

PARALEL MAKİNALARIN GENETİK ALGORİTMA İLE ÇİZELGELENMESİNDE MUTASYON ORANININ ETKİNLİĞİ

EFFICIENCY OF MUTATION RATE FOR PARALLEL MACHINE SCHEDULING WITH GENETIC ALGORITHM

Öğr. Gör. Dr. Murat KOCAMAZ, Ege Üniversitesi İktisadi ve İdari Bilimler
Fakültesi İşletme Bölümü, murat.kocamaz@ege.edu.tr

Araş. Gör. Ural Gökay ÇİÇEKLİ, Ege Üniversitesi İktisadi ve İdari Bilimler
Fakültesi İşletme Bölümü, gokay.cicekli@ege.edu.tr

ÖZET

Genetik algoritma (GA), karmaşık olarak bilinen paralel makinaların çizelgelenmesi problemlerinin çözümlenmesinde kullanılan sezgisel bir yöntemdir. GA, sahip olduğu operatörlerin gerçekleşme oranlarına bağlı olarak olumlu veya olumsuz performans göstermektedir. Bu operatörden bir tanesi de mutasyon oranıdır. Bu çalışmada paralel makinaların çizelgelenmesinde mutasyon oranının genetik algoritma performansı üzerine etkisi araştırılmıştır.

Anahtar Kelimeler: genetik algoritma, mutasyon oranı

ABSTRACT

Genetic algorithms (GA) have been extensively used in parallel machine scheduling as a type of heuristic method. Depends on rate of operators, GA represents affirmative or negative performance. One of these operators is mutation rate. In this study, we addressed efficiency of mutation rate on GA for parallel machine scheduling.

Keywords: genetic algorithm, mutation rate

1. GİRİŞ

Çizelgeleme üretim işletmeleri için en önemli fonksiyonlardan birisidir. Çizelgeleme, kıt kaynakların belirli bir zaman boyunca işlere tahsis edilmesiyle ilgili ve bir veya daha fazla hedefin optimizasyonunu amaçlayan bir karar alma sürecidir (Pinedo, 2008:1). Bir harmonik düzenleme konusu olan üretim çizelgelemede, çıktıyı üretmek için doğru görevlerin, doğru zamanda, doğru unsurlar ile yönetilmesi ile ilgilidir (Sheikh, 2003:474). Çizelgeleme problemleri için makine ve süreçlere bağlı olarak belirli sınıflamalar yapılmıştır. Bu sınıflamalar arasında en önemlilerinden bir tanesi gerçek hayatta kullanılmasından dolayı paralel makinalarda çizelgeleme problemleridir.

Paralel makinaların çizelgenmesi problemleri, n işin işlem göreceği paralel makinalara (M_1, M_2, \dots, M_m) göre tamamlanma zamanları (p_1, p_2, \dots, p_n) ile sıralanmasını içerir (Zhenbo and Wenxun, 2006).

Çizelgeleme problemleri farklı karmaşıklık düzeylerine sahip olmaktadır. Çizelgeleme problemlerinin sahip oldukları karmaşıklık düzeyi arttıkça optimum sonuca ulaşmayı zorlaştırmaktadır. Karmaşıklık düzeyi yüksek bu tür çizelgeleme problemlerinin çözümü için birçok yöntem kullanılmaktadır. Çizelgeleme gibi kombinasyonel problemler GA ile çözülebilmektedir (Mitchell, 1998).

GA karmaşık arama uzayında en iyi çözümü bulmayı amaçlayan rastgele arama tekniğidir (Mori and Tseng, 1997). Bazı problemlerde deterministik metotlar ile çözümü bulmak zordur. GA bu tür problemlere çözüm bulmayı sağlar. GA çizelgeleme gibi kombinasyonel optimizasyon problemlerini çözmek için kullanılabilir (Mitchell, 1998).

