

## DÖRT ROTORLU İHA'NIN LQT ALGORİTMASI KULLANILARAK OPTİMUM YÖRÜNGE TAKİBİ

Emre Can Suiçmez\* ve Ali Türker Kutay†  
ODTÜ, Ankara

### ÖZET

*Bu makalede, dikey iniş kalkış yapabilen, dört-rotorlu("quadrotor") bir insansız hava aracının(İHA), LQT algoritması kullanılarak optimum yörünge takibi elde edilmeye çalışılmıştır. LQT kontrolcü, ayrık-zamanlı matris Riccati denkleminin, zamanda tersine çözülmesiyle elde edilmiştir. İlk olarak, dört rotorlu'nun doğrusal olmayan dinamiği Newton hareket denlemleri kullanılarak formüle edilmiştir. Sonrasında, LQT algoritmasında kullanılmak üzere, dört-rotorlu'nun doğrusal olmayan dinamik modeli, "hover" denge durumu etrafında doğrusallaştırılmıştır. LQT algoritması MATLAB ortamında çözdürülmüştür. Zamana göre değişen ve yörünge takibi için optimize edilmiş kontrol kazançları çevrimdışı("off-line") olarak elde edilmiştir. Elde edilen LQT kontrolcü, MATLAB/Simulink ortamında modellenmiş ve test edilmiştir. Ayrıca, LQT kontrolcü, literatürde yaygın olarak kullanılan sabit kazançlı LQR kontrolcü ile simülasyonlar aracılığıyla karşılaştırılmıştır. Elde edilen sonuçlar, LQT kontrolcünün, sabit kazançlı LQR'a göre, daha yüksek doğrulukta ve az enerji harcayarak yörünge takibi gerçekleştirebildiğini göstermiştir.*

### GİRİŞ

Dört-rotorlu("quadrotor"), son 15 yılda bir çok araştırmaya konu olan, popüler bir insansız hava aracıdır. Dikey iniş kalkış(VTOL) özelliği ve yüksek manevra kabiliyeti sayesinde, klasik insansız hava araçlarına göre daha esnek operasyon kabiliyetine sahiptir. Ayrıca, VTOL özelliği bulunan diğer İHA'lara göre mekanik ve aerodinamik açıdan çok daha basit bir yapıya sahiptir [Bouabdallah, 2007]. Fakat, dört rotorlu'nun en belirgin dezavantajlarından biri yüksek enerji tüketimidir [Bouabdallah, 2007].

Dört-rotorlu insansız hava aracı, yüksek derecede doğrusal olmayan ve birbiriyle bağlantılı bir dinamik içeren, kararsız bir sistemdir [Bouabdallah, 2007; Li ve Song, 2012; Suiçmez ve Kutay, 2014]. Bu nedenle, kontrolcü tasarımı açısından, bir çok araştırmacının ilgisini çekmiştir.

Literatürdeki bir çok çalışma, dört rotorlu'nun yönelim kontrolü ve stabilizasyonu üzerine yoğunlaşmaktadır. Yörünge takibi ise, dört rotorlu'nun hem yönelimi hem de pozisyonu aynı anda

\*Araştırma Görevlisi, Havacılık Uzay Müh. Böl., E-posta: esuicmez@ae.metu.edu.tr

†Asst. Prof. Dr., Havacılık Uzay Müh. Böl., E-posta: kutay@metu.edu.tr

kontrol edilerek elde edilebilmektedir ve daha az sayıda çalışmanın konusu olmuştur [Bouabdallah, 2007; Madani ve Benallegue, 2007; Mellinger ve Kumar, 2011; Raffo, Ortega ve Rubio, 2008; Suiçmez ve Kutay, 2014; Zhou, Zhang, Rabbath ve Theilliol, 2010]

