

# BLM 4510 YAPAY ZEKA 1. ODEV RAPORU

**Dersin Yürütücüsü** : Doç.Dr.Mehmet Fatih AMASYALI

**Öğrenci Ad/Soyad** : Alaaddin Göktuğ AYAR

**Öğrenci No** : 19011603

**Ödev Konusu** : 9\*9 luk bir alanı X adet drone tarayacaktır. Drone'lar için alanın içinde ortak bir başlangıç ve bitiş konumu verilecektir. Hepsi verilen aynı noktadan başlayıp aynı yere dönecektir.Drone'ların olabildiğince maksimum alanı taraması hedeflenmektedir.



## Algoritmamin Anlatimi:

Genetik Algoritma kullanarak tamamladığım bu ödevde, 3 tane değerlendirme fonksiyonu kullandım. Bu fonksiyonlar sırayla:

F1 : En çok alan gezmesi, yani gezmediği hücre sayısının hesaplanıp daha sonra normalize edilmiş halinin tersini alarak fonksiyonu minimize etmeye çalıştım.

Bu fonksiyondan aldığımız gezilmediği alanın yüksek olduğu fonksiyonların tersi alındığında, seçilme ihtimalleri azalacağından. Çok sayıya kare gezen çözümlerin seçilme olasılığını arttıracak.

F2 : Her drone'un başlangıç noktasına yakınlığı oklid uzaklığı formülü ile hesaplanarak başlangıca en yakın olan ve başlangıca dönmesi muhtemel drone'ların seçilme ihtimalini arttırıyor

F3 : Drone'ların yaptığı hareketlerin olabildiğince az acı ile olmasını istediğimizden, az acı yapan drone'ların değerinin yüksek olduğu sonucu çıkartan ve bu çözümlerin seçilme ihtimalini arttıran fonksiyon

Bu 3 fonksiyonu kullanarak öncelikle , generation sayısı kadar bir ana loop etrafında. Her döngü için popülasyondaki bütün çözümlerin girilen drone sayısı ile orantılı bir şekilde bölünüp, her çözüm için fitness değerleri oluşturulur.

Oluşturulan bu değerler toplanarak genel bir fitness değeri elde edilir. Her generation için oluşan bu fitness değerlerden en çok uyan değer seçilerek jenerasyonun en iyi değerlerinin tutulduğu diziye atılır. Bu indisin kullandığı yol daha sonra grafik çiziminde kullanılmak üzere en iyi değerlerin yollarının tutulduğu diziye de atılır.

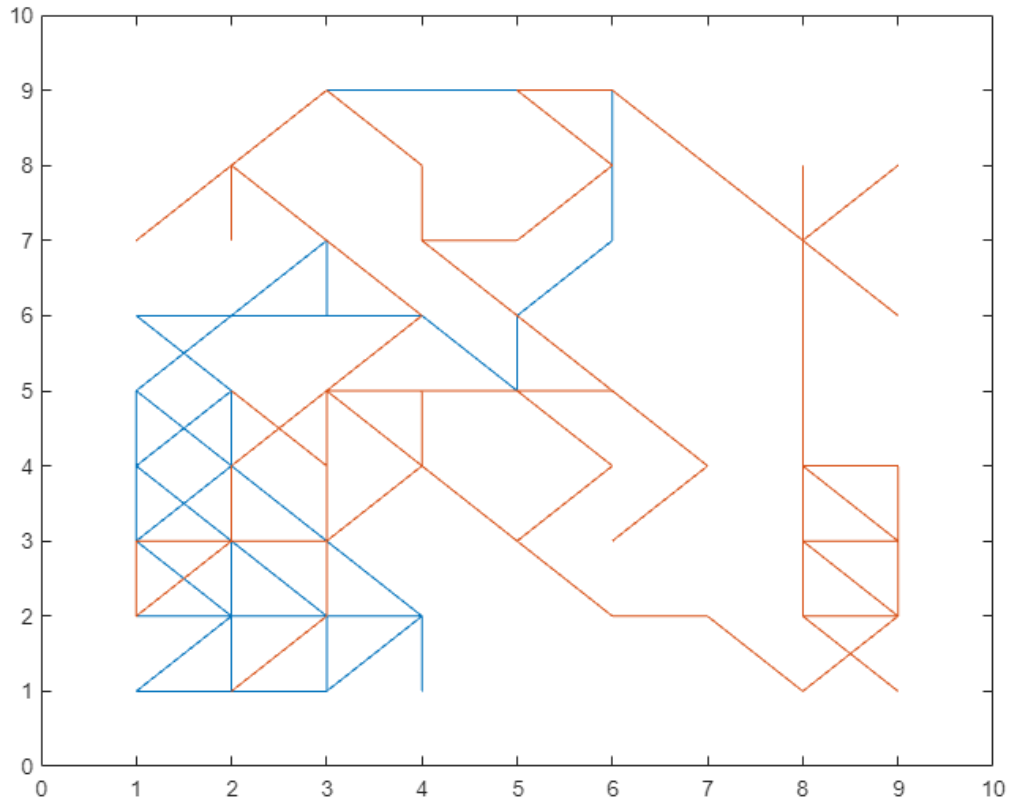
Rulet tekeri metodu kullanılarak bir sonraki jenerasyon için oluşturulacak bireylerin, şimdi bulunan bireylerden hangilerinin kullanılarak crossover yapılması gerektiği belirlenir ve ihtimali yüksek olan indisler seçilir. Her bir tur için en iyi population/2 sayısı kadar birey bir sonraki jenerasyona geçirilir.

Generation döngüsünün sonuçlanmasından sonra en iyileri tuttuğumuz dizi içerisinde en iyilerin en iyisi seçilir ve bu çözüm için grafikler çizilir.

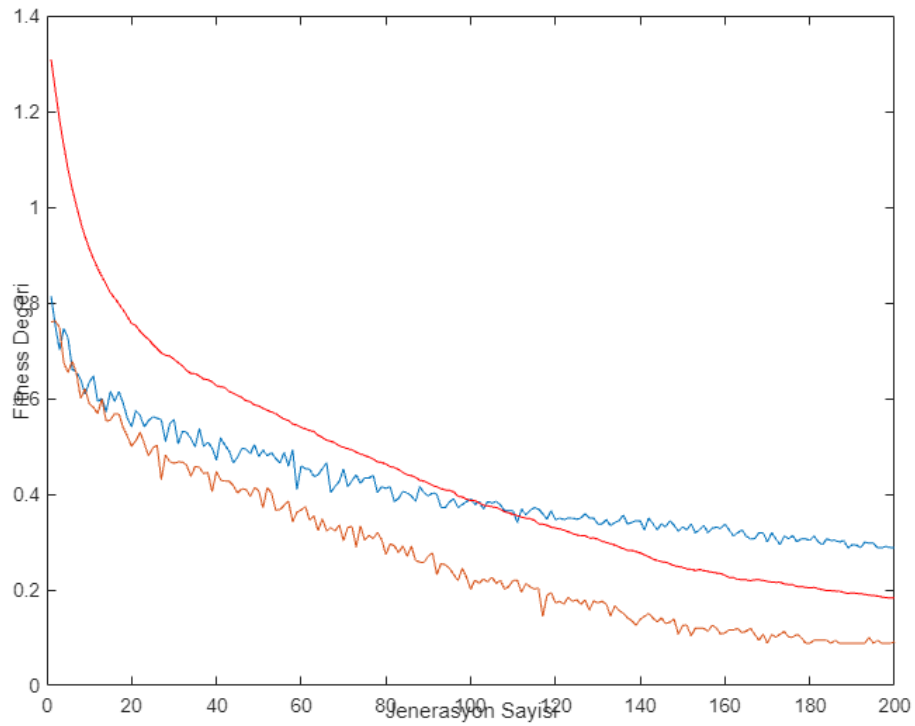
Sonuçların yorumlanabilmesi için ayrıca generation boyunca fitness fonksiyonunun değerinin nasıl bir yol izlediği görülür.



## 2 Drone'un izlediği yollar



Kırmızı ile gösterilen popülasyonun ortalama fitness değeri, mavi ve turuncu ile gösterilen 2 drone'un fitness değerleri.

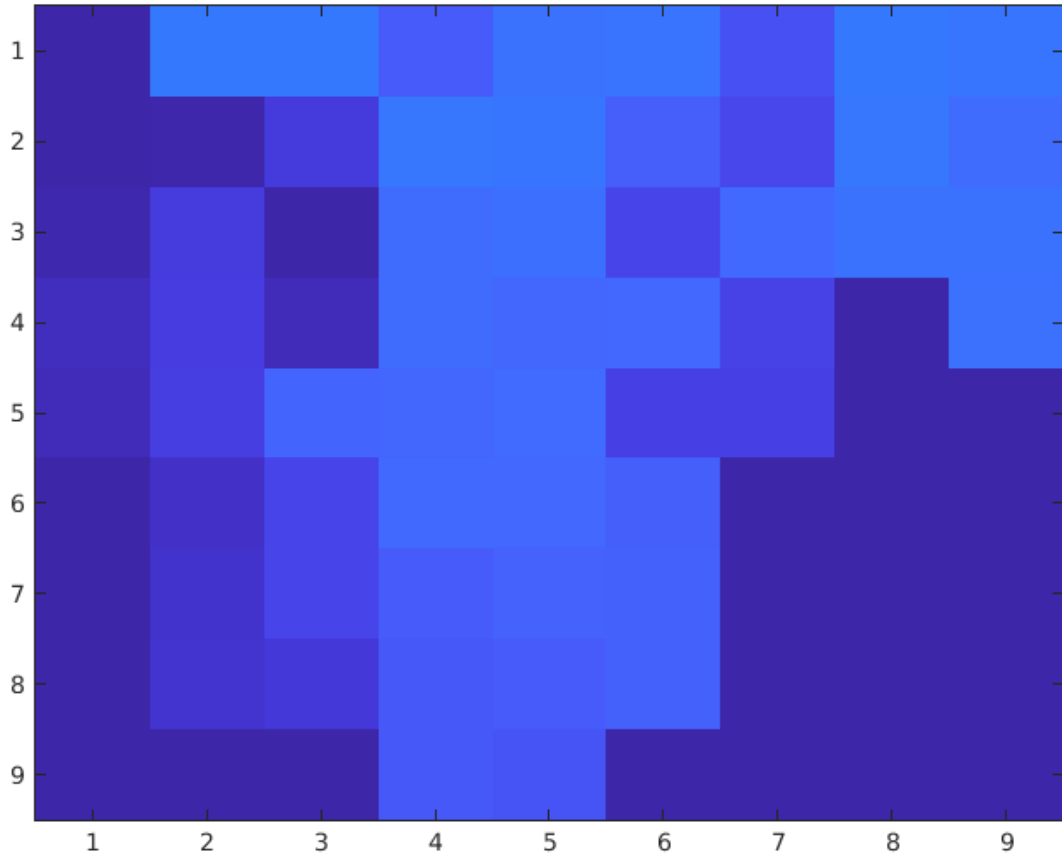


## Deneme 2

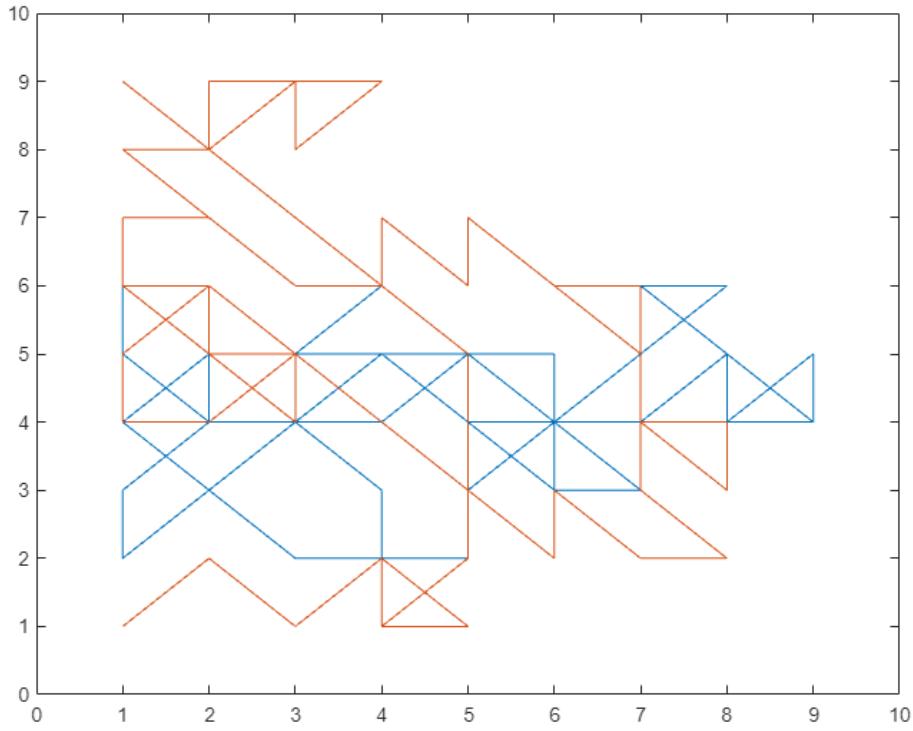
Population: 2000    Generation: 150    Mutation rate : 0.01

