IHJPAS. 36 (4) 2023

Ibn Al-Haitham Journal for Pure and Applied Sciences
Journal homepage: jih.uobaghdad.edu.iq

Solving Quadratic Assignment Problem by Using Meta-Heuristic Search Method

Rawaa Abdulsattar
Department of Mathematics, College of Education for Pure Sciences, Ibn Al-Haitham University of Baghdad, Iraq

Iraq T. Abass *
Department of Mathematics, College of Education for Pure Sciences, Ibn Al-Haitham University of Baghdad, Iraq

Leong WJ
Department of Mathematics,
University Putra Malaysia (UPM)
Kuala Lampur 0060, Malaysia

***Corresponding Author:** Iraq.t@sc.uobaghdad.edu.iq

Article history: Received 14 January 2023, Accepted 20 February 2023, Published in October 2023

doi.org/10.30526/36.4.3195

Abstract

While analytical solutions to Quadratic Assignment Problems (QAP) have indeed been since a long time, the expanding use of Evolutionary Algorithms (EAs) for similar issues gives a framework for dealing with QAP with an extraordinarily broad scope. The study's key contribution is that it normalizes all of the criteria into a single scale, regardless of their measurement systems or the requirements of minimum or maximum, relieving the researchers of exhaustively quantifying the quality criteria. A Tbu Search Algorithm for Qadratic Asignment Poblems (TSQAP) is proposed, which combines the limitations of tabu search with a discrete assignment problem. The effectiveness of the proposed technique has been compared to well-established alternatives, and its operating principle is illustrated with a numerical example.
After repeating the solution of each issue (8) once and recording the algorithm results, it showed its agreement, once from a total (375) repetition of the experiment while the number of times the Artificial Bee Colony (ABC) arrived (2) as for the Firefly (FA) giving (117), also Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) gives (120) and the Tabu Search algorithm (174). The proposed technique (TSQAP) is shown to yield a superior solution with low computing complexity. MATLAB was used to generate all of the findings (R2020b).

Keywords: Tabu Search, Quadratic Assignment Problems, Assignment Problems.

1. Introduction

The Quadratic Problem Formulation (QAP) is one of the Complex Optimization Problems (COP) that, in terms of difficulty, belongs to the difficulties of type (NP-hard). Modeling commercial transactions and the problem of quadratic allocation was also applied in a variety of applications, including scheduling and site development for hospitals and universities[1].

Researchers have developed a suite of algorithms, including the Genetic Algorithm (GA), the Artificial Bee Colony (ABC), the Tabu Search Algorithm (TS), the Firefly (FA) Algorithm, and others, that have proven their effectiveness in finding good solutions to complex optimization problems in record time. Finding the best answer to a large-scale (n>20) quadratic assignment problem is challenging, and the absence of a polynomial-time technique to do so presents a challenge for researchers.

Due to the breadth of the problem being solution space, utilizing estimated algorithms to find a polynomial solution algorithm for the quadratic assignment problem in a reasonable amount of time is doubtful until it is shown that (NP = P) [2].

The overall goal of the research is to create a polynomial approximation algorithm that finds the optimal solution in polynomial time by, among other things, enhancing the efficiency of the tabu search algorithm (Tabu Search) to accelerate its progress towards a certain algorithm's current state of quality improvement.

2. Tabu Search

Tabu Search (TS) is one of the commonly utilized met heuristics for COP. Glover 1989 [3, 4, 5] and 6] proposed the core ideas of TS, which were built on earlier ideas in Glover 1990 [7]. Glover and Laguna 1993 [8, 9 and 10] describe the technique and its concepts. TS makes explicit use of the search history to avoid local minima and to conduct an exploration plan. Figure (1) depicts the approach for the algorithm.

```
Procedure basic TS
s ← GenerateInitialSolution()
TabuList ← ϕ
while termination conditions not met do
s ← ChooseBestOf(N(s)\ TabuList)
Update(TabuList)
endwhile
```

Figure 1: Algorithm Simple Tabu Search (TS)

As just a basic element, the simple TS algorithm employs a superior local search for improvement and employs a short attention span to eliminate local minima and cycles. The short-term memory is implemented as a list of tables that allows you to monitor the last alternatives you examined and prevents you from moving toward them. The current solution's neighborhood is so confined to solutions that do not belong on the list of violations [11 and 12].

