

Ritmik metastabil robotik sistemler için veri-güdümlü ortalama ilk geçiş zamanı metriği

Jeanpiere DEMİR
Tasarım ve Önmodelleme
Uygulama ve Araştırma Merkezi
ODTÜ
Ankara, Türkiye
Email: jdemir@metu.edu.tr

M. Mert ANKARALI
Elektrik-Elektronik Mühendisliği
ODTÜ
Ankara, Türkiye
Email: mertan@metu.edu.tr

Özetçe —Robotikteki çoğu sistem deterministik olarak kararlı bir dinamik yapıya sahip olsa da, kaçınılmaz stokastik bozulardan dolayı metastabil (yarı kararlı) bir davranış içindedir. Ne yazık ki deterministik sistemler için üretilmiş veya geliştirilmiş kararlılık kavramları stokastik yapılar için yeterli kalmamaktadır. Bu problemi çözmek için değişik alanlardaki bazı çalışmalarda metastabilite ölçüsü olarak OIGZ (ortalama ilk geçiş zamanı) kullanılmıştır. Bu kavram bir hata tabiri üzerine kurulur. Bacaklı robotikte robotun düşmesi, kimyasal tepkimede beklenmeyen bir çıktının oluşması hata tabirlerine örnektir. Genel olarak bir sisteme baktığımızda, sistemin kararlı deterministik öğeleri sistemi denge noktasına (veya limit döngüsü) iterken, stokastik öğeleri sistemi hata bölgesine itmeye çalışır. Teknik olarak OIGZ sistemin bu öğeler altında ne zaman hatalanacağını beklentisini verir. Fakat, bu metriğin kestirimi benzetilmiş sistemler için bile kolay olmayabilir. Bu çalışmada, bu metriğin veri-güdümlü bir şekilde tahmin eden bir metod sunmaktayız. Yaklaşımımız, kararlı bir limit döngü etrafında hareket eden bir sistemin, seçilen bir Poincare kesit uzayında stokastik bir Markov zinciri olarak tahmin etmeye dayanır. Çalışmamızın en sonunda da yaklaşımımızı test etmek için, basit bir stokastik, ritmik bir sistemle simülasyon ortamında Monte Carlo deneyleri gerçekleştirdik. Performans testi için ise başka bir çalışmada önerilmiş bir yöntemle de karşılaştırıp yaklaşımın istatistiksel olarak performansı arttırdığımızı gördük.

Anahtar Kelimeler—metastabil sistemler, stokastik süreçler, ortalama ilk geçiş zamanı (mean first passage time)

I. GİRİŞ

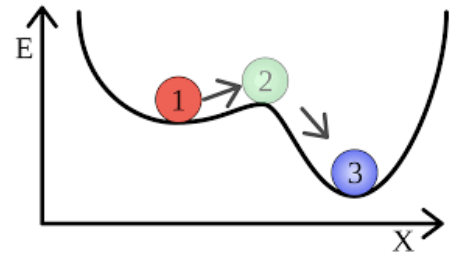
Metastabil sistemler uzun süreli kararlı bir yapı gösterse de uzun bir zaman geçtiğinde sistemin hata vermesi kaçınılmazdır (Şekil 1). Bu tarz sistemlere robotikten çok sayıda farklı örnekler verilebilir. Örneğin bacaklı robotların dinamiği metastabil sistemlere örnek olup sistemde bulunan farklı parametrelerin rastgeleliğinden dolayı (ayağın yere çarpma esnasındaki üzerine binen anlık etkiler, zeminin düzensiz oluşu, ...) analizi zor bir hale getirmiştir. Bu tarz sistemleri analiz etmekteki temel sıkıntının sistemin olasılıksal yapısı olduğundan sistemi Markov zincirine çevirme kolaylık sağlayabilmektedir.

Markov zincirine dönüştürme sonucunda ortaya çıkan geçiş matrisi, sistemin metastabil yapısı ile ilgili bilgiler barındırır. Bu bilgilerden biri ilk geçiş zamanıdır ve sistemin tanımlanan hata tabirine göre hatanın gerçekleştiği zamana kadar geçen süre olarak tanımlanmıştır. Son zamanlarda, ilk geçiş zamanı, pasif yürüyüşçü [2], haptik geri bildirim ile top sektirme [1]

Türkiye Robotbilim Konferansı, 2018

ve ters sarkaç dengeleme ([7]-[8]) çalışmalarında bir performans metriği olarak uygulanmış ve faydalı sonuçlar ortaya koymuştur. Bu faydalı sonuçlar, ilk geçiş zamanının robotik bilimindeki çoğu çalışmaya uygulanabilirliği konusunda bizim temel motivasyon kaynağımız olmuştur.