Crossover: 1    Number of Drones : 2    Starting Point : [1,1]

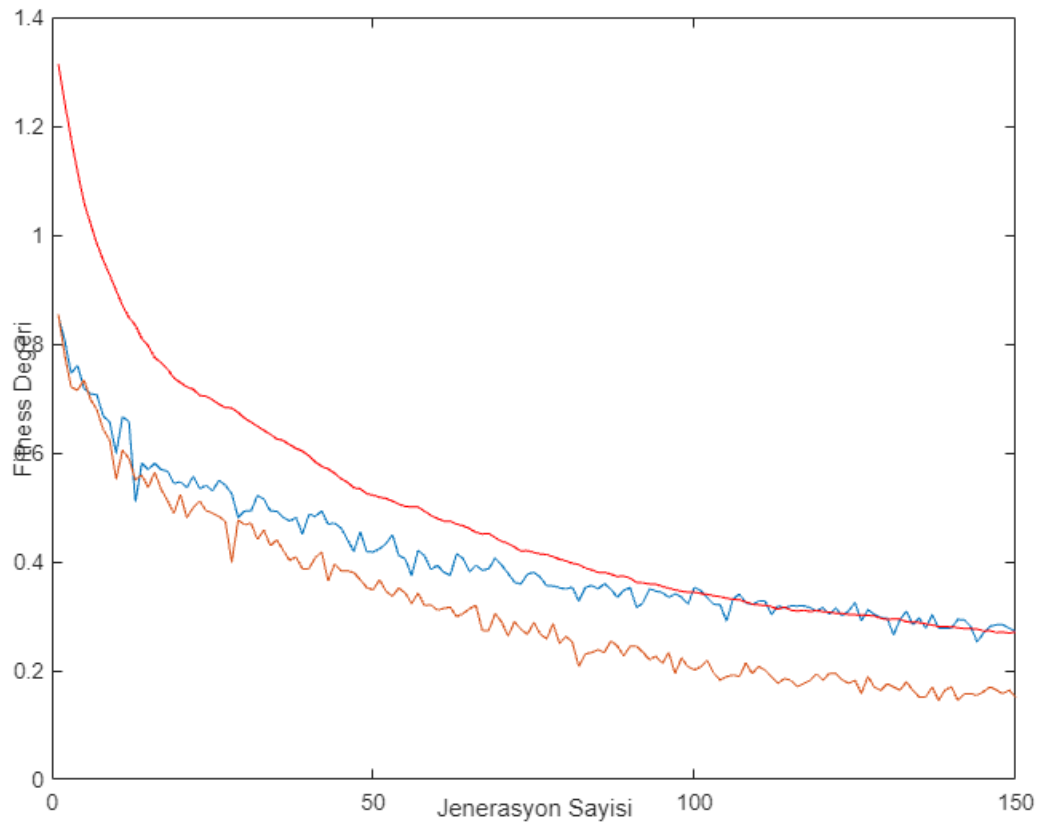
## Gezilen Noktalar



## 2 Drone'un izlediği yollar



Kırmızı ile gösterilen popülasyonun ortalama fitness değeri, mavi ve turuncu ile gösterilen 2 drone'un fitness değerleri.

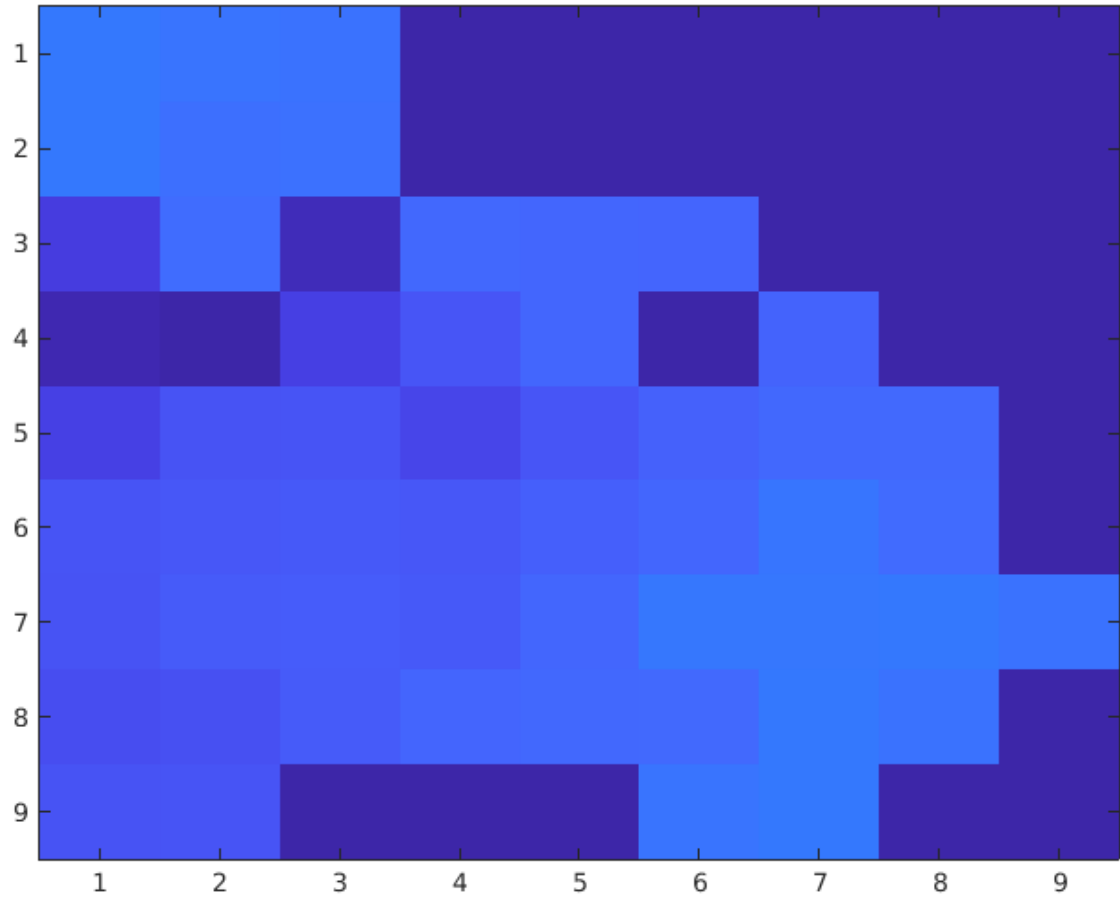


### Deneme 3

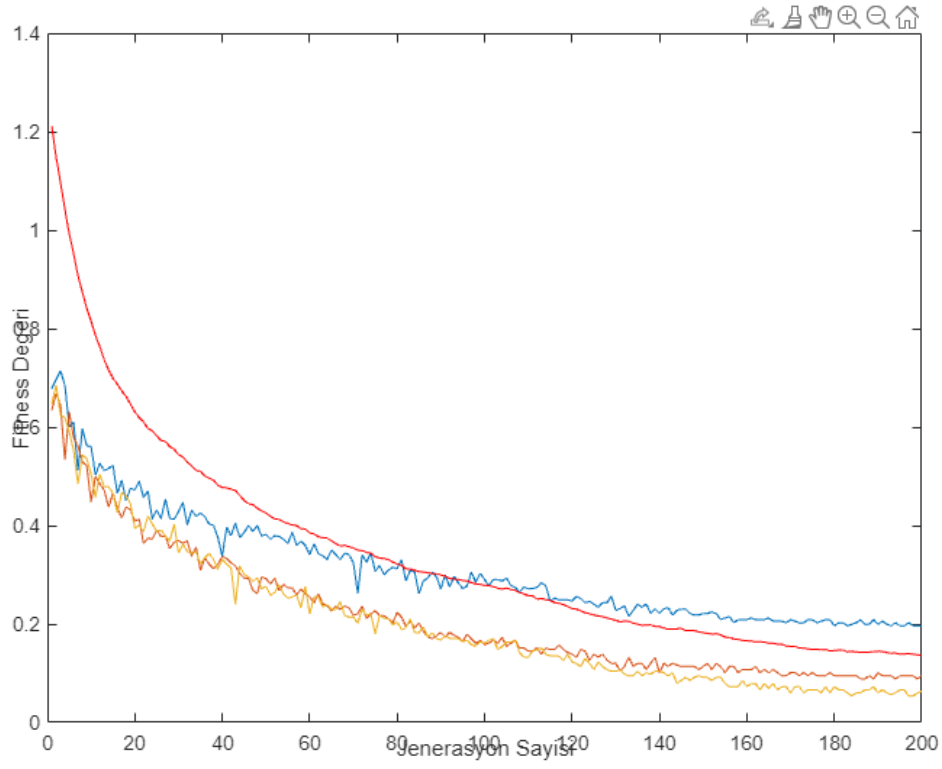
Population: 3000    Generation: 200    Mutation rate : 0.01

Crossover: 1    Number of Drones : 3    Starting Point : [1,1]

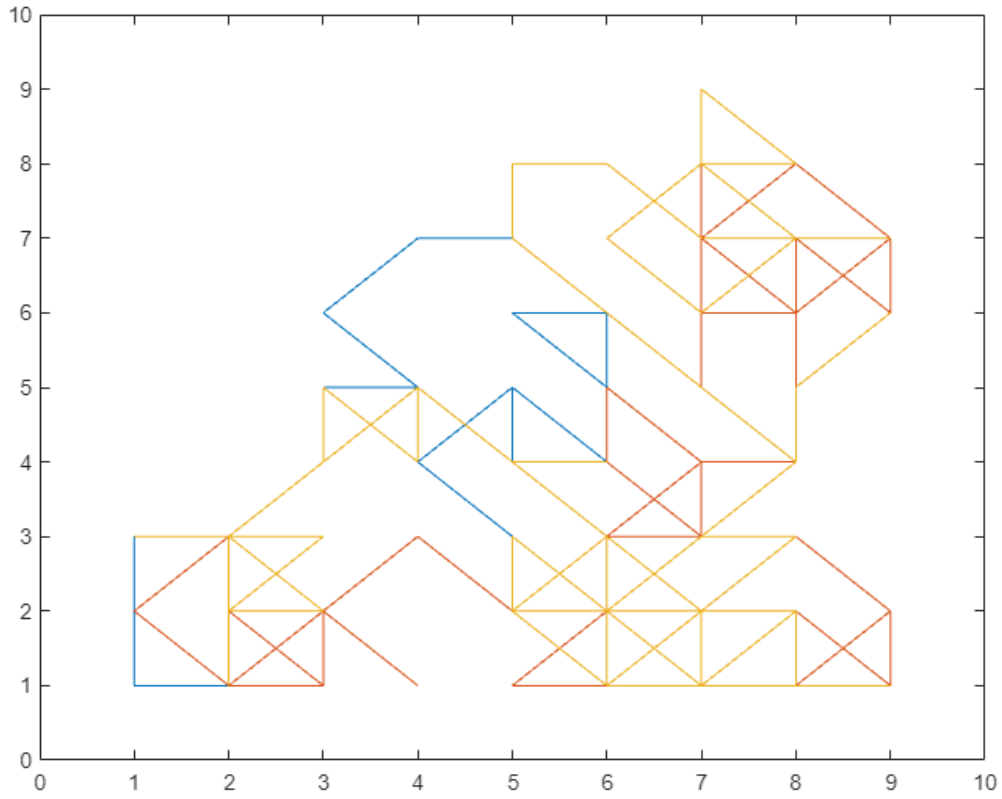
### Gezilen Noktalar



Kirmizi ile gösterilen popülasyonun ortalama fitness değeri, mavi,turuncu ve sari ile gösterilenler ise 3 drone'un fitness değerleri.