3. Firefly Algorithm

Yang created the Firefly algorithm in 2008 by animating the typical movements of fireflies [13, 14, 15 and 16]. It is classed as swarm intelligent, met heuristic, and nature-inspired. In reality, the population of fireflies exhibits unique luminary flashing activities that serve as a means of attracting partners, communicating, and warning predators [17]. Yang developed this method based on the premise that all fireflies are unisexual, that all fireflies have appealing potential for each other, and that attractiveness is precisely proportional to the brightness level of individuals [18]. As a result, the brighter fireflies encourage the less bright ones to travel toward them, and if there are no fireflies brighter than a specific firefly, it moves randomly.

The objective function of the firefly algorithm is connected with the flashing light features of the firefly population. Given that light intensity is inversely quadratic proportional to the square of the area, this concept allows us to develop a suitable function for the distance between any two fireflies. Individuals in the population are compelled to make systematic or random moves in order optimize the fitting function [17, 19, 20 and 21]. This ensures that all fireflies gravitate toward more the appealing ones with brighter flashing until the population converges around the brightest one. The fireflies' algorithm is used in this operation and is controlled by three parameters: attraction, randomization, and absorption. The attractiveness parameter is defined using exponential functions and is based on the intensity of light between two fireflies. When this parameter is set to zero, the random walk associated with the randomization parameter, which is governed by the Gaussian distribution principle, generates a number from the [0,1] interval [17]. However, absorption variables have an effect on the value of attractiveness parameters as they change from zero to infinity. In the scenario of converging to infinity, the movement of fireflies appears to be a chaotic system [17]. The firefly optimization technique is illustrated briefly in Figure (2).

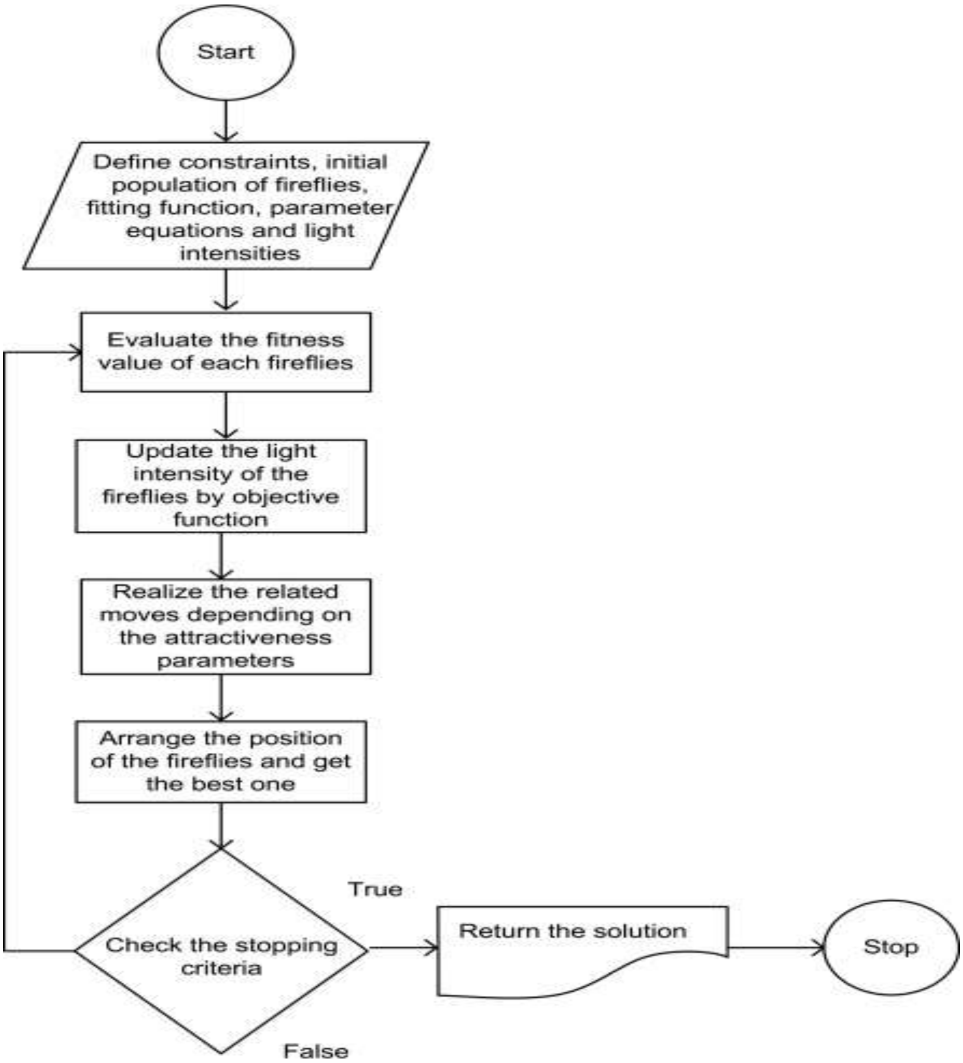

Figure (2): Flowchart of firefly algorithm technique.[22 and 23]

