Bio-Inspired Algorithms and Its Applications for Optimization in Fuzzy Clustering

Fevrier Valdez, Oscar Castillo* and Patricia Melin

Abstract: In recent years, new metaheuristic algorithms have been developed taking as reference the inspiration on biological and natural phenomena. This nature-inspired approach for algorithm development has been widely used by many researchers in solving optimization problems. These algorithms have been compared with the traditional ones and have demonstrated to be superior in many complex problems. This paper attempts to describe the algorithms based on nature, which are used in optimizing fuzzy clustering in real-world applications. We briefly describe the optimization methods, the most cited ones, nature-inspired algorithms that have been published in recent years, authors, networks and relationship of the works, etc. We believe the paper can serve as a basis for analysis of the new area of nature and bio-inspired optimization of fuzzy clustering.

Keywords: fuzzy; clustering; optimization algorithm

1. Introduction

Optimization is a discipline for finding the best solutions to specific problems. Every day we developed many actions, which we have tried to improve to obtain the best solution; for example, the route for going to work can be optimized depending on several factors, such as traffic and distance. On other hand, the design of the new cars implies an optimization process with many objectives such as wind resistance, reduce the use of fuel, and maximize the potency of motor. These best solutions are found by adapting the parameters of the algorithm to give either a maximum or a minimum value for the solution. Therefore, in the last years many optimization methods have been developed with the aim of improving existing solutions.

Nowadays, many optimization algorithms based on nature can be found in the literature, it is calculated that there are more than 150 different algorithms, and improved algorithms for finding the best results on the optimization problems [1–11]. However, it is not our aim to analyze all these methods. Instead, our approach will be on the bio-inspired algorithms that are dealing with optimizing fuzzy clustering. Therefore, we have selected only a few algorithms in this review. Although, we have worked with different algorithms in different ways, for example, with parameter adaptation using fuzzy logic, with the original methods, the selected methods were chosen because they have demonstrated to be a good alternative for solving many optimization problems, and we have experience working with them. However, we were focused on the applications related to optimization fuzzy clustering. The paper reports important information about relevant topics, applications with different optimization methods, etc., that are used by the researchers in this area. It is noteworthy that there are many available tools that can be used to collect information and then analyze any area in science. In a few seconds, we are able to observe the graphic representation in different ways, which is useful to research a particular topic. The scientific community can benefit from knowing and quickly accessing topics, authors, countries, journal types, years, etc., around the world. This is useful, because although there is a large quantity of references and stored records; we can use the tools for building networks
with the nodes and clusters automatically. Finally, to obtain, for example, who is working with a particular topic, what author has more citations in a particular year. In this case, we focused only on some topics about bio-inspired optimization of fuzzy clustering. However, with this idea, many authors can use different tools to extract interesting information about any area of science and technology.

The main contribution is to describe the algorithms based on nature that are used in the optimization of fuzzy clustering results for real applications. The analyzed fuzzy clustering methods are very interesting and up to date, being at the same time one of the mostly used tools in current research. The nature or bio-inspired optimization methods are very popular due to their relatively simple use, without depending and derivatives. In this regard, we show how the nature-inspired optimizers are used in fuzzy clustering. Finally, the general analysis of the area, meaning what has been published to the moment, offers to reader an up-to-date picture of the status of the area, but also can provide possible windows of opportunity for future lines of research.

Nature inspired algorithms can be classified as those based on biology and those inspired on natural phenomena. The algorithms based on biology can be further divided into those based on evolution and those based on swarm behavior. The evolutionary algorithms include the genetic algorithms, differential evolution, cultural evolution, evolutionary strategies, and genetic programming. The swarm category includes particle swarm optimization, ant colony optimization [12], artificial bees [13], termites [14], bats [15], birds [16], cats [17], bacterial foraging [18], cuckoo search [19], and firefly algorithm [20]. Moreover, there are algorithms based on the physical laws; for example, simulated annealing, the gravitational search algorithm, and the big bang big crunch algorithm. The main goal of this paper was analyzing which bio-inspired or nature-inspired algorithms have been used in the area of fuzzy clustering, so that reader can get an idea of what has been done in this area. We also describe the basic concepts and notations of both fuzzy clustering and optimization algorithms, and illustrate the general architecture of how the two areas interact for achieving the goal of optimizing the results of fuzzy clustering (Sections 2 and 3).

2. Fuzzy Clustering

The objective function for fuzzy clustering denoted by $J_m$ is given by the equation:

$$J_m(P, V) = \sum_{i=1}^{k} \sum_{j=1}^{n} \mu_{ij}^m \| x_j - v_i \|^2, \quad m > 1$$

(1)

where $P$ denotes a fuzzy partition of a data set $X$ formed by clusters $C_1, C_2, \ldots, C_k$. The $m$ parameter is a weight that determines the degree with which each datum belongs to a cluster in a partial way, and of course this parameter affects the clustering result. The fuzzy c-means (FCM) algorithm aims at finding a good partition by finding the centers of clusters $v_i$ that minimize the objective function $J_m$. The FCM algorithm also finds the membership values $\mu_{ij}$ that minimize $J_m$. To achieve both goals of finding a good partition and membership values, a condition is needed to guarantee a local minima of $J_m$, which can be mathematically derived from Equation (1) by calculating the partial derivatives and finding where they have a zero value. This mathematical derivation will produce the following two expressions to find the memberships and centers of clusters, respectively [21].

$$\mu_{ij} = \frac{1}{\sum_{i=1}^{k} \left( \frac{\| x_j - v_i \|^2}{\| x_j - v_l \|^2} \right)^{\frac{1}{m-1}}} , \quad 1 \leq i \leq k, \quad 1 \leq j \leq n$$

(2)

Based on Equations (2) and (3), the fuzzy c-means algorithm can be used for iteratively updating the centers of clusters and the membership degrees until a convergence criteria is satisfied. As a consequence, the FCM algorithm can be outlined with the following steps:

1. Given a preselected number of clusters $k$ and a selected value of $m$, an initialization of memberships $\mu_{ij}$ of $x_j$ from cluster $i$, with values from 0 to 1 is given.
2. Calculate the centroids \(v_i\) of the fuzzy cluster for \(i = 1, 2, \ldots, c\) with Equation (3).
3. Update the memberships \(\mu_{ij}\) using Equation (2).
4. If \(|J_m(t) - J_m(t-1)|\) is lower than a threshold \(\varepsilon\), then the process is stopped, if not then go to Step 2.

