Tikrit Journal of Administrative and Economic Sciences, (31/12/2022); Vol. 18, No. 60, Part (1): 635-652
Doi: www.doi.org/10.25130/tjaes.18.60.1.35

Tikrit Journal of Administrativ And Economics Sciences
مجلة تكريت للعلوم الإدارية والاقتصادية
ISSN: 1813-1719 (Print)

A hybrid algorithm to bypass some weak features in the particle swarm and firefly algorithms...with an application to the traveling salesman problem

Ruqayah Yassir Abdul Ameer[a] *, Muna Shaker Salman[b], Athraa Kamel Al-Mashhadani[c]

[a]Imam al-Kadhum college (IKC),Department of Banking and Financial Sciences. [b]Technical Institute- Suwaira, Middle Technical University. [c]Administrative Technical College, Middle Technical University College of Administration and Economics, University of Al-Hamdaniya.

Keywords:

Traveling Salesman Problem; Hybrid; Particle Swarm Optimization; Firefly Algorithm.

ARTICLE INFO

Article history:

Received 06 Aug. 2022
Accepted 14 Sep. 2022
Available online 24 Feb. 2023

*Corresponding author:

Ruqayah Yassir Abdul Ameer
College of Administration and Economics, University of Al-Hamdaniya.

Abstract: In this paper, the Traveling Salesman Problem (TSP) is solved through the use of some approximation techniques where the results of the previous work showed some defects in solving the problem to obtain an optimal or close to optimal solution,so the use of hybrid algorithms to solve some results from the use of intuitive and exact algorithms. A hybrid algorithm has been proposed that combines the characteristics of the firefly algorithm (FA) and Particle Swarm Optimization (PSO) to obtain an algorithm that works effectively in overcoming some of the problems resulting from the use of each algorithm separately. Then using an improvement factor to improve each solution within the resulting community and to obtain solutions with a high diversity. The efficiency of the proposed method was measured by solving some standard problems TSP, and the results showed a high convergence of the algorithm towards the known optimal solution for each problem by solving 13 standard problems.

Doi: www.doi.org/10.25130/tjaes.18.60.1.35

خوارزمية هجينه لتجاوز بعض الميزات الضعيفة في خوارزميتي اسراب الجسيمات واليراع... مع تطبيق على مشكلة البائع المتجول

رقية ياسر عبد الامير
كلية الامام الكاظم (ع)
قسم العلوم المالية والمصرفية

منى شاكر سلمان
المعهد التقني الصويرة
الجامعة التقنية الوسطى

عذراء كامل هرموش
الكلية التقنية الإدارية
الجامعة التقنية الوسطى

المستخلص

في هذا البحث تم حل مسألة البائع المتجول من خلال استعمال بعض التقنيات التقريبية حيث اظهر نتائج الاعمال السابقة بعض العيوب في حل المسألة للحصول على حل أمثل او قريب من الأمثل لذلك يتم استخدام خوارزميات الهجينة لحل بعض الضعف في النتائج من استخدام هذا السلوب من الخوارزميات الحدسية والمضبوطة.

تم اقتراح خوارزمية هجينة تجمع صفات خوارزمية اليراعات مع خوارزمية امثليه السرب للحصول على خوارزميه تعمل بشكل فعال في التغلب على بعض المشاكل الناتجة عن استعمال كل خوارزمية على حده. ومن ثم استعمال عامل تحسين، لتحسين كل حل داخل المجتمع الناتج والحصول على حلول ذات تنويع عالي. تم قياس كفاءة الطريقة المقترحة من خلال حل بعض المسائل القياسية الخاصة بالمسألة واظهرت النتائج تقارب عالي للخوارزمية نحو الحل الأمثل المعروف لكل مسألة من خلال حل 13 مسألة قياسية.

الكلمات الدالة: مشكلة البائع المتجول، تحسين سرب الطيور الهجينة، خوارزمية اليرعات.

1. Introduction:

The TSP considered an NP-hard problem. TSP is a tough problem, thus unless we're content with an approximate result, the calculations will take a very long time.Thus, several problems are to be studied individually and set special algorithms to find out their solutions (Brezina Jr, I., 2011: 12). The TSP defines as a person or a group of people, he or them wants to visit n from areas starting from a specific area, and the n-1 will visit the other area without passing through any of them more than once except for the base from which he set off (Kumbharana, S. N., & Pandey, G. M., 2013: 1), means that he will return to it at the end of his journey. Each possible trip is a switch of 1, 2, 3, ... n, where n is the number of towns, thus the number of trips is n, where n becomes large and it is impossible to find the cost of each trip in polynomial time. This approach will terminate with providing the optimum solution. It is not very possible due to the time exhaustion required to calculate all trips. The purpose of studying this problem lies in finding out the minimum cost, time, and distance (Zhang, D., et al., (2021: 11). It can also be mentioned that the most common realistic application for TSP is a uniform allocation of goods or supplies, determining the shortest way to serve customers and planning bus lines 'MTSP. (Pang, W. et, al., 2004: 10), (Gotoh, J. Y., et al., 2018: 440) stated some of them as an explanations(Shi, X.H. et al., 2007: 160), such as burrowing holes for electrical circuits for surface tools, positioning satellite of a starlight interferometer, implementations in crystallography, optimizing the series diagram, planning CPU motherboard composition, manufacturing robots control.

The only currently known way to ensure the optimal solution to TSP of any volume is to enumerate every possible trip and search for the trip at the minimum cost. It can be seen that even in small cases the time consumption is very high if we need to obtain every possible trip. Alternatively, an approximate algorithm can be efficiently used to reduce the time yielding a result that is not necessarily the best trip.