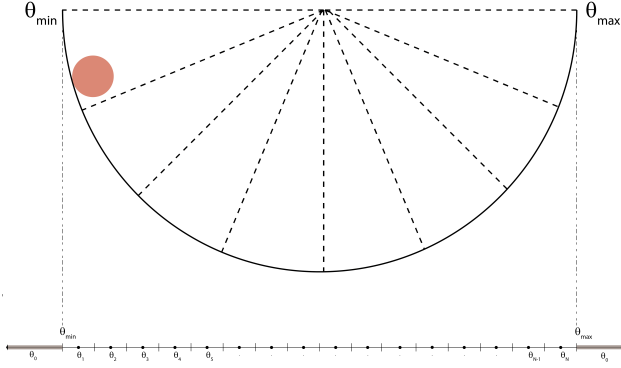
Byl ve Tedrake (2009) [2] çalışmalarında, ilk geçiş zamanının sistemin öz mod analizini yaparak, sistemin ikinci en büyük öz değeri ile bir ilişki içinde olduğunu ortaya koymuşlardır. Ancak Byl ve Tedrake (2009) [2] ilk geçiş zamanını bulmak için sistemin modeli odaklı bir çözüm bulmuş ve tam modeli olmayan herhangi bir sistemin metastabilite analizi konusunda çözümleri yoktur. Biz çalışmamızda sistemin ilk geçiş zamanını veri-güdümlü bir şekilde tahmin ederek, verileri toplanabilen kararlı bir limit döngü etrafında hareket eden herhangi bir sistemin metastabilite analizini yapmaya çözüm bulduğumuzu düşünüyoruz. Yaklaşımımızı anlatabilmek için, basit bir metastabil sistem oluşturduk ve sistemi Markov zincirine çevirmek için sezgisel olarak sistemin her başarılı bitirdiği turda çıkabildiği maksimum açığa göre sistemi kesikli hale getirdik. Sonra ortaya çıkan kesikli hale getirilmiş sonlu Markov zincirinin geçiş matrisinden ortalama ilk geçiş zamanını hesapladık. Bu metodu Monte Carlo deneyleri yaparak, hesapladığımız ortalama ilk geçiş zamanlarını farklı parametrelere göre analiz ettik ve hangi parametrelerin veri-güdümlü tahmin etme metodu için daha yararlı olacağına karar verdik. Bu çalışma da kullandığımız stokastik sistemin tanımlanması ve ayrıntıları Bölüm II'de, Monte Carlo deneyleri Bölüm III'de, simülasyon sonuçları Bölüm IV'te, çıkarım ve gelecekte çalışmayı nasıl geliştireceğimiz ise Bölüm V'da anlatılmıştır.



Şekil 1. Metastabil sistemler, uzun süreli kararlı yapılar gösterse de yeterli zaman geçtikten sonra hata vermeleri kaçınılmazdır. Sistem, 1 durumunda küçük çapta gürültüler için kararlı iken yeterli zaman geçtikten sonra gürültüler sonucu 3 durumuna geçmesi kaçınılmazdır. (Fotoğrafın kaynağı [9])

II. STOKHASTİK SİSTEM

A. Sistem



Şekil 2. Sistemin genel görselleştirilmesi için oluşturulmuştur. Görüldüğü gibi sistem $[-\pi/2, \pi/2]$ aralığındaki bir çanakta ve ağırlığı veyarçapı olan 2 boyutlu bir toptan oluşmuştur. Boyuna kesikli çizgiler sistemin nasıl kesikli hale getirildiğini görselleştirmek için kullanılmıştır.

Sistemimiz $[-\pi/2, \pi/2]$ aralığındaki yarım çember şeklindeki bir çanakta ve çanak içinde gitmeye çalışan bir toptan oluşuyor (Şekil 2). Topu bu çanağın herhangi bir yerinden serbest bırakıyoruz. Çanakta ilerleyen topumuza, her en alt noktadan geçişinde kontrolçümüz topun hızını:

$$\dot{\theta}^+ = \dot{\theta}^- + K \cdot (\dot{\theta}_{arz} - \dot{\theta}^-) + \eta \quad (1)$$

Denklem 1'e göre hesaplayıp topa uyguluyor. Bu denklem de $\dot{\theta}_{arz}$ sistemin çanak içinde gitmesini istediğimiz noktaya gidebilmesi için gereken açısal hızı, K hatayı çarptığımız oran katsayısını ve sistemin öz değerini, η sisteme eklenen $N(\eta, \sigma)$ yapısındaki Gauss dağılımına sahip gürültüyü temsil eder. Topun bu hareket serisi top çanakta dışarı çıkana kadar devam eder. Farkedildiği gibi sistemin deterministik bir şekilde kararlı olabilmesi için K nın yani öz değerini 1 den küçük olması gerekir. Sistem parametreleri kontrolcü, genel ve gürültü parametreleri olmak üzere üç ayrı parametre listesinden oluşur. Her bir parametre birimi ve açıklaması Tablo I'de verilmiştir.

Tablo I. GENEL SİSTEM PARAMETRELERİ

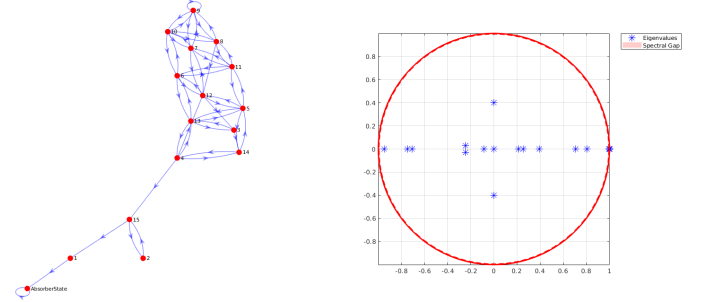
	Parametreler	Açıklama
Kontrolcü Parametreleri	$K = 0.01$ $arz = 0.4072 rad/sn$ $\theta_{init} = \pi/8$	Oransal Katsayısı Arzu edilen açısal hız Topun çanakta dönmeye başladığı açı
Genel Parametreler	$r = 10$ m $\rho = 1$ m $g = 9.8$ m/s ² $m = 1$ kg	Çanağın yarıçapı Topun yarıçapı Yerçekimi ivmesi Topun kütlesi
Gürültü Parametreleri	$\sigma = 0.05$ $\mu = 0$	Gauss dağılımı gürültünün standart sapması Gauss dağılımı gürültünün ortalama gradyenti

