outcome significantly. A better construction of a starting mesh will result in a more effective selection process. The mesh not only requires containing good candidates, but it also needs to avoid large candidates to produce fast results.

The VFSah reduction operator uses the metadata contained in the edges of the mesh, as it was indicated in Figure 1(c), where the metadata is formed with the Proximity and Secludeness values.

4.1. Metadata

4.1. Metadata. Metadata provides data about data. In the particular case of the TSP, data comes in the form of nodes with (x, y) coordinates that represent the position of cities on a map. The most basic form of metadata pertaining the TSP is the distance between nodes, also known as Euclidean distance; it can be calculated using (3).

The Distance Matrix (DM) is used to organize the distances between nodes (4). Most of the TSP solving algorithms use the DM as a starting point for generating solutions. Being one of the most basic forms of metadata, the DM provides the stepping stone for even more robust metadata generation and complex searching techniques. Consider

$$\mathrm{DM} = \begin{pmatrix} d_{11} & d_{12} & d_{13} & \cdots & d_{1n} \\ d_{21} & d_{22} & d_{23} & \cdots & d_{2n} \\ d_{31} & d_{32} & d_{33} & \cdots & d_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ d_{m1} & d_{m2} & d_{m3} & d_{m4} & d_{mn} \end{pmatrix}. \quad (4)$$

```
Require: $k \geq 1$
DM = GenerateDistanceMatrix()
for $i = 0$; $i <$ DM.Lenght; $i++$; do
    Select all distances values for $i$ row
    Sort all values ascending, keep track of each index
    Take $k$ first elements from the sorted indexes
    if Elements are not in NNNDM then
        Add the pair [row index, column index] to NNNDM
    else
        Go back to taking $k$ first elements from the sorted indexes and repeat the check
    end if
end for
return NNNDM
```

Algorithm 1: Pseudocode for implementing Nonrepeat Nearest Neighbor Distance Matrix. It is easier to save the indexes that refer to the original Distance Matrix.

ROE employs the DM between cities with the important modification of reducing the matrix to nearest neighbors on the premise that TSP connections rarely will join far cities. This modified DM is known as Nearest Neighbor Distance Matrix (NNDM) where the amount of nearest neighbors k needs to be specified. Additionally, calculating the NNDM and not allowing repeated nearest neighbors between nodes provides more paths without redundancy. This modified NNMD is called Nonrepeat Nearest Neighbor Distance Matrix (NNNDM) and is the preferred basic metadata calculation used by the proposed methodology. The pseudocode for implementing NNNDM can be found in Algorithm 1.

4.2. Fuzzy Classifier for Problem Reduction.

4.2. Fuzzy Classifier for Problem Reduction. People can solve reasonably sized TSP by using their own reasoning. A person may realize and say "this part of the route is the shortest" to describe a specific part of the instance. Such opinion is just an estimation of the distance, and it might be far from a precise number; however, it can be precise enough for certain applications and in some cases it can be even the optimal decision. Such powerful expressions that help us to solve problems by observation and applying common sense are fuzzy expressions, and they are based on *linguistic variables*. Fuzzy logic uses them to formulate rules and emulate human reasoning. Therefore, incorporating techniques based on the theory of fuzzy sets over more conventional approaches to solve complex problems lies in their capability of integrating a priori knowledge and human expertise about the problem.

In the particular case of the TSP, the problem is analyzed by making connections between the nodes and assigning such linguistic connotation, such as Far, Close, and Secluded, to shape our problem into a possible solution.

The Mamdani fuzzy classifier that evaluates the edges for possible problem reduction size is shown in Figure 2. The linguistic variables are variables whose values are words or sentences in a natural or artificial language; they allow us to create a link between conceptual thought and numbers. Here, two main linguistic variables are proposed for the inputs. These must be obtained from the problem's metadata and have been designed to be both global and locally impactful in the decision-making process.

The FIS inputs and output linguistic variables are Proximity, Secludedness, and IsVaccine, respectively.

4.2.1. Linguistic Variable Proximity. The linguistic variable Proximity is a normalized fuzzy measure that indicates how near are two cities. Figure 3 shows its linguistic terms: VeryFar, Far, Average, Close, and VeryClose; the corresponding membership functions (MFs) of these terms are evenly distributed between 0 and 1.

For applying the linguistic variable Proximity to a TSP problem, it is necessary to calculate the normalized distance between the cities of interest. This normalization is calculated using the following by considering the distance from the departure city to its nearest and its farthest city, $E_{\min}$ and $E_{\max}$, respectively:

$$\operatorname{Norm}(e_i) = 1 - \frac{e_i - E_{\min}}{E_{\max} - E_{\min}}, \quad E_{\min} \neq E_{\max}; \tag{5}$$

that is, e_i corresponds to each edge of the mesh, $E_{\min}$ is the minimum distance between nodes (cities) of the mesh, and $E_{\max}$ is the maximum distance between the nodes of the mesh.

4.2.2. Linguistic Variable Secludedness. The linguistic variable Secludeness is also a normalized fuzzy measure that indicates how remote is an edge; so Secludeness weights how separated a particular edge is when compared to its neighbors. This linguistic variable is locally impactful since it only considers edges that are close to it.

The Secleudeness value is the result of using an adaptive heuristic, and it is achieved with a rewarding iterative process that depends on the number of the considered nearest edges; only the edges with a bigger value than the segment that we are analysing will contribute with the reward which is calculated using (6), being different for each edge in the same problem. The higher Secludedness values represent the more isolated edge.

