International Conference on Advances in Materials, Computing and Communication Technologies
In Association with International Journal of Scientific Research in Science and Technology
Volume 9 | Issue 1 | Print ISSN: 2395-6011 | Online ISSN: 2395-602X (www.ijsrst.com)

Enhancement of Quadratic Assignment Problem Using Slime Mould Algorithm

S.Rajalakshmi[1], Dr. Kanmani[2], C. Varsha[2], E Daphne Auxilia[2]
[1]Department of Computer Science and Engineering, Pondicherry Engineering College, Puducherry, India
[2]Department of Information Technology, Pondicherry Engineering College, Puducherry, India

ABSTRACT

The QAP problem is one of the most challenging NP-hard combinatorial optimization problems.The aim is to minimise the sum of the distances multiplied by the corresponding flows by assigning all facilities to different locations. As a result, this paper proposes the SlimeMould Algorithm Quadratic Assignment Problem (SMAQAP) for solving this problem in a reasonable amount of time.Slime mould algorithm (SMA) is a new stochastic optimizer with a mathematical model that uses adaptive weights to simulate the slime mould's mechanism of producing positive and negative feedback propagation wave centred on the bio-oscillator in order to form the effective approach for linking food with high exploratory ability.Then, we tested our algorithm on QAPLIB's benchmark instances. The proposed SMAQAP is compared with genetic algorithm (GA), particle swarm optimization (PSO) algorithm and firefly algorithm (FA) to solve the quadratic assignment problems.
Keywords— Metaheuristics, Combinatorial optimization, Slime Mould Algorithm, Quadratic Assignment Problem

I. INTRODUCTION

Meta-heuristic algorithms are playing an increasingly important role in solving optimization problems in mathematics and computer science. An optimization algorithm is an iterative method that compares various solutions until to obtain one satisfactory solution. The optimization problems have three basic components (i) an objective function to optimize. (ii)A collection of unknown variables that influence the objective function's value and (iii)A collection of restrictions that enable the unknown to take a certain value while excluding others(Kar, Arpan Kumar, 2016).There are two types of optimization algorithms in use: deterministic and stochastic(Fleming, Wendell H., and Raymond W. Rishel,2012). Deterministic algorithms define the solution using a set of rules in a deterministic manner. While stochastic algorithms, on the other hand, evaluate the solution using probabilistic predictions(Sharma, Manik, and PrableenKaur, 2020).Nature is a habitat, a climate, and, most importantly, a diverse range of organisms. These species lives in different ways, have different behaviours, and forage in different manner. The study of their behaviour and their transformation into algorithms is referred to as "nature inspired algorithms." (Luthra, Ishani, et al., 2017).

Nature-inspired algorithms are a class of novel problem-solving methodologies and methods that have received widespread attention in various fields. Artificial Neural Networks (ANN)(Livingstone, David J., et al., 2008), fuzzy systems (FS)(Kosko, Bart., 1994), Evolutionary Algorithm (EA)(Eiben, AgostonEndre, and Selmar K. Smit., 2011), and Swarm Intelligence (SI) algorithm (Yang, Xin-She, et al., 2013) are examples of nature-inspired algorithms used to solve a wide range of real-world problems(Luthra, Ishani, et al., 2017).Nature-inspired algorithms are more efficient in solving optimization problems numerically. Insects, trees, territorial animals, birds, and flowers are examples of natural beings with characteristics that can be used to solve optimization problems. These features allow users to use them in a variety of fields, including research, engineering, and management. Nature-inspired algorithms (NIA) make mathematical modelling of any complex problem easier by adhering the rules and characteristics of nature(Adithyan, T. Ajay, et al., 2017).

Nature inspired algorithms are classified into two different types: Swarm Intelligence algorithm(Yang, Xin-She, et al., 2013) and Evolutionary algorithm(Eiben, AgostonEndre, and Selmar K. Smit., 2011).Evolutionary algorithms are based on Darwin's principle of survival of the fittest. The evolutionary algorithm (EA) is based on the natural stage of biological evolution. Swarm Intelligence Algorithms are concerned with the continuous and combinatorial collective intelligence of self-organized highly centralized systems(Yang, Xin-She, et al., 2013). In reality, Swarm intelligence is part of a larger category of bio-inspired algorithms that includes the vast majority of nature-inspired algorithms(Fan, Xumei, et al., 2020).Fig.1 depicts the categorization of heuristic methods.

