Short thesis for the degree of Doctor of Philosophy (PhD) in informatics

Advanced Metaheuristics for Optimization

by: Anahita Sabagh Nejad
Supervisor: Dr. Gabor Fazekas

UNIVERSITY OF DEBRECEN
Doctoral School of Informatics
Debrecen, 2024

Contents

1 Introduction

1.1 Reason for Choosing This Subject for Ph.D. Thesis

There are so many optimization problems that are considered NP-hard because of their timing and complexity, and practically solving such problems is impossible especially when it comes to big data. When the size of the program increases, the responding time increases as well. The case that I worked about is called the Travelling Salesman Problems (TSP). This case is important because it can be used for solving or simulating many problems like Vehicle Routing Problem (VRP) which is sometimes used as a synonym for TSP.

1.2 Objectives of the Dissertation

The objective of this dissertation is to introduce a better or more advanced method to solve a TSP which is a mathematical problem. This work has connected some different knowledge to solve the TSP more easily. The knowledge of Mathematics with so many formulas, and data mining came together to be used as an Artificial Intelligence (AI) tool.
Our focus was on clustering (unsupervised learning) which is a data mining technique. Today's research works have more focus on classification (supervised methods), and working with an unsupervised technique was a challenge to find some results out of some datasets without any class label or output. We used the knowledge of Metaheuristic algorithms that are also interesting as it is proved that they can use some 'nature-inspired' definitions for solving mathematical problems.

1.3 Theme and Method

We used international specialized literature for the objective of this dissertation such as books, articles, conferences, online document sources related to this field, and publications on data mining, Optimization Problems, and Artificial Intelligence.
Despite the sources, we focused our attention on:
1. Showing the possible connections between the different areas of research and the specific problem,
2. Factors usually parameters influencing functions and the results,
3. Introducing different trends and methods to solve the same problems with the same parameter,
4. Some statistical data analysis to prove the effectiveness of our methods,
5. Using the standard test dataset in all of the algorithms to make the research a benchmark for further research.
6. Surveying on Artificial Flora Optimization Algorithm (AFO), Ant Colony Optimization (ACO), FireFly Algorithm (FA), Whale Optimization Algorithm (WOA)
7. Applying data mining Algorithms such as K- means, and Birch as two partitioning-based and hierarchy-based algorithms respectively to solve TSP.
8. Dividing the problems into smaller parts called clusters
9. Proceeding the collected data in the form of tables and figures
10. Concluding the method as a useful and applicable method that can be used to reduce the time and complexity of the problem and address further research.

The summarizations of the new advanced method are considered as a result of our research which can be observed in different tables and figures. In the next section, we summarize two of the used algorithms: Artificial Flora and Whale Optimization algorithms.

2 Algorithms Definitions

Meta-heuristic algorithms can be grouped as:
1) evolution-based: Like a Genetic algorithm that improves during evolution.
2) physics-based: These algorithms are based on the physical rules in the world, like SA (simulated annealing)
3) Swarm-based methods: These algorithms are based on the group behavior of animals: Like Particle Swarm Optimization (PSO) by Kennedy and Eberhart 1995, Artificial Bee Colony(ABC) by Dorigo et al. 2006, Ant Colony Optimization (ACO) 2006, etc.
4) Inspired by human behaviors like Tabu Search (TS).

2.1 Flora Optimization Algorithm

This algorithm which was mainly used in my first research and I presented at a conference, has some inspirations from nature like other swarm intelligence algorithms. The main concentration of this algorithm is based on the movement of the seeds and their offspring which can lead to the evolution, distinction, or creation of a new plant. Flora refers to an environment that has the same plant. As the offspring move to a new place, based on the characteristics of that environment (fitness) like climate, the seeds behave differently:
1) Sometimes, they can't adapt to that environment so they can't survive. They are distinct.
2) Some of them become adapted to the new environment and during the time they evolve, they become original plants by themselves for the new generations, and the cycle repeats, like spreading offspring.
3) Some of them become a new planet (rebirth).
The four important elements of this algorithm are:
1) Original Plant- They can spread the seeds in any place in the propagation Distance. It helps for the local search capacity of the FA
2) Offspring Plant (Seeds of the original plants)

3) Plant Location
4) Propagation Distance or spreading distance which means learning from the previous original plants and refers to how far a seed can go [1].
In spreading behavior, always the past generation movements are considered to update the solution. It helps the algorithm to avoid running into the local extremum [1].
The spreading is constant, so flora can migrate until they find the best answer (fittest area).
It can take the local optimal position as the center to explore around space so It can converge to optimal faster [1]. Spreading the seeds is done randomly but in a special radius.
The seed dispersal of the plants has two modes:
1) Autochory (plants that spread by themselves like mechanical propagation) [1]
2) Allochory (the plants spread through external forces like biological propagation, anemochory, and hydrochory) Allochory provides the conditions for plants to migrate to farther uncharted regions such that the direction and distance are determined as the wind changes [1], so by increasing the scope of exploration of flora, the possibility of extinction reduces.
The survival probability:
1) For a suitable environment: a plant survives and after being ripe spreads the seeds [1].
2) For harsh environments:
a) The evolve probability to adapt to the environment
b) The extinction of that flora in that region.
But before the distinction of flora, there is a probability of multiplication of flora in other areas, because the seeds may be moved to any new environment where the flora regains reproduction. As the propagation type is multi-generation, the flora has the chance of finding the best (optimal) growth environment [1].
The three main patterns in flora:
1) evolution behavior, which means the evolving probability to adopt
2) spreading behavior, which means movements of seeds: Autochory,

