

Ölü Zamanlı Sistemlerde kullanılan Smith Öngörücüsü için Balina Sürüsü Optimizasyonu ile PID Tasarımı

*¹Murat Erhan Çimen, ¹Ömer Faruk Boyraz, ¹Muhammed Ali Pala,
¹Ali Fuat Boz, ¹Mustafa Zahid Yıldız

¹Faculty of Technology, Department of Electrical and Electronic Engineering, Sakarya University of Applied Sciences, Turkey

Özet:

Bu çalışmada balina sürüsü algoritması kullanılarak ölü zamanlı sistemlerin denetlenmesinde kullanılan Smith Öngörücüsü için PID denetleyicisinin parametreleri belirlenmiştir. Ayrıca Smith Öngörücüsü için kullanılan Zeigler-Nichols veya AMIGO yöntemleri ile yapılan PID tasarımında kontrol işareti sınırlandırılmamaktadır. Bu kısıt aşımı ise kontrol işaretini oluşturan aktüatöre zarar verebilmektedir. Bu çalışmada balina sürüsü algoritması kullanılarak kontrol işaretinin sınırlandırılması yapılarak ölü zamanlı sistemler için ISE performans kriterine göre Smith Öngörücülü PID denetleyici tasarımı yapılmıştır. Yapılan denemeler ile balina sürüsü algoritması ile tasarlanan PID denetleyicisinin daha başarılı olduğu görülmüştür.

Anahtar Kelimeler: Balina sürüsü optimizasyonu, Smith öngörücüsü, PID

PID Design with Whale Optimization Algorithm for Smith Predictor used in Dead Time Systems

Abstract:

In this study, parameters of PID controller for Smith predictor used to control time delay systems were determined using whale optimization algorithm. Furthermore, the control signal is not constrained in the PID design by Ziegler-Nichols or AMIGO methods used for Smith Predictor. This constraint violations can damage the actuator. PID controller was designed with Smith predictor according to ISE performance criteria for time delay systems by limiting the control signal by using the whale optimization algorithm. Experiments have shown that the PID controller designed with the whale optimization algorithm is more successful.

Key words: Whale optimization algorithm, Smith predictor, PID

1. Giriş

Zaman gecikmeli sistemler endüstride en çok karşılaşılan sistemlerdir. Bu tip sistemlerin yapısı itibari ile sisteme bir giriş işareti uygulandığında bu giriş belirli bir zaman sonra sistemin çıkışında gözlemlenmektedir. Bu tip sistemler zaman gecikmeli sistemler olarak isimlendirilmektedir. Zaman gecikmeli sistemlerin sistemin yapısı itibari ile zaman gecikmesi sabit olabileceği gibi durumlara bağlı olarak da değişebilmektedir. Bu tip sistemlerin kontrolünde klasik PI yada PID

*Corresponding author: Murat Erhan ÇİMEN Address: Faculty of Technology, Department of Electrical and Electronic Engineering Sakarya University of Applied Sciences, 54187, Sakarya TURKEY. E-mail address: muratcimen@subu.edu.tr, Phone: +9026461650279

denetleyicilerde kullanılabilir. Ama bu tip denetleyiciler uygulandığında sistem yavaş cevap verebilmektedir. Ayrıca zaman gecikmeli sistemler için PID denetleyici türev operatöründen dolayı kontrol işareti uygun olmayabilmektedir. Bu nedenle türev operatörü bu tip denetleyicilerde düşük tutulmaya çalışılmaktadır. Bununla birlikte referans işaretinin hızlı değişen sistemlerde yine kontrol işareti açısından problem oluşturulabilmektedir. Ayrıca zaman sabiti düşük ama ölü zamanı yüksek olan sistemlerdeki ani değişimlerde de yüksek kontrol işaretine neden olabilmektedir. Ayrıca sistemin osilasyona girip kararsız olmasına neden olabilmektedir. Bununla birlikte endüstrideki uygulamalarda kısıtlı aktuatörlerden ötürü çok uygun olmamaktadır. Bu nedenle ölü zamanlı sistemlere uygulanabilecek olan ölü zamanını hesaba katılarak kontrol işaretinin uygulanmasını sağlayan smith öngörücüsü önerilmiştir. Smith'in önermiş olduğu denetleyici yapısı ile uzun ölü zamanlı sistemlerin denetlenmesinde ölü zamanı efektif bir şekilde kompanze edebilmektedir [1]. Bu yapı zaman gecikmesini öngörmekte ve sisteme ona göre kontrol işareti uygulamaktadır. Bu denetleyici yapısı geliştirilerek farklı yapılarda da kullanılabilir.

