AN EFFECT ANALYSIS OF THE PARALLEL MIGRATING BIRDS OPTIMIZATION ALGORITHM PARAMETERS

*Gültekin KUVAT, Department of Computer Engineering, Faculty of Engineering, Balıkesir University, Turkey, gkuvat@balikesir.edu.tr

(https://orcid.org/0000-0001-8179-1497)

Abdullah TÜLEK, 100. Yil Vocational and Technical Anatolian High School, Turkey, abdullahtulek@balikesir.edu.tr

(https://orcid.org/0000-0002-7574-4480)

Abstract

Migration is the process of sending selected solutions from a sub-population to the neighboring sub-population at specified intervals in parallel metaheuristic algorithms (PMAs). Topology, migration rate (MR), migration interval (MI), migration policy and communication model are the factors which characterize the nature of migration. Identification of relationship between migration parameters and an accurate selection of such parameter values increase the performance of PMAs. The number of sub-populations (NS) denotes the number of different populations in which algorithm can perform simultaneous searches. In this study, Migrating Birds Optimization (MBO) Algorithm, no migration performed, was applied for four different NS values. Additionally, Parallel Migrating Birds Optimization (PMBO) Algorithm is executed using five MR values, five MI values and four NS values and obtained fitness values are provided. According to the results, PMBO algorithm outperforms MBO in 99% of case studies. Therefore, the contribution of migration to the performance of the algorithm is evidently demonstrated. Furthermore, the values obtained during the iterations are shown on graph to investigate the effect of MI and MR changes on search performance of algorithms. As MI decreases, it is confirmed that the algorithm produces good results in early steps of iterations, making faster searches. MR has a greater effect on performance if MI is kept low. If MI increases, the changes in MR have less affect. Additionally, the effect of MI, MR, NS values and their correlation on fitness value is analyzed with analysis of variance (ANOVA). According to the analysis, MI is identified to be the most significant factor. The least significant factor is NS. Combinations of such parameters are analyzed and it was shown that MR*MI combination has the most significant effect on performance.

Keywords: Parallel Migrating Birds Optimization Algorithm, migration parameters

PARALEL GÖÇMEN KUŞLAR OPTİMİZASYON ALGORİTMASININ PARAMETRELERİNİN ETKİ ANALİZİ

Özet

Göç, paralel metasezgisel algoritmalarda (PMAs), seçilen çözümlerin bir alt popülasyondan komşu alt popülasyona belirlenen aralıklar ile gönderilmesi işlemidir. Topology, göç oranı (MR), göç aralığı (MI), göç politikası ve haberleşme modeli göç işleminin yapısını belirleyen faktörlerdir. Göç parametrelerinin aralarındaki ilişkinin belirlenmesi ve bu sayede parametre değerlerinin doğru seçimi PMAs'ların başarısını arttırmaktadır. Alt popülasyon sayısı (NS) ise algoritmanın kaç farklı popülasyonda eşzamanlı olarak arama yapacağını belirler. Bu çalışmada, göç işleminin yapılmadığı Göçmen Kuşlar Optimizasyon (MBO) algoritması dört farklı NS değeri için uygulanmıştır. Bunun yanında beş MR değeri, beş MI değeri ve dört NS değeri kullanılarak Paralel Göçmen Kuşlar Optimizasyon (PMBO) Algoritması uygulanmış, bulunan uygunluk değerleri verilmiştir. Elde edilen sonuçlara göre PMBO algoritması durumların %99'unda MBO'dan daha iyi sonuç üretmiştir. Bu sonuca göre, göç işleminin algoritma başarısına katkısı açıkça ortaya konmuştur. Ayrıca, MI ve MR değişimlerinin aramaya etkisini incelemek amacıyla iterasyon boyunca elde edilen değerler grafiksel olarak verilmektedir. MI azaldıkça algoritmanın daha hızlı arama yaparak başarılı sonuçlara erken iterasyon adımlarında ulaştığı görülmektedir. MR ise MI düşük olduğunda sonuca daha fazla etki etmektedir. MI arttığında MR değişimlerinin etkisi azalmaktadır. Bunun yanında MI, MR, NS değerlerinin ve aralarındaki etkileşimin uygunluk değeri üzerindeki etkisi varyans analizi (ANOVA) ile incelenmiştir. Yapılan analizlere göre en etkili faktörün MI olduğu ortaya konmuştur. Sonuca en az etki eden faktör ise NS olmuştur. Bu faktörlerin ikili etkileşimleri incelenmiş ve MR*MI etkileşiminin sonuca en fazla tesir eden etkileşim olduğu gösterilmiştir.

Anahtar Kelimeler: Paralel Göçmen Kuşlar Optimizasyon Algoritması, göç parametreleri

1. Introduction

Metaheuristic algorithms are commonly-used optimization algorithms for solving multidimensional and difficult problems. However, for reasons such as the difficulty of the problems and the growth of the problem sizes, there is a need for optimization algorithms that can produce better results. Considering this need, metaheuristic algorithms used as numerous sub-populations have been implemented on parallel computers. For example, there are parallel optimization studies implemented with various algorithms such as the artificial bee colony algorithm [1, 2], particle swarm optimization [3, 4], genetic algorithms [5-7], ant colony optimization [8], differential evolution [9], and chaos optimization algorithm [10].

