Bulletin de la Société Royale des Sciences de Liège, Vol. 86, *special edition,* 2017, *p.* 493 - 508

Solving Graph Bandwidth Minimization Problem Using Imperialist Competitive Algorithm

Amir ALIABADIAN[1], Mohammad-Rasol JAFARI[2] , Ali AZARBAD[3]

[1]faculty member at the University of shomal, a.aliabadian@shomal.ac.ir
[2]Power Engineering student at the University of shomal, muhammad.rasol.75@gmail.com
[3]Electronic Engineering student at the University of shomal, aliazarbad@yahoo.com

Abstract

The bandwidth minimization problem can be used in data storage and VLSI design issues and saving large hypertext media, etc. The Matrix Bandwidth Minimization Problem involves finding matrix rows and columns permutation so that non-zero elements of the matrix A are located in a band that is as close as possible to the original diameter to minimize the amount of $\{\max\{|i-j|:aij\neq.\}$. The Bandwidth Minimization Problem for Graphs (BMPG) is a complicated problem; hence the deterministic algorithms are not appropriate to solve these kinds of problems. The purpose of this research is to reduce the required computations through the use of heuristic algorithms and evolutionary algorithms, so that instead of using purely mathematical methods to find answers, we can turn the problem into an optimization problem through the use of collective intelligence and evolutionary algorithms and the concepts in this field. In the present paper, the use of meta-heuristic algorithm, Imperialist competitive algorithm is proposed in order to solve minimization problem. In this paper, the performance of presented algorithm with random samples has been evaluated compared with the results of genetic algorithms. The results of tests show that the Imperialist competitive algorithm can be considered as an efficient method to solve the bandwidth minimization problem for graphs.

Keywords: Graph bandwidth, Matrix bandwidth, Imperialist competitive algorithm, Genetic algorithm

1. Introduction

The bandwidth minimization problem for graphs has formed in 1962 on the basis of studies conducted in Jet Propulsion Laboratory in Pasadena. These studies were aimed to minimize the maximum absolute errors of six-bit picture codes that were represented as edge differences in hypercube [1]. This problem can be used in data storage, VLSI design and network durability, industrial electromagnetics, finite element methods for

finding approximate solutions of partial differential equations, large-scale power conversion systems, circuit design, chemical kinetics, numerical geophysics and saving large hypertext media [7][6][5][3].

To define the bandwidth minimization problem for graphs, it is assumed that A is a graph with vertices set V and edge set E, so that in it we have $|V|$ = n. An L labeling from graph G assigns the set of integers {1, 2, 3, …n} to the graph G vertices, in other words, L labeling is a one-to-one mapping as $L{:}V\rightarrow\{1,2,3,\ldots,n\}$. If L(v) is an assigned labeling to vertex V, then vertex V bandwidth, that is represented as B_L(v), is equal to the maximum distance between L(v) and assigned labeling to the neighboring vertices v.Thus:

$$B_L(v)=max \quad \{|L(v)-L(u)|:(v,u)\in E\} \tag{1}$$

Graph G bandwidth by considerin L labeling will be equal to the maximum amount of the bandwidth of graph vertices. So, we have:

$$B_L(G)=max \quad \{B_L(v):v\in V\} \tag{2}$$

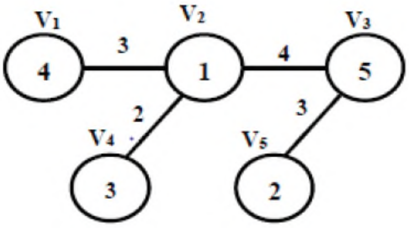

Figure 1: Example of labeling and calculation of graph bandwidth

Given the above definitions, optimal bandwidth for graph G is equal to the minimum amount of B_l (G) among possible L labelings for graph G. Therefore, the bandwidth minimization problem for graphs is defined as a problem that is just a matter of finding an L labeling that minimizes the value of B_l (G) [7][8].
The bandwidth minimization problem for graphs, also known as matrix bandwidth minimization, is defined assuming that $A=\{a_{ij}\}n\times$ is the adjacency matrix of the graph G. a_{ij}=1, when $(i,j)\in E$, otherwise a_{ij}=0. Bandwidth Minimization Problem involves finding matrix rows and columns permutation so that non-zero elements of the matrix A are located in a band that is as close as possible to the original diameter to minimize the amount of {max{|i−j|:aij≠.}[12][3].
Bandwidth minimization problem is a complicated problem, even if the graph is a tree log with a maximum grade 3, it is still a NP-Complete problem [9][8]. That is why different algorithms have been proposed by various people to solve the problem. The purpose of this research is to reduce the required computations through the use of heuristic algorithms and evolutionary algorithms, so that instead of using purely mathematical methods to find answers, we can turn the problem into an optimization problem through the use of collective intelligence and evolutionary algorithms and the concepts in this field. Heuristic algorithms are very suitable for

complicated problems and have the ability to search in the large state spaces. One good feature of these types of algorithms is that the solution does not engage with the details of the problem and this reduces the complexities of the proposed solution.

