International Journal of Electrical and Computer Engineering (IJECE)
Vol. 9, No. 3, June 2019, pp. 2064∼2074
ISSN: 2088-8708, DOI: 10.11591/ijece.v9i3.pp2064-2074

Parallel hybrid chicken swarm optimization for solving the quadratic assignment problem

Soukaina Cherif Bourki Semlali, Mohammed Essaid Riffi, Fayçal Chebihi
Department of mathematics, Faculty of Sciences, University of Chouaib Doukkali, El Jadida, Morocco

Article Info

Article history:
Received Aug 4, 2018
Revised Nov 23, 2018
Accepted Dec 20, 2018

Keywords:
Quadratic assignment problem
Metaheuristics
Hybrid Chicken swarm optimization
Tabu search
QAPLib

ABSTRACT

In this research, we intend to suggest a new method based on a parallel hybrid chicken swarm optimization (PHCSO) by integrating the constructive procedure of GRASP and an effective modified version of Tabu search. In this vein, the goal of this adaptation is straightforward about the fact of preventing the stagnation of the research. Furthermore, the proposed contribution looks at providing an optimal trade-off between the two key components of bio-inspired metaheuristics: local intensification and global diversification, which affect the efficiency of our proposed algorithm and the choice of the dependent parameters. Moreover, the pragmatic results of exhaustive experiments were promising while applying our algorithm on diverse QAPLIB instances. Finally, we briefly highlight perspectives for further research.

Corresponding Author:
Soukaina Cherif Bourki Semlali,
Laroseri Laboratory, Department of mathematics,
Faculty of Sciences, University of Chouaib Doukkali, El Jadida, Morocco.
Email: Soukaina.cherif.b.s@ucd.ac.ma

1. INTRODUCTION

Combinatorial optimization problems are an substantial paradigm [1] and an important challenge, a lot of algorithm has been applied in several fields of real life such as engineering and economics. One of the categories includes problems with continuous variables, then this refer to handling the discrete optimization, a simple example of this purview is to find the minimum or the maximum of a given function f under a set of constraints [2].

In the last two decades, nature has been a source of inspiration for many scientists. Many contemporary metaheuristic algorithms have been stemming from the behavior of biological systems in nature [3]. especially those who interact with each other in a swarm ([4],[5]). Some examples of studies are also proposed to shed light on the behavior of swarm intelligence which are stretched from the nature: bee algorithms[6] [7], harmony search [8], ant colony optimization [9] [10], bat algorithm [11], particle swarm optimization [12], firefly algorithm [13] and cuckoo search [14].

The quadratic assignment problem (QAP) is one of the most classical NP-hard combinatorial optimization problem [15]. There are no exact algorithms which could be used to solve large instances, QAP was initiated at the first time by koopmans and beekmann form [16], this problem has been drawing many scientists attention. The aim of the problem is to minimize the objective function $QAP(A, B, \phi)$ in order to organize the formulation of a given facilities that should be optimally assigned to a given locations. Given two matrices of size n×n called A=a_{ij} and B=$b_{\phi(i)\phi(j)}$ as A refers to the flows between pairs of facilities and B refers to the distance of their locations $\phi(i)$ and $\phi(j)$. The quadratic assignment problem can be depicted by the classical formulation version of Koopmans-Beckmann as:

$$min_{\phi \in \prod} QAP(A, B, \phi) = \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij}.b_{\phi(i)\phi(j)} \tag{1}$$

Where the function denotes the cost assignment of n facilities to n locations and $\prod$ is the set of all possible permutations, $\phi \in \prod$ is the permutation that minimizes the cost function, furthermore each solution is encoded by a permutation.

The fitness landscape was introduced in 1932 [17] to describe the behaviors of evolutionary algorithms, then by predicting its performance it help to design it. The fitness landscape (S, f, d) of any instance of the quadratic assignment problem consists of a set of solutions S, a fitness function $f : S \rightarrow \Re$, and a distance measure d of the spatial structure of the landscape. Moreover, the neighborhood operator N is applied in order to convert a solution $s \in S$ into a new solution S'. the search space can be generally represented as $G = (S, E)$ where S is the vertices and E is the edges of the graph.

The concept of fitness landscapes [17] has been shown to help understanding the behavior of combinatorial optimization algorithms. Thus, three different classes of instances as shown in Figure 1 could be defined by the fitness landscape analysis techniques, the entropy and the distribution of the local optima ([18],[19]), the first type of instances is the Nonstructured Landscape which all have a uniform random nature by applying a local search on this instances good solutions will be quickly found, the second landscape is the Massif central which surround only real instances then the use of tabu search allows to find a promising results and finally the Multimassif instances which comprises only large instances.

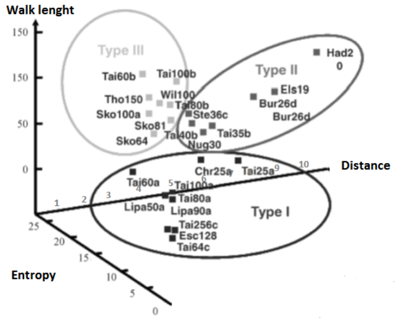


Figure 1. Different classes of instances by the Correlation between the walk length and the distribution entopy-distance [20]

Another classification is proposed according to Taillard [21], the classification includes four classes. The first classe is the unstructured and randomly generated instances; the second classe is the instances with the distance matrix based on the Manhattan distance on a grid; then the real-life instances; and finally the instances which are randomly generated and based on the structure of the real-life instances.