GA'lar optimal sonucu garanti etmezler, fakat genellikle optimal sonuca yakın değerler elde edilmektedir. Bunun yanında, tüm uzayı taramadıklarından dolayı etkin arama yaparak geleneksel yöntemlere göre çok daha kısa bir sürede çözüme ulaşırlar (Goldberg, 1989: 3). GA, zor ve büyük bir çözüm uzayında optimuma yakın bir çözümü hızlı bir şekilde bulabilir. Günümüzde, rekabetin yoğun bir şekilde yaşandığı iş ortamı, işletmeleri optimum fakat yavaş olan bir çözüm yerine, hızlı ve kabul edilebilir bir çözüm bulmaya yönlendirmektedir.

GA alt optimizasyona odaklanmak yerine, amaç fonksiyonu için optimum değer bulmayı hedefler. GA, çözümlerinde optimumu garanti etmemektedir. Bunun yerine optimuma yakın sonuçlar vermektedir (Yeo and Agyei, 1998). GA ile karmaşık problemlerin çözümünde uygun metot ile kodlamanın seçimi ve amaç fonksiyonun belirlenmesi önemlidir. Bunlar, değişkenlerin kendisi ile değil, değişkenlerin kodları ile ilgilidir (Wang, 2003). Bu yüzden GA sürekli olmayan fonksiyonları içeren problemlere de uygulanabilmektedir (Yeo and Agyei, 1998). GA sadece en iyi bireyi bir sonraki nesile taşımak ile ilgilidir.

PARALEL MAKİNALARIN GENETİK ALGORİTMA İLE
ÇİZELGELENMESİNDE MUTASYON ORANININ ETKİNLİĞİ

GA üreme, çaprazlama ve mutasyon olmak üzere üç operatör kullanılır. Mutasyon operatörü genetik algoritmada önemli rol oynayan süreçlerden birisidir. Özellikle mutasyon oranının değeri, GA performansını doğrudan etkilemektedir.

2. PARALEL MAKİNALARIN GENETİK ALGORİTMA İLE ÇİZELGELENMESİ

GA kodlama ile başlamaktadır. Problemin yapısına bağlı olarak en uygun kodlama kullanılması gerekmektedir. Bu yüzden genetik algoritmayı uygulamaya başlamadan önce, ikili değer kodlama, permutasyon kodlama ve ağaç kodlama gibi kodlama çeşitleri arasından uygun kodlama metodunu bulmak önemlidir. Permutasyon kodlama sıralama problemleri için en uygun yöntemdir (Borovska,2006:2). Bu yüzden çizelgeleme problemlerinde permutasyon kodlama kullanılmaktadır.

Paralel makinelerde çizelgeleme problemleri üretim süresini minimize etmeyi amaçlamaktadırlar. Her makinadaki süreç zamanı ayrı olarak hesaplanır ve en büyük süreç zamanı toplam üretim süresidir. Bu, paralel makinelerde çizelgeleme için hazırlık sürelerinin dikkate alınmadığı amaç fonksiyon değeridir.

Paralel makinelerin çizelgelenmesinde genetik algoritmanın işleyişini göstermek üzere 3 makina 5 işlik bir örnek ele alınmıştır. Her makinanın her işi tamamlama zamanı farklıdır. İşlerin makinalara dağılımı Tablo 1'de görüldüğü gibidir.

Tablo 1 - 3 makine 5 iş örnek çizelgelemesi

Makina 1	İş 5	
Makina 2	İş 4	İş 1
Makina 3	İş 2	İş 3

GA'ya uygun halde kromozomların gösterimi için her makinaya iş sayısı kadar gen ayrılır. Toplam kromozom uzunluğu makina sayısı ile iş sayısının çarpımına kadardır. Kullanılmayan kapasiteyi temsil eden genler "Boş" olarak gösterilir. Aşağıda ele alınan örneğin kromozomu gösterilmektedir.

