Application of the Invasive Weed Optimization Algorithm for Quadratic Assignment Problem

Furkan K. Yıldırım

Faculty of Engineering and Natural Sciences; Sabanci University
Orhanli, Tuzla, 34956 Istanbul, Turkey
furkanyildirim@sabanciuniv.edu

Keywords: *Quadratic assignment problem, invasive weed optimization algorithm, population-based metaheuristic.*

1 Introduction

The facility layout problem refers to the design of the optimal layout of working facilities in the workshop to minimize the operating costs and maximize the operational efficiency of the industrial production system for a given material flow and logistics relationship [1]. The quadratic assignment problem (QAP) is a subproblem of facility layout problem. The Quadratic Assignment Problem aims to assign a set of facilities to a set of locations by trying to minimize the total cost. It is called quadratic because the objective function contains a term of second degree in the unknown permutation [2]. Since the objective function contains a quadratic term in the unknown permutation, the number of possible assignments increases exponentially as the number of facilities-locations increases it is not possible to solve it in polynomial time. For this reason, QAP is an NP-Hard problem, and heuristic and metaheuristic approaches are used instead of exact approaches in its solution [3]. With these approaches, solution quality can be achieved in a short calculation time. In this paper, we use invasive weed optimization algorithm (IWO) which has proposed by Mehrabian and Lucas 2006 [4] is used to the solve QAP.

2 Solution Methodology

The IWO algorithm is inspired by a common agricultural phenomenon, the colonization of invasive weeds. Weed is a plant whose population grows widely or predominantly in areas that have been significantly damaged by human activities, without being intentionally cultivated [5]. IWO algorithm is a population-based heuristic algorithm characterized by properties such as seeding, growth, and competition in a weed colony [4]. Simulating the colonization of weeds with these characterized features comes out as follows: [6]

1) A restricted number of seeds are dispersed in the search area.

2) Depending on its fitness, every seed develops into a flowering plant that yields more seeds.

3) The generated seeds grow into new plants after being randomly scattered through the search area.

4) Until the maximum number of plants is achieved, this procedure is repeated; at this point, only the plants with lower fitness can persist and yield seeds; the remaining plants are eliminated. Until the maximum number of iterations is reached, the procedure keeps going. The plant with the best fitness is closest to the optimal solution.

The following is a detailed discussion of the method [4]:

2.1 Population Initializing

Over the *d* dimensional problem space, a population of starting solutions is distributed at random points. The initial search area is notated by X_{ini} which is bounded by a lower and upper bound. Typically, the upper bound is a positive real number, whereas the lower bound is typically a negative real number.

2.2 Reproduction

A plant can produce seeds based on its own fitness level as well as the lowest and highest fitness value of the colony. Every plant yields a certain minimum, S_{min}, and maximum, S_{max}, of seeds, which rises linearly with each subsequent production of seeds. This means that, in contrast to the process seen in nature, new plants have an opportunity to survive and reproduce.

2.3 Spatial dispersal

Using normal distributed random numbers with a mean equal to zero but varied variance, the produced seeds are dispersed randomly throughout the d dimensional search space. This implies that seeds will be dispersed at random to live close to their parent plant. Although at each step (generation), the random function's standard deviation will decrease from an initial value, $\sigma_{initial}$, to a final value, σ_{final}. The random function's standard deviation for each iteration given as follows:

$$\sigma_{iter} = \frac{(iter_{max} - iter)^n}{(iter_{max})^n}\left(\sigma_{initial} - \sigma_{final}\right) + \sigma_{final}$$

Where $iter_{max}$ is the maximum number of iterations, σ_{iter} is the standard deviation at the present time step and n is the nonlinear modulation index.

2.4 Competitive exclusion

To keep the number of plants in a colony from growing too large, there must be some kind of competition between plants. A colony's total number of plants will eventually reach its maximum through rapid reproduction after a few iterations; however, it is anticipated that the desirable plants would reproduce more often than the undesirable ones. A system for removing the plants that have poor fitness in the production process activates when the colony reaches its maximum number of plants, p_{max}. Once each seed has located itself inside the search area, it is ranked alongside its parents (such as a weed colony). After then, less fit weeds are removed in order to bring a colony's total number down to its limit. With the help of this process, plants with lower fitness can reproduce and perhaps survive if their progeny exhibit higher fitness within the colony.

