Journal of Information & Optimization Sciences
ISSN 0252-2667 (Print), ISSN 2169-0103 (Online)
Vol. 46 (2025), No. 3, pp. 737–746
DOI : 10.47974/JIOS-1791

Hybrid Niching Sparrow search algorithm for solving traveling salesman problem

Gagandeep Kaur Sidhu *
Jatinder Kaur †
Department of Mathematics
Chandigarh University
Gharuan 140413
Punjab
India

Abstract

The Traveling-Salesman-Problem (TSP) is a classic combinatorial optimization challenge with significant real-world applications in different fields, such as logistics, transportation, and manufacturing. In this research Sparrow Search Algorithm (SSA) is considered to solve TSP for random cities. SSA, when it comes to solving the Traveling Salesman Problem (TSP) alone, it does not exploit specific problem characteristics of the TSP. SSA basically operates on continuous domains and originate for solving optimization problems in real-valued spaces. The TSP, on the other hand, requires finding the optimal permutation of cities. So, this research paper introduces a novel hybrid algorithm, the hybrid Niching Sparrow Search Algorithm (NSSA) to handle TSP instances proficiently. The proposed algorithm integrates the analytic capabilities of the SSA with the niching techniques to find high-quality solutions to TSP instances. The integration of a niching strategy within NSSA aims to tackle these challenges by fostering solution diversity and encouraging exploration across various regions of SSA's search space. To evaluate the efficacy of the NSSA, computational experiments are conducted on a set of random cities for TSP. The outcomes of the experiment demonstrate the superior performance of the NSSA as compared to SSA.

Subject Classification: *90B25, 90C47, 90C90.*

Keywords: *Niching technique, Sparrow search algorithm (SSA), Optimization technique, Traveling salesman problem (TSP).*

* *E-mail:* **gagan.kw777@gmail.com (Corresponding Author)**
† *E-mail:* **jatinder.e1486@cumail.in**

© Taru Publications

1. Introduction

In the field of optimization, there are various optimization problems namely welded beam design optimization problem [1], diagnosing the breast cancer disease at its earliest [2], traveling-salesman-problem (TSP) [3] etc. that are solved till now by using various methods [4] to obtaining their optimal solution. The TSP is among them which is easy to state but difficult to solve. The goal of the well-known travelling-salesman-problem in operations research and computer science is to identify the lowest cost Hamiltonian circuit in any network. The problem can be expressed as follows: Let's say we have a list of cities (or nodes) and the distance among them. A salesman can only go to every city once, starting and ending at the same depot (or headquarters). Finding the lowest possible total distance (or expense) that the salesman must tour is the issue. A polynomial-time algorithm is not available to solve the TSP, which is also known as NP-hard**.** As a result, it is exceedingly challenging to find the optimal answer using the exact method. Heuristic algorithms are therefore applied nowadays to find an appropriate optimal solution to such situations within bearable computing bounds. A few of them include the following: CS Algorithm [5], ACO Algorithm [6], Genetic Algorithm (GA) [7]**,** Sparrow-Search- Algorithm (SSA) for TSP [8], Particle-Swarm-Optimization (PSO) [9], and numerous others. One notable algorithm among them is the Sparrow Search Algorithm (SSA) [10], given by Jiankai Xue and Bo Shen in 2020. SSA distinguishes itself from other metaheuristic algorithms with its ease of implementation, scalability, robustness, and efficiency in finding solutions. But the Sparrow search algorithm has some limitations in solving the Traveling Salesman Problem (TSP) on its own. As SSA is a generic optimization algorithm that does not exploit specific problem characteristics of the TSP. Therefore, in this research, the Hybrid Niching Sparrow Search Algorithm (NSSA) has been designed to efficiently process TSP instances.

In summary, this research paper examines the integration of two powerful optimization techniques, namely the Niching technique [11] and the Sparrow Search Algorithm, to address TSP (for random cities) competently. The focal point of the Niching technique is promoting diversity and maintaining multiple solutions in the population to circumvent premature convergence to a single optimal solution. On the other hand, the SSA, motivated by sparrows' foraging tendencies, leverages the concepts of exploration and exploitation to improve the search capabilities and rate of convergence of the algorithm. In this research, we propose an innovative approach that amalgamates the advantages of both the Niching technique and the Sparrow Search Algorithm, enabling an effective exploration of the solution space, leading to the discovery of multiple feasible and optimal solutions TSP. In NSSA, a niche approach has been used to update follower and scouter positions. The proposed NSSA algorithm plays an essential role in this work by showing the accuracy not only in convergence but also to obtain optimal path for the random cities as compared to original SSA.