4. Artificial Bee Colony Algorithm

Dorigo et al. developed the ACO technique to inspire honey bee intelligence in their search for food [24, 25, 26, 27, 28, and 29]. Karaboga [30, 31, 32, 33, and 34] presents the ABC algorithm based on this metaheuristic technique. Bees are grouped into three primary groups in this technique based on their tasks: employed, observer, and scout. Scout bees are in charge of locating food sources, while hired and observer bees are in charge of exploitation work at the source. To review the tasking of the bees, scout bees examine the region for food sources and mark them at random without regard for the richness of the food supply. The worker bees then fulfill their jobs, such as selecting some nectar-rich sources. Depending on the quantity of nectar, they modify their dancing style as well as their speed to attract spectator bees waiting in the hive. Onlooker bees immediately decide to go toward the sources.

5. Genetic Algorithm

Using a model of biological processes, GAs are a type of optimization technique. As GA draws parallels to biological systems, it employs concepts like "chromosome," "selection," "crossover," and "mutation" [35, 36, 37, and 38]. The candidate solution to the optimization problem can be thought of as an array of n bits (loci) in length, with the size of the solution dictating the number of bits used. Chromosomes are a sequence of 0 and 1s in binary code, where n is the number of bits in the code. Conversely, selection, crossover, and mutation operators use probabilistic models that replicate the functions in biological processes to control the candidate solutions and speed up the convergence rate of the optimization process [39, 40, 41 42, and 43].

6. Formulation of the Problem

Assume we get a quadratic assignment problem having three matrices (F, D, C) of size (nxn), where F represents the diffusion matrix between amenities, D represents the distance matrix across sites, and C represents the cost matrix, i.e. the cost of allocating facilities to sites. They also have the vector, which holds all of the possible combinations of the elements $1, \ldots, n$, therefore the mathematical model for the quadratic assignment problem is as follows [2, 44, 45, 46, and 47].

$$min \sum_{i=1}^{n} \sum_{j=1}^{n} f_{i,j} d_{\pi(i),\pi(j)} + \sum_{i=1}^{n} c_{i,\pi(i)} \quad \ldots(1)$$

Where $(f_{i,j})$ is the flow between facility (i) and facility (j), $(d_{\pi(i),\pi(j)})$ represents the distance between the site $(\pi(i))$ and site $(\pi(j))$, and $(c_{i,\pi(i)})$ represents the fixed cost of allocating the facility(i) to the site $(\pi(i)$.

As a result of the quadratic assignment problem resulting in the modification of a portion of the problem's facilities, the cost-related portion of the problem is typically disregarded. Change it, and assuming the cost matrix does have a value of zero, the model becomes the following:

$$min \sum_{i=1}^{n} \sum_{j=1}^{n} f_{i,j} d_{\pi(i),\pi(j)} \quad \ldots..(2)$$

The goal is to find the permutation vector (π) of the mathematical model (1,2) which reduces the objective function to the least possible.

7. Quadratic Assignment Problem

A quadratic problem in specialized knowledge arises when trying to assign n facilities to n regions in such a way that the total cost of allocation is minimized, with transmission cost being calculated by incrementing the flow among each pair of resources by the distance between their assigned locations. Facilities or methods can be used at any place to guarantee the lowest possible energy- momentum for the least possible outlay of money [1, 48, 49, and 50].

8. The Proposed Algorithm (TSQAP)

We will begin by discussing the several QAP capabilities available. It then proceeds to demonstrate the algorithm's theoretical underpinnings. We then describe the method for allocating available resources. Environmental selection mechanisms and reproductive strategies are discussed last.

It has been suggested that the banned search algorithm (TS) could benefit from the following additions:

1- The (block list possession) algorithm's memory set is of the same size, which means that the memory entered to block an attribute is equal to the other memory.

2- The block memory has variable dimensions, which means it can be increased or lowered within the algorithm depending on the solution's direction.

3- Memory can be lowered to obtain quick solutions while condensing the solution and achieving good places.