Parallel metaheuristic algorithms (PMAs) run the algorithm used on a large number of sub-populations. At certain iteration intervals, good solutions are shared with the neighboring sub-population. This process is called migration. PMAs produce more successful results through migration. In migration step, firstly, the target sub-population is determined according to the topology used in the migration process. After each migration interval (MI) iteration, solutions as much as the migration rate (MR) are sent to target sub-population and replace according to migration policy. A general improvement has been obtained in the performance of the algorithm by sharing good solutions with the migration process with the neighboring sub-population. Additionally, the solutions newly added to the sub-population reduce the risk of getting stuck in local optimums, which is a general problem of metaheuristic algorithms [11]. In the PMAs, the factors that determine the structure of migration process are given below [12].

• Topology: It determines the sub-population to which the immigrants will go.

• Migration Rate: It determines the number of solutions that will migrate. It is expressed as a percentage based on the sub-population size.

• Migration Interval, Migration Frequency: It is the iteration step or determined criteria between the two migration processes.

• Migration Policy:
  o It is the method of selecting the solutions that will migrate.
    ▪ Choose from the best,
    ▪ Choose randomly.
  o It determines the method of displacement after migration.
    ▪ Change places with the worst,
    ▪ Change places randomly.

• Communication Model:
Synchronized / asynchronized

In this study, Migrating Birds Optimization (MBO) Algorithm, which is a metaheuristic algorithm, has been implemented on parallel computers, and named as Parallel Migrating Birds Optimization (PMBO) Algorithm. The results obtained from MBO and PMBO have been demonstrated and the contribution of migration process to algorithm performance has been shown. The acquired results from PMBO have been analyzed and presented graphically in order to determine the effect of MI and MR on algorithm performance. In addition, the results obtained from different MI, MR and number of sub-population (NS) values have been analyzed by using analysis of variance (ANOVA) and their contribution to the algorithm performance, and their interactions have been revealed. The calculations in this study have been performed using the Open MPI library on infrastructure of TÜBİTAK ULAKBİM High Performance and Grid Computing Center. The migration of good solutions between the sub-populations has been ensured using the Message Passing Interface (MPI) protocol.

2. Parallel Migrating Birds Optimization Algorithm

The MBO algorithm has been developed by modelling the behavior of migrant birds that fly in a Λ shape [13]. The MBO algorithm is an algorithm that uses a neighbor search technique. The algorithm works in the following manner: Each bird in the flock corresponds to a solution in the optimization problem. Random position has been produced for the initial *n* birds. Straight-distributed, random solutions have been acquired in the interval in which the problem is defined, using Equation (1) [14, 15].

$$x_{ij} = x_j^{min} + rand.\left(x_j^{max} - x_j^{min}\right) \tag{1}$$

In Equation (1), i is the integer in the [1, *n*] range (*n* is the number of birds in the flock), j is an integer in the [1, *d*] range (*d* is the dimension of the problem), x_{ij} is the value of the ith solution in the jth dimension, x_j^{min} and x_j^{max} are the smallest and largest values that the problem can take in dimension j, and $rand$ is a random number in the [0, 1] range [14].

One of the birds in the flock has been accepted as the leader bird. Half of the other birds have been assumed to go to the leader bird's right and remaining birds have been assumed to go to the leader bird's left side, attaining a virtual Λ shape [13, 14]. For each produced solution, z neighborhoods have been produced using Equation (2) [14, 15].

$$\hat{x}_{ij} = x_{ij} + \emptyset.\left(x_{ij} - x_{kj}\right) \tag{2}$$

In Equation (2), $\hat{x}_{ij}$ is the new vicinity to be produced at the ith solution in the jth dimension, x_{ij} is the value of the ith solution in the jth dimension, $\emptyset$ is a random number in the [-1, 1] range, and x_{kj} is the value of the kth solution in the jth dimension (k is a randomly

selected number and is an index number different from i) [14]. If the produced neighboring solution exceeds the smallest and largest values that the problem can take in the jth dimension, the $\hat{x}_{ij}$ value is held in the limits specified using Equation (3) [14, 15].

$$\hat{x}_{ij} = \begin{cases} x_j^{min}, & \hat{x}_{ij} \leq x_j^{min} \\ \hat{x}_{ij}, & x_j^{min} < \hat{x}_{ij} < x_j^{max} \\ x_j^{max}, & \hat{x}_{ij} \geq x_j^{max} \end{cases} \quad (3)$$

The fitness values of initially produced solutions and neighborhoods have been calculated. The solution with the best fitness value is on the left of the leader bird. From the best $2x$ solutions, x solutions have been transferred to the bird on the back right and x solutions have been transferred to the bird on the back left. For other birds, x solutions coming from the bird in front have been combined with the (z-x) solutions produced. The best solution has been left on the relevant bird, and the other best x solutions have been sent to a subsequent bird. Taking into consideration the possibility that the position information of neighbors overflows beyond the specified limits for the relevant dimension of the problem space, the solutions have been drawn to the specified limits using Equation (3). An iteration of the algorithm has been completed when the alternative solution production process has been completed for all the birds in the flock. When the process of iteration has been completed until the m parameter, which has been determined in advance, the leader bird change has taken place by assuming that the leader bird has been tired. In the process of change of the leader bird, the leader has been sent to the far back right of the flock, and each of the birds on the right have winged in advance to a certain place, and the new leader bird has been determined. When a subsequent leader bird change interval has come, the process has taken place on the left wing of the flock, which has been assumed to be situated in a Λ shape. When the maximum iteration number (K) has been reached or when the stopping criteria have been provided, the algorithm has been stopped, and the solution with the best fitness value from among the existing solutions has been indicated as the result of the problem [13, 14].