Doğadaki canlı veya cansız varlıkların davranışlarından esinlenilerek geliştirilen sezgisel algoritmalar çözülmesi zor olan problemlerin çözümünde kullanılmaktadır. Çünkü doğa kendi içerisinde pek çok karmaşık ve zor problemleri bu davranışı ile en optimal çözümü verecek şekilde çözmektedir. Bu düşünce evrim teorisindeki doğal seçim ile başlayıp genetik algoritmaları, karıncaların yere döktüğü promonları ile karınca kolonisi algoritması, balık ve kuş sürülerinde yiyecek ve avlanmasından esinlenilerek parçacık sürü optimizasyonu, arılarda nektar bulmalarından esinlenilerek arı kolonisi algoritması, ateşböceklerinde ışığa duyarlılıklarından faydalanılarak ateş böceği algoritması, guguk kuşlarında doğaları gereği kuluçka asalaklığından esinlenilerek geliştirildiği gugu kuşu algoritması, kurtların avlanma stratejilerinden gri kurt optimizasyonu ve yine balinaların avlanma stratejilerinden faydalanılarak balina optimizasyon algoritması v.b. algoritmalar geliştirilmiştir. Bu algoritmalar mühendislikten lojistiğe, ekonomiden eğitime, planlamadan sağlığa çok farklı alanlara uygulanabilmektedir. Karşılaşılan bu tip problemler benchmark problem olarak isimlendirilmektedir. Bu tip problemler çözüm uzayı içerisinde genel olarak birden fazla yerel çözümü olan kısıtlı veya kısıtsız çözülmesi zor doğrusal veya doğrusal olmaya problemler olabilmektedir [2-4]. Bu tip sezgisel algoritmalar gradyen veya Hessian matrisi gerektirmediğinden ve sürekli-süreksiz problemlere kolaylıkla uygulanabilmektedir.

Ölü zamanlı sistemlerin denetlenmesinde kullanılan smith öngörücüsü sistemin ölü zamanı ile sistemin ölü zamansız modelini ayırmaktadır [1]. Bundan sonra modelini kullanarak sisteme denetleyici tasarlanmak ve tasarlanan denetleyici sisteme uyarlanarak kontrol işaretini uygulanmasını sağlamaktadır. Poorani ve diğ. yaptıkları çalışmada ısı eşanjörü için klasik PID ve smith öngörücüsü kullanarak karşılaştırma yapmıştır. Smith öngörücüsü kullanıldığında performansın arttığını ifade etmişlerdir [5]. Yücelen ise yine bir ısı sistem için yaptığı çalışmada smith öngörücüsü için PI denetleyici tasarımı gerçekleştirmiştir [6]. Xia ve diğ. ise yapmış oldukları çalışmada yine modifiye edilmiş Smith öngörücüsü kullanarak bulanık PI denetleyici kullanmışlardır. Değişen parametrelerde DC motorun performansı arttığı görmüşlerdir [7]. Normey-Rico ise yapmış olduğu çalışmada ise smith öngörücüsü genelleştirilmiş öngörülü kontrole uyarlamışlardır [8]. Slalov ölü zamanlı bir insülin konsantrasyon sistemi için PID denetleyicinin parametrelerinin belirlenmesi için genetik algoritma kullanmışlardır [9]. Gurban ve arkadaşları ise sera gazı kontrolü için modifiye edilmiş smith öngörücüsü önermişler ve bunun

parametrelerinin belirlenmesi de genetik algoritma kullanmışlardır [10]. Compas ve diğ. ise zaman gecikmeli sistemlerin kontrolünde bulanık mantık tabanlı PD denetleyici tasarımını gerçekleştirmişlerdir. Denetleyicinin parametrelerinin belirlenmesinde PSO kullanmışlardır [11]. Nicola ve diğ. ise yapmış oldukları çalışmada PMSM kontrolü için smith öngörücüsü kullanmışlardır [12]. Guessoum yapmış oldukları çalışmada uzun ölü zamanlı sistemlerin smith öngörücüsü ile kontrolü doğrusal olmayan PID'yi kullanmışlardır [13]. Bu çalışmada ise balina sürüsü optimizasyonu kullanılarak ölü zamanlı sistemler için PID denetleyici tasarımının balina sürüsü optimizasyonu ile ISE kriterine göre yapılması sağlanmıştır. Ayrıca kontrol edilecek olan işaretin algoritma ile kısıtlandırılarak kısıtlı problem çözümü gerçekleştirilmiş ve belirlenen performans değerlerine göre sonuçları karşılaştırılmıştır.