Şekil 1 - İşlerin Makinalara Dağılımının Kromozom şeklinde gösterimi

İş 5 | boş | boş | boş | boş | İş 4 | İş 1 | boş | boş | boş | İş 2 | İş 3 | boş | boş | boş

Permutasyon kodlama için çok sayıda genetik çaprazlama operatör çeşidi bulunmaktadır. Bu çalışmada pozisyon temelli çaprazlama kullanılmıştır. Pozisyon temelli çaprazlama yeni nesiller için hangi genin baskın olduğunu belirleyen rastsal olarak seçilmiş baskınlık koduna ihtiyaç duyar. Eğer baskınlık

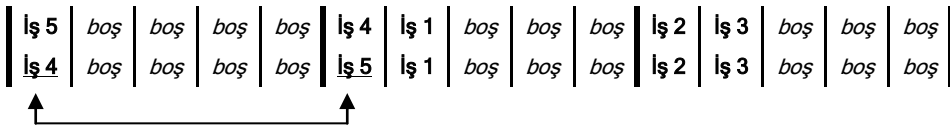
kodundaki değer “1” ise birinci ebeveyn baskın, değer “0” ise ikinci ebeveyn baskındır. Birinci nesil yaratılırken birinci ebeveynin baskınlığı, ikinci yeni nesil yaratılırken ikinci ebeveynin baskınlığı söz konusudur. Birinci nesil yaratılırken baskınlık kodunda “1” görülen genler direkt olarak birinci ebeveyninden alınır. Geri kalan boş yerler ikinci ebeveyndeki sıra göz önünde bulundurularak tekrar olmayacak şekilde doldurulur. Şekil 2’de pozisyon temelli çaprazlama kullanılarak iki tane yeni neslin yaratılması gösterilmektedir. Görüldüğü gibi iki çeşit kod kullanılmıştır. Bunlardan “İş” kelimesi ile başlayanlar var olan fiziksel işleri gösterirken, “100” lü sayılar ise atanmamış işleri (boş iş kapasitesi) göstermektedir.

Şekil 2 - Pozisyon temelli çaprazlama kullanılarak yapılan bir döngü sonucunda yaratılan iki yeni nesil

Ebeveyn 1	İş 5	101	102	103	104	İş 4	İş 1	105	106	107	İş 2	İş 3	108	109	110
Ebeveyn 2	İş 4	İş 5	İş 2	101	102	İş 3	103	104	105	106	İş 1	107	108	109	110
Baskınlık Kodu	1	1	0	0	1	0	0	1	0	1	0	0	1	1	1
Nesil 1	İş 5	101	İş 4	İş 2	104	102	İş 3	105	103	107	106	İş 1	108	109	110
Nesil 2	İş 5	102	İş 2	101	104	İş 3	103	İş 4	105	106	İş 1	107	108	109	110

Her döngüde baskınlık şeridi kullanılarak iki ebeveyninden iki tane yeni nesil meydana getirilir. Bunun dışında her döngüde rastgele seçilen iki genin yer değiştirilmesi ile yapılan mutasyon sonucunda bir tane daha yeni nesil oluşturulur. Bu mutasyon işlemi belirlenen oran olasılığındadır. Mutasyon işlemi sırasında kromozomdaki her gene bir değer verilmekte ve bu değer mutasyon olasılığı kapsamında ise o gen mutasyona girmektedir. Şekil 3’de görüldüğü üzere İş 4 ile İş 5 mutasyon sonucu yer değiştirmişlerdir. Çaprazlama ve mutasyon sonucu elde edilen toplam 5 kromozomdan en iyi ikisi bir sonraki nesile aktarılır.

Şekil 3 - Kromozomlardaki Gerçekleşen Mutasyonun İşleyişi (devamı???)



3. GENETİK ALGORİTMA PERFORMANSINDA MUTASYON ORANININ ETKİNLİĞİ

GA’da mutasyon, seçim süreci sırasında popülasyonda kaybolmuş genleri yerine koyma veya başlangıç popülasyonunda bulunmayan gen dizilimlerini ortaya çıkarma gibi kritik iki görevi yerine getirir (Gen ve Cheng, 1997:4). Ruiz ve

PARALEL MAKİNALARIN GENETİK ALGORİTMA İLE
ÇİZELGELENMESİNDE MUTASYON ORANININ ETKİNLİĞİ

Maroto'da (2006:788) benzer bir ifade ile, yerel optimumu engellemek ve yeni gen dizilimlerini ortaya çıkarmak için mutasyon operatöründen faydalanılması gerektiğini vurgulamaktadır. Mutasyon olmadan popülasyonlar ebeveynlere bağımlılık ve genlerin silinmesi nedeniyle alelleri (Genlerin alabileceği değerleri) kaybedebilirler (Hamilton ve Ridley, 2005:14).