Allochory
3) select behavior, Which means survival or distinction of the flora
There are some specifications in the form of rules to explain the above-mentioned behaviors:
Rule 1: A species can be the primitive one in a new environment after a random distribution of the seeds. In the figure (1), original plants were distributed randomly in the area, as the $\diamond(x1)$.
Rule 2: If a plant adapts to the environment, should be evolved, and during this evolution, its hierarchy is based on the propagation distance of the last two nearest generations, so the hierarchy is not complete.
Rule 3: The range of the seed distribution area is considered a circle. The radius r of this circle is equal to the maximum propagation distance. Every place in this circle with a radius (r) and its circumference is for the distribution of the seed. In the figure (1), the seeds are distributed in the propagation distance. Distance1, Distance2, and Distance3 define the propagation distances, and the offspring is described as (a1,a2,a3) [1].
Rule 4: The probabilities of survival for the plants are different since the environmental factors for different areas are not the same. This probability is related to the plant that how well is fitted to the environment. When a plant is more fitted, the probability of survival becomes higher, but in some cases, some inter-specific competition can change this rule.
The solid line in the figure depicts a living plant and the dotted line depicts that the plant has died [1].
Rule 5: When a seed spreads to a very far environment, the probability of survival reduces, because the new environment has some properties like climate change that are possibly much more different than the environment of the original plant. As we have:

$$d \propto \frac{1}{p} \tag{2.1}$$

Based on this rule: fitness (a1) > fitness (a2) > fitness (a3), because of their distance from the original plants, the fittest one

is the closest one to the original plants. Because of competition in the figure (as an exceptional case), offspring a1 died even though it had a high fitness value, but a2 became an original plant that can spread its seeds. Rule 6: There is a boundary for seeds spreading, in a way that the distance of the seeds should not become more than the maximum limit. These limit areas define constraints. The details are as follows: In Figure 2.1, $\square(b1, b2, b3)$ are described as the new plants, and two of them survived as b1 and b3, but b2 did not. The selection between the two survived plants b1, and b3 is random, In the figure, b1 has been selected as the latest one. The distance 2 is for a2, and the distance 3 is for b1. Distance 2 is based on distance 1, and distance 3 is learning from the previous distances 1, and 2. If b1 which is selected randomly between b1 and b3, spread, the distances will be based on distance2, 3. If all the offspring plants die, as (c1,c2,c3), a new original plant can be generated randomly [1].

Evolution Behavior Based on the theory, the propagation distance is evolved from the parent and grandparent plants:[2]

$$d_j = d_{1j} \times rand(0,1) \times C_1 + d_{2j} \times rand(0,1) \times C_2 \qquad (2.2)$$

The new grandparent propagation distance is defined as [1]:

$$d_{1j}^{'} = d_{2j} \qquad (2.3)$$

The new parent propagation distance is the standard deviation between the positions of the original plant and the offspring plant[1].

$$d_{2j}^{'} = \sqrt{\frac{\sum_{i=1}^{N}(P_{i,j} - P_{i,j}^{'})^2}{N}}; \qquad (2.4)$$

Spreading Behavior The first generation of the plants is randomly such that N plants for N solution. P defines the position of the original plant in the form of a matrix $P_{i,j}$ (i refers to the dimension and j is the number of plants in the flora) where d is the maximum limit area [3].

$$P_{i,j} = rand(0,1) \times d \times 2 - d \qquad (2.5)$$

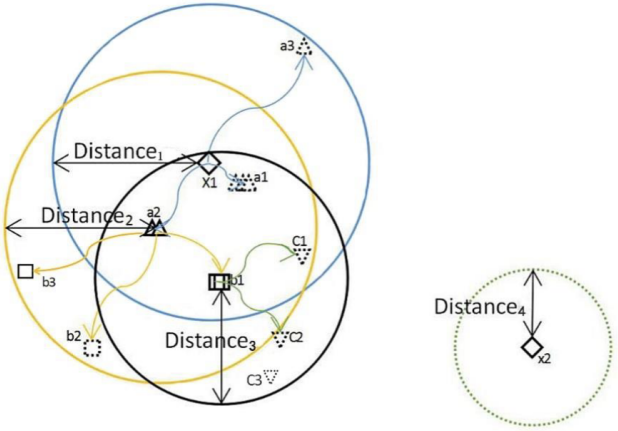

Figure 2.1: The Process of Migration and Reproduction of Flora seeds

[1]

d_j	propagation distance
d_{1j}	propagation distance of grandparent plant
d_{2j}	propagation distance of new parent plant
c1 and c2	Learning Coefficents
rand(0,1)	independent uniformly distributed number
$d_{1j}^{'}$	new grandparent propagation distance
$d_{2j}^{'}$	new parent propagation distance
P	Position
N	Number of solutions and Plants
d	maximum limit area
m	the number of seeds that one plant can propagate

Table 1: Parameters

[1]

The position of the offspring plant [1]:

$$P^{'}_{i,j\times m} = D_{i,j\times m} + P_{i,j} \tag{2.6}$$

where m is the number of seeds that one plant can propagate, we have [1]:
$P^{'}_{i,j\times m}$: The position of offspring plant
$P_{i,j}$: The position of the original plant
$D_{i,j\times m}$: a random number with the Gaussian distribution with mean 0 and variances j.

If no offspring plant survives, then a new original plant is generated according to:

$$P_{i,j} = rand(0,1) \times d \times 2 - d \tag{2.7}$$

Select Behavior
The survival probability is summarized as in below:

$$P = \left| \sqrt{\frac{F(P^{'}_{i,j\times m})}{F_{max}}} \right| \times Q_x^{(j\times m-1)} \tag{2.8}$$

Where our objective function is the fitness equation, we have:
Q_x: the selective probability
F_{max}: Maximum Fitness of Flora
$F(P^{'}_{i,j\times m})$: the Fitness of the jth answer
R: A random number between 0 and 1
P: Probability between 0 and 1

Deciding whether a plant survives or not is based on the proportion select method or roulette wheel method. Selection is based on score values and accepting probability. A higher score means greater probability.
r which is a random number in [0,1] becomes generated (uniform distribution) every time, and the offspring plant will be alive if the survival probability P is bigger than r ($P > r$), or it will die. N

offspring plants become selected among the alive offspring as new original plants and the above behaviors repeat until the accuracy requirement is reached or the maximum number of iterations is achieved [1].

