GAZİ ÜNİVERSİTESİ MÜHENDİSLİK FAKÜLTESİ BİLGİSAYAR MÜHENDİSLİĞİ

yazı tipi, logo, simge, sembol, ticari marka içeren bir resim

Açıklama otomatik olarak oluşturuldu

BM469 GENETİK ALGORİTMALAR PROJESİ

Asım Tuzla -191180080

2023

**İçindekiler**

[1. Özet 2](#_Toc156136690)

[2. Giriş 2](#_Toc156136691)

[3. Yöntem 4](#_Toc156136692)

[3.1. Birey Temsili 4](#_Toc156136693)

[3.2. Başlangıç Popülasyonu 4](#_Toc156136694)

[3.3. Seçim Yöntemi 4](#_Toc156136695)

[3.4. Çaprazlama 5](#_Toc156136696)

[3.5. Mutasyon 6](#_Toc156136697)

[3.6. Maliyet hesaplama 6](#_Toc156136698)

[4. Sonuç 7](#_Toc156136699)

[5. Kaynaklar 10](#_Toc156136700)

# Özet

Bu rapor, "Opposition Based Genetic Optimization Algorithm with Cauchy Mutation" adlı makalenin incelenmesini ve bu genetik algoritmanın iş atölyesi (job shop) zamanlama probleminin çözümü üzerindeki uygulamalarının test edilmesini içermektedir.

Makalede ele alınan genetik algoritma, karşıt (opposition) tabanlı genetik optimizasyon algoritmasıdır. Bu algoritmanın temel amacı, genetik çaprazlama ve mutasyon operatörlerini kullanarak, iş atölyesi zamanlama problemi gibi karmaşık optimizasyon problemlerini çözmektir. Ayrıca, Cauchy mutasyon operatörü, bireylerin genetik çeşitliliğini artırmak ve algoritmanın lokal minimumlara sıkışmasını engellemek amacıyla kullanılmaktadır.

Rapor, makalenin yöntemlerinin etkinliğini test etmek için gerçekleştirilen deneysel çalışmalara odaklanmaktadır. İş atölyesi zamanlama problemi için çeşitli örnekler üzerinde algoritmanın performansı incelenmiş ve elde edilen sonuçlar değerlendirilmiştir. Analizler, önerilen genetik algoritmanın, Cauchy mutasyonu ile birleştirildiğinde, iş atölyesi zamanlama probleminde etkili bir çözüm elde etme potansiyeline sahip olduğunu göstermektedir.

# Giriş

İş atölyesi zamanlama problemi (Job Shop Scheduling Problem), bir üretim ortamındaki işlerin ve makinelerin belirli sıralamalara göre düzenlenmesi gereken bir tür üretim planlama problemini temsil eder. Temel hedef, belirli kısıtlar altında işleri optimize etmektir. Bu problem genellikle imalat endüstrisinde karşılaşılan ve çeşitli makinelerde işlenmesi gereken bir dizi görevin belirli bir sıraya göre planlanması gereken durumları modellemek için kullanılır.

İş atölyesi zamanlama probleminin temel unsurları şunlardır:

İşler (Jobs): Belirli bir süre içinde tamamlanması gereken görevler veya iş parçalarıdır.

Makineler (Machines): İşlerin tamamlanması için kullanılacak makineleri temsil eder.

İş Süreçleri (Operations): Her bir iş, bir veya daha fazla işlemi içerir. Her işlem, bir makinede belirli bir sürede gerçekleştirilir.

İşlem Sırası (Operation Sequence): Her bir işin makinelerde işlem göreceği sıra.

Temel amaç, işlerin tamamlanma süresini minimize etmek veya belirli bir kısıtlamaya göre optimize etmek için işleri ve işlemleri uygun bir şekilde sıralamaktır. Bu optimizasyon, genellikle toplam iş tamamlanma süresini minimize ederek veya belirli bir kısıtlama (örneğin, makine kullanımı, süre sınırlamaları) altında en iyi çözümü elde etmeyi amaçlar.

İş atölyesi zamanlama problemi, kombinatoryal bir optimizasyon problemi olarak kabul edilir ve çözümü zorlu olabilir, çünkü birçok olası sıralama kombinasyonu vardır ve bu kombinasyonları değerlendirmek karmaşık bir süreçtir. Bu tür problemleri çözmek için genetik algoritmalar gibi heuristik optimizasyon yöntemleri kullanılabilir.