2. Materyal ve Metotlar

Denetleyicilerin tasarlanmasının temelinde PID denetleyici güçlü ve dayanıklı bir denetleyicidir. Fakat ölü zaman arttığı zaman türev katsayısından ötürü klasik PID denetleyici sistem performansı için iyi sonuç veremeyebilmektedir [6]. Bu çalışmada smith öngörücü içerisine yerleştirilen PID tipi bir denetleyicinin Zeigler Nichols, AMIGO ve optimizasyon tekniklerinden olan balina sürüsü optimizasyon tekniği ile tasarlanması yapılmıştır. Balina Optimizasyon Algoritması ile sisteme uygulanacak olan kontrol işareti sınırlandırılarak sistemin zarar görmesi engellenmek istenmiş ve ISE olan performans kriterine bu kısıt olarak girilmiştir. Kısıtlı optimizasyon probleminin çözümü için bariyer fonksiyonu kullanılmış bu sayede kısıt ihlal edildiğinde minimum yapılmak istenen performans değeri çok yüksek değer çıkması sağlanmıştır.

2.1. Balina optimizasyonu algoritması

Balina sürüsü algoritması, Mirjalili ve Lewis tarafından 2016 yılında balinaların avlanma stratejilerinden faydalanılarak geliştirilmiş bir algoritmadır. Özellikle kalbur balinaların davranışları balık sürüsünü çıkarmış oldukları hava kabarcıkları ile yönlendirmelerinden esinlenilmiştir. Şekil 1'de gösterilen şekilde balinaların avlanma davranışları verilmiştir [14].



Şekil 1. Balinaların su kabarcığı yöntemi ile avlanması

Balinaların avının etrafını çember şeklinde sarması eşitlik **Hata! Başvuru kaynağı bulunamadı.** 'de verilmiştir. Denklemdeki $\vec{X}^*(k)$ çözüm uzayında balinanın en iyi konumunu ifade etmektedir. Yani global en iyiyi tanımlamaktadır. \vec{A} ve \vec{C} ise yakınsamayı sağlayan vektörlerdir. Bunlar eşitlik **Hata! Başvuru kaynağı bulunamadı.**'deki gibi hesaplanmaktadır. Buradaki \vec{r} rastgele bir

yönelim vektörü iken \vec{a} ise iterasyon sayısına bağlı olarak azalan bir lineer bir vektördür. Bu parametrik değişkenler sürü içerisindeki her balina için her iterasyonda hesaplanmaktadır [14].

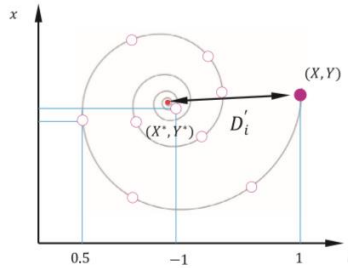
$$\vec{D} = |\vec{D}\vec{X}^*(k) - \vec{X}(k)| \quad (1.1)$$

$$\vec{X}(k+1) = |\vec{X}^*(k) - \vec{D}\vec{A}|$$

$$\vec{A} = 2\vec{a}\vec{r} - \vec{a} \quad (1.2)$$

$$\vec{C} = 2\vec{a}$$

Balina sürüsünde yerel arama balinaların ava spiral şekline veya lineer şekilde hareket etmektedir. Balinanın (X,Y) konumunda olduğu kabul edilirse en iyi konum olan (X*,Y*) 'ye hareketi verilmiştir. Bu davranışın matematiksel olarak modellenmesi ise eşitlik **Hata! Başvuru kaynağı bulunamadı.**'deki gibi modellenebilmektedir. Buradaki $\vec{D} = |\vec{D}\vec{X}^*(k) - \vec{X}(k)|$ 'dir. Yani sürü içerisindeki balinaların en iyi konumu ile sürü içindeki bir balina konumu arasındaki mesafeyi temsil etmektedir. b ise logaritmanın spiral sabiti, l ise -1 ile 1 arasında değer alan rastgele değişen bir değişkendir [14].