In this paper, we introduce a new hybrid Chicken Swarm Optimization (PHCSO) by enhancing the search strategy of the original CSO algorithm. The first step will be the use of the constructive procedure of GRASP by threads to create the popoluation and then we apply a new modifications to the operations and the operators of CSO (position representation and its update equation). The second step will be the incorporate of the random pairwise 2-way perturbation neighborhood mechanism as a mutation process in order to improve the movement equation of rooster. In the third step, we integrate a modified sequential constructive crossover operator in the movement equation of chicks. Finally, we will evaluate the quality of our approach by applying a fitness landscape analysis and a propabilty distribution , in order to ensure the balance between intensification and diversification capabilities of PHCSO. We will compare the performance of our proposed approach to other algorithms from the literature in term of run time and quality of solution.

The remainder of the paper is divided into different sections: The second section provides an overview of the related work of diverse QAP applications. In the third section, we give a brief description of the chicken swarm algotrithm used in this contribution. In the fourth section, we describe the tabu search, the constructive procedure of GRASP and 2-way perturbation neighborhood mechanism. In the fifth section, we present the new adapted method. In the next section, we provide the parameters used and the results of the final offspring obtained of this contribution. Finally, in the last section, we close this paper with a conclusion and the prospects.

2. RELATED WORK

Since the apparition of the mathematical model related to economic activities, Quadratic assignment problem have been ubiquitously supervening to solve several problems and various practical applications [22].

Recently several researchers around the world have demonstrated a huge interest to QAP, some of them have conducted a survey on theory and applications techniques used in order to solve this problem [23], QAP models and presents various theoretical and practical problems [24] such as the graphical partitioning, data analysis, hospital layout [25], image synthesis, typewriter keyboard design [26] and the problem of linear arrangement. We can notice that many NP-hard combinatorial optimization problems can be sculptured as QAP. For example, memory layout optimization in signal processors [27], VLSI design, campus planning model, dartboard design and scheduling process.

Some of the popularly studies have been worn out techniques for solving QAP such as Greedy 3-Opt [28], Ant Colony Optimization (ACO) [29], Genetic Algorithms (GA) [30], Scatter Search (SS)[31], Iterated local search algorithm [32], simulated annealing [33], Greedy Randomized Adaptive Search Procedure (GRASP)

Numerous natural inspired heuristics is applied to solve QAP and available as a guideline for improvement, among the contributions that have reached widespread success are: a new recombination operator for the genetic algorithm [34], the Backboard Wiring Problem by steinberg in 1961 [35], Heffley used the quadratic assignment problem to solve the best integer assignment and a set of sustaining prices in 1980 [36], Geoffrion and Graves in 1976 used the QAP to solve the problem of Scheduling Parallel Production Lines with Changeover Costs [37], Benjaafar in 2002 introduced a new methods for designing the next Generation Factory Layouts [38], Wess and Zeitlhofer in 2004 applied to solve the phase coupling problem between data memory layout generation and address pointer assignment [27].

3. CHIKEN SWARM OPTIMIZATION

Chicken swarm optimization (CSO) is a new stochastic algorithm introduced in 2014 [39]. CSO is based on the behavior of chickens, which simulates the hierarchical order in the swarm while searchinf for food. The approach of the algorithm spreads over the following four hypothesis:
(1) Each subgroup of the swarm includes a Roosters, hens and several chicks.
(2) The best fitness value helps to determine the subgroups and the type of each individual, the best fitness value denotes the roosters. In the other side, the worst fitness value defined the chicks. Moreover, the rest of individuals are chickens which randomly select their groups and the mother relationship with chicks. Then, each chick follows their mothers for seeking food.
(3) The hierarchical order and the relations between each mother and their childs are updated after a few generations.
(4) Individuals in each subgroup follow their group-leader to search for food, but may also steal good food that have been found by other individuals.
Each position depict aslo a given solution of our problem.

The first equation is the position update equation of the rooster:

$$x_{i,j}^{t+1} = x_{i,j}^{t} * (1 + Randn(0, \sigma^2)) \tag{2}$$

$$\sigma^2 = \left\{ \begin{array}{l} 1,\ if\ f_i \leq f_k, \\ \exp(\left(\frac{f_k - f_i}{|f_i| - \varepsilon}\right) \end{array} \right. otherwise\ k \in [1, N], k \neq i \tag{3}$$

Where K is the index of the rooster (the leader of the group) randomly selected and specified by the Gaussian distribution $Randn(0, \sigma^2)$ as the standard deviation is σ^2, in this case f is the fitness value of the current rooster.

The second equation is the position update equation of the hens:

$$\mathrm{x}^{t+1}_{i,j} = x^t_{i,j} + S1 * Rand * (x^t_{r1,j} - x^t_{i,j}) + S2 * Rand * (x^t_{r2,j} \text{ - } x^t_{i,j}) \tag{4}$$

Where $S1 = \exp(\left(\frac{f_i - f_{r1}}{|f_i| + \varepsilon}\right))$
and $S2 = \exp((f_{r2}$ -$f_i)$ $Finally, the position update equation of the chicks$:

$$x^{t+1}_{i,j} = x^t_{i,j} + FL * (x^t_{m,j} - x^t_{i,j})) \tag{5}$$

Where $m \in [1, N]$ represents the chick's mother and $FL \in [0, 2]$ is a randomly selected parameter that denotes the relationship between the chicks and its mother.

