**Yapay Sinir Ağları**

**5.Ödev**

**21.01.2021**

**Mehmet Şerbetçioğlu -- 040160056**

**Ahmet Hulusi Tarhan -- 040170738**

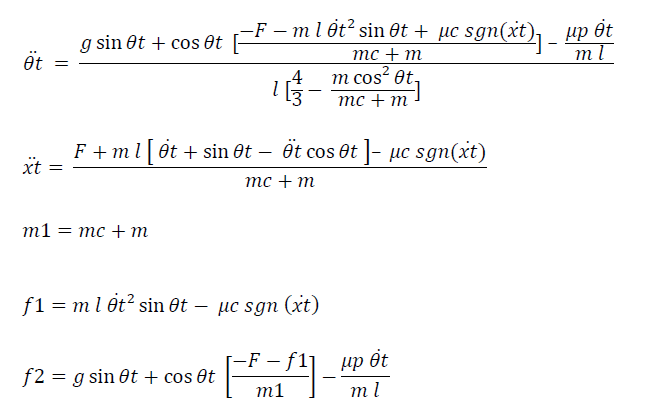
Hafif rüzgarlı bir ortamda, düz bir yol üzerinde bir araba ve arabanın üstünde bir çubuk bulunmaktadır. Arabanın ilerisinde ve arkasında duvar bulunmaktadır. Arabaya kuvvet uygulanarak arabanın duvara çarpmaması ve arabanın üzerindeki çubuğun düşmemesi istenmektedir. Amaç, farklı zamanlarda bu kuvvetin ne yönde uygulanacağına karar verecek bir ağ üretilmesidir.

Bu problemde kullanılacak ağ, pekiştirmeli öğrenme uygulayacaktır. Pekiştirmeli öğrenmede bir sistem, ortamdan aldığı cevap ve bulunduğu duruma göre durumu değiştirecek kararlar verir. Bu sırada ortamdan alacağı cevabı tahmin ederek, istediği cevabı almak için kararlar verir.

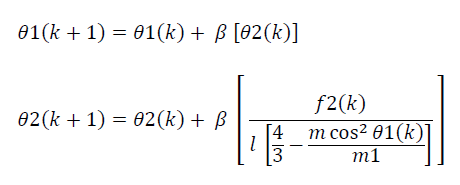
Kurulacak sistemin ve ortamın detayları Tez\_Kuyumcu.pdf dosyasında bulunmaktadır. Arabanın konumu, hızı, ivmesi ve çubuğun açısı, açısal hızı ve açısal ivmesine dair fiziksel eşitlikler de bu dosyadadır. Bunlara ek olarak, ortam koşulları, makalede kullanılan değerlerle alınacaktır.

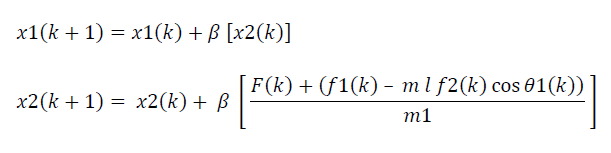
**Fiziksel Eşitlikler:**

Ortamın kurulmasında kullanılan, durum değişkenlerini etkileyecek olan fiziksel eşitlikler şu şekildedir:



Bu diferansiyel denklemler açık euler ile çözülecek olup, şu hali almaktadırlar:





Burada çubuk açısı, çubuk açısının hızı, arabanın konumu ve arabanın hızıdır. ß değeri ise açık euler için kullanılan adım boyutu olup bu sistem için 0.005 alınmıştır.

Fiziksel parametreler ise şu şekilde alınmıştır:

, , , , ,

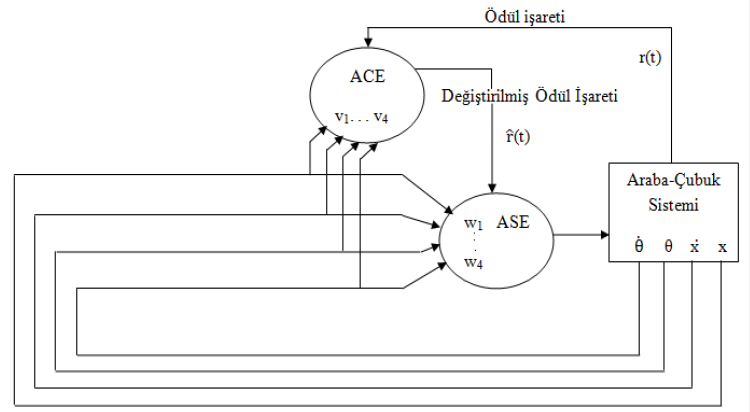
Burada yerçekimi ivmesi, araba kütlesi, çubuk kütlesi, çubuk uzunluğu, araba ve yer arasındaki sürtünme katsayısı ve çubuk ve araba arasındaki sürtünme katsayısıdır. Ayrıca, arabaya uygulanacak kuvveti, ödül beklentisine göre veya olacaktır.

**Ağ Yapısı:**

Ağ temel olarak iki parçadan oluşur. Birinci parça uyarlanabilir eleştiri bileşenidir (ACE). Bu bileşen ortamdan alınan tepki ve sistemin durumuna göre ödül beklentisi değerini oluşturur. Bu değeri ikinci parça olan uyarlanabilir karar bileşenine (ASE) verir.

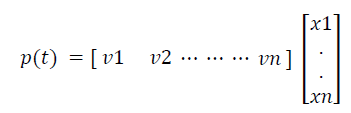
Uyarlanabilir karar bileşeni sistemin davranışını belirler. Bu kararı belirlerken ödül beklentisini kullanır. Ödül alabildiği durum doğrultusunda karar vermeye çalışır.

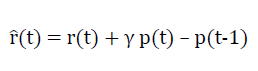
Ağ yapısı, Kuyumcu’nun tezinde şu şekilde ifade edilmiştir:



**Uyarlanabilir Eleştiri Bileşeni:**

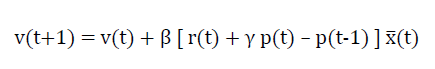
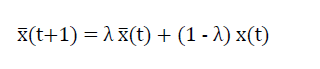
Bu bileşenin amacı ödül beklentisini oluşturmaktır. Bu sırada şu eşitlikleri kullanır:





Burada vektörü bileşenin ağırlıkları, vektörü ise girişindeki durum değişkenleridir. Ödül beklentisinin elde edildiği denklemdeki değişkeni, eleştiri bileşeninin yaptığı tahminin bir önceki tahminle benzerliğini gösterir. olup, makaledeki sistemde değeri kullanılmıştır.

Eleştiri bileşeninin vektörleri şu şekilde güncellenmektedir:

Burada terimi giriş işaretinin seçilebilirliğini göstermektedir. teriminin güncellenmesinde kullanılan ise bozulma oranı olup koşullarına sahiptir. Makaledeki sistemde kullanılmıştır.

**Uyarlanabilir Karar Bileşeni:**

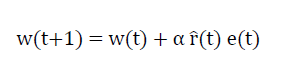
Ödül beklentisi ve sistem durumuna göre, sisteme uygulanacak kuvvetin yönüne karar verir. Bu işlem yapılırken kullanılan denklemler şu şekildedir:



Buradaki fonksiyonu aktivasyon fonksiyonudur. Makalede Bu fonksiyon için kullanılmış olup, kurulacak sistemde çoğunlukla olmak üzere, ve kullanılacaktır.