The main steps of artificial flora are as follows [1]:
(1) Generation of N original plants based on equation (2.7).
(2) Calculate propagation distance according to equation (2.2), equation (2.3), and equation (2.4)[1].
(3) Generation of offspring plants by equation (2.6) and calculate the fitnesses;
(4) Calculate the survival probability of offspring using equation (2.8) and roulette wheel for selection.
(5) For the surviving plants, randomly selection of N new original plants. If they don't survive, generate new original plants using equation (2.7).
(6) If the new original plant's solutions are better than the previous one, then succeed them and save the best answer;
(7) Estimate whether the results are meeting the termination criteria or not. It can be decided by the accuracy requirement or the number of iterations. If so, then it is the optimal solution, Otherwise, the algorithm starts from step number 2;
Where M and N are: M: Maximum branching number (the number of seeds that one original plant can produce), and N: Number of original plants, the time complexity of this algorithm becomes O(NM).
This algorithm has been tested by benchmark functions like Sphere, Rosenbrock, Rastrigin, Schwefel, Griewank, and Ackley to check its accuracy and is compared for Particle Swarm Optimization (PSO) and Artificial Bee Colony (ABC) algorithms and different datasets. The results concluded that the AFO can have higher accuracy and stability. The reasons are:
1)Every time, It takes the surviving offspring plants as the new original plants, so the local optimal location becomes the center and it can explore the space around and converge faster.
2) As far as the direction of spreading the seeds and distances are

within a circle (the seeds can be taken to any part of this propagation environment), there is always a local search guarantee.
3) The algorithm can ignore the local optimum and focus on global searching, because based on the algorithm, when there is no better offspring, it randomly generates the new original plants.

2.2 Whale Optimization Algorithm(WOA)

Meta-heuristic optimization algorithm (namely, Whale Optimization Algorithm, WOA) mimics the hunting behavior of humpback whales [4].

Generally speaking, swarm-based algorithms have some advantages over evolution-based algorithms. For example, swarm-based algorithms preserve search space information over subsequent iterations while evolution-based algorithms discard any information as soon as a new population is formed [4].
They usually include fewer operators compared to evolutionary approaches (selection, crossover, mutation, elitism, etc.) and hence are easier to implement. Population-based meta-heuristic optimization algorithms share a common feature regardless of their nature[4]. The search process is divided into two phases: exploration and exploitation[4].
The optimizer must include operators to globally explore the search space: in this phase, movements (i.e. perturbation of design variables) should be randomized as much as possible. The exploitation phase follows the exploration phase and can be defined as the process of investigating in detail the promising area(s) of the search space[4]. Exploitation hence pertains to the local search capability in the promising regions of design space found in the exploration phase. Finding a proper balance between exploration and exploitation is the most challenging task in the development of any meta-heuristic algorithm due to the stochastic nature of the optimization process[4].

Three types of mathematical models are proposed by the author of the algorithm (Mirjalili, et all):
1) Encircling prey,
2) Spiral bubble-net feeding maneuver,
3) Search for prey
In the first model, which is called encircling prey, since whales want to hunt prey, they need to encircle around it, and as far as they don't know the optimal location, they consider the current candidate solution(answer) to be the target prey, and later update (change) it by finding the best search agent. Later, the other search agents (whales) will change their location based on the best whale found so far. In this model, we have these definitions and four formulas:
$\overrightarrow{D}$: Distance
$\overrightarrow{D'}$: the best distance so far
$\overrightarrow{A}$ and $\overrightarrow{C}$: Coefficient Vectors defined in formula (2.11) and (2.12)
$\overrightarrow{t}$: Current Iterations
$\overrightarrow{X}(t)$: Position vector
$\overrightarrow{X}(t+1)$: The next position vector
$\overrightarrow{X^*}(t)$: Best answer or location so far that will be updated later
$\overrightarrow{a}$: A decreasing number from 2 to 0 (in both phases: exploration and exploitation)
$\overrightarrow{r}$: Random number [0,1]

$$\overrightarrow{D} = \left| \overrightarrow{C}.X^*(t) - \overrightarrow{X}(t) \right| \tag{2.9}$$

$$\overrightarrow{X}(t+1) = \overrightarrow{X}^*(t) - \overrightarrow{A} \cdot \overrightarrow{D} \tag{2.10}$$

Coefficient Vectors definitions in the form of mathematics [4]:

$$\overrightarrow{A} = 2\overrightarrow{a} \cdot \overrightarrow{r} - \overrightarrow{a} \tag{2.11}$$

$$\overrightarrow{C} = 2 \cdot \overrightarrow{r} \tag{2.12}$$

In the second model, which is called the Bubble-net attacking method' (exploitation phase), we have another strategy that whales

use to attack prey. This Strategy has two methods: 1) Shrinking encircling mechanism, and 2) Spiral updating position [5] which we define below:
In 1), we reduce the value of vector $\overrightarrow{a}$ in the formula. This way, the value of vector $\overrightarrow{A}$ which is in the interval $\overrightarrow{A} = [-\overrightarrow{a}, \overrightarrow{a}]$ changes.

When we set a random value (different values) for $\overrightarrow{A}$= [-1,1] over different iterations, we can calculate the position of the new whale, between the position of the original search agent and the current best whale [6]. For 2 dimensional data $\overrightarrow{A}$ can be: $0 \leq \overrightarrow{A} \leq 1$.