3 Experimental Results

In this part, we use many benchmarks pilot numerical problems to assess the effectiveness of the suggested technique. After running the algorithm several times for each problem with various parameter values, we have determined which solution works best and have included the matching parameter values. The parameters' values are as in Table 1.

Table 1. Initial values of the IWO parameters before tuning

Parameter	Value	Parameter	Value
X_{ini}	(-5,5),(-10,10),(-20,20)	S_{min}	1
$p_{initial}$	15, 20, 30	S_{max}	3, 4
p_{max}	20, 25, 30	$\sigma_{initial}$	20
$iter_{max}$	20, 25	σ_{final}	0.5, 0.25
n	2	dim	Prob. size

The variables given in the table were applied to chr12a, chr12b, chr12c, tai20a, tai20b, bur26a, bur26b, bur26c tai50a, tai50b, wil100a QAP test cases. Calculations were made on a MacBook Pro with an Apple M2 Pro processor with 12 CPU cores (8 performance cores and 4 efficiency cores and 19 GPU cores) and 32GB RAM. The resulting results are given in Table 2, along with the variables that give the best results, compared with the best results detected in QAPLIB [7].

Table 2. IWO best value in comparison to the best known for QAP and the best set of parameters

Test Case	**chr12a**	**chr12b**	**chr12c**	**tai20a**	**tai20b**	**bur26a**	**bur26b**	**bur26c**	**tai50a**	**tai50b**	**wil100a**
dim	12	12	12	20	20	26	26	26	50	50	100
X_{max}	10	10	20	5	20	10	5	5	5	20	5
X_{min}	-10	-10	-20	-5	-20	-10	-5	-5	-5	-20	-5
p_{max}	25	30	15	30	30	20	30	20	30	30	30
$p_{initial}$	20	30	20	15	30	20	30	20	30	20	30
$iter_{max}$	20	25	20	25	20	20	25	25	25	20	25
S_{max}	3	5	5	5	5	5	5	5	5	5	5
S_{min}	1	1	1	1	1	1	1	1	1	1	1
$\sigma_{initial}$	20	20	20	20	20	20	20	20	20	20	20
σ_{final}	0.5	0.25	0.25	0.25	0.5	0.5	0.25	0.25	0.25	0.5	0.5
n	2	2	2	2	2	2	2	2	2	2	2
Exc. Time	0.051s	0.084s	0.050s	0.250s	0.218s	0.276s	0.452s	0.275s	1.576s	1.333s	442.361s
IWO Value	10360	10162	11188	763144	126885440	5484652	3850892	5466391	5565998	547260549	289120
QAPLIB Best Value	9552	9742	11156	703482	122455319	5426670	3817852	5426795	4938796	458821517	273038
GAP	8.45%	4.31%	0.28%	8.48%	3.61%	1.06%	0.86%	0.72%	12.69%	19.27%	5.89%

When the results were examined, it was revealed that there was a GAP with an average of 5.97%. On the other hand, it has been observed that different results occur for different parameters when running test cases. For this reason, better results can be obtained by performing more parameter tuning operations with Machine Learning techniques.

References

[1] Jiang, F., Li, L., Tang, Y., Zhang, H., & Liu, X. (2023). A Facility Layout Algorithm for Logistics Scenarios Driven by Transport Lines. *Applied Sciences*.

[2] Koopmans T.C. and Beckmann M.J. (1957). Assignment problems and the location of economics activities. *Journal of Econometrica*, (Vol. 25, pp. 53-76).

[3] Finke, G., Burkard, R. E., & Rendl, F. (1987). Quadratic Assignment Problems, *North-Holland Mathematics Studies* (Vol. 132, pp. 61-82).

[4] Mehrabian A. R. and Lucas C. (2006). A novel numerical optimization algorithm inspired from weed colonization. *Journal of Ecological Informatics*, (Vol. 1, pp. 355–366).

[5] Baker H. G. and Stebbins G. L. (1965). *The Genetics of Colonizing Species*. Academic Press, New York

[6] Mallahzadeh A. R., Es'haghi S. and Alipour A. (2009). Design of an E-shaped mimo antenna using IWO algorithm for wireless application. *Journal of Progress in Electromagnetics*

[7] Zhang, H., Liu, F., Zhou, Y., & Zhang, Z. (2020). A hybrid method integrating an elite genetic algorithm with tabu search for the quadratic assignment problem. *Information Sciences*, *539*, 347-374.