In 2015 Dinghui Wu [40], the position update equation of the chicks has been strengthened by considering the learning factor C of the rooster r and the self-learning factor W which ensures the approach of diversification in large space, the new equation of chicks is surrogated as follows:

$$\mathrm{x}^{t+1}_{i,j} = W * x^t_{i,j} + FL * (x^t_{m,j} - x^t_{i,j})) + C * (x^t_{r,j} \;\; x^t_{i,j})) \tag{6}$$

4. THE CONSTRUCTIVE PROCEDURE OF GRASP

The greedy randomized adaptive search procedure (GRASP) is an iterative method for combinatorial optimization problems [41], many application of GRASP for the quadratic assignment problem was presented in the literature [42] [43].

GRASP is an iterative process, each iteration goes through two phase the first one is a construction phase, in which a possible solution with good quality is generated. The second phase is the local search, a local optimum could be found in the neighborhood of the feasible solution [44].

In this paper, we will accurately use the construction phase of GRASP [42], which includes two stages in order to provide better initial solutions for the new approach, allowing the swarm to converge more quickly towards an optimal solutions. In another side two components could illustrate the quality of this approach, the adaptive Greedy Measures and the randomized selection of assignments which evaluate the choise of each possible assignment and then garauntee a good intensification strategy [45].

The first stage of construction phase proposed by Li et al. selects two pairs of facility-location then the $n^2 - n$ distance entries in D are sorted in an ascending order and the smallest integer $[\beta(n^2 - n)]$ are selected. In each cycle on a graph of QAP, each facility-location is a vertex, the fully connected graph for n elements has $n^2 - n$ edges. On the other hand, the $n^2 - n$ flow entries in F are sorted in descending order and the largest integer $[\beta(n^2 - n)]$ are selected. In the next step, the cost $f_{ij} d_{kl}$ are sorted in increasing order by keeping the smalest $[\alpha\beta(n^2 - n)]$ elements, where $1 \leq i, j, k, l \leq n; i \neq j; k \neq l$. The percentage α and β will be defined as the candidate restriction parameters of GRASP ,where $\alpha \in [0; 1]$ and $\beta \in [0; 1]$.

The greedy function allows to select the couple facility-location which have the smallest $f_{i,\pi_s}.d_{k,s}$ from the $[\alpha\beta(n^2 - n)]$, where $i \notin A$ to location k, we note that A is the set of already-assigned facilities, its value change at each iteration, then it will be computed as follow:

$$l_{i,k} = \sum_{s \in A} f_{i,\pi_s}.d_{k,s} \tag{7}$$

The candidate restriction limits RCL$_k$ for location k is depicted as:

$$RLC_k = l_{i,k} \leq min(l_{i,k}) + \alpha(max(l_{i,k}) - min(l_{i,k}))/i \notin A \tag{8}$$

In the stage 2, two facility-location pairs (j_1, l_1) and (j_2, l_2) are randomly selected from the $[\alpha\beta(n^2 - n)]$ sorted pairs. Finally, the rest facilities are assigned to the rest of locations by respecting the greedy function.

5. TABU SEARCH

A neighborhood search algorithm looks for the neighbors of a current solution in order to find a better one. The main concern of this type of algorithm is that it can easily get caught in a local optimum. Accordingly, the use of a Tabu search can help to avoid this problem. Tabu search was introduced by Glover [46] in order to avoid the local optimum. Afterwards, Glover presented [47] the essential basis which are enlarged by a new strategy of tabu search for solving combinatorial optimization problems. In 1995, taillard offers a comparison of different tabu search strategy [21]. Many application of tabu search is adapted to solve combinatorial optimization problems especially for the quadratic assignment problem, such as: the robust tabu search (Ro-TS)[48], a parallel adaptive tabu search approach [49], a modified version of the tabu search algorithm [50] which unified many approaches as intensification and diversification (I&D approach). The aim key of tabu search is to allow moves of S a set of solutions even if ther is no improving neighboring solution where $s \in N(s)$, thus it's necessary to choose a random permutation as an initial solution, then by considering π as the current solution among $\phi(\pi)$ set of all possible permutations, $\phi(\pi)$ illustrate as well the set of neighbour solutions of π, the random perturbations are applied to diversify the probabilty in the search space, then $\Delta(\pi, r, s)$ is defined as the value of a move or the cost of permutation from the location π_r to the location π_s. All the moves is stored in a tabu list T, then the information is memorisied on the last h moves or the tabu list size. The tabu list is lightened during the process by discarding the oldest solution based on the FIFO mechanism (First In First Out), since the size of the the memory is limited which makes it impossible to record all the solutions found. Then, in the symmetric case, the mathematical equation of the cost is represented as follows:

$$\Delta(\phi, r, s)=2.\sum_{k\neq r,s}^{n}(a_{sk} - a_{rk})(b_{\phi(s)\phi(k)} - b_{\phi(r)\phi(k)}) \tag{9}$$

Then it is easy to compute the costs of all possible permutations in $0(n^2)$ time.