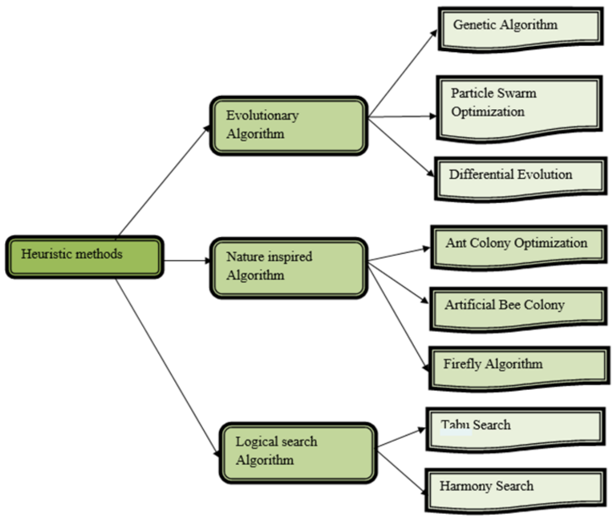

Fig. 1. Heuristic method classification

ABC's(Samanta, Suman, Deepu Philip, and Shankar Chakraborty, 2019)aquick convergent method is used to solve QAP. The performance of the algorithm is compared to that of other algorithms in the literature, and the results show that the algorithm can converge quickly in the majority of cases.Li, Shimin, et al (2020)proposed a new method for stochastic optimization called Slime Mould Algorithm. With a special mathematical model that employs adaptive weights to simulate the process of producing positive and negative feedback of the slime mould food with superior exploring properties. The proposed SMA is compared to current metaheuristics in a large collection of benchmarks to determine its efficiency.

Kılıç, Haydar, and UğurYüzgeç.(2019)proposed and developed a tournament selection based Antlion optimization algorithm (TALO) for quadratic assignment problems (QAP), which is an improved version of the Antlion optimizer (ALO) that overcomes ALO's drawbacks such as long run time, local optima stagnation, and premature convergence.When compared to other metaheuristic algorithms, the results of this algorithm show the best performance.Sharma, Manik, and PrableenKaur(2020)used to solve a variety of real-world and multidisciplinary problems with great success. The relevance and redundancy metrics, as well as the feature selection process, are briefly explained. It also shows a summary of nature-inspired meta-heuristic methods and their variants, as well as datasets and performance (mean, best, worst, error rate, and standard deviation).

A new ABC (Dokeroglu, Tansel, Ender Sevinc, and AhmetCosar, 2019) algorithm is proposed to solve the QAP Problem optimally, and the exploration and exploitation phases are well balanced. Better results can be obtained by using a larger number of processors and better-tuned parameters.Section II provides a brief overview of the Quadratic Assignment Problem (QAP), as well as a description of the Slime Mould Algorithm. Section III discusses the effectiveness of QAP using SMA and Section – IV discussed the proposed work's conclusion.

II. SLIME MOULD ALGORITHM FOR QUADRATIC ASSIGNMENT PROBLEM

The quadratic assignment problem is used in a variety of algorithms, including Antlion Optimization Algorithm(ALO)(Immanuel, Savio D., and Udit Kr Chakraborty, 2019), Genetic Algorithm(GA)(Azarbonyad, Hosein, and Reza Babazadeh, 2014), Artificial Bee Colony Algorithm(ABC) (Dokeroglu, Tansel, Ender Sevinc, and AhmetCosar, 2019),Firefly Algorithm(FA) (Guo, Meng-Wei, Jie-Sheng Wang, and Xue Yang, 2020), Memetic Algorithm(MA) (Cubukcuoglu, Cemre, et al., 2019) and Simulated Annealing (SA) Algorithm.

The Quadratic Assignment Problem (QAP) is a difficult combinatorial problem that has been proved to be NP-Hard. Koopmans and Beckmann suggested the Quadratic Assignment Problem (QAP) to formulate economic activities(Guo, Meng-Wei, Jie-Sheng Wang, and Xue Yang, 2020). The problem of assigning number of facilities to number of locations with given distances and flows betweenthem is known as the QAP.

A. Quadratic Assignment Problem

One of the most difficult combinatorial optimization problems is the Quadratic Assignment Problem (QAP). It has been demonstrated that QAP is Non-Deterministic polynomial (NP)-hard problem (Paz, Azaria, and Shlomo Moran, 1981) that takes a reasonable amount of time to solve when the problem size is high.Finding a feasible solution using traditional approaches would be a time-consuming process. As a consequence, meta-heuristic approaches such as GA (Azarbonyad, Hosein, and Reza Babazadeh, 2014), PSO (Congying, Lv, Zhao Huanping, and Yang Xinfeng, 2011),ACO, and others, are often employed to produce a feasible solution to improve an existing one.

The problem of assigning a set of facilities to a set of locations of QAP is given by Eq.1(Dokeroglu, Tansel, Ender Sevinc, and AhmetCosar, 2019).

$$\min c(\pi) = \sum_{i=1}^{n} \sum_{k=1}^{n} x_{ik} y_{mn} \qquad (1)$$

X =$[x_{ik}]$ where x_ikis the distance between locations i and k,

Y =$[y_{mn}]$ where y_mnis the flow between facilities m and n.