Byl ve Tedrake (2009) [2], Sağlam ve Byl (2015) [3] bacaklı robotların stokastik çetin arazilerdeki yürüyüşünü anlatmak için metastabiliteyi kullanmış ve metastabiliteyi, stokastik dinamik sistemlerin, uzun bir zaman dilimi içinde kararlı bir

davranış göstermesi özelliği olarak tanımlamışlardır. Ancak bu sistemlerin gürültülü yapısından dolayı yeterince zaman geçtikten sonra hata vermesi kaçınılmaz olduğunu da belirtmişlerdir. Bizim sistemimizde de bir topun çanak içinde ilerlemesi uzun bir zaman diliminde kararlı bir yapı gösterse de, kontrolçünün çıktılarının gürültü de içermesinden dolayı hata vermesi kaçınılmazdır.

B. Sonlu Markov Zinciri ve Metastabilite ile Bağlantısı

Bir Markov zinciri olası olayların sıralamasını açıklayan stokastik bir modeldir ve her olayın olasılığı sadece bir önceki olayda elde edilen duruma bağlıdır. Çanakta ilerleyen top örneğini düşündüğümüzde topun bir sonraki döngüde çıkabileceği açı, kontrolçünün etkisi haricinde sadece bir önceki çıktığı maksimum açıya bağlıdır. Byl ve Tedrake (2009) [2], Sağlam ve Byl (2014) [4] sonlu Markov zincirine kesikli hale getirmek için sistemin modeline göre oluşturdukları bir ağı(mesh) kullanmıştır. Bu metodun sistemin modelinin olmasına bağlı kalması ve kompleks bir algoritma olması bizi başka türlü bir kesikli hale getirme metoduna yönlendirmiştir. Biz çalışmamızda, Ankaralı ve diğ.'nin [1] Markov zincirini kesikli hale getirmek için izlediği veri-güdümlü metodu biraz daha geliştirerek, sistemin hata vermediğinde bile oluşan bilgiyi kullanılır hale getirdik. Sistemi nasıl kesikli hale getirdiğimiz Şekil 2'de gösterilmiş ve Bölüm II-C'de anlatılmıştır. Stokastik sistemimizin Markov zincirine döndürülmüş hali Şekil 3'de gösterilmiştir. Markov zincirinin soğurucu duruma (Absorber State) geçtikten sonra hiç bir şekilde oradan başka duruma geçmemesi metastabilite dağılım gösterdiğini desteklemektedir.



Şekil 3. Stokastik sistemin bir deneyi sonucu toplanan veriye göre oluşturulan parametreleri bilinmeyen Markov Zinciri ve Öz Değerler Grafiğini gösterir. Sistemdeki toplam Markov zinciri sayısı, Poincare ve Markov uzayını kestiğimiz kesit sayısına eşittir.

C. Kesikli Hale Getirme Algoritması

Sistemin her bir deneyi gerçekleştiğinde sistemin başarılı bitirdiği döngüdeki ulaştığı θ açılarından bir dizi oluşturulur ve kesikli hale getirme bu veriye uygulanır. Kesikli hale getirme işlemi 3 parçadan oluşur. Algoritma 1'de de anlatıldığı gibi ilk olarak sistem alt uzay sayısına göre sınıflandırılır, sonra sınıflandırılan dizide ardışık ikili değerlere bakılarak Algoritma 2'deki gibi geçiş matrisi oluşturulur. Son olarak geçiş matrisinin her bir sırası Algoritma 3'teki gibi normalize edilir. Algoritmalar psödo kod şeklinde açıklamalarıyla birlikte anlatılmıştır.

Algoritma'da,

- 1) Topun her başarılı şekilde bitirdiği yuvarlanma döngüsündeki gittiği maksimum açılar oluşturduğu dizideki değerler, oluşturulan $\Delta\theta$ dizisinde hangi aralığa karşılık geliyorsa onun dizin değerini alır ve $Norm\theta$ dizisine eklenir.

Algorithm 1: θ 'nın state sayısına göre sınıflandırılması

Input: $[\theta]$, Durum miktarı
Output: $[\theta]$ nin state sayısına göre sınıflandırılmış hali

```

1  $s \leftarrow Durum$ ;
2  $\Delta\theta \leftarrow -\pi/2 : \pi/s : \pi/2$ ;
3  $Norm\theta \leftarrow zeros(1, uzunluk([\theta]))$ ;
4  $T \leftarrow zeros(s + 1, s + 1)$ ;
5 for  $i \leftarrow 1$  to  $uzunluk([\theta])$  do
6   for  $j \leftarrow 1$  to  $s$  do
7      $min\theta = \Delta\theta(j)$ ;
8      $max\theta = \Delta\theta(j + 1)$ ;
9     if  $\theta(i) \geq min\theta \wedge \theta(i) \leq max\theta$ ; then
10    |  $Norm\theta(i) \leftarrow j$ 
```

- 2) Geçiş matrisini oluşturmak için, $Norm\theta$ dizisinde baştan başlayarak o anki indekste bulunan değer ve sonraki indeksteki değerlere bakılır. Bu değerler geçiş matrisinin indeksleri olduklarından çıkan değerlere göre Geçiş matrisinin o indekslerdeki değeri 1 artırılır.