In 2), we have the below formula where $\overrightarrow{A}$ is the best distance between the ith whale and prey, and b is a constant defining the shape of the logarithmic spiral, and l[-1, 1] is defined as a random value [7]. In the spiral updating position, we have a helix shape for whale(X, Y), and current best agent (X^*, Y^*) [7]. The below formula is defined for the helix shape:

$$\overrightarrow{X}(t+1) = \overrightarrow{D'}.e^{bl}.cos(2\pi l) + \overrightarrow{X^*}(t) \tag{2.13}$$

Where [8]:

$$\overrightarrow{D'} = \left|\overrightarrow{X}^*(T) - \overrightarrow{X}(t)\right| \tag{2.14}$$

indicates the distance of the i_{th}, b is a constant for logarithmic shape, l $[-1, 1]$ randomly chosen. As far as whales swim in two approaches simultaneously, we suppose that there is a probability p [0, 1] of 50 percent to choose between them, such that if $p < 0.5$, we use the equation (2.10), otherwise, we use equation (2.13) [7].

In the third model, searching for prey (exploration): we assign a random value for the vector -1$\leq A \leq$ 1.
$Arandomsearchagentischosenwhen$ $\left|\overrightarrow{A}\right| > 1$. Where $\overrightarrow{X}_{rand}$ is a random whale position in the form of a vector, we have:

$$\overrightarrow{D} = \left|\overrightarrow{C}.\overrightarrow{X}_{rand} - X\right| \tag{2.15}$$

for updating the position of the search agents:

$$\overrightarrow{X}(t+1) = \overrightarrow{X}_{rand} - \overrightarrow{A}.\overrightarrow{D} \tag{2.16}$$

The steps of the Whale Optimization Algorithm are [4]:

Step 1: Initialize the whale's population Xi (i = 1, 2, ..., n) Calculate the fitness of each whale

X_*=the best search agent

Step 2:

while (t < maximum number of iterations)

for each search agent Update a, A, C, l, and p

if1 (p<0.5)

if2 ($|A| < 1$)

Step 3: Update the position of the current search agent by equation (2.10) $\overrightarrow{X}(t+1) = \overrightarrow{X}^*(t) - \overrightarrow{A} \cdot \overrightarrow{D}$

else if2 ($|A| \geq 1$)

Step 4: Select a random search agent

Step 5: Update the position of the current search agent by equation (2.15): $\overrightarrow{X+1} = \overrightarrow{X}_{rand} - \overrightarrow{A} \cdot \overrightarrow{D}$

end if2 elseif1 ($p \geq 0.5$)

Step 6: Update the position of the current search by equation (2.13): $\overrightarrow{D'}.e^{bl}.cos(2\pi l) + \overrightarrow{X^*}(t)$ end if1
end for

Step 7: Check if any search agent goes beyond the search space and amend it

Calculate the fitness of each whale

Update X* if there is a better solution
t=t+1
end while
return X*

3 Methods Used in the Conference and Articles with Results

In the course of our first research that we presented at the Eszterhazy University of Eger in 2020 in the form of a conference [9], we tried to map the data mining aspect of a metaheuristic using the Artificial Flora Optimization Algorithm by analyzing our data with some classification measurements. We tried to map the methods and the used concepts of these algorithms concerning k-means. We specified three algorithms, that way, the readers can find the best match for the aim of their survey in character and size. The sample size or the number of our population N in the program was 50. The used algorithms were Artificial Flora, Bee colony, and Ant colony (combined with Genetics).
The method employs the Flora seeds(offsprings)to search for the set of cluster centers that minimize the distance. The newly proposed method used k-means for clustering data and our dataset was the iris dataset from TSPLIB [10]. We compared the results with the two other swarm intelligence algorithms (Ant colony and Bee colony). In the summarization of the class results, Accuracy was considered as the benchmark of the algorithms.
In the course of our first article [11] that was published in the International Journal of Artificial Intelligence (IJ-AI) which is a Q2 scopus index Indonesian Journal, we explained the whale optimization algorithm which was discovered in 2016, and we tried to introduce a clustering method to reduce the timing and improve the cost function. In all of our calculations, the average value of fitness and timing were applied as the benchmark [11].
In this article, Mainly, we focus on some mathematical formulas with their definitions and we explain the differences between the clusterized and unclusterized methods for three different standard datasets. In the end, we concluded that the clusterized method with k- means is a better method than an unclustered algorithm.
In the course of literature gathering, we primarily processed publi-

cations that describe the whale optimization algorithm, and some relevant state of the art from the newly published articles. We used 42 references upon the request of the journal editor.
Depending on the characteristics of the whale algorithm– we drew the attention of the readers to future works to hybridize the same algorithm with other clustering methods.
In the course of our second article which was published in the same journal [12], we expanded our previous research work. We hybridized the whale algorithm with Birch which is a Hierarchy-based algorithm and applied k- k-means in the third stage. The idea was to introduce a faster TSP solver. The results obtained from the tables prove that the new combined algorithm with Birch has empowered the algorithm and our algorithm converges to the optimal solution much faster than the unclustered whale algorithm.

For both articles and a conference paper, the results are depicted in the form of tables and figures to be more understandable for the reader. These results are obtained during 20 iterations.