Şeki 2. Ava Doğru Hareket

$$\vec{X}(k+1) = \vec{D}'e^{bl} \cos(2\pi l) - \vec{X}^*(k) \quad (1.3)$$

Bu spiral davranış ve lineer davranış ise rastgele bir sayı olan $p \in [0,1]$ 'e göre tercih edilmektedir. Bu ise eşitlik **Hata! Başvuru kaynağı bulunamadı.**'de verilmiştir [14].

$$\vec{X}(k+1) = \begin{cases} \vec{X}^*(k) - \vec{D}\vec{A} & p < 0.5 \\ \vec{D}'e^{bl} \cos(2\pi l) + \vec{X}^*(k) & p > 0.5 \end{cases} \quad (1.4)$$

En iyiden en kötüye yapılan yerel aramanın yanı sıra global aramada bu algoritmada yapılmaktadır. Global aramanın yapılması ise eşitlik **Hata! Başvuru kaynağı bulunamadı.** ile yapılmaktadır. Buradaki \vec{X}_{rand} vektör boyutunca oluşturulan arama uzayı içindeki rastgele bir balınayı göstermektedir. Global arama veya yerel aramalardan hangisinin yapılacağına \vec{A} vektörü karar

vermektedir. Eğer $\vec{A} > 1$ veya $\vec{A} < 1$ olmasına göre en iyi balinadan uzaklaşıp yaklaşmaktadır [14].

$$\vec{D} = \vec{C} \vec{X}^*(k) - \vec{X}(k) \quad (1.5)$$

$$\vec{X}(k+1) = \vec{X}_{rand} - \vec{A} \vec{D}$$

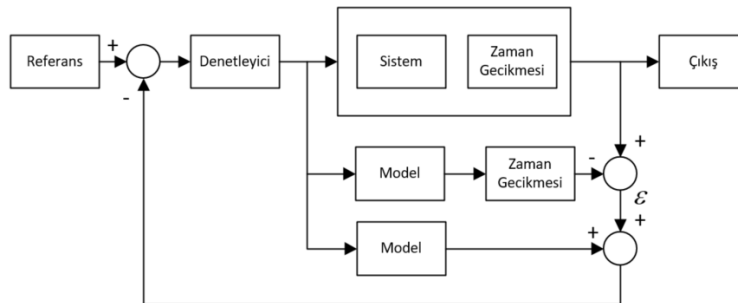
```

Sürüdeki balinalara başlangıç yerlerini rastgele ata
Her bir balinanın uygunluk fonksiyonundaki değerini hesapla
X* en iyiyi belirle
while (durdurma kriteri)
  for i=1:Sürüboyutu
    a,A,C,l ve p parametrelerini güncelle
    if (p<0.5)
      if (|A|<1)
        Denklem 1.4 ile balinaların konumunu güncelle
      else
        Denklem 1.5 ile balinaların konumunu güncelle
      end if
    else if (p>=0.5)
      Denklem 1.3 ile balinaların konumunu güncelle
    end if
  end for
  Balinaların yeni konumlarını karşılaştır ve en iyiyi güncelle
End while
    
```

Şekil 3. Balina Sürüsü Optimizasyonu psodo kodu

2.2. Smith Öngörücüsü

Smith Öngörücüsünün yapısı Şekil 4’de verilmiştir. Burada sistemin girişi ile çıkışı arasında bulunan sistem iki kısma ayrılmaktadır. İlki zaman gecikmesi, diğeri sistemin zaman gecikmesiz modelidir. İlk olarak bu iki kısmın birleşmesi ile oluşan yapı sistemin bir benzeridir. Bu iki yapı ne kadar birbirine benzer ise ε sifıra doğru gitmektedir. Dolayısıyla ölü zaman bu sayede sistemden ayrıştırılmış olmaktadır. Devam edildiğinde de sadece model çıkışı ve ε geri besleme ile referanstan çıkartılarak hata sinyali oluşturulmaktadır. Böylece sadece sistem modeli control edilerek sistem istenen performansı yakalaması sağlanmaktadır. Bu yapının tek dezavantajı parametrelerin birbirine uymasındır. Zaman gecikmesi ve model sistemi iyi bir şekilde temsil etmelidir. Aksi takdirde ε sıfır olmaz ve osilasyonlu bir cevaba neden olunmuş olur.