4- The incorporation of storage for past events our ability to recall a large number of fabricated solutions relies on this memory. This reminder is based on four different concepts, the incident, the problem, the solution, and some attribute and it enables us to gain from all past solutions and some atributes in producing a new solution. Measures of how often, how well, and how significantly something affects you. The quality memory is based on the achievement of the objective function, the brightness memory is based on the number of times this very same solution or attribute is replicated, and the event memory is based on the recording of each way to solve or characteristic from the last incarnation in which the solution occurred. The density of the solution and the duration of the created solution's effects on the improvement are both recorded.

There are undoubtedly many benefits to using this algorithm, and it does get pretty close to the optimal solution on a regularly basis. On the other hand, it does take a little longer than other algorithms, and the fact that it discards some solutions and doesn't use them again lessens the value of diversification, even after the addition of long-term memory. Last but not least, the programmer might tailor the algorithm's design to the problem's data. Removing elements that are irrelevant to the topic at hand [3].

9. Multi-Memory Tabu Search Algorithm Procedures

The algorithm procedures as follows:

a- Generate a specific solution.
b- Calculate the value of the objective function of the current solution Z(S).
c- Make S $\rightarrow$ X and Z(S) $\rightarrow$ Z(X).
d- Tabu list $(TL1, TL2, \ldots.TLn)$ $\rightarrow$ $\emptyset$.
e- The process is repeated.
f- Generate a new solution X' by moving a given neighborhood N(S) from the current solution that does not exist in $1, TL2, \ldots.TLn$.
g- Recording the best solution found in $TL1, TL2, \ldots.TLn$ and delete the old solution.
h- Calculate the value of the objective function for the new substitution $Z(X')$.
i- If Z(S) < Z(X') make X' $\rightarrow$ S the conditional.
j- Repeat the process until the stopping condition is met and print the best solution that was found.
k- End.

The steps to make the forbidden search algorithm used to solve the quadratic assignment problem as follows:

a- Entering the issue of the issue in the form of descriptions in size (nx) where (n) represents the size of the issue the flow matrix (F) and the distance matrix (D) include a memory matrix that records the frequency for each solution that has been transferred $(Frequancy = \emptyset)$ and made it the size of the issue.
b- Solving a preliminary solution (S) is complete solution to the issue with the calculation of the goal function for the first solution according to the equation (2).
c- Make the solution $(S = S^*)$ and it is necessary to make $(f^* = f(S))$.
d- Repetition meter set $(i = 1)$.
e- The creation of a preliminary workaround in the style of neighborhood research) N (s) (and here the functionality of order switching will be used by optional points that are not registered in the prohibition remembrance and their sites are replaced and a new solution is replaced and then the search process in the prohibition memory and the following form shows a mutation of a change. Ranks [9]:

Before

2	4	7	1	8	3	5	6

After

2	6	5	3	4	7	1	8

Table (1): Ranking change mutation

f- Repetition ($i = i + 1$).

g- The stopping of the stops is continuing with steps (e, f) until the condition of stopping is achieved with the end of the number of total repetitions.

10. Simulation Experiment and Meanwhile Analysis

The problem of quadratic assignment has become a measure of the efficiency of algorithms due to the difficulty of finding an optimal solution in a suitable time, so any method that is developed to solve the problem must be applied to standard problems taken from the library of the quadratic assignment problem (QAPLIB) [9]. Standard problems taken from the library with the solution repeated several times once and calculating the results based on the algorithm's efficiency law, which is [9]:

$$Gap = \frac{X-opt}{opt} \times 100\% \; \ldots.(3)$$

Such that:

X: the solution found.

Opt: the best solution to the problem.

In this paragraph, 8 standard problems were solved, taken from the virtual library of the quadratic assignment problem using the Tabu Sarch algorithm (TS), after repeating the solution of each problem (25) times and recording the results by calculating the mean gap of approach (GAP) and standard deviation, Min and Max that was found by comparing the results with Artificial Bee Colony (ABC), Firefly (FA), Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) the Tabu Search algorithm outperformed other algorithms for most of the results, as the results of reaching the optimal solution for each algorithm were as follows in a Table (2):

Table (2): Shows the details of the system used to implement the algorithm

Name	**Type**
Operating system	Windows 10 Professional
Programming language	Matlab 2020
C.P.U	Intel Pentium Celeron 2.9 GHZ
RAM	3048 MB

After repeating the solution of each issue (8) once and recording the algorithm results, it showed its agreement, once from a total (375) repetition of the experiment while the number of times the local research algorithm arrived (2) as for the algorithm of the giving birth simulation (117) and the Tabu Search algorithm (174), we can see the details in a Table (3) below:

Table (3): The number of times to reach the optimal solution for each algorithm

FA	2/375	13.34
ABC	117/375	5.06
GA and PSO	120/375	7.05
TS	174/375	1.21

Now we can see some graphics of random selection of the existing problems.