Gen (1996)'e göre mutasyon oranı yeni genlerin popülasyona deneme amaçlı sunulma oranını kontrol eder. Bu oran çok düşük ise, faydalı bir çok gen hiç denenmeyecek, bunun tam tersi bu oran çok yüksek olursa, daha çok tesadüf tedirginlik oluşacak, yeni nesil ebeveynlere benzemeyi kaybetmeye başlayacaktır ve bunun sonucunda algoritma zamanla araştırma geçmişinden öğrenme yeteneğini kaybedecektir.

Hamilton ve Ridley (2005) GA'da mutasyon oranının doğada olduğu gibi genellikle oldukça düşük olduğunu belirtmektedir. Bunun nedeni yönetim altında olmayan yüksek mutasyon oranı ile mutasyonun GA'nın performansını geliştirmekten çok ona engel olma olasılığı daha fazladır. (Hamilton ve Ridley, 2005:14,15)

Sadegheih (2006)'e göre performans, değerlerin iyi olduğu aralıklar arasında mutasyon oranının değişmesinden etkilenmez ama değerlerin iyi olduğu aralıklar dışında GA performansının bozulmasına neden olabilir (Sadegheih, 2005:153).

Şekil 4 - Mutasyon oranının amaç fonksiyon değerine etkisi

“Kötü” Mutasyon Oranı (0 - 0,001)	“İyi” Mutasyon Oranı 0,005 - 0,01	“Kötü” Mutasyon Oranı (0,02 - 1)
---	---	--

4. BULGULAR

Çalışmada kullanılmak amacıyla 20 iş ve 3 paralel makinadan oluşan bir örnek ele alınmıştır. Her makinanın farklı işleri tamamlama zamanları farklıdır. Bununla birlikte tüm işler tüm makinalarda yerine getirilememektedir. Bu durum karşısında işlerin tamamlanma zamanlarına oldukça yüksek değerler verilerek o işin o makinaya verilmesi engellenmektedir. Makinaların işlere göre üretim süreleri Tablo 2'de verilmektedir. Tablo 2'deki 1000 değerleri atanmış işin o makinada tamamlanmasının mümkün olmadığını ifade etmektedir.

Tablo 2 - Makinaların İşlere Göre Üretim Süreleri

	İŞLER									
	1	2	3	4	5	6	7	8	9	10
Makina 1	150	65	300	30	1000	45	140	48	240	180
Makina 2	170	55	290	45	90	60	130	1000	220	190
Makina 3	180	60	280	1000	100	50	120	56	210	220

	İŞLER									
	11	12	13	14	15	16	17	18	19	20
Makina 1	210	245	1000	180	160	140	120	140	280	85
Makina 2	205	270	90	200	150	1000	115	160	245	95
Makina 3	195	1000	100	210	170	160	100	175	255	105

GA her yeni döngü dizisine başlarken iki tane başlangıç ebeveynine ihtiyaç duymaktadır. Çalışmamızda seçilen başlangıç ebeveynlerindeki işlerin makinalara dağılımı rastsal olarak belirlenmiştir. Başlangıç ebeveynlerin çizelgeleme planları Tablo 3'de görülebilmektedir.