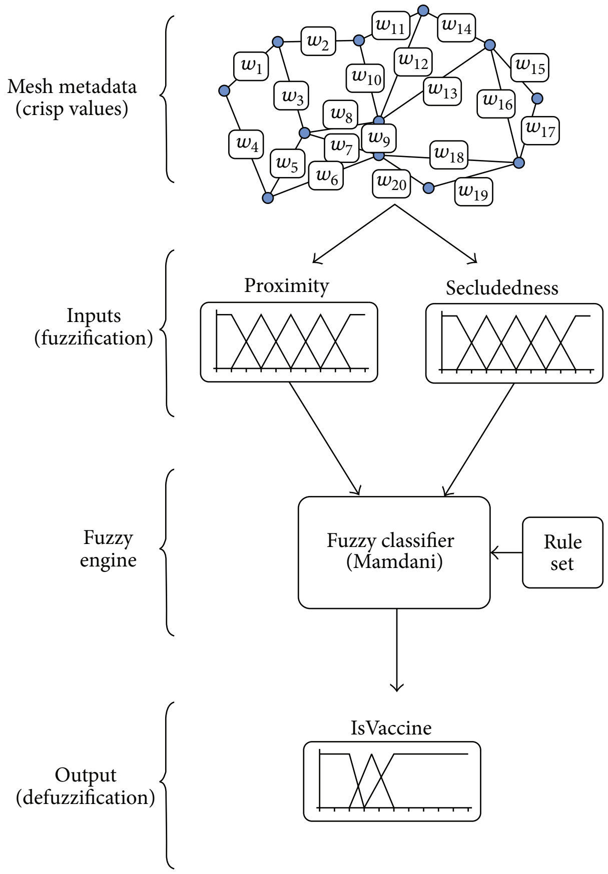


Figure 2: Fuzzy logic classifier.

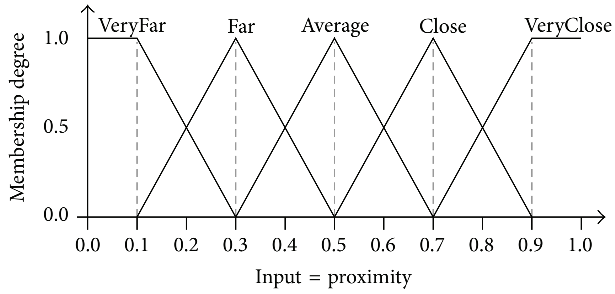


Figure 3: Membership functions of the input linguistic variable Proximity.

To explain this concept Figure 4 is used. We start with a Secludeness value equal to zero. Considering that, in this case, the distance between cities A and B ($\overline{AB}$) is 1, $\overline{AE} = 4$, and $\overline{BD} = 3$, and so forth. To obtain the Secludeness value of the edge $\overline{AB}$, it is necessary to compute the reward, in this case, 0.25, since there are four nearest neighbor edges, and every time the length $\overline{AB}$ is shorter than one of the edges an increment of 0.25 on the Secludedness value is achieved. Secludedness is also a normalized value between 0 and 1 and the increments can be determined by

$$\text{Increment} = \frac{1}{\text{Number Of Nearest Edges}}. \qquad (6)$$

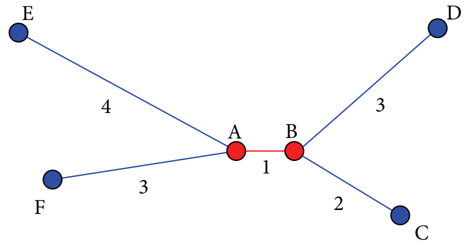


Figure 4: Example of the calculation of the Secludedness value for edge AB. The value is equal to 1 because AB's length is shorter than any other neighboring edges.

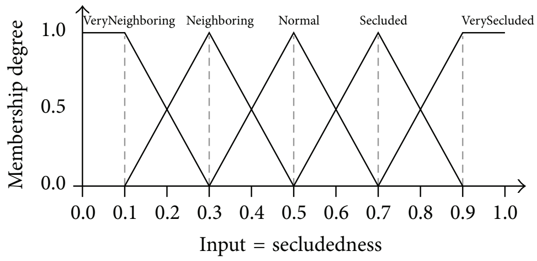


Figure 5: Membership functions of the input linguistic variable Secludedness.

Figure 4 shows an example of how to calculate the Secludedness value of the edge AB (red). The distance AB is equal to 1; the increment is calculated with (6) as $1/4 = 0.25$ because we have four nearest edges, two from A and two from B (blue). The Secludedness value is then calculated resulting in 1 because the edges AE, AF, BC, and BD are longer than AB, giving four increments of 0.25.

Figure 5 describes the terms of the Secludedness linguistic variable, and they are VeryNeighboring, Neighboring, Normal, Secluded, and VerySecluded.

4.2.3. Output Linguistic Variable IsVaccine. The linguistic variable IsVaccine provides the fuzzy classification of a particular edge after being processed by the FIS illustrated in Figure 2. The MFs for IsVaccine are NotVaccine, CouldBeVaccine, and Vaccine; they are shown in Figure 6.

4.2.4. Rule Base. The rule base is shown in Table 1; at the present, the only cities that can be reduced are those classified as "V"; the other fuzzy outputs "CV" and "NV" could not be considered as vaccines.

5. Case of Study

To illustrate the whole methodology, a TSP instance with 343 cities that is called pma343 in the TSPLIB was selected. This particular size of TSP was chosen because it allows a convenient graphic representation; a higher node count would lead to confusing images.

