



The Effects of PSO Parameters on an LQR Controlled Quadrotor System Gain

Sevinç Günsel and Şeref Naci Engin

EasyChair preprints are intended for rapid dissemination of research results and are integrated with the rest of EasyChair.

September 25, 2021

PSO Parametrelerinin LQR Kontrollü Dört Pervaneli Sistem Kazancı Üzerindeki Etkileri

The Effects of PSO Parameters on an LQR Controlled Quadrotor System Gain

Sevinç Günsel¹, Şeref Naci Engin²

^{1,2}Kontrol ve Otomasyon Mühendisliği Bölümü
Yıldız Teknik Üniversitesi, İstanbul

¹f6720007@std.yildiz.edu.tr

²nengin@yildiz.edu.tr

Özetçe

Doğrusal Karesel Düzenleyici (LQR), çoğunlukla çok-giriş çok-çıkışlı (MIMO) sistemlerde kullanılan dayanıklı ve uygulaması kolay denetleyicilerden biridir. Dört pervaneliler, kontrolü zor ve doğrusal olmayan çok-giriş çok-çıkışlı sistemlerdir. LQR denetleyiciler doğrusal olmayan çok-giriş çok-çıkışlı sistemlerle başa çıkabilen denetleyiciler olmalarına rağmen, denetleyiciyi en uygun performansına ulaştıracak parametrelerini, başka bir deyişle, ağırlık matrislerini belirlemek oldukça zordur. Ağırlık matrislerinin en iyilenmesi, Parçacık Sürü Optimizasyonu (PSO) metodu ile sağlanabilir. Bu çalışmada daha önceki çalışmamızda tasarladığımız LQR denetleyicili, yörünge takibi yapan bir dört pervanelinin PSO algoritmaları kullanılarak sistem kazancının elde edilmesi, PSO algoritmalarında yer alan parametrelerin değişimiyle sistem kazancına ve yörünge takibine etkileri incelenmiştir.

Abstract

Linear Quadratic Regulator (LQR) is one of the robust and convenient controllers which are mostly used in multiple-input multiple-output (MIMO) systems. Quadrotors are difficult-to-control and they are non-linear MIMO systems. Although LQR controllers are the controllers that can handle nonlinear MIMO systems, it is quite difficult to determine the parameters in other words, the weight matrices that will bring the controller to its optimal performance. Optimization of the weight matrices can be achieved by the Particle Swarm Optimization (PSO) method. In this study, system gain matrix is obtained by means of a PSO algorithm for an LQR controlled trajectory tracking quadrotor that was designed in our previous work. The effects of changing the parameters in the PSO algorithm on the system gain and trajectory tracking were investigated.

1. Giriş

Dört pervaneliler yüksek manevra kabiliyetine sahip, düşük kalkış ve iniş gerçekleştirebilen insansız hava araçlarıdır. Askerî ve sivil olmak üzere birçok alanda yaygın olarak kullanılmaktadırlar. Bu alanlardan en önemlisi arama-kurtarma çalışmalarıdır [1]. Dört pervanelilerin üç doğrusal ve üç dönel hareket olmak üzere altı serbestlik derecesi vardır. Eksenlerdeki öteleme ve dönme hareketleri, üzerindeki motor hızlarının değiştirilmesiyle sağlanmaktadır [2]. Sistem dinamiğinin doğrusal olmayışı kontrol edilebilirliğini güçleştirmektedir. Dört pervaneliler için yörünge takibi kapsamlı bir çalışma gerektirir. Uçuş kontrolünü sağlamak için PD, PID, LQR ve

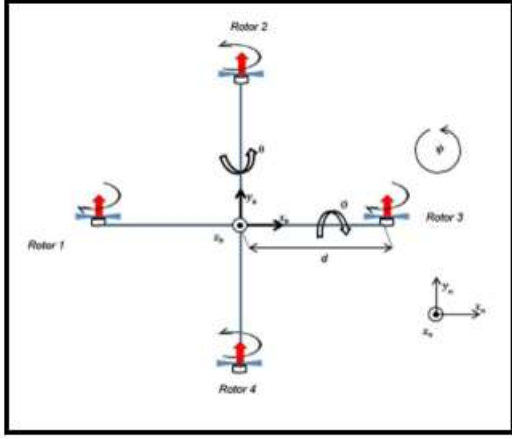
MPC (Model Öngörülü Kontrol) gibi çeşitli denetleyiciler kullanılmıştır [3].