There are many studies carried out using the MBO algorithm [16-30]. The desire to obtain better results has produced the idea of the implementation of the MBO algorithm on parallel computers using more than one sub-population. The problem has been processed on all the processors, and by exchanging data between the processors in certain intervals, the search has been accelerated and improved. As can be seen clearly in the flow chart given in Fig. 1, firstly, in the PMBO algorithm, MBO algorithm has processed as much as MI iteration, and then good solutions as much as MR have migrated to the neighbor sub-population. The PMBO algorithm implemented in this study has been created using the island model.
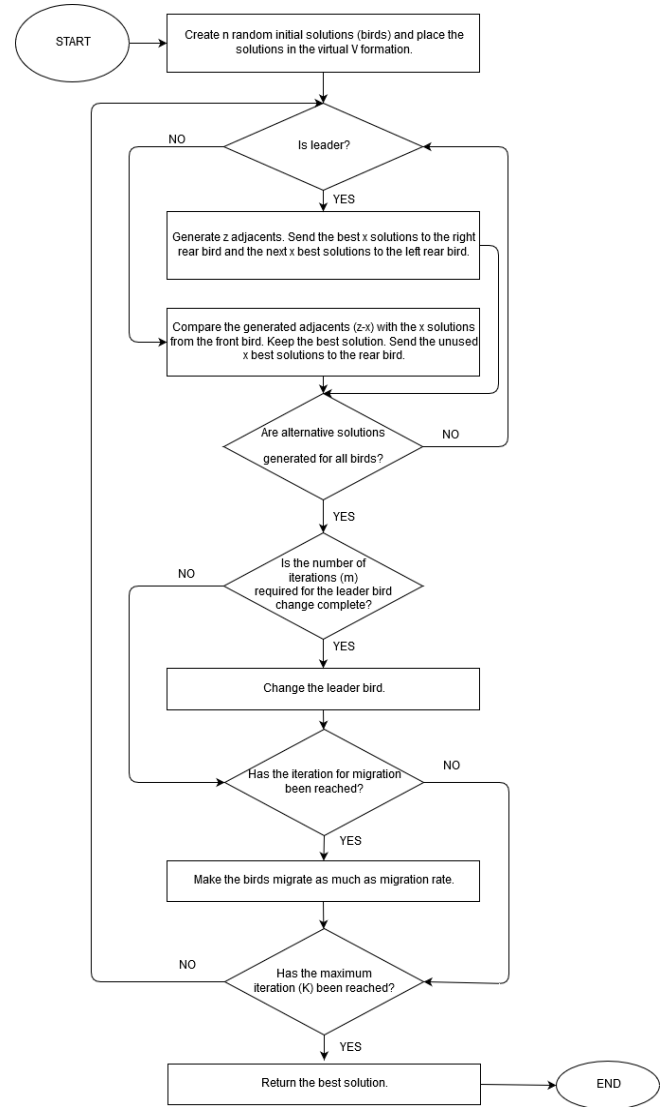

Figure 1. PMBO Algorithm flow chart.

3. Application and Acquired Results

Migration is the fundamental process that ensures PMAs produce successful results. For this reason, studies conducted over migration parameters will contribute to the increase of the performance of the algorithm. In this study, the effect of the MI and MR values used in the PMBO algorithm on the performance of the algorithm has been examined. Results for different MR values, keeping MI fixe, and the results obtained for different MI values keeping MR fixed have been provided graphically for the 16 sub-populations. Thus, the effects of the MI and MR values on the algorithm performance have been shown. In this study,

- Topology: Ring topology
- MR: 2%, 6%, 10%, 20%, 40%
- MI: 5, 10, 25, 50, 100 iterations
- NS: 8, 16, 24, 32
- Migration policy:
  - Method of selecting the solutions that will migrate:

- Choose from the best,
  - o Displacement method after migration:
    - Change places with the worst,
- Communication model: The synchronized model has been used.

The Griewank function [31] provided in Equation (4) below has been used in the study.

$$f(x) = \sum_{i=1}^{d} \frac{x_i{}^2}{4000} - \prod_{i=1}^{d} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \qquad (4)$$

$x_i \in [-600, 600]$ for all, $i = 1, \dots, d$

Global minimum $f(x^*) = 0$ at $x^* = (0, 0, \dots, 0)$

In this study, 50-dimensional Griewank function has been solved with the MBO and PMBO algorithms for different migration parameters and sub-population numbers. The sub-population size was 51. The average of the results acquired from 30 independent tests throughout 800 iterations has been provided. According to the results given in Table 1, which is indicated below, PMBO have been more successful than MBO in 99% of various situations. The PMBO have produced the most successful result for MR=40%, MI=5, and NS=32. The PMBO have produced the most unsuccessful result for the MR=2%, MI=100, and NS=8 values. The MBO have produced more successful results than the PMBO only for MR=2%, MI=100, and NS=32. When the migration parameter values have been examined, it has been concluded that as MI decreases and MR and NS increase, the PMBO algorithm produces more successful results.