B. Slime Mould Algorithm

Slime Mould Algorithm is based on food foraging behaviour and its oscillation mode for approaching food sources. Fig.2 depicts the slime mould's structure and interconnected venous network. It employs weights to encourage the formation of positive and negative feedback in the propagation wave of slime mould based on a bio-oscillator, leading to the formation of an optimal route for integrating food with outstanding exploratory and exploitation capability. Slime mould algorithm can use multiple food resources at the same time due to its unique biological characteristics (Li,Shimin, et al., 2020)

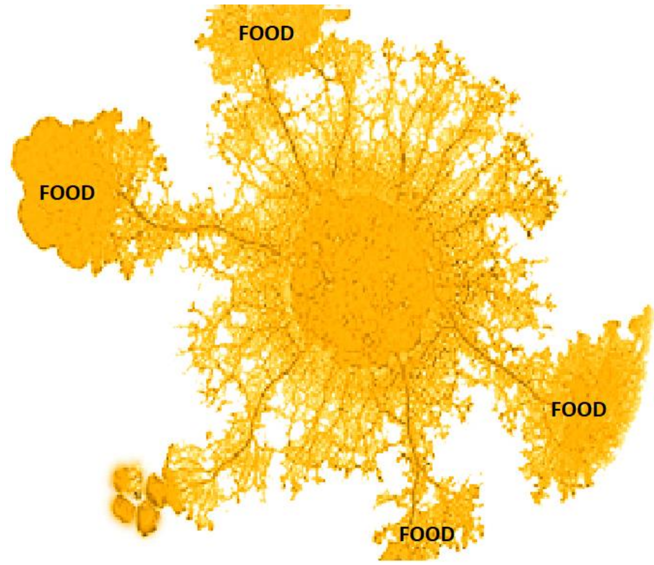

Fig.2.Slime Mould Morphology

If the food quality is good, slime mould can use a region-based search method; otherwise, if the food density is low, slime mould may leave that area. Food quality determines vein thickness; a bio oscillator generates a propagating wave, which increases cytoplasmic flow in the vein. The combination of slime's positive and negative feedback will aid in connecting food in an efficient manner, resulting in an optimal path. Slime mould is drawn to food by the odour in the air. The following formulas are proposed to mimic the contraction mode's approaching actions in mathematical formulas:

$$\overrightarrow{Y(u+1)} = \begin{cases} \overrightarrow{Y_{c(u)}} + \overrightarrow{wc}\cdot\left(\vec{X}\cdot\overrightarrow{Y_B(u)} - \overrightarrow{Y_C(u)}\right), ra < p \\ \overrightarrow{wd}\cdot\overrightarrow{Y(u)}, \mathrm{ra} \geq \mathrm{p} \end{cases}$$

(2)

Where, $\overrightarrow{wc}$is a parameter with [−a, a]range, $\overrightarrow{wd}$ from one to zero, decreases linearly, u represents the most recent iteration $\overrightarrow{Y_c}$ the specific place where the highest odour concentration has been discovered, $\vec{X}$ represents the slime mould's position, $\overrightarrow{Y_B}$ **and** $\overrightarrow{Y_C}$ reflect two slime mould individuals chosen at random.

$$\mathrm{p} = \tanh|E(i) - OF| \quad (3)$$

Where i∈1,2,…n, E(i) represents the fitness ofY ⃗ ,OF represents the highest degree of fitness attained through all iterations.

The formula of (wc) ⃗ is as follows:

$$\overrightarrow{wc} = [-a,a] \quad (4)$$

$$a = \operatorname{arctan}h\left(-\left(\frac{u}{max\_u}\right) + 1\right) \quad (5)$$

The following is a list of X ⃗'s formula:

$$\overrightarrow{X(SmellScale(n))} = \begin{cases} 1 + r.\log\left(\frac{iF-E(i)}{iF-bF} + 1\right), situation \\ 1 - r.\log\left(\frac{iF-E(i)}{iF-bF} + 1\right), others \end{cases} \quad (6)$$

$$SmellScale = \mathrm{sort(E)} \quad (7)$$

Where, E(i) ranks first half of the population, ra denotes the random value in the interval of [0,1], iF denotes the ideal fitness obtained in the iterative method currently,bF denotes the bad fitness obtained in the iterative method currently,SmellScale denotes the sequence of fitness value sorted.

When searching, this section mathematically simulates the contraction mode of slime mould venous tissue structure. The following is the mathematical formula for updating the position of slime mould:

$$\vec{Y} = \begin{cases} rand.(UB-LB) + LB, rand < L \\ \overrightarrow{Y_{c(u)}} + \overrightarrow{wc}.\left(\vec{X}.\overrightarrow{Y_B(u)} - \overrightarrow{Y_C(u)}\right), \mathrm{ra} < q \\ \overrightarrow{wd}.\overrightarrow{Y(u)}, \mathrm{ra} \geq \mathrm{q} \end{cases} \quad (8)$$

Where*LB* and *UB* denote the lower and upper boundaries of search range,*rand*andra denote the random value in [0,1].

Slime mould primarily relies on the biological oscillator's propagation wave to alter the cytoplasmic flow in veins, causing them to be in a better location of food concentration. We used□((X) ⃗),□((wc) ⃗), and □(□((wd) ⃗)) to simulate differences in slime mould venous width.As the number of iterations increases, the value of (wc) ⃗oscillates randomly between [-a, a] and eventually reaches zero. The value of□((wd) ⃗) oscillates between [-1,1] and ultimately tends to zero.Furthermore, the (wc) ⃗ oscillation mechanism mimics slime mould determineswhether to approach the food source or search out other food sources. Various obstacles, such as light and a dry atmosphere, can limit the spread of slime mould during this period. However, it enhances the slime mould's capacity to locate higher-quality food and prevents the trapping of the local optimum. Fig.3 illustrates the process for determining slime mould fitness values and the pseudo code of the SMA is explained in Fig.4