Aktivasyon fonksiyonu argümanları içinde bulunan ise gürültü olup sistemin durgun kalmasına engel olur. Kurulacak sistemde gürültü, dağılımlı rastgele bir değer alır.

Karar bileşeni ağırlıkları ise şu şekilde güncellenmektedir:





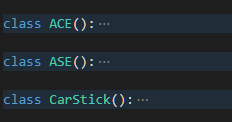
Ağırlıkların güncellendiği denklemdeki değeri ağırlıkların değişme oranını gösterir. terimi davranışlar ve sistemden gelecek hata işareti arasındaki uygunluk derecesini göstermektedir. Bu terimin güncellenmesinde kullanılan değeri ise bozulma sabitidir ve ilişkisine sahiptir.

Karar bileşeni çıkışı, sistemde uygulanacak kuvvetin yönünü belirler. Bu ilişki;

şeklinde ifade edilir.

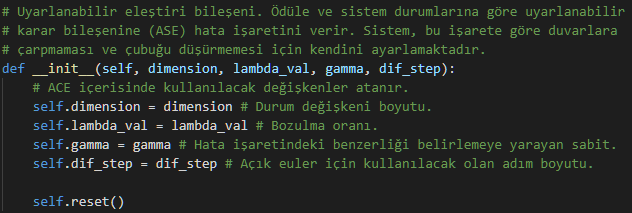
**Sistemin Kurulumu:**

Öncelikle, iki bileşen ve ortam için üç obje oluşturulur.

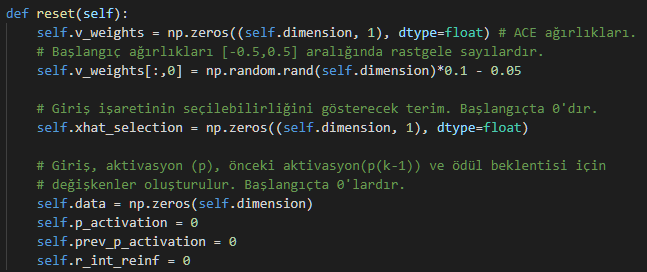


**ACE:**

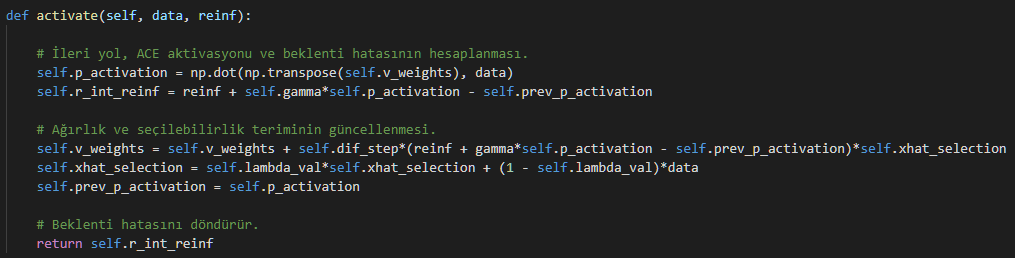
Uyarlanabilir eleştiri bileşeni (ACE) ilk başlatıldığında giriş verisinin (durum değişkenleri) boyutu, ve değerlerini atar. Ek olarak, açık euler çözümünde kullanılacak adım büyüklüğü ß atanır. Değerler atandıktan sonra sistem ağırlıklarının sıfırlandığı “reset” fonksiyonu çağrılır.



Bu fonksiyonda sistemde kullanılacak , , , , terimlerinin ilk değerleri atanır ve başlangıçtaki ağırlıkları belirlenir. Başlangıçta ağırlıklara aralığında rastgele sayılar atanır.

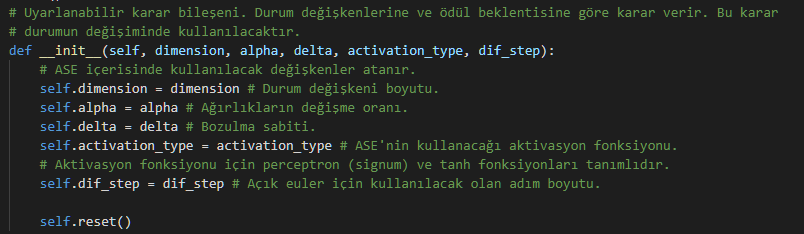


Obje içerisinde ayrıca “activate” fonksiyonu bulunur. Bu fonksiyon, beklenen ödül işaretini oluşturup dışarı verir. Aynı zamanda ağırlıklar bu fonksiyon içinde güncellenmektedir.

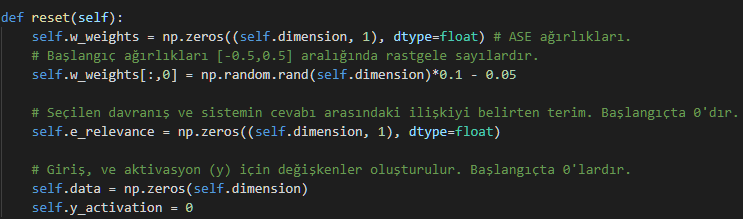


**ASE:**

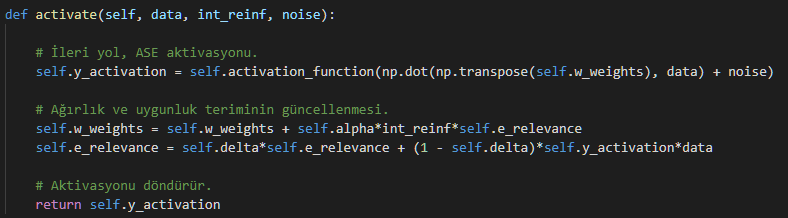
Uyarlanabilir karar bileşeni (ASE) de ACE gibi ilk başlatıldığında kullanılacak sabitleri atar. Giriş boyutu, ve değerleri atanır. Ayrıca kullanılacak aktivasyon fonksiyonu ve açık euler çözümünde kullanılacak adım büyüklüğü atanır. Daha sonra “reset” fonksiyonu çağrılır.



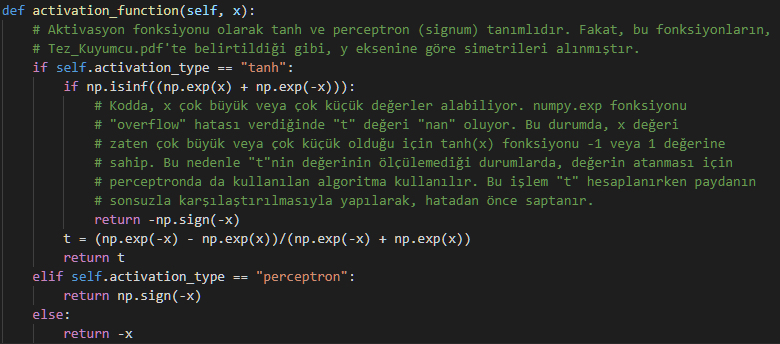
Bu fonksiyonda , ve terimleri atanır. Ayrıca başlangıçtaki ağırlıklar ACE’de olduğu gibi aralığında rastgele değerler alır.