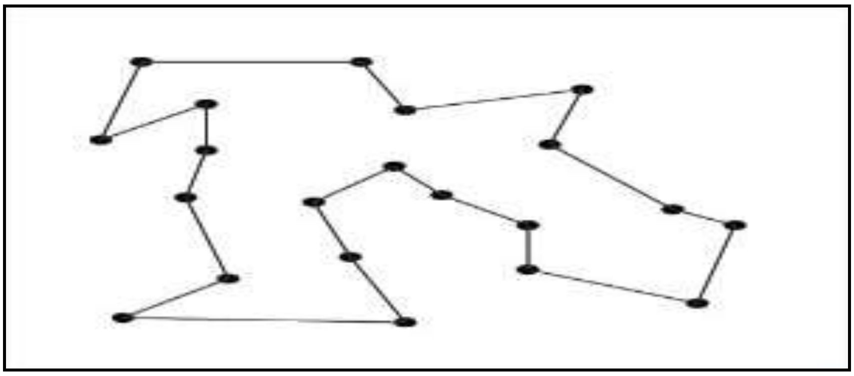


Figure (1): the (TSP) path

Source: TSPLIB test library

However, instead of a tour close to the best (Kumbharana, S. N., & Pandey, G. M., 2013: 7). This paper focuses on studying the TSP or the traveler problem. The problem of circular tourism and the methods used to solve it.Therefore, we present a hybrid method that combines two optimization methods, Firefly (FA) algorithm and Particle Swarm Optimization (PSO) to effectively optimize the advantage of getting the best path (round) that can be reached which considers the best possible solution (Haroun, 2010:2).

2. Contributions: The contributions of this paper are summarized as follows:

- We propose a hybrid algorithm called "PSO+FA" that combines the ideas of the PSO and the FA for effectively getting a near-optimal solution and an optimal result to solve the TSP problem.
- We validate the proposed algorithm with extensive experiments/ simulations.

3. Literature Review: This section reviews previous algorithms on TSP. They are an opulent group of literature on finding optimal solutions for the

TSP. In this paper, we focus on the studies of the Firefly Algorithm (FA) algorithm and (PSO). (EF. G et al., 2008: 3), have presented their idea PSO as an effective approach to PSO to TSP. They Goldberg and de Souza showed in 2008 a paper on developing PSO algorithms for TSP where computational experiments were presented with cases up to 7397 towns (EF.G. et al., 2008: 5), (Goldbarg et al., 2008: 2). Similarly, two algorithms proposed by (Udaiyakumar, 2014:2) was applied to find the 'minimum' of 'Maxspan' for the 25 datasets and to 'control' the coefficients of the algorithm from the 'absorption coefficient, the level of 'randomness', the value of' attractiveness, the number of' fireflies', and iterations'. Similarly, (SaraeiI M. et, al., 2015: 3), have presented their paper Firefly Algorithm FA Solving Multiple TSP in which a mathematical model was created to solve the TSP, in which they proved that a standard algorithm to solve equal- tasks problems. In 2015 the researcher Bakhayt used the MOTSP algorithm to speed up and standardize the transmission mechanism used in the State Company for Grain Processing.). (Kora, 2016) have used the Fireflies and Particle swarm Optimization technology in FFPSO to detect heart disease, where the two methods were combined and integrated in a hybrid method that combines them to improve the feature of Electrocardiogram ECG. The outcomes of these tests demonstration that the suggested method yields high-quality results, when matched with four effective extension methods that have been adopted by researchers specifically designed for the problem studied (Shibeeb, A. K., & Ahmed, 2020:6).It is able to compare three high-quality methods to reach the best possible solutions (Zhang et al., 2021:12). However, the proposed algorithms do not take into consideration some factors, for example finding an optimal path with minimum time, cost, and distance.

4. **Problem Formulation of TSP** (Cai & Sun, 2021: 344)**:** Suppose that C is the matrix of shortest distances (dimension n x n), where n is the number of nodes of the graph G. The elements of the matrix represent C Shortest distances between all pairs of nodes (i, j), i, j = 1, 2, …, n. The traveling salesman problem can be formulated in a binary programming class, where the variables are 0 or 1,

Depending on whether the path from node I to node j is true ($x_{ij} = 1$) or not ($x_{it} = 0$). So, the mathematical formula of TSP (Brezina Jr, I., 2011: 10) is the following (the idea of this formula is to assign numbers from 1 to n to the nodes using additional variables u_i, so that this numbering corresponds to the order of the node in the rounds. The mathematical model for TSP is an allocation model that excludes sub-paths, so a mathematical model can be formulated for the problem as follows:

$$Minimize = \sum_{i=1}^{n}\sum_{j=1}^{m} C_{ij}X_{ij}, For\ all\ i = j \quad .... \quad \textbf{(1)}$$

Subject to

$$\sum_{i=1}^{n} X_{ij} = 1, j = 1,2,3, \dots, n\ ,\ i \neq j \quad .... \quad \textbf{(2)}$$

$$\sum_{j=1}^{m} X_{ij} = 1, i = 1,2,3, \dots, n\ ,\ i \neq j \quad .... \quad \textbf{(3)}$$

$$u_i - u_j + nX_{ij} \geq n - 1, for\ i \neq j\ ,\ i,j = 2,3,4, \dots, n.. \quad \textbf{(4)}$$

$$X_{ij} \in \{0,1\} \quad \forall \boldsymbol{i,j} \in \boldsymbol{n} \quad .... \quad \textbf{(5)}$$

Equation (1) expresses the target function that reduces the total distance travelled. Constraint (2) indicates that the tour visits node j only once. Constraint (3) indicates that the tour leaves each node i only once, while restriction (4) cancels all incomplete tours (Marinakis, Y., & Marinaki, M, (2010): 430). The constraint (5) is a binary entry. Let C_{ij} represents the travel distance from town i to town j, X_{ij} is a decision variable and take the following two states (EF.Gouvêa et al., 2008: 201)

- ❖ As for X_{ij} =1, it means that the optimal solution obtained uses the road connecting the two cities (i and j).
- ❖ $X_{ij} = 0$ means that the optimal solution obtained does not use the road connecting the two cities (i and j).
- ❖ Constraints (1), (2), (3) are considered an ordinary allocation model, while restriction (4) indicates the best solution, so it must be taken into account when resolving the path. N represents the sum of the nodes, ui of the additional variables. The mathematical formula for TSP is to assign numbers 1 ton for nodes using additional variables ui. Represents the location in the path to city (i). Thus, this numbering matches to the rank of the nodes in the round.