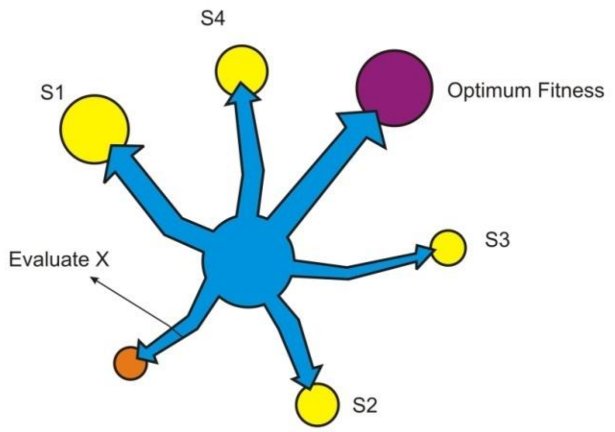

Fig.3. Fitness Evaluation

Initialize the parameters, population_sizeand iteration _limit

Set the pos. of slime mould $Y(i = 1,2, \dots , n)$;

While (u ≤ iteration_limit)

Determine the fitness of all slime mould;

Update optimum fitness, Y_c

Determine the Weight byEq. (5)

For each part of the search area

Update q, wb, wc;

Update position byEq.(1)

End for

u= u+1;

end while

return optimum fitness, Y_C;

Fig.4. SMAPseudocode

C. Application of QAP using SMA

As previously stated, the Slime Mould Algorithm uses the global optimum and local optimum value to guide the position of Slime Mould.Every slime corresponds to an independent process in which a solution is built during the search space and to find the best solution. Table 1 gives the details about the parameters used in SMAQAP. The location of SMA is updated using Eq. 7. We'll use Slime Mould Algorithm to find the best solution to the QAP.SMA can be used to solve the QAP directly.In QAP, SMA positions are used to locate facilities.The total fitness value of QAP is determined using a Distance matrix and a Weight matrix based on the facility's assigned location.

The results of proposed SMAQAP was compared to those of well-known meta-heuristic algorithms likeGenetic Algorithm(GA)(Azarbonyad, Hosein, and Reza Babazadeh, 2014),Particle Swarm Optimization(PSO)(Congying, Lv, Zhao Huanping, and Yang Xinfeng,2011), and Firefly Algorithm (FA)(Guo, Meng-Wei, Jie-Sheng Wang, and Xue Yang, 2020).

TABLE 1. EXPERIMENT PARAMETERS

WEIGHT MATRIX	R[20×20]
DISTANCE MATRIX	T[20×20]
ITERATIONS	500
POPULATION SIZE	30
LOCATIONS	40
NO OF RUNS	10
DIMENSIONS	30

Fig.5.depicts the location of the QAP.In QAP, 40 locations (not allocated facilities) will be used. In QAP, the population size is determined by the number of allocated facilities. The population size was used as half of the number of facilities in this study. According to assigned locations of facilities, QAP's total fitness value is calculated using R[20x20] weight matrix and the T[20x20] distance matrix.Fig.6 explains a pseudo code for solving QAP using the Slime Mould Algorithm (SMAQAP).

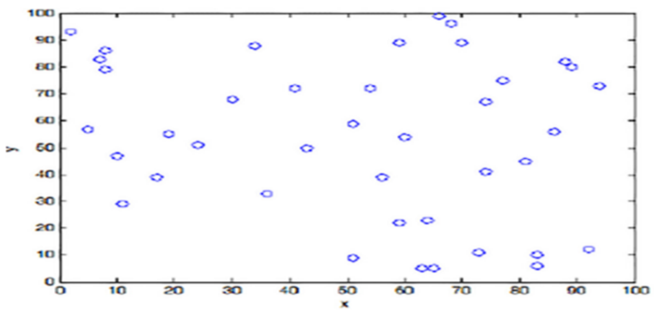

Fig. 5. QAPLocations

Input: SMA-produced locations, number of location, location vectors (X, Y), weight matrix (R), number of facilities.

Output: cost value production

Create a facility list based on the positions created by SMA.

Determine the distance between two points.

for g:number of locations

    for h = g + 1:number of locations

        calculate distance (s,h): $R_{sh} = ((X_s - X_h)^2 - ((Y_s - Y_h)^2))^{1/2}$

        distance (s, h) = distance (h, s)

    end for

Calculate total cost

cost = 0

    for g:number of facilities

        for h = g + 1:number of facilities

            cost = cost + weight (g, h)*distance(facility(g), facility(h)