This is the original proposal of the fuzzy clustering that was put forward by Jim Bezdek, but now there exist many improved versions of the original FCM algorithm that can be found in the literature. Improved versions have been aimed at improving convergence or the quality of the clusters, but the original idea is the same [21].

3. Bioinspired Optimization of Fuzzy Clustering

We describe in this Section how bio-inspired optimization algorithms can be used to optimize the performance of fuzzy clustering algorithms for real-world problems. In Figure 1 the general architecture of this kind of optimization of fuzzy clustering is presented. As was previously described in Section 2, usually we have to predefine the values of parameters \(m\) and \(c\) before executing fuzzy clustering, but in reality, we do not know a-priori their values. In this sense, these two parameters can be subject to optimization. For this reason, in Figure 1 we are showing \(m\) and \(c\) as outputs of the optimization algorithm, and will be used as inputs by the fuzzy clustering algorithm. The fuzzy clustering block will also receive as inputs, the threshold \(\varepsilon\) and the dataset \(X\) that needs to be clustered. The outputs of the fuzzy clustering block will the vector of centroids \(V\) of the clusters that have been formed. In the clustering block of Figure 1, four clusters are shown as illustration of the result, and after the clusters were formed a validation index \(I\) of the clustering quality is calculated, which is then feedback to the optimization algorithm to be used an objective function. After that, process is repeated until a termination criterion is satisfied.

![Figure 1](image)

**Figure 1.** Architecture of bio-inspired optimization fuzzy clustering approaches.

Figure 1 illustrates the general idea of bio-inspired optimization of fuzzy clustering, but if other more advanced variants of FCM are considered, then just more parameters will need to be produced by the optimization algorithm.
4. Literature Review

In this section, we made a general review about the methods using optimization fuzzy clustering with different bio-inspired optimization methods. However, in the following sections a deep study is developed by doing specific queries to Web of Science, and the VOSviewer tool is used to calculate the clusters of the analyzed works. In Table 1, is presented a list with the most popular bio-inspired optimization algorithms based on swarms, physics, populations, chemistry, and evolution. This table shows many methods in chronological orders that have been used since 1975 to date. However, only there are some methods, but can be useful to expand the knowledge about these methods and to observe the inspiration type. We made the query from Web of Science: “Optimization fuzzy clustering”, we found a total of 2208 papers with this topic. However, in this paper only a description of the most recent works is presented, but with the query above mentioned can be seen the updated works. Figure 2 shows the countries with greater number of publications. Additionally, to validate this study in the best way, with the collected dates, we applied a filter to calculate the number of papers and citations from 2011 to 2021. In total, were found 1752 records of Web of Science. With this available information, was possible to build a preferred reporting items for systematic reviews and meta-analyses (PRISMA) flow diagram described in Figure 3. It was the aim to observe with more detail the retrieved papers presented in Figure 2, with the main topic “Optimization fuzzy clustering”, and using the “citespace” tool mentioned below in this paper. We built the network corresponding to flow diagram of Figure 3. Therefore, in Figure 4 it can be appreciated the network with a total of 456 nodes, 3137 links using 1717 qualified records. Furthermore, in this figure the clusters were automatically labeled using the keyword that appears most commonly in the search. We only reviewed information from Web of Science because it contains many relevant journals. However, the tools used in this review can also use information from Scopus or other available datasets, if required.

Figure 2. Plot of 2225 papers of Web of Science for topic: (optimization fuzzy clustering) by countries.
Recently, multi-view clustering research has attracted a lot of attention because of the rapidly increasing demand for unsupervised analysis of multi-view data in practical applications. In [21], a novel two-level weighted collaborative multi-view fuzzy clustering (TW-Co-MFC) approach was presented, to address the aforementioned issues.
Furthermore, in [22], authors proposed image segmentation using the bat algorithm with fuzzy C means clustering. The proposed segmentation technique was evaluated with existing segmentation techniques. On the other hand, in [23], the authors presented a hybridization of stud krill herd optimization with refined kernel fuzzy c-means clustering algorithm for efficient moving object exploration.

Another recent study on this area is shown in [24], where the authors presented a hybrid semi-supervised interval type-2 possibilistic fuzzy c-means clustering and particle swarm optimization for optimizing results in satellite image analysis.

Additionally, in [25] the glow-worm swarm algorithm was used for optimization of fuzzy based unequal clustering and context aware routing procedure. This was developed for random way point based dynamic wireless sensor networks. Based on fuzzy systems the unequal clustering is formed and the optimal cluster centroid is nominated to convey the information from cluster member to base station to increase the system lifespan and to decrease the energy consumption.

Table 1. Popular bio-inspired optimization algorithms based on swarm intelligence, physics, population, chemistry, and evolution.

| Year | Algorithms and References                                      |
|------|----------------------------------------------------------------|
| 2021 | Horse herd optimization algorithm [26]                        |
| 2020 | Mayfly Optimization Algorithm [27]                            |
| 2020 | Chimp Optimization Algorithm [28]                             |
| 2020 | Coronavirus Optimization Algorithm [29]                       |
| 2020 | Water strider algorithm [30]                                  |
| 2020 | Newton metaheuristic algorithm [31]                            |
| 2020 | Black Widow Optimization Algorithm [32]                       |
| 2019 | Harris hawks optimization [33]                                |
| 2019 | Sailfish Optimizer [34]                                       |
| 2019 | Spider Monkey Optimization [35]                               |
| 2017 | Grasshopper Optimisation Algorithm [36]                        |
| 2017 | Fractal Based Algorithm [37]                                  |
| 2017 | Bacterial Foraging Inspired Algorithm [18]                    |
| 2017 | Rain-fall Optimization Algorithm [38]                         |
| 2016 | Dragonfly algorithm [39]                                      |
| 2016 | Sperm Whale Algorithm [40]                                    |
| 2015 | Water Wave Optimization [41]                                  |
| 2015 | Ant Lion Optimizer [42]                                       |
| 2014 | Symbiotic Organisms Search [43]                               |
| 2013 | Egyptian Vulture Optimization Algorithm [44]                  |
| 2013 | Dolphin echolocation [45]                                     |
| 2012 | Great Salmon Run [46]                                         |
| 2012 | Big Bang-Big Crunch [47]                                      |
| 2012 | Flower Pollination Algorithm [48]                             |
| 2011 | Spiral Optimization Algorithm [49]                            |
| 2011 | Galaxy-based Search Algorithm [50]                            |
Table 1. Cont.