In this paper, the results of bandwidth minimization are implemented using meta-heuristic imperialist competitive algorithm and are compared and evaluated with genetic algorithm.

2. A Review of the Phenomenon of Imperialism

Sometimes some countries attempt to expand their sovereignty using applying some policies and exerting pressures to bring the countries with weak governance under their control and rule over those countries. This phenomenon is known as imperialism.

The competition among colonies and the increased awareness of people of their basic rights made imperialists to resort some reforms in these countries based on policy of assimilation in order to perpetuate their domination on their colonies. And thus the situation of colonies improved slightly in terms of power. However, this improvement was not without cost and the colonized country suffers irreversible social and cultural changes and turns in this respect to the colonial country.

2-1- imperialist Competitive Algorithm

Like other evolutionary algorithms, this algorithm begins with a number of random initial populations, each of which is called a "country". Some of the best elements of population are selected as imperialists. The remaining population is considered as a colony. Given their power, imperialists absorb these colonies through a process that is described in the following. Figure 1 shows the flowchart of the proposed algorithm.

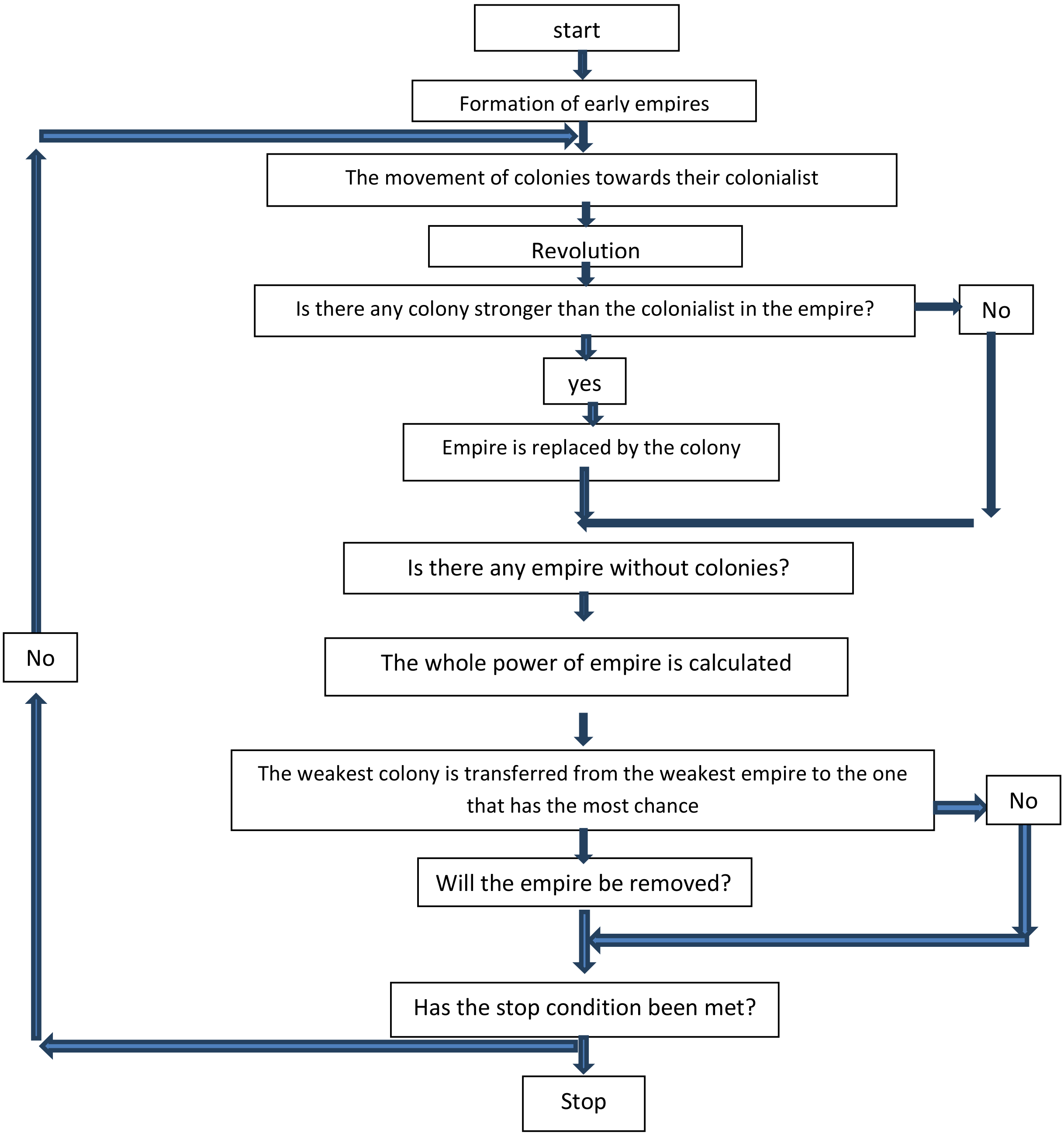
start
Formation of early empires
The movement of colonies towards their colonialist
Revolution
Is there any colony stronger than the colonialist in the empire?
No
yes
Empire is replaced by the colony
Is there any empire without colonies?
The whole power of empire is calculated
No
The weakest colony is transferred from the weakest empire to the one that has the most chance
No
Will the empire be removed?
Has the stop condition been met?
Stop