        end for

    end for

end for

Fig. 6. SMAQAPPseudocode

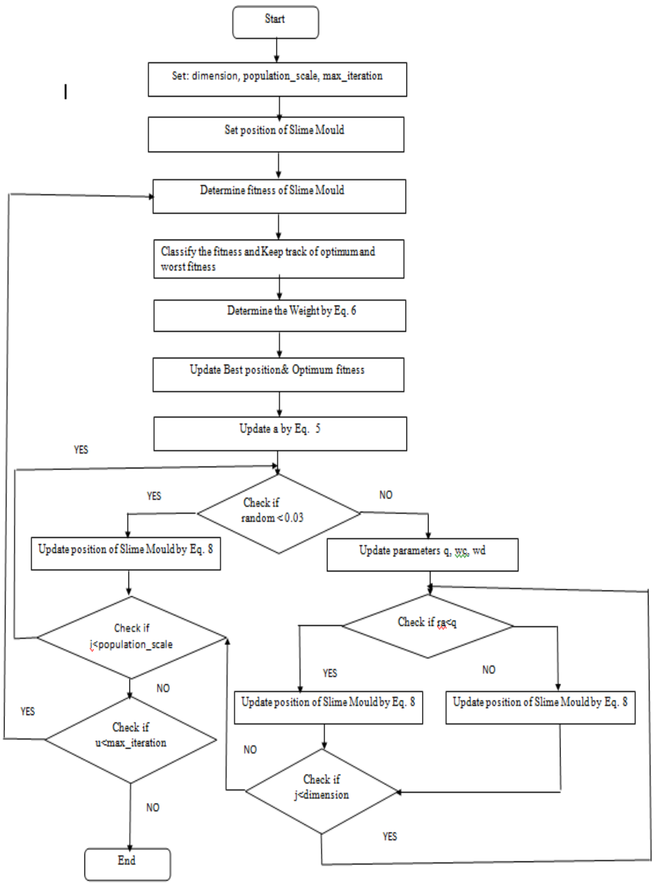

Fig.7. SMAFlowchart

The parameters of meta-heuristic algorithms like GA(Azarbonyad, Hosein, and Reza Babazadeh, 2014),PSO (Congying, Lv, Zhao Huanping, and Yang Xinfeng,2011)and FA(Guo, Meng-Wei, Jie-Sheng Wang, and Xue Yang, 2020) for QAP. The Slime Mould Algorithm mechanism is depicted in Fig.7.The convergence curves for SMAQAP and other Metaheuristics algorithm are shown in Fig. 8.

III. EXPERIEMENT AND RESULT ANALYSIS

For the proposed SMAQAP, a number of benchmark tests are used to demonstrate and a performance comparison was carried out using various metrics.The common standard benchmark is widely used to assess the reliability, performance, and validation of optimization algorithms. Every new optimization algorithm must be validated and compared to other existing algorithms using a comprehensive set of test functions. The benchmark functions used in this analysis are from QAPLIB instances (Burkard, Rainer

E., Stefan E. Karisch, and Franz Rendl,1997).We used six different benchmark function to evaluate the SMAQAP.To compare the results of SMAQAP and other Metaheuristics algorithm, we used three metrics: mean cost, Standard deviation and best cost. Table.3 represents the QAPLIB results of SMAQAP.

TABLE 2. BENCHMARK INSTANCES OF QAP

Name	**Size**	**Feasible Solution**	**Bound /Permutation**	**Gap = (solution - bound)/(solution)*100%**
Sko56	56	34458	32610 (TDB)	5.37 %
Sko81	81	90998	86072 (TDB)	5.41 %
Sko90	90	115534	109030 (TDB)	5.63 %
Tai17a	17	491812	(12,2,6,7,4,8, 14,5,11,3,16, 13,17,9,1,10, 15)	--
Tai25b	25	344355646	(4,15,10,9,13,5,25,19,7,3, 17,6,18,20,16,2,22,23,8,11,21,24,14,12,1)	--
Tai150b	150	498896643	441786736 (GLB)	11.45 %

*TDB =Triangle Decomposition Bound, GLB = Gilmore - Lawler Bound

TABLE 3.THE SMAQAP RESULTS OF VARIOUS BENCHMARK INSTANCES

Problem instance	**Mean**	**Best**	**Worst**	**Standard Deviation**
Sko56	34453	34357	34567	34826
Sko81	91886	90720	92345	91722
Sko90	115556	115421	115634	113432
Tai17a	510,298	496,596	527,335	543,732
Tai25b	1,233,441	1,223,346	1,244,438	1,233,342
Tai50b	5,214,352	5,198,731	5,248,332	5,235,463

In this paper, PSO(Congying, Lv, Zhao Huanping, and Yang Xinfeng,2011),GA(Azarbonyad, Hosein, and Reza Babazadeh, 2014), FA(Guo, Meng-Wei, Jie-Sheng Wang, and Xue Yang, 2020) and the SMA are used to solve QAP to examine the ability of each algorithm to obtain the optimal solution and their convergence characteristics. The QAP results of SMA and other meta-heuristic algorithm are shown in Table 4. Table.5 displays the best cost-benefit ratio for each iteration of QAP using nature-inspired algorithms.