| Year | Algorithms and References                      |
|------|-----------------------------------------------|
| 2010 | Japanese Tree Frogs [51]                      |
| 2010 | Bat Algorithm [15]                            |
| 2010 | Termite Colony Optimization [14]              |
| 2010 | Firefly Algorithm [20]                        |
| 2009 | Cuckoo Search [19]                            |
| 2009 | Glowworm Swarm Optimization [52]              |
| 2009 | Bee Colony Optimization [53]                  |
| 2009 | Gravitational Search Algorithm [54]           |
| 2008 | Fast Bacterial Swarming Algorithm [55]        |
| 2007 | River Formation Dynamics [56]                 |
| 2007 | Imperialistic Competitive Algorithm [57]      |
| 2008 | Roach Infestation Optimization [58]           |
| 2006 | The bees Algorithm [13]                       |
| 2006 | Cat Swarm Optimization [17]                   |
| 2004 | BeeHive [59]                                  |
| 2003 | Queen-Bee Evolution [60]                      |
| 2001 | Harmony Search Algorithm [61]                 |
| 1995 | Particle Swarm Optimization [16]              |
| 1992 | Genetic Programming [62]                      |
| 1992 | Ant Colony Optimization [12,63]               |
| 1989 | Tabu Search [64]                              |
| 1975 | Genetic Algorithms [65]                       |

5. Bio-Inspired Optimization Methods

This section presents the algorithms used as reference in this study. After making a complete review of the above-mentioned methods, we decided to include, in making this study, some important and relevant methods along the history of this area. Though, there are many algorithms, it is impossible to include all methods. However, with these selected methods it is possible to give us an idea of the relationship of authors, citations, cluster of work networks with specific queries from high impact journals and other important information. The main aim of this section is to briefly outline the basic concepts of several bio-inspired optimization algorithms for a better comprehension of the importance of this area. The selected algorithms are presented in the following sub-sections.

5.1. Genetic Algorithms

John Holland initiated his pioneering work on genetic algorithms at the beginning of the 1960s. His first achievement was the publication of Adaptation in Natural and Artificial Systems in 1975, in which he put forward a popular method in the evolutionary computation field, known as genetic algorithm. In the simple genetic algorithm, the representation that is used is a binary string. Each position in the string is assumed to represent a particular feature of an individual, and the value stored in that position represents how that feature is coded in the solution. Usually, the string is “evaluated as a collection of structural features of a solution that have little or no interactions”. The analogy may be drawn directly to genes in biological organisms. Each gene represents an entity that is structurally independent of other genes. The main reproduction operator used is bit-
string crossover, in which two strings are used as parents and new individuals are formed by swapping sub-sequences between the two strings. Another popular operator is bit-flipping mutation, in which a single bit in the string is flipped to form a new offspring string. A variety of other operators has also been put forward by other researchers, but these less frequently used. A primary distinction that may be made between the various operators is whether or not they introduce any new information into the population. All operators are also constrained to manipulate the strings in a manner consistent with the structural interpretation of genes. For example, two genes at the same location on two strings may be swapped between parents, but not combined based on their values. Traditionally, individuals are probabilistically selected to be the parents based upon their fitness values, and the offspring that are created replace the parents. One popular method for doing this is the roulette wheel selection approach. For example, if N parents are selected, then N offspring are generated, which replace the parents in the next generation [65].

5.2. Particle Swarm Optimization

Particle swarm optimization (PSO) is a population based stochastic optimization technique that was put forward by Eberhart and Kennedy in 1995, which was inspired by social behavior of bird flocking or fish schooling. However, in the algorithm birds or fish are viewed as particles, which is a more general abstraction.

PSO shares many similarities with evolutionary computation techniques, such as with the genetic algorithm (GA). The system is initialized with a random population solutions and searches for optima by updating generations. However, unlike the GA, the PSO has no evolution operators, such as crossover and mutation. Instead, parameters of social and individual behaviors are used in PSO, and velocity and position of the particles. In PSO, the potential solutions, called particles, fly through the problem search space by following the current best particles [1,11,16].

Another reason that PSO is attractive is that there are few parameters to adjust. One version, with slight variations, works well in a wide variety of applications. Particle swarm optimization has been used for approaches that can be applicable across a wide range of problems, as well as for specific applications focused on a specific requirement.

5.3. Cuckoo Search Algorithm

Cuckoo optimization algorithm (CSA) is inspired on the life of a bird called “cuckoo”. The idea of this algorithm is the specific breeding and egg laying of this bird. Adult cuckoos and eggs are used in this method. The cuckoos which are adults lay eggs in the habitat of other birds. These eggs grow and become a mature cuckoo if they are not found and not eliminated by the host birds. The immigration of populations of birds and environmental specifications hopefully lead them to converge and reach the best place for reproduction and breeding. The objective function is in this best place [19]. CSA is a new continuous metaheuristic based on the life of a cuckoo bird. In the same way to other optimization methods based on populations, CSA starts with a population, a set of cuckoos. These cuckoos lay some eggs in the habitat of other host birds. A random set of potential solutions is generated that are considered to represent the habitat in CSA. The movement of cuckoos is usually modeled with Levy flights, which is given by a stochastic mathematical model.