Doğrusal karesel en iyileyci denetim (LQR), karesel yapıdaki maliyet fonksiyonunu azaltan sistem girişleri ve çıkışları için optimal kontrol problemini çözmektedir. Karesel maliyet fonksiyonu cebirsel Riccati denkleminin çözülmesiyle elde edilmektedir [4]. LQR denetleyici, doğrusallaştırılmış sistemlerin tüm durumlarının geri beslendiği denetleyicilerdir [5]. Bazı çalışmalarda LQR denetleyici Kalman filtresiyle birlikte kullanılmıştır. Kalman filtresinin kullanımı, sistem durumlarının ölçülemediği ya da ölçümlerin gürültülü olduğu durumlarda tercih edilmektedir [6]. Bir diğer en iyilenmesi tekniği Genetik Algoritmadır (GA). GA çeşitli uygulamalarda kullanılabilen, ancak popülasyonun optimal değerini bulmanın zaman aldığı bir metottur. Konvansiyonel en iyilenme yaklaşımlarında temel amaç diğer kontrol hedefleri yerine maliyet fonksiyonunu en aza indirmektir. Buna karşılık PSO yöntemi ile diğer kontrol hedefleri de yerine getirilmektedir [7]. [8]'de yapılan çalışmada, PSO algoritmaları kullanılarak dinamik süspansiyon sistemi için LQR denetleyicinin parametreleri olan Q ve R matrislerinin elemanları belirlenmiş ve LQR denetim performansının arttığı gözlemlenmiştir. Bilindiği üzere LQR denetleyicinin performansı doğrudan Q ve R ağırlık matrislerine bağlıdır. [9]'da yapılan çalışmada, ters sarkaç sistemi için LQR denetleyici tasarlanmıştır. Ayrıca, PSO algoritmaları ile sistemin en uygun kazancı elde edilmiştir. Başka bir çalışmada ise rotasyonel ters sarkaç sistemi için en iyilenme problemi GA ve PSO kullanılarak çözülmüştür [10]. PSO algoritmaları Doğrusal Karesel Gaussyan (LQG) denetleyicinin ağırlık matrislerinin belirlenmesinde de kullanılmaktadır [11]. En iyilenme parametrelerini ayarlamak için kullanılan bir başka yöntem ise Yapay Arı Kolonisidir (ABC). Bu metod ile çeşitli en iyilenme problemleri başarılı bir şekilde çözülebilmektedir [12].

Bu çalışmada, PSO algoritmasının parametrelerindeki değişimin LQR denetleyicili yörünge takibi yapan bir dört pervanelinin sistem kazancına etkileri incelenmiştir.

2. Matematik Model

Dört pervanelilerin hareket kontrolü motor hızlarının değiştirilmesiyle sağlanan küçük insansız hava araçlarıdır. Simetrik yapısı gereği motorları karşılıklı hizalanarak yerleştirilmiştir. Karşılıklı iki motor saat yönünde dönerken diğer iki motor saat yönünün aksine dönmektedir. Dört pervaneliler, üçü doğrusal üçü dönel hareket olmak üzere altı serbestlik derecesine sahip araçlardır [1].



Şekil 1: Dört pervaneli dinamiği [2]

Yuvarlanma hareketi iki ve dört numaralı motor moment farkı alınarak hesaplanmaktadır. Birinci ve üçüncü motorlar eksen üzerinde olduğundan momentleri sıfır olacaktır. Bu sebeple hesaba katılmazlar.

$$\tau_x = dT_2 - dT_4 \quad (1)$$

d (mm) moment kolu, T_2 ve T_4 ise ikinci ve dördüncü motorlardan üretilen itki kuvvetleridir. y -ekseni üzerindeki dönme hareketi yunuslama olarak adlandırılmaktadır. Birinci ve üçüncü motorların moment farkı alınarak hesaplanır.

$$\tau_y = dT_1 - dT_3 \quad (2)$$

z -eksenindeki momenti hesaplamak için, motorların dönme yönlerine göre itki kuvvetleri (+) ve (-) olarak işaretlenip toplanır ve bu toplam rüzgardan kaynaklanan sürtünme katsayısı ile çarpılır.

$$\tau_z = C_d(-T_1 + T_2 - T_3 + T_4) \quad (3)$$

C_d sürtünme katsayısını temsil etmektedir. x , y , z yönlerindeki doğrusal ve dönele hareket denklemleri küçük açı varsayımı yapılarak verilen denklemlerdeki gibi elde edilmiştir:

$$\begin{aligned} \dot{x} &= \dot{x}_b & \ddot{x} &= g\theta \\ \dot{y} &= \dot{y}_b & \ddot{y} &= -g\phi \\ \dot{z} &= \dot{z}_b & \ddot{z} &= g - T/m \\ \dot{\phi} &= \dot{p} & \ddot{\phi} &= (1/J_x)\tau_x \\ \dot{\theta} &= \dot{q} & \ddot{\theta} &= (1/J_y)\tau_y \\ \dot{\psi} &= \dot{r} & \ddot{\psi} &= (1/J_z)\tau_z \end{aligned} \quad (4)$$

g yerçekimi ivmesidir. Verilen denklemlerde dört pervanelinin kendi eksen ve dünya eksenindeki dönüşüm ve doğrusallaştırmalar yapılmıştır. Bu denklemlerin durum-uzay gösterimleri: $\dot{x}(t) = Ax(t) + Bu(t)$; $y(t) = Cx(t)$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ g & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -g & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & g & -1/m & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/J_x & 0 \\ 0 & 0 & 0 & 0 & 0 & 1/J_y \\ 0 & 0 & 0 & 0 & 0 & 1/J_z \end{bmatrix} \quad (5)$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