5. Metaheuristic Algorithm

5-1. Particle Swarm Optimization Algorithm: The algorithms are inspired by the behavior of particles swarms and fish in moving from one place to another. These animals move instinctively to search for food or migrate. The algorithm includes two primary processes are as follows: (1) the process of Exploration, (2) the Exploitation process about the best solutions available within the specified search area. The PSO algorithm is used to solve problems related to optimization, changing over time, and cost. Due to the characteristics that this algorithm carries out to find solutions to these problems. An individual in a swarm is affected by three main factors as follows: (Mijbas et al., 2020:1), (Elizabeth et al., 2006: 102).

- ❖ Attraction to the position of the leader, which is considered in the algorithm the best solution.
- ❖ Attraction to the best position he went through and storing it in the memory.
- ❖ Continue in his current status.

The Algorithm of PSO consists of swarm population called particles and being symbolized as N = (N_1, N_2, N_3, … N_i), which move inside the swarm depends on the type of multi-dimensional problem and search for better initial solutions. The movement of particles relies on their own experience and the experiences of neighboring particles. The PSO algorithm has been configured from several randomly generated swarm particles. When the algorithm is initialized, the swarm particles depend on the velocity particle which consists of $V_i^t= (V_1^t, V_2^t,\ldots,V_i^t)$. The position of the particle position consists of $X_i^t= (X_1^t, X_2^t,\ldots,X_i^t)$. Where all updated based on the previous cases of the best position of the particle itself and it is symbolized as $G_{\mathbf{best}}$ (the best local position). The best position for the particle in whole swarm, symbolized by $G_{\mathbf{best}}$ (best global position) according to the dimensions of the problem d, which consists of d = (d_1, d_2, d_3, ... d_j). In the PSO algorithm each particle has associated with the location and velocity, associated value, best position that can be reached, neighbors, values, and best location. The movement of particles from one position to another, for each particle the velocity and position are preserved according to the updated equations: (Mijbas et al., 2020: 2)

$$X_i^{t+1} = X_i^t + V_i^{t+1} \qquad \ldots \textbf{(6)}$$

$$V_i^{t+1} = V_i^t + C_1 r_1^t\left(P_{best,i}^t - X_i^t\right) + C_2 r_2^t\left(G_{best,i}^t - X_i^t\right) \quad \ldots \textbf{(7)}$$

Where V_i^t is a velocity for particle i in the swarm with dimension j and in iteration t. X_i^t is the position of the i particle in the swarm with dimension j and at iteration t. C_1 and C_2 is the acceleration coefficients constant (C_1 is the cognitive component) and ($C_2$is the social component),

r_1^t and r_2^t refers to random numbers that are distributed according to the uniform distribution within the period (0,1), t is the number of recurrences determined by type of the problem, $P(best, i)^t$ is the best position of the i particle for itself and is called the best local position, and $G (best, i)^t$ is the best position for the particles i in the swarm completely and is called the best global position.

5.1.1. Parameters of PSO: There are basic parameters of the PSO that affect its operation and it has a strong effect on the performance efficiency of the PSO algorithm (Mijbas et al., 2020: 3), (Udaiyakumar, & Chandrasekaran, 2014: 1790), These parameters help to optimize the search process in the problem area and the parameters are as follows:

- Swarm size: It represents the number for swarms' particles and is proportional to number of the iterations.
- Number of iterations: It represents the number of iterations of the algorithm, determined according to the size of the problem.

inertia weight (W) is defined as a parameter that is added to the algorithm, where the weight factor makes the tendency of the body to resume moving in the same path it was travelling in previously, and it was added to optimize the performance and efficiency of the algorithm. The equation of updating the particle velocity (11)

$$V_i^{t+1} = WV_i^t + C_1 r_1^t (P_{best,i}^t - X_i^t) + C_2 r_2^t (G_{best,i}^t - X_i^t) \dots \quad \textbf{(8)}$$

$$w^{t+1} = wmax - \left(\frac{wmax - wmin}{T\,max}\right) t, wmax > wmin \quad \dots. \quad \textbf{(9)}$$

Where the value of inertia weight W is calculated according to the equation 8.

wmax: Is the maximum value for the inertia weight

wmin: Is the minimum value for inertia weight

t: Is the number of iterations specified for the problem.

T max: Is the maximum value for the specified number of iterations.