İş atölyesi zamanlama problemi gibi karmaşık ve kombinatoryal optimizasyon problemlerini çözmek için çeşitli yaklaşımlar kullanılmaktadır. İşte bu tür problemleri çözmek için yaygın olarak kullanılan bazı yöntemler:

Genetik Algoritmalar (Genetic Algorithms): Genetik algoritmalar, evrimsel hesaplama temelli bir optimizasyon tekniğidir. İş atölyesi zamanlama problemi için genetik algoritmalar, bireylerin genetik materyalinin çaprazlanması ve mutasyona uğraması yoluyla popülasyonu evrimleştirir. Bu şekilde, daha iyi çözümlere doğru evrimleşen bir popülasyon elde edilir.

Simülasyon Tabanlı Optimizasyon (Simulation-Based Optimization): İş atölyesi zamanlama probleminin çözümü için sistem dinamiklerini modellenen ve simüle edilen bir yaklaşımdır. Bu yöntemde, farklı sıralama stratejileri simülasyonlarla değerlendirilir ve en iyi performans gösteren sıralama stratejisi seçilir.

Tabu Arama (Tabu Search): Tabu arama, bir çözüm uzayındaki çeşitli noktalara iteratif olarak ilerleyen bir metaheuristik algoritmadır. Kısıtlamalar ve tabu listeleri kullanarak çözüm uzayını araştırır ve en iyi çözümü bulmaya çalışır.

Yerel Arama Algoritmaları (Local Search Algorithms): İş atölyesi zamanlama problemleri için kullanılan yerel arama algoritmaları, başlangıç çözümünü iyileştirmek amacıyla mevcut çözüm uzayını araştırır. Bu algoritmalar, mevcut çözümleri iteratif olarak optimize eder.

İki Aşamalı Optimizasyon (Two-Stage Optimization): Bu yaklaşım, önce işlerin sırasını belirleyen bir aşama ve ardından bu sıraya göre makine zamanlarını optimize eden bir aşama içerir. İki aşamalı optimizasyon, problemi daha küçük parçalara böler ve her parçayı ayrı ayrı ele alarak çözüme ulaşmayı amaçlar.

Bu yaklaşımların her biri, problem özelliklerine ve gereksinimlerine bağlı olarak farklı avantajlar sağlar. Problemin büyüklüğü, karmaşıklığı ve kullanılabilir kaynaklar gibi faktörler, hangi yöntemin daha etkili olacağını belirlemede önemli rol oynar.

Makalede genetik algoritma yöntemi kullanılmış olup geleneksel genetik algoritmalara ek olarak farklı çözümler sunulmuştur. Genetik algoritmaların temel sorunlarından olan erken yakınsama sorunun çözmek ve popülasyon çeşitliliğini arttırmak için farklı çözümler sunulmuştur.

# Yöntem

İncelenen makalede temel genetik algoritma sorunlarına çözümler sunulmuştur ve bu genetik algoritma adımlarında izlenen ve uygulanan yöntemler aşağıdaki gibidir;

## Birey Temsili

Genetik algoritmalar, evrimsel hesaplama yaklaşımını temel alarak, doğal seçilim ve genetik varyasyon prensiplerini kullanarak problemleri optimize etmeye yönelik bir heuristik optimizasyon yöntemidir. Bu algoritmaların temel öğelerinden biri de "popülasyon"dur. Popülasyon, bir çözüm uzayındaki potansiyel çözümleri temsil eden bireylerin topluluğunu ifade eder. Permütasyonel temsil JSSP için çoğunlukla kullanılan bir temsil olup bu makalede de kullanılmıştır.

Permütasyonel popülasyon temsili, genetik algoritmaların bir popülasyon içindeki bireyleri, genetik materyalinin permütasyonlarını temsil eden bir yapı kullanarak ifade ettiği bir yöntemdir. Bu tip temsil, özellikle sıralama veya düzenleme problemleri gibi durumlarda kullanılır. İş atölyesi zamanlama problemi, bir permütasyonel popülasyon temsilini gerektiren bir örnektir.