ASE’de bulunan “activate” fonksiyonu, ödül beklentisi ve durum değişkenlerine göre karar verilmesini sağlar. Bu karar verilirken durgunluk olmaması için gürültü de bulunmaktadır. Daha sonra sistem ağırlıkları ve uygunluk terimi güncellenir ve aktivasyon verilir.

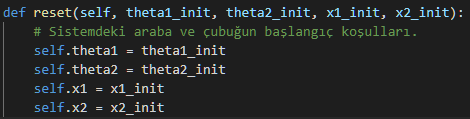
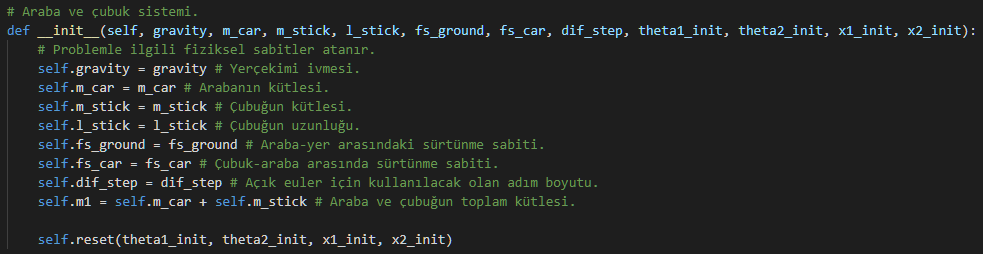


Aktivasyonun bulunması için seçilen aktivasyon fonksiyonu kullanılır. Tanımlı ve fonksiyonları bulunmaktadır. Kodun bu kısmında numpy.exp fonksiyonu kullanılmaktadır. Bu fonksiyonun “overflow” yaşaması durumunda ’in işaretine göre -1 veya +1 değeri seçilir.

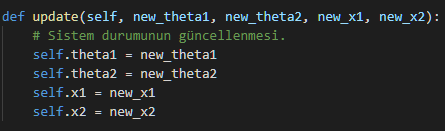
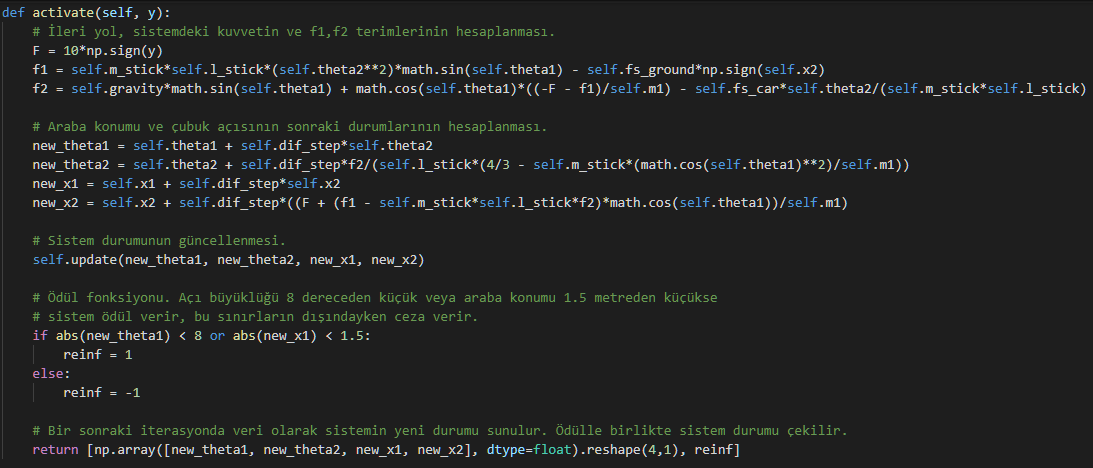


**Araba-Çubuk Sistemi:**

Araba ve çubuğun durum değişkenlerini güncelleyeceği ve ACE’nin kullanması için ödül verecek olan objedir. Başlangıçta sistemle ilgili fiziksel parametreler girilir. Ardından durum değişkenleri için başlangıç koşullarının atanacağı “reset” fonksiyonu çağrılır.

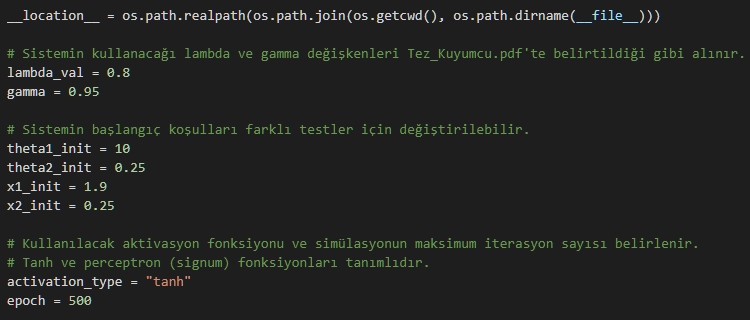


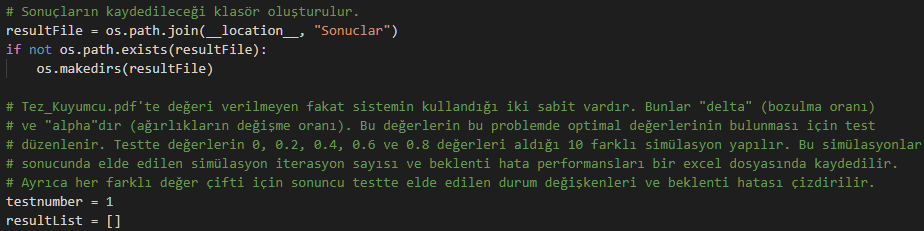
“activate” fonksiyonu, ASE’de alınan karara göre kuvvetin uygulanıp yeni durum değişkenlerinin elde edilmesini sağlar. Bu işleme ve terimlerinin ve kuvvetinin hesaplanmasıyla başlar. Ardından açık euler çözümüyle yeni durum değişkenlerini elde eder ve günceller. Ortamın vereceği ödül veya cezayı seçerek, yeni durum değişkenleriyle birlikte dışarı verir.