2. Sparrow Search Algorithm

Teamwork is the main operating principle of SSA, focusing on foraging behaviour and the sparrows' natural defence mechanisms against predators. While there exists a variety of sparrow species, the experiment utilized virtual sparrows, categorized into two distinct roles: leaders and followers. Among the sparrow population, 20% were designated as leaders due to their heightened strength, with the remaining 80% serving as followers. Followers mimic the movements of leaders to locate food, occasionally swapping roles, yet maintaining a consistent follower-to-leader ratio. This dynamic extends to instances where certain sparrows, identified as "scouters," detect potential threats, prompting the entire population to migrate to safety under their guidance. In this experiment for TSP, firstly initialize the distance matrix for the TSP ("randomly generated") and the sparrows' location by the matrix of size (n, d) given in eq. (1):

$$X = \begin{bmatrix} x_{1,1} & \cdots & x_{1,d} \\ \vdots & \ddots & \vdots \\ x_{n,1} & \cdots & x_{n,d} \end{bmatrix} \tag{1}$$

In eq. (1), the count of sparrows is n, and there are d cities in the TSP. The location of leaders is located by using the eq. (2).

$$= \begin{cases} X_{i,j}^{t} \,.\, exp\left(\frac{-i}{\alpha \,.\, iter_{max}}\right) & ; \quad \text{if } R_2 < ST \\ X_{i,j}^{t} + Q \,.\, L & ; \quad \text{if } R_2 \geq ST \end{cases} \tag{2}$$

In eq. (2) above, the range of "j" is "1 to d", with the present iteration is "t". The parameter "$\alpha \in (0,1]$" is treated as a random variable. $X_{i,j}$ is the "i^{th} sparrow in j^{th} dimension". $iter_{max}$ symbolize for the maximum iterations. "ST (where $ST \in [0.5, 1.0]$)" work for the "safety threshold", while "R_2 (where $R_2 \in [0, 1]$)" acts as the alarm value. L denotes "1×d" matrix with every element equal to 1. The random variable Q follows a normal distribution. When "$R_2 < ST$", it demonstrates an absence of threats around the sparrows, allowing leaders to conduct searches for food on a large scale. Conversely, if "$R_2 \geq ST$", it tells the presence of threats, prompting all sparrows to promptly shift to a secure area. The location of the followers is located by using the eq. (3).

$$X_{i,j}^{t+1} = \begin{cases} Q \,.\, exp\left(\frac{X_{worst}^{t} - X_{i,j}^{t}}{i^2}\right) & ; \quad \text{if } i > {}^{n}/_{2} \\ X_{p}^{t+1} + \left|X_{i,j}^{t} - X_{p}^{t+1}\right| * A^{+} * L & ; \quad \text{if } i \leq {}^{n}/_{2} \end{cases} \tag{3}$$

In eq. (3) presented earlier, "X_{worst}" denotes the "present global worst position", whereas "X_P" represents the optimal location assumed by leaders. The whole population of sparrows is represented by the variable n. "A" is a "$1 \times d$" matrix consisting of elements that are either 1 or -1, and it satisfies the condition "$A^{+} = A^{T} (AA^{T})^{-1}$". The condition "$i > \frac{n}{2}$" indicates that the fitness value of the i^{th} follower is poor, leading to a lack of food and potential risk of starvation. The location of the scouters is located by using the eq. (4).

$$X_{i,j}^{t+1} = \begin{cases} X_{best}^{t} + \beta . \left| X_{i,j}^{t} - X_{best}^{t} \right| & ; \quad \text{if } f_i > f_g \\ X_{i,j}^{t} + K . \left(\frac{\left| X_{i,j}^{t} - X_{worst}^{t} \right|}{(f_i - f_w) + \varepsilon} \right) & ; \quad \text{if } f_i = f_g \end{cases} \tag{4}$$

In eq. (4) above, "X_{best} is the current global optimal location". The parameter "$\beta \in \text{N}(0,1)$" depicts a random number, while "$\text{K} \in [-1, 1]$" represents a random variable. The fitness value of the present sparrow is indicated by "f_i" with "f_w" and "f_g" representing the present worst and global best fitness values, respectively. "ε" represents the smallest constant utilized to prevent division by zero errors. When "$f_i > f_g$", it indicates sparrows located at the group's periphery, whereas "$f_i = f_g$" implies that scouts are fully vigilant and need to adjust their positions closer to others to minimize insecurity.