3 Drone'un izlediği yollar



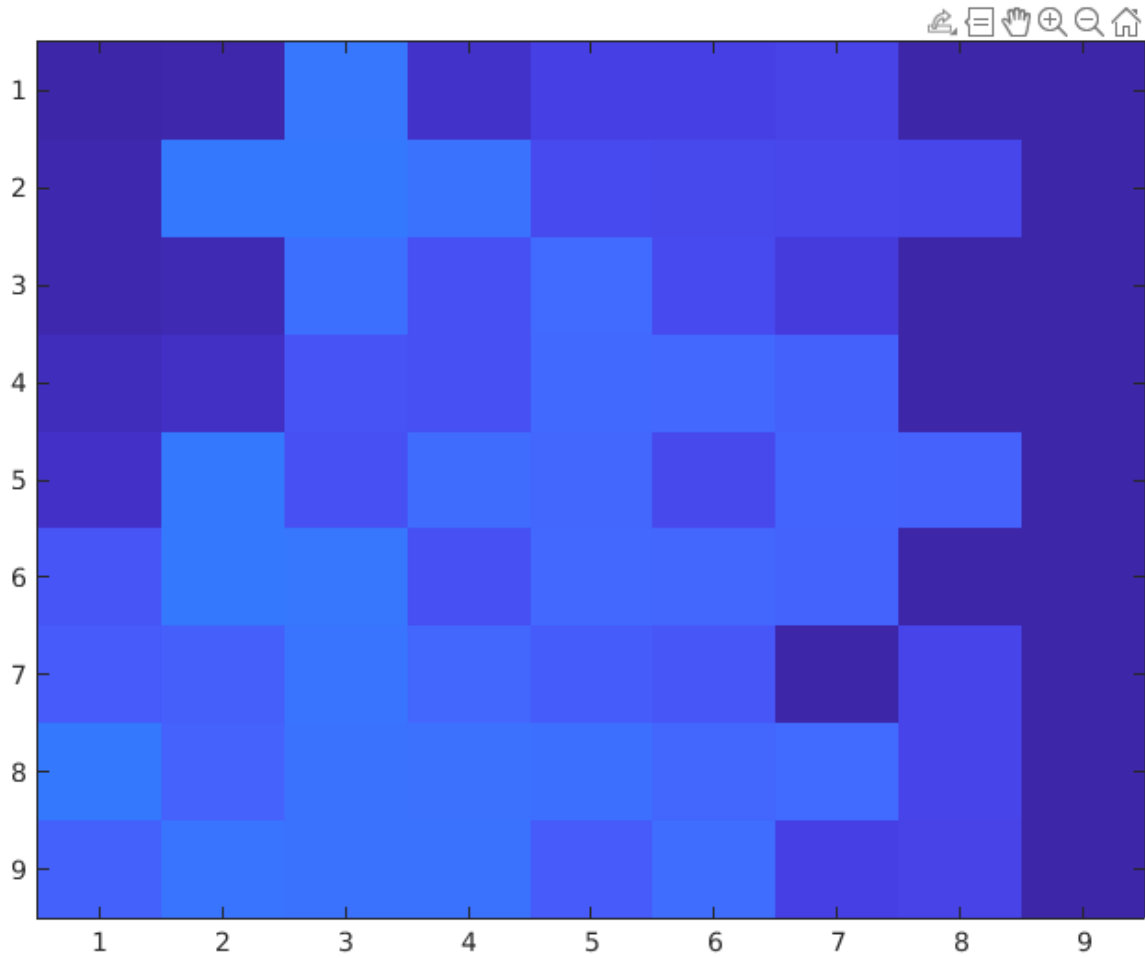


## Deneme 4

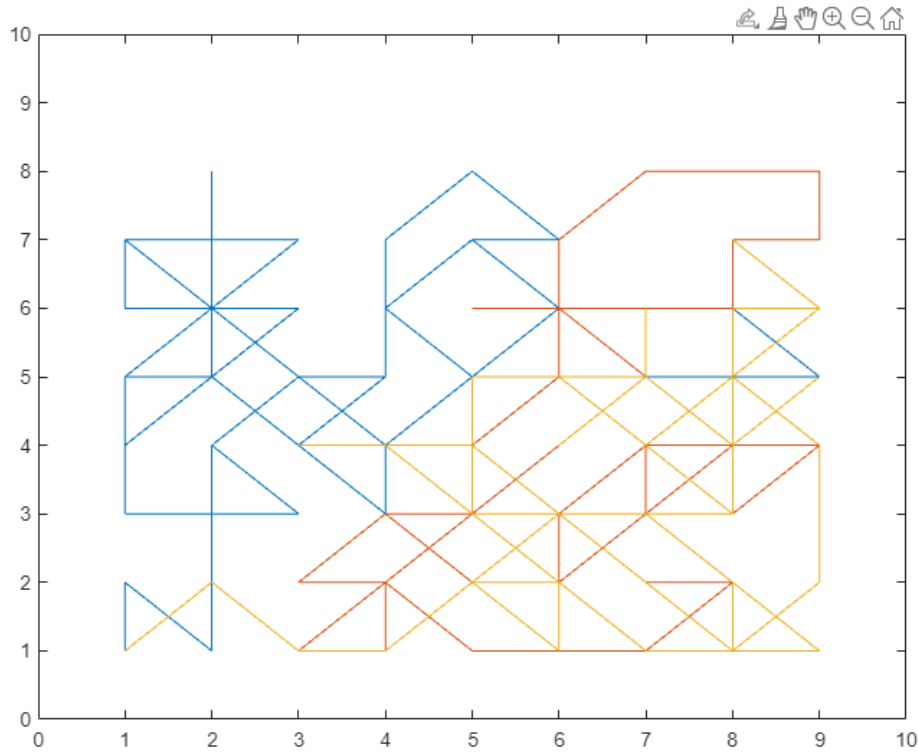
Population: 3000    Generation: 200    Mutation rate : 0.01

Crossover: 2      Number of Drones : 3      Starting Point : [1,1]

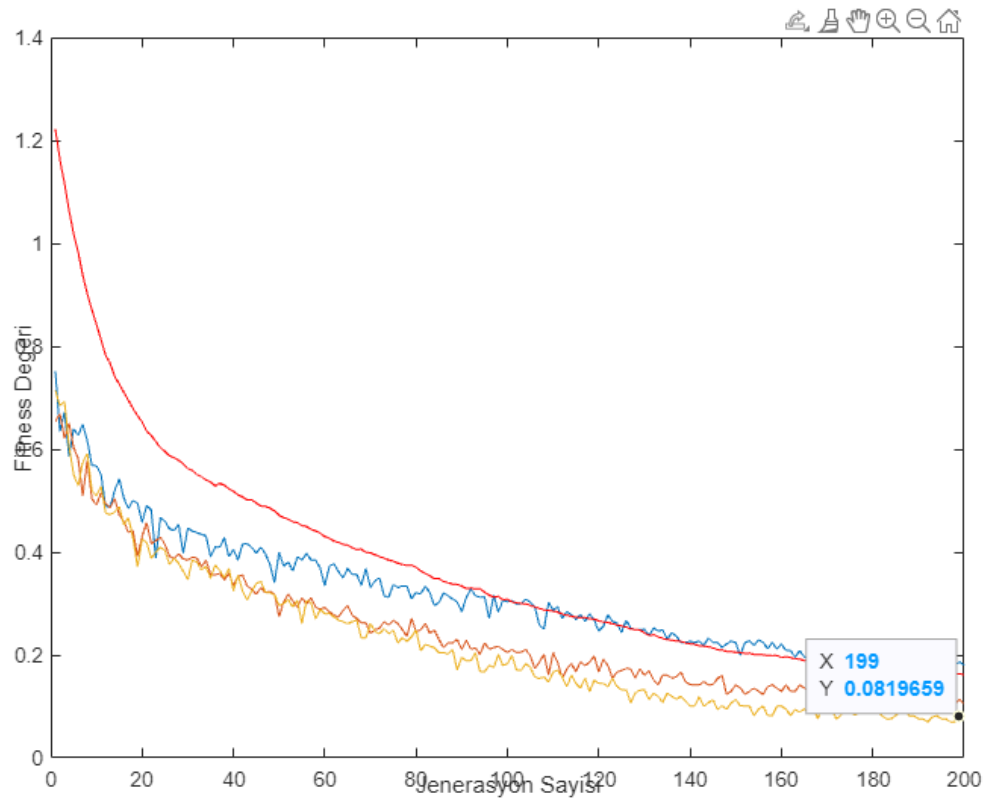
## Gezilen Noktalar



### 3 Drone'un izlediği yollar



Kırmızı ile gösterilen popülasyonun ortalama fitness değeri, mavi,turuncu ve sarı ile gösterilenler ise 3 drone'un fitness değerleri.

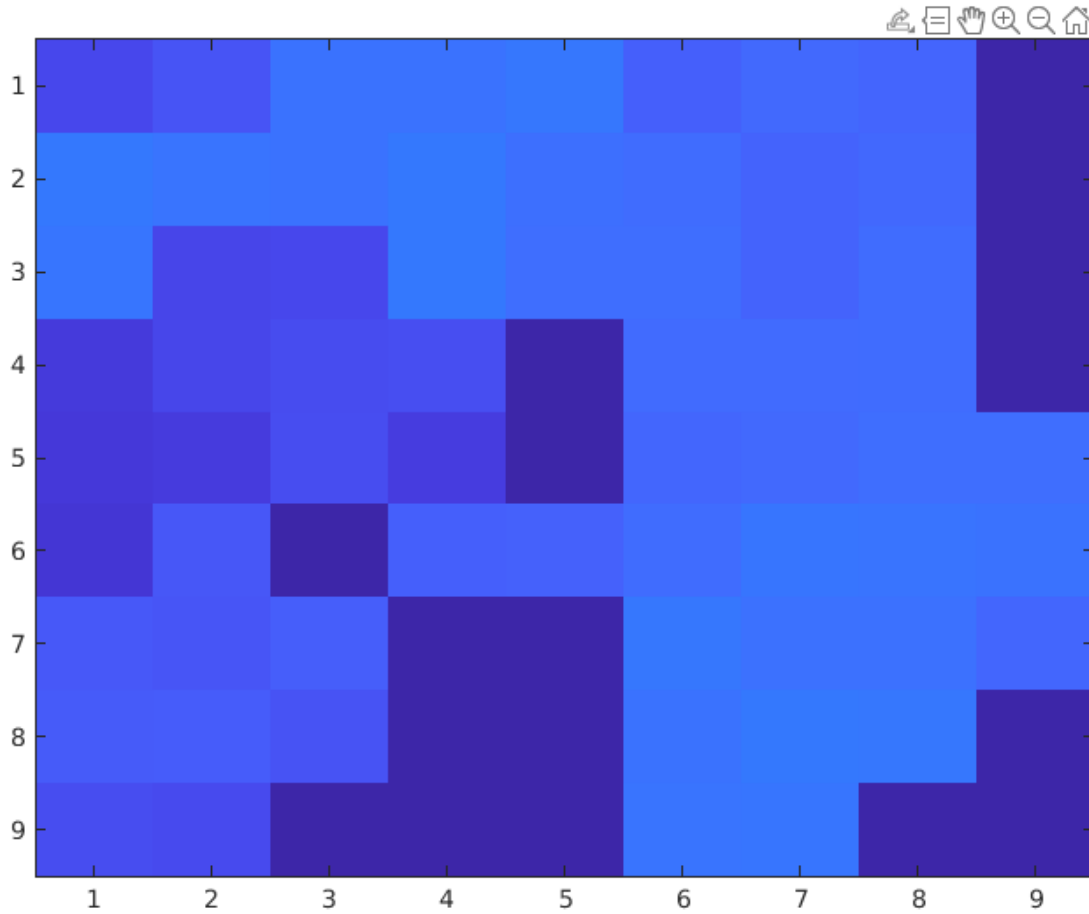


## Deneme 5

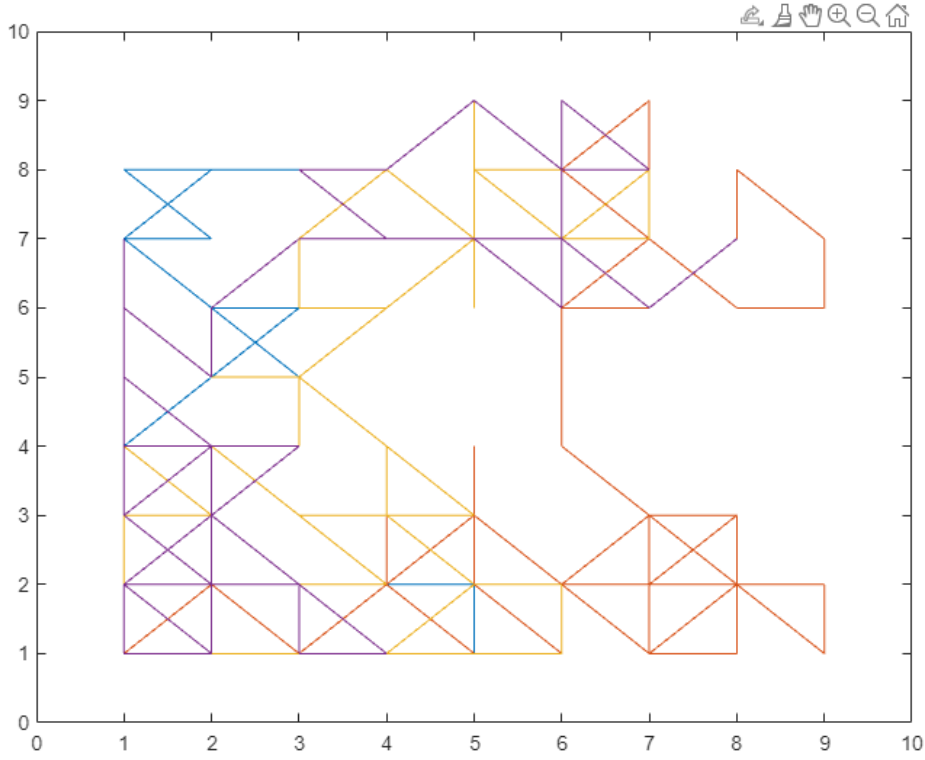
Population: 3000    Generation: 200    Mutation rate : 0.01