Figure 2: algorithm implementation process

The total power of any empire depends on both part of forming this empire; that is the imperialist country (the core) and its colonies. This dependence is modeled mathematically by defining the total cost of N-empire in equation (3) as the sum of colonial country's cost and a percentage of the average cost of its colonies:

$$T.C._n = Cost\left(imperialist_n\right) + \xi\, mean(Cost(colonies\ of\ \ impire_n)) \tag{3}$$

ξ is a positive number, usually between zero and one, and is considered close to zero. When ξ is considered small, the total cost of an empire becomes nearly equal to the cost of its central government (the colonial country) and an increase in the amount of ξ adds to the effect of the costs of an empire’s colonies in determining its total costs. Generally, in most implementations, ξ=0/05 will lead to good results.

With the formation of early empires, colonial rivalry between them begins. Every empire that fails to succeed in the competition and increase its power or at least prevents from reducing its influence will be eliminated from the colonial competition scene.

Over time, the colonies will become closer to the empire in terms of power and we will be witnessing a kind of convergence. The ultimate imperialist competition is when there is one single empire and the colonies are very close to the imperialist country in terms of location.

2-1-1 The Formation of Early Empires

Here, to find an optimal solution, we create an array of the problem variables that is called the country. In an N_{var}-dimensional optimization problem, a country is a $1\times N_{var}$ -dimensional array. The array is defined as follows.

$$country \quad = [\, x_1 , x_2 , x_3 ,..., \ x_{N_{\text{var}}} \,] \tag{4}$$

The amounts of variables in a country are displayed as decimal. From cultural-historical perspective, a country's socio-political characteristics such as culture, language, economic structure and other characteristics can be considered as its components. Figure 2 illustrates this problem. In fact, by solving an optimization problem using an ICA algorithm, we are looking for a country with the best political-social characteristics. Finding the country is, in fact, equivalent to finding the best parameters for the problem that produce the least amount for the cost function. For example, suppose that we want to minimize a certain function (for example f). In this problem, the country is defined as the equation (4).

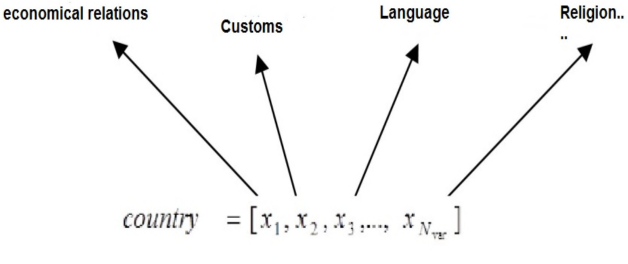

Figure 2: the socio-political components of a country (Religion/language/customs/economic relations)

A number of early countries should be created at the beginning of the algorithm. Therefore, the matrix of all early countries is randomly formed according to Equation (5).

$$COUNTRY = \begin{pmatrix} country_1 \\ country_2 \\ . \\ . \\ . \\ country_N \end{pmatrix} = \begin{pmatrix} x_{11}\,x_{12}\ldots x_{1Nvar} \\ x_{21}\,x_{22}\ldots x_{2Nvar} \\ . \\ . \\ . \\ x_{N1}\,x_{N2}\ldots x_{NNvar} \end{pmatrix} \tag{5}$$