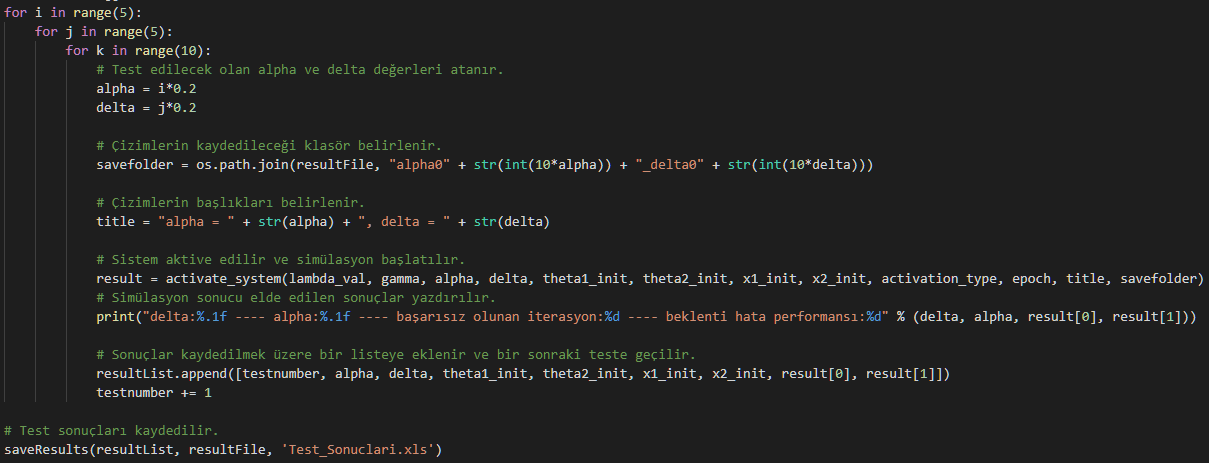
**Sistemin Başlatılması:**

Öncelikle, sistemin kullanacağı ve değerleri makalede kullanıldığı gibi 0.8 ve 0.95 olarak atanır. Ardından sistemin başlangıç koşulları seçilir. Başlangıç koşulları farklı testler için farklı şekilde seçilecektir. Daha sonra ASE’nin kullanacağı aktivasyon fonksiyonu ve simülasyonun devam edeceği maksimum iterasyon sayısı seçilir.

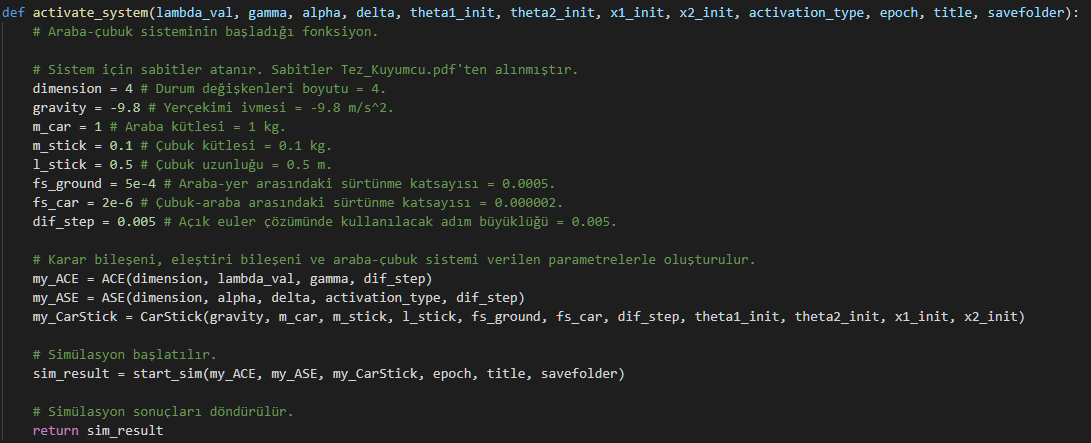


Sonuçların kaydedilmesi için kaydedilecek dosya adresi, test numarası ve boş bir sonuç listesi oluşturulur. 

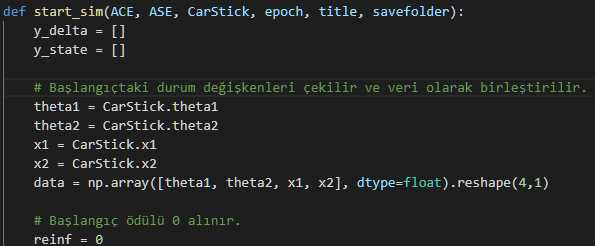
ASE’nin kullanacağı ve değerleri seçilir. Bu değerler makalede verilmediği için, farklı değerler seçilerek farklı durumlarda sistemin performansı karşılaştırılacaktır. Figürlerin kaydedileceği dosya adresi ve başlıkları oluşturulur. Simülasyon başlatılır. Simülasyon sonucunda araba-çubuk sisteminin başarısız olduğu iterasyon sayısı ve simülasyonun beklenti hata performansı elde edilir. Bu değerler, sistemin koşullarıyla birlikte sonuç listesine eklenir. Tüm testler bittikten sonra sonuç listesi bir excel dosyasına kaydedilecektir.



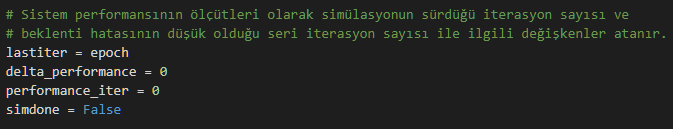
Simülasyon başlamadan önce ACE, ASE ve araba-çubuk sistemleri seçilen fiziksel sabitler, başlangıç koşulları ve sistem katsayıları kullanılarak oluşturulur. Sistemler oluşturulduktan sonra simülasyon başlatılır.



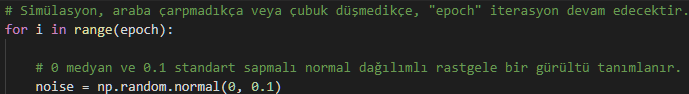
Simülasyonda öncelikle sonuçların çizdirilmesi için beklenti hatası ve durum değişkenlerinin boş listeleri oluşturulur. Daha sonra başlangıçtaki durum değişkenleri çekilir ve bir dizide toplanır. Başlangıç ödülü 0 alınır.



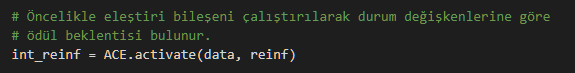
Sonuçların elde edilmesi ve kaydedilmesi için kullanılacak değişkenler atanır. “lastiter” simülasyonun sürdüğü iterasyon sayısını, “delta\_performance” beklenti hatasının 0.5’in altında en uzun süre kaldığı iterasyon sayısını ve “performance\_iter” simülasyon sırasında o an için beklenti hatasının ardışık olarak 0.5’in altında kaldığı iterasyon sayısını belirtir. “simdone” ise simülasyonu durdurma kararıdır. Eğer durdurma kararı alınmışsa son ölçümler yapılır ve simülasyon sonlandırılır. Bu kararın alınması için durum değişkenlerinin belirlenen limitlerin dışına çıkması gerekir.



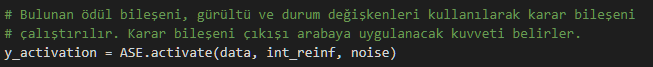
İlk iterasyon başlatılır. İterasyon başında gürültü değeri için daha önce belirtildiği koşullarda rastgele bir sayı seçilir.



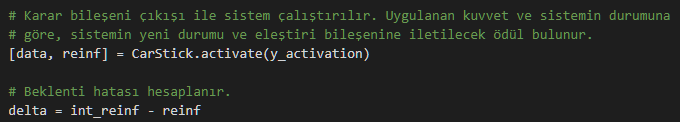
İlk sistem olarak ACE aktive edilerek ödül beklentisi bulunur.