Şekil 4. Smith Öngörücüsü

2.2.1. Smith öngörücüsü için Zeigler Nichols yöntemi

Zeigler Nichols 1947 yılında farklı sistemler için yaptığı deneme sonucunda klasik P, PI ve PID denetleyiciler için önerdiği iki farklı yöntem bulunmaktadır. Bunlardan ilki sisteme basamak girişi uygulandığından sistemin yanıtına bakılarak sistemin kazancı(K), zaman sabiti (T) ve zaman gecikmesi (L) belirlenir. İkinci yöntem ise kapalı çevrimde sistem oransal bir kazançla denetlenir. Sistem osilasyona girene kadar bu oransal kazanç arttırılır. Sistemin osilasyona girmesini sağlayan bu oransal kazanç kritik kazanç (Kc) iken sistemin osilasyon periyodu belirlenebilmektedir. Bu kritik periyottan (Pu) da kritik osilasyon frekansı (ω_c) belirlenebilir. Bu parametreler kullanılarak tablo 1’den denetleyici katsayıları belirlenmektedir.

Tablo 1. Zeigler Nichols PID Denetleyicisi

Zeigler Nichols PID	K	Ti	Td
Basamak Yanıtına Göre	$\frac{1,2T}{KL}$	$2L$	$\frac{L_u}{2}$
Frekans Yanıtına Göre	$\frac{K_u}{1,7}$	$\frac{P_u}{2}$	$\frac{P_u}{8}$

2.2.2. Smith öngörücüsü için AMIGO yöntemi

Son yıllarda geliştirilen AMIGO yöntemi Astrom ve Hagglund tarafından geliştirilmiştir. Bu teknik Zeigler-Nichols yönteminin revise edilmiş hali de denilebilmektedir. 134 farklı sistem üzerinde yapmış oldukları denemeler sonucunda elde ettikleri sonuçları kullanarak belirli kısıtlar altında denetleyici tasarımı gerçekleştirmişlerdir. Bu tekniğin temelinde kararlılığı arttırılması için duyarlılığın belirledikleri değerlerde olması istenmektedir. Ayrıca integral katsayısını maksimum olacak şekilde denetleyiciler tasarlamaktır. Buna göre önerdikleri yöntemde yine sistemin basamak cevabına göre ve sistemin frekans cevaplarına göre sisteme denetleyici tasarımı yapılmaktadır. Denetleyicinin basamak ve frekans yanıtına yapılan denemelerden elde edilecek olan denetleyici katsayıları tablo 2’de verilmiştir. Frekans yanıtında yapılan yakınsamalardan dolayı bu denetleyicinin uygulanabilmesi için $\kappa = 1/(K_u T_u) > 0,2$ şartını sağlaması gerekmektedir [15].

Tablo 2. AMIGO PID Denetleyicisi

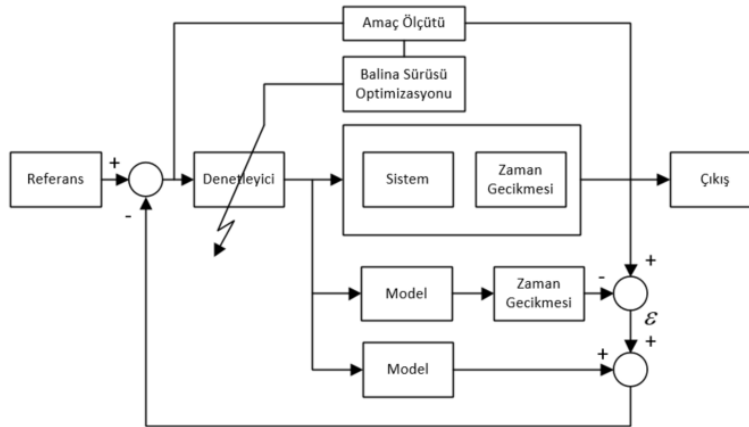
AMIGO PID	K	Ti	Td
Basamak Yanıtına Göre	$\frac{1}{K_p} \left(0,2 + 0,45 \frac{T}{L} \right)$	$\frac{0,4L + 0,8T}{L + 0,1T} L$	$\frac{0,5LT}{0,3L + T}$
Frekans Yanıtına Göre	$(0,3 - 0,1\kappa^4)K_u$	$\frac{0,6}{1 + 2\kappa} T_u$	$\frac{0,15(1 - \kappa)}{1 - 0,95\kappa} T_u$