The constriction parameter: Is the new parameter of updating the velocity of the object and is symbolized by the symbol K. The reason for using the constriction coefficient is to control the exploration process of particles inside the swarm and to ensure convergence between the particles. Equation (6) is to be replaced by equations (10) and (11).

$$V_i^{t+1} = K[V_i^t + C_1 r_1^t (P_{best,i}^t - X_i^t) + C_2 r_2^t (G_{best,i}^t - X_i^t)] \dots \quad \textbf{(10)}$$

Initialize position X_{ij}^{t}, C1, C2, V_{ij}^{t}

P=max. no of particles, N= max.no of iterations, D= max. no of dimensions. K is calculated according to the following equation:

$$K = \frac{2}{|2-\alpha-\sqrt{\alpha^2-4\alpha}|} \quad \alpha > 4 \qquad \ldots \qquad \textbf{(11)}$$

$$\alpha = C_1 + C_2 \qquad \ldots \qquad \textbf{(12)}$$

$$\alpha_1 = C_1 r_1 \qquad \ldots \qquad \textbf{(13)}$$

$$\alpha_2 = C_2 r_2 \qquad \ldots \qquad \textbf{(14)}$$

4.1.2 Steps of PSO Algorithm

The essential steps of the PSO is shown in Figure (2):

Tikrit Journal of Administrative and Economic Sciences, (31/12/2022); Vol. 18, No. 60, Part (1): 635-652
Doi: www.doi.org/10.25130/tjaes.18.60.1.35

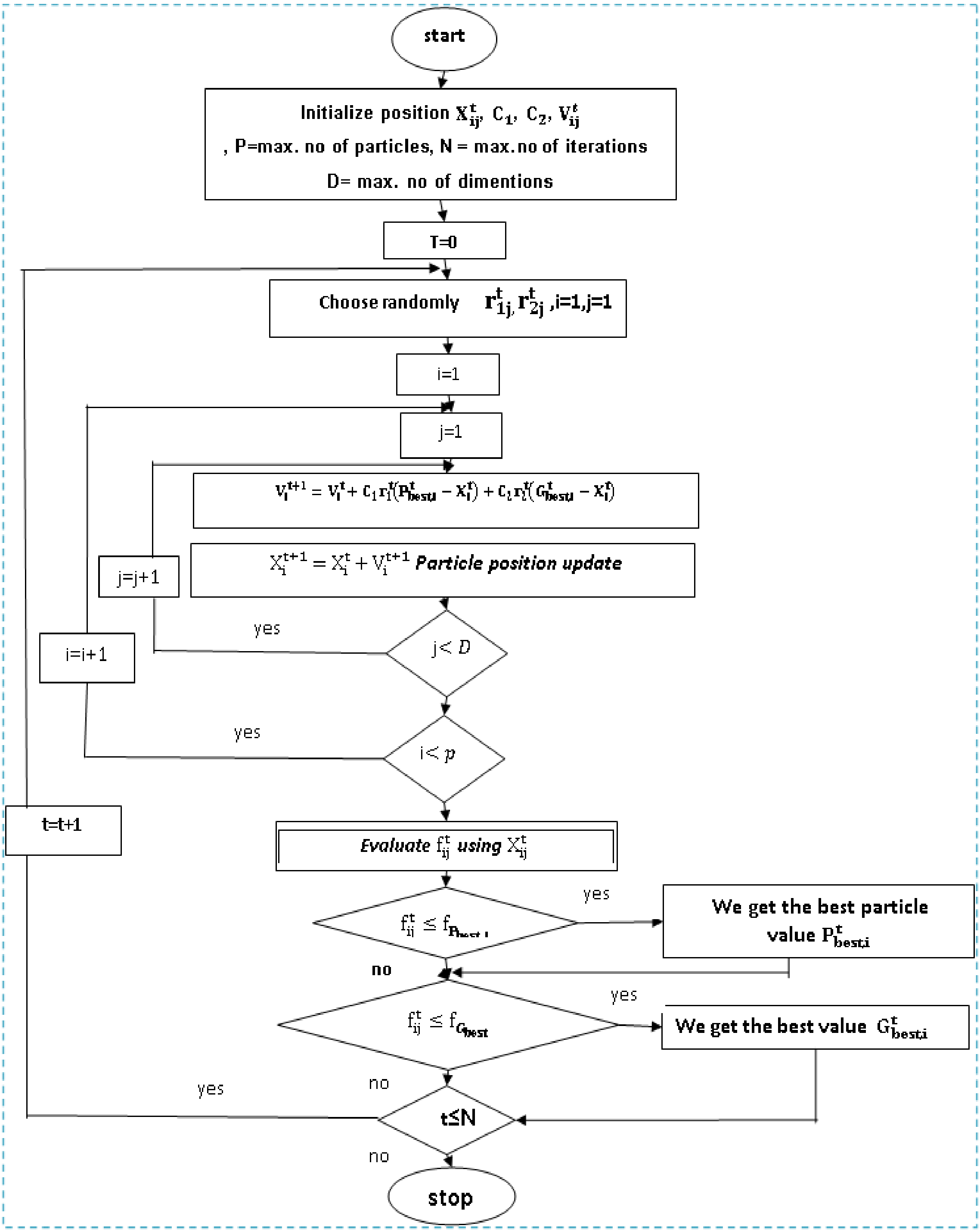


Figure (2): The Steps of PSO

Source: Prepared by the researcher depending on (Talukder, 2011: 44)

5.2. Firefly Algorithm (FA) (Kora & Krishna, 2016:11): The firefly, or luminous mosquito is a primary solution for some research problems due to its capability to simulate the best ingredients in nature. So, it is taken into consideration one of the algorithms stimulated by nature and is deemed one of the utmost important and contemporary algorithms (EF. Gouvêa G.et, al.(2008):101). This algorithm is divided into three sections: (body intelligence, developmental algorithms, and bacterial forage algorithms). Body intelligence technology is based on taking a group or a swarm of fireflies or swarms that inspire dialogue to obtain potential results for real world problems, the algorithm has given suitable results so we will take the firefly algorithm, which is considered one of the most recent algorithms.