By evaluation of the function f in the variables $x_1, x_2, x_3, ..., x_{N_{\text{var}}}$, one can obtain the cost of a country using the equation (6).

$$cost_i = f\,(country_i) = f\,(x_1, x_2, \ldots, x_{N_{var}}) \tag{6}$$

To start the algorithm, N_{country} early countries are created. N_{imp} of the best members of this population (countries with the least amount of the cost function) are selected as the imperialist. The remaining countries constitute N_{col} colonies, each of which will belong to an empire. For the initial division of the colonies among the empires, a number of colonies are devoted to each imperialist proportional to the power that imperialist. To do so, by knowing the cost of all imperialists we consider their normalized cost according to the equation (7).

$$C_n = \max\{cost\} - cost_n \tag{7}$$

where, Cost_n is the cost of n-imperialist, max{cost} is the maximum cost among the imperialists and C_n is the normalized cost of this imperialist. Each imperialist that has more costs or namely is weaker; its normalized cost will be less. By determining the normalized cost, relative normalized power of each imperialist is calculated using the equation (8) and the colonial countries are divided among the imperialists accordingly.

$$p_{\text{n}} = \left| \frac{C_i}{\sum_{i=1}^{N_{imp}} C_i} \right| \tag{8}$$

From another perspective, an imperialist normalized power is an imperialistic correlation managed by that imperialist. The initial number of the colonies of an imperialist according to the equation (9) is equal to

$$N.C_n = round\{p_n \times N_{col}\} \tag{9}$$

Where, $N.C_n$ is the initial number of the colonies of an empire and N_{col} is the total number of colonial countries in the early population of countries. round is the function that gives the nearest integer to a decimal number. Considering $N.C_n$ for each empire, $N.C_n$ countries are randomly selected among the early colonies and are devoted to the N-imperialist. By knowing the initial state of all empires, the imperialist competitive algorithm I(CA) begins. The process of evolution is a loop that continues to meet a stop condition.

Figure 3 shows the formation of the early empires. Accordingly, a larger empire has a greater number of colonies. In the figure, the asterisk represents the imperialist and circles of the same color represent their colonies. Dark-colored imperialist has created a more powerful empire had the highest number of colonies.

According to equation (8), if the imperialist cost function is maximum, the relative power of this imperialist will be equal to zero and it will be removed at the beginning of the algorithm. Though, its cost discrepancies with other colonizers may be negligible. Given this, in the articles written after the introduction of this algorithm, the relation has been modified.

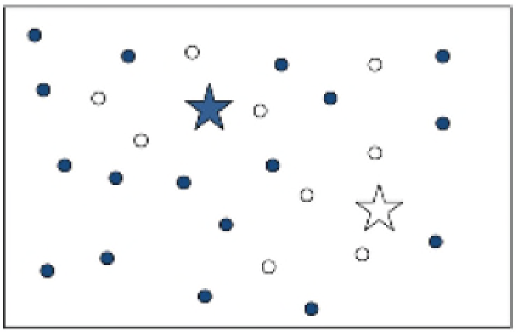

Figure 3: The formation of early empires

2-1-2 Modeling the policy of assimilation

In this section, the model of colonization process in the optimization algorithm is presented as the movement of colonies towards the imperialist country. In fact, given the way of representing each country in solving an optimization problem, the central government applies a policy of assimilation or absorption in an attempt to make the colonized country close to itself along the various socio-political demotions.

Figure 4 shows an overview of the movement. According to this figure, the colonizer country absorbs the colonized country along the lines of their culture and language. As shown in the figure, the colonized country moves towards the line connecting the colony to the colonizer as many as X unit according to the equation (10) and is dragged to the new location. In the figure, the distance between the colonizer and colonized country is represented by d. In the equation (10), x is a random number with uniform or any other suitable distribution

$$x \sim U(0, \beta \times d) \qquad (10)$$

Where, β is a number larger than 1 and close to 2 and even with a suitable selection, it can be β=2 . The coefficient makes the colonized country approaches from different directions to the colonizer country during its movement towards it.