2.3 Balina sürüsü optimizasyonu ile Smith öngörücüsü için PID denetleyici tasarımı

Balina sürüsü optimizasyonu daha önceden denildiği gibi bir çok farklı problemleri çözmek için son zamanlarda önerilmiş iyi metotlardan biridir. Smith öngörücüsü için denetleyici tasarlamak için sistemin genel yapısı Şekil ’de verilmiştir. Şekil 4’deki Smith öngörücüsüne amaç ölçütü ve Balina Sürüsü Optimizasyonu ilave olarak gelmiştir. Balina Sürüsü optimizasyonunun anlayacağı şekilde bu problemi çözebilmesi için problemi bu algoritmaya tanıtmamız gerektirir. Bu ise

geleneksel yada meta sezgisel algoritmalar arasında performans indeksi, amaç ölçütü gibi isimlerle isimlendirilmektedir. Burada Smith öngörücüsü için denetleyici tasarlanacağı için bu amaç ölçütü eşitlik **Hata! Başvuru kaynağı bulunamadı.**'da verilmiştir. Amaç ölçütünde t_f uygulama zamanı süresince referans ile çıkış işareti arasındaki farkın karesinin integrali yani ISE değeri alınmıştır. Ayrıca uygulama kısıtı olarak da kontrol işaretinin sonsuza gitmemesi için kısıt ilave edilmiştir. Burada $t_f = 1000 \text{ sn}$ ve kısıt ise $u \in [-3,3]$ olarak belirlenmiştir. Kısıt aşımında ise amaç fonksiyonuna bariyer fonksiyonu olarak 10^5 hata payı eklenmiştir.

$$\begin{aligned} J \\ \min (u) &= \int_0^{t_f} (r(t) - y(t))^2 dt \\ \text{Kısıt} & \quad u^- < u(t) < u^+ \end{aligned} \quad (1.5)$$



Şekil 5. Balina Sürüsü ile Smith Öngörücüsü için PID Denetleyici Tasarımı

Sürü içerisindeki her bir balinanın durumunu temsil eden kontrol parametreleri algoritmanın çalışma mantığına göre değerlendirilerek iterasyon içerisinde her bir balina için amaç ölçütü hesaplanmaktadır. Ardından balinalar bir önceki bölümde aktarılan davranışına göre hareket yine amaç ölçütleri hesaplanmada ve Smith öngörücüsü için en iyi PID denetleyicisinin katsayıları belirlenmeye çalışılmaktadır.

3. Simülasyon Sonuçları

Önerilen yöntemin başarısının karşılaştırılması için 2 farklı zaman gecikmeli sistem kullanılmıştır. Bu sistemlerin genel yapısı eşitlik (3.1)'de verilmiştir. Birinci dereceden zaman gecikmeli sistem açık çevrim kazancından (K), zaman sabitinden (T) ve zaman gecikmesinden (L) oluşmaktadır. Denemeleri yapılan ilk sistem eşitlik (3.2)'de diğer sistem ise eşitlik (3.3)'de verilmiştir.

İlk sistem olan $G_1(s)$ için tasarlanan denetleyicilerin katsayıları tablo 3'de verilmiştir. Klasik yöntemleri ile tasarlanan denetleyicilere bakıldığından hem kısıt ihlali olmuş hem de hata değeri yani amaç ölçüt değeri daha yüksek çıktığı görülmüştür. Ama balina sürüsü optimizasyonu ile tasarlanan denetleyicinin daha iyi sonuç verdiği görülmüştür. Şekil 6.1'de görüldüğü gibi sistem daha hızlı cevap vermiştir. Ayrıca Şekil 6.2'de görüldüğü üzere kontrol işareti sınırlar içinde

kalmıştır. Şekil 6.3’de görüldüğü üzere kontrol işaretinde diğer kontrol tekniklerine göre çaptırdamalar görülmemiştir.