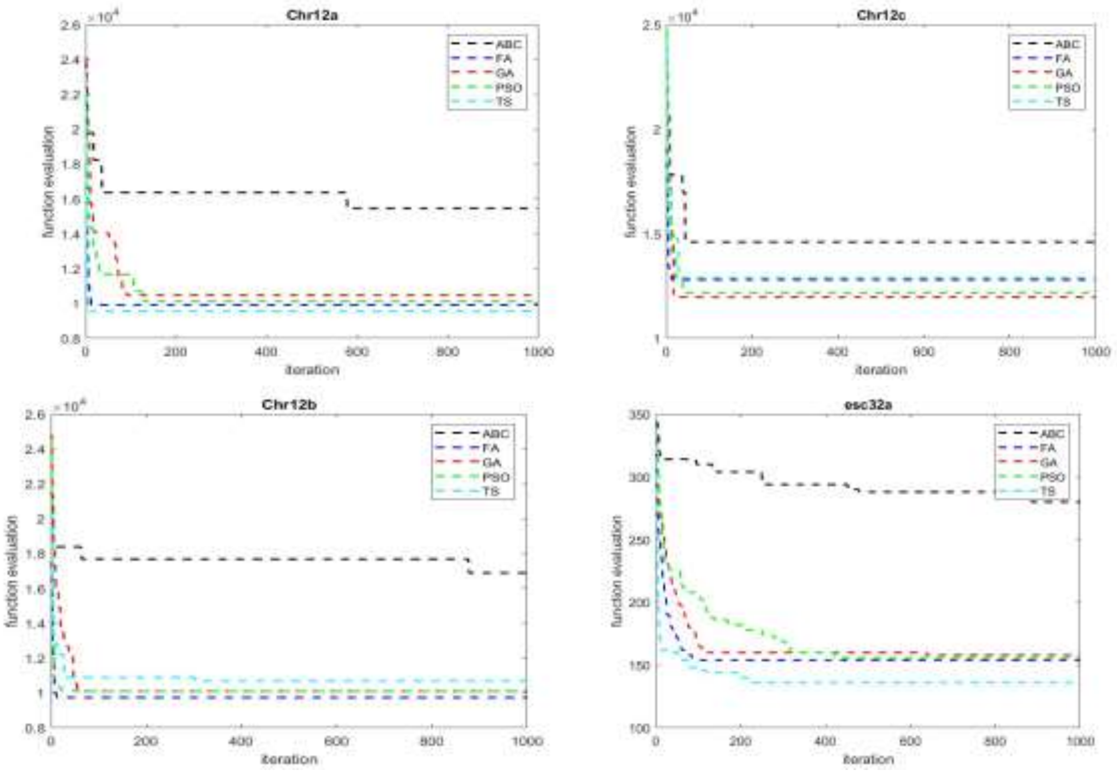

Figure (2): Shows the results of the approach gap

Table (4) Shows the Results of the Comparison Between Algorithms

Problems	Optimal Sol		ABC	FA	GA	PSO	TSQAP
Chr12a	9552	Mean	10232.00	13889.00	10904.00	10791.00	**1019**
		SD	449.29	138.960	829.56	707.75	**307.75**
		Min	9552.00	11734.00	9552.00	9916.00	**9216**
		Max	11688.00	16744.00	12190.00	12174.00	**10174**
Chr12b	9742	Mean	9904.00	12687.00	10371.00	11411.00	**9741**
		SD	183.75	1355.20	1150.00	2305.80	**110.58**
		Min	9742.00	11326.00	9742.00	9742.00	**9542**
		Max	10102.00	16860.00	13170.00	16472.00	**10072**
Chr12c	1115	Mean	11761.00	14893.00	12436.00	12644.00	**1064**
		SD	**527.71**	1297.80	594.38	939.52	939.52
		Min	**11156.00**	13030.00	11186.00	**11156.00**	**1113**
		Max	12806.00	17706.00	13316.00	15022.00	**11022**
Chr15a	9896	Mean	11916.00	18067.00	12979.00	11877.00	**1087**
		SD	1174.70	1523.40	1910.70	1346.30	**104.63**
		Min	9936.00	14744.00	10106.00	**9896.00**	**9895**
		Max	13856.00	20398.00	16686.00	15152.00	**12152**
Chr15b	7990	Mean	10050.00	16628.00	11865.00	10861.00	**1003**
		SD	845.16	1504.50	1442.90	929.29	**729.29**
		Min	8640.00	14674.00	9902.00	**7056.00**	**7056**
		Max	12126.00	19412.00	14976.00	12750.00	**12150**
Chr15c	9504	Mean	11944.00	18393.00	13220.00	12784.00	**1117**
		SD	1466.90	1505.50	1307.70	**992.11**	**992.11**
		Min	9504.00	16098.00	10644.00	11400.00	**9240**
		Max	14990.00	21006.00	15036.00	15344.00	**12344**
Chr18a	1109	Mean	14778.00	28298.00	16599.00	**13607.00**	**1360**
		SD	1747.50	1899.70	2313.20	2092.40	**169.24**
		Min	**12320.00**	24010.00	11870.00	12804.00	1280
		Max	19196.00	31338.00	21814.00	20652.00	**10652**
Chr18b	1534	Mean	1584.80	2134.80	1600.20	1650.10	**1350**
		SD	59.67	109.50	53.68	71.99	**419.90**
		Min	1534.00	1978.00	1534.00	1534.00	**1434**
		Max	1750.00	2306.00	1732.00	1802.00	**15020**

11. Conclusions

In this paper, several conclusions were reached, which can be summarized as follows:

1- The use of the first stage of the algorithm and the directed search feature helps to greatly improve the memory of the Tabu Search (TS).

2- These results in a quick convergence of the optimal solution to the problem, where the percentage of reaching the optimal solution in the first stage of the algorithm was approximately (75%).