Crossover: 2    Number of Drones : 4    Starting Point : [1,1]

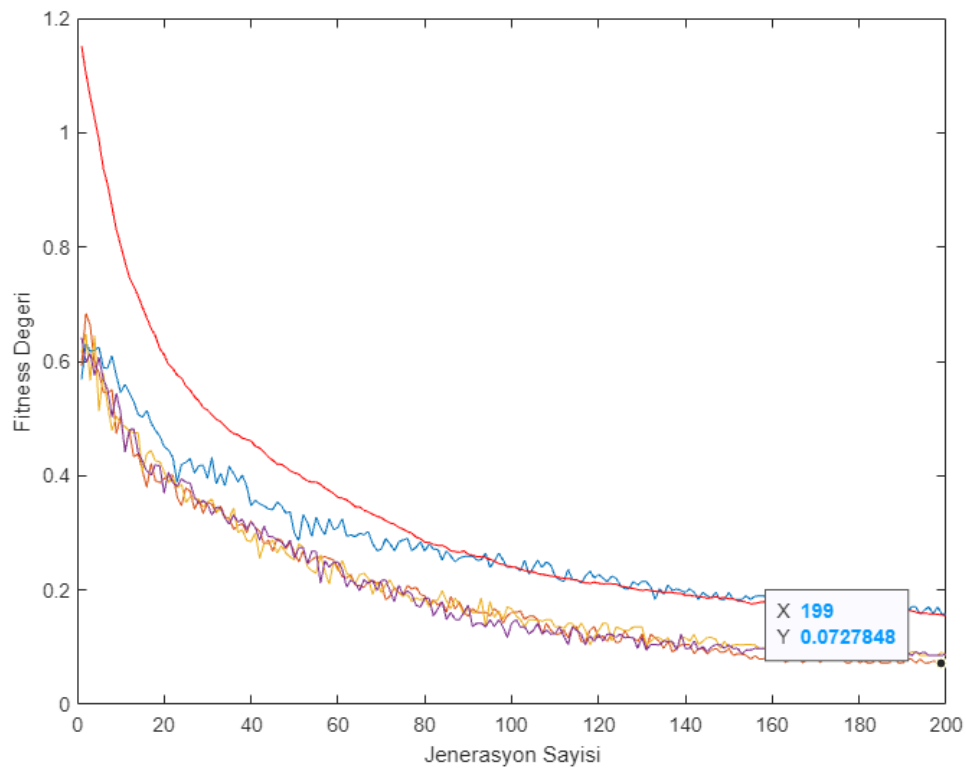
## Gezilen Noktalar



#### 4 Drone'un izlediği yollar



Kirmizi ile gösterilen popülasyonun ortalama fitness değeri, mavi,turuncu,mor ve sari ile gösterilenler ise 4 drone'un fitness değerleri.

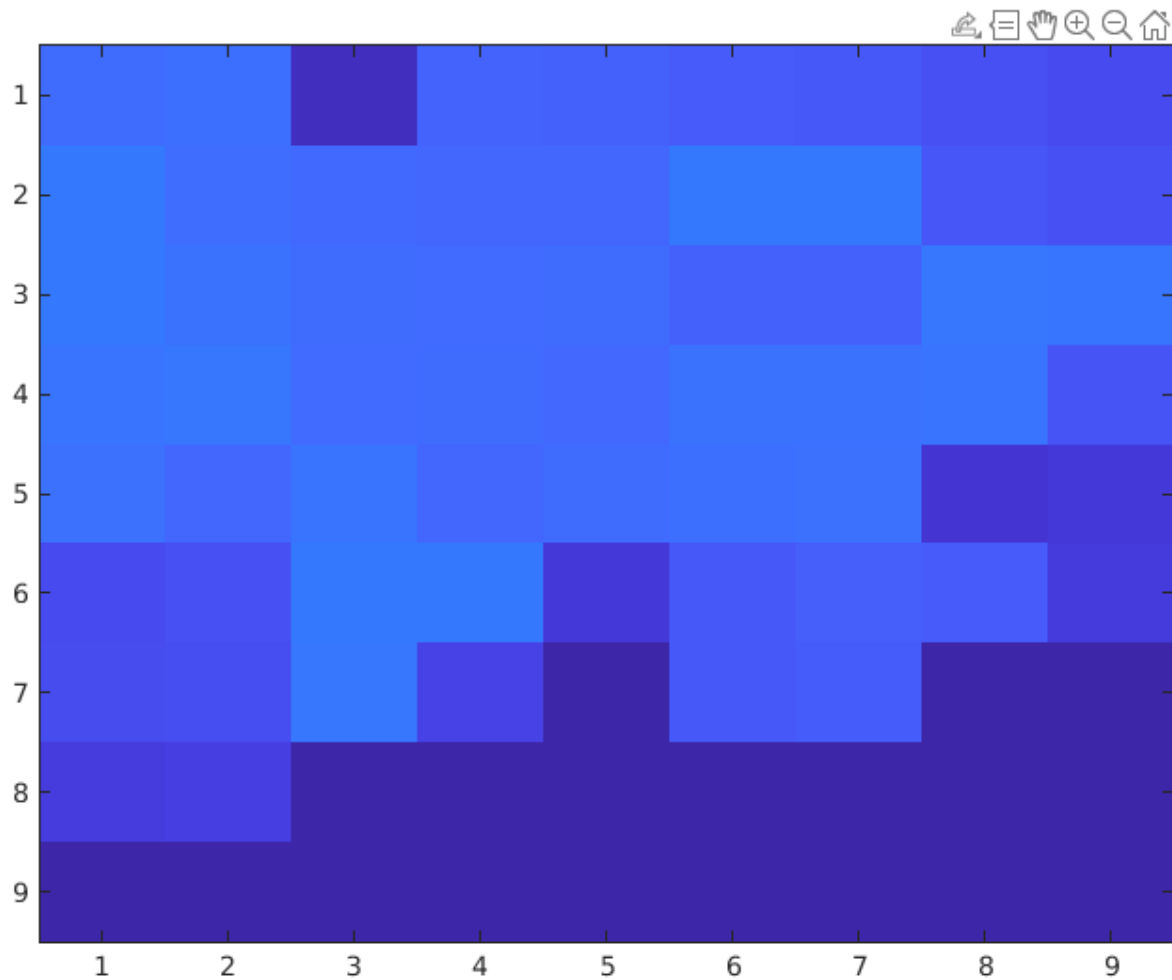


## Deneme 6

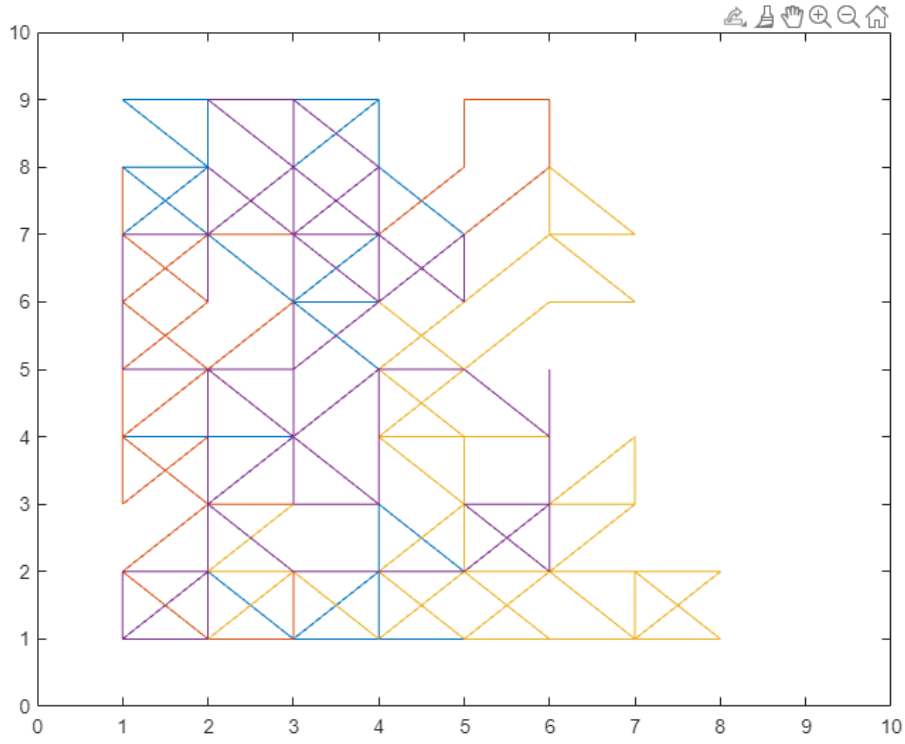
Population: 2000    Generation: 200    Mutation rate : 0.01

Crossover: 2      Number of Drones : 4      Starting Point : [1,1]

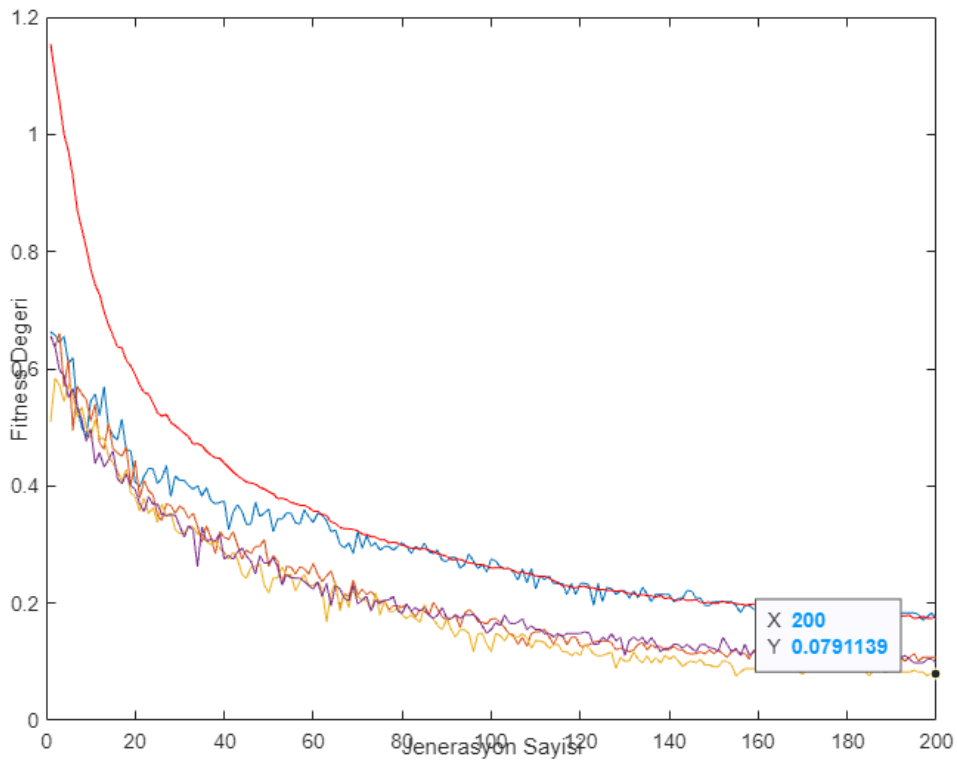
## Gezilen Noktalar



#### 4 Drone'un izlediği yollar



Kirmizi ile gösterilen popülasyonun ortalama fitness değeri, mavi,turuncu,mor ve sari ile gösterilenler ise 4 drone'un fitness değerleri.