The move from the current solution π_i to the neighboring solution π_j is defined by 2-exchange neighborhood function $N_2 : \pi \rightarrow 2^\pi$ and $N_2(\pi) = (\pi'/\pi' \in \phi, d(\pi, \pi') = 2)$ where d is the distance between solutions, otherwise 2-exchange neighborhood operator $p_{ij} : \pi \rightarrow \pi'(i, j = 1, 2..n)$ allows to exchange ith and jth elements in the current permutation then $\pi' = \pi \oplus p_{ij}$.

6. MULTIPLE THREADS

In this paper, we integrate a search mechanism based on a cooperative behavior of many individuals of chicken swarm in order to design an hybrid parallel CSO, several groups of the chicken swarm cooperate to find better solutions. The new parallel approach allows obtaining high quality results than the sequential model in reasonable execution time. Each thread executed one individual and each thread block was used for an independent execution of one group. The parallel algorithm runs on GPU, while the CPU is used to initialize the solutions by the constructive procedure of GRASP.

7. PARALLEL HYBRID CSO FOR QUADRATIC ASSIGNMENT PROBLEM

The pseudocode of the new metaheuristic, called parallel hybrid chicken swarm optimization (PHCSO) is summarized in Algorithm 1.

In this paper, we provide a new redefinition of operators and operations already introduced by CSO algorithm, the new approach is based on three operations, the subtraction operation $\ominus$, the multiplication operation $\otimes$ and the addition operation $\oplus$. On the one side, the contructive procedure of GRASP allows to have better initial solutions for the adapted algorithm. On the other side, the 2 way-perturbation procedure of Tabu search will be intergrated to solve the problem found in the rooster's movements, as in the previous studies, a rooster can move only if a better fitness value is checked.

The main equations used to analyze the movement of roosters, hens and chicks are respectively defined by the following equations 10 ,11 and 12.

$$x_i^{t+1} = x_i^t \oplus Randn(0, \sigma^2) \otimes x_i^t \tag{10}$$

$$x_i^{t+1} = x_i^t \oplus S1 \otimes Rand \otimes (x_{r1}^t \ominus x_i^t) \oplus S2 \otimes Rand \otimes (x_{r2}^t \ominus x_i^t) \quad (11)$$

$$x_i^{t+1} = W \otimes x_i^t \oplus FL \otimes (x_m^t \ominus x_i^t) \oplus C \otimes (x_r^t \ominus x_i^t) \quad (12)$$

In this paper, $\ominus$ operator depiced the crossover operator used in the position update equation of hens and chicks. Furthermore, $\otimes$ is redefined as the choice of one crossover operators. The uniform crossover (UX) [51] illustrates the movement towards the leader of groups and the sequential constructive crossover (SCX) [52] to the movement towards the neighbors, which could guarantee the competition in each group of the swarm.

The mutation process inspired of the genetic algorithm increases the diversity in a population by using random permutations in a selected solutions. In this work, we consider 2-exchange neighborhood as a mutation operator. The probabilty of applying the crossover is higher than the probabilty of mutation. Consequently, while designing the strategy of search, each individual of the swarm inherited the best qualities from each of the parents (rooster and hen).

Moreover, we integrate the constructive phase of GRASP using multiple threads, this operation will generate 10 solutions by using 10 threads per phase which could reduce the run time. The multi-groups approach lets several groups of chickens cooperate to find good solutions. In this paper, we consider that all the individuals show a similar behaviour, we can adapt the same approache [53]. Every chicken then sends its solution to the leader of the group.

Algorithm 1: Pseudo-code of PHCSO

```
1. Initialize the size of population PS
2. Generate PS using constructive procedure of GRASP
3. Initialize parameters: PS, G ,FL , C and W.
4. Evaluate the fitness values for each chicken
5. Rank and establish a hierarchal order
6. Create groups
7. Assign chicks to mother-hens
8. Update the equations 10 ,11 and 12
9. Improve the equation 10 by 2-exchange neighborhood mechanism
10. Update the new solution when it is better.
11. Return to step 5 if G is reatched until stop criterion
12. Post process results
```

The experimental results, discussion and comparison among PHCSO and some other metaheuristics will be detailed in the next section.

8. EXPERIMENTAL RESULTS AND DISCUSSION

8.1. Experimental protocol

The new algorithm PHCSO is programmed using python and tested on different QAPLIB instances, we test each instance 20 times in 100 iterations. The empirical research was implemented on a DELL in visual studio 2017 and simulated with Intel(R) Core(TM) i7-6500 U CPU 2.5GHZ (4 CPUs) 2.6 GHz and 16.00 GB of RAM and Microsoft Windows 10 Professional (64-bit) operating system.