Algorithm 2: Sınıflandırılmış θ dizisine göre Bağlantı Matrisinin oluşturulması

Input: $Norm\theta$, Durum
Output: $Norm\theta$ 'e göre oluşturulmuş bağlantı matrisi

```

1 for  $i \leftarrow 1$  to  $uzunluk([\theta]) - 1$  do
2   if  $Norm\theta(i) == 0 \wedge Norm\theta(i + 1) == 0$  then
3     |  $continue$ ;
4   else
5     |  $T(Norm\theta(i) + 1, Norm\theta(i + 1) + 1) ++$ ;
```

- 3) Son olarak Geçiş matrisinin her bir sırasının toplamı 1 olacak şekilde bütün sıralar normalize edilir.

Algorithm 3: Bağlantı dizisinin normalize edilmesi, (T : Bağlantı matrisi)

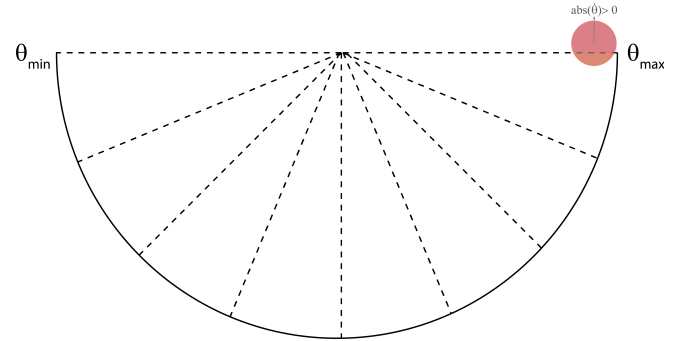
Input: T
Output: T_{norm}

```

1 for  $i \leftarrow 1$  to  $s + 1$  do
2   if  $sum(T(i, :)) = 0$  then
3     |  $T_{norm}(i, :) = T(i, :)/sum(T(i, :))$ ;
4  $T(1, 1) \leftarrow 1$ ;
5  $T(1, 2 : end) \leftarrow 0$ ;
6  $T_{norm} \leftarrow T$ ;
```

III. MONTE CARLO DENEYLERİ

Metastabil sistemlerin kararlılık analizi için doğrusal metotları kullandığımızda sistemin yapısı ile ilgili düzgün bir bilgi vermemekle birlikte, uygun bir analiz metodu olmamıştır. Bu yüzden metastabil sistemler için farklı analiz metotları aranmış ve ortalama ilk geçiş zamanı, sistemin ilk çalışmaya başladığından hata verene kadar geçen süre, sistem için mantıklı bir performans ölçütü olmuştur. Önceden de bahsettiğimiz gibi Byl ve Tedrake (2009) [2] bu performans ölçütünü bacaklı robotlar için, Ankaralı ve diğ. [1] haptik geri bildirim ile top sektirme için uygulamış ve elle tutulur sonuçlar bulmuşlardır. Her metastabil sistemin hata tanımı farklılık gösterdiğinden, biz de kendi sistemimizin hata tanımını çanakdan dışarı çıkması olarak karar verdik (Şekil 4) ve parametreleri buna göre hesapladık. Değerlendirme metrikleri simülasyonlarda çıkan sonuçların neye göre analiz edildiğini göstermektedir ve odaklandığımız parametreler, direkt OİGZ tahmini (Bölüm III-A1) her bir deney sonucu çıkan sonuca göre hesaplanırken, geçiş matrislerinin toplamının OİGZ si (Bölüm III-A2a) ve son olarak geçiş matrislerinin toplamının OİGZ vektörü (Bölüm III-A2a) bütün deneyler bittikten sonra bütün geçiş matrislerinin toplamının sonucuna göre hesaplanır.



Şekil 4. Stokastik sistemler için hata tanımı için çok önemli olup sistemin soğurucu durumunun dinamiğini etkilemektedir. Bu çalışmada, soğurucu durum topun çanak içinden çıkması olarak tanımlanmıştır.

Deneylerimizde sistemin hata vermemesi durumundaki veriyi kullanışlı hale getirmek için yapay bir hata değeri tanımladık. Çanaktaki topun başarılı bitirdiği toplam döngü sayısı belirli bir değeri geçince bunu bir hata durumu olarak tanımladık. Bu işlemi 100,1000 olmak üzere iki ayrı değere göre yaptık. Yapay hata 100 olduğunda, verinin düşük bir değerde nasıl etkileneceğini gösterilirken, 1000 olduğunda da sistemin hiç yapay hata tanımlanmadan oluşacak durumu benzetimlemek için kullanılmıştır. Ayrıca bu yapay hata değerinin 1000 olduğunda, sistemin daha az hesapsal masraf harcadığı gösterilmiştir. Yapay hata değerinin 1000'e eşit olduğu durumda yapay hata tanımlanmadan oluşacak duruma benzediğini de göstermiş bulunmaktayız.