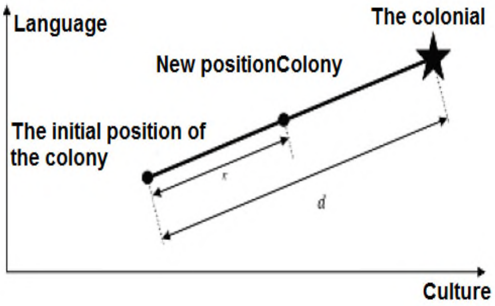

Figure 4: An overview colonies movement towards the colonialist

By studying the historical phenomenon of assimilation, a fact evident in this area is that despite the fact that imperialist countries had seriously followed the policy of assimilation, the events didn’t proceed fully according to their applied policy and there were some deviations in the results. In the presented algorithm, the possible deviation is took place by adding a random angle to the assimilation path of colonies. For this purpose, during the movement of colonies towards the colonialist, the amount of random angle is also added to the direction of the colonized. Figure 5 shows this case. To do so, instead of transferring the colonized the size of X unit towards the imperialist and the line connecting the colonized to the colonizer, we will move it the same amount but with a deviation of θ from the connecting vector. Usually, we select θ randomly and with a uniform distribution, however, any other distribution is appropriate.

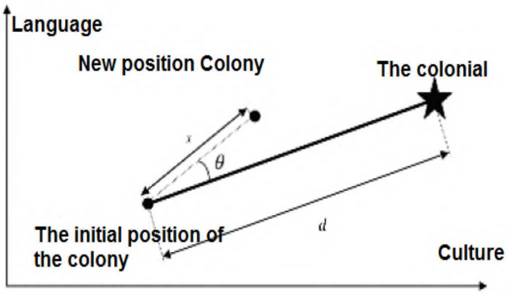

Figure 5: The actual movement of colonies toward colonialist

In the equation (11), γ is an optional parameter that its increase adds to the search around the imperialist and its decrease causes that the colonies move as close as possible to the connecting vector towards the imperialist. By considering radian unit for γ, a number close to $\frac{\pi}{4}$ is a convenient option in most implementations.

$$\theta \sim U(-\gamma,\gamma) \qquad (11)$$

2-1-3 Revolution

A sudden change in socio-political profile of the society is referred to as revolution that takes place in a relatively short time. In ICA algorithm, this means large changes in variables (political, social,..characteristics).
In fact, a country instead of changing according to an imperialist view, alters its position in terms of language, culture, etc., accidentally and in a short time. Revolution increases the search space and prevents algorithm being trapped in local minima. Revolution rate the colonies change percentage in the whole colonies. For example, the rate of 0.3 or 3% means that all colonies randomly change their positions. The high rate of revolution reduces the strength of algorithm and its speed of convergence.

2-1-4 Position displacement of colony and colonialist

Policy of assimilation, while destroying the socio-political structures of colonies, in some cases leads to positive results for them. As a result of applying this policy, some colonies develop a kind of public self-esteem. After a while, the educated class (in other words, those who absorbed in the colonial culture) begins to lead the people to escape from the clutches of imperialism. Many examples of this can be found in the English and France colonies. On the other hand, a look at the ups and downs of the rotation of power in the countries shows that the countries once in the heyday of the political-military power had fallen after some time. On the contrary, the countries grabbed the reins of power that until then had no power. Based on the shape (6), in the ICA algorithm to model this historic occasion, during the movement towards the imperialist some colonies may reach a better position than imperialist.

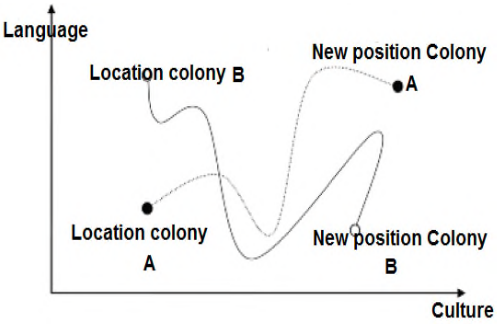

Figure 5: revolution

In this case, the colonizer and colonized countries switch their positions and the algorithm continues with the new imperialist country. Figure (7) shows the switching. In Figure (7 (a)), the best colony of the empire, which has less cost than the imperialist itself, is shown in a darker color. Figure (7 b) shows the entire empire after the change of positions [17].