3- The extent of the algorithm’s ability to reach the optimal solution, especially in solving Complex Optimization Problems (COP), depends mainly on the algorithm’s ability to diversify the search for the solution, as well as diversify solutions and the speed of reaching the optimal solution depends on the criteria of rejection and acceptance of the new solution because these criteria help determine The search space is in the issue, so whenever this space is reduced, the speed of the algorithm in reaching is faster and also the same higher diversification.

4- The percentage of access of the Tabu Search (TS) Algorithm to the optimal solution was (75.2%), and the Bee Colony algorithm was (31.2%), while the Firefly and Genetic algorithms were the worst if the percentage of reaching the optimal solution was (0.5%).

References

1. Hammadi, A. M. K., & Ibrahim, H. A. A. .Proposed Combinatorial Algorithms for Solving Quadratic Assignment Problems. *Diyala Journal for Pure Science*, (**2016**). 12(3-part 1),

2. Al-Bayati, Hassan Abdel-Sattar, "Using the Compound Harmonious Search Algorithm to Solve the Quadratic Specialization Problem with Practical Application," Master's thesis, College of Administration and Economics, University of Baghdad, **2015**.

3. Glover F. -Tabu Search, Part I - ORSA Journal on Computing (**1989**).1, pp. 190-206,

4. Sheah, R. H., & Abbas, I. T. Using Multi-Objective Bat Algorithm For Solving Multi-Objective Non-Linear Programming Problem. *Iraqi Journal of Science*, (**2021**). 997-1015.

5. Sattar, H. A., & Cheetar, AA New Strategy Based on GSABAT to solve single objective optimization problem. *International Journal of Swarm Intelligence Research (IJSIR)*, . (**2019**). *10*(3), 1-22.

6. Atiya, B.; Bakheet, A. J. K.; Abbas, I. T.; Bakar, M.; Abu, R.; Soon, L. L.; Monsi, M. B.. Application of Simulated Annealing to Solve Multi-Objectives for Aggregate Production Planning. In *AIP Conference Proceedings*(**2016**, June) (Vol. 1739, No. 1). AIP Publishing.

7. Glover F. -Tabu Search, Part II - ORSA Journal on Computing(**1990**). 2, pp. 4-32,

8. Glover F.; Taillard E.; Laguna M., de Werra D. -Tabu Search -Volume 41 of the Annals of Operations Research, (**1993**).

9. Burkard, R.E.; Karisch, S.E.; Rendl, F., "QAPLIB – a Quadratic Assignment Problem Library", *Journal of global optimization*, **1996**, 10(4), 391-403,.