A. Değerlendirme Parametreleri

1) *Direkt OİGZ (Ortalama İlk Geçiş Zamanı) Tahmini:* Bölüm II'de anlatıldığı gibi sistemin her başarılı bitirdiği turdaki çıktığı açılardan oluşan dizi sistemin Markov zincirine çevrilmesi için kullanılmıştır. Başarılı bitirdiği toplam döngü

sayısı sistemin direkt OIGZ tahmini olarak adlandırılmıştır. Simülasyon sonuçlarında da göstereceğimiz gibi, direkt OIGZ tahmini kapsamlı performans metriği olsa da hatasız biten deney sonuçlarında oluşan veri dizisinden fazlasıyla etkilenmesi bizi başka parametrelere yönlendirmiştir. Ancak hata, herhangi bir döngü sayısı limiti konulmadığında bu parametrenin sistemin kendi başarı ölçütü olarak ortaya çıkar.

2) *Geçiş Matrislerin Toplamına Göre:* Buradaki iki parametre de bütün deneylerin toplamı sonucu olduğundan, sezgisel olarak metastabil dağılımı daha iyi yansıttığını düşündük. Ayrıca sistem hata vermese de o deneydeki diziyi de veri olarak kullanmamız buradaki parametrelerin daha yararlı ve daha kapsamlı olmalarını sağlamıştır.

a) *OIGZ:* Her bir deney sonucu oluşturulan diziye, Algoritma [1-2]'deki işlemler uygulanır ve geçiş matrislerinin normalize edilmemiş hali ortaya çıkar. Her deney sonucu çıkan bu geçiş matrisleri toplanıp, sonra Algoritma 3'teki normalizasyon uygulanır. Oluşan geçiş matrisi, Ankaralı ve diğ.'nin [1] öngördüğü şekilde bulunmuştur ve matematiksel yapısı, Denklem 2'te gösterildiği gibidir.

$$T = \begin{bmatrix} 1 & 0 \\ r & \bar{T} \end{bmatrix} \quad (2)$$

Denklem 2'de kullanılan \bar{T} , geçiş matrisinin soğurucu durumlarının çıkarılmış hali olup (T 'nin ilk sıra ve sütununun kaldırılmış hali), ayrıca kullanılan parametrelerin buldukları uzaylar ise,

- $T^{n \times n} \in R^{n \times n}$
- $\bar{T}^{(n-1) \times (n-1)} \in R^{(n-1) \times (n-1)}$
- $r^{(n-1) \times 1} \in R^{(n-1) \times 1}$
- $0 \in R^{1 \times (n-1)}$ şeklindedir.

Ortaya çıkan geçiş matrisinin en büyük ikinci öz değeri hesaplanır. OIGZ, Denklem 3'e göre hesaplanır.

$$oigz = \frac{1}{1 - \lambda_2} \quad (3)$$

b) *OIGZ Vektörü:* Geçiş matrisi III-A2a'deki gibi oluşturulur. Oluşan geçiş matrisinin yapısı Denklem 2'deki gibidir. Ankaralı ve diğ. [1], metastabil dağılımının güvenilirliğini test etmek için, soğurucu durum haricindeki her bir durumdan başladıklarında ortaya çıkacak ortalama ilk geçiş zamanı vektörünün yararlı olacağını düşünmüş ve bu vektörü Denklem 4'e göre bulmuşlardır. Biz de bu vektörü, sistemimiz için bir performans metriği olarak kabul ettik. bu değeri hesaplamak için, ilk başta bulduğumuz vektörün başlangıç değerine göre karşılık gelen indeksini hesapladık ve sonra bu indekse tekabül eden değeri parametremiz olarak belirledik. Bu parametrenin çalışma içindeki adı **Geçiş matrislerinin toplamından oluşan OIGZ vektörü** dür.

$$\begin{aligned} \mu_{oigz} &= (1 - \bar{T})^{-1} \mathbf{1} \\ \mu_{oigz} &= [\mu_{oigz,2} \quad \mu_{oigz,3} \quad \dots \quad \mu_{oigz,N}]^T \end{aligned} \quad (4)$$

B. Monte Carlo Deneyleri

Monte Carlo yöntemleri sayısal sonuçlar elde etmek için tekrarlanan rastgele örneklemeyle dayanan geniş bir hesaplama algoritması sınıfıdır. Biz bu çalışmamızda sonuçlarımızı test etmek için kısmi bir Monte Carlo metodu olan Ön yükleme (*Bootstrapping*) denilen istatistiksel ölçüm yöntemini kullandık.