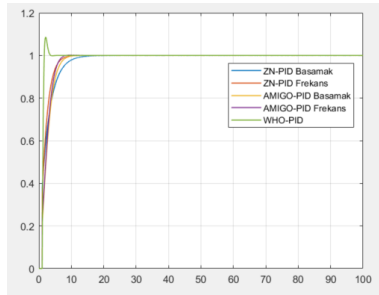
$$G(s) = \frac{K}{\tau s + 1} e^{-Ls} \quad (3.1)$$

$$G_1(s) = \frac{K}{s + 1} e^{-s} \quad (3.2)$$

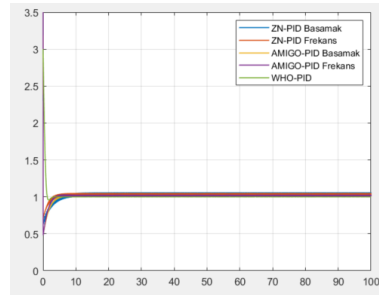
$$G_2(s) = \frac{K}{(s + 1)^2} e^{-0,5s} \quad (3.3)$$

Tablo 3. G1 Sistemi için

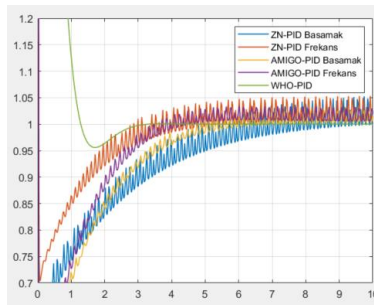
	K	K _i	K _d	Hata Değeri	Kısıt İhlali
ZN basamak	1.206	0.6029	0.6029	0.5180	1E5
ZN frekans	1.315	0.8766	0.5197	0.3891	1E5
AMIGO Basamak	0.6562	0.5966	0.252	0.7474	1E5
AMIGO frekans	0.6726	0.6794	0.2959	0.6709	1E5
WHO PID	3	6.4208	3.7142E-20	0.1445	0



Şekil 6.1. G₁ sistemi cevabı



Şekil 6.2. G₁ sistemi kontrol işareti

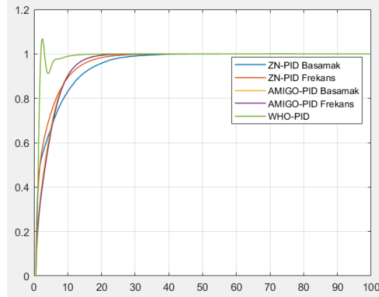


Şekil 6.3. G₁ sistemi kontrol işareti yakınlaştırılmış hali

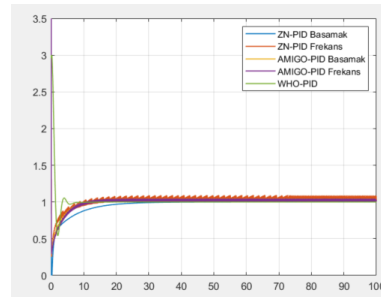
İkinci system olan $G_2(s)$ için tasarlanan denetleyicilerin katsayıları tablo 4’de verilmiştir. Klasik yöntemleri ile tasarlanan denetleyicilere bakıldığından hem kısıt ihlali olmuş hemde hata değeri yani amaç ölçüt değerleri daha yüksek çıktığı görülmüştür. Ama balina sürüsü optimizasyonu ile tasarlanan denetleyicinin daha iyi sonuç veridiği görülmüştür. Şekil 7.1’de görüldüğü gibi sistem daha hızlı cevap vermiştir. Ayrıca Şekil 7.2’de görüldüğü üzere kontrol işaretleri sınırlar içinde kalmıştır. Şekil 7.3’de görüldüğü üzere kontrol işaretinde diğer kontrol tekniklerine göre çattırdamalar görülmemiştir.

