© 2014, Scienceline Publication www.science-line.com
ISSN: 2322-5114
Journal of World's Electrical Engineering and Technology
J. World. Elect. Eng. Tech., 3(3): 135-139, 2014
JWEET

Discretization of Cuckoo Optimization Algorithm for Solving Quadratic Assignment Problems

Elham Kazemi and Sanaz Dejam*

Department of Computer Engineering, Islamic Azad University of Abadan, Abadan, Iran

*Corresponding author's Email: sanaz.dejam@gmail.com

ORIGINAL ARTICLE
Received 13 Aug. 2014
Accepted 09 Oct. 2014

Abstract *–Quadratic Assignment Problem (QAP) is one the combinatorial optimization problems about which research has been done in many companies for allocating some facilities to some locations. The issue of particular importance in this process is the costs of this allocation and the attempt in this problem is to minimize this group of costs. Since the QAP's are from NP-hard problem, they cannot be solved by exact solution methods. Cuckoo Optimization Algorithm is a Meta-heuristic method which has higher capability to find the global optimal points. It is an algorithm which is basically raised to search a continuous space. The Quadratic Assignment Problem is the issue which can be solved in the discrete space, thus the standard arithmetic operators of Cuckoo Optimization Algorithm need to be redefined on the discrete space in order to apply the Cuckoo Optimization Algorithm on the discrete searching space. This paper represents the way of discretizing the Cuckoo optimization algorithm for solving the quadratic assignment problem.*

Keywords*: Quadratic Assignment Problem (QAP), Meta-Heuristic Algorithms, Discrete Cuckoo Optimization Algorithm (DCOA).*

INTRODUCTION

Koopmans and Beckmann [1] introduced the Quadratic Assignment Problem (QAP) as a mathematical model related to economic activities for the first time. QAP is one of the most difficult problems of combinatorial optimization, which deals with allocation of a set of facilities, machines, or units to a set of locations or activities with minimal cost. Grading archaeological data, planning the space, designing keyboards for typists, arranging hospitals, etc. are some examples of the application of QAP. Assigning n facilities to n locations is proportional to the flow between the facilities multiplied with their distances is done with the purpose of allocating each facility to each location in such a way that the total cost is minimized. Therefore, there will be two n×n matrices: the flow matrix, A=(aij) and the distance matrix B=(bij).

$$\min_{\pi \in S_n} \sum_{i=1}^{n} \sum_{j=1}^{n} a_{\pi(i)\pi(j)} b_{ij} \qquad (1)$$

Where, Sn is a set of permutation of {1, 2, 3,…, n}. Any individual product of aπ(i)π(j)bij is the cost of assigning facility π(i) to location i and facility π(j) to location j. A QAP with input matrix A, B is shown as (A, B). Sometimes, if any of the coefficient matrices A, B is symmetric, QAP (A, B) is called symmetric. Otherwise, it is called asymmetric QAP [2].

Another problem, which is a little different and has been investigated by several authors, is also taken as a QAP as follows. In addition to the two coefficient matrices A and B, a third matrix C=(cij) is given where cijis the cost of placing facility i at location j, and the problem will be:

$$\min_{\pi \in S_n} \sum_{i=1}^{n} \sum_{j=1}^{n} a_{\pi(i)\pi(j)} b_{ij} + \sum_{i=1}^{n} c_{\pi(i)i} \qquad (2)$$

There are two major groups of methods for solving optimization problems: exact methods and meta-heuristic methods. Some exact algorithms for solving QAP include dynamic programming [3] and branch and bound family algorithms [4, 5]. Exact methods determine optimum solutions and fulfill the optimization condition. However, problems with sizes greater than 20 are not usually solvable by exact methods [6], thus calling for meta-heuristic methods which produce high quality solutions in a sensible time but do not guarantee finding the most optimized overall solution. Meta- heuristic algorithms include construction methods [7, 8], limited enumeration methods [9, 10], improvement methods [11], simulated annealing methods [12], Tabu search [13, 14], genetic algorithm [15], greedy randomized adaptive search procedure [16], ant colonies [17], [18], and imperialist competitive algorithm [19]. Durkota [20] presented a new Discrete Firefly Algorithm (DFA), which consists of constructing a suitable conversion of the continuous functions into new discrete functions, to solve the Quadratic Assignment Problems (QAP). Later, Hashni and Amudha [21] hybridized Consultant Guided Search algorithm (CGS) with Genetic Algorithm (GA) to solve the Quadratic Assignment Problems (QAP). Subsequently, Sanaz et al. [22] solved the Quadratic Assignment Problems (QAP) by the meta-heuristic Cuckoo algorithm, and then combined this algorithm with the Tabu algorithm and compared the results. They showed that the combination of Cuckoo and Tabu algorithms leads to more optimized solutions. They also compared the results with other meta-heuristic algorithms and showed that the combination of Cuckoo and Tabu algorithms is better than other single algorithms. Recently, Zhang et al. [23] solved the Quadratic Assignment Problems (QAP) by method of linearization, where one formulates the QAP as a mixed integer linear programming (MILP). Here, the discretization of cuckoo optimization algorithm for solving QAP is explained.