Algorithm 4: Önyükleme metodunun uygulanma algoritması

Input: $[data]$, $[Tsum]$, $OrneklemeSayIsi$, $DeneySayIsi$

Output: Deney sayısı uzunluğunda, Örnekleme sayısı kadar rastlantısal seçilen değerlerin ortalamasından oluşan dizi ($[ToplamData]$)

```

1  $samp \leftarrow DeneySayIsi$ ;
2 for  $i \leftarrow 1$  to  $DeneySayisi$  do
3    $randInd \leftarrow RastgelePermtasyon(samp)$ ;
4    $ToplamData(i) \leftarrow ortalama([data](randInd))$ ;

```

Ön yükleme metodunu Algoritma 4 anlatıldığı gibi kullandık ve örnekleme sayımızı verimizin altında tuttuk. Verilerin analizini daha gerçekçi kılabilmek için, örnekleme işlemini her bir örnekleme sayısına göre deneyimizi bin kere tekrar ederek gerçekleştirdik. Örnekleme sayılarımız 5^n n'nin [1,6] katsayıları şeklindedir. Bölüm II-A'da başarılı gidilen döngü miktarını 100 ile sınırlandırdığımızda, bu sınırlandırmanın direkt OIGZ tahmin metodunu fazlasıyla etkilediğini ve bunun nedeninin direkt OIGZ tahmininin doğrudan başarılı döngü sayısına eşit olmasından dolayı kaynaklandığını düşünüyoruz. Diğer parametrelere baktığımızda bundan etkilenmemiş olduklarını böylelikle hata vermediğinde bile oluşacak veriyi kullanmak için yararlı bir parametre olabileceğini göstermiştir. Deneylerin sonuçlarını gösterirken, direkt OIGZ tahminini ve Geçiş Matrislerinin Toplamının OIGZ vektörünü parametreleri kullandık. Bunu yapmamızın nedeni Geçiş Matrislerinin Toplamının OIGZ vektörünün hep aynı başlangıç noktasından başladığımızdan dolayı Geçiş Matrislerinin Toplamının OIGZ si ile yakın çıkacaklarını ve fazladan figürle sonuçlarda asıl odaklanmak istediğimiz noktayı kaçırmamak istedik.

IV. SİMÜLASYON SONUÇLARI

Bölüm III'te bahsettiğimiz değerlendirme parametrelerini analiz edebilmek için, simülasyon sonuçlarında her bir parametrenin ön yükleme metoduna göre farklı örnekleme sayımları için dağılımına odaklandık. Sonuçları iki ayrı parçaya ayırdık ve bunlar, parametrelerin yapay hata limitine göre etkilenmeleri ve Geçiş Matrislerinin Toplamının OIGZ vektörünün yapay hata tanımladığımızda gerçek başarı ölçütü olan direkt OIGZ tahmini ile karşılaştırılmasıdır.

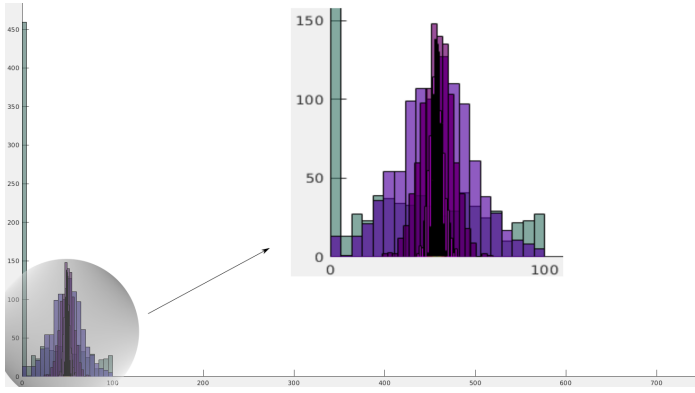
A. Parametrelerin Karşılaştırılmasının Yapay hata Limitine Göre Yapılması

Tablo II'deki değerleri oluşturmak için, bölüm III'te anlatılan deneyler yapılmış ve sonuç olarak ortaya çıkan parametreler sunulmuştur. Parametreler, örnekleme sayısına ve tanımlanan maksimum döngü sayısına göre ayrı ayrı hesaplanmıştır. Bu deneylerde, oluşturulan veriler 8 kesite göre kesikli hale getirilmiştir. Ayrıca Tablo II'den de görülebileceği gibi, yapay

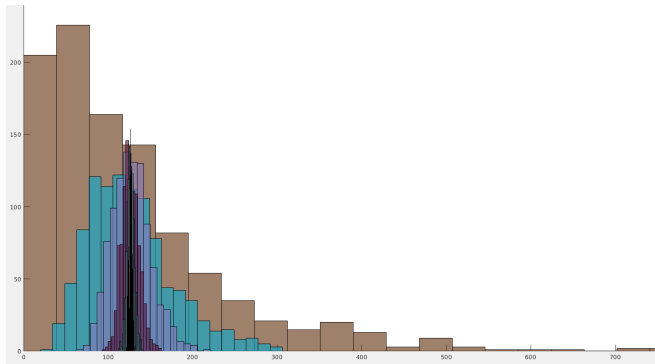
Tablo II. MONTE CARLO DENEYİ SONUÇLARI

Parametre Adı	Örnekleme Sayısı	Maksimum toplam döngü sayısı	Değer
Direkt OİGZ Tahmini	5	100	49.7232
	125	100	49.6790
	15625	100	49.7115
	5	1000	127.4407
	125	1000	123.0966
	15625	1000	122.8210
Geçiş Matrislerinin Toplamına Göre OİGZ Vektörü	5	100	126.2605
	125	100	126.3415
	15625	100	126.4311
	5	1000	131.0338
	125	1000	126.7358
	15625	1000	126.4481

hata tahminini 100 ve 1000 yaptığımızda, direkt OİGZ tahmini çok farklılık göstermiştir. Önce dende söylediğimiz gibi bunun nedeni direkt OİGZ tahminin sistemin toplam döngü sayısına eşit olmasıdır. Ancak diğer parametreye baktığımızda bu durumdan etkilenmemesi, hata vermediğinde bile oluşan veriyi kullanmamızın sistemi etkilemeyeceğini aksine metastabil dağılımı geliştirdiğini ortaya koymuştur. Yapay hata tahmini 1000 değerine göre yaptığımızda Tablo II ve Şekil 7'dan da görebileceğimiz gibi direkt OİGZ tahmini diğer parametrelere daha çok yaklaşmış ve daha düzgün bir dağılım göstermiştir.