Daha öncede belirtildiği gibi, dört-rotorlu İHA'nın önemli bir dezavantajı, yüksek enerji tüketimidir [Bouabdallah, 2007]. Bu nedenle bu çalışmada, enerji tüketimini azaltacak ve kompleks yörüngelerin yüksek doğrulukla takip edilmesini sağlayacak bir optimum kontrol algoritması kullanılmıştır [Naidu, 2013]. Literatürdeki çalışmaların çoğunda, optimum kontrol yöntemi olarak sabit kazançlı LQR (Lineer Quadratic Tracking) kullanılarak, dört rotorlu'nun yörünge takibi sağlanmıştır [Al-Younes, Al-Jarrah ve Jhemi, 2010; Bouabdallah, 2007; Li ve Song, 2012; Minh ve Ha, 2010]. Bu çalışmada ise, literatürde LQT (Lineer Quadratic Tracking) olarak adlandırılan ve yörünge takibi için modifiye edilmiş bir optimum kontrol yöntemi kullanılmıştır. LQT yönteminde, çevrimdışı ("off-line") hesaplanan, zamana göre değişen ve istenilen yörüngeyi takip etmek için optimize edilmiş kazançlar kullanılmaktadır [Naidu, 2013; Suiçmez ve Kutay, 2014]. Bu nedenle LQT'nin, literatürde yaygın olarak kullanılan sabit kazançlı LQR yöntemine göre daha avantajlı sonuçlar vermesi beklenilmiştir. Elde edilen simülasyon sonuçları, LQT yönteminin LQR'a göre daha yüksek doğrulukta ve az enerji harcayarak kompleks yörüngeleri takip edebildiğini göstermiştir.

Bu çalışmada kullanılan LQT algoritması, ayrık-zamanlı matris Riccati denkleminin, zamanda tersine integral alınarak çözülmesini içermektedir [Naidu, 2013]. Elde edilen optimum kontrol kazançları, zamana göre değişmektedir ve MATLAB ortamında çevrimdışı bir şekilde elde edilmiştir. Bir başka ifadeyle, ilk olarak, takip edilmesi istenilen yörünge oluşturulmaktadır, sonrasında ise, istenilen yörüngeyi takip etmek için optimize edilmiş kontrol kazançları çevrimdışı olarak hesaplanmaktadır.

İlk olarak, dört rotorlu'nun doğrusal olmayan dinamik modeli Newton hareket denklemleri kullanılarak elde edilmiştir. Öte yandan, LQT algoritmasının yapısı gereği, doğrusal bir dinamik model kullanılmaktadır [Naidu, 2013]. Bu nedenle, dört-rotorlu'nun, doğrusal olmayan dinamik modeli, "hover" denge durumu etrafında doğrusallaştırılmıştır. Algoritma ayrık zamanda çözüldüğü için, doğrusallaştırılmış sürekli zamanlı ("continuous-time") dinamik model ayrık zamana ("discrete-time") çevrilmiştir. Sonrasında, minimize edilmeye çalışılan performans indeksi ("cost function") ayrık zamanda tanımlanmıştır. Son olarak, zamana göre değişen optimum kontrol kazançları, MATLAB ortamında LQT algoritmasının çözdürülmesi sonucu çevrimdışı olarak elde edilmiş ve simülasyon modeline entegre edilmiştir.

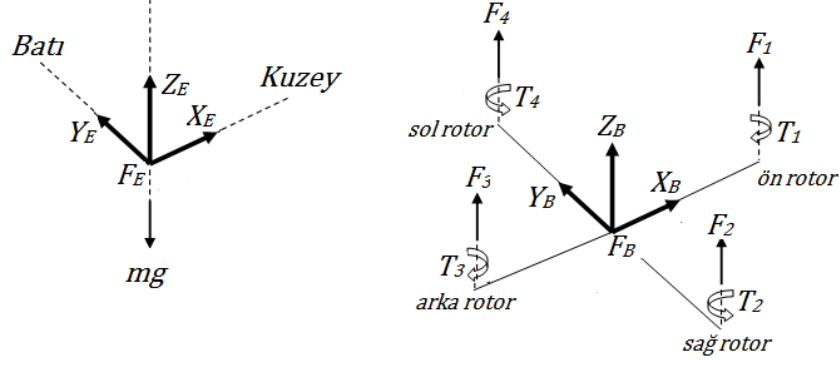
Ayrıca, literatürde yaygın olarak kullanılan sabit kazançlı LQR kontrolcü, karşılaştırma amacıyla oluşturulmuştur. Karşılaştırmaların gerçekçi olması amacıyla, LQR ve LQT kontrolcülerde aynı dinamik model ve performans indeksi matrisleri ( $Q$  ve  $R$ ) kullanılmıştır.