2.1. LQR Denetleyicinin Parametrelerinin PSO ile Belirlenmesi

2.1.1. Doğrusal Karesel Düzenleyici (LQR)

Optimal denetleyici, tam durum geri-beslemeli kapalı-çevrim kontrol sistemidir. Optimal kontrol problemlerinin çözümünde LQR yaygın olarak kullanılmaktadır. Optimal kontrol, durum, girdi, çıktı ve maliyet fonksiyonunu içermektedir. Bazı kaynaklarda maliyet fonksiyonu performans indeksi olarak da adlandırılabilir. Optimal kontrolde temel amaç maliyet fonksiyonunu azaltmaktır. Maliyet fonksiyonu karesel yapıda bir fonksiyon olup, aşağıda yer alan denklemlerdeki gibi verilmiştir:

$$J = \int_0^{\infty} (x^T Q x + u^T R u) dt \quad (6)$$

Q durum ağırlık matrisini, R kontrol ağırlık matrisini temsil etmektedir. $x(t)$ durum vektörü ve $u(t)$ kontrol giriş vektörüdür ve verilen kontrol kanunuyla hesaplanmaktadır:

$$u(t) = -Kx(t) \quad (7)$$

K , sistemin kazanç matrisini ifade etmektedir ve verilen formülle hesaplanmaktadır. Matlab ile $lqr(A,B,Q,R)$ komutu kullanılarak da kazanç matrisi elde edilebilir. Kazanç matrisi,

$$K = R^{-1}B^T P \quad (8)$$

P optimal matristir. Pozitif ve simetrik P matrisi cebirsel Riccati denklemi çözülerek hesaplanmaktadır [13].

$$A^T P + PA - PBR^{-1}B^T P + Q = 0 \quad (9)$$

2.1.2. Parçacık Sürü Optimizasyonu (PSO)

Parçacık sürü optimizasyonu algoritmaları LQR denetleyicinin parametreleri olan Q ve R ağırlık matrislerinin alacağı en uygun değerlerin belirlenmesini sağlamaktadır. Algoritma, Kennedy ve Eberhart tarafından kuş gibi sürü halinde hareket eden canlılardan esinlenilerek geliştirilmiştir. PSO algoritması, sürüdeki parçacıkların davranışına dayanmaktadır. Sürüdeki her bir parçacık bilgisi bir diğer parçacığa aktarılmaktadır. Böylece, grubun verimi artmaktadır. Sürüdeki bir parçacık şimdiki hızını, geçmiş tecrübesini ve komşu parçacığın tecrübesini kullanarak en uygun değerine yaklaşır. Sürüdeki parçacıkların her biri en iyi konumuna gelir, konumu en iyi olarak belirlenen parçacığa göre sürünün en iyi konumu belirlenmektedir. Parçacık en iyi konumu ve sürü en iyi konumu sırasıyla *yerel en iyi* ve *global en iyi* olarak adlandırılmaktadır. n -boyutlu bir sistem için PSO algoritmasında, aşağıda verilen hız ve konum denklemleri kullanılarak parçacıkların en iyi konumu elde edilmektedir.

$$v_i(t+1) = wv_i(t) + c_1 \text{rand}(pbest(t) - x_i(t)) + c_2 \text{rand}(gbest(t) - x_i(t)) \quad (10)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (11)$$

Denklemlerde yer alan $x_i(t+1)$ ve $v_i(t+1)$ sırasıyla, i . parçacığın $(t+1)$ iterasyonundaki konum ve hız denklemlerini temsil etmektedir. w atalet ağırlık faktörüdür. c_1 ve c_2 , bireysel öğrenme ve küresel öğrenme katsayılarıdır. $\text{rand}()$ ile ifade edilen fonksiyon pseudo rastgele fonksiyonu olarak adlandırılır ve $[0, 1]$ aralığında değerler almaktadır. $pbest$ parçacığın en iyi konumunu, $gbest$ ise sürünün en iyi konumunu ifade etmektedir. Ağırlık katsayısı w sayesinde, parçacık hızı dinamik olarak ayarlanmaktadır ve parçacığın yerel, sürünün global pozisyonu aranırken bir denge sağlanmaktadır. w değeri başlangıçta büyük olursa algoritma önce sürünün en uygun

konumunu daha sonra parçacıkların yerel en iyi konumunu arar. Ağırlık katsayısı verilen formülle hesaplanmaktadır:

$$w = w_{\max} \frac{(w_{\max} - w_{\min}) \text{iter}}{\text{iter}_{\max}} \quad (12)$$

w_{\min} başlangıç w_{\max} ise son ağırlık değerlerini ifade etmektedir. iter anlık iterasyon sayısı, iter_{\max} belirlenen maksimum iterasyon sayısıdır [14].