10. Zaidan, A. A.; Atiya, B.; Abu Bakar, M. R.; Zaidan, B. B A New Hybrid Algorithm of Simulated Annealing and Simplex Downhill for Solving Multiple-Objective Aggregate Production Planning on Fuzzy Environment. *Neural computing and applications*, . (**2019**). *31*, 1823-1834.

11. Balicki, Jerzy. "Tabu Programming for Multiobjective Optimization Problems." International *Journal of Computer Science and Network Security* 7.10: 44-51, (**2007**).

12. Hansen, Michael Pilegaard. "Tabu Search for Multiobjective Optimization: MOTS." Proceedings of the 13th International Conference on Multiple Criteria Decision Making, (**1997**).

13. Yang X-S. Nature-Inspired Metaheuristic Algorithms. Beckington, UK*: Luniver Press*; **2008**.

14. Bakar, M. R. A.; Abbas, I. T.; Kalal, M. A.; AlSattar, H. A.; Bakhayt, A. G. K.; Kalaf, B. A. Solution for Multi-Objective Optimisation Master Production Scheduling Problems Based on Swarm Intelligence Algorithms. *Journal of Computational and Theoretical Nanoscience*, (**2017**). *14*(11), 5184-5194.

15. Abu Bakar, M. R., Bakheet, A. J. K., Kamil, F., Kalaf, B. A., Abbas, I. T., & Soon, L. L. Enhanced Simulated Annealing for Solving Aggregate Production Planning. *Mathematical Problems in Engineering*, **2016**.

16. Abdul-Zahra, I., Abbas, I. T., Kalaf, B. A., Bakar, R. A., June, L. W., & Monsi, M. B. The Role of Dynamic Programming in the Distribution of Investment Allocations between

Production Lines with an Application. *International Journal of Pure and Applied Mathematics*, (**2016**). *106*(2), 365-380.

17. Fister I, Yang X-S, Brest J. A Comprehensive Review of Firefly Algorithms. Swarm Evol Comput **2013**;13:34_46.

18. Yang X-S. Firefly Algorithms for Multimodal Optimization. International symposium on stochastic algorithms. Sapporo, Japan: Springer Berlin Heidelberg; **2009**.

19. Dey, N., Chaki, J., Moraru, L., Fong, S., & Yang, X. S. Firefly Algorithm and its Variants in Digital Image Processing: A Comprehensive Review. *Applications of Firefly Algorithm and Its Variants: Case Studies and New Developments*, (**2020**). 1-28.

20. Kumar, V., & Kumar, D. A Systematic Review on Firefly Algorithm: Past, Present, and Future. *Archives of Computational Methods in Engineering*, (**2021**). *28*, 3269-3291.

21. Nayak, J., Naik, B., Pelusi, D., & Krishna, A. V. A Comprehensive Review and Performance Analysis of Firefly Algorithm for Artificial Neural Networks. *Nature-Inspired Computation in Data Mining and Machine Learning*, (**2020**). 137-159.

22. Xiong, T., Bao, Y., & Hu, Z. Multiple-Output Support Vector Regression with a Firefly Algorithm for Interval-Valued Stock Price Index Forecasting. *Knowledge-Based Systems*, (**2014**). *55*, 87-100.

23. Lolla, P. R.; Rangu, S. K.; Dhenuvakonda, K. R.; Singh, A. R. A Comprehensive Review of Soft Computing Algorithms for Optimal Generation Scheduling. *International Journal of Energy Research*, (**2021**). *45*(2), 1170-1189.

24. Dorigo M. Optimization, learning and natural algorithms. PhD thesis. Italy: Dipartimento di Elettronica, *Politecnico di Milano*; **1992** [in Italian].

25. Dorigo, M.; Maniezzo, V., Colorni A. An System Optimization by a Colony of Cooperating Agents. *IEEE Trans Syst Man Cybern*: B **1996**;26:29_41.

26. Dorigo, M.; Birattari, M., ; Stutzle, T. (**2006**). Ant Colony Optimization. *IEEE computational intelligence magazine*, *1*(4), 28-39.

27. Yang, X. S. (2020). *Nature-inspired optimization algorithms*. Academic Press.

28. Kotary, J., Fioretto, F., Van Hentenryck, P., & Wilder, B. (**2021**). End-to-end Constrained Optimization Learning: A survey. *arXiv preprint arXiv:2103.16378*.

29. Yang, X. S. (**2020**). Nature-Inspired Optimization Algorithms: Challenges and open problems. *Journal of Computational Science*, *46*, 101104.