## Başlangıç Popülasyonu

Opposition based yaklaşım, bir popülasyonun oluşturulmasında zıt bireylerin kullanıldığı bir genetik algoritma stratejisidir. Bu yaklaşım, popülasyon içindeki çeşitliliği artırmak ve potansiyel olarak daha geniş bir çözüm uzayını keşfetmek amacıyla kullanılır. Zıt birey, belirli bir bireyin genetik materyalinin her elemanının tersini içerir. Örneğin, bir iş atölyesi zamanlama problemi düşünelim; eğer bir bireyde iş sıralaması [1, 2, 3, 4] ise, zıt birey bu sıralamanın tersi olan [4, 3, 2, 1] olacaktır.

## Seçim Yöntemi

Rulet Tekerleği Seçim Yöntemi (Roulette Wheel Selection), genetik algoritmaların popülasyon içinden bireyleri seçmek için kullanılan bir olasılıklı seçim yöntemidir. Bu yöntem, bireylerin seçilme olasılıklarını bireylerin uygunluk değerlerine dayanarak belirler ve daha uygun bireylerin seçilme olasılığını artırır.

İşleyiş şu adımları içerir:

Uygunluk Değerlerinin Hesaplanması: Her birey için uygunluk değerleri belirlenir. Uygunluk değeri, bireyin problemde ne kadar iyi bir çözüm sunduğunu ölçen bir metriktir. Genellikle, uygunluk değerleri daha yüksek olan bireyler, problemin daha iyi çözümlerini temsil eder.

Olasılık Dağılımının Oluşturulması: Her bireyin seçilme olasılığı, uygunluk değeri ile doğru orantılıdır. Uygunluk değeri daha yüksek olan bireyler, seçilme olasılığı daha yüksek olan bir aralığa sahip olacaktır. Rulet tekerleği gibi düşünülebilecek bir tekerlek oluşturulur, bu tekerleğin her dilimi bir bireyi temsil eder ve dilimlerin büyüklükleri seçilme olasılıklarıyla orantılıdır.

Rulet Tekerleğinin Çevrilmesi: Rulet tekerleği çevrilir ve rastgele bir durakta durulur. Bu durak, seçilecek bireyi belirler.

Seçilen Bireyin Yeni Popülasyona Eklenmesi: Rulet tekerleği üzerinde durulan birey, yeni oluşturulan popülasyona eklenir. Bu işlem popülasyon boyutu kadar tekrarlanır, böylece yeni popülasyon oluşturulur.

Rulet Tekerleği Seçim Yöntemi, daha iyi uygunluk değerine sahip bireylere daha yüksek bir seçilme olasılığı tanıdığı için genetik algoritmanın konverjansını hızlandırabilir ve çeşitliliği artırabilir. Ancak, uygunluk değerleri arasındaki büyük farklar durumunda daha popüler olan bireylerin seçilme olasılığı çok yüksek olabilir, bu da konvergansı etkileyebilir.

## Çaprazlama

Two-Point Crossover (İki Nokta Çaprazlama), genetik algoritmaların popülasyon içindeki bireyleri çaprazlamak için kullanılan bir yöntemdir. Bu çaprazlama operatörü, iki ebeveyn birey arasında belirli iki noktadan kesilerek çocuk bireylerin oluşturulmasına dayanır. İşleyiş şu adımları içerir:

İki Ebeveyn Birey Seçimi: Çaprazlama işlemi için yukarıda belirtilen seçim yöntemi olarak roulette wheele yöntemi kullanılarak çaprazlanması istenen bireyler popülasyon içerisinden seçilir.

Çaprazlama Noktalarının Belirlenmesi: İki ebeveyn birey arasında çaprazlama noktaları belirlenir. Bu noktalar, bireylerin genetik materyali üzerinde çaprazlanacak yerleri gösterir. Two-Point Crossover'da, genellikle rastgele seçilen iki nokta kullanılır.

Çocuk Bireylerin Oluşturulması: Seçilen iki çaprazlama noktası arasındaki genetik materyal, iki ebeveyn arasında değiştirilir. Bu değiştirme işlemi, iki ebeveyn arasındaki belirli bir bölgenin birbirleriyle yer değiştirmesi anlamına gelir. Böylece, iki çocuk birey oluşturulur.