2.1.3. Arama Uzayı ve Maliyet Fonksiyonu Optimizasyonu

Algoritma uygulaması boyunca parçacıklar, kontrol giriş vektörü değerlerini ararlar. Önce, Q ve R matrislerine başlangıç değerleri atanmaktadır. Tüm sürü popülasyonunun hareketinden sonra her bir parçacık için o anki pozisyonları kullanarak maliyet fonksiyonu hesaplanmaktadır. Karesel yapıdaki maliyet fonksiyonu, yörünge takibi yapan bir dört pervaneli için aşağıda yer alan formüldeki gibi verilmiştir.

$$J_i^t = [x_{ref}(k+1) - x_i(k+1)]^T Q [x_{ref}(k+1) - x_i(k+1)] + [u_{nom} - u_i(k)]^T R [u_{nom} - u_i(k)] \quad (13)$$

k kaçınıcı iterasyon olduğunu, i kaçınıcı parçacık olduğunu göstermektedir. $u_i(k)$ vektörü i . parçacığın koordinatını belirtmektedir. u_{nom} denge noktasındaki nominal giriş vektörünü temsil etmektedir. Bu çalışmada sistemdeki ölçüm hataları ve sisteme etki eden bozulmalar yok sayılmıştır. Referans değerleri verilen sistem için $x_i(k+1)$ vektörünün değerleri yukarıda verilen denklemler sayesinde direkt olarak hesaplanabilmektedir. Q ve R köşegen matrisleri ise maliyet fonksiyonunda hangi durum ve girişin ne kadar ağırlıkta etki etmesi gerektiğini belirlemektedir. Q matrisi durum vektörü ve referans değerleri arasındaki hata büyüklüğüne etki ederken, R matrisi de sistemin nominal değerler ile kontrol sinyali arasındaki davranışını belirler. Q ve R matrisleri veya kontrol sinyali için belirlenen alt ve üst limitler sayesinde algoritma, parçacıkların arama uzayında optimal konumlarını bulmasına olanak sağlamaktadır [20].

3. Tartışma

Optimal kontrollü yörünge takibi yapan bir dört pervaneli için Q ve R ağırlık matrislerinin ayarlanması önem arz etmektedir. PSO algoritmaları, bu matrislerin optimal değerlerinin bulunmasına yardımcı olan, doğrusal olmayan sistemlerde etkili bir şekilde kullanılabilen bir yöntemdir. Fakat Q ve R parametreleri kadar PSO algoritmasında yer alan parametrelerin belirlenmesi de önem arz etmektedir. Bu çalışmada, PSO algoritmasında yer alan parametrelerde yapılan değişikliklerin yörünge takibi yapan bir dört pervanelinin kazanç matrisine ve sistem çıkışlarına etkileri incelenmiştir. Yedi farklı durum ele alınarak, sistem çıkışları gözlemlenmiş ve optimal kontrollü çıkışlarla her bir sonuç karşılaştırılmıştır. Başlangıçta, Q ve R matrisleri tüm sistem durumları ve girişleri için eşit ağırlığa sahiptir. Sistemin optimal kontrollü kazanç matrisi:

$$K1 = \begin{bmatrix} 1 & 1.0972 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & -1.0972 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1.0995 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 17.8490 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 7.8490 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 5.5949 & 0 \end{bmatrix} \quad (14)$$

Birinci durumda, algoritma parametrelerinden w ağırlık katsayısı 2, c_1 ve c_2 öğrenme katsayıları 2, sürü popülasyonu

40, iterasyon sayısı ise 50 olarak seçilmiştir. Algoritma sonunda elde edilen Q ve R değerleri kullanılarak, Matlab'de $lqr(A,B,Q,R)$ komutu ile sistemin kazanç matrisi olan K matrisi hesaplanmıştır.

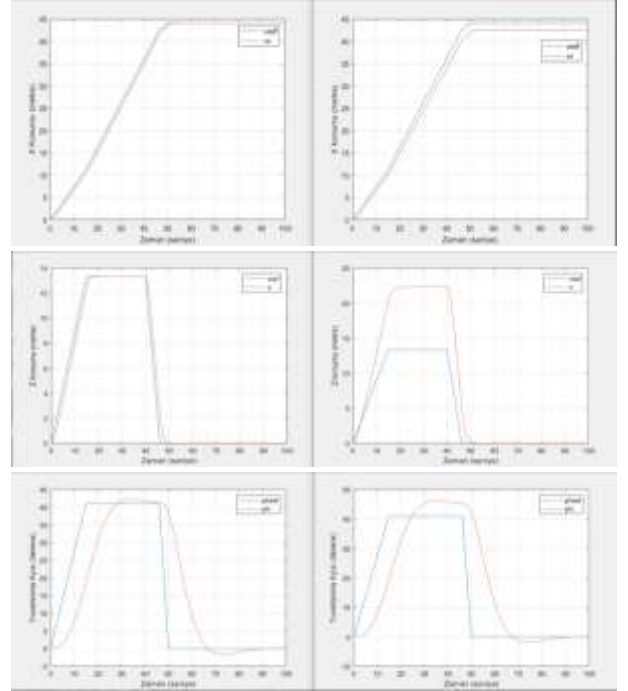
Tablo 1: Birinci durum parametreleri

Parametre	Değer
w	2
c_1	2
c_2	2
iterasyon	50
popülasyon	40

$$K2 = \begin{bmatrix} 1.0364 & 1.0334 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1.1126 & -2.5055 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.5983 & 0.9069 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.9174 & 7.4675 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.8203 & 7.0994 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.8679 & 5.1842 & 0 \end{bmatrix} \quad (15)$$

Elde edilen matris Simulink programında tasarlanan sisteme aktarılıp yörünge takibindeki sonuçlar optimal kontrol sonuçlarıyla karşılaştırılmıştır.