Tablo 3 - Başlangıç Ebeveynlerin Çizelgeleme Planları

Ebeveyn 1	Makina 1	İş 1	İş 2	İş 20	İş 12	İş 18	İş 9	İş 7
	Makina 2	İş 3	İş 4	İş 19	İş 14	İş 10	İş 17	
	Makina 3	İş 5	İş 6	İş 13	İş 11	İş 16	İş 8	İş 15
Ebeveyn 2	Makina 1	İş 2	İş 12	İş 14	İş 7	İş 10	İş 17	
	Makina 2	İş 1	İş 4	İş 16	İş 8	İş 11	İş 20	İş 15
	Makina 3	İş 3	İş 5	İş 6	İş 9	İş 18	İş 13	

İşler arasında geçişler yapılırken hazırlık süreleri meydana gelmektedir. Çalışmamızda işler benzerliklerine göre gruplara ayrılmıştır. Aynı grupta olan işler arasındaki geçişlerde hazırlık süresi yaşanmamaktadır. Tablo 4'de işler arasında meydana gelen hazırlık süreleri yer almaktadır. Tabloda ilk sütun yapılan işi, ilk satır ise geçiş yapılacak işi göstermektedir.

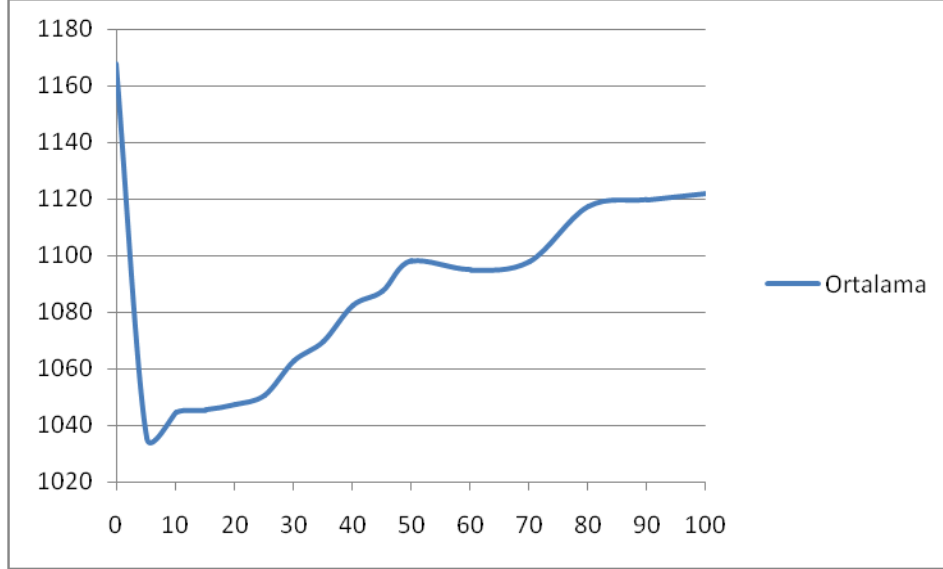
PARALEL MAKİNALARIN GENETİK ALGORİTMA İLE
ÇİZELGELENMESİNDE MUTASYON ORANININ ETKİNLİĞİ