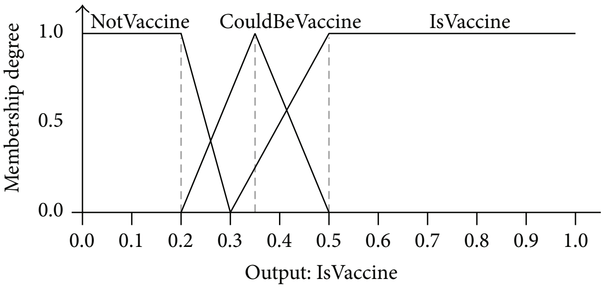


Figure 6: Membership functions of the output linguistic variable IsVaccine.

Table 1: FIS rules.

Secludedness	Proximity				
	VeryFar	Far	Average	Close	VeryClose
VeryNeighboring	NV	CV	NV	NV	CV
Neighboring	CV	V	V	NV	CV
Normal	NV	NV	NV	NV	V
Secluded	NV	NV	V	CV	V
VerySecluded	V	NV	CV	V	V

NV: NotVaccine, CV: CouldBeVaccine, and V: Vaccine. The input variables are Proximity and Secludedness. The output variable is IsVaccine.

Figure 7 shows the pma343, and it is represented as a series of (x, y) coordinates on the plane. This figure corresponds to the first step of the proposed method.

Figure 8 illustrates the creation of the starting mesh. The Nonrepeat Nearest Neighbor Mesh strategy was implemented. The first node of the mesh was the first selected node of the TSP instance, and two neighbors per node were calculated. Experiments show that this provides adequate coverage and enough optional paths for the selection strategy to choose from. Note that this strategy does not guarantee that all the mesh will be interconnected, and, in Figure 8, four mesh groups can be identified. Each connection between nodes has its own metadata values corresponding to Proximity and Secludedness (not illustrated for clarity in the graph).

After the metadata creation and assignment of the crisp input values for the linguistic variables based on the established logical rules, the FIS calculates which one of the connections will be chosen for reduction. Figure 9 shows those connections that were classified as reduction. Note, in the section enclosed in black, the edges will be removed and replaced by a new node representative of the removed nodes connected by the edges.

In Figure 10, the reduced TSP instance is shown. The TSP solving algorithms will work with this new reduced instance. It is a substitute instance of the original but with fewer nodes. The new instance will allow faster solving times and, in some cases, the TSP solving algorithm will be capable of finding a solution which previously was not possible due to hardware or algorithm limitations when handling large instances. Note, in the section enclosed in black, that the nodes which were connected by the edges have been removed and replaced by two new nodes corresponding to the center of each removed edge.

Figure 11 shows the optimal route. A comparison between this route and the route with the selected reductions is presented in Figure 12.

6. Experimental Results

With the aim of showing that the VFSah can produce effective reductions that are part of the global optimal route, we have chosen ten different TSP instances from the TSPLIB; they vary in size from 343 to 3056 cities, and their optimal route is known, which is important in order to evaluate the method quantitatively.

A summary of results for ten instances is shown in Table 2. At a first glance, a clear tendency is presented. The methodology provides significant reduction of the problem instances, and a high percentage of these belong to the global optimum. The table provides information such as mesh size, number of reductions made, and the percentage of reductions that are part of the reported global optimum, according to the TSPLIB. With particular emphasis on the percentage of optimal reductions, a range from 71.87% to 86.67% with no notable dependency on the instance size was observed. The smallest instance of 343 cities had 84.92% of optimal reductions; meanwhile the largest instance, with 3,056 cities, resulted in 73.22% of optimal reductions; however, in the larger instance there were 2,254 more reduced cities than in the smaller instance. The last columns indicate the percentage of reduction of the original problem size, which is very important, since the TSP requires exponential time to be solved; therefore, reducing the size by at least ≈29% is quite good since the needed time to solve the TSP will be impacted drastically. Note, in the table, that there exists a tendency to grow up this last factor; for example, for the 3,056 cities instance, the problem was reduced in a 40.12%.

6.1. Comparison among VRS, VES, and VFSah. To show the advantages of VFSah reduction operator, against the existing VRS and the VES operators for the ROE method, Table 3 shows a qualitative comparison of the methods, and Table 4 presents a quantitative comparison.

Table 3 shows that the only operator that is able to achieve reductions based on rules, applying the expertise knowledge, as well as reductions based on advanced metadata is the VFSah operator. This operator also provides the best amount of positive reductions. The other two existing operators, the VRS and the VES, are also good operators; however, these operators require that the user provides a numeric input that forces achieving a specific number of reductions in the problem instance, which sometimes may result in obtaining poor quality reductions. Comparing these three operators, the VFSah has the best qualities; the only issue is that it is little more complex to implement since it requires a fuzzy classifier with adaptive heuristics and implementing advanced metadada. However, their computational complexity remains being linear.

Table 2: Experimental results for 10 TSP instances from TSPLIB treated with Vaccination by Fuzzy Selector with adaptive heuristics (VFSah). Significant degree of problem reduction is achieved, and a high percentage of optimal is reached with the proposed methodology.