5.4. Bat Algorithm

The bat algorithm (BA) is a bio-inspired algorithm based on bat behavior that has been found to be very efficient in solving complex optimization tasks. If we idealize some of the echolocation characteristics of bats, we can propose various bat-inspired algorithms or bat algorithms. For simplicity, we now use the next idealized rules:

1. All bats use echolocation to sense distance, and they also “know” the difference between food/prey and background barriers in some unknown way.
2. Bats fly randomly with velocity $v_i$ at position $x_i$ with a fixed frequency, varying wavelength and loudness $A_0$ to search for prey. Bats can automatically adjust the wavelength (or frequency) of their emitted pulses and adjust the rate of pulse emission $r \in [0,1]$, depending on the proximity of their target.

3. Although loudness can vary in many ways, we assume that the loudness varies from a large (positive) $A_0$ to a minimum constant value $A_{\text{min}}$. For simplicity, the frequency $f$ is in the interval $[0, \text{max}]$, the new solutions and velocity at a specific time step $t$ are represented by a random vector drawn from a uniform distribution $[15]$.

6. Experimental Results

This section summarizes the obtained results with the different tools available in the literature for building the networks clusters, relationships, and citations, with the analyzed methods. To validate the queried information of Web of Science, the VOSviewer tool [66] was used. However, this type of studies can be made by other similar tools for making bibliometric analysis. For example, Bibliometrix [67] is a free tool that provides various routines for importing bibliographic data from SCOPUS and Clarivate Analytics’ Web of Science databases; Bibliotool [68], is a set of Python scripts written by Sebastian Grauwin. They can read ISI data in CSV format and do some studies including co-occurrence map and bibliographic coupling. Finally, CiteSpace [69] is a free Java-based software for visualizing and analyzing trends and patterns in the scientific literature. It was designed as a tool for progressive knowledge domain visualization. Additionally, to show in a flow diagram how the collected information was obtained, we used the PRISMA [70] methodology, which help us to analyze in the best possible way the obtained results.

6.1. Study with Genetic Algorithms

This section is presenting the obtained results of the queries from Web of Science with the topic “optimization fuzzy clustering with genetic algorithms”. First, it was necessary to access the web of science, and then make the desired queries. Once the information was extracted, and using the VOSviewer tool, it was possible to calculate the related works, citations, authors, etc. Figure 5 represents a map based on network data collected from the bibliographic database in Web of Science. The type of analysis represented in this figure is by co-occurrence, the unit of analysis was by keywords, the minimum number of documents of an author was 5, minimum number of citations was 0, the counting method was full counting, minimum number of occurrences of a keyword was 5, and finally, for each of the 92 keywords, the total strength of the co-occurrence links with other keywords was calculated. On VOSviewer, if the keyword “genetic algorithm” is selected, then we can appreciate the number of clusters is 7 for this selection, with 88 links, and 141 occurrences. Figure 6, represents the selection of the keyword “genetic algorithm” that corresponds to the information obtained from Figure 5.

Figure 7 shows the total number of papers collected from Web of Science and that were used to make the calculus above described in Figures 5 and 6. It can be seen how the number of citations and papers with the analyzed topic has been increasing in recent years.

Also, with this information was possible to observe, the record by authors, where in Figure 8, it can be appreciated that two authors are the leaders in this area with the topic “optimization fuzzy clustering with genetic algorithms”.

6.2. Study with Particle Swarm Optimization

In this section, is presented the obtained results of the queries from Web of Science with the topic “optimization fuzzy clustering with particle swarm optimization algorithms”. First, it was necessary to access the web of science, and then make the desired queries. Once the information was extracted, and using the VOSviewer tool, it was possible to calculate the related works, citations, authors, etc. Figure 9 represents a map based on network data collected from the bibliographic database from Web of Science. The type of analysis
represented in this Figure is by co-occurrence, the unit of analysis was by keywords, the minimum number of documents of an author was 5, minimum number of citations was 0, the counting method was full counting, minimum number of occurrences of a keyword was 5, and finally, for each of the 116 keywords, the total strength of the co-occurrence links with other keywords was calculated. On VOSviewer, if the keyword “particle swarm optimization” is selected, we can appreciate the number of clusters is 8 for this selection, with 108 links, and 234 occurrences.

Figure 5. Total cluster obtained with the search “optimization fuzzy clustering with genetic algorithms” from VOSviewer.

Figure 6. Representation of the selection of the keyword “genetic algorithm”.

Figure 7 shows the total number of papers collected from Web of Science and that were used to make the calculus above described in Figures 5 and 6. It can be seen how the number of citations and papers with the analyzed topic has been increasing in recent years.
Figure 7. Citation report for 369 results from Web of Science Core Collection.

Figure 8. Record by authors for the topic: Optimization fuzzy clustering with genetic algorithms.
6.2. Study with Particle Swarm Optimization

In this section, is presented the obtained results of the queries from Web of Science with the topic "optimization fuzzy clustering with particle swarm optimization algorithms". First, it was necessary to access the web of science, and then make the desired queries. Once the information was extracted, and using the VOSviewer tool, it was possible to calculate the related works, citations, authors, etc. Figure 9 represents a map based on network data collected from the bibliographic database from Web of Science. The type of analysis represented in this Figure is by co-occurrence, the unit of analysis was by keywords, the minimum number of documents of an author was 5, minimum number of citations was 0, the counting method was full counting, minimum number of occurrences of a keyword was 5. Finally, for each of the 116 keywords, the total strength of the co-occurrence links with other keywords was calculated. On VOSviewer, we can appreciate the number of clusters is 8 for this selection, with 108 links, and 234 occurrences.

Figure 9. Total clustering obtained with the search “optimization fuzzy clustering with particle swarm optimization” from VOSviewer.

Figure 10, represents the selection of the keyword “particle swarm optimization” that corresponds to the information obtained from Figure 9.

Furthermore, with this information it was possible to observe, the records by authors, where in Figure 12, it can be appreciated that two authors are the leaders in this area with the topic “optimization fuzzy clustering with particle swarm optimization”.