Table 1. The results obtained from the Griewank function.

| Griewank Function		Number of Subpopulations			
Migration Rate	Migration Interval	8	16	24	32
MBO		0,0186989	0,0141578	0,0115492	0,00993514
2%	5	0,000328011	0,000257336	0,000222563	0,00021679
	10	0,00300654	0,0022835	0,00220351	0,0019781
	25	0,00931974	0,00804954	0,00708641	0,00638103
	50	0,0113209	0,0105354	0,00974909	0,00881062
	100	*0,0163765*	0,0127718	0,010756	*0,0113048*
6%	5	0,000018428	1,38424E-05	1,25955E-05	1,20292E-05
	10	0,00100582	0,00050734	0,000467898	0,000453067
	25	0,00542287	0,00473527	0,00446425	0,00391113
	50	0,0105623	0,00926834	0,00876217	0,00800941
	100	0,0130839	0,0115621	0,0108148	0,00988058
10%	5	0,000250173	2,68135E-06	2,38605E-06	2,0898E-06
	10	0,000250284	0,000246481	0,000217936	0,000191557
	25	0,0040936	0,00338099	0,00313018	0,00273297
	50	0,00965738	0,00776249	0,00724891	0,0054757
	100	0,0124609	0,0105963	0,00986128	0,00978441
20%	5	4,27307E-07	3,80175E-07	3,11214E-07	2,95161E-07
	10	0,000656643	6,65246E-05	6,74029E-05	5,63812E-05
	25	0,002727	0,00217109	0,00214882	0,00189009
	50	0,00692918	0,00585223	0,00539053	0,00539053
	100	0,0122193	0,00994577	0,00944142	0,0053582
40%	5	0,000328688	1,01471E-07	8,31607E-08	*7,7968E-08*
	10	0,000289962	3,41415E-05	3,21159E-05	2,91509E-05
	25	0,00174669	0,00149193	0,00136347	0,00122018
	50	0,00590917	0,00527998	0,00484464	0,0043149
	100	0,0105868	0,00946035	0,00809324	0,00711638
PMBO < MBO		25	25	25	24
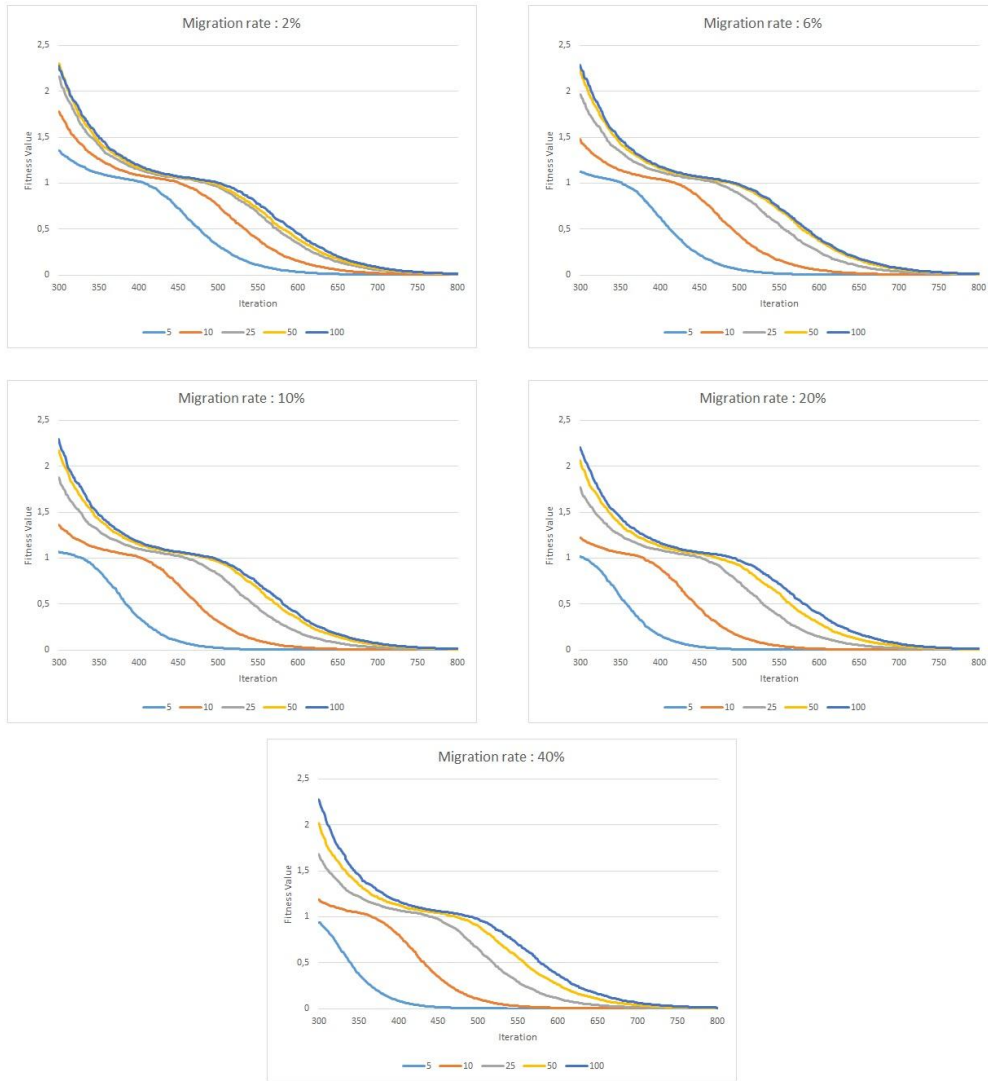
Figure 2. Migration interval analysis.