TSP instance	Size	Mesh size	Reductions	Optimal reductions	% of optimal reductions	Reduced TSP size	% of reduction
pma343	343	686	199	169	84.92%	244	28.86%
pbk411	411	822	284	220	77.46%	269	34.55%
pbm436	436	872	306	343	77.45%	283	35.09%
xql662	664	1324	518	390	75.29%	405	39.01%
rbu737	737	1474	375	325	86.67%	550	25.37%
lim963	963	1926	638	544	85.27%	644	33.13%
dka1376	1376	2752	911	735	80.68%	921	33.07%
rby1599	1599	3198	1321	961	72.75%	939	41.28%
dcb2086	2086	4172	1852	1331	71.87%	1160	44.39%
pia3056	3056	6112	2453	1796	73.22%	1830	40.12%

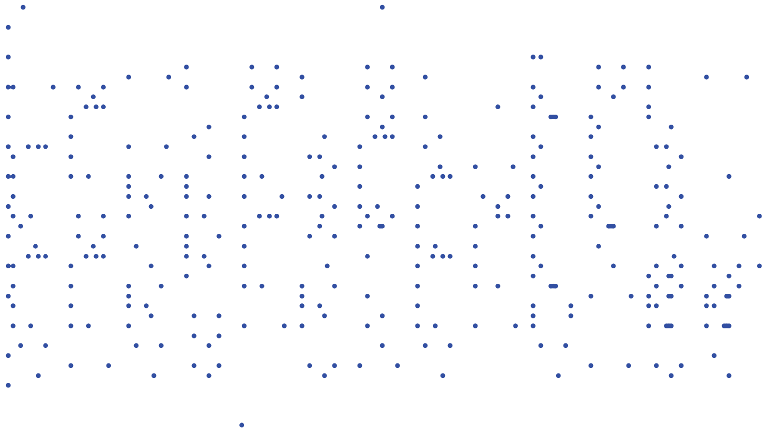


Figure 7: Problem Analysis Step 1. Loading and representing the TSP instance.

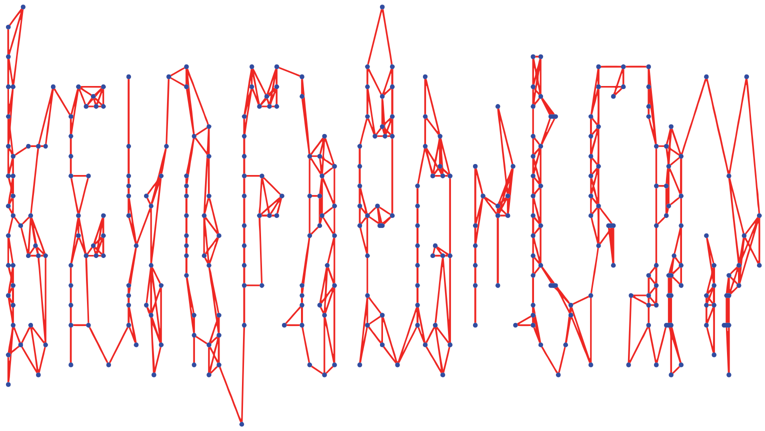


Figure 8: Problem Analysis Step 2. Mesh generation by Nonrepeat Nearest Neighbor Distance Matrix. Note that particularly dense zones have more connections than the unions between those zones and some zones are isolated due to the nature of the nearest neighbor strategy.

Table 4 shows in the first column ten instances of the TSP; columns two to four show the amount of reductions achieved by the VRS, VES, and VFSah reduction operators. Column five shows the percentage of improvements of the VFSah with respect to the VRS and, similarly, column six, the VFSah versus the VES.

7. Conclusion

Combinatorial optimization (CO) is an important mathematics branch that has many applications in artificial intelligence, machine learning, and other science and engineering fields. In general, the computational cost of finding the global

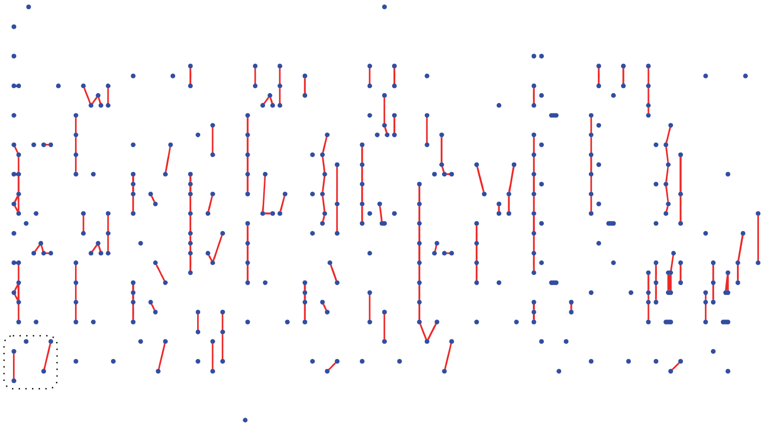


Figure 9: Problem Analysis Step 3. The selected vaccines from the mesh are shown.

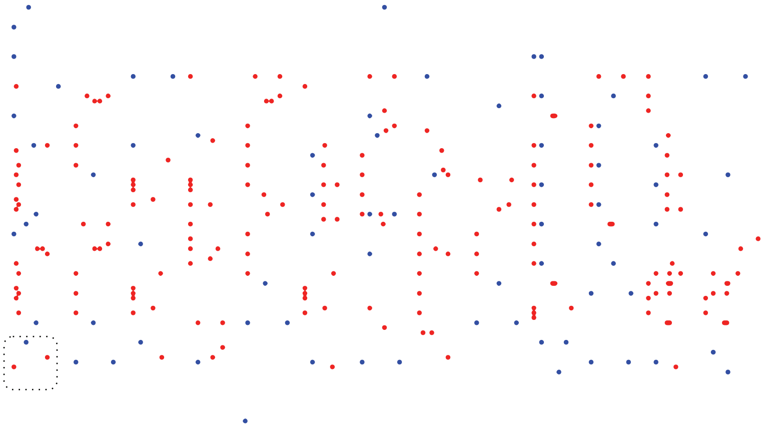


Figure 10: Problem Analysis Step 4. The reduced TSP instance is shown. In blue color are the nodes that remained unchanged; in red color are the nodes representing removed nodes.