The Firefly Algorithm (FA) is counted one of the modern algorithms which uses of randomness in the search for possible solutions as a result of the light produced by the firefly. The algorithms of the fireflies are based on changing the light intensity and on maintaining gravity, meaning that it can always rely on gravity by determining the intensity of the brilliance, which in role determines the objective function in the situation of maximizing the optimization problems, that is, the gravity (I) of the firefly of the site (X) is inversely proportional to the objective function $I(X) \propto F(X)$ that is, it is visual to the other levers. The light density with the distance is inverse, with the observation of light absorption by the effective medium around the firefly. Thus, gravity allows to vary the degree of light absorption

5.2.1. Basic Rules for the FA: (Yang, 2013: 172) (Kumbharana & Pandey, 2013: 22)

FA depend on basic rules are as follows:

- Fireflies consist of two sexes and move around other fireflies in search of the most attractive and bright firefly, as the female responds to the male by giving continuous flashes.
- The distance between fireflies and others is an important factor, as the greater the distance is the less attractive the firefly. Consequently, it searches for other more attractive fireflies at random.
- • The brightness of the firefly is determined from the objective function since the problems of maximization of the surfaces are directly related to the objective function value.

5.2.2. The Attractiveness of FA: FA depends on the difference in light intensity (brightness) and attraction. Distance of the light and analysis by air. It produces firefly visional only over a limited space. Firefly can be deemed a point light resource. It is familiar that the density of light at a certain distance r from the light source follows the inverse square law. This law states that the density of light decreases with increasing distance r. As mentioned, air disperses feebler and feebler light with growing distance. In the easiest case, the light density can be deemed as a mark sourer by analyzing the parameter, at a distance r as equation (16) (I is the light density in r =0 (saraei et al., 2015: 268), (Yang, 2013: 52).

$$I \propto \frac{1}{r} \quad \dots\dots\dots. \textbf{(15)}$$

$$I(r) = I_0 e^{-y^{r^2}} \quad \dots\dots\dots. \textbf{(16)}$$

For formulating fireflies attractiveness, each firefly has its attraction,, which is described by tune, while r is a function of the distance among fireflies. Since attractiveness of a firefly is proportional to the density of light seen by a nearby)

$$\beta(r) = \beta_0 e^{-y^m} \quad m \geq 1 \quad \dots\dots\dots \textbf{(17)}$$

Whereas

$I_{0:}$ is the greatest intensity of light at r = 0 .

β0: is the greatest attractiveness when r = 0 .

y: is the absorption factor that controls the reduction in the light density, since the degree of light absorption usually ranges between (0,1).

$$\beta(r) = \beta_0 e^{-y^{r^2}} \quad \dots\dots\dots \textbf{(18)}$$

The space between the firefly i and the firefly j is as in the following equation:

$$r_{ij} = \|x_i - x_j\| = \sqrt{\sum_{k=1}^{d}(x_{i,k} - x_{j,k})^2} \quad \dots\dots\dots(\textbf{19})$$

$$x_i = (x_{i1}, x_{i2}, \dots x_{id})$$

Whereas

$x_{(i,k)}$: is k th of the special format.

X_i: is for i th of fireflies.

d: number of dimensions.

The movement of fireflies is determined according to the following equation:

$$X_i^{t+1} = X_i^t + \beta_0 e^{-yij^2}(X_j - X_i) + \alpha \left(rand - \frac{1}{2}\right) \quad \dots \quad \textbf{(20)}$$

The first part of equation No. (20) Indicates to the current location of firefly i, while the second part indicates to the attractiveness of the firefly. The third and last parts indicates to the random move if there is no brightness in

the firefly and rand is the random number generated in a uniform distribution with a range of (0,1)) In the greatest case ℵin(0, 1), β = 1.
The diagram shown in Figure (3)