3.1. Migration Interval Effect

In this portion of the study, when MR has been constant, the effect of the change in MI values (5, 10, 25, 50, and 100) on PMBO performance has been analyzed. The graphs given in Fig. 2 are the averages of 30 independent run at the relevant iteration point for 16 sub-populations. When examining Fig. 2, the MI values have exhibited similar behavior for all MR values. Additionally, better results have been gained in the early iteration steps with the algorithm conducting a more successful search in the low MI values. When the low MR values have been examined, it has been noticed that the 25, 50, and 100 MI values produce results close to one another. When MR has increased, the effect of the MI value has become more pronounced.

3.2. Migration Rate Effect

The conclusions attained throughout the iteration from the PMBO algorithm for the 2%, 6%, 10%, 20%, and 40% rates to study the MR effect has been given in Fig. 3. According to the graphs acquired using 16 sub-populations, the algorithm performs a better search as MR increases and produces more successful results. For the MI 5 and 10 values, the search characteristic has changed as MR has changed. For MI 50 and 100, it has been realized that MR change did not affect the search behavior. As MI increases, the contribution of the MR change on the algorithm performance decreases. This result indicates that MI has been more influential than MR.
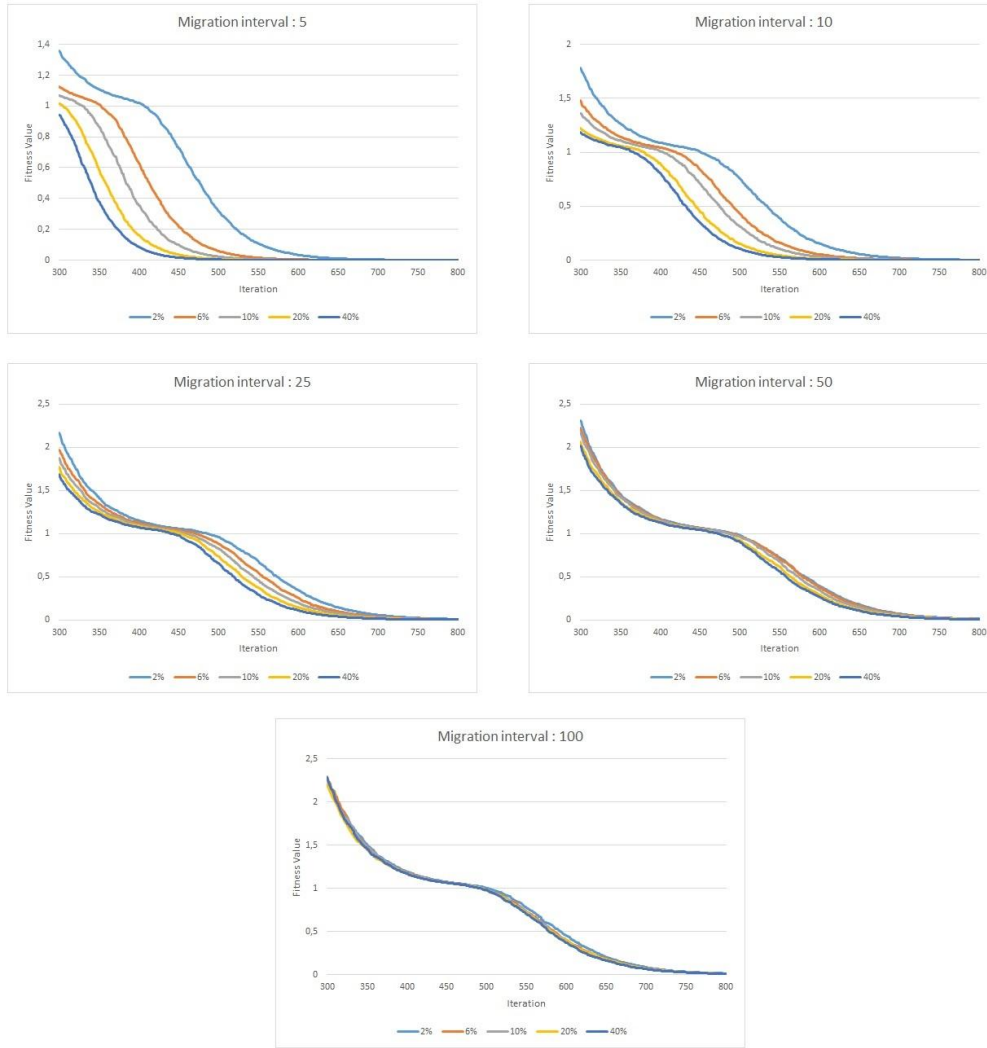
45

Figure 3. Migration rate analysis.