LQT ve LQR kontrolcüler, MATLAB/Simulink ortamında modellenmiştir. Kontrolcüler, simülasyon modelinde denenirken, daha gerçekçi sonuçlar almak amacıyla, doğrusal olmayan dinamik model kullanılarak test edilmiştir.

Simülasyon sonuçları, LQT kontrolcününün kompleks yörüngeleri, LQR'a göre daha yüksek doğrulukta takip ettiğini göstermiştir. Ayrıca LQT kontrolcününün, özellikle keskin dönüşler içeren yörüngelerde, LQR'a göre daha düşük enerji tükettiği gözlenmiştir.

## DİNAMİK MODEL

Bu kısımda, Newton hareket denklemleri kullanılarak, dört rotorlu'nun doğrusal olmayan dinamik modeli elde edilmiştir. Dinamik model çıkarılmadan önce, kullanılan referans koordinat sistemleri tanımlanmalıdır.



Şekil 1: Referans koordinat sistemleri  $F_B$  ve  $F_E$ , rotorlar tarafından üretilen kuvvet ( $F$ ) ve dönme momentleri ( $T$ ) [Suiçmez ve Kutay, 2014].

### Referans Koordinat Sistemleri

Referans koordinat sistemi olarak, gövde sabit koordinat sistemi ( $F_B$ ) ve yer sabit koordinat sistemi ( $F_E$ ) kullanılmıştır. Gövde sabit koordinat sistemi dört rotorlu'nun ağırlık merkezine sabitlenmiştir ve dört rotorlu ile birlikte hareket etmekte ve dönmektedir [Suiçmez ve Kutay, 2014]. Yer sabit koordinat sistemi ise eylemsiz koordinat sistemi olarak kullanılacaktır ve dört rotorlu'nun dünya üzerindeki pozisyon ve yönelimini belirlemektedir [Etkin, 2005; Suiçmez ve Kutay, 2014]. Koordinat sistemleri ile ilgili ayrıntılı bilgi Şekil 1'de verilmiştir.

### "AscTech Hummingbird" İHA ve Motor Katsayıları

Bu çalışmada kullanılan dört rotorlu İHA, "Ascending Technologies" firması tarafından üretilen "AscTech Hummingbird" hava aracıdır [AscTech Hummingbird quadrotor, 2014]. Şekil 2'de görüldüğü üzere "AscTech Hummingbird" LL (low-level) ve HL (high-level) olarak adlandırılan iki farklı mikro işlemciye sahiptir. MATLAB/Simulink ortamında modellenmiş kontrolcüler, HL mikro işlemciye gömülerek, gerçek zamanlı deneylerle doğrulanabilmektedir.

Literatürde, "AscTech Hummingbird" kullanılarak yapılan çalışmalarda, rotorlar tarafından üretilen kuvvet ( $F$ ), dönme momenti ( $T$ ) ve rotorların açılma hızı ( $\omega$ ) arasındaki ilişki denklem (1) ve (2)'de belirtilen şekilde elde edilmiştir [Achtelik, 2010].