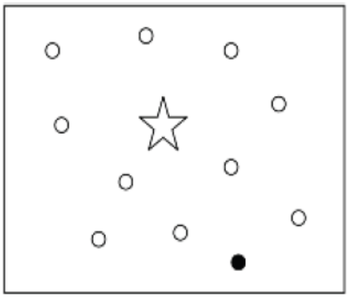

Figure 7: a) determination of the colony more powerful than imperialist

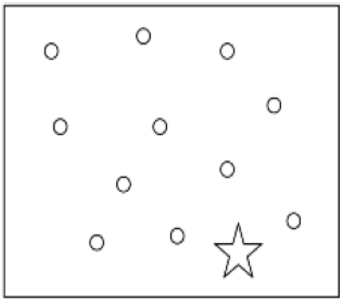

Figure 7: b) changing more powerful colony into imperialist

2-1-5 Imperialist competition

As said previously, each empire, which fails to increase its power and loses its competitiveness, will be gradually removed during the imperialist competition. This means that over time, weak empires lose their colonies and the powerful empire one conquer the colonies and enhance their power. To model this fact, we assume that the empire at the threshold of being removed is the weakest one. Thus, in each run of the algorithm, we take one or more of the weakest colonies of the weakest empire and create competition between all empires to seize the colonies. The mentioned colonies will not necessarily be seized by the most powerful empire, but powerful empires have more chance for take- over. Figure 8 shows an overview of the algorithm. In the figure, the empire (1) is considered as the weakest empire. One of its colonies is subjected to imperialist competition and empires 2,3,…N compete with each other to seize it. To model the competition for appropriation of the colony among the empires, we determine the probability of each empire to seize the colony (which is proportional to the power of that empire), by considering its total normalized cost according to the equation (12).

$$N.T.C._n = \max_i \{T.C._i\} - T.C._n \tag{12}$$

Where T.C_n is the total cost of the N-empire and N.T.C_n is the total normalized cost of that empire. An empire with less T.C_n has more N.T.C_n In other words, T.C_n is the total cost of an empire and N.T.C_n represents its total power. An empire with the least cost is the most powerful one. By having the value of total normalized cost, one can determine the strength or probabilty of appropriation of the intended colony by each empire according to equation (13).

$$p_{p_n} = \left| \frac{N.T.C._n}{\sum_{i=1}^{N_{imp}} N.T.C._i} \right| \tag{13}$$

By having the probability of appropriation of the colony by each empire, a mechanism similar to the roulette wheel in the genetic algorithm is needed to assign the intended colony to the empire with a probability proportional to the power of that empire.

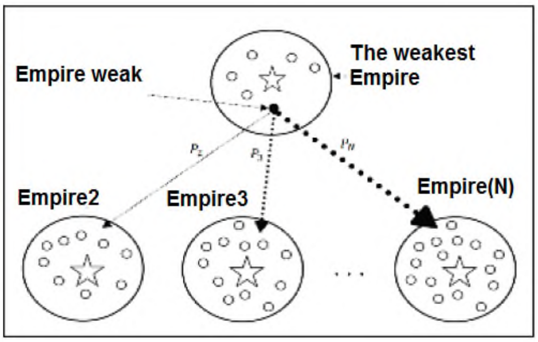

Figure 8: Larger empires are more likely to seize other empires' colonies

Besides the possibility of using a roulette wheel, here a new mechanism to implement this process is introduced that has much less computational cost compared to the roulette wheel. This reduction is associated with the omission of calculation of cumulative probability distribution which is required only to have a probability density function. In the following, the proposed mechanism to allocate the intended colony to the competitor empires is explained. To divide the colonies among empires given the power or the probability of appropriation for each empire, we can form p vectors based on the values above probabilities as follows.

$$P = [p_{p_1}, p_{p_2}, \ldots, p_{p_{N_{imp}}}] \tag{14}$$

P is a vector with $1 \times N_{imp}$ dimensions which is composed of the probability values of appropriation for each empire. We develop the random vector with the same dimensions as the vector P. The arrays of this vector are random numbers with uniform distribution in the interval [0,1].

$$R = [r_1, r_2, \ldots, r_{N_{imp}}] \tag{15}$$

$$r_1, r_2, \ldots, r_{N_{imp}} \sim U(0,1) \tag{16}$$

Then, we develop vector D as follows.

$$D = P - R = [D_1, D_2, \ldots, D_{N_{imp}}] \tag{17}$$

$$D = [p_{p1} - r_1\ ,\ p_{p2} - r_2\ ,\ldots,\ p_{N_{imp}} - r_{N_{imp}}] \tag{18}$$

Having vector D, we assign mentioned colony to the empire whose index in the vector D has the highest value. Every empire that has the greatest power of appropriation is more likely to have the highest-value index in the vector D. Since there is no need to calculate CDF, the performance speed of this mechanism is much more than the roulette wheel. With the appropriation by one of the empire's colony, also ends the operation of step algorithm. The appropriation of the colony by one of empires terminates this stage of algorithm.