Grafiklerde sol tarafta optimal kontrol sonuçları yer alırken sağ tarafta PSO sonuçları yer almaktadır. Elde edilen sonuçlara göre x -eksenindeki geçiş hareketi ile yalpa açısında referans değerlerine yakın sonuçlar elde edilmiştir. Fakat z -eksenindeki hareketinde istenilen irtifanın çok üzerine çıkarak referansı takip ettiği gözlemlenmiştir. Bu durumda, sistemin z durumu için algoritma tarafından belirlenen ağırlığın uygun olmadığı sonucuna varabiliriz.



Şekil 2: Birinci durum x, z ve ϕ sonuçları

İkinci durumda, iterasyon sayısı iki katına çıkarılıp diğer parametreler ve popülasyon sayısı değiştirilmemiştir. Elde edilen K kazanç matrisi:

Tablo 2: İkinci durum parametreleri

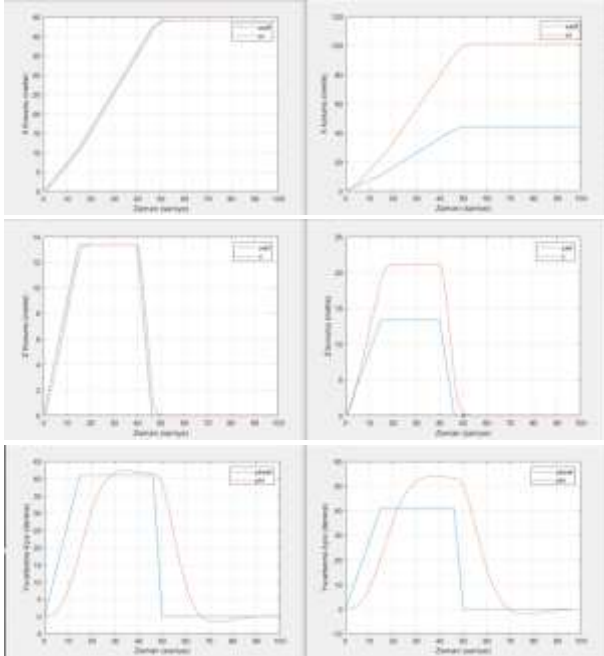
Parametre	Değer
w	2
c_1	2
c_2	2
iterasyon	100
popülasyon	40

$$K^3 = \begin{bmatrix} 0.4360 & 1.0334 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.6577 & -0.8382 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.6316 & 1.0221 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.7832 & 7.0130 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5977 & 6.0394 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2.1498 & 8.2184 & 0 & 0 & 0 \end{bmatrix} \quad (16)$$

Yeni elde edilen kazanç matrisinin çıktıları aşağıda verilen grafiklerdeki gibidir. Sistem çıktılarında tüm durumların verilen referans değerlerini epey aştıkları görülmektedir. İterasyonun belirli bir limite olması gerekmektedir. Aksi takdirde iterasyonun çokluğuna ya da azlığına bağlı olarak istenilen konumun ilerisinde veya gerisinde olabilir.

Üçüncü durumda, iterasyon sayısı tekrar 50 olarak güncellenip popülasyon sayısı 40'tan 100'e çıkarılmıştır. Diğer parametreler değiştirilmemiştir. Algoritma sonunda elde edilen Q ve R matrisleriyle hesaplanan sistem kazancı Tablo 3'te verildiği gibi elde edilmiştir.

Sistem çıktılarıyla optimal kontrol çıktılarının karşılaştırıldığı grafikler aşağıda yer alan çizimlerde görülmektedir. Popülasyon artışıyla, dört pervanelinin x yönünde istenilen konumdan geride kalmasına sebep olurken z yönünde kazandığı irtifa ve yuvarlanma açısı referans değerinin yaklaşık iki katına çıkmıştır. Bu durumda, dört pervanelinin yörüngeyi tam olarak takip ettiği söylenemez. Dördüncü durumda, popülasyon sayısı tekrar 40'a indirilip w ağırlık katsayısı 0.99 olarak güncellenmiştir. Diğer parametreler ise değiştirilmemiştir. Elde edilen kazanç matrisi sunulmuştur.