Figure 11 shows the total number of papers collected from Web of Science and that were used to make the calculations above described in Figures 9 and 10. It can be seen how the number of citations and papers with the analyzed topic have been increasing continuously in recent years.

6.3. Study with the Cuckoo Search Algorithm

In this section, we present the obtained results of the queries in Web of Science with the topic “optimization fuzzy clustering with Cuckoo Search Algorithm”. The main difference with the other analyzed algorithms was that only 23 papers were found with the reviewed topic. Moreover, it was necessary to access the web of science, and then make the desired queries. Once the information was extracted, and using the VOSviewer tool, it was possible to calculate the related works, citations, authors, etc. Figure 13 represents a map based on network data collected from the bibliographic database from Web of Science. The type of analysis represented in this Figure is by co-occurrence, the unit of analysis was by keywords, the minimum number of documents of an author was 5, minimum number of citations was 0, the counting method was full counting, minimum number of occurrences of a keyword was 5. Finally, for each of the 3 keywords, the total strength of the co-occurrence links with other keywords was calculated. On VOSviewer, we can appreciate the number of clusters is 2 as can be seen in Figure 13 with only 1 link. With these results, it can be seen that this method has not been widely used or combined with fuzzy clustering. However,
we expect that in the future this optimization algorithm will become more popular due to its good performance in other problems.

Figure 10. Representing the selection of the keyword “particle swarm optimization”.

Figure 11. Citation report for 508 results from Web of Science Core Collection.
Furthermore, with this information it was possible to observe, the records by authors, where in Figure 12, it can be appreciated that two authors are the leaders in this area with the topic “optimization fuzzy clustering with particle swarm optimization”.

Figure 12. Record by authors for topic: Optimization fuzzy clustering with particle swarm optimization.

Figure 14 shows the total number of papers collected from Web of Science and that were used to make the calculations above described in Figure 13. It can be noted how the number of citations and papers are less than the other analyzed methods.

Also, with this information was possible to observe, the records by authors, where in Figure 15 it can be appreciated that two authors are the leaders in this area with the topic “optimization fuzzy clustering with cuckoo search algorithm”.

Figure 13. Total cluster obtained with the search “optimization fuzzy clustering with cuckoo search optimization” from VOSviewer.
6.4. Study with Bat Algorithm

In this section a summary of the obtained results of the queries in Web of Science with the topic “optimization fuzzy clustering with Bat Algorithm” is presented. The main difference with the other analyzed algorithms was that only 14 papers were found with the
reviewed topic. Furthermore, it was necessary to access the web of science, and then make
the desired queries. Once the information was extracted, and using the VOSviewer tool, it
was possible to calculate the related works, citations, authors, etc.

Figure 16 illustrates a map based on network data collected from the bibliographic
database in Web of Science. The type of analysis represented in this figure is by co-
ocurrence, the unit of analysis was by keywords, the minimum number of documents
of an author was 5, minimum number of citations was 0, the counting method was full
counting, minimum number of occurrences of a keyword was 5. Finally, for each of the 3
keywords, the total strength of the co-occurrence links with other keywords was calculated.
On VOSviewer, we can appreciate the number of clusters is 1, as can be seen in Figure 16
with only 1 link.

With these results, it can be seen that this method has not been widely used or
combined with fuzzy clustering.

Figure 17 shows the total of papers collected from Web of Science and that were used
to make the calculations above described in Figure 16. It can be seen how the number of
citations and papers are less than the other analyzed methods.

Also, with this information it was possible to observe, the record by authors, where in
Figure 18, it can be appreciated that two authors are the leaders in this area with the topic
“optimization fuzzy clustering with bat algorithm”.

6.5. Analysis by Authors

In this section an analysis by authors, considering the total citations from web of
science is presented. We can appreciate that the author with more publications in this
area with the analyzed algorithms in this paper is Witold Pedrycz from the University of
Alberta, Canada. According with the information collected from Web of Science, Figure 19
shows the total of the publications of this author.

Figure 16. Total cluster obtained with the search “optimization fuzzy clustering with bat algorithm” from VOSviewer.
Figure 17. Citation report for 14 results from Web of Science Core Collection.

Figure 18. Record by authors for topic: Optimization fuzzy clustering with bat algorithm.
6.5. Analysis by Authors

In this section an analysis by authors, considering the total citations from web of science is presented. We can appreciate that the author with more publications in this area with the analyzed algorithms in this paper is Witold Pedrycz from the University of Alberta, Canada. According with the information collected from Web of Science, Figure 19 shows the total of the publications of this author.

Figure 19. Citation report for 1001 results from the Web of Science Core Collection by the author Witold Pedrycz.

Figure 20 was calculated in VOSviewer and represents the relationship of authors with Witold Pedrycz in the area of fuzzy clustering. The graph was made considering the global work with a total of 1001 publications collected from Web of Science.

The previous results provide a picture of the state of the art of what has already been done in the area of bio-inspired optimization for fuzzy clustering applied in real-world applications. We can note that GAs and PSO are the two bio-inspired method that have been used the most in optimizing fuzzy clustering, while the cuckoo search and bat algorithm have been used very little. In addition, we envision that more bio or nature inspired optimization algorithms will become more popular in fuzzy clustering applications in future years.

7. Conclusions

After reviewing the state of the art about the optimization of fuzzy clustering area with bio-inspired optimization methods, we decided to make an analysis, considering
four optimization methods, which we have recently used in this area. With all collected information of Web of Science and the VOSviewer tool, we can observe that genetic algorithms and particle swarm optimization are two very popular methods that the authors have been using in the last years. On the other hand, the cuckoo search and bat algorithms, are two newer methods that have more recently becoming more popular. However, not many authors have yet attempted to optimize fuzzy clustering using these two methods. Furthermore, we were able to review the author with more publications in this area. As a future work, this review can be extended analyzing other types of optimization methods with fuzzy clustering. The type of queries can be made by authors, keywords, occurrences, etc. Finally, with this study we can see, that not all optimization methods that were analyzed here, have been used with fuzzy clustering, or maybe with the same frequency, but we believe that some of the recent methods in optimization can become an alternative to the most popular methods in specific applications. For example, we have recently made experiments with methods like the multiverse optimization algorithm or the method based on self-defense of plants, which show very good properties in searching for optimal solutions and could become viable alternatives for optimizing fuzzy clustering in real applications.