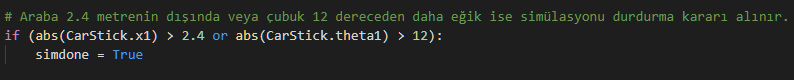
Bulunan ödül beklentisi, durum değişkenleri ve gürültü ile ASE’ye verilir. ASE kararı belirler.



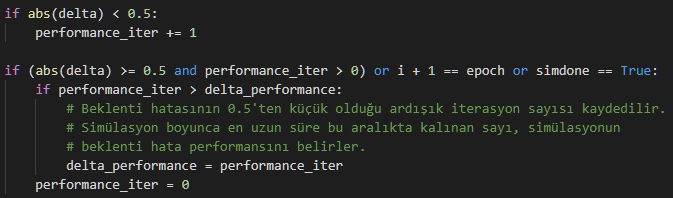
ASE’nin verdiği karar ile araba-çubuk sisteminde kuvvet uygulanır. Değişen açı, açı hızı, konum ve hız değerleri ACE’ye verilecek ödül/ceza ile birlikte elde edilir. Ödül beklentisi ve elde edilen ödülün farkı alınarak beklenti hatası hesaplanır.



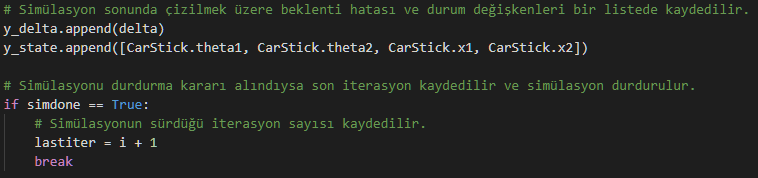
Durum değişkenlerinin belirlenen limitler içinde olup olmadığı kontrol edilir. Bu limitler ve şeklindedir. Limitler dışına çıkılmışsa simülasyonu durdurma kararı alınır. Karar alındıktan sonra son ölçümler ve beklenti hata performansı hesaplanıp simülasyon durdurulur.



Beklenti hatasının büyüklüğünün 0.5’in altında olması durumunda “performance\_iter” değeri artırılır. Eğer beklenti hatası bu sınırın dışına çıkmışsa veya simülasyonun son iterasyonuysa, “performance\_iter” ve “delta\_performance” karşılaştırılır. Eğer “performance\_iter” daha büyükse, “delta\_performance” değerine atanır. Bu şekilde simülasyon süresince beklenti hatasının büyüklüğünün 0.5’in dışına çıkmadığı en uzun iterasyon sayısı elde edilir.



Çizdirilmek üzere beklenti hatası ve durum değişkenleri kaydedilir. Durdurma kararı kontrol edilir ve duruma göre simülasyon durdurulur veya sonraki iterasyona geçilir.



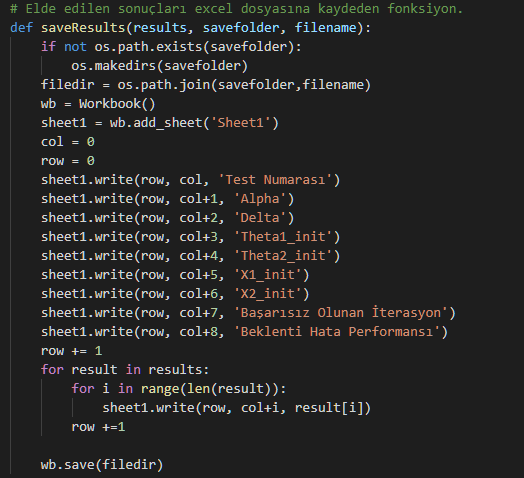
Simülasyon boyunca beklenti hatası ve durum değişkenleri çizdirilip testin kullandığı ve değerlerine göre belirlenen isimle kaydedilir.



Simülasyon sonunda, simülasyonun sürdüğü iterasyon sayısı ve beklenti hatası performansı döndürülür.



Elde edilen sonuçlar daha sonra yorumlanmak üzere xlwt ve xlrd modülleri kullanılarak excel dosyasında kaydedilir.



**Test Sonuçları:**

Test edilecek durumlar;

* ve değerleri {0, 0.2, 0.4, 0.6, 0.8} kümesinden seçilirken sistemin merkezi ve sınıra yakın koşullarda davranışı, seçilen değer çiftlerinin performansları,
* araba ve çubuk merkezi bir konumdayken signum ve tanh fonksiyonlarının performansları,

şeklindedir. Öncelikle tanh fonksiyonu kullanılan bir sistemde ve değer çiftleri karşılaştırılarak iyi performansa sahip değerler seçilecek, ardından bu değerler kullanılarak aktivasyon fonksiyonu olarak signum ve tanh fonksiyonları kullanılacak ve karşılaştırılacaklardır. Karşılaştırılacak iki ölçüt simülasyonun sürdüğü iterasyon sayısı ve beklenti hatası performansıdır. Ayrıca, karşılaştırma yapılırken aynı durumlar 10 kere test edilip karşılaştırma ölçütlerinin ortalamaları alınıp kullanılacaktır.

**, Testi:**

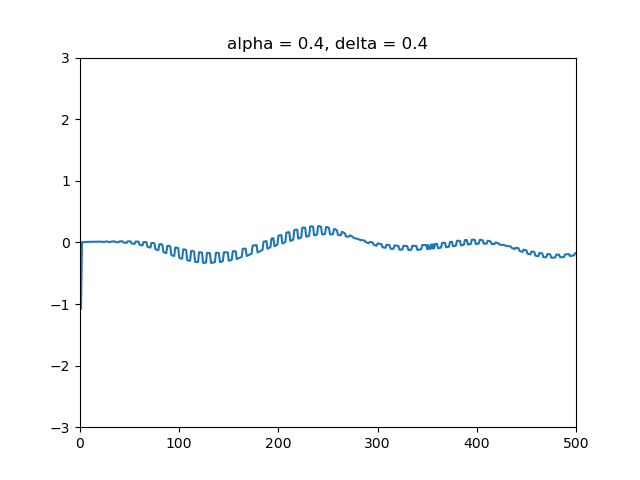
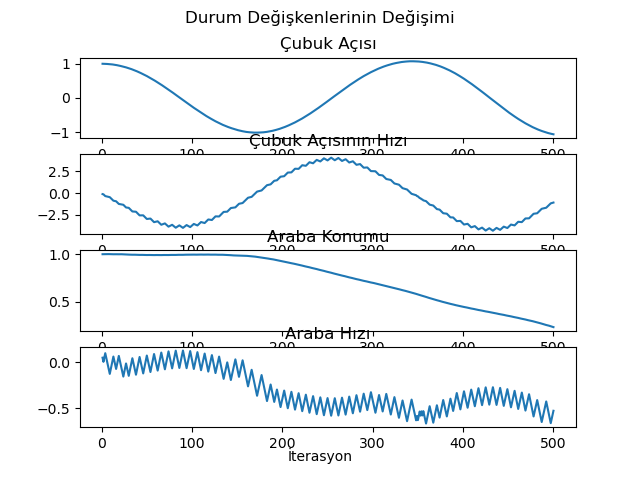
Test daha önce belirtildiği gibi ve değerlerinin {0, 0.2, 0.4, 0.6, 0.8} kümesinden seçilip farklı başlangıç koşullarında simüle edilmesiyle yapılacaktır. Başlangıç koşulları dört şekilde seçilecektir:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Merkezi | Sınırda | Sınırda | ve Sınırda |
|  | 1 | 10 | 1 | 10 |
|  | 0 | 0.5 | 0 | 0.25 |
|  | 1 | 1 | 1.9 | 1.9 |
|  | 0 | 0 | 0.5 | 0.25 |

**Merkezi Durum:**

Bu durumda araba ve çubuk herhangi bir zorlukla karşılaşmazlar. Rahat bir pozisyonda başlayıp sadece küçük düzenlemeler yaparak simülasyona devam ederler.