Şekil 3: İkinci durum x , z ve ϕ sonuçları

Tablo 3: Üçüncü durum parametreleri

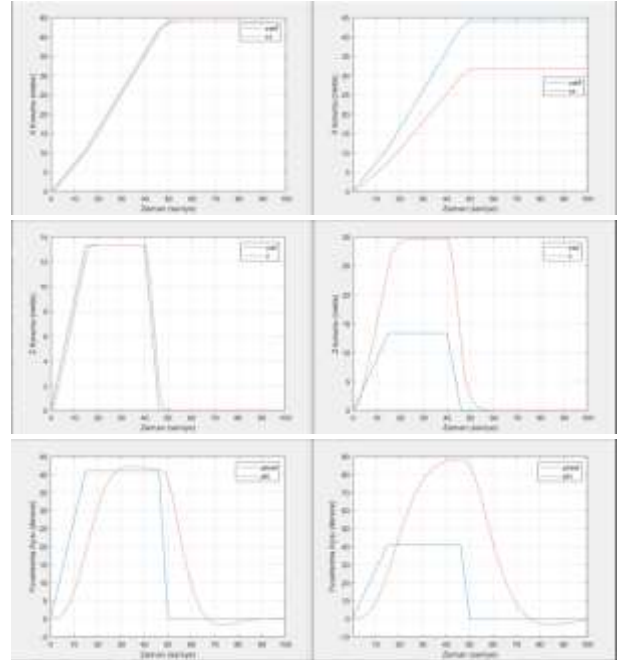
Parametre	Değer
w	2
c_1	2
c_2	2
iterasyon	50
popülasyon	100

$$K^4 = \begin{bmatrix} 1.3892 & 1.5871 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1.0130 & -1.0806 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.4501 & 1.1403 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.4785 & 5.5452 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0091 & 7.8388 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.8277 & 5.0559 & 0 & 0 & 0 \end{bmatrix} \quad (17)$$

Hesaplanan kazanç matrisi ile Simulink programında koşturulan sistem modelinin çıktıları ve optimal kontrol çıktıları verilen grafiklerdeki gibidir. Bir önceki duruma göre x eksenindeki konumu aynı kalırken z ekseninde kazandığı irtifa ve x yönündeki yalpa açısı için elde edilen değerler, verilen referans değerlerinin iki katından daha azdır. Referans ve çıktıları arasındaki hata payının ciddi oranda azaldığını söylemek mümkündür.

Beşinci durumda, w ağırlık katsayısı 2 olarak güncellenmiştir ve global öğrenme katsayısı iki katına çıkarılıp diğer parametrelerde değişiklik yapılmamıştır. Elde edilen kazanç matrisi ve sistem çıktıları Tablo 5'te verilmiştir.

Altıncı durumda, c_2 parametresi eski değerine getirilerek bireysel öğrenme katsayısı olan c_1 iki katına çıkarılmıştır. Diğer parametrelerde değişiklik yapılmamıştır. Hesaplanan sistem kazanç matrisi ve çıkışlar Tablo 6'da sunulmuştur.

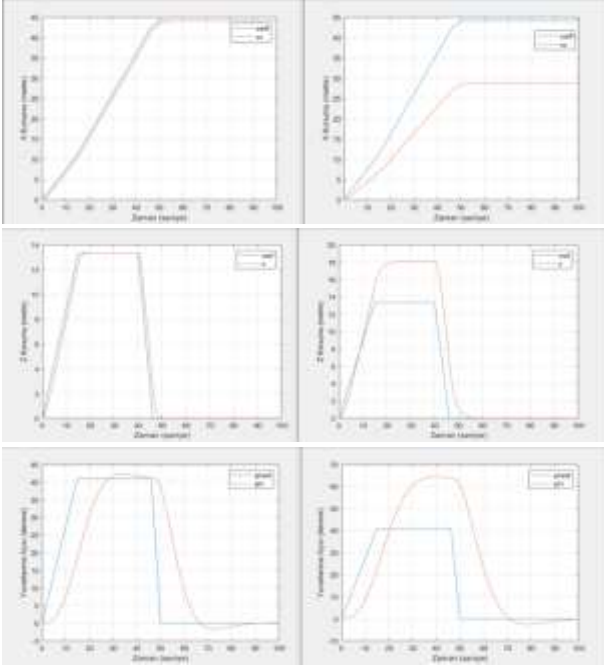


Şekil 4: Üçüncü durum x , z ve ϕ sonuçları

Tablo 4: Dördüncü durum parametreleri

Parametre	Değer
w	0.99
c ₁	2
c ₂	2
iterasyon	50
popülasyon	40

$$K5 = \begin{bmatrix} 1.5351 & 1.4313 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -0.6803 & -1.1106 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.7371 & 1.7151 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.6588 & 6.3314 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5440 & 5.7611 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.8175 & 5.0143 & 0 \end{bmatrix} \quad (18)$$



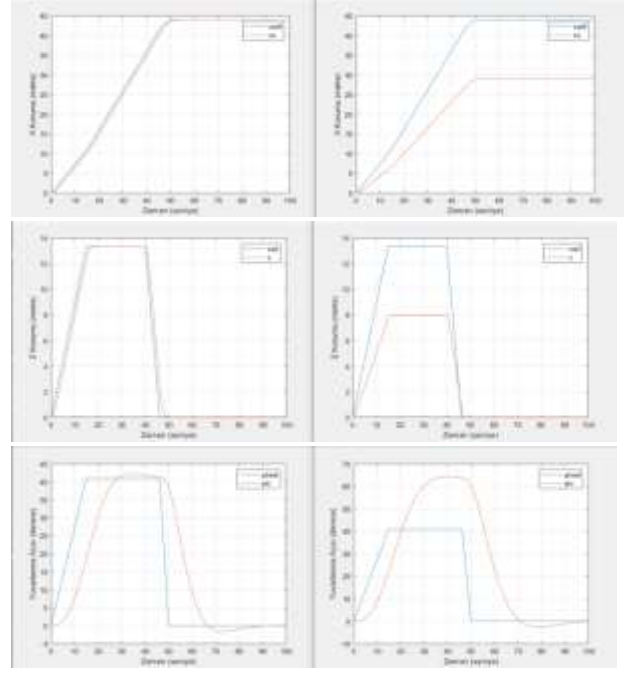
Şekil 5: Dördüncü durum x, z ve φ sonuçları