Örnek:

Ebeveyn 1: [A, B, C, D, E, F, G, H]

Ebeveyn 2: [W, X, Y, Z, U, V, T, S]

Belirlenen çaprazlama noktaları: 3 ve 6

Çocuk 1: [A, B, C, Z, U, V, T, S]

Çocuk 2: [W, X, Y, D, E, F, G, H]

Bu örnekte, Ebeveyn 1 ve Ebeveyn 2 arasında belirtilen çaprazlama noktalarındaki genetik materyal yer değiştirmiştir, böylece iki çocuk birey oluşturulmuştur. Two-Point Crossover, genetik çeşitliliği artırarak ve yeni çözüm uzaylarını keşfetmeye yardımcı olarak genetik algoritmanın performansını iyileştirebilir.

## Mutasyon

Cauchy Mutasyonu, genetik algoritmaların popülasyon içindeki bireyleri mutasyona uğratmak için kullanılan bir mutasyon yöntemidir. Bu yöntem, bireylerin genetik materyalinin belirli bir dağılım prensibine göre rastgele değiştirilmesine dayanır. Cauchy dağılımı, genellikle genetik algoritmalar içinde çeşitli optimizasyon problemlerine uyarlanabilir ve özellikle çeşitlilik arttırma amacıyla kullanılır.

Cauchy mutasyonu için belirli parametreler seçilmelidir. En önemli parametrelerden biri, Cauchy dağılımının genişliğini kontrol eden ölçek parametresidir. Bu parametre, mutasyonun genetik materyali üzerindeki etkisini belirler. Seçilen bireyin genetik materyali, Cauchy dağılımı kullanılarak rastgele değiştirilir.

Cauchy mutasyonu, genetik algoritmaların lokal minimumlardan çıkma yeteneğini artırabilir ve popülasyon içindeki çeşitliliği artırarak daha geniş bir çözüm uzayını araştırmalarına yardımcı olabilir. Bu, genetik algoritmaların daha keşifçi ve global bir arama stratejisi benimsemelerine olanak tanır.

## Maliyet hesaplama

Maliyetin hesaplanması için FT10 problemi için tanımlı olan her iş için makinelerde tamamlanma sürelerinin bulunduğu bir matristen okunarak oluşturulan bireyin sıralamaları ile toplanarak hesaplanmaktadır.

M1 M2 M3 M4 M5 M6 M7 M8 M9 M10

J1 29 78 9 36 49 11 62 56 44 21

J2 43 90 75 11 69 28 46 46 72 30

J3 91 85 39 74 90 10 12 89 45 33

J4 81 95 71 99 9 52 85 98 22 43

J5 14 6 22 61 26 69 21 49 72 53

J6 84 2 52 95 48 72 47 65 6 25

J7 46 37 61 13 32 21 32 89 30 55

J8 31 86 46 74 32 88 19 48 36 79

J9 76 69 76 51 85 11 40 89 26 74

J10 85 13 61 7 64 76 47 52 90 45

Örnek:

[3,2,1,3,8,3] dizisini çözümün bir kısmını örnek olarak alacak olursak sırası ile işlerin gönderildiği makineleri okuyup ardından tablodan bakılarak maliyet hesaplanmalıdır.

İş 3, Makine 2: 85

İş 2, Makine 1: 43

İş 1, Makine 1: 29

İş 3, Makine 1: 91

İş 8, Makine 3: 46

İş 3, Makine 4: 74

Bu sıralamadaki toplam maliyet 368 olarak birim zaman olarak hesaplanır.

# Sonuç

Bu çalışmada, "Opposition Based Genetic Optimization Algorithm with Cauchy Mutation" başlıklı makalenin yöntemleri üzerine kapsamlı bir araştırma ve uygulama gerçekleştirilmiştir. İş atölyesi zamanlama problemlerini çözmek amacıyla önerilen genetik algoritma, özellikle başlangıç popülasyonunun oluşturulmasında opposition based yaklaşımı ve çaprazlamada two-point crossover, mutasyonda ise Cauchy mutasyonu yöntemlerini kullanmaktadır.