TABLE 4.SMAQAP AND OTHER META-HEURISTIC ALGORITHMS' QAP RESULTS (Tai* Instance)

Algorithm	**Mean Cost**	**Standard Deviation**	**Worst Cost**	**Best Cost**
Slime Mould Algorithm using QAP	-1019910.74	66823.23	-915828.83	-1109325.59
Particle Swarm Optimization using	-1050518.95	48744.72	-972582.99	-1094306.47

QAP				
Genetic Algorithm using QAP	-1040715.59	32096.99	-998807.47	-1078899.84
Firefly Algorithm using QAP	-909843.67	128040.30	-704467.49	-1084973.23

TABLE 5.SMAQAP AND OTHER META-HEURISTIC ALGORITHMS' BEST FITNESS VALUES (Tai* Instance)

Number of iteration	**Slime Mould Algorithm (SMA)**	**Particle Swarm Optimization (PSO)**	**Genetic Algorithm(GA)**	**Firefly Algorithm(FA)**
50	-951590.0053	-895517.6311	-954532.342	-948349.3743
100	-979009.4995	-964550.5046	-982321.243	-995783.3784
150	-1059042.0091	-1075362.7233	-1074231.324	-1037388.3432
200	-1109258.0091	-1116896.5766	-1092421.243	-1083272.3432
250	-1129262.0091	-11253456.566	-1105342.232	-1103243.2341
300	-1139286.0091	-1128596.5766	-1106321.343	-1109276.2622
350	-1145292.0091	-1130900.6123	-1117232.493	-1115628.2373
400	-1149302.0091	-1135907.0943	-1123344.455	-1123628.2783
450	-1149312.0091	-1138964.043	-1133730.3484	-1137467.2722
500	-1159325.5901	-1144306.4731	-1138899.8435	-1138368.3832

Firefly Algorithm(Guo, Meng-Wei, Jie-Sheng Wang, and Xue Yang, 2020) is the worst algorithm in terms of all metrics as shown in Table 6. The SMAQAP in comparison to other algorithms has thehighest cost effectiveness. In terms of both exploration and exploitation, SMAQAPoutperforms better and other algorithms can struck in local optimum due to early convergence. SMAQAP's exploration and exploitation activities over a number of iterations, as well as its ability to avoid local optimum values are clearly demonstrated in Table 4 and 5.

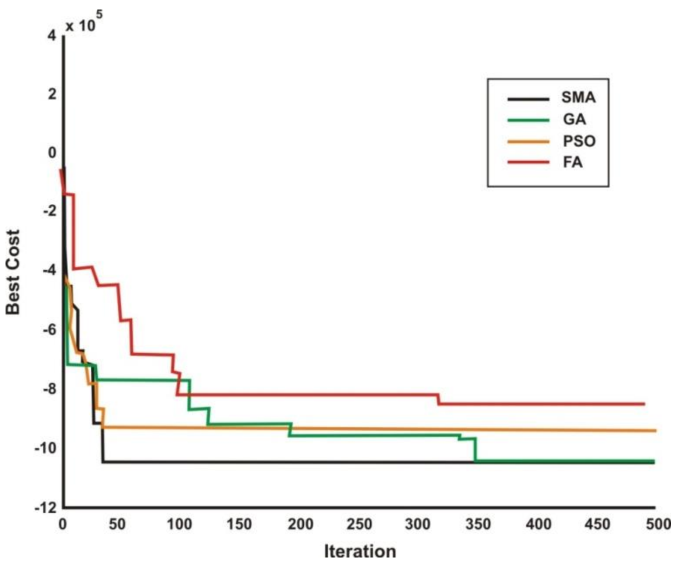

Fig. 8. The Convergence Curve Comparison Study

The simulation results for comparing the convergence effects of four algorithms for solving the quadratic assignment problem are shown in Fig.8.The proposed SMAQAP algorithm outperforms the other three algorithms in terms of simulation efficiency. Fig.8. shows that the best fitness value is close to -10, indicating the minimum value of the objective function during the optimization process.Slime Mould Algorithm was previously applied in many applications like Tokyo map (Beekman, Madeleine, and Tanya Latty,2015), Maze problem(Brabazon, Anthony, and Seán McGarraghy,2020), Travelling Salesman Problem(Lenstra, Jan Karel, and AHG RinnooyKan, 1975), networks problem (Zubaidi, Salah L., et al, 2020), etc. and it was discovered that it produces better results for these types of problems.

By integrating QAP with bio-inspired algorithms likePSO(Congying, Lv, Zhao Huanping, and Yang Xinfeng,2011), GA(Azarbonyad, Hosein, and Reza Babazadeh, 2014),FA(Guo, Meng-Wei, Jie-Sheng Wang, and Xue Yang, 2020) etc.. Provides the best results in terms of efficiency.Even when QAP is applied to these algorithms, some drawbacks exist, such as local optimum stagnation.QAP has similar features like routing problems, and SMA is already suitable for solving these types of problems. As a result, we are the first to apply QAP to SMA, and the problem will produce excellent results when compared toother algorithms listed in Table 6, as well as overcome the aforementioned disadvantage.