Tablo 5: Beşinci durum parametreleri

Parametre	Değer
w	2
c ₁	2
c ₂	4
iterasyon	50
popülasyon	40

$$K6 = \begin{bmatrix} 1.5108 & 1.6963 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1.5349 & -1.5769 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.6772 & 0.9845 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.6602 & 6.3578 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0301 & 8.0078 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.3677 & 6.7558 & 0 \end{bmatrix} \quad (19)$$

Çıkan sonuçlara göre, x ve z yönündeki hareketinde istenilen konumun gerisinde kaldığı gözlemlenmiştir. x yönündeki yalpa açısı ise oldukça artmıştır. Dört pervanelinin istenilenden fazla bir eğimle yörüngede ilerlediği söylenebilir. Bu durum için dört pervanelinin, istenilen yörüngeyi takip etmediği açıkça görülmektedir.



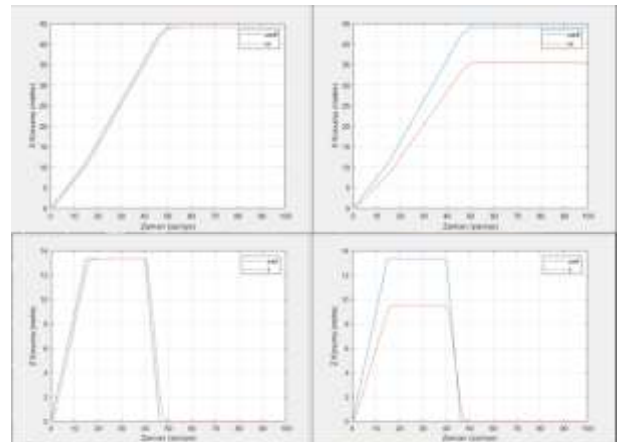
Şekil 6: Beşinci durum x, z ve φ sonuçları

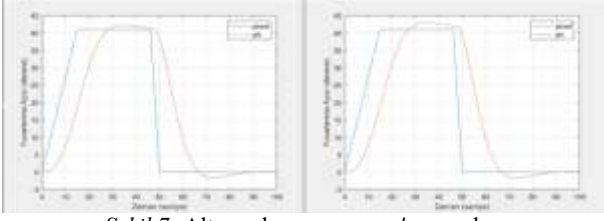
Sonuçlar c₁ parametresinin yalpa açısı üzerinde olumlu etkisi olduğu göstermiştir. x ve z yönündeki hareketinde ise bir önceki durumda olduğu gibi istenilen konumun gerisinde kaldığı görülmektedir. Yedinci durumda, tüm parametreler eski değerlerine getirilip, Q ve R matrisleri için belirlenen alt limit 0.1 oranında azaltılmıştır. Yapılan bu değişiklik sonucunda sistem kazanç matrisinin değişmediği görülmüştür.

Tablo 6: Altıncı durum parametreleri

Parametre	Değer
w	2
c ₁	4
c ₂	2
iterasyon	50
popülasyon	40

$$K7 = \begin{bmatrix} 1.2401 & 1.2673 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1.0814 & -0.9052 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.4047 & 1.1839 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.9841 & 7.7544 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2.1249 & 11.5765 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0422 & 5.6933 & 0 \end{bmatrix} \quad (20)$$





Şekil 7: Altıncı durum x, z ve ϕ sonuçları

4. Sonuçlar

Bu çalışmada, optimal kontrollü yörünge takibi yapan bir dört pervaneli insansız hava aracı için, PSO algoritmaları kullanılarak Q ve R ağırlık matrisleri belirlenmiştir. PSO parametrelerinde yapılan değişikliklerin sistem kazancına ve çıkışlarına etkisi incelenmiştir. Sonuçlar optimal kontrol çıkışlarıyla karşılaştırılmıştır. Elde edilen kazanç matrisleri doğrultusundaki sistem cevaplarına bakarak iterasyon sayısının, popülasyon sayısının ve öğrenme katsayılarının sonuca ciddi oranda etkisi olduğu gözlemlenmiştir. Ağırlık katsayısı w , daha önceki bölümlerde bahsedildiği üzere parçacıkların en uygun konumlarını bulmalarında etkili bir parametredir.

Başlangıçta Q ve R matrisleri, tüm durumlar ve kontrol girişleri için eşit ağırlık değerleri belirlendiğinden, algoritmada Q ve R matrisleri için hesaplanan değerlerin, istenen durumlara, diğer bir deyişle x, z ve ϕ durumlarına gerektiği kadar ceza verilemediği, bu sebeple istenen referans değerinin ya çok altında ya da çok üstünde kaldığı gözlemlenmiştir.