3.3. The Interaction of Migration Interval, Migration Rate, and Number of Sub-Population

The effect of MR and MI, which includes five different levels, and of NS, which includes four different levels, on the fitness value has been analyzed with ANOVA. According to the results provided in Table 2, it can be stated that, since the p-values for MR (F=141.38; P=0.000), MI (F=1433.03; P=0.000), and NS (F=49.61; P=0.000) are smaller than 0.05, they significantly affect the fitness value results. In addition, MR*MI (F=14.58; P=0.000) and MI*NS (F=8.93; P=0.000) interactions cause significant effect on the fitness value. On the grounds that the P value for MR*NS (F=0.87; P=0.580) is greater than 0.05, it has been concluded that it does not play a significant role over fitness value change.

Thereby, it can be stated that the effects of the MR, MI, NS, MR*MI, and MI*NS factors and interactions on the average fitness value have been meaningful. As a result of the variance analysis, it has obtained the values of R-Sq 99.30% and R-Sq(Adj) 98.55%. These values demonstrate that the variability is quite well explained [32].

Fig. 4 given below shows the effect of MR, MI, and NS on fitness value. The effect of the changes in MI with the highest value of F (1433.03) on the fitness value can be clearly noticed. The changes in MR with the lower value of F (141.38) has affected the fitness value less. NS has the lowest F (49.61) value. As a result of this, the effect of the changes in NS on the fitness value is limited.

Table 2. Analysis of variance results.

| Source	DF	Seq SS	Adj SS	Adj MS	F	P
MR	4	0,0001589	0,0001589	0,0000397	141,38	0,000
MI	4	0,0016104	0,0016104	0,0004026	1433,03	0,000
NS	3	0,0000418	0,0000418	0,0000139	49,61	0,000
MR*MI	16	0,0000655	0,0000655	0,0000041	14,58	0,000
MR*NS	12	0,0000029	0,0000029	0,0000002	0,87	0,580
MI*NS	12	0,0000301	0,0000301	0,0000025	8,93	0,000
Error	48	0,0000135	0,0000135	0,0000003		
Total	99	0,0019232				
S = 0,000530043		R-Sq = 99,30%		R-Sq(adj) = 98,55%		
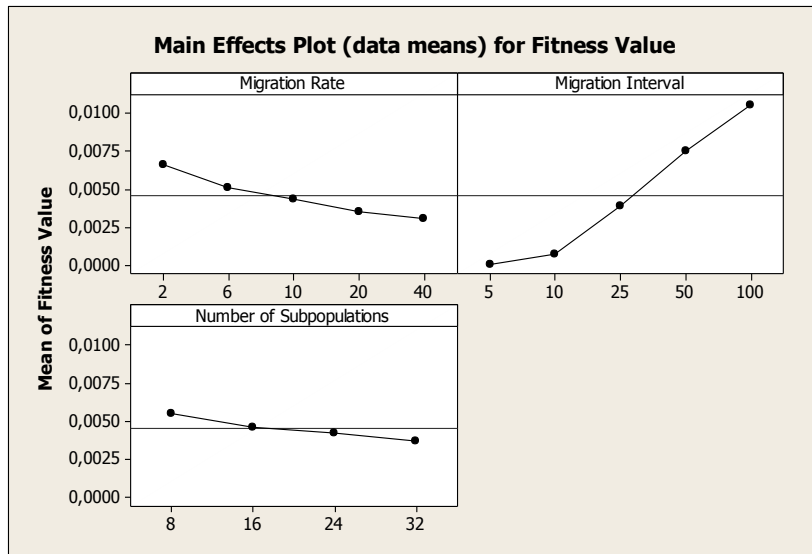
Figure 4. Main effect graph.

Fig. 5 below provides the interaction between MI, MR, and NS, which affects the success of the PMAs. When examining the MR*MI interaction, it can be realized that it produces a value of F=14.58 and produces good results without being affected by the MR value in the low MI values. When MI increases, the MR effect emerges. When regarding the MI*NS (F=8.93) interaction when MI is low, it can be noticed that the NS value affects the fitness value level the least. When MI increases, the NS effect increases, as well. When Fig. 5 is examined as a whole, it is clearly seen that the influence of MI over the fitness value is much greater compared with MR and NS.

4. Conclusion

In this study, the effect of migration process, which substantially affects the performance of PMAs, on PMBO has been analyzed. First, without migration process, the respective performances of MBO algorithm executed for various populations and the respective performances of PMBO with different NS (8, 16, 24, 32), MI (5, 10, 25, 50, 100) and MR (2%,6%,10%,20%,40%) have been provided. According to these results, PMBO outperforms MBO in 99 out of 100 cases. And also, it has been confirmed that PMBO performs better as MI decreased,

and MR and NS values increased. Besides; the effects of changes in MI value when MR is constant, and the effects of changes in MR value when MI is constant on search performance have been shown on graph to investigate independent effects of MI and MR. According to the results obtained by using 16 sub-populations, the algorithm has been performed better searches for lower MI values. If MI increases, the effect of change in MR values on search performance decreases.