IV. CONCLUSION

QAP is a well-known NP Hard Combinatorial optimization problem.In this paper, we present a novel Slime Mould Algorithm (SMA) with the implementation of Quadratic Assignment Problem (QAP). The SMA metaheuristic is found to be effective in solving the QAP. Slime Mould Algorithm was adapted to the difficult quadratic assignment problem (QAP) and it is a new approach.The algorithm primarily employs weights to model the bio-oscillator's positive and negative feedback during foraging to the food source, resulting in a different thickness of the feeding vein network.The problem is to assign all facilities to different locations with the aim of minimizing the sum of the distances multiplied by the corresponding flows.In QAP, SMA positions are used to locate facilities.The total fitness value of QAP is determined using a Distance matrix and a Weight matrix based on the facility's assigned location. We first used three well-known meta-heuristic algorithms such as Genetic Algorithm(GA) (Azarbonyad, Hosein, and Reza Babazadeh, 2014),Particle Swarm Optimization(PSO) (Congying, Lv, Zhao Huanping, and Yang Xinfeng,2011), and Firefly Algorithm (FA)(Guo, Meng-Wei, Jie-Sheng Wang, and Xue Yang, 2020) to evaluate the algorithm's performance on the QAP instance. Finally, we looked at the performance of proposed SMAQAPfor some QAP instances (Tai*) from the QAPLIB platform(Burkard, Rainer E., Stefan E. Karisch, and Franz Rendl, 1997).The proposed SMAQAP algorithm solves the majority of the benchmark problem instances efficiently.Better results are identified when the exploration and exploitation phases are balanced well. The outcomes have been very promising.Our future work will be to add an improvement mechanism to SMA and apply it to QAP.

V. REFERENCES

[1]. Samanta, Suman, Deepu Philip & Shankar Chakraborty.(2019). "A quick convergent artificial bee colony algorithm for solving quadratic assignment problems." Computers & Industrial Engineering, 137 : 106070.

[2]. Li, Shimin, et al.(2020)."Slime mould algorithm: A new method for stochastic optimization." Future Generation Computer Systems.

[3]. Kılıç, Haydar, and UğurYüzgeç.(2019)."Tournament selection

based antlion optimization algorithm for solving quadratic assignment problem." Engineering science and technology, an international journal 22.2 , 673-691.

[4]. Cubukcuoglu, Cemre, et al.(2019)."A Memetic Algorithm for the Bi-Objective Quadratic Assignment Problem." Procedia Manufacturing, 39, 1215-1222.

[5]. Yang, Xin-She.(2020)."Nature-inspired optimization algorithms: challenges and open problems." Journal of Computational Science, 101104.

[6]. Bujok, Petr, Josef Tvrdík, and RadkaPoláková.(2019)."Comparison of nature-inspired population-based algorithms on continuous optimisation problems." Swarm and Evolutionary Computation, 50 , 100490.

[7]. Kumar, Manoj, AryabarttaSahu, and PinakiMitra.(2020)."A comparison of different metaheuristics for the quadratic assignment problem in accelerated systems." Applied Soft Computing , 106927.

[8]. Hameed, AsaadShakir, et al.(2018)."Review on the Methods to Solve Combinatorial Optimization Problems Particularly: Quadratic Assignment Model." International Journal of Engineering & Technology 7.3.20 , 15-20.

[9]. Hu, Kang, Guo-Li Zhang, and Bo Xiong.(2018)."An improved particle swarm algorithm for constrained optimization problem." 2018 International Conference on Machine Learning and Cybernetics (ICMLC).Vol. 2.IEEE.

[10]. Luthra, Ishani, et al.(2017)."Comparative study on nature inspired algorithms for optimization problem." 2017 International conference of Electronics, Communication and Aerospace Technology (ICECA).Vol. 2.IEEE, 2017.

[11]. Fan, Xumei, et al.(2020)."Review and classification of bio-inspired algorithms and their applications." Journal of Bionic Engineering 17 , 611-631.

[12]. Sharma, Manik, and PrableenKaur.(2020). "A comprehensive analysis of nature-inspired meta-heuristic techniques for feature selection problem." Archives of Computational Methods in Engineering , 1-25.

[13]. Immanuel, Savio D., and Udit Kr Chakraborty.(2019)."Genetic algorithm: an approach on optimization." 2019 International Conference on Communication and Electronics Systems (ICCES).IEEE.

[14]. Dokeroglu, Tansel, Ender Sevinc, and AhmetCosar.(2019)."Artificial bee colony optimization for the quadratic assignment problem." Applied soft computing 76 , 595-606.

[15]. Mamaghani, Ali Safari, and Mohammad Reza Meybodi.(2012)."Solving the Quadratic Assignment Problem with the modified hybrid PSO algorithm." 2012 6th International Conference on Application of Information and Communication Technologies (AICT).IEEE.

[16]. Rameshkumar, K.(2018)."Extension of PSO and ACO-PSO algorithms for solving Quadratic Assignment Problems." IOP Conference Series: Materials Science and Engineering. Vol. 377.No. 1.IOP Publishing.

[17]. Guo, Meng-Wei, Jie-Sheng Wang, and Xue Yang.(2020)."An chaotic firefly algorithmto solve quadratic assignment problem." EngLett 28.2 , 337-342.

[18]. Congying, Lv, Zhao Huanping, and Yang Xinfeng.(2011)."Particle swarm optimization algorithm for quadratic assignment problem." Proceedings of 2011 International Conference on Computer Science and Network Technology.Vol. 3.IEEE.

[19]. Azarbonyad, Hosein, and Reza Babazadeh.(2014)."A genetic algorithm for solving quadratic assignment problem (QAP)." arXiv preprint arXiv:1405.5050.

[20]. Ahuja, Ravindra K., James B. Orlin, and Ashish Tiwari.(2000)."A greedy genetic algorithm for

the quadratic assignment problem." Computers & Operations Research 27.10 , 917-934.