Table 3: Comparison between VFSah, VRS, and VES.

	VFSah	VRS	VES
Adaptive reductions based on advanced metadata	√		
Forced amount of reduction		√	√
Best amount of positive reductions	Best	Good	Better
Experiments consistency	√	√	√
Implementation difficulty	High	Low	Medium
Computational complexity	Linear	Linear	Linear
Reductions based on rules	√		
Usage of expert knowledge	√		

optima of a COP grows up as the number of nodes increases. The TSP is a classical problem in CO; in fact, it can be considered an intractable problem when the number of nodes increases because there is no efficient algorithm that can solve it efficiently. Since the original formulation of the TSP, hundreds of proposals to solve it emerged; however, only few methods demonstrated to be efficient doing this task, and, broadly speaking, they are based on mathematical foundations reinforced by technological advances, but the problem remains when the problem size is increased. Other mathematical tendencies search for reducing the problem size, and at the present there are efficient proposals, with their own pros and drawbacks, but the only methodology that treats COP in a general way, leaving aside the optimization methods because it can use any of the existing methods, is the ROE method.

The ROE method in its original formulation was proposed with two vaccines (operators) to reduce temporally the size of a TSP. In this work, a new reduction operator for the ROE, named VFSah, which is based on a fuzzy classifier reinforced with an adaptive heuristic, was introduced.

The experiments show that the VFSah outperforms the existing ROE operators. The ROE method is the only existing proposal that treats COPs in a general way, so that it does not make the optimization step be the fundamental part, because there are many successful proposals to achieve this task, but they fail for large instances of COPs; therefore, the essential task of the ROE method is to reduce the problem size by providing flourishing reductions to a problem instance, after which an appropriate optimization algorithm produces a solution, where in the last step the solution is reconstructed, generating the optimal result. So, the objective

Table 4: Summary of values to compare the three different selector strategies: Vaccination by Random (VRS), Elitist (VES), and Fuzzy Selector with adaptive heuristics (VFSah) for percentage of optimal reductions. The VFSah strategy provides higher percentage of optimal reductions for all tested TSP instances. The last two columns compare VRS and VES to VFSah, respectively.

TSP instance	VRS	VES	VFSah	VFSah versus VRS	VFSah versus VES
pma343	43.92%	64.32%	84.92%	41.00%	20.60%
pbk411	43.31%	62.68%	77.46%	34.15%	14.78%
pbm436	44.84%	59.48%	77.45%	32.61%	17.97%
xql662	46.06%	62.36%	75.29%	29.23%	12.93%
rbu737	45.17%	70.67%	86.67%	41.49%	16.00%
lim963	44.39%	66.93%	85.27%	40.88%	18.34%
dka1376	44.72%	64.65%	80.68%	35.96%	16.03%
rby1599	43.73%	53.82%	72.75%	29.02%	18.93%
dcb2086	45.31%	58.59%	71.87%	26.56%	13.28%
pia3056	44.08%	58.62%	73.22%	29.14%	14.60%

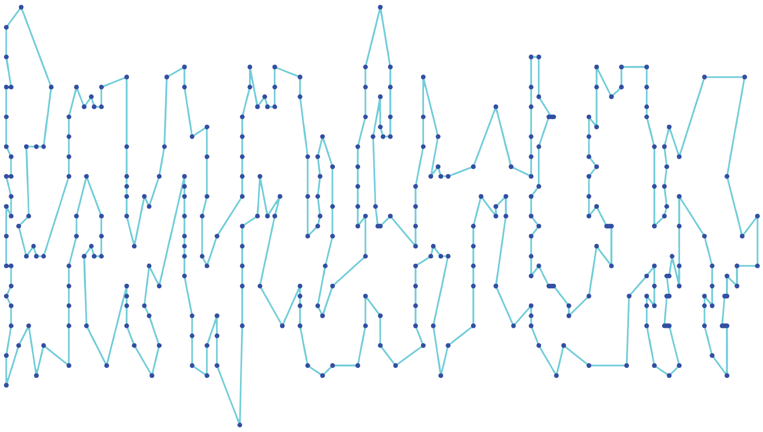


Figure 11: Problem Analysis Step 5. Optimal route for optimum for the pma343 TSP instance is shown.

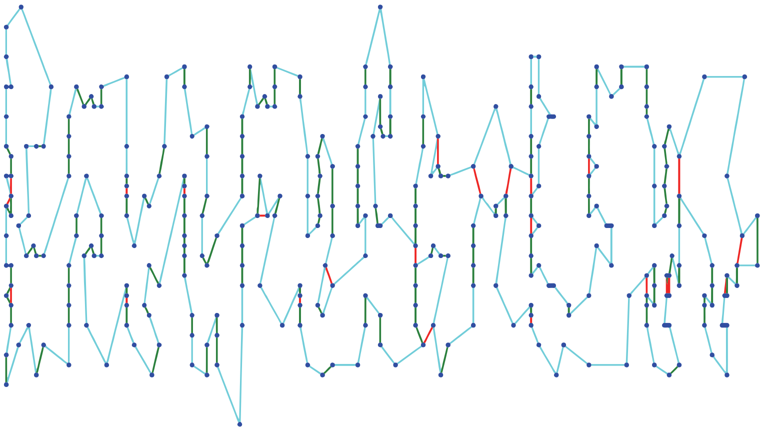


Figure 12: Problem Analysis Step 6. Comparing the generated vaccines with the optimal route. The optimal route is shown in cyan color. The vaccines that are part of route are shown in green color. False-positives vaccines are shown in red.

is to provide *faster high-quality solutions* by empowering the ROE providing higher-quality solutions than before. As a direct consequence of this improvement, the amount of huge problems with unknown solutions that can be treated using known optimization algorithms can increase.