Gelecek çalışmada, Q ve R matrislerinin başlangıç değerleri farklı verilerle ve PSO parametreleri bu çalışmadan elde edilen sonuçlar doğrultusunda güncellenerek sistem optimizasyonu sağlanmaya çalışılacaktır. Sistem davranışı gözlemlenip sonuçlardaki iyileşme sayısal olarak verilmelidir.

Kaynakça

- [1] S. Ghazbi, Y. Aghli, M. Alimohammadi and A. Akbari, "QUADROTORS UNMANNED AERIAL VEHICLES: A REVIEW", *International Journal on Smart Sensing and Intelligent Systems*, vol. 9, no. 1, pp. 309-333, 2016.
- [2] Z. Tahir, M. Jamil, S. Liaqat, L. Mubarak, W. Tahir and S. Gilani, "State Space System Modeling of a Quad Copter UAV", *Indian J. of Science and Tech.*, vol. 9, no. 27, 2016.
- [3] C. Liu, J. Pan and Y. Chang, "PID and LQR trajectory tracking control for an unmanned quadrotor helicopter: Experimental studies", *2016 35th Chinese Control Conference (CCC)*, 2016.
- [4] H. Trentelman, "Linear Quad. Optimal Control", *Encyclopedia of Systems and Control*, pp. 1-8, 2013.
- [5] L. Martins, C. Cardeira and P. Oliveira, "Linear Quadratic Regulator for Trajectory Tracking of a Quadrotor", *IFAC-PapersOnLine*, vol. 52, no. 12, pp. 176-181, 2019.
- [6] C. Hajiyeve and S. Vural, "LQR Controller with Kalman Estimator Applied to UAV Longitudinal Dynamics", *Positioning*, vol. 04, no. 01, pp. 36-41, 2013.
- [7] A. Al-Mahturi; and H. Wahid, "Optimal Tuning of Linear Quadratic Regulator Controller Using a Particle Swarm Optimization for Two-Rotor Aerodynamical System", *World Academy of Science, Engineering and Technology*, 2017.
- [8] M. Assahubulkahfi, Y. Md. Sam, A. Maselena and M. Huda, "LQR Tuning by Particle Swarm Optimization of Full Car Suspension System", *International Journal of Engineering & Technology*, vol. 7, no. 213, p. 328, 2018.
- [9] S. Howimanporn, S. Thanok, S. Chookaew and W. Sootkaneung, "Design and implementation of PSO based LQR control for inverted pendulum through PLC", *2016 IEEE/SICE International Symposium on System Integration (SII)*, 2016.
- [10] S. Ghoreishi and M. Nekoui, "Optimal Weighting Matrices Design for LQR Controller Based on Genetic Algorithm and PSO", *Advanced Materials Research*, vol. 433-440, pp. 7546-7553, 2012.
- [11] S. Bouallègue and R. Fessi, "LQG controller design for a quadrotor UAV based on particle swarm optimisation", *International Journal of Automation and Control*, vol. 13, no. 5, p. 569, 2019.
- [12] A. Abdulla, I. Mohammed and A. Jasim, "Roll Control System Design Using Auto Tuning LQR Technique", *International Journal of Engineering and Innovative Technology (IJEIT)*, vol. 7, no. 1, 2017.
- [13] S. Günsel, Y. İstefanopulos and E. Dinçmen, "Optimal Control of a Quadrotor", *TURK-COSE2020*, Kazakhstan, 2020.
- [14] P. Pawłowski and S. Konatowski, "Linear controller design with the use of PSO algorithm for UAV trajectory tracking", *Radioelectronic Systems Conference, 2019*.
- [15] Q. Bai, "Analysis of Particle Swarm Optimization Algorithm", *Computer and Info. Sci.*, vol. 3, no. 1, 2010.
- [16] "Optimal Tuning of Linear Quadratic Regulators Using Quantum Particle Swarm Optimization", *Int. Conf. Control, Dynamic Systems, and Robotics*, Ottawa, Ontario, Canada, 2014.
- [17] Y. Nazaruddin, A. Andrini and B. Anditio, "PSO Based PID Controller for Quadrotor with Virtual Sensor", *IFAC-PapersOnLine*, vol. 51, no. 4, pp. 358-363, 2018.
- [18] G.R. Yu and P. H. Hsieh, "Optimal Design of Helicopter Control Systems Using Particle Swarm Optimization", *IEEE International Conference on Industrial Cyber Physical Systems (ICPS)*, pp. 346-351, 2019.
- [19] H. Shakhathreh, A. Khreishah, A. Alsarhan, I. Khalil, A. Sawalmeh and N. S. Othman, "Efficient 3D placement of a UAV using particle swarm optimization", *8th Communication Systems (ICICS)*, pp. 258-263, 2017.
- [20] S. Madruga, A. Tavares, G. Basso, T. Nascimento and A. Brito, "A PSO-Based Tuning Algorithm For Quadcopter Controllers", *SBA Sociedade Brasileira de Automática*, vol. 1, no. 1, 2019.
- [21] Ü. Önen, "Particle Swarm Optimization Based LQR Control of an Inverted Pendulum", *Engineering and Technology Journal*, vol. 2, no. 5, 2017.