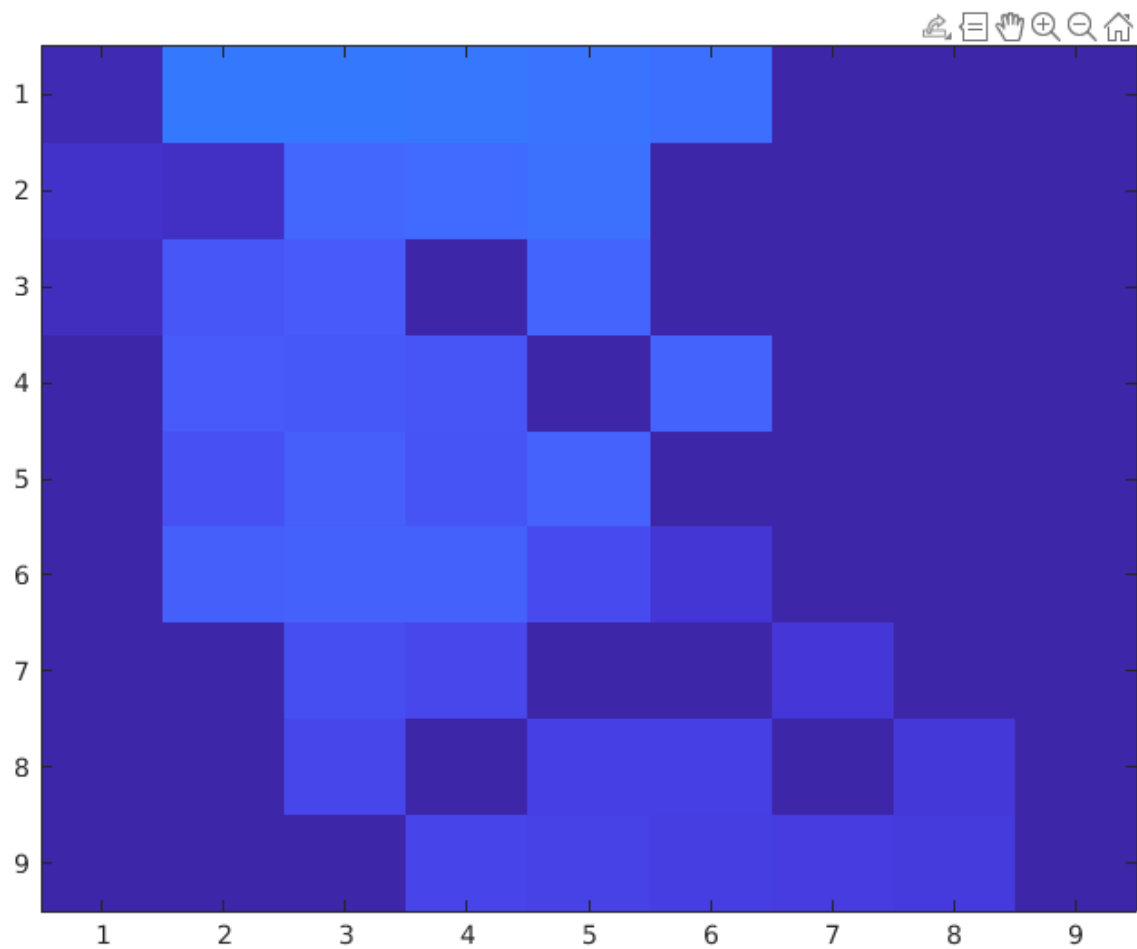


## Deneme 7

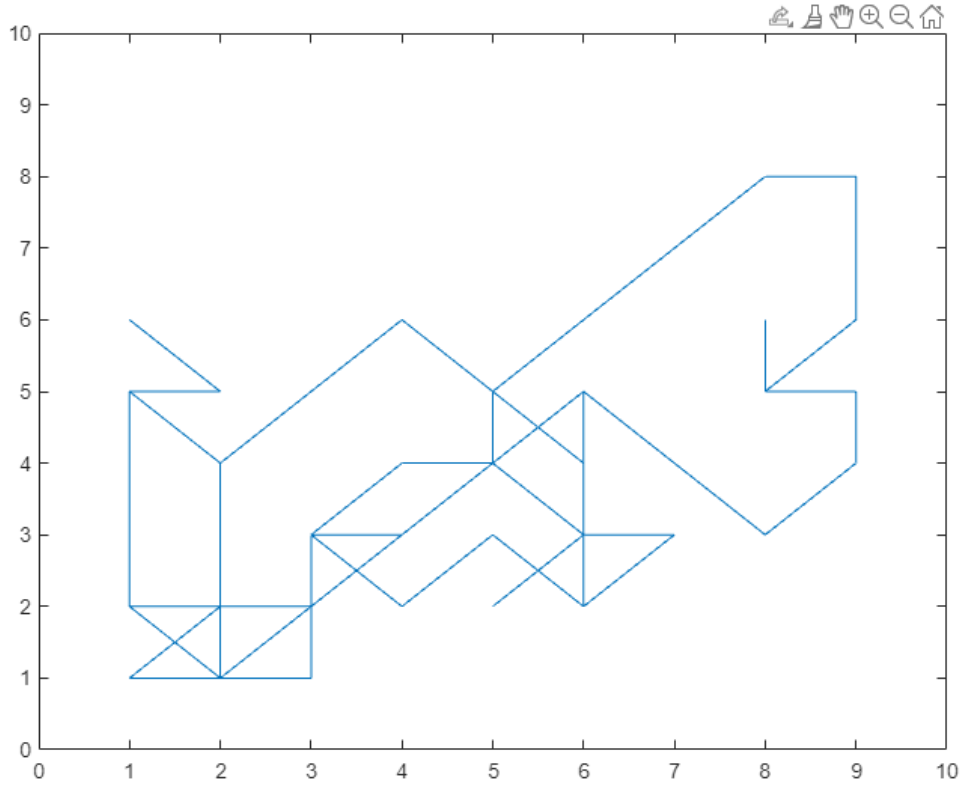
Population: 2000    Generation: 200    Mutation rate : 0.01

Crossover: 2      Number of Drones : 1      Starting Point : [1,1]

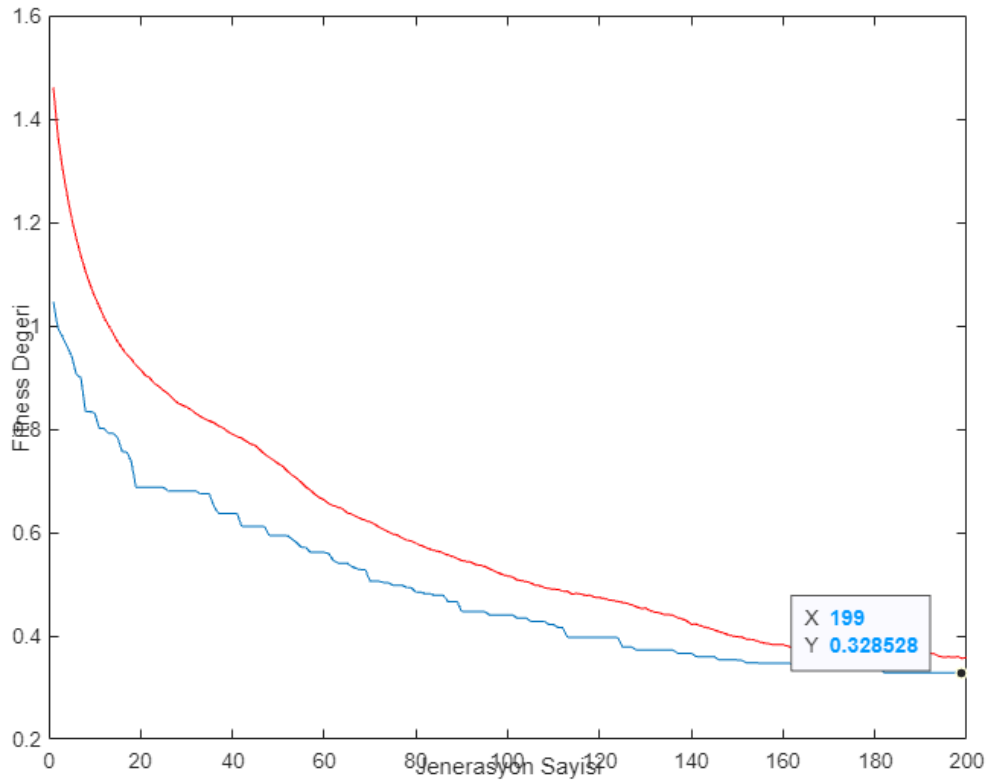
## Gezilen Noktalar



## 1 Drone'un izlediği yol



Kırmızı ile gösterilen popülasyonun ortalama fitness değeri mavi ile gösterilen ise 1 drone'un fitness değeri.



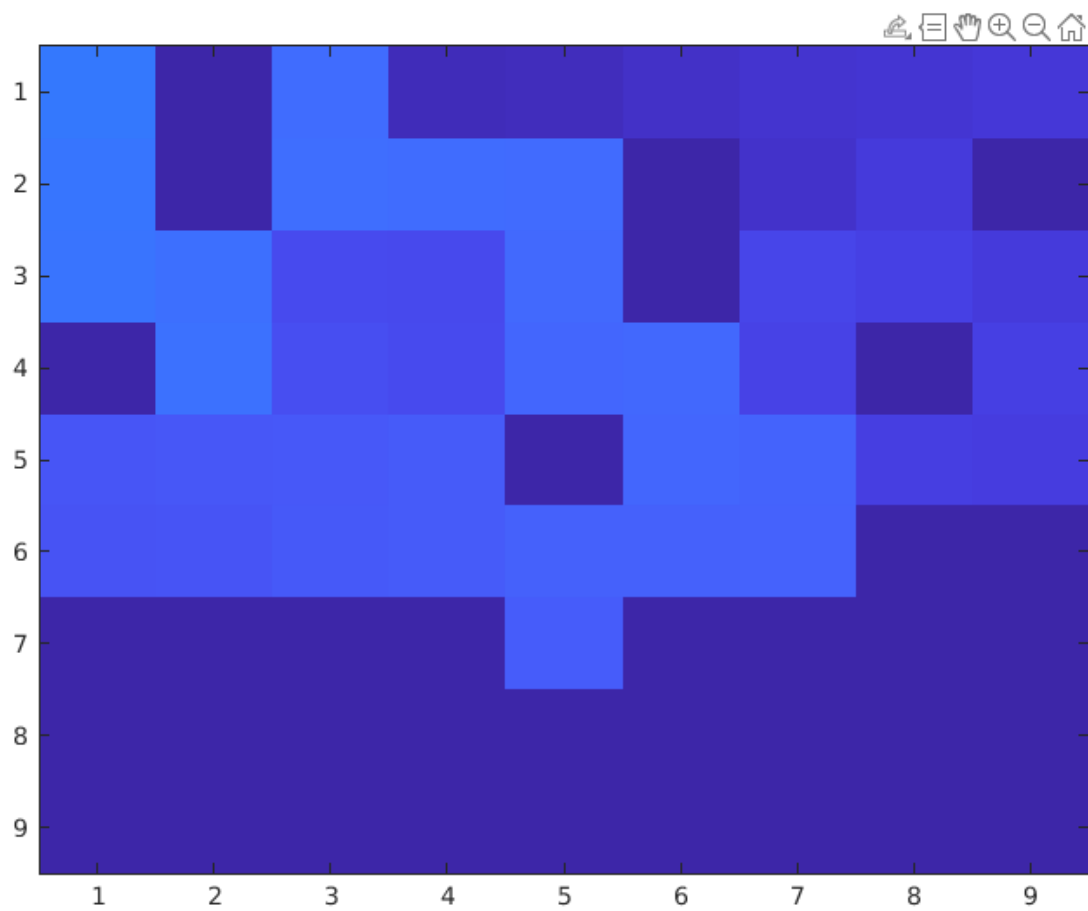


## Deneme 8

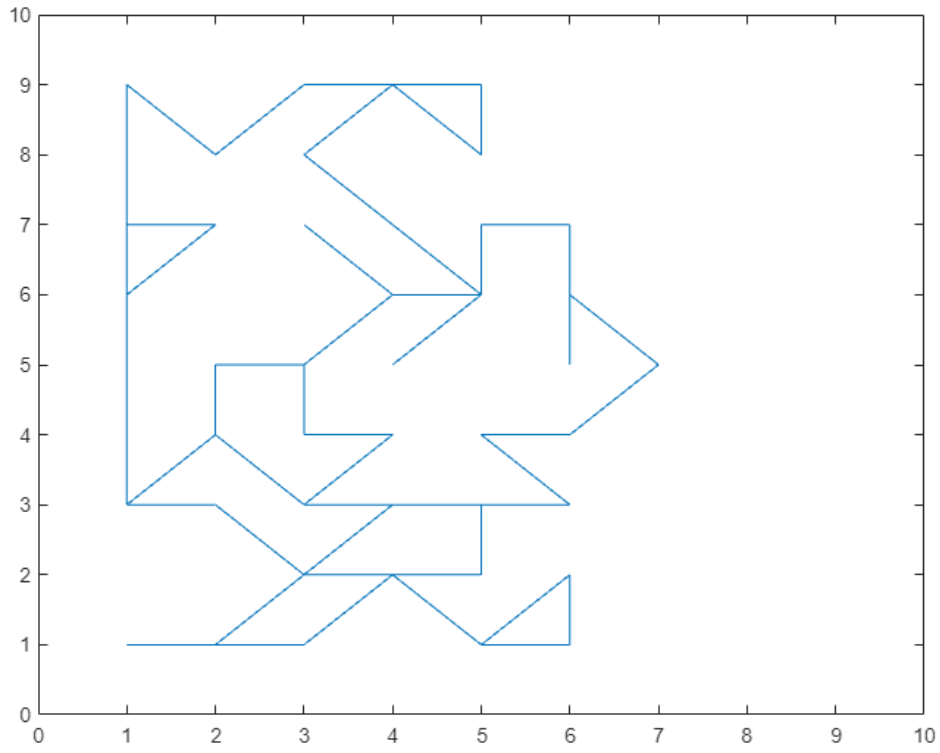
Population: 2000    Generation: 100    Mutation rate : 0.01