At the present, the rule matrix of the fuzzy classifier and the fuzzy terms of the linguistic variables were set using the expertise knowledge. To improve these results, as future work, for the VFSah, it is possible to obtain a better distribution of the linguistic terms as well as an optimized rule base matrix to provide better reductions; hence the speed and quality of solutions will increase.

7.1. Methodology Scope and Limitations. It is well-known that there are a large number of classic difficult computational problems from different fields such as graph theory, mathematical programming, and combinatorics that can be reduced among them under certain considerations; therefore, it is expected that finding a solution for one of the aforementioned problems will serve the others [53]. Focusing on the TSP, state-of-the-art clustering algorithms are usually unable to locate the global optima because of an underlying data structure that is difficult and most of times impossible to propose optimally, providing low quality fast solutions.

On the other hand, the proposal presented in this work, the VFSah operator, provides a broader scope than the existing ones, mainly because it enhances the ROE method with a new operator that improves results supplying better solutions in similar times; the ROE proposal is based on reducing the problem size of a COP, instead of working on a solving algorithm for a specific problem as is the common factor in the existing methods. Therefore it is expected that, using a similar approach, a larger number of classic unsolved problems of routing, assignment, packing, and others can be solved more efficiently. Clustering algorithms also work by grouping the problem instance; however, they do not make reductions in an intelligent way as it is the case of the VFSah operator.

The ROE methodology is recent, and, at the present, the VFSah operator offers the best results; however, new operators that might outperform it can be developed. There are no true limitations to apply the ROE methodology and the VFSah operator to different sizes of TSP problems, even those with unknown solution, but its performance will depend on the hardware computer limitations. The world of COPs is very complex and extensive, so this methodology has not been tested yet on other problems different from the TSP; but based on the reducibility principle it is expected that this methodology will serve to solve other problems of the same class.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The authors thank Instituto Politécnico Nacional (IPN), the Comisión de Fomento y Apoyo Académico del IPN (COFAA), and the Mexican National Council of Science and Technology (CONACYT) for supporting their research activities.

References

[1] V. V. Vazirani, *Approximation Algorithms*, Springer, New York, NY, USA, 2010.

[2] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman, 1979.

[3] T. Crilly, "Paul Urysohn and Karl Menger, papers on dimension theory (1923–1926)," in *Landmark Writings in Western Mathematics 1640–1940*, I. Grattan-Guinness, R. Cooke, L. Corry, P. Crépel, and N. Guicciardini, Eds., chapter 66, pp. 844–855, Elsevier Science, Amsterdam, The Netherlands, 2005.

[4] M. Fischetti, A. Lodi, and P. Toth, "Exact methods for the asymmetric traveling salesman problem," in *The Traveling Salesman Problem and Its Variations*, G. Gutin, A. Punnen, D.-Z. Du, and P. M. Pardalos, Eds., vol. 12 of *Combinatorial Optimization*, pp. 169–205, Springer, New York, NY, USA, 2004.

[5] G. Laporte, "The traveling salesman problem: an overview of exact and approximate algorithms," *European Journal of Operational Research*, vol. 59, no. 2, pp. 231–247, 1992.

[6] M. Rajesh, S. Surya, and M. M. Lal, *Traveling Salesman Problem: An Overview of Applications, Formulations, and Solution Approaches*, InTech, Rijeka, Croatia, 2010.

[7] G. B. Dantzig, D. R. Fulkerson, and S. M. Johnson, "Solution of a large-scale traveling salesman problem," *Operational Research*, vol. 2, pp. 393–410, 1954.

[8] D. Applegate, R. E. Bixby, V. Chvátal, and W. J. Cook, "Implementing the Dantzig-Fulkerson-Johnson algorithm for large traveling salesman problems," *Mathematical Programming*, vol. 97, no. 1-2, pp. 91–153, 2003.

[9] C. William, *Concorde Home*, Office of Naval Research and National Science Foundation, 2011, http://www.tsp.gatech.edu/concorde/index.html.

[10] D. L. Applegate, R. E. Bixby, V. Chvatal, and W. J. Cook, *The Traveling Salesman Problem: A Computational Study*, Princeton Series in Applied Mathematics, Princeton University Press, Princeton, NJ, USA, 2007.

[11] W. L. Eastman, *Linear programming with pattern constraints [Ph.D. thesis]*, Department of Economics, Harvard University, Cambridge, Mass, USA, 1958.

[12] J. D. C. Little, K. G. Murty, D. W. Sweeney, and C. Karel, "An algorithm for the traveling salesman problem," *Operations Research*, vol. 11, no. 6, pp. 972–989, 1963.

[13] S. Arora, "Polynomial time approximation schemes for euclidean traveling salesman and other geometric problems," *Journal of the ACM*, vol. 45, no. 5, pp. 753–782, 1998.

[14] S. Lin and B. W. Kernighan, "An effective heuristic algorithm for the traveling-salesman problem," *Operations Research*, vol. 21, pp. 498–516, 1973.