, değerleri 0.4 iken (0.4 değeri ortalama değer olduğu için seçilmiştir, diğer değer çiftlerinin sonuçları ‘Sonuclar\_Merkez’ klasöründe bulunmaktadır.) durum değişkenleri ve beklenti hatası şu şekilde değişmiştir:

Beklenti hatasının 0’a yakın bir şekilde ilerlediği görülür. Ayrıca çubuk açısının 0 etrafında değişmekte olup araba konumunun 0’a daha da yaklaştığı görülür.

Bu test sonucunda elde edilen beklenti hata performansı tablosu şu şekildedir:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 0 | 0,2 | 0,4 | 0,6 | 0,8 |
| 0 | 293,5 | 284 | 302,8 | 284,9 | 213,6 |
| 0,2 | 499 | 499 | 495,1 | 499 | 499 |
| 0,4 | 499 | 499 | 499 | 499 | 494,5 |
| 0,6 | 499 | 499 | 499 | 499 | 499 |
| 0,8 | 499 | 499 | 499 | 499 | 499 |

Simülasyonun sürdüğü iterasyon sayısı ise şöyledir:

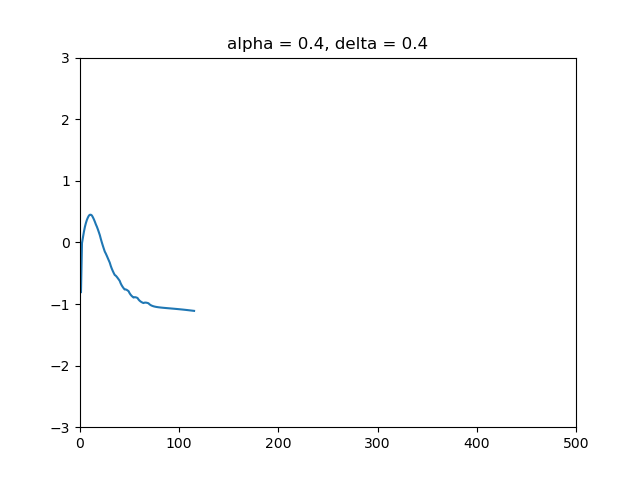
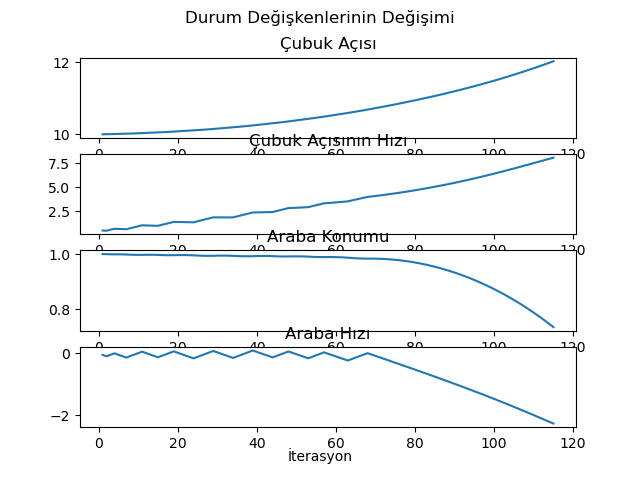
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 0 | 0,2 | 0,4 | 0,6 | 0,8 |
| 0 | 332 | 304,1 | 333,4 | 305,6 | 235,5 |
| 0,2 | 500 | 500 | 500 | 500 | 500 |
| 0,4 | 500 | 500 | 500 | 500 | 500 |
| 0,6 | 500 | 500 | 500 | 500 | 500 |
| 0,8 | 500 | 500 | 500 | 500 | 500 |

Görülebileceği gibi, = 0 olduğu durumlarda sistem istenilen şekilde çalışmamıştır. Araba ve çubuk belirtilen sınırların dışına çıkmış ve simülasyon erken bitmiştir. Buna rağmen beklenti hata performansı, simülasyonun sürdüğü iterasyon sayısına yakın değerler almıştır. Bu testten değerinin sisteme etkisi hakkında yorumlanabilecek bir veri elde edilmemiştir.

**’nın Sınırda Olması Durumu:**

Bu durumda başlangıçta çubuğun açısı 10 derece ve açının hızı 0.5 seçilir. Arabanın, çubuk düşmeden açısını düşürmeye çalışması gerekir.

, değerleri 0.4 iken durum değişkenleri ve beklenti hatası şu şekilde değişmiştir:

Simülasyon sırasında arabanın çubuk açısını düzeltmek için yeterince hızlı karar vermediği ve ilerlemek istemediği, çubuğun düşmesine izin verip simülasyonun bitmesine sebep olduğu görülür. Bu durum, diğer değer çiftleri için gerçekleştirilen simülasyonlarda da gözlemlenmektedir.

Test sonucunda elde edilen beklenti hata performansları:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 0 | 0,2 | 0,4 | 0,6 | 0,8 |
| 0 | 33,2 | 33,36 | 32,3 | 33,2 | 33,1 |
| 0,2 | 33,7 | 32,4 | 33,3 | 33,2 | 33,3 |
| 0,4 | 33,5 | 33,1 | 33,3 | 33,1 | 33,5 |
| 0,6 | 33,1 | 33,4 | 33,5 | 33,1 | 33,3 |
| 0,8 | 33,7 | 32,5 | 33,2 | 33,2 | 33,3 |

Test sonucunda elde edilen simülasyon iterasyonu:

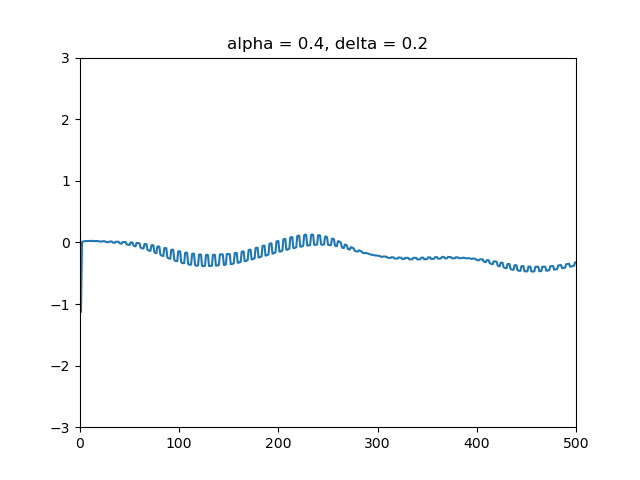
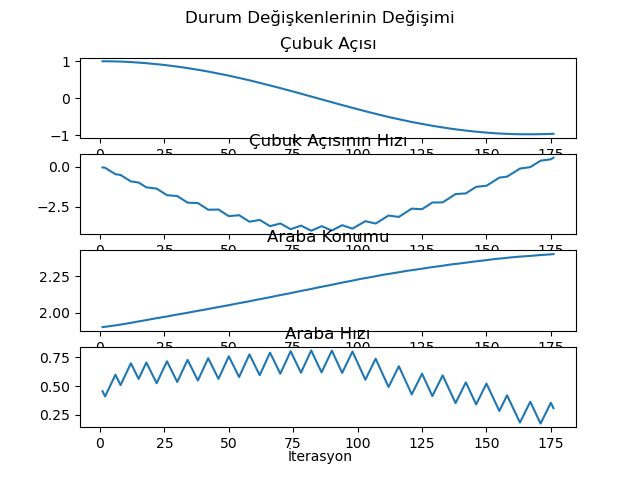
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 0 | 0,2 | 0,4 | 0,6 | 0,8 |
| 0 | 144,4 | 126,1 | 124,6 | 119,5 | 125,2 |
| 0,2 | 114,6 | 113,9 | 113 | 114,2 | 113,5 |
| 0,4 | 113,8 | 114,4 | 113 | 113,5 | 113,8 |
| 0,6 | 112,9 | 113 | 113,4 | 113,4 | 113,4 |
| 0,8 | 113,8 | 113,5 | 113,5 | 112,4 | 113,3 |

Test sonucunda yeterince değerli veriler elde edilememiştir. Arabanın başlangıç koşullarından toparlanması fazlasıyla zordur ve bu sebeple tüm testlerde yakın veriler elde edilmiştir. Göze çarpan tek sonuç, ve düşük olduğu değerlerde simülasyonun daha uzun süre devam etmesidir. Bu özellikle değeri için geçerlidir.

**’in Sınırda Olması Durumu:**

Önceki test gibi sınır koşullarda başlatılan bir durumdur. Bu sefer çubuk rahat bir açıdayken araba duvara yakın bir konumda bulunmakta ve hızlanmaktadır.

, değerleri 0.4 iken durum değişkenleri ve beklenti hatası şu şekilde değişmiştir:

Beklenti hatasının 0’a yakın bir değerde ilerlediği görülür. Ayrıca arabanın hızını yeterince çabuk düşüremeyip önündeki duvara çarptığı ve simülasyonun durdurulduğu gözlenir.

Test sonucunda elde edilen beklenti hata performansları:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 0 | 0,2 | 0,4 | 0,6 | 0,8 |
| 0 | 287 | 260,27 | 186,2 | 236,3 | 185,2 |
| 0,2 | 293,6 | 474,6 | 255,6 | 234,1 | 236,3 |
| 0,4 | 429 | 473 | 350,6 | 336,1 | 272,3 |
| 0,6 | 451,8 | 468,4 | 412,7 | 307,9 | 221,8 |
| 0,8 | 463,8 | 471,4 | 380,9 | 356,7 | 356,4 |

Test sonucunda elde edilen simülasyon iterasyonu:

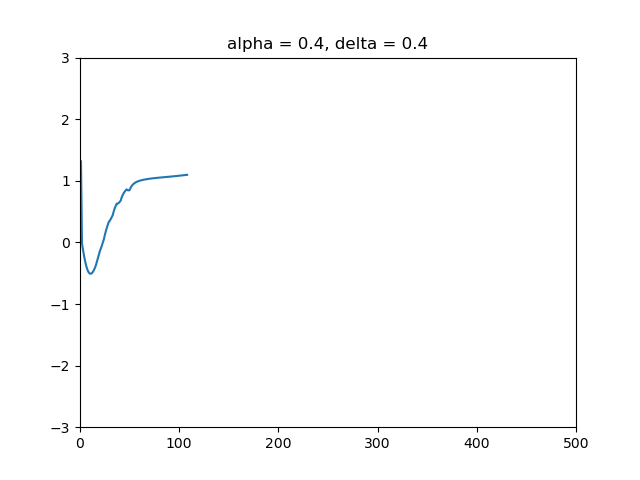
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 0 | 0,2 | 0,4 | 0,6 | 0,8 |
| 0 | 315,6 | 273,2 | 191,5 | 245,1 | 186,2 |
| 0,2 | 308,7 | 500 | 278,3 | 241,4 | 253,2 |
| 0,4 | 493,9 | 500 | 395 | 371,8 | 283,8 |
| 0,6 | 500 | 500 | 470,3 | 341,6 | 228,7 |
| 0,8 | 500 | 500 | 430,9 | 405 | 384,9 |

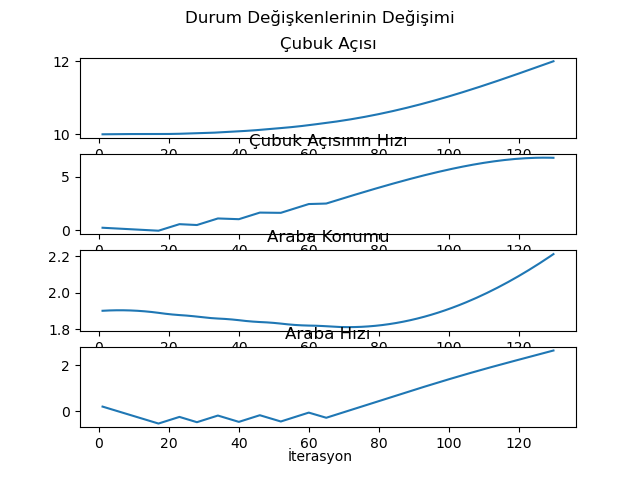
Beklenti hata performansları incelendiğinde, değerinin yüksek ve değerinin düşük olmasının bu değeri iyi yönde etkilediği görülebilir. Bunu özellikle değerinin yüksek olması etkiler. Beklenti hata performansındaki farklılığın sebebi ise çoğunlukla simülasyonun sürdüğü iterasyon sayısındaki farklılıktır. ’nın yüksek ve ’nın düşük olduğu durumlarda araba-çubuk sistemi limitlerin dışına çıkmayıp maksimum iterasyona ulaşmayı başarmışlardır. Bu test için en başarılı değer ikilisi olup maksimum iterasyona ulaşılan diğer simülasyonlar arasındaki fark fazlasıyla düşüktür.

**ve ’in Sınırda Olması Durumu:**

Bu durumda hem araba hem de çubuk limitlerine fazlasıyla yakın bir durumda başlamaktadırlar. Çubuğun düşmemesi için arabanın hızlanması gerekir fakat hızlanırsa duvara çarpma ihtimali vardır.

, değerleri 0.4 iken durum değişkenleri ve beklenti hatası şu şekilde değişmiştir:





Koşullar, sistemin dengeyi bulması ve simülasyonu bitirebilmesi için fazlasıyla zordur. Buna rağmen bazı testlerde sistemlerin diğerlerinden daha iyi performans sergilediği görülebilir.