Crossover: 2      Number of Drones : 1      Starting Point : [1,1]

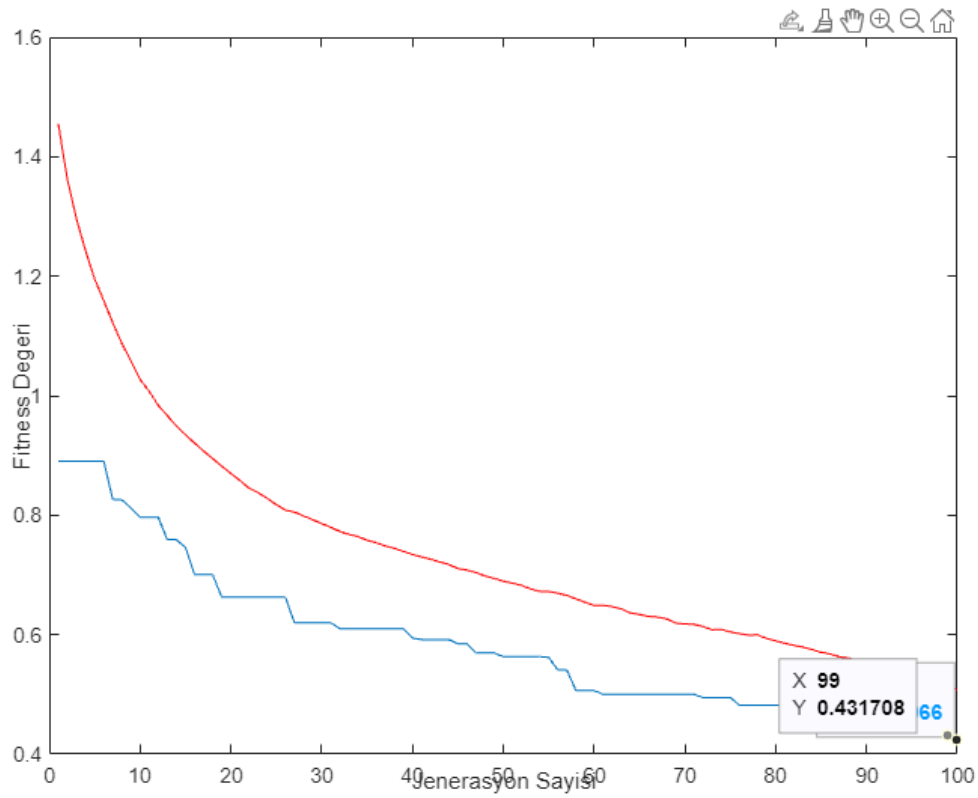
## Gezilen Noktalar



## 1 Drone'un izlediği yol



Kırmızı ile gösterilen popülasyonun ortalama fitness değeri mavi ile gösterilen ise 1 drone'un fitness değeri.

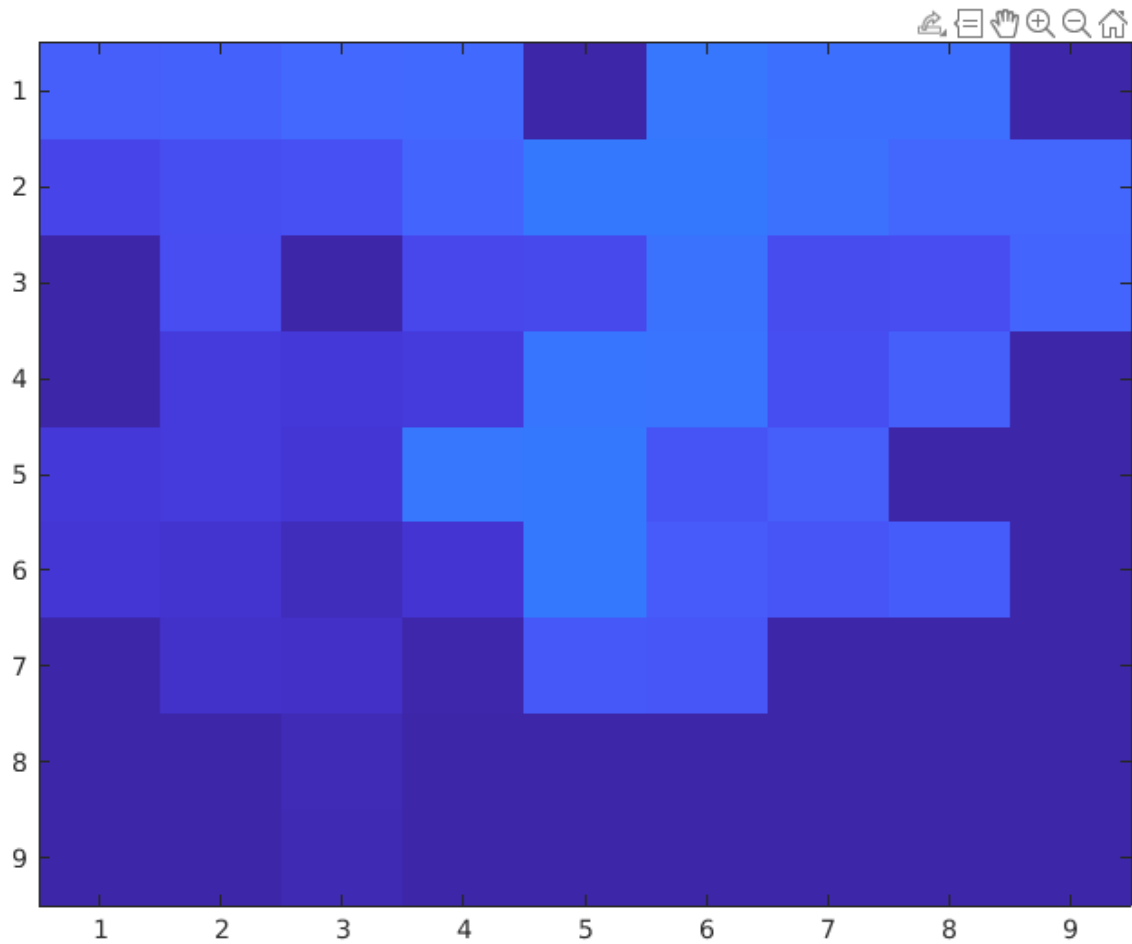


## Deneme 9

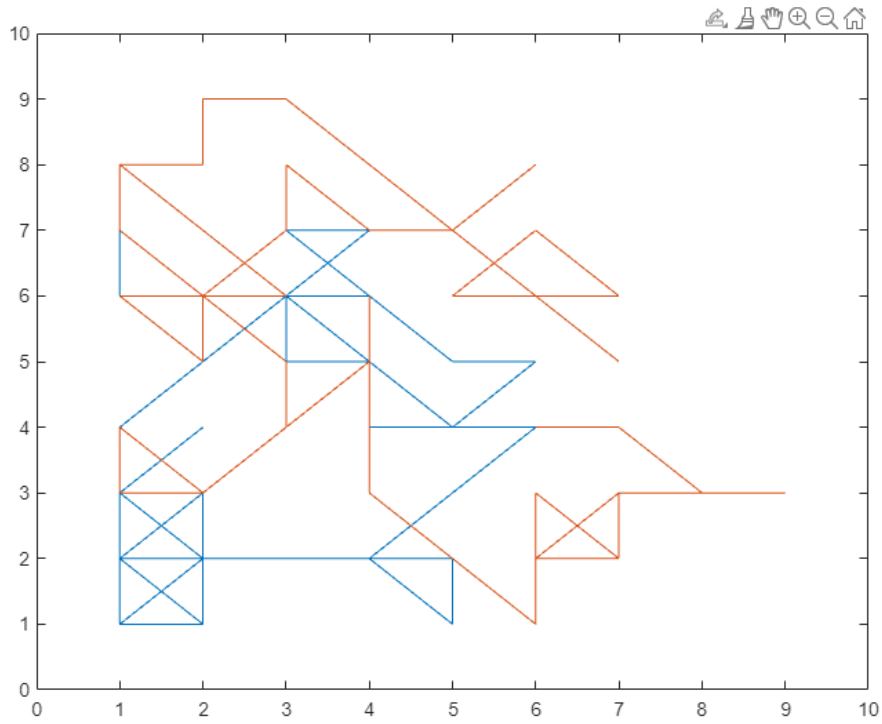
Population: 3000    Generation: 200    Mutation rate : 0.01

Crossover: 2      Number of Drones : 2      Starting Point : [5,5]

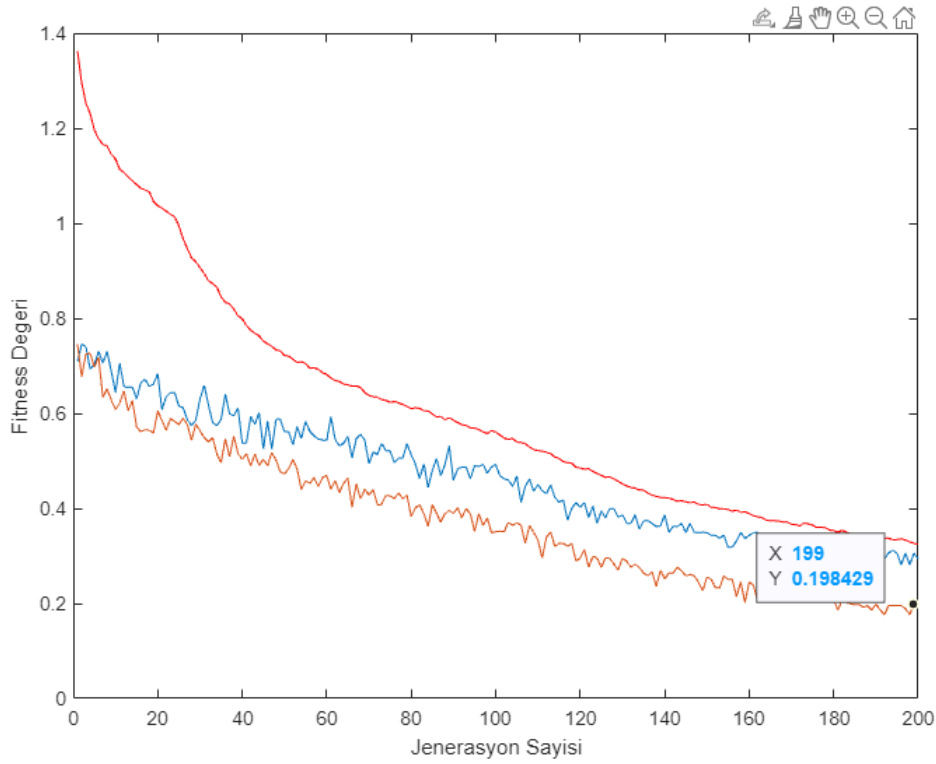
## Gezilen Noktalar



## 2 Drone'un izlediği yollar



Kırmızı ile gösterilen popülasyonun ortalama fitness değerleri mavi ve turuncu ile gösterilen ise 2 drone'un fitness değerleri.