3. Niching Technique

The suggested approach utilizes a niching method in which "The Fitness Euclidean-distance ratio SSA (FER-SSA)," developed based on the FDR-PSO principle [11]–[12], to identify optimal neighborhood solutions instead of relying solely on the global optimum. The eq. (5) utilized to compute the FER for two sparrows, denoted as i and j:

$$FER_{ij} = \alpha . \frac{f(p_i) - f(p_j)}{\|p_i - p_j\|} \tag{5}$$

In the above eq. (5), p_i and p_j represent the personal best positions of sparrows i and j respectively, while $f(p_i)$ and $f(p_j)$ indicate their respective objective function values. The scaling factor α is determined by the provided eq. (6):

$$\alpha = \frac{\|S\|}{f(p_b) - f(p_w)} \tag{6}$$

In equation (6), $f(p_b)$ and $f(p_w)$ denote the objective function values of the best and worst sparrows. The search space, denoted as $\|S\|$, is determined by the diagonal distance, as described in eq. (7):

$$\|S\| = \sqrt{\sum_{r=1}^{D} (x_r^u - x_r^l)^2} \tag{7}$$

In equation (7) above, x_r^u signifies the upper bound and x_r^l denotes the lower bound of the r^{th} dimension within the search space. Niching stands out as an effective approach capable of meeting the demands of extensive exploration in tackling optimization problems. To monitor the most effective solutions uncovered while conducting the search, we integrated personal best attributes into the proposed SSA algorithm. Every iteration guides the sparrows toward the most optimal neighborhood values by computing the FER values. A notable benefit of the FER value lies in its parameter-free nature.

4. Niching Sparrow Search Algorithm (NSSA)

During every iteration, we compute nbest_i and the process of figuring out the nbest for the ith sparrow in FER-SSA is outlined in "Algorithm 1".

Algorithm 1: Pseudocode for identifying the $nbest$ solution for sparrow i in FER-SSA

Input: Total no. of sparrows considers as n

Result: Neighborhood best $(nbest)$ relies on the FER value of the i^{th} sparrow.

FER ← 0, Ep ← 0, $nbest_i \leftarrow pbest_i$

For i =1 to n

  For j =1 to n

    Compute the FER utilizing equation (6)

    If j =1

      Ep = FER

    Endif

    If FER > Ep

      Ep = FER

      $nbest_i = pbest_i$

    Endif

  Endfor

Endfor

Both followers' and scouters' equations can be changed by using the nbest_i. Changes to the followers and scouters are indicated in equations (8) and (9) respectively:

$$X_{i,j}^{t+1} = \begin{cases} Q\ .\ exp\left(\frac{X_{worst}^{t} - X_{i,j}^{t}}{i^2}\right) & ; \quad \text{if } i > {}^{n}/_{2} \\ nbest_i + \left|X_{i,j}^{t} - X_{p}^{t+1}\right| * A^{+} * L & ; \quad \text{if } i \leq {}^{n}/_{2} \end{cases} \tag{8}$$

$$X_{i,j}^{t+1} = \begin{cases} nbest_i + \beta.\left|X_{i,j}^{t} - nbest_i\right| & ; \quad \text{if } f_i > f_g \\ X_{i,j}^{t} + K.\left(\frac{\left|X_{i,j}^{t} - X_{worst}^{t}\right|}{(f_i - f_w) + \varepsilon}\right) & ; \quad \text{if } f_i = f_g \end{cases} \tag{9}$$

A niching constant, denoted as "NC = 0.5" is defined, and a random number, r_i is generated during each iteration. If "r_i > NC", followers and scouters are directed by nbest_i using equations (8) and (9) respectively. Otherwise, they are guided by the original equations (3) and (4) respectively. It's essential to remember that every time leaders are guided by the real position update equation (2). The functioning of Niching SSA is illustrated in "Algorithm 2".

Algorithm 2: Structure of NSSA

Input:

$\boldsymbol{n}$ **:** total no. of sparrows, $\boldsymbol{r_i}$: random number, **NC:** 0.5, **MaxIt :** maximum iterations

Initialize the population while defining its applicable parameters then after calculate its fitness value

Output: X_{best} , f_g.

1: While (t < MaxIt)

2: Order the fitness values and determine both the current best and worst individuals.
3: Divide the population accordingly 20% of the population that has the finest fitness values as leaders and rest 80% as followers.
4: Apply niching technique as mentioned in section 3 and calculate $nbest_i$ from Algorithm 1.
5: Update the position of leaders by using equation (2);
6: r_i = rand()
7: If r_i > NC
8: Update the position of followers by applying equation (8);
9: Update the position of scouters by implementing equation (9);
10: Else $r_i \leq$ NC
11: Update the position of followers by using equation (3);
12: Update the position of scouters by using equation (4);
13: End If
14: Obtain the present updated position;
15 : If the new position surpasses the previous one, proceed to update it;
15 : t = t + 1
16 : End While
17 : Return X_{best} , f_g.