Author Contributions: Conceptualization, O.C. and P.M.; methodology, O.C.; software, F.V.; validation, O.C., F.V. and P.M.; formal analysis, O.C.; investigation, P.M.; resources, P.M.; data curation, F.V.; writing—original draft preparation, O.C.; writing—review and editing, P.M.; visualization, F.V.; supervision, O.C.; project administration, P.M.; funding acquisition, P.M. All authors have read and agreed to the published version of the manuscript.

Funding: This paper did not receive funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank CONACYT and Tecnológico Nacional de Mexico/Tijuana Institute of Technology for the support during this research work.

Conflicts of Interest: The authors declare no conflict of interest.

References
1. Reyes-Sierra, M.; Coello, C. Multi-Objective Particle Swarm Optimizers: A Survey of the State-of-the-Art. Int. J. Comput. Intel. Res. 2006, 2, 287–308.
2. Valdez, F.; Melin, P.; Castillo, O. A survey on nature-inspired optimization algorithms with fuzzy logic for dynamic parameter adaptation. Expert Syst. Appl. 2014, 41, 6459–6466. [CrossRef]
3. Fister, I.; Fister, I.; Yang, X.S.; Brest, J. A comprehensive review of firefly algorithms. Swarm Evol. Comput. 2013, 13, 34–46. [CrossRef]
4. Chen, M.R.; Huang, Y.Y.; Zeng, G.Q.; Lu, K.D.; Yang, L.Q. An improved bat algorithm hybridized with extremal optimization and Boltzmann selection. Expert Syst. Appl. 2021, in press. [CrossRef]
5. Odili, J.B.; Noraziah, A.; Babalola, A.E. Flower pollination algorithm for data generation and analytics—A diagnostic analysis. Sci. Afr. 2020, 8, e00440. [CrossRef]
6. Gao, S.; Gao, Y.; Zhang, Y.; Li, T. Adaptive cuckoo algorithm with multiple search strategies. Appl. Soft Comput. 2021, 107181. [CrossRef]
7. Valdez, F. A review of optimization swarm intelligence-inspired algorithms with type-2 fuzzy logic parameter adaptation. Soft Comput. 2020, 24, 215–226.
8. Yue, C.; Suganthan, P.N.; Liang, J.; Qu, B.; Yu, K.; Zhu, Y.; Yan, L. Differential evolution using improved crowding distance for multimodal multiobjective optimization. Swarm Evol. Comput. 2021, 62, 100849. [CrossRef]
9. Wang, Y.; Gao, S.; Yu, Y.; Cai, Z.; Wang, Z. A gravitational search algorithm with hierarchy and distributed framework. Knowl. Based Syst. 2021, 218, 106877. [CrossRef]
10. Yan, Z.; Zhang, J.; Zeng, J.; Tang, J. Nature-inspired approach: An enhanced whale optimization algorithm for global optimization. Math. Comput. Simul. 2021, 185, 17–46. [CrossRef]
11. Bonyadi, M.R.; Michalewicz, Z. Particle Swarm Optimization for Single Objective Continuous Space Problems: A Review. Evol. Comput. 2017, 25, 1–54. [CrossRef] [PubMed]
12. Coloni, A.; Dorigo, M.; Maniezzo, V. An Investigation of Some Properties of an Ant Algorithm. In Proceedings of the Parallel Problem Solving from Nature Conference (PPSN 92), Brussels, Belgium, 28–30 September 1992; Männner, R., Manderick, B., Eds.; Elsevier Publishing: Amsterdam, The Netherlands, 1992; pp. 509–520.

13. Pham, D.T., Ghanbarzadeh, A.; Koç, E.; Otri, S.; Rahim, S.; Zaidi, M. The Bees Algorithm—A Novel Tool for Complex Optimization Problems; Elsevier Science Ltd.: Oxford, UK, 2006; pp. 454–459.

14. Hedayatzadeh, R.; Salmassi, F.A.; Keshigari, M.; Akbari, R.; Ziarati, K. Termitic colony optimization: A novel approach for optimizing continuous problems. In Proceedings of the 2010 18th Iranian Conference on Electrical Engineering. Isfahan, Iran, 11–13 May 2010; pp. 553–558.

15. Yang, X.S. A New Metaheuristic Bat-Inspired Algorithm. In Nature Inspired Cooperative Strategies for Optimization (NICSO 2010); Gonzalez, J.R., Pelta, D.A., Cruz, C., Terrazas, G., Krasnogor, N., Eds.; Springer: Berlin/Heidelberg, Germany, 2010; pp. 65–74.

16. Kennedy, J.; Eberhart, R. Particle swarm optimization. In Proceedings of the ICNN’95—International Conference on Neural Networks, Perth, WA, Australia, 27 November–December 1995; Volume 4, pp. 1942–1948.

17. Chu, S.C.; Wei Tsai, P.; Pan, J.S. Cat Swarm Optimization. In PRICAI 2006: Trends in Artificial Intelligence; Yang, Q., Webb, G., Eds.; Springer: Berlin/Heidelberg, Germany, 2006; pp. 854–858.

18. Lindfield, G.; Penny, J. Bacterial Foraging Inspired Algorithm. Introd. Nat. Inspired Optim. 2017, 101–117.

19. Yang, X.; Deb, S. Cuckoo Search via Levy flights. In Proceedings of the 2009 World Congress on Nature Biologically Inspired Computing (NaBIC), Coimbatore, India, 9–11 December 2009; pp. 210–214.