[15] D. Karapetyan and G. Gutin, "Lin-Kernighan heuristic adaptations for the generalized traveling salesman problem," *European Journal of Operational Research*, vol. 208, no. 3, pp. 221–232, 2011.

[16] C.-K. Ting, S.-T. Li, and C. Lee, "On the harmonious mating strategy through tabu search," *Information Sciences*, vol. 156, no. 3-4, pp. 189–214, 2003.

[17] O. Pedro, R. Saldanha, and R. Camargo, "A tabu search approach for the prize collecting traveling salesman problem," *Electronic Notes in Discrete Mathematics*, vol. 41, pp. 261–268, 2013.

[18] A. Uğur, "Path planning on a cuboid using genetic algorithms," *Information Sciences*, vol. 178, no. 16, pp. 3275–3287, 2008.

[19] H.-K. Tsai, J.-M. Yang, Y.-F. Tsai, and C.-Y. Kao, "An evolutionary algorithm for large traveling salesman problems," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 34, no. 4, pp. 1718–1729, 2004.

[20] Y. Jin, "Surrogate-assisted evolutionary computation: recent advances and future challenges," *Swarm and Evolutionary Computation*, vol. 1, no. 2, pp. 61–70, 2011.

[21] M. Dorigo and L. M. Gambardella, "Ant colonies for the travelling salesman problem," *BioSystems*, vol. 43, no. 2, pp. 73–81, 1997.

[22] Y. Zhou, "Runtime analysis of an ant colony optimization algorithm for TSP instances," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 5, pp. 1083–1092, 2009.

[23] K. Jun-man and Z. Yi, "Application of an improved ant colony optimization on generalized traveling salesman problem," *Energy Procedia A*, vol. 17, pp. 319–325, 2012, Proceedings of the 2012 International Conference on Future Electrical Power and Energy System.

[24] Y. Marinakis, M. Marinaki, and G. Dounias, "Honey bees mating optimization algorithm for the euclidean traveling salesman problem," *Information Sciences*, vol. 181, no. 20, pp. 4684–4698, 2011.

[25] D. S. Johnson and L. A. McGeoch, *The Traveling Salesman Problem: A Case Study*, Princeton University Press, Princeton, NJ, USA, 2003.

[26] Y. Wen, H. Xu, and J. Yang, "A heuristic-based hybrid genetic-variable neighborhood search algorithm for task scheduling in heterogeneous multiprocessor system," *Information Sciences*, vol. 181, no. 3, pp. 567–581, 2011.

[27] U. Benlic and J.-K. Hao, "A multilevel memetic approach for improving graph k-partitions," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 5, pp. 624–642, 2011.

[28] F. Peter, J. Botzheim, and L. T. Koczy, "Eugenic bacterial memetic algorithm for fuzzy road transport traveling salesman problem," *International Journal of Innovative Computing*, vol. 7, no. 5, pp. 2775–2798, 2011.

[29] S. A. Mulder and D. C. Wunsch II, "Million city traveling salesman problem solution by divide and conquer clustering with adaptive resonance neural networks," *Neural Networks*, vol. 16, no. 5-6, pp. 827–832, 2003.

[30] Y.-T. Wang, J.-Q. Li, K.-Z. Gao, and Q.-K. Pan, "Memetic algorithm based on improved Inver-over operator and Lin-Kernighan local search for the Euclidean traveling salesman problem," *Computers & Mathematics with Applications*, vol. 62, no. 7, pp. 2743–2754, 2011, Computers & Mathematics in Natural Computation and Knowledge Discovery.

[31] O. Montiel, F. J. Díaz-Delgadillo, and R. Sepúlveda, "Combinatorial complexity problem reduction by the use of artificial vaccines," *Expert Systems with Applications*, vol. 40, no. 5, pp. 1871–1879, 2013.

[32] I. Chakroun, N. Melab, M. Mezmaz, and D. Tuyttens, "Combining multi-core and GPU computing for solving combinatorial optimization problems," *Journal of Parallel and Distributed Computing*, vol. 73, no. 12, pp. 1563–1577, 2013.

[33] K. Rocki and R. Suda, "Accelerating 2-opt and 3-opt local search using GPU in the travelling salesman problem," in *Proceedings of the 10th Annual International Conference on High Performance Computing and Simulation (HPCS '12)*, pp. 489–495, July 2012.

[34] A. Delévacq, P. Delisle, M. Gravel, and M. Krajecki, "Parallel ant colony optimization on graphics processing units," *Journal of Parallel and Distributed Computing*, vol. 73, no. 1, pp. 52–61, 2013.

[35] A. Uchida, Y. Ito, and K. Nakano, "An efficient gpu implementation of ant colony optimization for the traveling salesman problem," in *Proceedings of the 3rd International Conference on Networking and Computing (ICNC '12)*, pp. 94–102, December 2012.

[36] J. Hofmann, S. Limmer, and D. Fey, "Performance investigations of genetic algorithms on graphics cards," *Swarm and Evolutionary Computation*, vol. 12, pp. 33–47, 2013.

[37] U. Cekmez, M. Ozsiginan, and O. K. Sahingoz, "Adapting the ga approach to solve traveling salesman problems on cuda architecture," in *Proceedings of the 14th IEEE International Symposium on Computational Intelligence and Informatics (CINTI '13)*, pp. 423–428, November 2013.