Tablo 4 - İş Geçişlerinde Meydana Gelen Hazırlık Süreleri

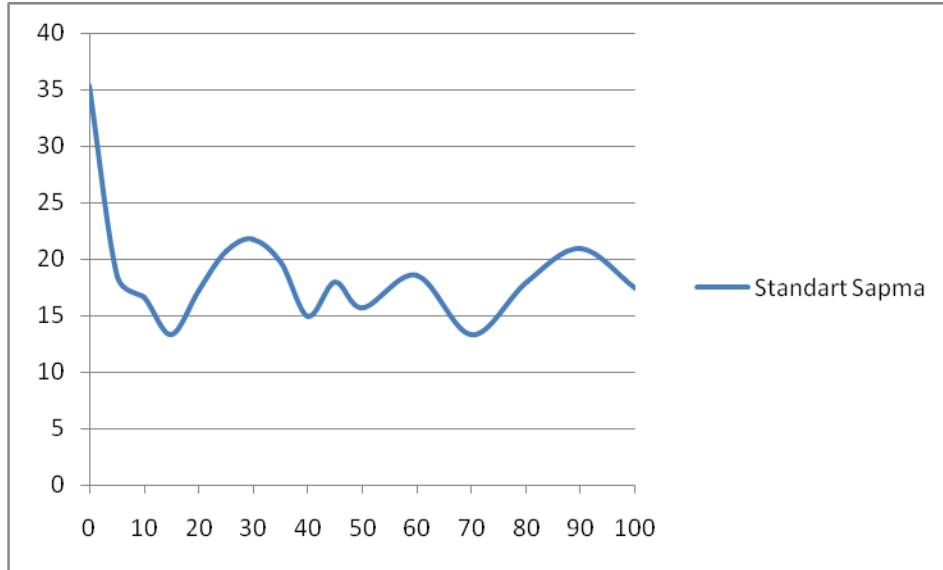
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	0	15	0	30	45	45	15	30	0	45	15	0	30	30	45	0	15	30	45	0
2	20	0	20	15	30	30	0	15	20	30	0	20	15	15	30	20	0	15	30	20
3	0	15	0	30	45	45	15	30	0	45	15	0	30	30	45	0	15	30	45	0
4	40	20	40	0	15	15	20	0	40	15	20	40	0	0	15	40	20	0	15	40
5	60	40	60	20	0	0	40	20	60	0	40	60	20	20	0	60	40	20	0	60
6	60	40	60	20	0	0	40	20	60	0	40	60	20	20	0	60	40	20	0	60
7	20	0	20	15	30	30	0	15	20	30	0	20	15	15	30	20	0	15	30	20
8	40	20	40	0	15	15	20	0	40	15	20	40	0	0	15	40	20	0	15	40
9	0	15	0	30	45	45	15	30	0	45	15	0	30	30	45	0	15	30	45	0
10	60	40	60	20	0	0	40	20	60	0	40	60	20	20	0	60	40	20	0	60
11	20	0	20	15	30	30	0	15	20	30	0	20	15	15	30	20	0	15	30	20
12	0	15	0	30	45	45	15	30	0	45	15	0	30	30	45	0	15	30	45	0
13	40	20	40	0	15	15	20	0	40	15	20	40	0	0	15	40	20	0	15	40
14	40	20	40	0	15	15	20	0	40	15	20	40	0	0	15	40	20	0	15	40
15	60	40	60	20	0	0	40	20	60	0	40	60	20	20	0	60	40	20	0	60
16	0	15	0	30	45	45	15	30	0	45	15	0	30	30	45	0	15	30	45	0
17	20	0	20	15	30	30	0	15	20	30	0	20	15	15	30	20	0	15	30	20
18	40	20	40	0	15	15	20	0	40	15	20	40	0	0	15	40	20	0	15	40
19	60	40	60	20	0	0	40	20	60	0	40	60	20	20	0	60	40	20	0	60
20	0	15	0	30	45	45	15	30	0	45	15	0	30	30	45	0	15	30	45	0

GA her tekrarda 2000 döngü olmak üzere farklı mutasyon oranlarında toplam 870 defa çalıştırılmıştır. GA, ilk olarak mutasyon oranı %0 - %100 arasındaki tüm %5'lik dilimlerde çalıştırılmıştır. GA ile oluşturulan modelin, 21 farklı mutasyon oranının 30 tekrar çalıştırılmasıyla elde edilen farklı sonuçların ortalamaları Şekil 5'de verilmektedir. %10 - %100 mutasyon aralığında, ortalama üretim süresi değerleri git gide artmaktadır. Şekil 6'da ise elde edilen bu sonuçların standart sapma değerleri verilmektedir.