20. Yang, X.S. Firefly Algorithm, Stochastic Test Functions and Design Optimisation. Int. J. Bio Inspired Comput. 2010, 2, 78–84. [CrossRef]

21. Hu, J.; Pan, Y.; Li, T.; Yang, Y. TW-Co-MFC: Two-level weighted collaborative fuzzy clustering based on maximum entropy for multi-view data. Tsinghua Sci. Technol. 2021, 26, 185–198. [CrossRef]

22. Jai Shankar, B.; Murugan, K.; Obulesu, A.; Finney Daniel Shadrach, S.; Anitha, R. MRI Image Segmentation Using Bat Optimization Algorithm with Fuzzy C Means (BOA-FCM) Clustering. J. Med. Imaging Health Inform. 2021, 11, 661–666. [CrossRef]

23. Mahalingam, T. A hybridization of SKH and RKFCM clustering optimization algorithm for efficient moving object exploration. Multimed. Tools Appl. 2021, 1–32. [CrossRef]

24. Mai, D.S.; Ngo, L.T.; Trinh, L.H.; Hagras, H. A hybrid interval type-2 semi-supervised possibilistic fuzzy c-means clustering and particle swarm optimization for satellite image analysis. Inf. Sci. 2021, 548, 399–422. [CrossRef]

25. Vinodhini, R.; Gomathy, C. Fuzzy Based Unequal Clustering and Context-Aware Routing Based on Glow-Worm Swarm Optimization in Wireless Sensor Networks: Forest Fire Detection. Wirel. Pers. Commun. 2021, 1–22. [CrossRef]

26. MiarNaeimi, F.; Azizyan, G.; Rashki, M. Horse herd optimization algorithm: A nature-inspired algorithm for high-dimensional optimization problems. Knowl. Based Syst. 2021, 213, 106711. [CrossRef]

27. Zervoudakis, K.; Tsafarakis, S. A mayfly optimization algorithm. Comput. Ind. Eng. 2020, 145, 106559. [CrossRef]

28. Khishe, M.; Mosavi, M.R. Chimp optimization algorithm. Expert Syst. Appl. 2020, 149, 113338. [CrossRef]

29. Martínez-Álvarez, F.; Asencio-Cortés, G.; Torres, J.F.; Gutiérrez-Avilés, D.; Melgar-García, L.; Pérez-Chacón, R.; Rubio-Escudero, C.; Riquelme, J.C.; Troncoso, A. Coronavirus Optimization Algorithm: A Bioinspired Metaheuristic Based on the COVID-19 Propagation Model. Big Data 2020, 8, 308–322. [CrossRef] [PubMed]

30. Kaveh, A.; Eslamlooi, A.D. Water strider optimization algorithm: A new metaheuristic and applications. Structures 2020, 25, 520–541. [CrossRef]

31. Gholizadeh, S.; Danesh, M.; Gheyatmand, C. A new Newton metaheuristic algorithm for discrete performance-based design optimization of steel moment frames. Comput. Struct. 2020, 234, 106250. [CrossRef]

32. Hayyolalam, V.; Kazem, A.A.P. Black Widow Optimization Algorithm: A novel meta-heuristic approach for solving engineering optimization problems. Eng. Appl. Artif. Intell. 2020, 87, 103249. [CrossRef]

33. Heidari, A.A.; Mirjalili, S.; Faris, H.; Alajah, I.; Mafarja, M.; Chen, H. Harris hawks optimization: Algorithm and applications. Future Gener. Comput. Syst. 2019, 97, 849–872. [CrossRef]

34. Shadranvan, S.; Najj, H.R.; Bardsiri, V.K. The Sailfish Optimizer: A novel nature-inspired metaheuristic algorithm for solving constrained engineering optimization problems. Eng. Appl. Artif. Intell. 2019, 80, 20–34. [CrossRef]

35. Sharma, H.; Hazrati, G.; Bansal, J.C. Spider Monkey Optimization Algorithm. In Evolutionary and Swarm Intelligence Algorithms; Bansal, J.C., Singh, P.K., Pal, N.R., Eds.; Springer International Publishing: Cham, Switzerland, 2019; pp. 43–59.

36. Sarem, S.; Mirjalili, S.; Lewis, A. Grasshopper Optimisation Algorithm: Theory and application. Adv. Eng. Softw. 2017, 105, 30–47. [CrossRef]

37. Kaedi, M. Fractal-based Algorithm: A New Metaheuristic Method for Continuous Optimization. Int. J. Artif. Intell. 2017, 15, 76–92.

38. Kabol, S.H.A.; Selvaraj, J.; Rahim, N.A. Rain-fall optimization algorithm: A population based algorithm for solving constrained optimization problems. J. Comput. Sci. 2017, 19, 31–42. [CrossRef]

39. Mirjalili, S. Dragonfly algorithm: A new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. Neural Comput. Appl. 2016, 27, 1053–1073. [CrossRef]

40. Ebromahi, A.; Khamehchi, E. Sperm whale algorithm: An effective metaheuristic algorithm for production optimization problems. J. Nat. Gas. Sci. Eng. 2016, 29, 211–222. [CrossRef]

41. Zheng, Y.J. Water wave optimization: A new nature-inspired metaheuristic. Comput. Oper. Res. 2015, 55, 1–11. [CrossRef]

42. Mirjalili, S. The Ant Lion Optimizer. Adv. Eng. Softw. 2015, 83, 80–98. [CrossRef]
43. Cheng, M.Y.; Prayogo, D. Symbiotic Organisms Search: A new metaheuristic optimization algorithm. *Comput. Struct.* **2014**, *139*, 98–112. [CrossRef]

44. Sur, C.; Sharma, S.; Shukla, A. Egyptian Vulture Optimization Algorithm—A New Nature Inspired Meta-heuristics for Knapsack Problem. *Adv. Intell. Syst. Comput.* **2015**, *209*, 227–237.

45. Kaveh, A.; Farhoudi, N. A new optimization method: Dolphin echolocation. *Adv. Eng. Softw.* **2013**, *59*, 53–70. [CrossRef]

46. Mozaffari, A.; Fathi, A.; Behzadipour, S. The great salmon run: A novel bio-inspired algorithm for artificial system design and optimisation. *Int. J. Bio-Inspired Comput.* **2012**, *4*, 286–301. [CrossRef]

47. Zandi, Z.; Afjei, A.; Sedighizadeh, M. Reactive power dispatch using Big Bang-Big Crunch optimization algorithm for voltage stability enhancement. In Proceedings of the 2012 IEEE International Conference on Power and Energy (PECon), Kota Kinabalu, Malaysia, 2–5 December 2012; pp. 239–244.