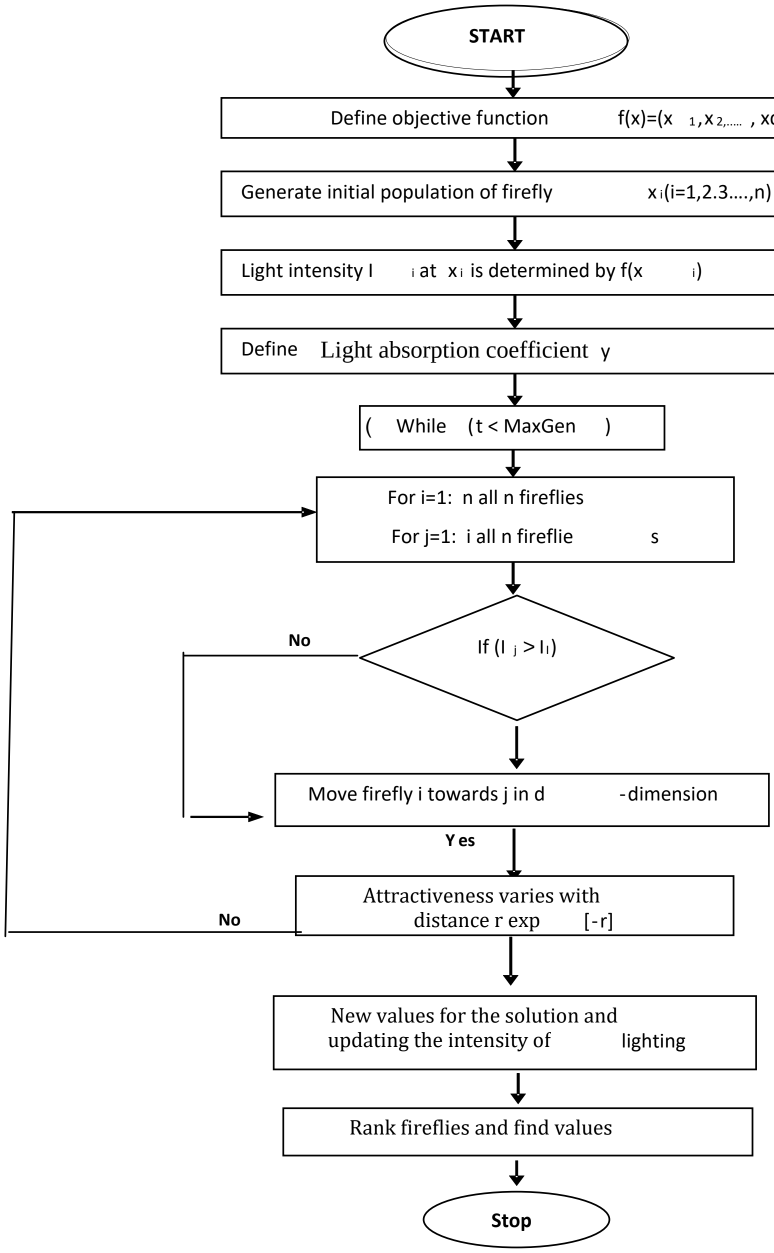


Figure 3: Diagram of the Fireflies Algorithm
Source: Prepared by the researcher depending on the (Saraei et al., 2015: 269)

After assessing the first generation, the FA enters the major loop, which illustrates the greatest number for fireflies generation for each generation. The fireflies whose light density is strong are chosen, meaning that the result with the best value of objective function is selected and the possible optimal solution is chosen. The FA simulates a parallel operational for n of fireflies population generates n of results.

5.3. Hybrid Algorithm for Particle Swarm Algorithm with the Fireflies Algorithm (PSO+FA): (Tien-S. 2015: 4)

The particle swarm algorithm is one of the common algorithms in solving many complex optimization problems and is considered useful to some extent in solving many medium and low complexity problems, but it is relatively unable to solve problems with family complexity. This shortcoming is due to the algorithm's weakness in the speed of convergence to the optimal solution due to poor exploitation.

Many researchers resort to solving such a problem by adding auxiliary techniques that address this shortcoming and to obtain, as much as possible, a technique that combines the mechanism of diversification and approach.

The firefly algorithm is one of the modern technologies in the world of optimization and has the ability to solve many highly complex problems. In this paper, the particle swarm algorithm is combined with the Firefly algorithm to handle the little convergence in the particle swarm algorithm. The hybrid optimization algorithm is designed using the Firefly (PSOFA) algorithm founded on the original PSO and FA. (EF.Gouvêa, 2008:98) Each algorithm develops autonomously, that is, PSO has its individuals and best solution to replenish the worst artificial fireflies in FA.

Meanwhile, the best artificial fireflies for FA are to replace the poorer or less attractive and efficient individuals in PSO after doing several steady iterations. The entire frequency includes the number for communications R, where $3R_1$, $2R_1$, $R = R_1$. Suppose N is the population number of the hybrid PSOFA, and N_1, N_2 is the population number of PSO and FA correspondingly. If clients with top k fitness in N_1 will be copied to N_2 to substitute the same number of individuals with the worst fitness, where t indicates the current iteration count, the R_1 and k are predefined. The stages of communication between the two PSO-FA algorithms can be illustrated as follows:

- **Initialization**: Generate populations for both PSO and FA independently. Described the iteration set R for implementing the communication. The N_1

Doi: www.doi.org/10.25130/tjaes.18.60.1.35

and N_2 is the number of particles and artificial agents in solutions S_{ij}^T and X_{ij}^T

- **Assessment:** Assess the value of (f_1) and (f_2) to both PSO and FA in each community. Individual development is carried out independently by both algorithms PSO and FA.
- **Updating:** Update the velocity and location of the PSO applying Equations (5) and (6). Update FA's position and velocity with the best value the fireflies found applying equations (18), (19).
- **Communications:** Emigrate best artificial fireflies among all individuals of the FA population, copy k fireflies with the best attractiveness or k fitness in N_1 and replace the poorest particles in N_2 from the population of PSO and update for each group in each R_1 iteration.
- **Termination:** We repeat step2 to step5 till the predetermined the function value is reached or the maximum number for iterations is accessed. We record best of the value for function $f_{(St)}$ and the best particle position of all the particles. We record best of the value for function f $_{(Xt)}$ and the best position of all the fireflies X^t.

6. **Results:** This section highlights the experimental results for proposed hybrid approach. In this paper, we used Matlab language to design and implement the hybrid algorthim for solving the TSP problem. Moreover, several sets of examples in the TSPLIB test library are selected for the simulation analysis. The experiments run on processor intel Core i5, 4 G.B memory, Microsoft visual studio platform.

6.1 Expermentil results: The experiments on the Oliver problem 30 with the exact running time (0.2 seconds) are conducted. The results of the traditional method and the implementation of various improvement strategies are obtained respectively as shown in Table 1.

6.2 Discussion

Figures (4): shows that the hybrid algorithm can better converge to the optimal in a shorter time as a solution. The following sets of examples in the internationally universal TSPLIB test library are used for simulation analysis.(1)

Set the population size to 20. The experimental results are shown in Table 1, we can observe that when the population size is small, the hybrid algorithm requires a shorter running time to obtain the optimum solution, and the optimum result is higher than that in the literature; although the firfly algorithm is shorter A solution can be found within time, but it is simple to fall to a local optimal (such as pr76). Through the simulation of the program, the optimal route of the Hybrid algorithm to solve some examples is shown in Figures 4,5,6,7,8.