ANOVA has been used to analyze the effect of migration parameters at different levels and NS values of PMBO on fitness value. According to the results obtained from the analysis, it has been established that MI, MR and NS have a significant effect on fitness value results since P values are less than 0.05. Upon analysis of the effects of MR*MI and MI*NS combinations, P has been found to be 0.000. The P value indicates that the correlations in question significantly affect fitness values of the results. Since the P value of MR*NS interaction has been greater than 0.05, it does not make a significant effect on fitness value. According to these results; the selection of parameter values with significant interactions emerges as a new subject to study. In the further studies, it is

intended to use experimental design methods to determine the parameters with significant interactions.
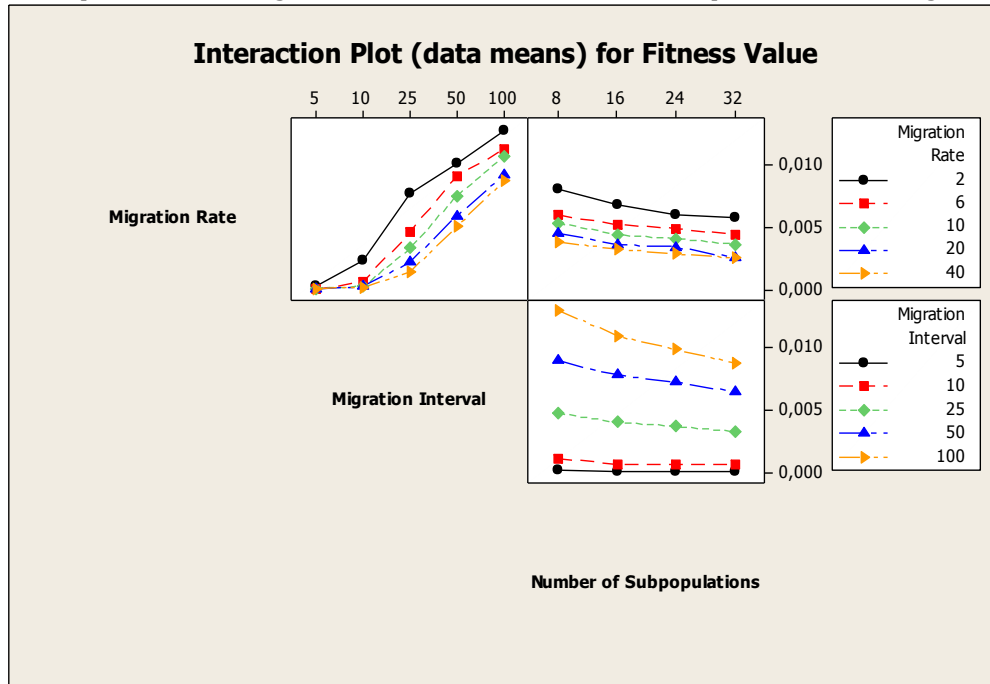

Figure 5. Interaction graph.

5. Acknowledgment

6. References

[1] Baştürk, A. and Akay, R., "Performance Analysis of the Coarse-Grained Parallel Model of the Artificial Bee Colony Algorithm", *Information Sciences*, 253, 34-55, 2013.

[2] Asadzadeh, L., "A Parallel Artificial Bee Colony Algorithm for the Job Shop Scheduling Problem with a Dynamic Migration Strategy", *Computers & Industrial Engineering*, 102, 359-367, 2016.

[3] Gülcü, Ş. and Kodaz, H., "A Novel Parallel Multi-Swarm Algorithm Based on Comprehensive Learning Particle Swarm Optimization", *Engineering Applications of Artificial Intelligence*, 45, 33-45, 2015.

[4] Tu, K.Y. and Liang, Z.C., "Parallel Computation Models of Particle Swarm Optimization Implemented by Multiple Threads", *Expert Systems with Applications*, 38(5), 5858-5866, 2011.

[5] Alba, E., Nebro, A.J. and Troya, J.M., "Heterogeneous Computing and Parallel Genetic Algorithms", *Journal of Parallel and Distributed Computing*, 62(9) 1362-1385, 2002.

[6] Doorly, D.J. and Peiró, J., "Supervised Parallel Genetic Algorithms in Aerodynamic Optimisation", *Artificial Neural Nets and Genetic Algorithms*, Vienna, 1998.

[7] Mühlenbein, H., Schomisch, M. and Born, J., "The Parallel Genetic Algorithm as Function Optimizer", *Parallel Computing*, 17(6-7) 619-632, 1991.

[8] Randall, M. and Lewis, A., "A Parallel Implementation of Ant Colony Optimization", *Journal of Parallel and Distributed Computing*, 62(9), 1421-1432, 2002.

[9] Wang, H., Rahnamayan, S. and Wu, Z., "Parallel Differential Evolution with Self-Adapting Control Parameters and Generalized Opposition-Based Learning for Solving High-Dimensional Optimization Problems", *Journal of Parallel and Distributed Computing*, 73, 62-73, 2013.

[10] Xiaofang, Y., Ting, Z., Yongzhong, X. and Xiangshan, D., "Parallel Chaos Optimization Algorithm with Migration and Merging Operation", *Applied Soft Computing*, 35, 591-604, 2015.