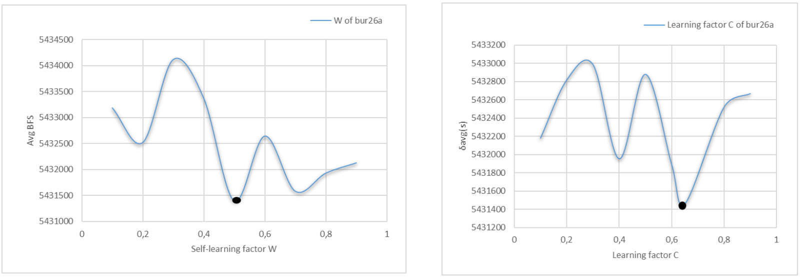
Based on preliminary experiments Figure 2a shows that PHCSO can achieve good results in term of quality of solutions for chr12a and bur26a when the population size is fixed on 500. Figure 2b presents the best solution found in term of quality of solution and average of run time of chr25a when the iteration G is equal to 10. The parameter G allows the redistibution of the swarm, which prevents to get caught in local optimum. Furthermore, Figure 3a, Figure 3b and Figure 3c indicate respectively the choice of parameters W,C and FL. Table 1 summarizes the parameters values of PHCSO algorithm.

Table 1. The Parameters for the PHCSO Algorithm

Parameters of PHCSO	Values
PS : Population size	500
RN : Number of roosters (%)	12
HN : Number of hens (%)	25
CN : Number of chicks (%)	63
G : Number of iterations to update the algorithm	10
W : Self-learning factor	0.5
FL : Learning factor from the mother hens	0.4
C : Learning factor from the rooster	0.65

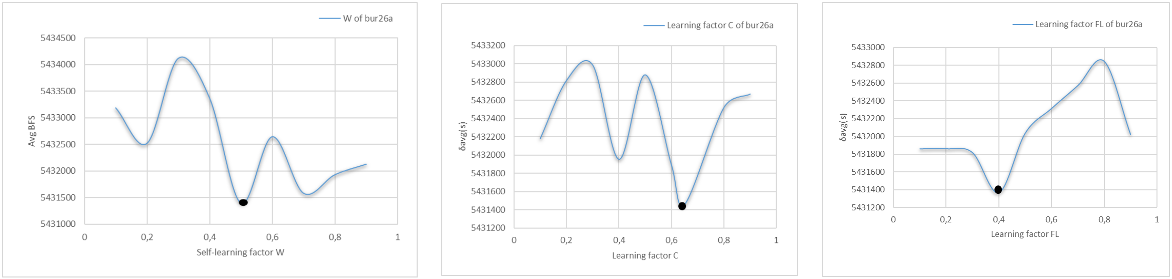
Table 2. Numerical Results by PHCSO Applied to Some Instances of QAPLIB

Instance	BKS	BFS	δ_{avg}	$t_{avg}(s)$	Er(%)	PSD	Suc(%)
bur26a.dat	5426670	5426670	5426670	1,02	0	0	100
bur26b.dat	3817852	3817852	3817852	0,78	0	0	100
bur26c.dat	5426795	5426795	5426795	0,65	0	0	100
bur26d.dat	3821225	3821239	3821239	0,8	0	0	100
bur26e.dat	5386879	5386879	5386879	0,79	0	0	100
bur26f.dat	3782044	3782044	3782044	0,33	0	0	100
bur26g.dat	10117172	10117172	10117172	0,15	0	0	100
bur26h.dat	7098658	7098658	7098658	0,29	0	0	100
chr12a.dat	9552	9552	9552	0,07	0	0	100
chr15a.dat	9896	10136	10136	0,96	0	0	100
chr15b.dat	7990	7990	7990	0,77	0	0	100
chr18a.dat	11098	11098	11098	0,03	0	0	100
chr20c.dat	14142	14142	14142	0,4	0	0	100
chr25a.dat	3796	3816	3825,8	4,65	1,224152456	0,154648201	76
els19.dat	17212548	17212548	17212548	0,06	0	0	100
esc16a.dat	68	68	68	0	0	0	100
esc16b.dat	292	292	292	0	0	0	100
esc16c.dat	160	160	160	0	0	0	100
esc16d.dat	16	16	16	0	0	0	100
esc32a.dat	130	130	130	0,01	0	0	100
esc32g.dat	6	6	6	0,01	0	0	100
esc32h.dat	438	438	438	0,03	0	0	100
esc64a.dat	116	116	118,4	0,09	0,845482754	0,963504299	86
had12.dat	1652	1652	1652	0,54	0	0	100
had14.dat	2724	2724	2724	0,46	0	0	100
had20.dat	6922	6922	6922	0,9	0	0	100
kra30a.dat	88900	88900	90670	0,21	0,456037779	0,155668723	70
kra30b.dat	91420	91420	91950	1,91	0,124863438	0,475663245	45
tai12a.dat	224416	224416	224416	0	0	0	100
tai12b.dat	39464925	39464925	39464925	0,02	0	0	100
tai15a.dat	388214	388214	388214	0,33	0	0	100
tai15b.dat	51765268	51765268	51765268	0,07	0	0	100
tai17a.dat	491812	491812	491812	0,21	0	0	100
tai20a.dat	703482	703482	703482	0,12	0	0	100
tai20b.dat	122455319	122455319	122452668,1	0,06	0,195103824	0,132581236	75
tai25a.dat	1167256	1167256	1167256	8,09	0	0	100
tai25b.dat	344355646	344355646	344355646	1,12	0	0	100
tai30a.dat	1818146	1824318	1843220	8,44	0,673556331	0,211723256	0
tai30b.dat	637117113	637117113	637817426	2,04	1,183859442	0,802901585	30
tai35a.dat	2422002	2428322	2450094	16,19	0,563779923	0,277992451	0
tai35b.dat	283315445	283315445	284108340	10,54	1,137765617	1,225566683	70
tai40a.dat	3139370	3139370	3222045	17,15	0,627579935	0,376791525	60
tai40b.dat	637250948	637250948	637409733	9,61	0,55632584	0,624665444	53
tai50a.dat	4938796	5090356	5094687	12,19	0,176659532	0,157465496	0
tai50b.dat	458821517	458845260	461827312	37,61	1,344666884	1,133887455	0
tai60a.dat	7205962	7351256	7414373	59,87	0,837172335	0,966633885	0
tai60b.dat	608215054	608351453	610225577	32,22	0,953344891	1,725866636	0
tai64c.dat	1855928	1855928	1855928	0,14	0	0	100
tai80a.dat	13499184	13657560	13791379	62,13	0,863398752	0,178666868	0
tai80b.dat	818415043	819081613	830812584	21,1	1,237856678	0,533677254	0
tai100a.dat	21052466	21503812	216046216	86,13	0,136558896	0,256695228	0
tai100b.dat	1185996137	1190142016	1187096921	96,15	0,127566983	0,245669879	0
sko42	15812	15812	15951	31,02	0,355624789	0,245666587	55
sko49	23386	24124	23564	78,96	0,765481242	0,987451666	0
sko81	90998	91113	91452	215.91	0,567741557	0,456884455	85