Test sonucunda elde edilen beklenti hata performansları:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 0 | 0,2 | 0,4 | 0,6 | 0,8 |
| 0 | 31,8 | 28 | 31,6 | 31,8 | 30,8 |
| 0,2 | 30 | 29,9 | 29,9 | 31,6 | 30,3 |
| 0,4 | 30,7 | 28,1 | 29,4 | 31,4 | 27,7 |
| 0,6 | 32,2 | 28,8 | 30,3 | 28,9 | 29,8 |
| 0,8 | 32,8 | 30,9 | 29,5 | 29,5 | 31,2 |

Test sonucunda elde edilen simülasyon iterasyonu:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 0 | 0,2 | 0,4 | 0,6 | 0,8 |
| 0 | 136,3 | 123,9 | 150,1 | 110,8 | 115 |
| 0,2 | 123,9 | 124,9 | 119 | 135,2 | 173,1 |
| 0,4 | 120 | 114,1 | 119,1 | 120,9 | 116,7 |
| 0,6 | 117,4 | 115,7 | 119,4 | 119,2 | 118,9 |
| 0,8 | 122,1 | 117,8 | 123 | 120,3 | 115,4 |

Görüldüğü üzere testler arasında pek bir farklılık bulunmamaktadır. Buna iki istisna simülasyonun , durumunda 150 iterasyon, , durumunda 173 iterasyon devam etmesidir. Bu iki değer bir düzen göstermedikleri için , değerleri seçilimi için fazla bilgi vermezler. Ek olarak bu iki değerin beklenti hata performansları diğer testlere oldukça yakın, hatta birçoğundan düşüktür.

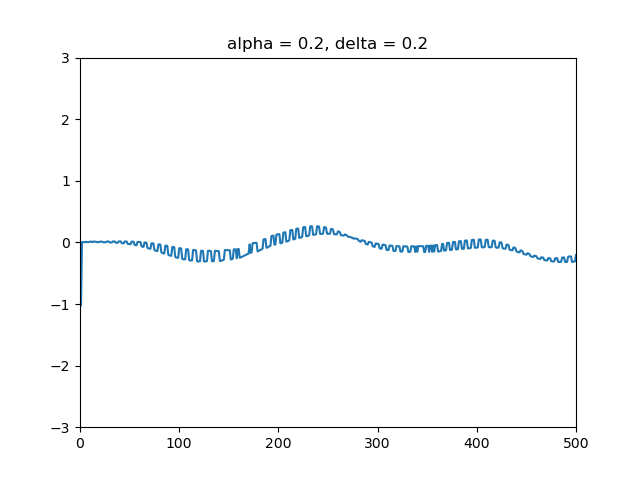
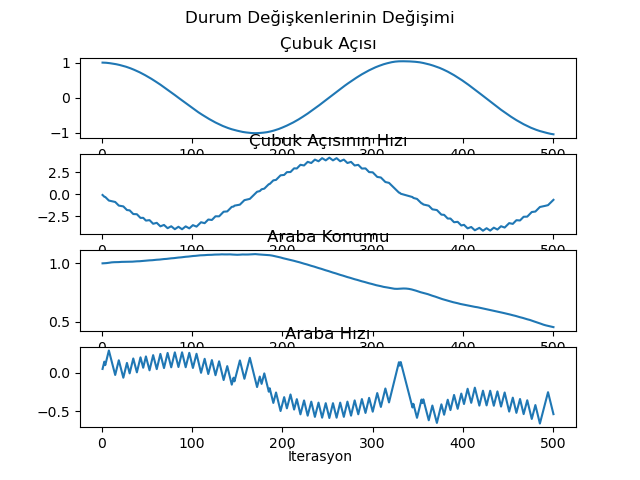
Açının sınırda olması, sistemin toparlanmasını fazlasıyla zorlaştırmaktadır. Sistem açıyı toparlamakta yeterli cevabı veremez. Bu durum konumun sınırda olması durumunda farklı olup, doğru , değerlerinin seçilmesiyle sistem durum değişkenlerini rahat bir konuma getirip simülasyonu tamamlayabilmektedir.

Test sonuçları incelendiğinde için , değerleri için en değerli bilgiyi konumun sınırda olduğu testin verdiği görülür ve bir sonraki test için değerleri seçilir.

**Aktivasyon Fonksiyonu Seçimi:**

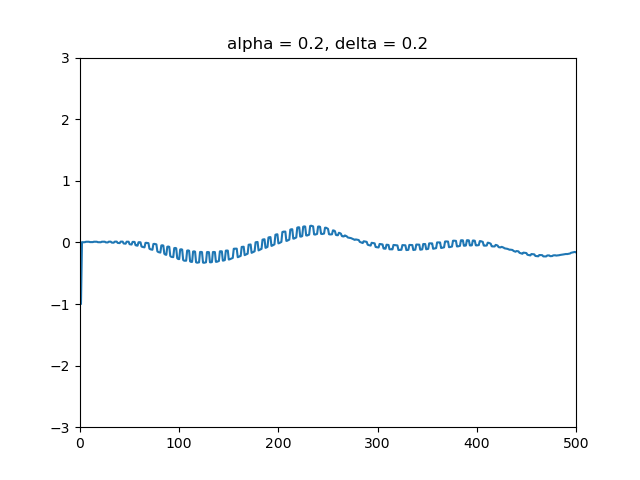
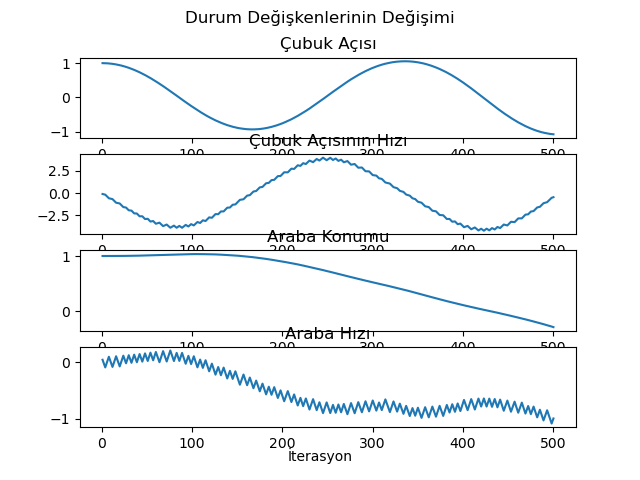
ASE’nin kullanacağı aktivasyon fonksiyonu için ve seçenekleri arasındaki farklılık incelenecektir. En iyi performansı verdiği gözlenen değer çifti seçilir. Merkezi durumdaki başlangıç koşulları kullanılacaktır.

Tanh seçilmesi durumunda:

Araba, konumunu 0 noktasına yaklaştırmış ve üstündeki çubuğu dengede tutmuştur.

Signum seçilmesi durumunda:

Araba ve çubuk, signum fonksiyonu seçildiğinde de dengede kalmışlardır. Beklenti hatasındaki değişimin çok az bir farkla da olsa 0’a daha yakın ilerlediği görülür. Ayrıca sistem daha hızlı ve daha stabil kararlar almaktadır. Bu özellikle arabanın hızındaki değişimlerde görülür. Bu problem için signum fonksiyonunun daha iyi bir performans sergilediği görülebilir.