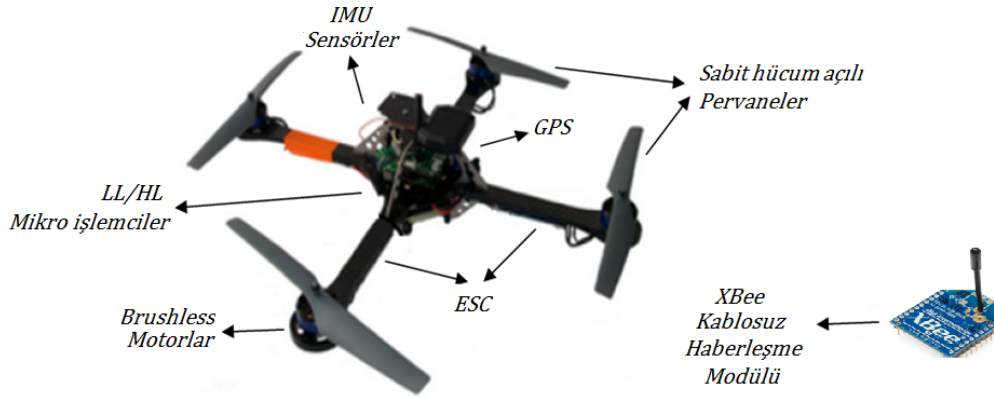
$$F_i = \omega_i^2 k_n, \quad i = 1, 2, 3, 4 \quad (1)$$

$$T_i = F_i k_m, \quad i = 1, 2, 3, 4 \quad (2)$$

"AscTech Hummingbird" için, denklem (1)'deki  $k_n$  değeri  $5.7 \cdot 10^{-8} \text{ N/rpm}^2$ , denklem(2)'deki  $k_m$  değeri ise  $k_m = 0.016 \text{ m}$  olarak bulunmuştur [Achtelik, 2010].

### Kontrol Girdileri

Dinamik model çıkarılmadan önce kontrol girdileri tanımlanmalıdır. 4 farklı kontrol girdisi kullanılmaktadır ve denklem (3), (4), (5), (6)'daki şekilde tanımlanmıştır.



Şekil 2: "AscTech Hummingbird" dört rotorlu İHA [AscTech Hummingbird quadrotor,2014]

$$U_1 = F_1 + F_2 + F_3 + F_4 \quad (3)$$

$$U_2 = F_4 - F_2 \quad (4)$$

$$U_3 = F_3 - F_1 \quad (5)$$

$$U_4 = T_2 + T_4 - T_1 - T_3 \quad (6)$$

Denklem (3)'de görüldüğü üzere,  $U_1$  kontrol girdisi rotorlar tarafından üretilen toplam kuvvete eşittir ve direkt olarak  $z$  yönündeki doğrusal hareket ile ilgilidir.  $U_2, U_3, U_4$  ise sırasıyla yalpalama( $\phi$ ), yunuslama( $\theta$ ) ve sapma( $\psi$ ) hareketiyle, bir başka deyişle, rotasyonel hareket ile ilgilidir.

Ayrıca,  $\phi$  ve  $\theta$  Euler açıları istenilen şekilde ayarlanarak,  $U_1$  kontrol girdisinin  $x$  ve  $y$  yönlerindeki bileşenlerinin değiştirilmesi sonucu,  $x$  ve  $y$  yönündeki doğrusal hareket sağlanmaktadır.

### Doğrusal Olmayan Dinamik Model

İlk olarak dört rotorlu'nun dünya yüzeyine göre pozisyonunu ve yönelimini gösteren  $\xi$  ve  $\eta$  vektörleri denklem (7) ve (8)'deki şekilde tanımlanmıştır. Denklem (7) ve (8) yer sabit referans koordinat sisteminde ( $F_E$ ) yazılmıştır.

$$\xi = [x, y, z]^T \quad (7)$$

$$\eta = [\phi, \theta, \psi]^T \quad (8)$$

Ayrıca, dört rotorlu ile birlikte hareket eden ve dönen gövde sabit referans sisteminin ( $F_B$ ), yer sabit referans sistemine ( $F_E$ ) göre doğrusal ve açısal hız vektörleri denklem (9) ve (10)'da tanımlanmıştır. Denklem (9) ve (10) gövde sabit referans sisteminde yazılmıştır.

$$V_B = [u, v, w]^T \quad (9)$$

$$\omega_B = [p, q, r]^T \quad (10)$$

Tablo 1 : Semboller, kısaltmalar ve tanımları.