30. Karaboga D. An Idea Based on Honey Bee Swarm for Numerical Optimization. Technical report-tr06, Vol. 200. Kayseri, Tu¨rkiye: Erciyes University; **2005**.

31. Simon, D. (**2013**). *Evolutionary optimization algorithms*. John Wiley & Sons.

32. Van den Bosch, A. Wrapped Progressive Sampling Search for Optimizing Learning Algorithm Parameters. In *Proceedings of the Belgium-Netherlands Conference on Artificial Intelligence, BNAIC'04, 21-22 oktober*, ***2004*** (pp. 219-226). Unknown Publisher.

33. Yu, F., & Xu, X. A Short-Term Load Forecasting Model of Natural Gas Based on Optimized Genetic Algorithm and Improved BP Neural Network. *Applied Energy*, (**2014**). *134*, 102-113.

34. Zang, H., Zhang, S., & Hapeshi, K. A Review of Nature-Inspired Algorithms. *Journal of Bionic Engineering*, (**2010**). *7*(4), S232-S237.

35. Mitchell M. An Introduction to Genetic Algorithm. Cambridge: The MIT Press; **1999**.

36. Thede, S. M. An Introduction to Genetic Algorithms. *Journal of Computing Sciences in Colleges*, (**2004**). *20*(1), 115-123.

37. Han, S., & Xiao, L. An Improved Adaptive Genetic Algorithm. In *SHS Web of Conferences*(**2022**). (Vol. 140, p. 01044). EDP Sciences.

38. Niu, X. F., & Ma, W. P. An Improved Multiple Populations Quantum Genetic Algorithm. *Laser Physics Letters,* (**2022**). *19*(9), 095203.

39. Zalzala AMS, Fleming PJ. Genetic Algorithms in Engineering Systems. London: Institution of Electrical Engineers; **1997**.

40. Khatri, K. A., Shah, K. B., Logeshwaran, J., & Shrestha, A. Genetic Algorithm Based Techno-Economic Optimization of an Isolated Hybrid Energy System. *CRF*, (**2023**). *1*(1), 1.

41. Sabeti, V., Sobhani, M., & Hasheminejad, S. M. H. An Adaptive Image Steganography Method Based on Integer Wavelet Transform Using Genetic Algorithm. *Computers and Electrical Engineering*, (**2022**). *99*, 107809.

42. Lotf, J. J., Azgomi, M. A., & Dishabi, M. R. EAn Improved Influence Maximization Method for Social Networks Based on Genetic Algorithm. *Physical A: Statistical Mechanics and its Applications*, . (**2022**). *586*, 126480.

43. Liu, Q., Wang, C., Li, X., & Gao, L. An Improved Genetic Algorithm with Modified Critical Path-Based Searching for Integrated Process Planning and Scheduling Problem Considering Automated Guided Vehicle Transportation Task. *Journal of Manufacturing Systems*, (**2023**). *70*, 127-136.

44. Burkad.R,,Dell'Amico.M and Martello.S., ."Assignment Problems", Society for Industrial and Applied Mathematics ,Philadephia, .(**2009**).

45. Tang, M., Ji, B., Fang, X., & Yu, S. S. Discretization-Strategy-Based Solution for Berth Allocation and Quay Crane Assignment Problem. *Journal of Marine Science and Engineering*, (**2022**). *10*(4), 495.

46. Liu, L., Yang, J., Kong, X., & Xiao, Y. Multi-Mission Selective Maintenance and Repairpersons Assignment Problem with Stochastic Durations. *Reliability Engineering & System Safety*, (**2022**). *219*, 108209.

47. Zhu, L., Zhou, Y., Jiang, R., & Su, Q. Surgical Cases Assignment Problem Using A Multi-Objective Squirrel Search Algorithm. *Expert Systems with Applications*, (**2023**). 121217.

48. Raheem, S. H., Kalaf, B. A., & Salman, A. N. Comparison of Some of Estimation Methods of Stress-Strength Model: R= P (Y< X< Z). *Baghdad Science Journal*, (**2021**). *18*(2 (Suppl.)), 1103-1103.

49. Kalaf, B. A. A New Algorithm to Estimate the Parameters of Nonlinear Regression. In *Journal of Physics: Conference Series* (**2021**, May). (Vol. 1879, No. 3, p. 032042). IOP Publishing.

50. Mahdi, G. J., Kalaf, B. A., & Khaleel, M. A.. Enhanced Supervised Principal Component Analysis for Cancer Classification. *Iraqi Journal of Science,* (**2021**) 1321-1333.