Priorities of COA algorithm respect to other optimization algorithms can be classified as:

1) Convergence is achieved more quickly,
2) Run time is too short,
3) Accuracy is very high,
4) Ability for local research beside overall research,
5) Probability for sticking in local optimized points is too low,
6) Research by variable population (as a result of destroying of population in inappropriate situations),
7) Overall movement of population toward better points due to destroying of inappropriate answers, and
8) Capability to solve optimization problems with large dimensions.

Discretization of Cuckoo through changing the operator

Cuckoo optimization, a meta-heuristic algorithm inspired from the nature, was developed by Yang and Deb in 2009 [24]. Cuckoo Optimization Algorithm (COA) was introduced by Rajabioun in 2011 [25]. COA is an algorithm which is basically raised for searching the continuous space and the combinatorial optimization problems are the issues which can be in solved in the discrete space, thus the standard arithmetic operators of COA need to be redefined on the discrete space in order to apply the COA algorithm on the discrete searching space. To do this, the concepts of distance and geometric rules are introduced for solutions of discrete spaces and redefined based on the theory of COA operators distance. In general, this section presents the need for changing the basic COA algorithm for discrete optimization problems in order to indicate the changed relation of its migration, but according to the permutation nature of combinatorial optimization problems, another change, changing the process of hatchery, is observed in the permutation problems as follows.

A. Definition of hatchery for permutation problems

B. Three different operators are defined in the section in order to generate the appropriate positions for new hatchery. One of three operators are randomly selected and applied for determining the chicks' new positions in each iteration of optimization algorithm.

Method 1:

Suppose that [x1, x2, x3, x4, x5, x6] are the examples of a Cuckoo. Two random points are selected for assignment as follows:

X1	X2	X3	X4	X5	X6

Then a new position for laying the eggs is generated through replacing these two points as follows.

X1	X5	X3	X4	X2	X6

Method 2:

In this method, several sequential positions are selected as follows:

X1	X2	X3	X4	X5	X6

And all selected bits are reversed from left to right.

X1	X5	X4	X3	X2	X6

Method 3:

Several bits between the positions I and J are selected as follows:

		I		J	
X1	X2	X3	X4	X5	X6

Then an assignment is done as follows; the contents of positions I +1 to J are shifted to the position I and the content of position I is put after them.

X1	X2	X4	X5	X3	X6

After determining the new positions for the eggs through one of three methods, the eggs are laid and the priority of positions is calculated.

C. Redefinition of migration operator

We should to redefine all arithmetic operators in the migration operator in order to generalize the COA to the discrete searching space; the most important operator is the positional difference between two habitats. To understand this idea, consider the Cuckoo position for the permutation problem QAP with 8 location as follows:

$$X_{CurrentHabitat} = \begin{array}{|c|c|c|c|c|c|c|c|}\hline 8 & 3 & 6 & 7 & 5 & 1 & 2 & 4 \\ \hline\end{array}$$

Then, assume that the current goal point is as follows:

$$X_{GoalPoint} = \begin{array}{|c|c|c|c|c|c|c|c|}\hline 6 & 1 & 3 & 7 & 5 & 2 & 8 & 4 \\ \hline\end{array}$$

The movement list, which is obtained from assignment operator, is as follows:

$$M = X_{GoalPoint} - X_{CurrentHabitat} = \{(1,3);(2,3);(2,6);(6,7)\}$$

As mentioned, F$\in [0,1]$ is equal to three first movements of list M through considering F=0.8,M'=F $\times |M|$ as follows; $M' = \{(1,3);(2,3);(2,6)\}$

The new position, which is generated by applying the final movements, is as follows:

$X_{CurrentHabitat} + F \times M =$

6	1	3	7	5	8	2	4

Therefore, the new position is generated for the migration operator. The pseudo code of this discretization for permutation problems, called DCOA, is shown as follows:

1. Initialize cuckoo habitats with some random points on the profit function in range
2. Dedicate some eggs to each cuckoo
3. Define ELR for each cuckoo
4. Let cuckoos lay eggs inside their corresponding ELR with three different methods which are defined
5. Kill those eggs that are recognized by host birds
6. Let eggs hatch and chicks grow
7. Evaluate the habitat of each newly grown cuckoo through objective function
8. Limit cuckoos' maximum number in environment and kill those who live in worst habitats
9. Cluster cuckoos and find best group and select goal habitat
10. Let new cuckoo population immigrate toward goal habitat
    a) Build the differential list of movement with swap operator
    M j→i = XGoalPoint – XCurrentHabitat
    b) Generate the nexthabitat vector
    $$X_{NextHabitat} = X_{CurrentHabitat} \oplus F \otimes M_{j \to i}$$

if stop condition is satisfied stop, if not go to 2

Experiments and Simulation Results

This study addresses the general form of QAP and its purpose is to evaluate the behavior of COA in solving QAP so that its applicability is confirmed and it can then be used in solving specific real cases in next studies. For this reason, those problems are chosen which are more famous and have been used for testing other algorithms. Therefore, the most credible reference of QAPs, QAPLIB, was used which is prepared by Peter Hann, Berkard, Chella, Randal, and Karisch who are mathematics professors that specialize in QAP. In QAPLIB, different QAPs of different sizes are defined and solved by scientists such as Berkard, Al-Shaafi, Steinburg, etc. using exact, heuristic, and meta-heuristic methods.

The results of experiments and simulations are presented in Tables 1, 2 and 3 and graphically in Figures 1 and 2. First, we determine the results for a small population. Then, the resulting error percent is calculated and compared to the genetic and honeybee improved algorithms. The following results are for at most 200 repeats andinitial cuckoo population of 5 with at most 5 eggs for each bird. The maximum number of the cuckoos possible for living in each step is determined as 30.

Table 1. Results of running the program with 10 repeats for the problems chosen from qaplib. Errors are in percent

Problem Name	Problem Size	Best Solution Found So Far	Solution Method	Errors of Previous Methods	HBMO Error	DCOA Error
Lipa30a	30	13178 •	Exact	-	3.74	2.01
Lipa60a	60	107218 •	Exact	-	2.25	0.95
Lipa90a	90	360630 •	Exact	-	1.67	0.39
Sko49	49	23386 •	RO-TS	5.91	16.11	3.56
Sko56	56	34458 •	RO-TS	5.37	18.49	4.39
Sko64	64	48498 •	RO-TS	5.7	16.91	4.47
Sko72	72	66256 •	RO-TS	5.38	14.34	4.85

Table 2. Results of running the program with 100 repeats for the problems chosen from qaplib. Errors are in percent.

Problem Name	Problem Size	Best Solution Found So Far	Solution Method	Errors of Previous Methods	HBMO Error	DCOA Error
Lipa30a	30	13178	Exact	-	3.78	2.007
Lipa60a	60	107218	Exact	-	2.3	0.96
Lipa90a	90	360630	Exact	-	1.65	0.38
Sko49	49	23386	RO-TS	5.91	18.82	3.55
Sko56	56	34458	RO-TS	5.37	15.88	4.388
Sko64	64	48498	RO-TS	5.7	14.36	4.459
Sko72	72	66256	RO-TS	5.38	13.78	4.78

Even though the GA and HB algorithms were assumed to be the best optimization algorithms in the past, they are still used in many technical publications.

These optimization algorithms are so famous which are used as a criteria to examine the accuracy of other new optimization algorithms. There is no reason to emphasize that one optimization algorithm can be better than the other optimization algorithms, because each optimization algorithm due to its evolution should be able to find the optimized solution. There is a point here which demonstrates that because some of algorithms are completed very slowly like their actual models in nature; therefore the number of iterations should be increased in these algorithms to arrive at optimized solution. For example, evolution of genetic, which is based on the actual human genes, takes many years to be completed. And maybe this is a reason to find out that GA method needs a lot of number of iterations to achieve to the optimized solution. In contrast to GA method, because HB algorithm is based on the group of birds which find their meals so faster, therefore, HB algorithm gives optimized solution very faster than GA method.