Tablo 4. G2 Sistemi için

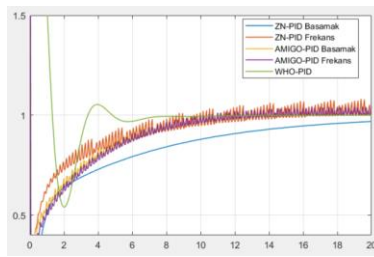
	K	K_i	K_d	Hata Değeri	Kısıt İhlali
ZN basamak	0.9895	0.2151	1.138	1.541	1E5
ZN frekans	1.058	0.2859	0.9264	1.247	1E5
AMIGO Basamak	0.5641	0.2565	0.5026	1.821	1E5
AMIGO frekans	0.4943	0.2472	0.4844	1.955	1E5
WHO PID	2,9592	0,99537	1,1633E-20	0.5755	0



Şekil 7.1 G₂ sistemi Cevabı



Şekil 7.2 G₂ sistemi kontrol işaretleri



Şekil 7.3 G₂ sistemi kontrol işaretleri yakınlaştırılmış hali

Sonuçlar

Bu çalışmada zaman gecikmeli sistemler için daha önceden önerilmiş olan smith predictor yapısına balina sürüsü optimizasyonu ile PID denetleyici tasarımı yapılmıştır. Önerilen yöntemde

control işaretinin sınırlandırılması ve amaç ölçütünü minimum yapan değeri sağladığı için klasik yöntemlere göre daha başarılı sonuçlar verdiği görülmüştür. İlerleyen çalışmalarda ise balina sürüsü optimizasyonunun performansının geliştirilmesi için hibrit yapılar önerilecek ve yine literatürde geliştirilmiş olan smith öngörücü yöntemlerine uygulanarak zaman gecikmeli sistemlerin performansın da olumlu sonuç verip vermediği incelenecektir.

Referanslar

- [1] Smith OJ. A controller to overcome dead time, ISA J 1959:28-33.
- [2] Çimen ME, Boz AF. İkinci dereceden ölü zamanlı ve geri tepmeli sistem parametrelerinin, Röle testi ve PSO, CS, FA algoritmaları ile belirlenmesi. Journal of the Faculty of Engineering & Architecture of Gazi University 2019-34.1.
- [3] Boz AF, Çimen ME. An Interface Design for Controlling Dead Time Systems Using PSO, CS and FA Algorithms. 8th International Advanced Technologies Symposium (IATS'17) 2017, Elazığ, Turkey.
- [4] Boz AF, Çimen ME. Geliştirilmiş Ateş Böceği Algoritması ile PID Denetleyici Tasarımı. 8th International Advanced Technologies Symposium (IATS'17) 2017, Elazığ, Turkey.
- [5] Poorani VJ, Vijay Anand LD. Comparison of PID controller and Smith predictor controller for heat exchanger. IEEE International Conference ON Emerging Trends in Computing, Communication and Nanotechnology (ICECCN) 2013.
- [6] Yücelen T. Uzun ölü zamanli sistemler için smith öngörücüsü yöntemi ile pi-p kontrolör tasarımı.
- [7] Xia C, Gengming G. Brushless DC Motors Control Based on Smith Predictor Modified by Fuzzy-PI Controller. Fifth International Conference on Fuzzy Systems and Knowledge Discovery 2008.
- [8] Normey-Rico JE., Juan GO, Camacho EF. A Smith-predictor-based generalised predictive controller for mobile robot path-tracking. *Control Engineering Practice* 7.6 1999:729-740.
- [9] Slavov T, Roeva O. Genetic algorithm tuning of PID controller in Smith predictor for glucose concentration control. *Int. J. Bioautomation*, 2011:101-114.
- [10] Gurban EH, Gheorghe-Daniel A. Comparison of modified Smith Predictor and PID Controller tuned by genetic algorithms for greenhouse climate control. 9th IEEE International Symposium on Applied Computational Intelligence and Informatics (SACI) IEEE 2014.
- [11] Campos J, et al. PD+ I Fuzzy Controller optimized by PSO applied to a variable dead time process. IEEE Third Ecuador Technical Chapters Meeting (ETCM) 2018.
- [12] Nicola M, Claudiu-Ionel N, Marian D. Delay Compensation in the PMSM Control by using a Smith Predictor. 8th International Conference on Modern Power Systems (MPS) 2019.
- [13] Maddi A, Abderrezak G, Daoud B. Design of nonlinear PID-smith predictor controllers with large time delays. Third World Conference on Complex Systems (WCCS) IEEE 2015.
- [14] Mirjalili S, Lewis A. The whale optimization algorithm. *Advances in engineering software* 95 2016:51-67.
- [15] Åström, KJ, Hägglund T. Revisiting the Ziegler–Nichols step response method for PID control. *Journal of process control* 14.6 2004:635-650.