(a) (b)

Figure 2. (a) size PS of chr12a, (b) G for chr12a



(a) (b) (c)

Figure 3. (a) W for bur26a, (b) C for bur26a, (c) FL for bur26a

8.2. Computational results

The quality of the obtained solutions shown in Table 3 by PHCSO is evaluated using different measures which are the Percentage of the Standard Deviation PSD (13) and the percentage of error Er (15). The new proposed PHCSO allows to obtain the best-known solution about 72 % from all tested instances.

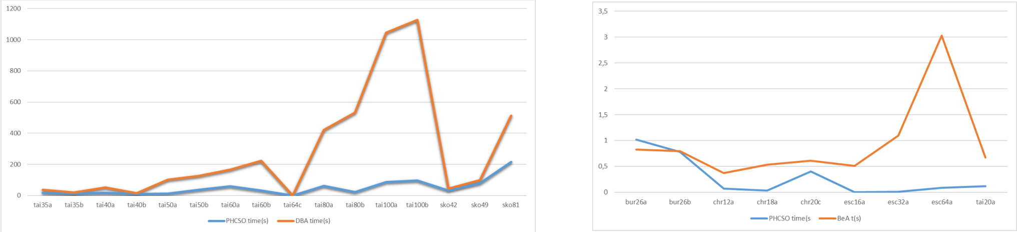
$$PSD = \frac{SD}{\delta_{avg}} \times 100 \tag{13}$$

$$SD = \sqrt{\frac{\sum_{i=1}^{20} (S_{best} - \delta_{avg})^2}{20}} \tag{14}$$

$$Er = \frac{(\delta_{avg} - BKS)}{BKS} \times 100 \tag{15}$$

where BKS is the best known solution, S_{best} is the best found value and δ_{avg} the average of the best found solution.

Figure 4a and Figure 4b show the results obtained while applying PHCSO algorithm and the comparison with other algorithms from the literature such as DBA [54] and BeA [55]. The experiments show that PHCSO algorithm is able to provide better results which prove that combining constructive procedure of GRASP and Tabu search neighborhood mechanism in the equation of rooster is able to enhance the performance of our algorithm.



(a)

(b)

Figure 4. (a) Comparative Average time between PHCSO and DBA, (b)Comparative Average time between PHCSO and BeA

8.3. Discussion of results

PHCSO algorithm shows good performance when compared to other state-of-the-art algorithms used to solve the Quadratic Assignment Problem. As result, we found that when we applied nonparametric Wilcoxon signed-rank tests, at the $= 0.05$, the null hypotheses are accepted in the case since the QAP is a minimization problem, which means that samples in all groups of the population are obtained from groups with the same mean value. We observe from Table 4 that PHCSO returns better solutions than BeA [55], HPSO [56] and DBA [54] and the p-values are smaller than the corresponding Holm step-down threshold, which signify that the use of constructive procedure of GRASP and Tabu Search greatly improve the performance of PHCSO.

Table 3. Wilcoxon signed-rank tests (at the = 0.05 level), with the Holm-Bonferroni step-down procedure used to control the family-wise error rate

Comparison	Instances	p-values	Holm
PHCSO vs HPSO	44	5E-23	0.023
PHCSO vs BeA	16	0.005	0.0119
PHCSO vs DBA	18	3.726E05	0.013

9. CONCLUDING REMARKS

In this paper, a new algorithm called Parallel Hybrid Chicken Swarm Optimization PHCSO was proposed to solve the quadratic assignment problem by suggesting a new modification on the original version of Chicken Swarm Algorithm. The constructive procedure of GRASP was integrated in order to obtain a better initial population using multiple threads, then 2-exchange neighborhood in Tabu search incorporated as a mutation operator in order to enhance the local search capability. The sequentianl constructive crossover (SCX) and the uniform crossover (UX) as crossover operators were used to improve the position update equation of individuals in the swarm. The computational results show that PHCSO algorithm is much better than other algorithms in terms of the quality of the solutions and the computing time. Our future research will aim to apply PHCSO in other combinatorial optimization problems as JSSP and VRP.