Table 3. Results of running the program for the problems chosen from qaplib with large population

Problem Name	Optimized Solution	Problem Size	Error of the Best GA Solution (%)	Error the Best HBMO Solution (%)	Error of the Best DCOA Solution (%)
Esc32a	130	32	21.69	54.86	18.46
Esc32b	168	32	20.75	50.56	17.04
Esc32c	642	32	0	9.7	0
Esc32d	200	32	0	29.57	3
Esc32e	2	32	0	0	0
Esc32h	438	32	1.79	22.06	8.21

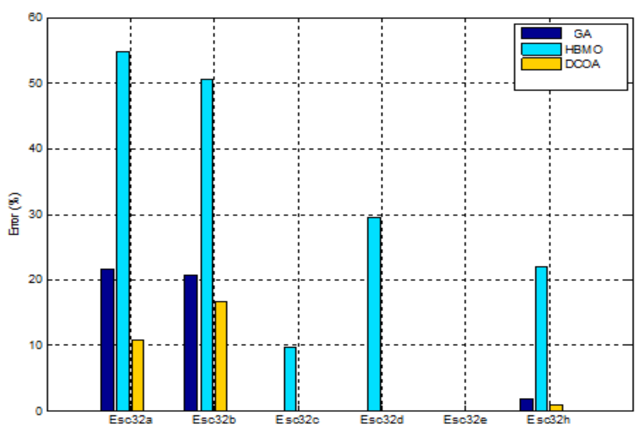

Figure 1. Comparison of Algorithms results for population number of 30.

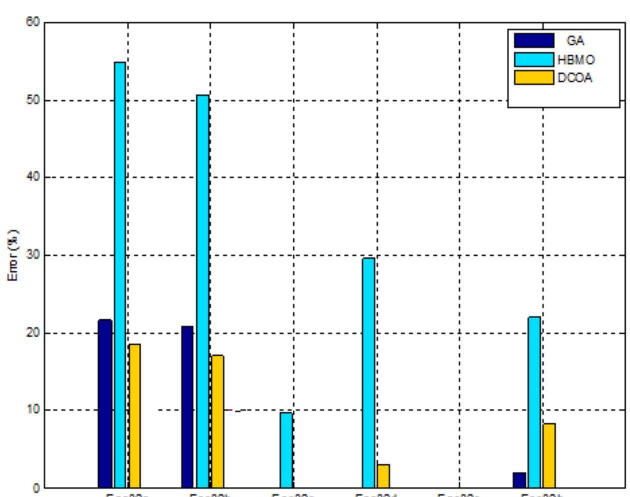

Figure 2. Comparison of Algorithms results for population number of 100.

Since DCOA method has special and improved algorithm and also the problems and the weak points of other optimization algorithms (like GA, HBMO and the other new methods like colonial completion algorithm) are not seen in this method, therefore DCOA method has high capability to converge to the optimized solution faster than the other algorithms and also has this ability to find overall optimized points with higher accuracy. By combination of various operators in DCOA method, which helps us to do local research during overall research, it is possible to find more accurate and reliable solutions. The ability of DCOA method was examined for problems with high dimensions and the results showed that this method works very well in this issue.

CONCLUSION

This paper investigates the idea of applying the Discrete Cuckoo Algorithm. The results obtained for low, medium and high population. Furthermore, this research studies the effect of enhancing the number of optimization iterations on the increased accuracy of algorithms solution. The obtained results indicated that the Cuckoo Optimization Algorithm had better performance than the genetic and honeybee algorithms. While the error of genetic and bee mating algorithms were slowly increasing in the problems with medium dimensions, the Cuckoo algorithm resulted in very reasonable solution for the problems with high dimensions.

REFERENCE

[1] Koopmans TC and Beckmann MJ (1957).Assignment problems and the location of

economic activities.*Econometrica*, Vol(25), pp 53-76.

[2] Tate DD and Smith AE (1995).A genetic approach to the quadratic assignment problem.*Computers and Operations Research*, Vol(1), pp 855–865.

[3] Ahuja RK, Orlin JB and Tiwari A (2000).A greedy genetic algorithm for the quadratic assignment problem.*Computers and Operations Research*, Vol(27), pp 917-934.

[4] Lawler EL (1963). The quadratic assignment problem.Manag.*Sci.*,Vol(9), pp 586-599.