3.1 Results of the two articles

WOA Fitness	Eil51	Linhp318	Rl1323
Avg	1176.421	393928.2	8262213
Max	1262.614	409489.4	8740947
Min	1100,269	376614.9	8174234.322
Stdev.s	49.53605	8468.606	49003.98
WOA Timing	Eil51	Linhp318	Rl1323
Avg	3.765785	16.63365	63.50511
Max	4.7562	20.79	79.21
Min	3.1421	13.642	51.669
Stdev.s	0.396958	1.77977	6.286887

Table 2: Unclustered Whale Optimization Algorithm for Solving TSP Problem

[11]

3.1.1 The clustering Method for whale algorithm by k-means

Here for the TSP problem, the clustering method is proposed. Based on my article: First, the method that we used to solve the TSP problem was the WOA algorithm, later we used k-means to divide our problem into smaller parts and solve these small parts [11].
Algorithm: K-means,
Inputs:
K: cluster numbers or the initial centroids,
D: a set of n objects,
Output: K clusters,
Method:
(1) arbitrarily choose K objects from D,
(2) repeat,
(3) according to the average of the objects within the cluster, (re)assign the objects to the closest one
(4) update by calculating the average of the objects for each cluster, and new assignments
(5) until no change happens in the clusters [11].
The steps of our algorithm are as follows [11]:
Step 1: Initialize the number of the population as shown in Table 1
Step 2: Specify K based on the (11)
Step 3: Applying the Kmeans algorithm
Step 4: Applying whale optimization algorithm for i=1: K
Step 5: Find the position of the cities
Step 6: Sorting by indexing
Step 7: Find the nodes (cities) in each cluster that are closer to the centroid of that cluster
Step 8: Join the closest cities to another cluster
Step 9: Stopping criterion till no cluster remains unjoined.
The next table presents the fitness and the timing of the best tour for the clusterized method. The timing has improved as the fitness.

WOA+K Fitness	Eil51	Linhp318	Rl1323
Avg	489.8785	82908.2	1.19E+06
Max	526.76	88040	1.23E+06
Min	454,9	77273	1.14E+06
Stdev.s	21.2760405	3180.155	2.78E+04
WOA+K Timing	Eil51	Linhp318	Rl1323
Avg	1.2205855	5.707375	24.81385
Max	1.4997	7.953	55.258
Min	1.0021	4.6754	20.345
Stdev.s	0.1708681	0.979334	7.647444

Table 3: Clusterized Whale Optimization Algorithm for Solving TSP Problem

3.1.2 The clustering Method for whale algorithm by Birch Algorithm

In another method [12], we clustered the dataset with BIRCH (Balance Iterative Reducing and Clustering using Hierarchies) and in the last step, we applied k-means to solve the TSP problem. [12]
Step1: Set the parameters (see table.) for 20 iterations and 100 population numbers:
a: decreasing number from 2 to 0 coefficient such that A= [-a, a]
r: random number [0,1]
X: the whale's position
X*: The best solution
P: random number [0, 1] used for probability
b: a constant for logarithmic shape
l: a value in [-1,1]
D: distance
N: city's length
Step 2: Specify K randomly or using the below equation for big tours among the cities, and the Branching factor:

$$Br - Factor = \frac{N}{5}$$

$$k = \sqrt{(\frac{N}{10})}$$

Step 3: Applying the Birch algorithm to cluster our data
Step 4: Applying the WOA algorithm for all of the found clusters i =1:K
Step 5: Find the location of the Agents
Step 6: Sorting by indexing
Step 7: Joining the clusters by finding the cities that are closer to the centroid of that cluster
Step 8: Repeat till joining all the clusters
Here is the summarization of our research:

WBirch Fitness	Ali535	Rat783	dsj1000
Avg	35903.55	1.3169e+05	5.2681e+08
Max	36400,23	1.1434e+05	5.3129e+08
Min	35763,49	1.2895e+05	5.2170e+08
Stdev.s	318.74	1.4941e+03	2.2455e+06
WBirch Timing	Ali535	Rat783	dsj1000
Avg	4.15	5.18	6.66
Max	4.29	5.26	6.82
Min	4.11	5.15	6.62
Stdev.s	0.04	0.02	0.06

Table 4: Clusterized Whale Optimization Algorithm with Birch algorithm for Solving TSP Problem

4 New Proposed Methods

Solving an NP-hard problem was the main subject matter. We want to modify the introduced methods used in the main two articles to improve fitness function in an acceptable fitness and time interval. We introduce two close methods and the third one becomes a practice for future works. The modified algorithms are tested over 318 to 1323 cities. The results of these two modified methods are close, but they can find more optimized clusters compared to the previous algorithms to achieve the most optimized solutions.

4.1 Thesis 1: The First New Proposed Method

In this method, after the initial clustering, the algorithm checks the number of cities. If the number exceeds the specified number (threshold), it applies clustering for that cluster again and applies the whale algorithm and k-means to find the shortest path. For example, we have five clusters(as in figure 5.4.1), and in each cluster, we have 3 or 2 cities, but in the first cluster, the number of cities is 9 (suppose 6 is a threshold), so, We cluster that again. After the second clustering, three subsets remain (a, b, c), and the total number of clusters becomes seven. This is an example of a possible sequence: a, b, c, Cluster 2, Cluster 3, Cluster 4, Cluster 5. It means After splitting cluster 1 into three clusters, cluster 1 won't exist anymore and we have a,b, and c as three individual clusters. The algorithm joins these six clusters together using the k-means and the Whale algorithm to find the optimal path. These seven clusters will be connected and the algorithm will check the path again. In each iteration, we will have new connections between these clusters, because the subset clusters will join to different outer clusters (cluster 2, cluster 3, Cluster 4, Cluster 5) to find the best path. This method repeats till meeting the stopping criteria which is finding the most optimized path. This method is called **the type one Cluster-Thresholding Method (CTM1)**. Figure 4.1 shows a small dataset but in reality, we can see clusters with 1323 cities and many subsets(sub-clusters).