REFERENCES

[1] Slawomir Koziel and Xin-She Yang. *Computational optimization, methods and algorithms*, volume 356. Springer, 2011.

[2] Emile HL Aarts and Jan Karel Lenstra. *Local search in combinatorial optimization*. Princeton University Press, 1997.

[3] Xin-She Yang. *Nature-inspired metaheuristic algorithms*. Luniver press, 2010.

[4] James Kennedy. Swarm intelligence. In *Handbook of nature-inspired and innovative computing*, pages 187–219. Springer, 2006.

[5] Christian Blum and Xiaodong Li. Swarm intelligence in optimization. In *Swarm Intelligence*, pages 43–85. Springer, 2008.

[6] Xin-She Yang. Engineering optimizations via nature-inspired virtual bee algorithms. *Artificial Intelligence and Knowledge Engineering Applications: A Bioinspired Approach*, pages 317–323, 2005.

[7] Jianfeng Qiu, Jiwen Wang, Dan Yang, and Juan Xie. A comparison of improved artificial bee colony algorithms based on differential evolution. *Indonesian Journal of Electrical Engineering and Computer Science*, 11(10):5579–5587, 2013.

[8] Kang Seok Lee and Zong Woo Geem. A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice. *Computer methods in applied mechanics and engineering*, 194(36):3902–3933, 2005.

[9] Marco Dorigo, Mauro Birattari, and Thomas Stutzle. Ant colony optimization. *IEEE computational intelligence magazine*, 1(4):28–39, 2006.

[10] Ko-Wei Huang, Abba Suganda Girsang, and Tjeng Wawan Cenggoro. Fast ant colony optimization for clustering. *Indonesian Journal of Electrical Engineering and Computer Science*, 12(1), 2018.

[11] Xin-She Yang. A new metaheuristic bat-inspired algorithm. *Nature inspired cooperative strategies for optimization (NICSO 2010)*, pages 65–74, 2010.

[12] James Kennedy. Particle swarm optimization. In *Encyclopedia of machine learning*, pages 760–766. Springer, 2011.

[13] AH Gandomi, X-S Yang, S Talatahari, and AH Alavi. Firefly algorithm with chaos. *Communications in Nonlinear Science and Numerical Simulation*, 18(1):89–98, 2013.

[14] Gai-Ge Wang, Amir H Gandomi, Xin-She Yang, and Amir H Alavi. A new hybrid method based on krill herd and cuckoo search for global optimisation tasks. *International Journal of Bio-Inspired Computation*, 8(5):286–299, 2016.

[15] Christos H Papadimitriou and Kenneth Steiglitz. *Combinatorial optimization: algorithms and complexity*. Courier Corporation, 1998.

[16] Tjalling C Koopmans and Martin Beckmann. Assignment problems and the location of economic activities. *Econometrica: journal of the Econometric Society*, pages 53–76, 1957.

[17] Sewall Wright. *The roles of mutation, inbreeding, crossbreeding, and selection in evolution*, volume 1. na, 1932.

[18] V Bachelet, P Preux, and EG Talbi. The landscape of the quadratic assignment problem and local search methods. In *10th Meeting of the European Chapter on Combinatorial Optimization*, 1997.

[19] Peter Merz. Advanced fitness landscape analysis and the performance of memetic algorithms. *Evolutionary Computation*, 12(3):303–325, 2004.

[20] El-Ghazali Talbi. *Metaheuristics: from design to implementation*, volume 74. John Wiley & Sons, 2009.

[21] Eric D Taillard. Comparison of iterative searches for the quadratic assignment problem. *Location science*, 3(2):87–105, 1995.

[22] Ravi Kumar Bhati and Akhtar Rasool. Quadratic assignment problem and its relevance to the real world: a survey. *International Journal of Computer Applications*, 96(9), 2014.

[23] Eliane Maria Loiola, Nair Maria Maia de Abreu, Paulo Oswaldo Boaventura-Netto, Peter Hahn, and Tania Querido. A survey for the quadratic assignment problem. *European journal of operational research*, 176(2):657–690, 2007.

[24] Gerd Finke, Rainer E Burkard, and Franz Rendl. Quadratic assignment problems. *North-Holland Mathematics Studies*, 132:61–82, 1987.

[25] Alwalid N Elshafei. Hospital layout as a quadratic assignment problem. *Journal of the Operational Research Society*, 28(1):167–179, 1977.

[26] MA Pollatschek, N Gershoni, and YT Radday. Optimization of the typewriter keyboard by simulation. *Angewandte Informatik*, 17(0):438–439, 1976.

[27] Bernhard Wess and Thomas Zeitlhofer. On the phase coupling problem between data memory layout generation and address pointer assignment. *Lecture notes in computer science*, pages 152–166, 2004.

[28] Mahdi Bashiri and Hossein Karimi. Effective heuristics and meta-heuristics for the quadratic assignment problem with tuned parameters and analytical comparisons. *Journal of Industrial Engineering International*, 8(1):6, 2012.

[29] Vittorio Maniezzo and Alberto Colorni. The ant system applied to the quadratic assignment problem. *IEEE Transactions on knowledge and data engineering*, 11(5):769–778, 1999.