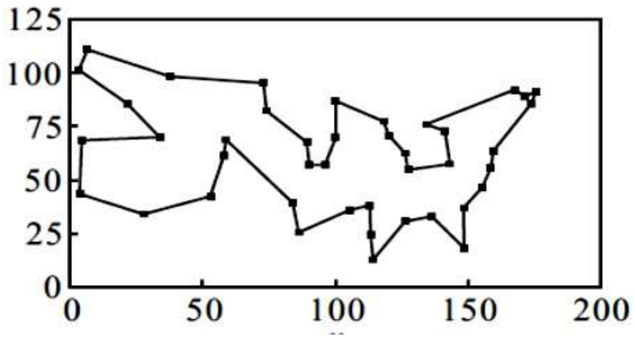


Figure (4): The dantzig-42 optimal road map

Source: TSPLIB test library.

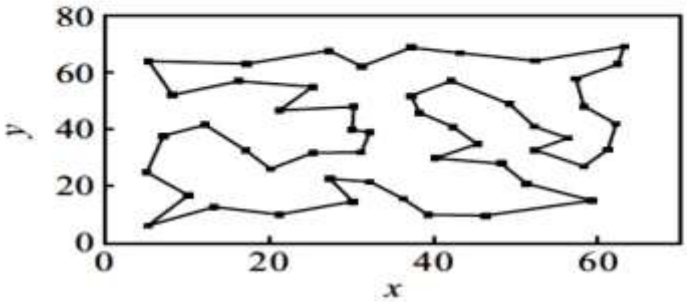


Figure (5): The optimal route map of eil51

Source: TSPLIB test library

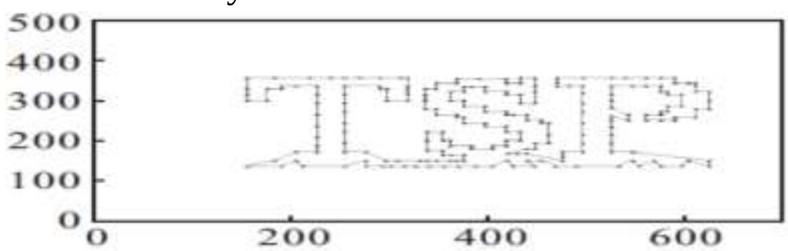


Figure (6): The optimal route map of tsp255

Source: TSPLIB test library.

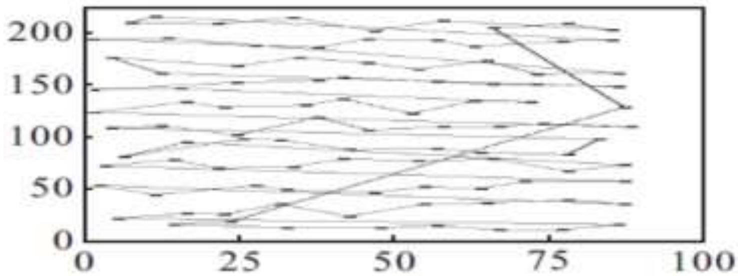


Figure (7): The optimal route map of rat99

Source: TSPLIB test library.

Table 1: demonstrates that the introduction of the firefly algorithm to generate the initial population is able to speed up the optimization of the initial stage of evolution. Through experimental data, the hybrid firefly algorithm can find the optimal solution in a shorter time.

Table (1): Comparative analysis of experimental results

Instance	BKS	FA			PSO			HFA		
		Obj.	time	GAP	Obj.	time	GAP	Obj.	time	GAP
berlin52	7542	8228	4.468	0.091	8231	4.478	0.091	7609	3.858	0.009
dantzig42	676	684	1.171	0.012	700	1.179	0.036	676	0.467	0.000
eil51	426	442	5.625	0.038	449	5.668	0.054	436	5.056	0.023
eil76	538	552	10.781	0.026	553	10.830	0.028	547	10.788	0.017
eil101	629	661	7.745	0.051	676	7.754	0.075	637	12.297	0.013
lin105	14379	37445	9.522	1.604	37447	9.558	1.604	14462	6.809	0.006
st70	675	700	4.467	0.037	716	4.486	0.061	688	2.721	0.019
pr76	108159	119315	4.381	0.103	119320	4.409	0.103	108873	10.577	0.007
pr107	44303	47954	14.599	0.082	47977	14.607	0.083	44478	49.381	0.004
rat99	1211	1291	3.41	0.066	1306	3.450	0.078	1251	6.031	0.033
rat195	2323	2482	20.578	0.068	2499	20.619	0.076	2379	19.862	0.024
tsp255	3916	4246	62.881	0.084	4256	62.971	0.087	3939	56.493	0.006
Oliver30	423.74	472.22	1.187	0.114	501.22	1.195	0.183	423.74	0.065	0.000

Source: Prepared by the researcher based on the Matlab program

Where

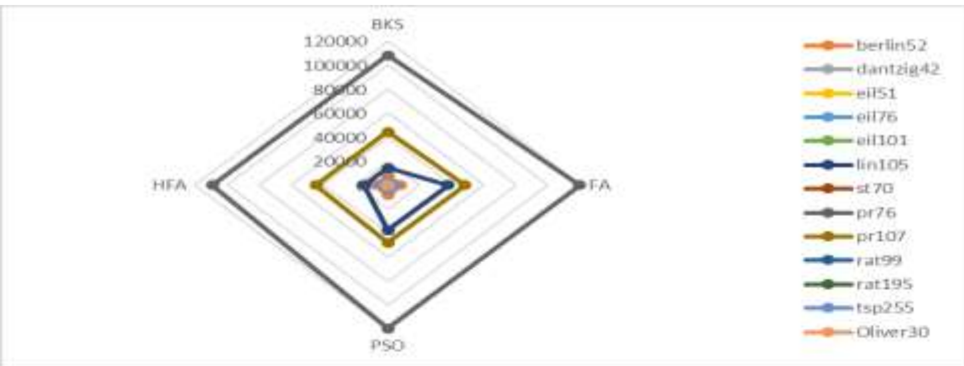
Source: Prepared by the researcher depending on the Matlab program.