Şekil 5 - %0 - %100 Mutasyon Oranı Arasındaki Sonuçların Ortalaması

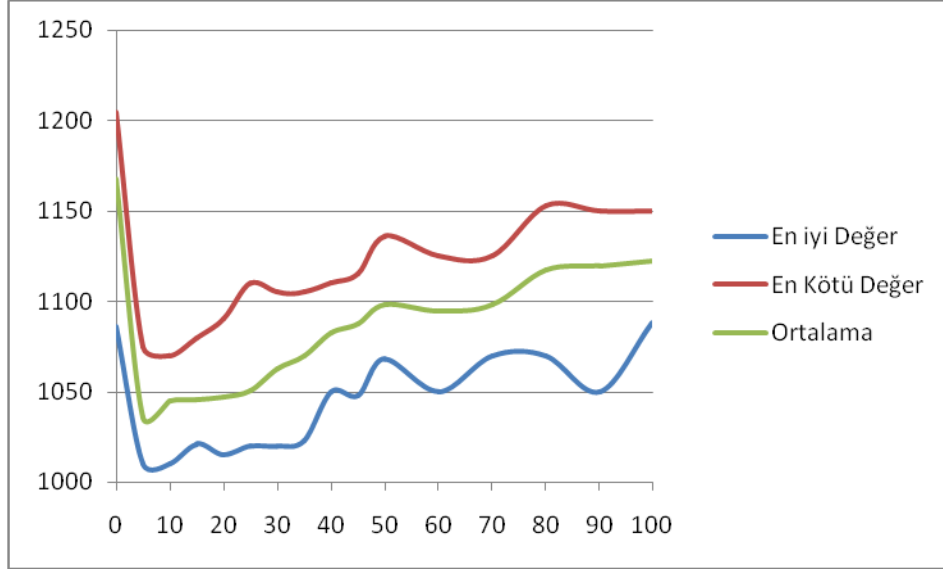


Şekil 6 - %0 - %100 Mutasyon Oranı Arasındaki Sonuçların Standart Sapmaları



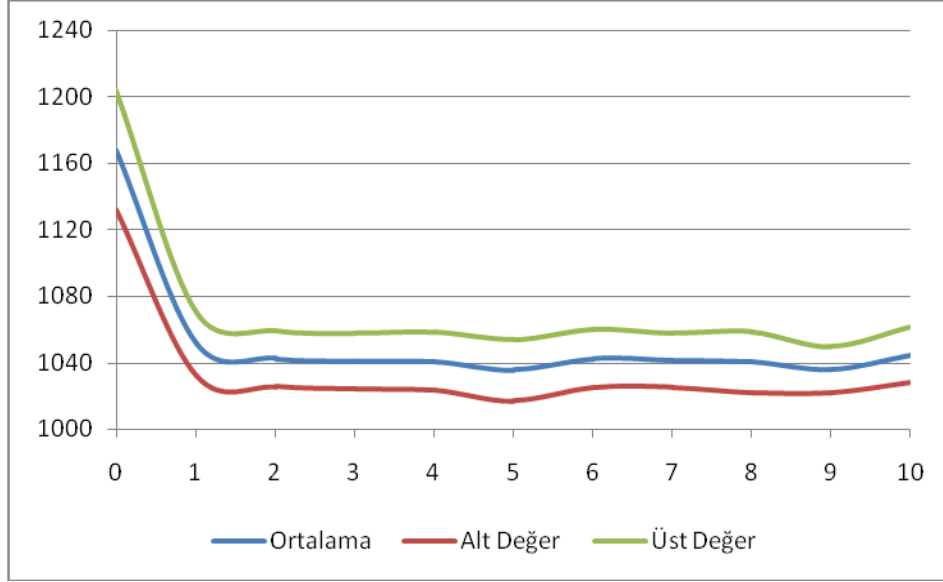
Farklı mutasyon oranları ile çalıştırılan GA sonucu elde edilen tekrarlardaki en iyi değerlerin ve en kötü değerlerin ortalama değer ile karşılaştırılması Şekil 7'de gösterilmektedir.

Şekil 7 - %0 - %100 Mutasyon Oranı Arasında Elde Edilen En İyi ve En Kötü Değerlerin Ortalama Değerler ile Karşılaştırılması



İlk aşama sonucunda en düşük üretim sürelerinin %0 - %10 arasında olduğu görülmektedir. Bu sebep ile %0 - %10 arasındaki mutasyon oranları ayrı olarak incelenmiştir. %0 - %10 arasındaki mutasyon oranları %1 aralıklarla tekrar çalıştırılmıştır. Yeni verilerden elde edilen ortalama değerler ve bu değerlerin ± 1 standart sapma aralığı Şekil 8'de görülmektedir.