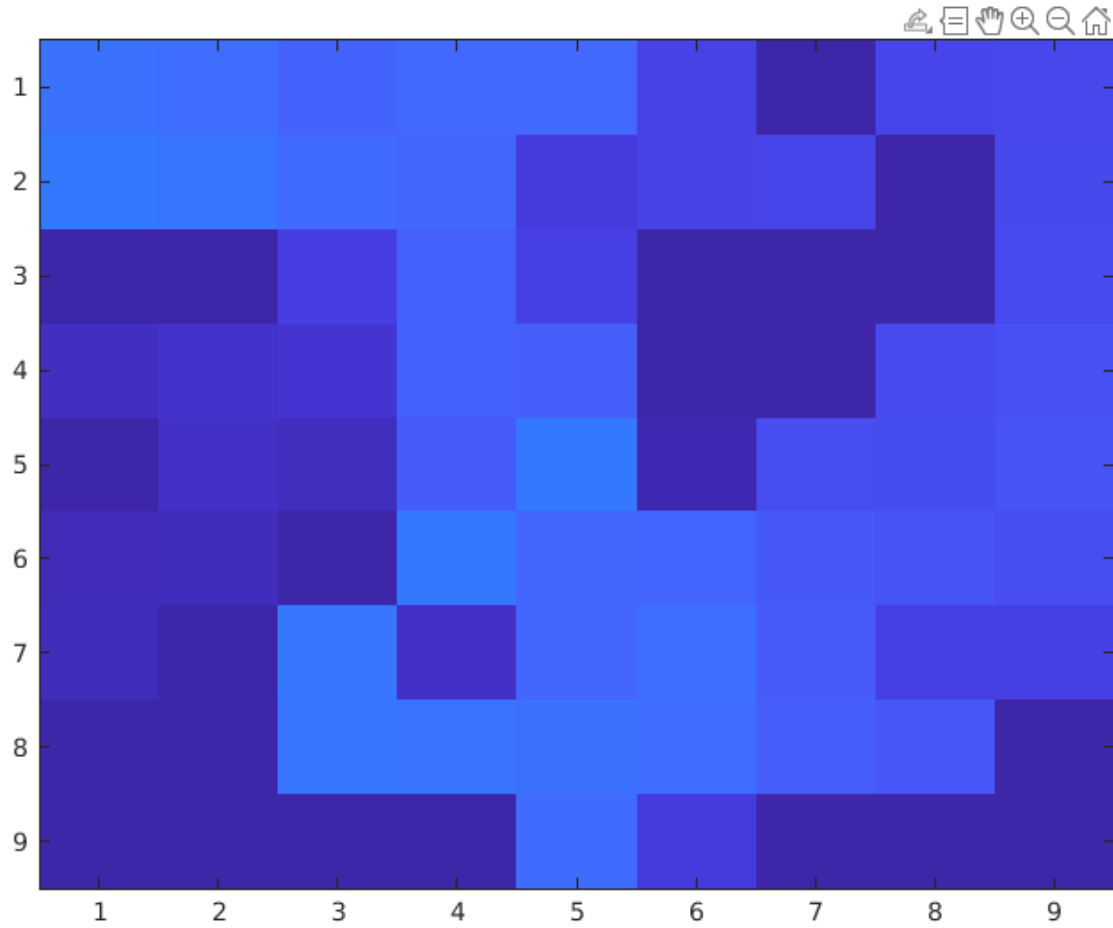


## Deneme 10

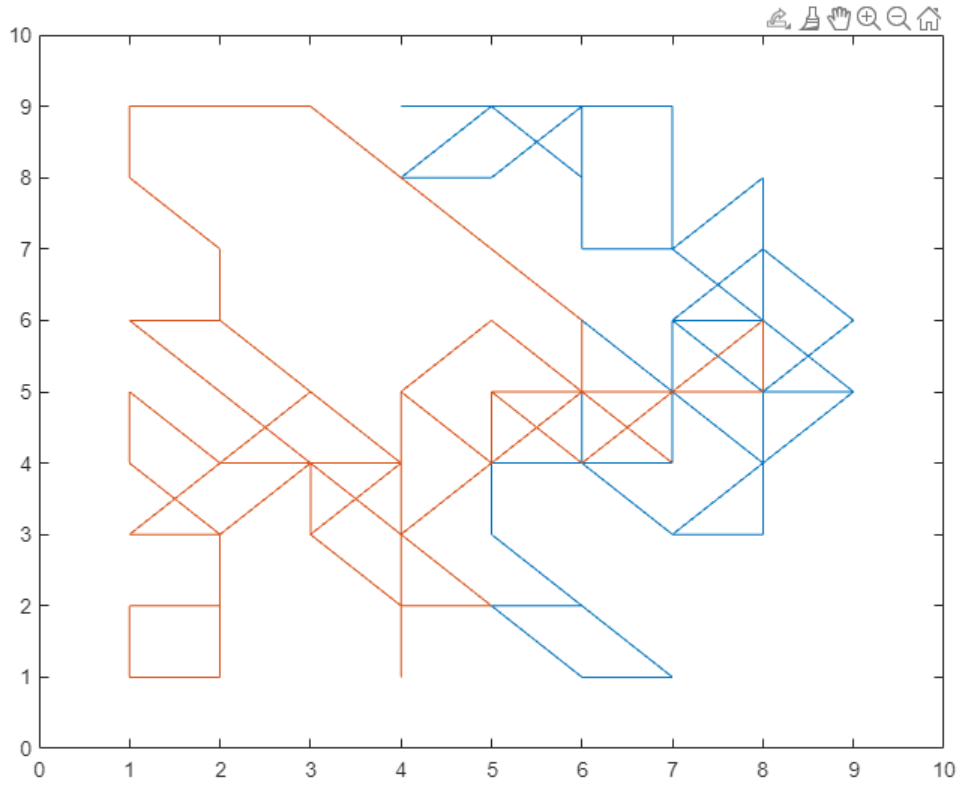
Population: 2000    Generation: 100    Mutation rate : 0.01

Crossover: 2      Number of Drones : 1      Starting Point : [5,5]

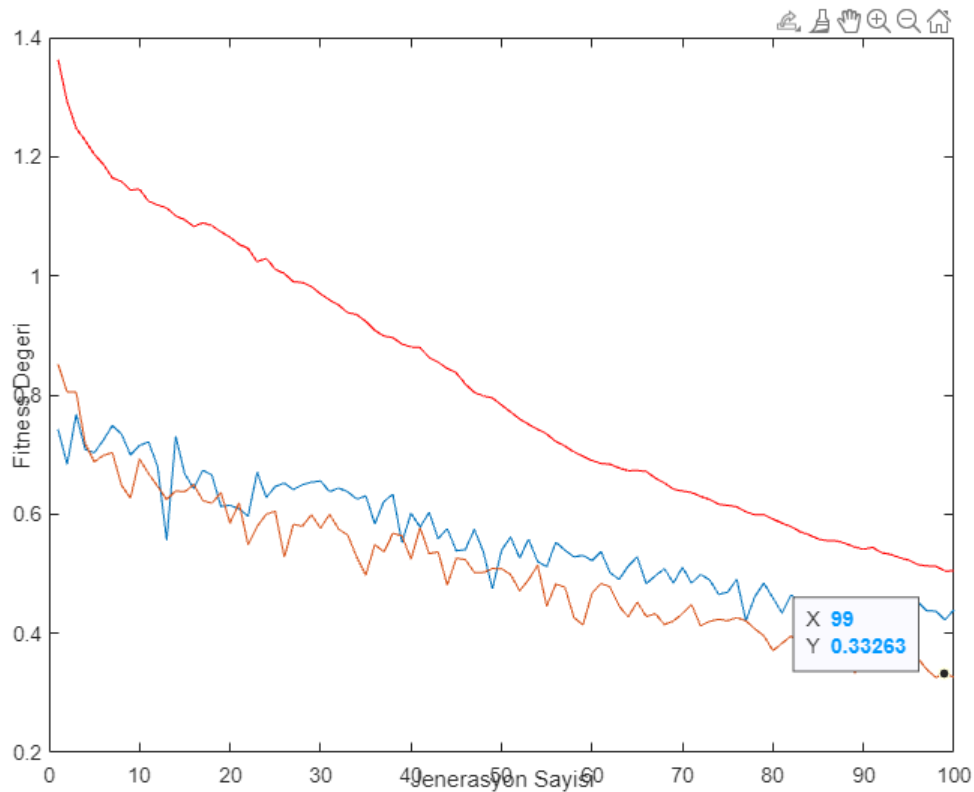
## Gezilen Noktalar



## 2 Drone'un izlediği yollar



Kırmızı ile gösterilen popülasyonun ortalama fitness değerleri mavi ve turuncu ile gösterilen ise 2 drone'un fitness değerleri.

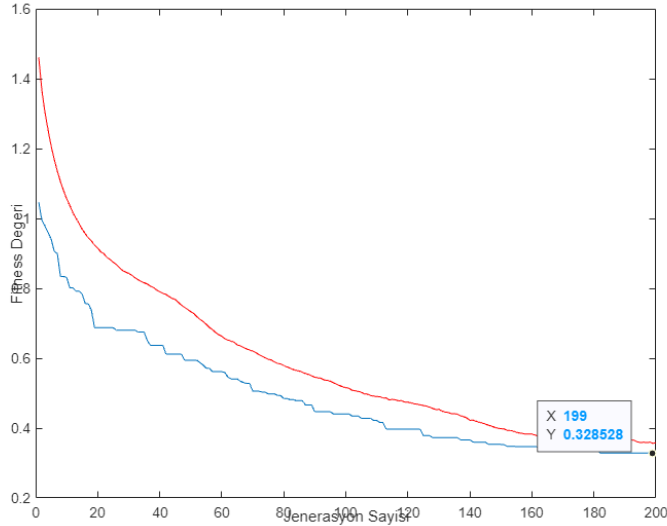


## Hiperparametreler için bulgularım :

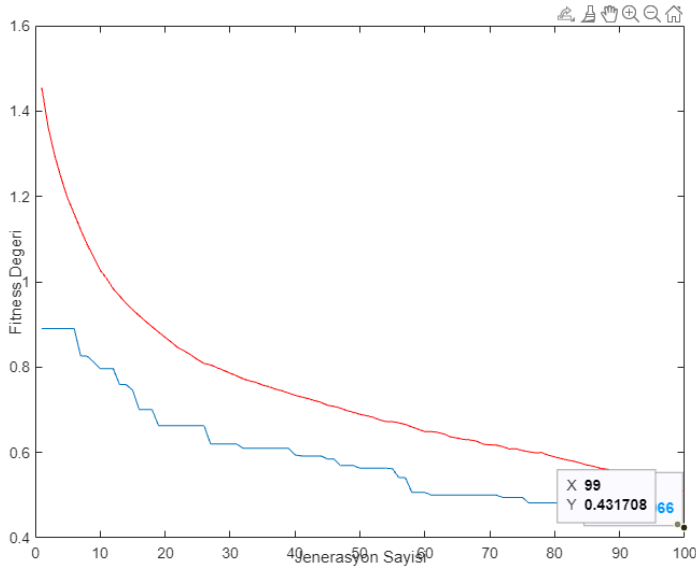
Ben population , generation ve mutation rate degerleri hiperparemetlerinin sabit drone sayilari için degisim miktarlarini inceledim.

Deneme 7 ve Deneme 8 sonuclarinda , girilen deęerler;

Deneme 7 population : 2000 generation: 200 drone sayisi: 1



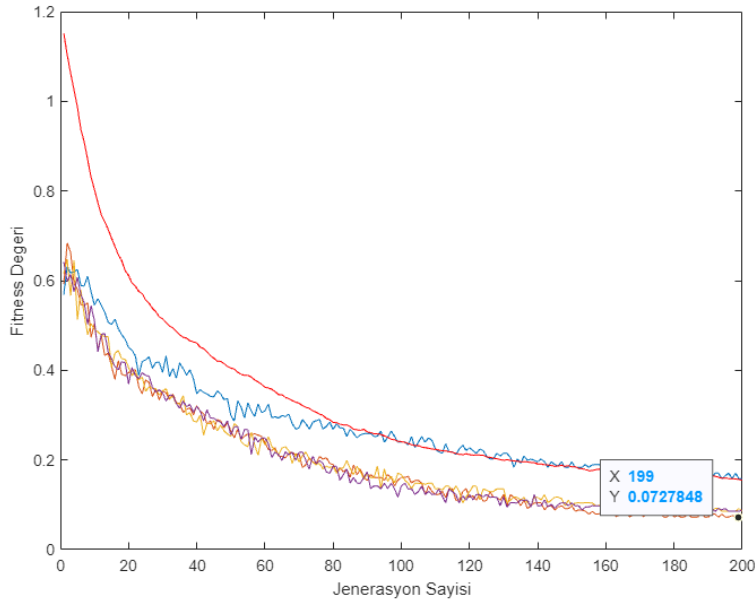
Deneme 8 population : 2000 generation : 100 drone sayisi:1



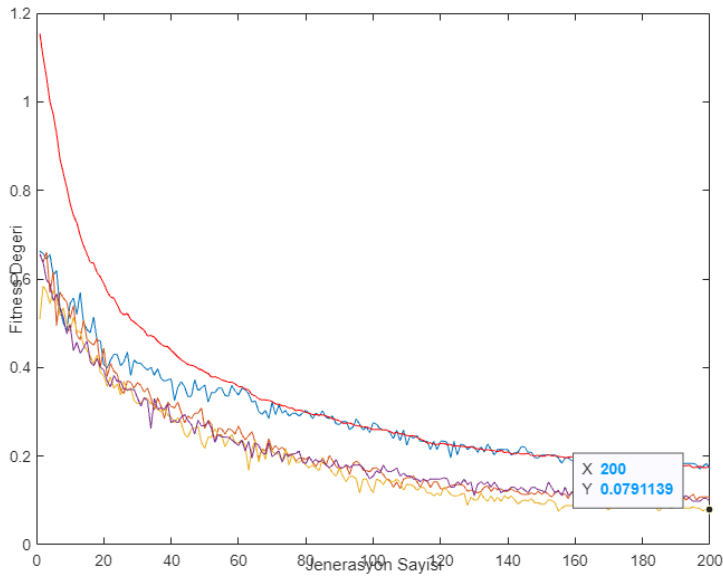
Deneme 7 için generation sayısı , deneme 8 generation sayısından 100 fazladır. Bu fazlalığın grafikler üzerinden incelemesi yapılırsa. Deneme 7 de ekstre incelediğimiz 100 jenerasyon bizi deneme 8 de elde ettiğimiz en iyi fitness deęerlerinden olan 0.431708 den 0.328528 seviyesine çekmiştir. Bu azalmanın(iyileşmenin) sebebi ise ekstradan denediğimiz, crossover yapıp yeni bireyler elde edip, elde edilen bu bireyler ile yeni cozumlerin üretilmesi sonucu olusmustur. Generation sayisinin artmasının cozum bulunmasına pozitif bir etkisi vardır.