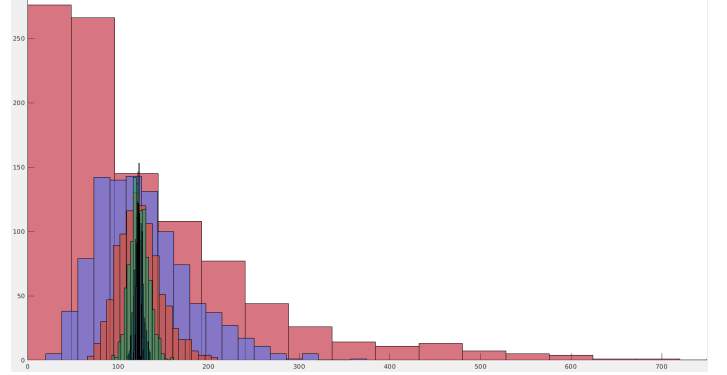


Şekil 5. Deney sonuçlarına, ön yüklemeyi farklı örneklem sayıları uyguladığımızda direkt OİGZ Tahmini parametresinin nasıl değiştiğini gösterir. Bu şekilde Yapay hata tespiti 100 döngü olarak tanımlanmış ve deney ona göre yapılmıştır.

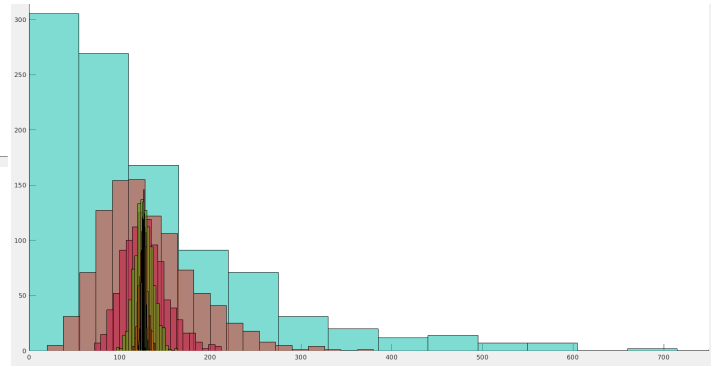


Şekil 6. Deney sonuçlarına, ön yüklemeyi farklı örneklem sayıları uyguladığımızda Geçiş matrisi toplamının OİGZ'ının Vektörü parametresinin nasıl değiştiğini gösterir. Bu şekilde, Yapay hata tespiti 100 döngü olarak tanımlanmış ve deney ona göre yapılmıştır.

Şekil 6 ve 8 baktığımızda aralarında çok fark olmaması geçiş matrisi toplamının OİGZsi vektörünün yapay hata tespitimizden çok etkilenmediğini ve hata vermese de oluşan veriyi yararlı bir şekilde kullanabileceğimizi göstermiştir. Şekil 7 ve 8 karşılaştırdığımızda ise yapay hata tespiti 1000 değerine göre yapıldığında dağılımın yakın olduğunu ve Geçiş Matrisi Toplamının OİGZsi vektörünün sistemin ne zaman hata vereceğini yaklaşık olarak yakın bir şekilde bulduğu görülmüştür. Direkt OİGZ tahminin dağılımına baktığımızda yapay hata tespitini belirleyen değerlerin artmasının, sistemin üzerindeki sapmayı azaltarak sistemin metastabil dağılımı üzerinde daha düzgün bilgi vermesini sağlamıştır (Şekil 5 ve 7).



Şekil 7. Deney sonuçlarına, ön yüklemeyi farklı örneklem sayıları uyguladığımızda direkt OİGZ tahmini parametresinin nasıl değiştiğini gösterir. Bu şekilde Yapay hata tespiti 1000 döngü olarak tanımlanmış ve deney ona göre yapılmıştır.



Şekil 8. Deney sonuçlarına, ön yüklemeyi farklı örneklem sayıları uyguladığımızda Geçiş matrisi toplamının OİGZ'ının Vektörü parametresinin nasıl değiştiğini gösterir. Bu şekilde, Yapay hata tespiti 1000 döngü olarak tanımlanmış ve deney ona göre yapılmıştır.