Şekil 8 - %0 - %10 Mutasyon Oranı Arasındaki Ortalama Değerler ile Standart Sapmalarla Elde Edilen Alt ve Üst Sınır Değerlerin Karşılaştırılması



5. SONUÇ

Bu çalışmada, paralel makinaların GA ile çizelgelenmesindeki mutasyon oranının etkinliği araştırılmıştır. Kocamaz ve diğerleri (2009) tarafından geliştirilen gen kodlama sistemi temel alınarak farklı mutasyon oranlarında çalıştırılmıştır. Bu kodlama sistemi için paralel makinaların genetik algoritma ile çizelgelenmesinde mutasyon oranının etkinliği %5 - %9 oranları arasında en iyi olduğu saptanmıştır. Tablo 5’de bu aralıktaki sonuçlar gösterilmektedir. Bu aralıktaki mutasyon oranları dikkate alınarak çalıştırılan modeller daha başarılı sonuçlar vermektedir.

Tablo 5 - %5 - %10 Mutasyon Oranı Arasındaki Sonuçlar

	5	6	7	8	9	10
Ortalama	1035,53	1042,42	1041,47	1040,35	1035,9	1044,56
En Büyük Değer	1075	1085	1095	1085	1065	1070
En Küçük Değer	1010	1010	1018	1000	1001	1010
Standart Sapma	18,48	17,47	16,24	18,28	13,85	16,56

KAYNAKÇA

Borovska P. 2006. 'Solving the Travelling Salesman Problem in Parallel by Genetic Algorithm on Multicomputer Cluster'. International Conference on Computer Systems and Technologies - CompSysTech'06, 15-16 June 2006, University of Veliko Tarnovo, Bulgaria

Gen, M. ve Cheng, R. (1997): Genetic Algorithms and Engineering Design, Wiley-IEEE.

Goldberg D.E. (1989), Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley, USA.

Hamilton, W.D. ve Ridley, M. (2005): Narrow roads of gene land: the collected papers of W.D. Hamilton, Oxford University Press.

Kocamaz, M., Çiçekli, U.G. ve Soyuer, H. (2009): "A Developed Encoding Method for Parallel Machine Scheduling with Permutation Genetic Algorithm", European and Mediterranean Conference on Information Systems EMCIS 2009, Crowne Plaza Hotel, Izmir, Brunel University and Dokuz Eylül University, 13 - 14 July 2009.

Mitchell, M. (1998): An Introduction to Genetic Algorithms, MIT Press.

MORI M. ve TSENG C.C. (1997): "Theory and Methodology: A genetic algorithm for multi-mode resource constrained project scheduling problem", European Journal of Operational Research, 100: 134-141

Pinedo, M. (2008): Scheduling: Theory, Algorithms, and Systems, Edition: 3, Springer.

Ruiz, R. Ve Maroto, C. (2006): "A genetic algorithm for hybrid flowshops with sequence dependent setup times and machine eligibility", European Journal of Operational Research, 169: 781-800

Sagegheih, A. (2006): "Scheduling problem using genetic algorithm, simulated annealing and the effects of parameter values on GA performance", Applied Mathematical Modelling, 30: 147-154.

Sheikh, K. (2003): Manufacturing Resource Planning (MRP II) With an Introduction to ERP, SCM, and CRM, McGraw-Hill.

WANG Y.Z. (2003): "Using genetic algorithm methods to solve course scheduling problems" Expert Systems with Applications, 25: 39-50

YEO M. F. and Agyei E. O. (1998): "Optimising engineering problems using genetic algorithms", Engineering Computations, 15(2): 268-280.

Murat KOCAMAZ ve Ural Gökay ÇİÇEKLİ

Zhenbo W. and Wenxun X. 2006. 'Parallel Machine Scheduling with Special İşs', Tsinghua Science and Technology, 11(1): 107-110.