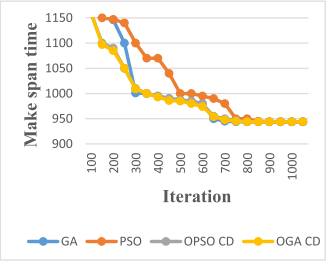
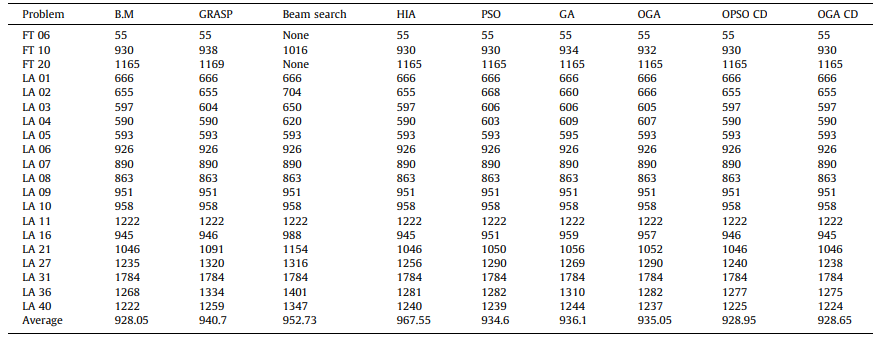
Çalışmanın başlangıcında, iş atölyesi zamanlama problemleri üzerine genetik algoritmaların uygulanabilirliği ve etkisi literatürde incelenmiş, daha sonra makalenin önerdiği yöntemlerin detaylı bir açıklaması yapılmıştır. Ardından, bu yöntemlerin Python programlama dilinde kodlanarak, farklı problem örnekleri üzerinde test edilmiştir.

Yapılan deneyler, opposition based yaklaşımının başlangıç popülasyonunu çeşitlendirmede ve Cauchy mutasyonunun çözüm uzayındaki çeşitliliği artırmada etkili olduğunu göstermiştir. Two-point crossover ise genetik algoritmanın çaprazlama operatörü olarak başarılı sonuçlar vermiş ve çözüm uzayında genetik materyalin daha etkili bir şekilde değiştirilmesine katkı sağlamıştır.

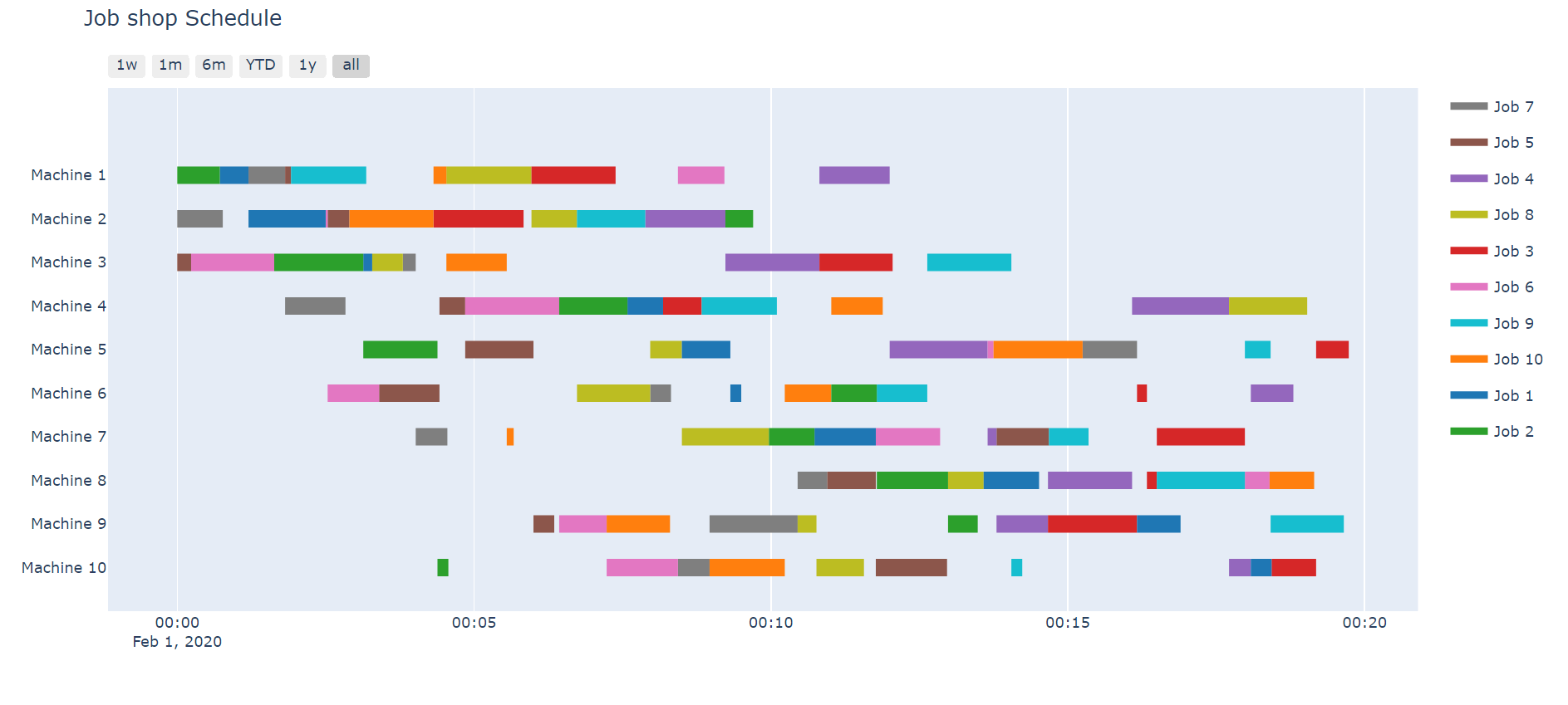
Bu çalışma, genetik algoritmaların iş atölyesi zamanlama problemlerine uygulanmasında önemli olan başlangıç popülasyonunun çeşitliliği, çaprazlama ve mutasyon operatörlerinin etkinliği üzerine değerli bir bilgi sunmaktadır. Ancak, her bir problem için en uygun parametre değerlerini belirlemek ve daha geniş bir problem sınıfında genelleme yapmak için daha fazla araştırma gereklidir.