Sembol	Tanım
$\phi$	Yalpalama açısı
$\theta$	Yunuslama açısı
$\psi$	Sapma açısı
Euler açıları	$\phi, \theta, \psi$
$[p \ q \ r]$	Gövde açısal hızları
$U_1, U_2, U_3, U_4$	Kontrol girdileri
$K_t$	Doğrusal hareket için sürüklenme matrisi
$m$	Dört rotorlu'nun kütlesi
$L_{BE}$	$F_E$ ile $F_B$ arasındaki dönüşüm matrisi
$I_x, I_y, I_z$	Gövde eylemsizlik terimleri
$g$	Yer çekimi ivmesi
$d$	Rotorlar ile kütle merkezi arasındaki kol uzunluğu
$L(k), L_g(k), g(k)$	LQT kontrolcünün zamana göre değişen kazanç matrisleri
$K$	LQR kontrolcünün sabit kazanç matrisi
$k$	Zaman adımı
$\dot{f}(\cdot)$	Birinci dereceden zamana göre türev
$\ddot{f}(\cdot)$	İkinci dereceden zamana göre türev
$\times$	Vektörel çarpım ("cross product")
$N$	Newton
LQT	Linear Quadratic Tracking
LQR	Linear Quadratic Regulator
ARE	Algebraic Riccati Equation

Newton hareket denklemleri kullanılarak, doğrusal ve rotasyonel hareket denklemleri sırasıyla denklem (11) ve (12)'deki şekilde elde edilmiştir [Madani ve Benallegue, 2007]. Denklem (11) ve (12), gövde sabit referans sisteminde yazılmıştır.

$$\sum F_{ext} = m\dot{V}_B + \omega_B \times (mV_B) \quad (11)$$

$$\sum M_{ext} = J\dot{\omega}_B + \omega_B \times (J\omega_B) \quad (12)$$

Denklem (11) ve (12)'deki  $\sum F_{ext}$  ve  $\sum M_{ext}$  terimleri dört rotorlu'ya etki eden net kuvvet ve momenti temsil etmektedir ve denklem (13) ve (14)'te tanımlanmıştır. Denklem (11) ve (12) gövde sabit referans sisteminde yazıldığı için,  $\sum F_{ext}$  ve  $\sum M_{ext}$  terimleri de gövde sabit referans sisteminde yazılmıştır.

$$\sum F_{ext} = -L_{BE} m \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ U_1 \end{bmatrix} - K_t V_B \quad (13)$$

$$\sum M_{ext} = \begin{bmatrix} U_2 \ d \\ U_3 \ d \\ U_4 \end{bmatrix} \quad (14)$$

Gerekli matematiksel işlemler ve sadeleştirmeler yapıldıktan sonra dört rotorlu'nun doğrusal olmayan dinamik modeli denklem (15) ve (16)'daki şekilde elde edilmiştir.

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = - \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} + L_{EB} \begin{bmatrix} 0 \\ 0 \\ U_1/m \end{bmatrix} - (K_t/m) \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} \quad (15)$$

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} (I_y - I_z)qr/I_x \\ (I_z - I_x)pr/I_y \\ (I_x - I_y)pq/I_z \end{bmatrix} + \begin{bmatrix} U_2d/I_x \\ U_3d/I_y \\ U_4/I_z \end{bmatrix} \quad (16)$$

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin(\phi)\tan(\theta) & \cos(\phi)\tan(\theta) \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi)/\cos(\theta) & \cos(\phi)/\cos(\theta) \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$

Denklem (16)'da görüldüğü üzere,  $[\ddot{\phi} \ \ddot{\theta} \ \ddot{\psi}]^T$  ile  $[\dot{p} \ \dot{q} \ \dot{r}]^T$  arasındaki dönüşüm matrisi çok fazla sayıda trigonometrik terim ve türevini içermektedir. Bu nedenle, kontrolcü tasarımı kolaylaştırmak adına, denklem (16)'da elde edilen doğrusal olmayan rotasyonel dinamik model basitleştirilmiştir. Basitleştirme, dört rotorlu'nun "hover" durumundan çok fazla sapmadığı durumda, denklem (17)'de tanımlanan eşitlikler varsayılarak yapılmıştır [Bouabdallah, 2007].