[38] A. A. Khan, M. U. Khan, and M. Iqbal, "Multilevel graph partitioning scheme to solve traveling salesman problem," in *Proceedings of the 9th International Conference on Information Technology (ITNG '12)*, pp. 458–463, April 2012.

[39] F. J. D. Delgadillo, O. Montiel, and R. Sepúlveda, "Applications of artificial immune algorithms," in *High Performance Programming for Soft Computing*, O. Montiel and R. Sepúlveda, Eds., chapter 10, CRC Press, 2014.

[40] K. Jebari, A. El Moujahid, A. Bouroumi, and A. Ettouhami, "Unsupervised fuzzy clustering-based genetic algorithms to traveling salesman problem," in *Proceedings of the International Conference on Multimedia Computing and Systems (ICMCS '12)*, pp. 1013–1015, May 2012.

[41] K. Sheibani, "The fuzzy greedy search in combinatorial optimization with specific reference to the travelling salesman problem," in *Proceedings of the IEEE International Conference on Industrial Engineering and Engineering Management (IEEM '10)*, pp. 1367–1370, December 2010.

[42] J.-W. Yoon and S.-B. Cho, "An efficient genetic algorithm with fuzzy c-means clustering for traveling salesman problem," in *Proceedings of the IEEE Congress of Evolutionary Computation (CEC '11)*, pp. 1452–1456, June 2011.

[43] S. H. Nasseri and M. H. Khaviari, "Solving TSP by considering processing time: meta-heuristics and fuzzy approaches," *Fuzzy Information and Engineering*, vol. 3, no. 4, pp. 359–378, 2011.

[44] N. Ernest and K. Cohen, "Fuzzy logic clustering of multiple traveling salesman problem for self-crossover based genetic algorithm," in *Proceedings of the 50th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, January 2012.

[45] Y.-F. Liao, D.-H. Yau, and C.-L. Chen, "Evolutionary algorithm to traveling salesman problems," *Computers and Mathematics with Applications*, vol. 64, no. 5, pp. 788–797, 2012.

[46] A. Chaudhuri, K. De, and D. Chatterjee, "A study of the traveling salesman problem using fuzzy self organizing map," in *Proceedings of the 3rd International Conference on Industrial and Information Systems (ICIIS '08)*, pp. 1–5, IEEE, Kharagpur, India, December 2008.

[47] T. Fischer and P. Merz, "Reducing the size of traveling salesman problem instances by fixing edges," in *Evolutionary Computation in Combinatorial Optimization: 7th European Conference, EvoCOP 2007, Valencia, Spain, April 11-13, 2007. Proceedings*, vol. 4446 of *Lecture Notes in Computer Science*, pp. 72–83, Springer, Berlin, Germany, 2007.

[48] C. H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization Algorithms and Complexity*, Dover, New York, NY, USA, 1998.

[49] Y. Zhu, Z. Tang, H. Dai, and S. Gao, "Cooperation artificial immune system with application to traveling sales-man problem," *ICIC Express Letters*, vol. 2, no. 2, pp. 143–148, 2008.

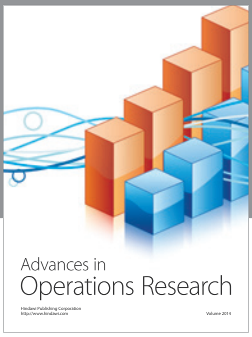
[50] M.-Y. Zhao, K. E. Tang, G. Lu et al., "A novel clonal selection algorithm and its application," in *Proceedings of the International Conference on Apperceiving Computing and Intelligence Analysis*

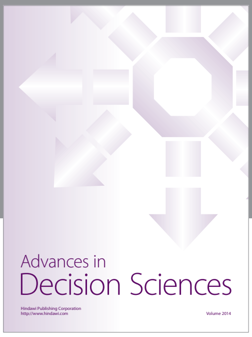
(ICACIA '08), pp. 385–388, IEEE, Chengdu, China, December 2008.

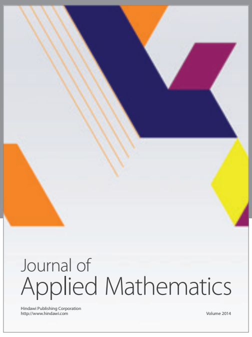
[51] D. Dasgupta and L. F. Nino, *Immunological Computation: Theory and Applications*, CRC Press, Taylor & Francis, Boca Raton, Fla, USA, 2009.

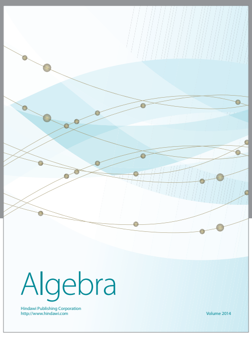
[52] L. de Castro and J. Timmis, *Artificial Immune Systems: A New Computational Intelligence Approach*, Springer, London, UK, 2002.

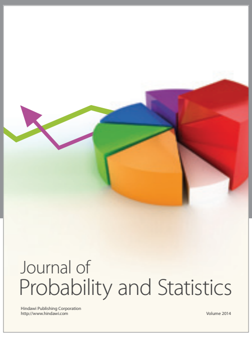
[53] S. Arora and B. Barak, *Computational Complexity A Modern Approach*, Cambridge University Press, Cambridge, UK, 2010.

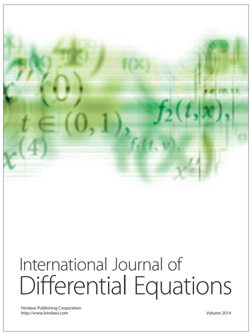












Hindawi

Submit your manuscripts at
http://www.hindawi.com