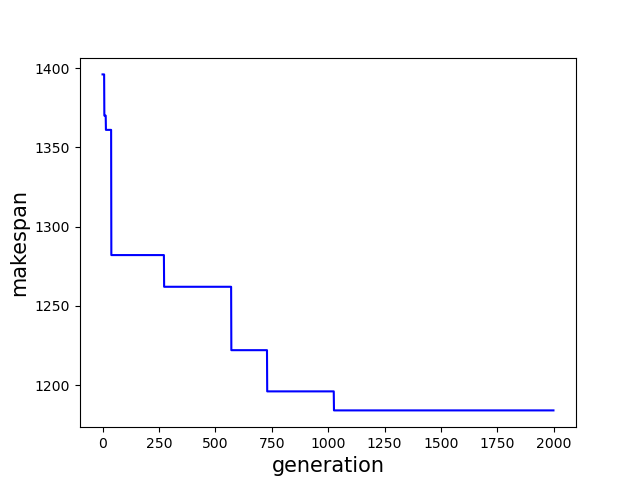
Sonuç olarak, bu çalışma, genetik algoritmaların iş atölyesi zamanlama problemleri üzerinde etkili bir optimizasyon aracı olarak kullanılabilirliğini göstermektedir. Elde edilen sonuçlar, genetik algoritmaların problem çözme yeteneklerini artırmak ve karmaşık üretim süreçlerinde verimliliği optimize etmek isteyen araştırmacılara ve endüstri uzmanlarına yararlı bir referans sunmaktadır.

Makalede verilen FT10 problemi için grafikler ve sonuçlar şu şekildedir;



Yapılan testler ve denemeler sonucunda makalede sunulan çözümlerin simüle edilerek elde edilen sonuçlar aşağıdaki gibidir;





# Kaynaklar

1. Kumar, K. A., & Dhas, E. R. (2023). Opposition based genetic optimization algorithm with Cauchy mutation for job shop scheduling problem. Materials Today: Proceedings, 72, 3006-3011.
2. Liu, S. C., Chen, Z. G., Zhan, Z. H., Jeon, S. W., Kwong, S., & Zhang, J. (2021). Many-objective job-shop scheduling: A multiple populations for multiple objectives-based genetic algorithm approach. IEEE Transactions on Cybernetics.
3. <http://jobshop.jjvh.nl/instance.php?instance_id=7>
4. Atay, Y., & Kodaz, H. (2014). Optimization of job shop scheduling problems using modified clonal selection algorithm. *Turkish Journal of Electrical Engineering and Computer Sciences*, *22*(6), 1528-1539.