[5] Pardalos PM and Crouse J (1989).A parallel algorithm for the quadratic assignment problem.Proceedings of the Supercomputing Conference, ACM Press., New York.

[6] Yong L, Pardalos PM and Resende MGC (1993). A Greedy Randomized Adaptive Search Procedure for the Quadratic Assignment Problem. DIMACS Series in Discrete Mathematics and Theoretical Computer Science, May 20-21.

[7] Bula ES, Armour GC and Vollmann TE (1962).Allocating facilities with CRAFT.*Harvard Business Rev.*Vol(42), pp 136-158.

[8] Armour GC and Buffa ES (1963). Heuristic algorithm and simulation approach to relative location of facilities. *Manag. Sci.*, Vol(9), pp 294-309.

[9] Burkard RE and Bonniger T (1983). A heuristic for quadratic boolean programs with applications to quadratic assignment problems. *European J. Oper. Res.*, Vol(13), pp 374-386.

[10] West DH (1983). Algorithm 608: approximate solution of the quadratic assignment problem. *ACM Trans. Math. Software*, Vol(9), pp 461-466.

[11] Li T, Pardalos PM and Resende MGC (1994).A greedy randomized adaptive search procedure for the quadratic assignment problem. In: Pardalos PM and Wolkowicz H (Eds.), Quadratic Assignment and Related Problems. DIMACS Series in Discrete Mathematics and Theoretical Computer Science. Providence. RI: *American Mathematical Society*, pp 237-261.

[12] Wilhelm MR and Ward TL (1987). Solving quadratic assignment problems by simulated annealing.*IEEE Trans.*, Vol(19), pp 107-119.

[13] Skorin-Kapov J (1990). Tabu search applied to the quadratic assignment problem. *ORSA J. Comput.*,Vol(2), pp 33-45.

[14] Taillard E (1991). Robust Tabu search for the quadratic assignment problem. Parallel Comput. Vol(17), pp 443-455.

[15] Fleurent C and Ferland JA (1994).Genetic hybrids for the quadratic assignment problem. DIMACS Series in Discrete Mathematics and Theoretical Computer Science, Vol(16), Providence. RI: *American Mathematical Society*, pp 173-187.

[16] Resende MGC, Pardalos PM and Li Y (1994).Fortran subroutines for approximate solution of dense quadratic assignment problems using GRASP. *ACM Trans. Math. Software*, Vol(22), pp 104-118.

[17] Gambardella L, Taillard E and Dorigo M (1997).Ant Colonies for the QAP, Tech. Report IDSIA-4-97, IDSIA, Lugano, Switzerland.

[18] Stutzle T and Dorigo M. (1999).ACO algorithms for the quadratic assignment problem. In: Corne, D., Dorigo, M., Glover, F. (Eds.), New Ideas for Optimization. McGraw-Hill.

[19] Safari Mamaghani A and Meybodi MR (2011). An Application of Imperialist Competitive Algorithm to Solve the Quadratic Assignment Problem, 6th international conference internet technology and secured translation, 11-14 December.

[20] Durkota K (2011). Implementation of a Discrete Firefly Algorithm for the QAP problem within the Seage framework. B.Sc. Dissertation, Czech Technical University in Prague, Prague, Czech.

[21] Hashni T and Amudha T (2012). Solving Quadratic Assignment Problems using a hybrid nature-inspired technique. International Journal of Current Research, Vol 4, no 2, pp. 216-221.

[22] Dejam, S., Sadeghzadeh, M., Mirabedini, S. J.: Combining Cuckoo and Tabu Algorithms for Solving Quadratic Assignment Problems. Journal of Academic and Applied Sciences, 2(12): 1-8, 2012

[23] Zhang H, Beltran-Royo C and Ma L (2013). Solving the quadratic assignment problem by means of general purpose mixed integer linear programming solvers. Annals of Operations Research, Vol 207, no 1, pp. 261-278.

[24] Yang XS and Deb S (2009) Cuckoo search via Lévy Flights. In: World Congress on Nature & Biologically Inspired Computing (NaBIC 2009). IEEE Publications, pp. 210–214.

[25] Rajabioun R. (2011). Cuckoo Optimization Algorithm. In: Applied Soft Computing journal, Vol(11), pp 5508-5518.

[26] Mahmoudi, S.: Discretization of Cuckoo Optimization Algorithm: A Case Study for Graph Coloration Problem. M.Sc. Dissertation, College of NabiAkram, Tabriz, Iran, 2012.