The steps of CTM1 are as follows:
1-Parameter Settings (as in the previous section), T= Threshold (maximum number of cities)= 20, $k = \sqrt{\frac{N}{2}}$, iteration=20, Population size=100, a=2;
2- Initial Clustering;
3- Checking the number of the cities,

For found clusters $C_i = 1 : K$,

If the Number of the cities >T, Split that cluster;
4- Apply the proposed K-means and WOA algorithm (in my first article);
5-Find the position of the newly found **inner clusters(a, b, c)**;
6- Sorting by indexing;
7- Join inner clusters to outer Clusters;
8- Repeat the previous steps until you find the optimal path or meet the stopping criteria like till no cluster remains un-joined;

CTM 1 Fitness	Linhp318	Pr1002	Rl1323
Average	5.8023e+04	3.2570e+05	4.3395e+05
Max	5.9915e+04	3.6205e+05	4.5737e+05
Min	5.5231e+04	2.9857e+05	4.1532e+05
Stdev.s	1525.77	17340.43	11909.58
CTM 1 Timing	Linhp318	Pr1002	Rl1323
Average	2.20668	7.04759	9.911245
Max	2.4493	7.1882	10.0954
Min	2.0761	6.8703	9.7484
Stdev.s	0.08	0.07	0.09

Table 5: The Fitness and Timing of the First New Method for Solving the TSP Problem

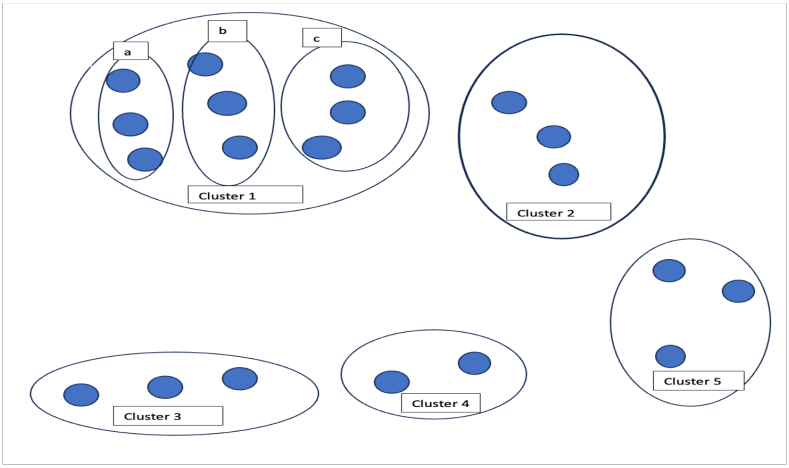

Figure 4.1: Model used to explain CTM1 and CTM2

4.2 Thesis 2: The Second New Proposed Method

The first method and the second method are so close, but the main difference between them is that in the first model, the subset clusters could join to all the outer clusters (like cluster2 to cluster5) using the method of k-means and WOA, but in the second approach, the subset clusters can just join inside cluster1 (in the example). This method is called **the type two Cluster-Thresholding Method (CTM2)**.

Suppose we have a set of k clusters, and their size is defined by the number of cities. We assign a value for the threshold(T) which is the maximum number of cities for each cluster.

If the number of cities exceeds T, our algorithm divides that cluster into smaller clusters. Then, the Whale algorithm and k- means start to look for the shortest path in each cluster. After sorting and indexing the cities, the final results(clusters) or solutions of each cluster must be connected. The final result will be the optimized

path. If we have 9 cities (like in the previous example) and our Threshold is equal to 6, we cluster that again, and finally, we join the subset clusters together. This way the subset clusters become optimized and they can be connected inside of their main cluster(here, cluster1). In the previous figure, the algorithm splits cluster 1 into 3 subset clusters. Then, the three subsets (a, b, c) can be connected in a way that cluster 1 becomes optimized, like (a,b,c), (a,c,b), (b, c, a). The result of cluster 1 which is an optimized path would become connected to the other 4 clusters (cluster 2 to cluster 5). In this approach, cluster 1 still exists and one possible solution for this method can be the sequence of Cluster 1, Cluster 2, Cluster 3, Cluster 4, Cluster 5. It means we don't consider a, b, and c as individual clusters. We use them to optimize cluster 1 (inside optimization). They help the algorithm to find an optimal path but they don't have any connection with Cluster 2, Cluster 3, Cluster 4, and Cluster 5, individually, and that's Cluster 1(as an optimized cluster) that connects to the outer cluster. This way, the number of connections will be reduced as well.
These two algorithms are suitable for big data to solve an optimization problem (here TSP). In case we want to use them for a small dataset, the threshold value should be a small number because if we assign the threshold (T=50) for a dataset like Eil51, we will only have just one cluster. For dataset Linhp318 we will have only 6 clusters, etc.

The steps of CTM2 are as follows:
1-Parameter Settings (as in the previous section), T= Threshold (maximum number of cities, here is n=20), Iteration=20, population size=100, a=2, $K = \sqrt{\frac{N}{2}}$,
2- Initial Clustering
3- Checking the number of the cities,

For found clusters $C_i = 1 : k$,

If the Number of the cities >T, Split that cluster;
4- Apply the proposed K-means and WOA Algorithm;

5-Find the position of the newly found **inner clusters(a, b, c)** and Connect them in an optimized path,
6- Sorting by indexing;
7- Join Cluster 1 to outer clusters;
8- Repeat the previous steps until you find the optimal path or meet the stopping criteria like till no cluster remains un-joined;

These two methods have close results, but they can find better answers compared to the simple version of the WOA algorithm for solving TSP.

CTM2 Fitness	Linhp318	Pr1002	Rl1323
Average	5.87518e+04	3.47833e+05	4.49069e+05
Max	6.2708e+04	3.6535e+05	4.7162e+05
Min	5.2073e+04	3.3138e+05	4.3298e+05
Stdev.s	2.4841e+03	8.079e +03	9.968e+03
CTM2 Timing	Linhp318	Pr1002	Rl1323
Average	2.181765	7.32871	10.146345
Max	2.2754	7.6146	10.4448
Min	2.0723	7.1189	10.0152
Stdev.s	0.06	0.12	0.10

Table 6: The Fitness and Timing of the Second New Method for Solving the TSP Problem

Minimum Fitness	Linhp318	Pr1002	Rl1323
WOA- TSP	376614.9	3.9822e+06	8174234.322
WOA- TSP K-means	77273	6.7806e+05	1.14E+06
WOA- Birch TSP	3.1516e+05	3.2389e+06	9.1221e+06
CTM1	5.5231e+04	2.9857e+05	4.1532e+05
CTM2	5.2073e+04	3.3138e+05	4.3298e+05