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}, \quad \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} \quad (17)$$

Sonuç olarak, kontrolcü tasarımında kullanılmak üzere, basitleştirilmiş rotasyonel dinamik model denklem (18)'deki şekilde elde edilmiştir.

$$\begin{bmatrix} \ddot{\phi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} (I_y - I_z)\dot{\theta}\dot{\psi}/I_x \\ (I_z - I_x)\dot{\phi}\dot{\psi}/I_y \\ (I_x - I_y)\dot{\phi}\dot{\theta}/I_z \end{bmatrix} + \begin{bmatrix} U_2d/I_x \\ U_3d/I_y \\ U_4/I_z \end{bmatrix} \quad (18)$$

Kontrol sistemleri simülasyonlar aracılığıyla test edilirken, dinamik model olarak, denklem (15) ve (16)'da elde edilen doğrusal olmayan ve basitleştirilmemiş dinamik model kullanılmıştır. Bu sayede denklem (17)'deki varsayımların geçerliliği test edilmiş ve daha gerçekçi sonuçlar elde edilmiştir.

## LQT KONTROLÇÜ

LQT kontrolcünün zamana göre değişen optimum kontrol kazançları, [Naidu, 2013]'de tanımlanan algoritma, MATLAB ortamında zamanda tersine integral alınarak elde edilmiştir. Daha önce de belirtildiği gibi, sabit kazançlı LQR yönteminin aksine, LQT yönteminde kontrol kazanç matrisleri, takip edilmesi istenilen yörüngeye göre optimize edilmiştir ve zamana göre değişmektedir.

Denklem (24)'te görüldüğü üzere, LQT algoritması ayrık zamanlı olarak tanımlanmıştır. Ayrıca, LQT doğrusal bir kontrol yöntemi olduğu için, dinamik modelin doğrusallaştırılması da gerekmektedir. Bu nedenle, ilk olarak, dört rotorlu'nun doğrusal olmayan dinamik modeli, doğrusallaştırılmalı ve ayrık zamanlı olarak yazılmalıdır.

## Doğrusallaştırma ve Ayrık Zamanlı Durum-Uzay Denklemleri

Denklem (15) ve (18)'de elde edilen doğrusal olmayan model "MATLAB Symbolic Toolbox" kullanılarak "hover" durumu etrafında doğrusallaştırılmıştır. Doğrusal dinamik model, denklem (19)'daki şekilde sürekli zamanlı durum-uzay modeli olarak yazılmıştır.

$$\dot{X} = AX + BU, \quad Y = CX \quad (19)$$

$$X = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \\ x_{10} \\ x_{11} \\ x_{12} \end{pmatrix} = \begin{pmatrix} \phi \\ \dot{\phi} \\ \theta \\ \dot{\theta} \\ \psi \\ \dot{\psi} \\ x \\ \dot{x} \\ y \\ \dot{y} \\ z \\ \dot{z} \end{pmatrix}, \quad Y = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \\ x_{10} \\ x_{11} \\ x_{12} \end{pmatrix}, \quad U = \begin{pmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{pmatrix} \quad (20)$$

Daha sonrasında, denklem (19)'da elde edilen sürekli zamanlı durum-uzay denklemleri, 0.01 saniyelik zaman aralığı seçilerek, ayrık zamanlı hale dönüştürülmüştür. Sonuç olarak dört rotorlu'nun dinamiği, ayrık zamanlı durum-uzay sistemi olarak denklem (21)'deki şekilde elde edilmiştir.

$$X(k+1) = A_d X(k) + B_d U(k), \quad Y(k) = C_d X(k) \quad (21)$$

Denklem (21)'de  $A_d$ ,  $B_d$  ve  $C_d$  sırasıyla, ayrık zamanlı durum, girdi ve çıktı matrisleridir. Benzer şekilde  $X(k)$ ,  $U(k)$  ve  $Y(k)$  sırasıyla, ayrık zamanlı durum, girdi ve çıktı vektörleridir. Denklem (21)'deki  $k$  terimi zaman adımını belirtmektedir ( $k = 1, 2, \dots, k_f \mid k \in \mathbb{Z}^+$ ).