V. ÇIKARIM VE GELECEK İŞLER

Bu çalışma kapsamında, ritmik robotik sistemlerin metastabilite performansını ölçebilmek için deneysel/veri güdümlü bir yöntem önerdik. Yaklaşımımız öncelikle seçilen bir Poincare kesit uzayını kesikli hale getirir ve daha sonra bu kesit içindeki elemanların birbirine olan bağlantısını/geçişini stokastik bir Markov zinciri olarak modeller. Benzer uygulamalarda bu Markov zincirinin parametreleri bilindiği varsayılan sistem modelinin parametreleri kullanılarak ve sonlu elemanlara benzer analizler ile hesaplanmıştır. Bizim çalışmamızda model

ile ilgili radikal varsayımlar yapmadan, doğrudan ölçülen veriler üzerinden istatistiki bazı çıkarımlar ile Markov zinciri oluşturulur. Bu Markov zincirinde yapısal olarak ilk durum soğuran yani hata durumdur. Bu bilgi kullanılarak öz değer analizi yaparak veya çeşitli matris operasyonları ile OIGZ hesaplanır/tahmin edilir. Yaklaşımımızı test etmek ve literatürdeki başka bir istatistiksel yöntem ile karşılaştırmak için Monte Carlo deneyleri gerçekleştirdik.

Bu benzetimler sonunda gördük ki eğer robotik sistem “çok uzun” süreler deney yapmaya elverişli ise, literatürdeki yöntem ile bizim yaklaşımımız benzer sonuçlar verdi. Fakat robotik sistem uzun süreler deney yapımına izin vermiyorsa, ki bu çok muhtemel bir durumdur, var olan yöntemin yapısal kısıtlarından dolayı istatistiki olarak düşük değerlere meyilli (*biased*) olduğunu fakat bizim yöntemimizin daha gürbüz olduğunu gözlemledik.

Yaklaşımımızda, veri-güdümlü bir metot kullanmamız farklı robotik sistemlere de uygulanabilirliği konusunda gelecek vaat eden bir performans metriği olduğu konusunda bizi motive etmiştir. Ayrıca sistem hata vermese de oluşacak veriyi sistemde kullanabiliyor olmamız, daha az deneyle daha hızlı bir şekilde sistem hakkında bir analiz edebileceğimizi de ortaya koymuştur.

Robotikteki çoğu problemin metastabil yapısı içinde oluşu, bizi bu metriği çok uzaylı bacaklı robotlara uygulama konusunda motive etmiştir. Bir sonraki aşamada RHex robotu [6] baz alınarak geliştirilmiş model [5] üzerinde bu yöntemi geliştirip test etmeyi planlıyoruz. Daha sonra OIGZ metriğini istemi/kontrolcülerini optimize etmek için bir masraf değeri olarak kullanmayı planlıyoruz.

VI. TEŞEKKÜR

Tartışmalarıyla konuyu daha ayrıntılı bir şekilde irdelememizi sağlayan Uluç Saranlı ya teşekkür ederiz.

KAYNAKÇA

- [1] Ankarali, M. M., Tutkun Sen, H., De, A., Okamura, A. M., Cowan, N. J. (2014). Haptic feedback enhances rhythmic motor control by reducing variability, not improving convergence rate. *Journal of Neurophysiology*, 111(6), 1286–1299. <https://doi.org/10.1152/jn.00140.2013>
- [2] Byl, K., Tedrake, R. (2009). Metastable Walking Machines. *The International Journal of Robotics Research*, 28(8), 1040–1064. <https://doi.org/10.1177/0278364909340446>
- [3] Saglam, C. O., Byl, K. (2015). Metastable legged locomotion: methods to quantify and optimize reliability. *Proceedings SPIE 9467, Micro- and Nanotechnology Sensors, Systems, and Applications VII*, 9467(May 2015), 94671T. <https://doi.org/10.1117/12.2177102>
- [4] Oguz Saglam, C., Byl, K. (2014). Robust Policies via Meshing for Metastable Rough Terrain Walking. In *Robotics: Science and Systems X*. Robotics: Science and Systems Foundation. <https://doi.org/10.15607/RSS.2014.X.049>
- [5] Ankarali, M. M., Saranlı, U., Saranlı, A. (2010). Control of underactuated planar hexapedal pronking through a dynamically embedded SLIP monopod. *Proceedings - IEEE International Conference on Robotics and Automation*, 4721–4727. <https://doi.org/10.1109/ROBOT.2010.5509450>
- [6] Saranlı, U., Buehler, M., Koditschek, D. E. (2001). RHex: A Simple and Highly Mobile Hexapod Robot. *Int. J. of Robotics Research*, 20(7), 616–631.
- [7] Milton JG, Cabrera JL, Ohira T, Tajima S, Tonosaki Y, Eurich CW, Campbell SA. The time-delayed inverted pendulum: implications for human balance control. *Chaos* 19: 026110, 2009a.
- [8] Milton JG, Ohira T, Cabrera JL, Fraiser RM, Gyorffy JB, Ruiz FK, Strauss MA, Balch EC, Marin PJ, Alexander JL. Balancing with vibration: a prelude for “drift” and “act” balance control. *PLoS One* 4: e7427, 2009b

- [9] By Georg Wiora (Dr. Schorsch) (self made drawing) [GFDL (<http://www.gnu.org/copyleft/fdl.html>), CC-BY-SA-3.0 (<http://creativecommons.org/licenses/by-sa/3.0/>) or CC BY-SA 2.5-2.0-1.0 (<https://creativecommons.org/licenses/by-sa/2.5-2.0-1.0/>), via Wikimedia Commons