Deneme 5 ve Deneme 6 sonuclarinda , girilen deęerler;

Deneme 5 population : 3000 generation: 200 drone sayisi: 4



Deneme 6 population : 2000 generation: 200 drone sayisi: 4



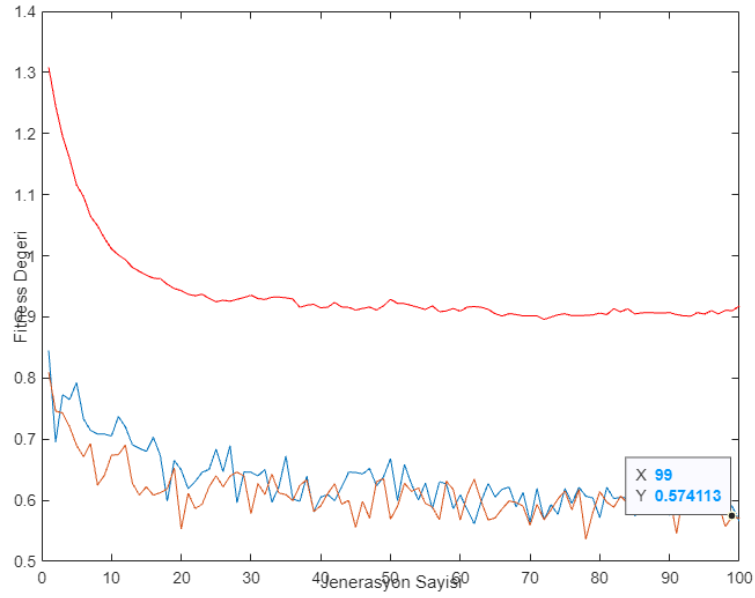
Deneme 5 icin kulllanilan population sayisi 3000 iken , bu sayi deneme 6 icin 2000'dir. Deneme 5'de her generation icin fazladan 1000 cozum deęerlendiriliyor ve bu deęerlendirilmelerin sonucuna gore daha uygun ve iyi bir fitness deęerine ulasma imkanimiz artiyor. Sekildeki grafiklerden de gorulecegi uzere population sayisi daha fazla olan deneme 5'in fitness deęeri daha iyi bir durumdadir. Ve daha dusuk deęere inebilmiřtir(fitness fonksiyonlari nin normalize edilmiř ancak tersi alınmamis halleri grafikleri çizilmede kullandigi icin deęeri duřuęe giden fitness benim kodum icin daha iyi bir duruma ilerliyor). Sonuc olarak daha yuksek population deęeri kullannmamiz, sorunun cozumu icin daha olumlu sonuę vermiřtir.



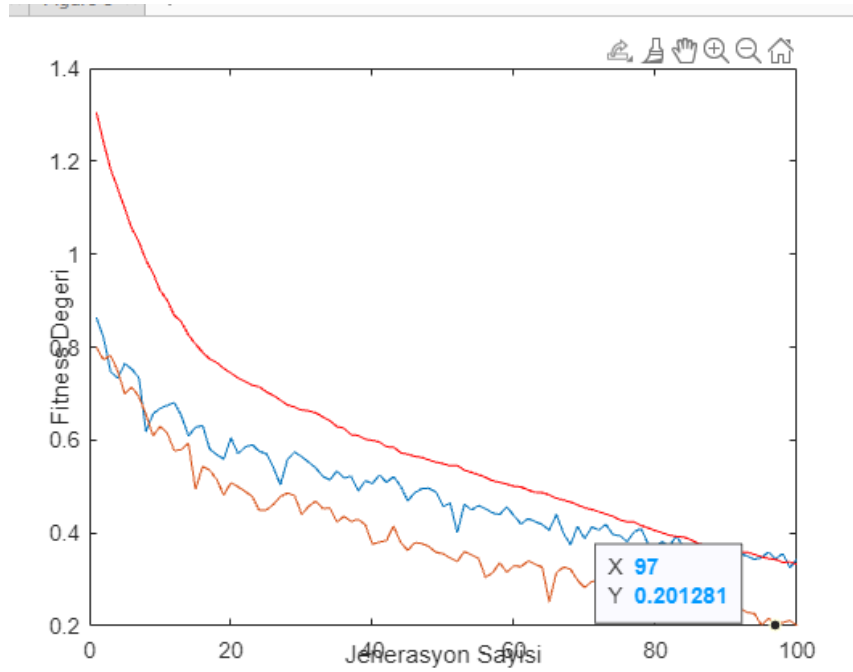
Mutation rate değerinin sonuca etkilerini denemek için;

population : 2000 generation: 100 degerlerini sabit tutup bu değerler etrafında cesitli mutation rate değeri denedim.

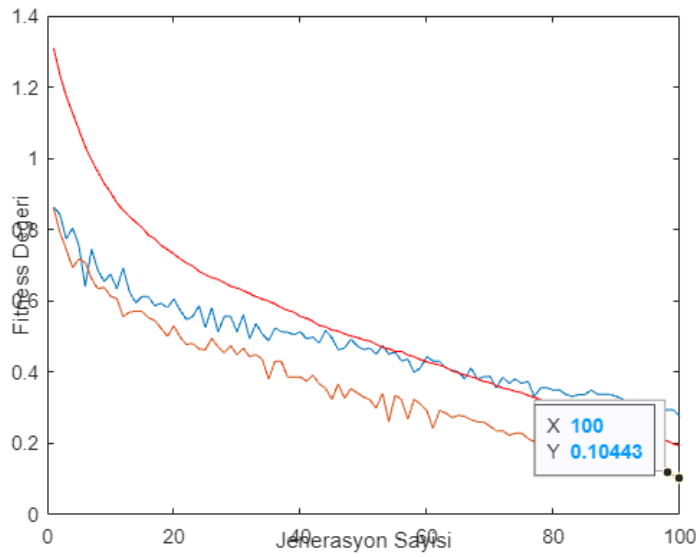
Mutation rate : 0.1



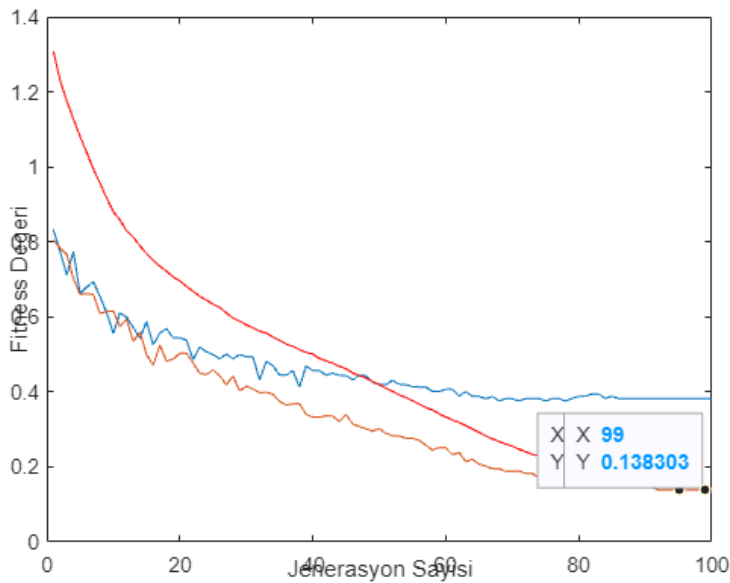
Mutation rate : 0.01



Mutation rate : 0.001



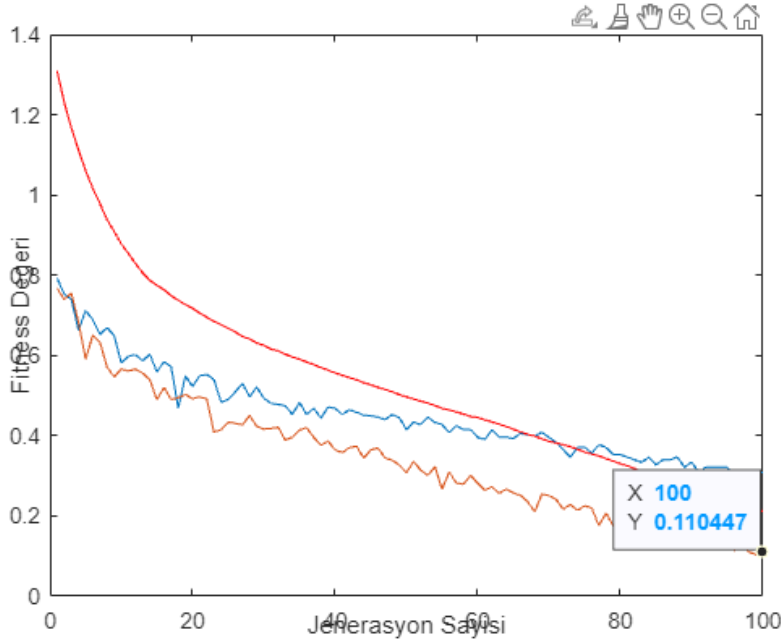
Mutation rate : 0.0001



Mutation rate 0.1 değeri , 2000 population için biraz fazla kaçıyor elimizde olan değerleri çok değiştirmeye baslarsak bir zaman sonra elde ettiğimiz iyi cozumleri de kaybetmeye başlayabiliriz . Mutation rate'yi biraz azaltıp 0.01'e çekersek bu sefer istenilene yakın ve daha stabil elde edeceğimiz iyi cozumleri kaybetmeyen ancak tıkanma durumlarımızda ise yapacağı mutation'lar ile yardım edebileceği bir seviyede . Bu sonuçlar 0.001 ve 0.0001 değerleri için de paralellik gösteriyor. Fitness değerlerimiz iyileşmeye ve daha iyi sonuçlar vermeye devam ediyor ancak 0.001 den 0.0001'e geçiş durumunda fitness değeri iyi bir durumda kalsa da bir önceki duruma göre kötüleşiyor, mutation rate'yi düşürürken de population ile olan oranına da dikkat etmeliyiz.

Mutation rate değerinin fazla yüksek olması iyi bir durum değil ve bizi cozumden uzaklaştırabilir ancak population büyüklüğüne göre çok düşük bir mutation rate ile çalışırsak bu sefer de takıldığımız durumlarda bizi o durumdan çıkaramayacaktır.

Mutation rate : 0.0001 population : 6000



Mutation rate'in fazla düşmesi bir önceki grafikte fitness değerinin biraz kotulesmesine neden olmuştu. Mutation ratemiz kucuk ise bu az degisimin acigini , population sayimizi biraz arttirarak yani daha cok ihtimali mutation işlemi uygulamadan deneme imkani bulabiliriz. Dusuk mutation değerinde population değeri arttırılarak fitness fonksiyonunda iyileşme sağlanabilir.

### Bulgularima Dair Yorumlarım :

Yaptigim calismada kullandigim 3 fitness fonksiyonunun amacına uygun sonuçlar verdiğini ve bu sonuclarin dogrultusunda seçilen değerlerin yeni bireyler uretmesi ile farkli drone sayilari ile tamamini olmasa bile matrix'in cogu kismini dolasıp geri basladiklari yere donen drone'lar oldu. Ancak population , generation , mutation rate , crossover gibi hiperparametre kisimlarinin da farkli denemeler ile hangi değerlerde daha iyi sonuçlar aldigimizin değerlendirilip bu duruma gore hiperparametre değerleri kullanmaliz . Sonuclardan da gorulduđu uzere population , generation değerlerinin yüksek olduđu denemelerde program daha iyi sonuçlar almistir . Bunlarin yaninda mutation rate değerinin de ne az ne cok olacak sekilde, population sayimizi ve çözümlerimizin uzunluğunu da dikkate alarak belirlemeliyiz.

Genetik programlama ile yapilan bu calismada, genel anlamda yaptigimdan daha iyi bir sonuç elde edebilirdim. Coklu drone kullanma durumlarında drone'larin sonuclarini 81 ile değil de 81/drone\_sayisi ustune yeni bir fitness fonksiyonu ekleyerek daha iyi bir sonuç alabileceğime inanıyorum ancak yeterli zamanim kalmadigi için bu durumu deneme imkanim olmadı.

Programın video ile denenmiş halinin youtube linki :

<https://www.youtube.com/watch?v=yIvM9IyprPo>