5. Preparation of the Experiment and Analysis

5.1 *Setting up the Experimental Environment and Establishing the Parameters.*

Both NSSA and SSA algorithms were carried out on a "Windows 10 system equipped with a 64-bit operating system", 32.0 GB of RAM, and an "Intel(R) Core(TM) i5-7300U CPU @ 2.60GHz 2.71 GHz" processor. The experiments were coded using MATLAB R2023a. Random sets of cities were selected to solve the TSP problem, comprising 3, 6, 9, and 12 cities, respectively. This study intends to determine the optimality, effectiveness, stability of the Niching Sparrow Search Algorithm (NSSA) in determining the optimal route with the minimum distance travelled by the salesman. NSSA's performance is compared with that of SSA. Both algorithms employed 100 search agents and were run for a maximum of 500 iterations. The parameter setting for both algorithms was kept identical, with ST=0.8.

5.2 *Results of Experiment and Analysis*

This study involved conducting 30 separate executions of NSSA and SSA algorithms on randomly selected sets of cities, like 3, 6, 9, and 12 cities. We computed the optimal route with the minimum distance traveled by the salesman, as well as the worst-case scenario (maximum distance), the average distance (mean), and the standard deviation. The optimal and worst-case values assess the algorithms' optimality, while the standard deviation reflects algorithm stability and the mean value indicates convergence accuracy. The outcomes for NSSA and SSA applied to the TSP are summarized in Table 1.

Table 1
Output of NSSA and SSA

No. of Cities		Minimum Distance	Maximum Distance	Mean Value	Std. Dev
3 Cities	**NSSA**	0.009419 km	1.554125 km	0.720820 km	0.40902 km
	SSA	0.043959 km	1.579809 km	0.527165 km	0.37464 km
6 Cities	**NSSA**	0.02154 km	1.599999 km	0.710076 km	0.51582 km
	SSA	0.060865 km	2.543271 km	0.7000 km	0.64128 km
9 Cities	**NSSA**	0.021197 km	2.606539 km	0.910736 km	0.74540 km
	SSA	0.116152 km	2.655132 km	1.030155 km	0.71534 km
12 Cities	**NSSA**	0.035201 km	3.778348 km	1.759382 km	1.15818 km
	SSA	0.117765 km	3.965159 km	1.066742 km	1.03978 km

From Table 1, it is crystal clear that the optimal route with minimum distance is obtained by NSSA as compared to SSA for all the random cities such as 3, 6, 9, 12 cities. Moreover, the maximum distance of NSSA is also better than the maximum distance travelled by SSA for every city. Best city route order found by NSSA for 3, 6, 9, 12 cities is given in Table 2, 3, 4, and 5 respectively. Figure 1 (a-d), represents the NSSA's distance matrix for 3, 6, 9, 12 cities. The best route for the 3, 6, 9, 12 cities, as demonstrated in Figure 2 (a-d), in which the salesman starts traveling from point S denotes the initial position (starting node or city) and moves towards point E depicts the ending position (last node or city) then come back to the point S.