2-1-6 Fall of weak empires

As stated, during the imperialist competitions, weak empires gradually fall and more powerful empires seize their colonies. Different conditions can be considered for the fall of an empire. In the proposed algorithm, an empire is treated as a removed empire when it has lost its colonies and has no strength to compete.

2-2-7 Convergence

The intended algorithm continues until the convergence condition is met or the total number of iterations is completed. After a while, all empires fall, and only one empire remains; the rest of countries will be placed under the control of a single empire. In this new ideal world, all colonies are run by a single empire, and the status as well as costs of colonies will be equal to those of the imperialist country. In this new world, not only the difference between the colonies is eliminated, but there is no difference between the colonies and the imperialist country. In other words, all countries are colony and colonizer at the same time. In this situation, imperialist competition is ended and the algorithm stops.

2-2 Imperialist competitive algorithm Pseudo-code

Imperialist competitive algorithm pseudo-code is given in Figure 9 below.

```
1- Choose some ransom points on the function and form the early empires.
2- Move the colonies toward the imperialist country (the policy of assimilation).
3- If there is a colony in an empire, that its cost is less than the imperialist cost, change the
   position of colony and imperialist.
4- Calculate the total cost of an empire (take into account the cost of the imperialist and its
   colonies).
5- Choose a colony from the weakest empire and give it to the empire with the highest
   probability of appropriation.
6- Delete the weak empire.
7- If only one empire is remained, stop otherwise go to stage 2.
```

Figure 9: Imperialist competitive algorithm pseudo-code

3. Simulation and conclusion

In imperialist competitive algorithm, in the presented labeling (permutations), the critical vertex is identified. In fact, this vertex will be the one with the maximum bandwidth in the intended labeling. For example, if the graph bandwidth is obtained by subtracting the label of two vertices 4 and 8, one of the vertices 4 or 8 is selected as the critical point. Then, the label of the remaining vertices of the graph are respectively changed with the label of critical point and the fitness value of the permutation resulting from this switching is calculated. The permutation that leads to the greatest reduction in graph bandwidth is selected as the solution of imperialist competitive algorithm as the answer.

In this section, the results of empirical experiments and comparisons conducted on random graphs are shown. Algorithms are implemented by MATLAB are run on a computer with specifications Intel Core i5 4200M CPU and. 64GB RAM The parameters involved in the implementation of the imperialist competitive algorithm are as follows.

Table 1: Initializing the imperialist competitive algorithm parameters

Parameters	values
No. Population	180
No. Imperials	18
No. Decades (Iterations)	100
Revolution rate	0.3
Varmin	-10
Varmax	+10
β	2
γ	Π/4
Zeta	0.03
Training	LOOCV
Testing	LOOCV

After various implementations of imperialist algorithm and determining appropriate values for these parameters, in order to find optimal labels, the results are presented in the table below.

Table 2: The results of imperialist competitive and genetic algorithms on randomly created samples

Nodes	Relations	Population	Iteration	Genetic algorithm	imperialist competitive algorithm
10	23	30	150	5	**5**
		30	250	5	**5**
20	95	30	150	14	**14**
		30	250	14	**13**
30	208	30	150	24	**23**
		30	250	24	**22**
40	418	30	150	35	**33**
		30	250	34	**32**
50	601	30	150	44	**42**
		20	200	43	**41**

As can be seen, the results indicate that the imperialist competitive algorithm generates optimal solutions compared to the genetic algorithm.

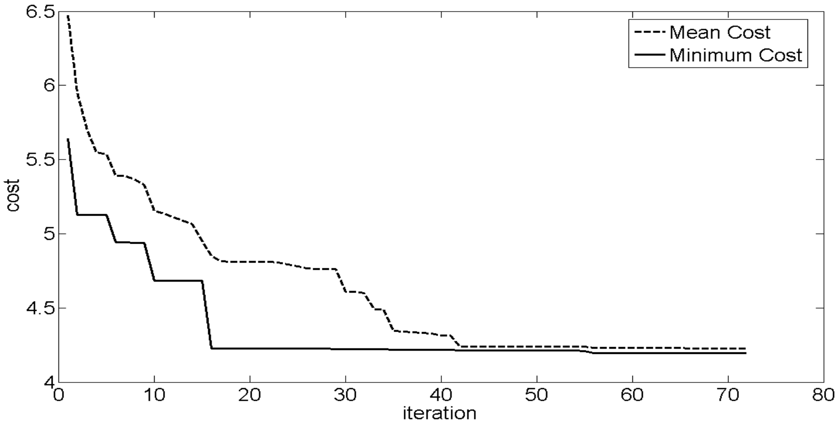

Figure 10: Minimum cost and average cost function in imperialist competitive algorithm based on the repetition of algorithm

Figure 10 shows the minimum cost of objective function in the imperialist competitive algorithm with the average cost of objective function in terms of repetition in algorithm.