Table 7: Comparison Between the Fitness of the Five Methods for TSP

Minimum Timing	Linhp318	Pr1002	Rl1323
WOA- TSP	13.642	16.1910	51.669
WOA- TSP- K-means	4.6754	5.2557	20.345
WOA-Birch-TSP	2,9233	6.46	8.3500
CTM1	2.0761	6.8703	9.7484
CTM2	2.0723	7.1189	10.0152

Table 8: Comparison Between the Timing of the Five Methods for TSP

5 Future Works

The third method will be based on using the environment. As we know, Big Data needs big scale, so for these kinds of data, we need some more scalable algorithms. Our dataset can be divided into a pre-defined number, not based on similarity. We can apply k-means and WOA for each cluster. Then we combine the clusters. In the end, we have different sequences for the clusters that must be connected, and one of them which offers the shortest path will be the best. In Figure 5.5.1, we suppose that we have 4 clusters. For each cluster, we can apply K-means and WOA separately to find the best path. In the end, we join the clusters. We have different ways to connect them that is k!. For this example: 4!= 24. Some of the possible sequences are: $\{1-2-3-4\}$, $\{1-2-4-3\}$, $\{1-3-2-4\}$,

$\{1-4-3-2\}$, and etc. One of these three ways of connecting the clusters will be the best path. We select the first-founded best path among the clusters.

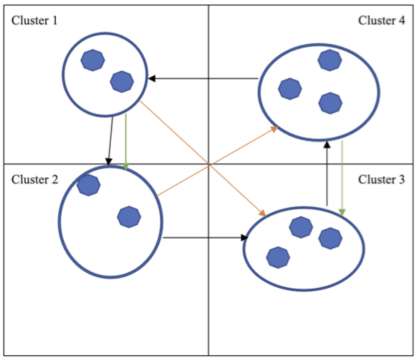

Figure 5.1: The Third New Proposed Method of Solving the TSP Problem as Future Works

6 Conclusion

Table 5 compares the fitness, and Table 6 compares the timing for all the discussed methods. We can conclude that the two newfound methods have presented better results than the other methods as they decide the problem to a smaller problem to reduce the complexity and they assign a T value(Threshold) to limit the number of cities that can be placed in a cluster. This way, our combined algorithm which is the whale algorithm and K-means can be applied and solve the problem. Deciding about the way that the clusters can be combined is also a method to optimize the clusters and reduce their complexity. These two methods are defined as two approaches for combining the

optimized clusters. In these methods, k-means is applied to cluster data. For further research, the algorithm can be combined with other clustering methods or other optimization techniques as the Flora algorithm, Firefly, and Ant colony among the most famous ones.

Minimum Fitness	Linhp318	Pr1002	Rl1323
WOA	376614.9	3.9822e+06	8174234.322
K-means	77273	6.7806e+05	1.14E+06
WOA-Birch	3.1516e+05	3.2389e+06	9.1221e+06
New Method 1	5.5231e+04	2.9857e+05	4.1532e+05
New Method 2	5.2073e+04	3.3138e+05	4.3298e+05

Table 9: Comparison Between the Fitness of the Five Methods

Minimum Timing	Linhp318	Pr1002	Rl1323
WOA	13.642	16.1910	51.669
K-means	4.6754	5.2557	20.345
WOA-Birch	2,9233	6.46	8.3500
New Method 1	2.0761	6.8703	9.7484
New method 2	2.0723	7.1189	10.0152

Table 10: Comparison Between the Timing of the Five Methods

7 Figures

The next figure shows an example of applying the whale algorithm for TSP in the clustered approaches:

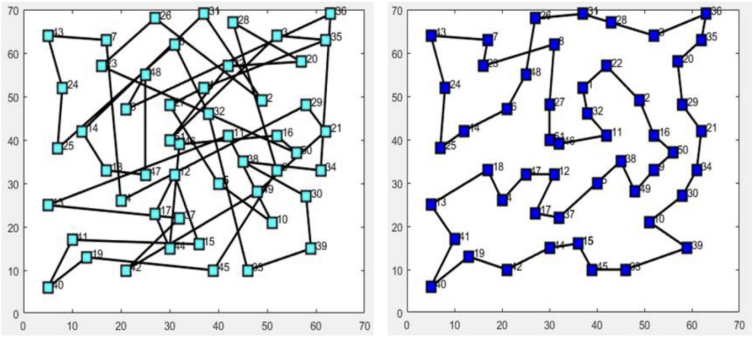

Figure 7.1: a) unclustered approach b) clustered approach with K-means [11]