Table 2
Best city route order found by NSSA for 3 cities

3	2	1	3

Table 3
Best city route order found by NSSA for 6 cities

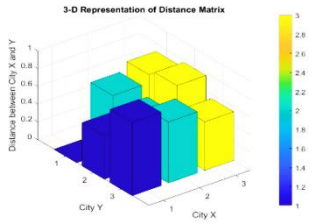
5	3	4	6	2	1	5

Table 4
Best city route order found by NSSA for 9 cities

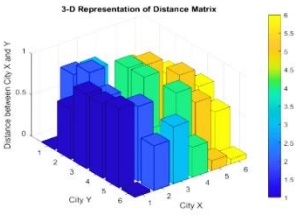
2	1	8	5	4	9	7	6	3	2

Table 5
Best city route order found by NSSA for 12 cities

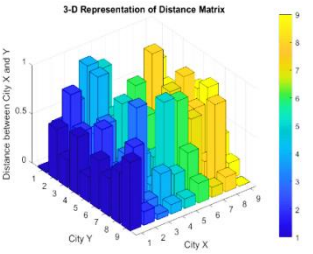
5	8	3	7	12	1	9	10	4	2	6	11	5



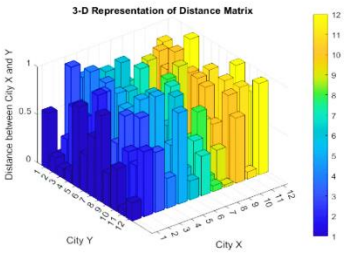
(a) NSSA's Distance Matrix for 3 Cities



(b) NSSA's Distance Matrix for 6 Cities

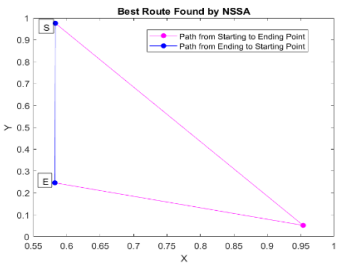


(c) NSSA's Distance Matrix for 9 Cities

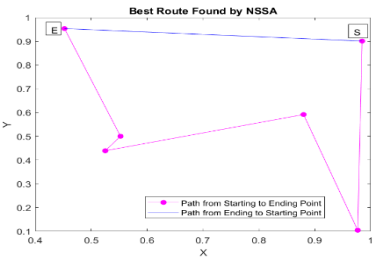


(d) NSSA's Distance Matrix for 12 Cities

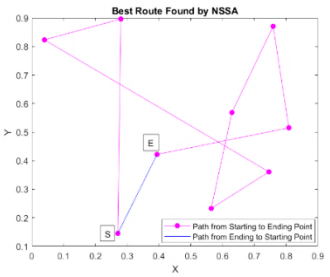
Figure 1
3D bar graph representation of NSSA's distance matrix for random cities



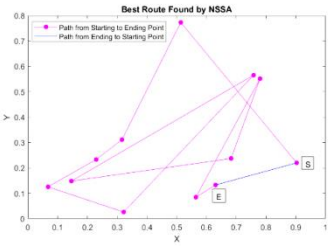
(a) NSSA's Best Route for 3 Cities



(b) NSSA's Best Route for 6 Cities



(c) NSSA's Best Route for 9 Cities



(d) NSSA's Best Route for 12 Cities

Figure 2
NSSA's best rout representation for random cities

6. Conclusion

This research introduces an innovative approach that combines the strengths of both the Sparrow Search Algorithm and the Niching technique, allowing for efficient solution space exploration and the identification of several feasible and best TSP solutions. We employ a niche strategy in this NSSA to update the positions of scouters and followers. The suggested NSSA algorithm, which displays the best path with the shortest distance when compared to SSA, has significance to this work. Additionally, whereas SSA has covered a larger maximum distance, NSSA has covered a shorter maximum distance. Hence NSSA provides the best results for TSP of random cities.

References

[1] M. Sarkar and T. K. Roy, "Optimization of welded beam structure using neutrosophic optimization technique: A comparative study," *Int. J. Fuzzy Syst.,* vol. 20, no. 3, pp. 847–860 (2018).

[2] Doreswamy and U. S. M., "A binary bat inspired algorithm for the classification of breast cancer data," *Int. J. Soft Comput. Artif. Intell. Appl.,* vol. 5, no. 2/3, pp. 01–21 (2016).

[3] X. Meng, J. Li, and M. Zhou, "A colored traveling salesman problem with varying city colors," *Discret. Dyn. Nat. Soc.,* vol. 2021 (2021).

[4] G. Venter, "Review of optimization techniques," *Encycl. Aerosp. Eng.,* pp. 1–12 (2010).

[5] X.-S. Yang, Nature-Inspired Metaheuristic Algorithms. Luniver Press (2010).

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

[7] J. H. Holland, Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence. Cambridge, MA, USA: MIT Press (1992).

[8] C. Wu, X. Fu, J. Pei, and Z. Dong, "A novel sparrow search algorithm for the traveling salesman problem," *IEEE Access,* vol. 9, pp. 153456–153471 (2021).

[9] W. Zhong, J. Zhang, and W. Chen, "A novel discrete particle swarm optimization to solve traveling salesman problem," in Proc. 2007

IEEE Congress on Evolutionary Computation, Singapore, pp. 3283–3287 (2007).

[10] J. Xue and B. Shen, "A novel swarm intelligence optimization approach: Sparrow search algorithm," *Syst. Sci. Control Eng.,* vol. 8, no. 1, pp. 22–34 (2020).

[11] B. Y. Qu, J. J. Liang, and P. N. Suganthan, "Niching particle swarm optimization with local search for multi-modal optimization," *Inf. Sci.*, vol. 197, pp. 131–143 (2012).

[12] T. Peram, K. Veeramachaneni, and C. K. Mohan, "Fitness-distance-ratio based particle swarm optimization," in Proc. 2003 IEEE Swarm Intell. Symp. (SIS 2003), Indianapolis, IN, USA, pp. 174–181 (2003).

Received September, 2024