[21]. Poli, Riccardo, James Kennedy, and Tim Blackwell.(2007)."Particle swarm optimization." Swarm intelligence 1.1, 33-57.

[22]. Shi, Yuhui.(2001)."Particle swarm optimization: developments, applications and resources." Proceedings of the 2001 congress on evolutionary computation (IEEE Cat. No. 01TH8546). Vol. 1. IEEE.

[23]. Fister, Iztok, et al.(2013)."A comprehensive review of firefly algorithms." Swarm and Evolutionary Computation 13, 34-46.

[24]. Yang, Xin-She.(2020)."Firefly algorithm, stochastic test functions and design optimisation." International journal of bio-inspired computation 2.2 , 78-84.

[25]. Yang, Xin-She.(2010)."Firefly algorithm, Levy flights and global optimization." Research and development in intelligent systems XXVI. Springer, London, 209-218.

[26]. Mirjalili, Seyedali.(2019)."Genetic algorithm." Evolutionary algorithms and neural networks. Springer, Cham, 43-55.

[27]. Houck, Christopher R., Jeff Joines, and Michael G. Kay.(1995)."A genetic algorithm for function optimization: a Matlab implementation." Ncsu-ie tr 95.09 , 1-10.

[28]. Genlin, Ji.(2004) "Survey on genetic algorithm [J]." Computer Applications and Software 2 , 69-73.

[29]. Loiola, Eliane Maria, et al.(2007)."A survey for the quadratic assignment problem." European journal of operational research 176.2, 657-690.

[30]. Burkard, Rainer E., Stefan E. Karisch, and Franz Rendl.(1997). "QAPLIB–a quadratic assignment problem library." Journal of Global optimization 10.4 391-403.

[31]. Karaboga, Dervis, and Bahriye Akay.(2009)."A comparative study of artificial bee colony algorithm." Applied mathematics and computation 214.1, 108-132.

[32]. Karaboga, Dervis, et al.(2014)."A comprehensive survey: artificial bee colony (ABC) algorithm and applications." Artificial Intelligence Review 42.1, 21-57.

[33]. Ghandeshtani, Kambiz Shojaee, et al.(2010)."New simulated annealing algorithm for quadratic assignment problem." The fourth international conference on advanced engineering computing and applications in sciences.

[34]. Rutenbar, Rob A.(1989). "Simulated annealing algorithms: An overview." IEEE Circuits and Devices magazine 5.1 ,19-26.

[35]. Zhao, Juan, Zheng-Ming Gao, and Wu Sun.(2020)."The improved slime mould algorithm with Levy flight." Journal of Physics: Conference Series. Vol. 1617. No. 1. IOP Publishing.

[36]. Houssein, Essam H., et al.(2021)."Hybrid slime mould algorithm with adaptive guided differential evolution algorithm for combinatorial and global optimization problems." Expert Systems with Applications 174 , 114689.

[37]. Adithyan, T. Ajay, et al.(2017)."Nature inspired algorithm." 2017 international conference on trends in electronics and informatics (ICEI). IEEE.

[38]. Dixit, Manish, Nikita Upadhyay, and Sanjay Silakari.(2015)."An exhaustive survey on nature inspired optimization algorithms." International Journal of Software Engineering and Its Applications 9.4, 91-104.

[39]. Kar, Arpan Kumar.(2016). "Bio inspired computing–a review of algorithms and scope of applications." Expert Systems with Applications 59, 20-32.

[40]. Yang, Xin-She, et al.,(2013). Swarm intelligence and bio-inspired computation: theory and applications. Newnes.

[41]. Livingstone, David J., et al.(2008). Artificial neural networks: methods and applications. Totowa, NJ, USA:: Humana Press.
[42]. Kosko, Bart. (1994). "Fuzzy systems as universal approximators." IEEE transactions on computers 43.11 ,1329-1333.
[43]. Eiben, Agoston Endre, and Selmar K. Smit. (2011). "Evolutionary algorithm parameters and methods to tune them." Autonomous search. Springer, Berlin, Heidelberg, 15-36.
[44]. Fleming, Wendell H., and Raymond W. Rishel.(2012). “Deterministic and stochastic optimal control”. Springer Science & Business Media, Vol.1.
[45]. Beekman, Madeleine, and Tanya Latty. (2015). "Brainless but multi-headed: decision making by the acellular slime mould Physarum polycephalum." Journal of molecular biology, 3734-3743.
[46]. Brabazon, Anthony, and Seán McGarraghy. (2020) "Slime mould foraging: an inspiration for algorithmic design." International Journal of Innovative Computing and Applications, 30-45.
[47]. Lenstra, Jan Karel, and AHG Rinnooy Kan. (1975) "Some simple applications of the travelling salesman problem." Journal of the Operational Research Society, 717-733.
[48]. Zubaidi, Salah L., et al.(2020) "Hybridised Artificial Neural Network Model with Slime Mould Algorithm: A Novel Methodology for Prediction of Urban Stochastic Water Demand." Water ,2692.
[49]. Paz, Azaria, and Shlomo Moran. (1981) "Non deterministic polynomial optimization problems and their approximations." Theoretical Computer Science,251-277.