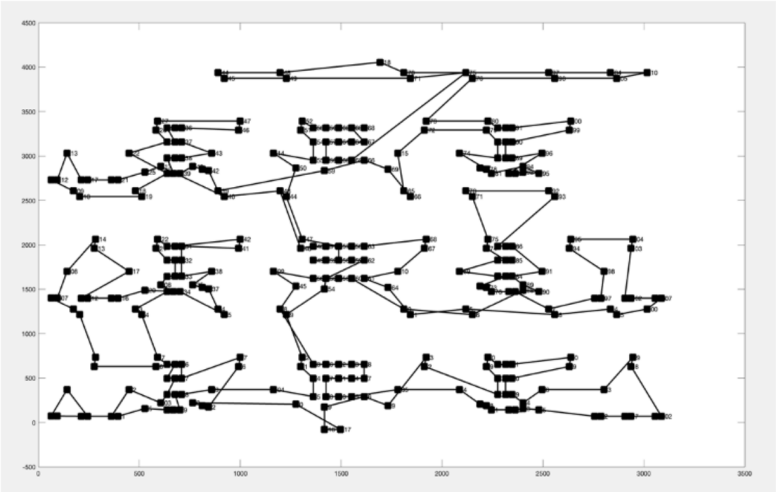

Figure 7.2: Applying CTM1 for Solving TSP

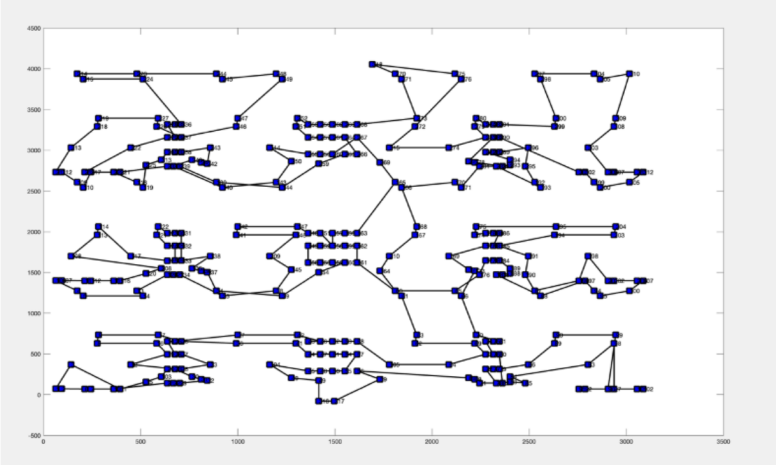

Figure 7.3: Applying CTM2 for Solving TSP

References

[1] Cheng, Long, Xue-han Wu, and Yan Wang. "Artificial flora (AF) optimization algorithm." Applied Sciences 8.3 (2018): 329.

[2] Misra, R. et al. (2023) in Machine learning and big data analytics 2nd international conference on machine learning and big data analytics-ICMLBDA, IIT Patna, India, March 2022. Cham: Springer International Publishing.

[3] Sreejith, S., Khanna H. Nehemiah, and A. Kannan. "A Framework to Classify Clinical Data Using a Genetic Algorithm and Artificial Flora-Optimized Neural Network." International Journal of Swarm Intelligence Research (IJSIR) 13.1 (2022): 1-22.

[4] Mirjalili, Seyedali, and Andrew Lewis. "The whale optimization algorithm." Advances in engineering software 95 (2016): 51-67.

[5] Tubishat, Mohammad, et al. "Improved whale optimization algorithm for feature selection in Arabic sentiment analysis." Applied Intelligence 49 (2019): 1688-1707.

[6] Hu, Hongping, Yanping Bai, and Ting Xu. "A whale optimization algorithm with inertia weight." WSEAS Trans. Comput 15.8 (2016).

[7] Jin, Haiyan, et al. "Multi-sensor image fusion based on contrast and directional features optimization." International Journal of Distributed Sensor Networks 14.12 (2018): 1550147718815841.

[8] Hassanien, Aboul Ella, and Diego Alberto Oliva, eds. Advances in soft computing and machine learning in image processing. Vol. 730. Springer, 2017.

[9] Sabagh Nejad, A.: Data Clustering Using Hybrid Algorithm. In: The 11th International Conference on Applied Informatics (ICAI 2020). Ed.: Fazekas Istvan, Kovasznai Gergely, Eszterhazy Karoly Egyetem, Eger, 1-8, 2020.

[10] S. George, “Mp-testdata—the tsplib symmetric traveling salesman problem instances.” 2008.

[11] Nejad, Anahita Sabagh, and Gabor Fazekas. "Solving a traveling salesman problem using meta-heuristics." IAES International Journal of Artificial Intelligence (IJ-AI) 11.1 (2022): 41.

[12] Sabagh Nejad, Anahita, and Gábor Fazekas. "Reducing the time needed to solve a traveling salesman problem by clustering with a Hierarchy-based algorithm."

DEBRECENI EGYETEM

DEBRECENI EGYETEM
EGYETEMI ÉS NEMZETI KÖNYVTÁR
H-4002 Debrecen, Egyetem tér 1, Pf.: 400
Tel.: 52/410-443, e-mail: publikaciok@lib.unideb.hu

Nyilvántartási szám: DEENK/490/2023.PL
Tárgy: PhD Publikációs Lista

Jelölt: Sabagh Nejad, Anahita
Doktori Iskola: Informatikai Tudományok Doktori Iskola
MTMT azonosító: 10090185

A PhD értekezés alapjául szolgáló közlemények

Idegen nyelvű tudományos közlemények külföldi folyóiratban (2)

1. **Sabagh Nejad, A.**, Fazekas, G.: Reducing the time needed to solve a traveling salesman problem by clustering with a Hierarchy-based algorithm.
   IAES Int J Artif Intell. 12 (4), 1619-1627, 2023. ISSN: 2089-4872.
   DOI: http://dx.doi.org/10.11591/ijai.v12.i4.pp1619-1627
2. **Sabagh Nejad, A.**, Fazekas, G.: Solving a traveling salesman problem using meta-heuristics.
   IAES Int J Artif Intell. 11 (1), 41-49, 2022. ISSN: 2089-4872.
   DOI: http://dx.doi.org/10.11591/ijai.v11.i1.pp41-49

Idegen nyelvű konferencia közlemények (1)

3. **Sabagh Nejad, A.**: Data Clustering Using Hybrid Algorithms.
   In: The 11th International Conference on Applied Informatics (ICAI 2020). Ed.: Fazekas István, Kovásznai Gergely, Eszterházy Károly Egyetem, Eger, 1-8, 2020.

A DEENK a Jelölt által az iDEa Tudóstérbe feltöltött adatok bibliográfiai és tudománymetriai ellenőrzését a tudományos adatbázisok és a Journal Citation Reports Impact Factor lista alapján elvégezte.

Debrecen, 2023.11.02.

DEBRECENI EGYETEM Egyetemi és Nemzeti Könyvtár