48. Yang, X.S. Flower Pollination Algorithm for Global Optimization. In *Unconventional Computation and Natural Computation*; Durand-Lose, J., Jonoska, N., Eds.; Springer: Berlin/Heidelberg, Germany, 2012; pp. 240–249.

49. Tamura, K.; Yasuda, K. Primary study of spiral dynamics inspired optimization. *IEEE Trans. Electr. Electron. Eng.* **2011**, *6*, 98–100. [CrossRef]

50. Shah-Hosseini, H. Principal components analysis by the galaxy-based search algorithm: A novel metaheuristic for continuous optimisation. *Int. J. Comput. Sci. Eng.* **2011**, *6*, 132–140.

51. Pibernat, H.; Blum, C. Distributed Graph Coloring: An Approach Based on the Calling Behavior of Japanese Tree Frogs. *Swarm Intell.* **2012**, *6*, 117–150.

52. Krishnand, K.N.; Ghose, D. Glowworm Swarm Optimisation: A New Method for Optimising Multi-Modal Functions. *Int. J. Comput. Intell. Stud.* **2009**, *1*, 93–119. [CrossRef]

53. Teodorovic’, D. Bee Colony Optimization (BCO). In *Innovations in Swarm Intelligence*; Lim, C.P.; Jain, L.C.; Dehuri, S., Eds.; Springer: Berlin/Heidelberg, Germany, 2009; pp. 39–60.

54. Rashedi, E.; Nezamabadi-pour, H.; Saryazdi, S. GSA: A Gravitational Search Algorithm. *Inf. Sci.* **2009**, *179*, 2232–2248. [CrossRef]

55. Chu, Y.; Mi, H.; Liao, H.; Ji, Z.; Wu, Q.H. A Fast Bacterial Swarming Algorithm for high-dimensional function optimization. In Proceedings of the 2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence), Hong Kong, China, 1–6 June 2008; pp. 3135–3140.

56. Rabanal, P.; Rodríguez, I.; Rubio, F. Using River Formation Dynamics to Design Heuristic Algorithms. In Proceedings of the International Conference on Unconventional Computation, Kingston, ON, Canada, 13–17 August 2007; Springer: Berlin/Heidelberg, Germany, 2007; Volume 4618, pp. 465–478.

57. Atashpaz-Gargari, E.; Lucas, C. Imperialist competitive algorithm: An algorithm for optimization inspired by imperialist competition. In Proceedings of the 2007 IEEE Congress on Evolutionary Computation, Singapore, 25–28 September 2007; pp. 4661–4667.

58. Havens, T.C.; Spain, C.J.; Salmon, N.G.; Keller, J.M. Roach Infestation Optimization. In Proceedings of the 2008 IEEE Swarm Intelligence Symposium, St. Louis, MO, USA, 21–23 September 2008; pp. 1–7.

59. Wedde, H.F.; Farooq, M.; Zhang, Y. BeeHive: An Efficient Fault-Tolerant Routing Algorithm Inspired by Honey Bee Behavior. In *Ant Colony Optimization and Swarm Intelligence*; Dorigo, M., Birattari, M., Blum, C., Gambardella, L.M., Mondada, F., Stützle, T., Eds.; Springer: Berlin/Heidelberg, Germany, 2004; pp. 83–94.

60. Jung, S. Queen-bee evolution for genetic algorithms. *Electron. Lett.* **2003**, *39*, 575–576. [CrossRef]

61. Geem, Z.W.; Kim, J.H.; Loganathan, G.V. A New Heuristic Optimization Algorithm: Harmony Search. *Simulation* **2001**, *76*, 60–68. [CrossRef]

62. Kozá, J. Genetic programming as a means for programming computers by natural selection. *Stat. Comput.* **1994**, *4*, 87–112. [CrossRef]

63. Dorigo, M. Optimization, Learning and Natural Algorithms. Ph.D. Thesis, Politecnico di Milano, Milan, Italy, 1992.

64. Glover, F. Tabu Search—Part I. *Orsa J. Comput.* **1989**, *1*, 190–206. [CrossRef]

65. Holland, J.H. Genetic Algorithms and Adaptation. In *Adaptive Control of Ill-Defined Systems*; Selfridge, O.G., Risland, E.L., Arbib, M.A., Eds.; Springer: Boston, MA, USA, 1984; pp. 317–333.

66. Perianes-Rodríguez, A.; Waltman, L.; van Eck, N.J. Constructing bibliometric networks: A comparison between full and fractional counting. *J. Informetr.* **2016**, *10*, 1178–1195. [CrossRef]

67. Aria, M.; Misuraca, M.; Spano, M. Mapping the Evolution of Social Research and Data Science on 30 Years of Social Indicators Research. *Soc. Indic. Res. Rev. 2020*, *8*, 493–831. [CrossRef]

68. Grauwin, S.; Szell, M.; Sobolevsky, S.; Hövel, P.; Simini, F.; Vanhoof, M.; Smoreda, Z.; Barabasi, A.L.; Ratti, C. Identifying and modeling the structural discontinuities of human interactions. *Sci. Rep.* **2017**, *7*, 46677. [CrossRef] [PubMed]

69. Chen, C.; Ibekwe-Sanjuan, F.; Hou, J. The Structure and Dynamics of Co-Citation Clusters: A Multiple-Perspective Co-Citation Analysis. *J. Am. Soc. Inf. Sci. Technol.* **2010**, *61*, 1386–1409. [CrossRef]

70. Moher, D.; Liberati, A.; Tetzlaff, J.; Altman, D.G. The PRISMA Group (2009) Preferred Reporting Items for Systematic Reviews and Meta-Analyses: The PRISMA Statement. *PLoS Med.* **2009**, *6*, e1000097. [CrossRef] [PubMed]