References

[1] Esmaeil Atash paz-gargare, “Swarm optimization algorithm development and a review of its performance”, MA Thesis, Faculty of Electrical Engineering and computer of Tehran University, 2000.

[2] Liwei Fan, Sijia Pan, Zimin Li, Huiping Li, "An ICA-based support vector regression scheme for forecasting crude oil prices" Technological Forecasting and Social Change, Volume 112, Pages 245-253, November 2016.

[3] Niko Lietzén, Klaus Nordhausen, Pauliina Ilmonen, "Minimum distance index for complex valued ICA" Statistics & Probability Letters, Volume 118, Pages 100-106 , November 2016.

[4] Yao Junliang, Ren Haipeng, Liu Qing, "Fixed-point ICA algorithm for blind separation of complex mixtures containing both circular and noncircular sources" The Journal of China Universities of Posts and Telecommunications, Volume 23, Issue 2, Pages 15-2, April 2016.

[5] E. Pinana, I. Plana, V. Campos and R. Marti, "GRASP and path relinking for the matrix bandwidth minimization", European Journal of Operational Reseach, Vol. 153, pp. 200-210, .2004

[6] A. Esposito, M. S. Catalano, F. Malucelli and L. Tarricone, "Sparse matrix band-width reduction:Algorithms, Applications and real industrial cases in electromagnetics, high performance algorithms for structured matrix problems", Advances in the theory of Computation and Computational Mathematics, Vol. 2, pp. 27-45, 1998.

[7] P. Chinn, J. Chavtalova, A.K.Dewdney and N.E.Gibbs, "The bandwidth problem for graphs and matrices-a survey", Journal of Graph Theory, Vol. 6, pp. 223-254, 1982.

[8] M. Berry, B. Hendrickson and P. Raghavan,"Sparse matrix reordering schemes for browsing hypertext", Lectures in Applied Mathematics,Vol. 32, pp. 99-123, 1996.

[9] C. H. Papadimitriou, "The NP-completenss of the bandwidth minimization problem", Computing, Vol. 16,pp. 263-270, 1976.

[10] M. Garey, R. Graham, D. Johnson and D. E. Knuth, "Complexity results for band-width minimization", SIAM Journal of Applied Mathematics, Vol. 34, pp. 477-495, 1978.

[11] A. Lim, B. Rodrigues and F. Xiao, "Heuristics for matrix bandwidth reduction", European Journal of Operational Research, Vol. 174, pp. 69-91, 2006.

[12] R. Marti, M. Laguna, F. Glover and V. Campos, "Reducing the bandwidth of a sparse matrix with tabu search", European Journal of Operational Research, Vol. 135, No. 2, pp. 2001 ,211-220

[13] Safari Mamaghani, A. and Meybodi, M. R., "A Learning Automaton Based Approach to Solve the Graph Bandwith Minimization Problem", Proceedings of 5th International Conference on Applications of Information and Communication Technologies , pp. 1-5,2011.

[14] Atashpaz-Gargari,E.,Lucas,C.,.Imperialist competitive algorithm : an algorithm for optimization inspired by imperialistic competition , Proceeings of the IEEE Congress on Evolutionary Computation , Singapore , pp.4661-4667, 2007.

[15] Niknam,T.,Taherian-Fard,E.,Pourjafarian,N.,Rousta,A.,An efficient hybrid algorithm based on modified imperialist competitive algorithm and K-means for data clustering, Engineering Applications of Artificial Intelligence,24,pp.306–317,2011.

[16] Nazari-Shirkouhi,S.,Eivazy,H.,Ghodsi,R.,Rezaie,K.,Atashpaz-Gargari,E., Solving the integrated product mix-out sourcing problem using the Imperialist Competitive Algorithm, Expert Systems with Applications 37,pp.7615–7626,2010.

[17] Gorodilov,A.,Morozenko,V.,Genetic Algorithm For Finding The Key’s Length And Cryptanalysis Of The Permutation Cipher, International Journal "Information Theories & Applications" Vol.15, 2008.