[11] Rebaudengo, M. and Reorda, M.S., "An Experimental Analysis of the Effects of Migration in Parallel Genetic Algorithms", *1993 Euromicro Workshop on Parallel and Distributed Processing*, Gran Canaria, 1993.

[12] Alba, E. and Troya, J.M., "Improving Flexibility and Efficiency by Adding Parallelism to Genetic Algorithms", *Statistics and Computing*, 12(2), 91-114, 2002.

[13] Duman, E., Uysal, M. and Alkaya, A.F., "Migrating Birds Optimization: A New Metaheuristic Approach and Its Performance on Quadratic Assignment Problem", *Information Sciences*, 217, 65-77, 2012.

[14] Makas, H., "*Güncel En İyileme Algoritmalarının Paralel ve Birlikte Uygulamaları ve Performans Analizleri*", Doktora Tezi, Sakarya Üniversitesi Fen Bilimleri Enstitüsü, Bilgisayar ve Bilişim Mühendisliği Anabilim Dalı, Sakarya, 2015.

[15] Karaboğa, D., *Yapay Zeka Optimizasyon Algoritmaları*, Nobel Akademik Yayıncılık, Ankara, 2014.

[16] Öz, D., "Göç Eden Kuşlar Algoritmasında Kaos Fonksiyonlarının Kullanılması," *Gazi Üniversitesi Fen Bilimleri Dergisi*, 225-233, 2016.

[17] Tongur, V. and Ülker, E., "Tek Boyutlu Kesme Probleminin Göçmen Kuşlar Optimizasyon Algoritması ile Çözümü", *Uluslararası Bilgisayar Bilimleri ve Mühendisliği Konferansı*, Tekirdağ, 2016.

[18] Tongur, V. and Ülker, E., "The Analysis of Migrating Birds Optimization Algorithm with Neighborhood Operator on Traveling Salesman Problem", *Intelligent and Evolutionary Systems*, 5, 227-237, 2015.

[19] Makas, H. and Yumuşak, N., "New Cooperative and Modified Variants of The Migrating Birds Optimization Algorithm", *Electronics, Computer and Computation (ICECCO)*, 2013 International Conference, Ankara, 2013.

[20] Gao, L. and Pan, Q.-K., "A Shuffled Multi-Swarm Micro-Migrating Birds Optimizer for A Multi-Resource-Constrained Flexible Job Shop Scheduling Problem", *Information Sciences*, 372, 655-676, 2016.

[21] Zhang, B., Pan, Q.-K., Gao, L., Zhang, X.-L., Sang, H.-Y. and Li, J.-Q., "An Effective Modified Migrating Birds Optimization for Hybrid Flowshop Scheduling Problem with Lot Streaming", *Applied Soft Computing*, 52, 14-27, 2017.

[22] Gao, K., Suganthan, P.N. and Chua, T.J., "An Enhanced Migrating Birds Optimization Algorithm for No-Wait Flow Shop Scheduling Problem", *2013 IEEE Symposium on Computational Intelligence in Scheduling (CISched)*, Singapore, 2013.

[23] Pan, Q.-K. and Dong, Y., "An Improved Migrating Birds Optimisation for A Hybrid Flowshop Scheduling with Total Flowtime Minimisation", *Information Sciences*, 277, 643-655, 2014.

[24] Ramanathan, L. and Ulaganathan, K., "Nature-Inspired Metaheuristic Optimization Technique-Migrating Bird's Optimization in Industrial Scheduling Problem", *SSRG International Journal of Industrial Engineering (IJIE)*, 1(3) 1-6, 2014.

[25] Benkalai, I., Rebaine, D., Gagné, C. and Baptiste, P., "The Migrating Birds Optimization Metaheuristic for The Permutation Flow Shop with Sequence Dependent Setup Times", *IFAC-PapersOnLine*, 49(12), 408-413, 2016.

[26] Tongur, V. and Ülker, E., "Migrating Birds Optimization for Flow Shop Sequencing Problem", *Journal of Computer and Communications*, 02(04), 142-147, 2014.

[27] Niroomand, S., Hadi-Vencheh, A., Şahin, R. and Vizvári, B., "Modified Migrating Birds Optimization Algorithm for Closed Loop Layout with Exact Distances in Flexible Manufacturing Systems", *Expert Systems with Applications*, 42(19), 6586-6597, 2015.

[28] Öz, D., "An Improvement on The Migrating Birds Optimization with A Problem-Specific Neighboring Function for The Multi-Objective Task Allocation Problem", *Expert Systems with Applications*, 67, 304-311, 2017.

[29] Makas, H. and Yumuşak, N., "System Ident Cation by Using Migrating Birds Optimization Algorithm: A Comparative Performance Analysis", *Turkish Journal of Electrical Engineering & Computer Sciences*, 24, 1879-1900, 2016.

[30] Makas, H. and Yumuşak, N., "Balancing Exploration and Exploitation by Using Sequential Execution Cooperation Between Artificial Bee Colony and Migrating Birds Optimization Algorithms", *Turkish Journal of Electrical Engineering and Computer Sciences*, 24(6), 4935-4956, 2016.

[31] https://www.sfu.ca/~ssurjano/griewank.html [online], (06.10.2019).

[32] Erdem, İ., *Minitab Uygulamalı İstatistik Yöntemler*, Seçkin Yayıncılık, Ankara, 2017.