[30] Zvi Drezner. A new genetic algorithm for the quadratic assignment problem. *INFORMS Journal on Computing*, 15(3):320–330, 2003.

[31] Van-Dat Cung, Thierry Mautor, Philipe Michelon, and Andréa Tavares. A scatter search based approach for the quadratic assignment problem. In *Evolutionary Computation, 1997., IEEE International Confer-*

ence on, pages 165–169. IEEE, 1997.

[32] AS Ramkumar, SG Ponnambalam, N Jawahar, and RK Suresh. Iterated fast local search algorithm for solving quadratic assignment problems. *Robotics and Computer-Integrated Manufacturing*, 24(3):392–401, 2008.

[33] Kambiz Shojaee Ghandeshtani, N Seyedkashi, Nima Mollai, and Mohammad Mohsen Neshati. New simulated annealing algorithm for quadratic assignment problem. In *The Fourth International Conference on Advanced Engineering Computing and Applications in Sciences*, pages 87–92, 2010.

[34] Umut Tosun. A new recombination operator for the genetic algorithm solution of the quadratic assignment problem. *Procedia Computer Science*, 32:29–36, 2014.

[35] Leon Steinberg. The backboard wiring problem: A placement algorithm. *Siam Review*, 3(1):37–50, 1961.

[36] Dennis R Heffley. Decomposition of the koopmans-beckmann problem. *Regional Science and Urban Economics*, 10(4):571–580, 1980.

[37] Arthur M Geoffrion and Glenn W Graves. Scheduling parallel production lines with changeover costs: Practical application of a quadratic assignment/lp approach. *Operations Research*, 24(4):595–610, 1976.

[38] Saif Benjaafar, Sunderesh S Heragu, and Shahrukh A Irani. Next generation factory layouts: research challenges and recent progress. *Interfaces*, 32(6):58–76, 2002.

[39] Xianbing Meng, Yu Liu, Xiaozhi Gao, and Hengzhen Zhang. A new bio-inspired algorithm: chicken swarm optimization. In *International Conference in Swarm Intelligence*, pages 86–94. Springer, 2014.

[40] Dinghui Wu, Fei Kong, Wenzhong Gao, Yanxia Shen, and Zhicheng Ji. Improved chicken swarm optimization. In *Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), 2015 IEEE International Conference on*, pages 681–686. IEEE, 2015.

[41] Thomas A Feo and Mauricio GC Resende. Greedy randomized adaptive search procedures. *Journal of global optimization*, 6(2):109–133, 1995.

[42] L Pardalos and M Resende. A greedy randomized adaptive search procedure for the quadratic assignment problem. *Quadratic Assignment and Related Problems, DIMACS Series on Discrete Mathematics and Theoretical Computer Science*, 16:237–261, 1994.

[43] Panos M Pardalos, Leonidas S Pitsoulis, and Mauricio GC Resende. Fortran subroutines for approximate solution of sparse quadratic assignment problems using grasp. 1995.

[44] Paola Festa and Mauricio GC Resende. An annotated bibliography of grasp–part i: Algorithms. *International Transactions in Operational Research*, 16(1):1–24, 2009.

[45] Charles Fleurent and Fred Glover. Improved constructive multistart strategies for the quadratic assignment problem using adaptive memory. *INFORMS Journal on Computing*, 11(2):198–204, 1999.

[46] Fred Glover. Future paths for integer programming and links to artificial intelligence. *Computers & operations research*, 13(5):533–549, 1986.

[47] Fred Glover. Tabu search—part i. *ORSA Journal on computing*, 1(3):190–206, 1989.

[48] Éric Taillard. Robust taboo search for the quadratic assignment problem. *Parallel computing*, 17(4-5):443–455, 1991.

[49] El-Ghazali Talbi, Z Hafidi, and Jean-Marc Geib. A parallel adaptive tabu search approach. *Parallel computing*, 24(14):2003–2019, 1998.

[50] Alfonsas Misevicius. A tabu search algorithm for the quadratic assignment problem. *Computational Optimization and Applications*, 30(1):95–111, 2005.

[51] David M Tate and Alice E Smith. A genetic approach to the quadratic assignment problem. *Computers & Operations Research*, 22(1):73–83, 1995.

[52] ZH Ahmed. A simple genetic algorithm using sequential constructive crossover for the quadratic assignment problem. 2014.

[53] E-G Talbi, Olivier Roux, Cyril Fonlupt, and Denis Robillard. Parallel ant colonies for the quadratic assignment problem. *Future Generation Computer Systems*, 17(4):441–449, 2001.

[54] Mohammed Essaid Riffi, Yassine Saji, and Mohammed Barkatou. Incorporating a modified uniform crossover and 2-exchange neighborhood mechanism in a discrete bat algorithm to solve the quadratic assignment problem. *Egyptian Informatics Journal*, 18(3):221–232, 2017.

[55] Wojciech Chmiel and Piotr Szwed. Bees algorithm for the quadratic assignment problem on cuda platform. In *Man–Machine Interactions 4*, pages 615–625. Springer, 2016.

[56] Ayah M Helal and Ashraf M Abdelbar. Incorporating domain-specific heuristics in a particle swarm optimization approach to the quadratic assignment problem. *Memetic Computing*, 6(4):241–254, 2014.