$$GAP = \frac{Obj - BKS}{BKS} \qquad \dots \dots \dots \qquad (21)$$

BKS: best-known solution found

Obj.: objective function of the solution found by algorithms

Figure (8):



comparing results between algorithms

Source: Prepared by the researcher depending on the Matlab program

7. **Conclusion:** In this research, we propose a hybrid firefly algorithm that gathers the thoughts of PSO and FA to resolve the typical discrete optimization TSP problem. For combinatorial optimization problems that have no good solutions at present, it is easy to modify this algorithm. Solved. This new system of imitating natural organisms has a bright future, more In-depth and detailed work needs to be further developed. In future work, we will examine new model with convergence and theoretical basis and will prepare additional developments on the hybrid algorithm to be able to provide effective solutions for the TSP.

References

1. Shibeeb, A. K., & Ahmed, M. H., (2019, May), Use of a new approach to automated break transposition cipher system. In IOP Conference Series: Materials Science and Engineering (Vol. 518, No. 5, p. 052020). IOP Publishing.
2. Mijbas, A. F., Hasan, B. A. A., & Salah, H. A., (2020), Optimal stabilizer PID parameters tuned by chaotic particle swarm optimization for damping low frequency oscillations (LFO) for single machine infinite bus system (SMIB). Journal of Electrical Engineering & Technology, 15(4), 1577-1584.
3. Brezina Jr, I., & Čičková, Z., (2011), Solving the travelling salesman problem using the ant colony optimization. Management Information Systems, 6(4), 10-14.
4. Cai, Z., & Sun, D., (2021), Bayesian analysis for quantile smoothing spline. Statistical Theory and Related Fields, 5(4), 346-364.
5. Udaiyakumar, K. C., & Chandrasekaran, M., (2014), Application of firefly algorithm in job shop scheduling problem for minimization of makespan. Procedia Engineering, 97, 1798-1807.
6. EF.Gouvêa G.,M. C.Goldbarg,G. R. d. Souza, (2008), Particle Swarm Optimization Algorithm for the Traveling Salesman Problem" Federico Greco, ISBN 978-953-7619-10-7, pp. 202, September 2008,DOI:10.5772/5580 In book: Traveling Salesman Problem Project: Experimental Algorithms Authors:
7. Elizabeth F. G. Goldbarg, Marco C. Goldbarg and Givanaldo R. de Souza, (2008), Particle Swarm Optimization Algorithm for the Traveling Salesman Problem, Federico Greco (Ed.), ISBN: 978-953-7619-10-7.
8. Elizabeth F. Gouvła Goldbarg, Givanaldo R. de Souza, and Marco Csar Goldbarg, (2006), Particle Swarm for the Traveling Salesman Problem", J. Gottlieb and G.R. Raidl (Eds.): EvoCOP 2006, LNCS 3906, pp. 99 110, 2006.
9. Gotoh, J. Y., Kim, M. J., & Lim, A. E., (2018), Robust empirical optimization is almost the same as mean–variance optimization. Operations research letters, 46(4), 448-452.

10. Haroun A., (2010), Traveling Salesman Problem and Improved Algorithm for its Solution", Tishreen University Journal for Research and Scientific Studies-Basic Sciences Series Vol. (32) No. (3).
11. Li, M., Ma, J., Zhang, Y., Zhou, H., & Liu, J., (2015), Firefly algorithm solving multiple traveling salesman problem. Journal of Computational and Theoretical Nanoscience, 12(7), 1277-1281.
12. Kora, P., & Krishna, K. S. R., (2016), Hybrid firefly and particle swarm optimization algorithm for the detection of bundle branch block. International Journal of the Cardiovascular Academy, 2(1), 44-48.
13. Kumbharana, S. N., & Pandey, G. M., (2013), Solving travelling salesman problem using firefly algorithm. International Journal for Research in science & advanced Technologies, 2(2), 53-57.
14. Saraei, M., Analouei, R., & Mansouri, P., (2015), Solving of travelling salesman problem using firefly algorithm with greedy approach. Fen Bilimleri Dergisi (CFD), 36(6).
15. Talukder, S., (2011), Mathematical modelling and applications of particle swarm optimization.
16. Shi, X. H., Liang, Y. C., Lee, H. P., Lu, C., & Wang, Q. X., (2007), Particle swarm optimization-based algorithms for TSP and generalized TSP. Information processing letters, 103(5), 169-176.
17. Pan, T. S., Dao, T. K., & Chu, S. C., (2015), Hybrid particle swarm optimization with bat algorithm. In Genetic and evolutionary computing (pp. 37-47). Springer, Cham.
18. Gao, W., (2020), New ant colony optimization algorithm for the traveling salesman problem. International Journal of Computational Intelligence Systems, 13(1), 44-55.
19. Pang, W., Wang, K. P., Zhou, C. G., & Dong, L. J., (2004, September), Fuzzy discrete particle swarm optimization for solving traveling salesman problem. In the Fourth International Conference on Computer and Information Technology, (2004). CIT'04. (pp. 796-800). IEEE.
20. Yang, X. S., (2013), Multiobjective firefly algorithm for continuous optimization. Engineering with computers, 29(2), 175-184.
21. Marinakis, Y., & Marinaki, M., (2010), A hybrid multi-swarm particle swarm optimization algorithm for the probabilistic traveling salesman problem. Computers & Operations Research, 37(3), 432-442.
22. Zhang, D., Wu, L., Ye, K., & Wang, M., (2021), Bayesian quantile semiparametric mixed-effects double regression models. Statistical Theory and Related Fields, 5(4), 303-315.

الروبط الالكترونية:

1. http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/