### Performans İndeksinin Seçilmesi

Minimize edilmeye çalışılan performans indeksi denklem (22)'deki şekilde tanımlanmıştır [Naidu, 2013].

$$J = \frac{1}{2} [C_d X(k_f) - r(k_f)]^T F [C_d X(k_f) - r(k_f)] + \frac{1}{2} \sum_{k=k_0}^{k_f-1} ([C_d X(k) - r(k)]^T Q [C_d X(k) - r(k)] + U^T(k) R U(k)) \quad (22)$$

Denklem (22)'deki  $Q$  ve  $R$  matrisleri sırasıyla, durum ve kontrol ağırlık matrisleridir ve denklem (23)'deki şekilde seçilmiştir.  $Q$  ve  $R$  matrisleri seçilirken, dört rotorlu'nun performans sınırları aşılmadan, istenilen yolu yüksek doğrulukta takip etmesi amaçlanmıştır.

$$\left. \begin{aligned} Q &= \text{diag}[100, 50, 10, 5, 0, 0, 100, 1, 100, 1, 1000, 0.1] \\ R &= \text{diag}[10, 0, 0, 0] \end{aligned} \right\} \quad (23)$$

Dört rotorlu'nun en son andaki durumu( $X(k_f)$ ) serbest olduğu için, denklem (22)'deki  $F$  matrisi  $12 \times 12$ 'lik bir sıfır matrisidir. Ayrıca, sınır koşulu olarak, ilk zamandaki( $k_0$ ) "hover" durumu kullanılmıştır.

Denklem (22)'deki  $r(k)$  vektörü, her bir zaman adımı( $k$ ) için, istenilen durum vektörünü temsil etmektedir. Bir başka deyişle, minimize edilmeye çalışılan performans indeksi ( $J$ ), takip edilmesi istenilen yörünge bilgisini( $r(k)$ ) içermektedir. Bu nedenle, LQT algoritması kullanılarak elde edilen kontrol kazanç matrisleri, istenilen yörüngeyi takip etmek için optimize edilmiştir ve çevrimdışı olarak hesaplanmaktadır. Öte yandan, denklem (26)'da belirtilen sabit kazançlı LQR kontrolcüde, performans indeksi yörünge bilgisini içermemektedir ve bu nedenle LQR kontrolcününün optimum kontrol kazanç matrisi zamana göre değişmemektedir.

### LQT Algoritması

[Naidu, 2013]'de tanımlanan ayrık zamanlı LQT algoritması denklem (24)'teki şekilde yazılmıştır.

$$\left. \begin{aligned} P(k) &= A_d^T P(k+1) [I + EP(k+1)]^{-1} A_d + V \\ &V = C_d^T Q C_d \\ &E = B_d R^{-1} B_d^T \\ g(k) &= [A_d^T - A_d^T P(k+1) [I + EP(k+1)]^{-1} E] \\ &g(k+1) + C^T Q r(k) \\ X^*(k+1) &= [A_d - B_d L(k)] X^*(k) + \\ &B_d L_g(k) g(k+1) \\ L(k) &= [R + B_d^T P(k+1) B_d]^{-1} B_d P(k+1) A_d \\ L_g(k) &= [R + B_d^T P(k+1) B_d]^{-1} B_d^T \\ U^*(k) &= -L(k) X^*(k) + L_g(k) g(k+1) \end{aligned} \right\} \quad (24)$$

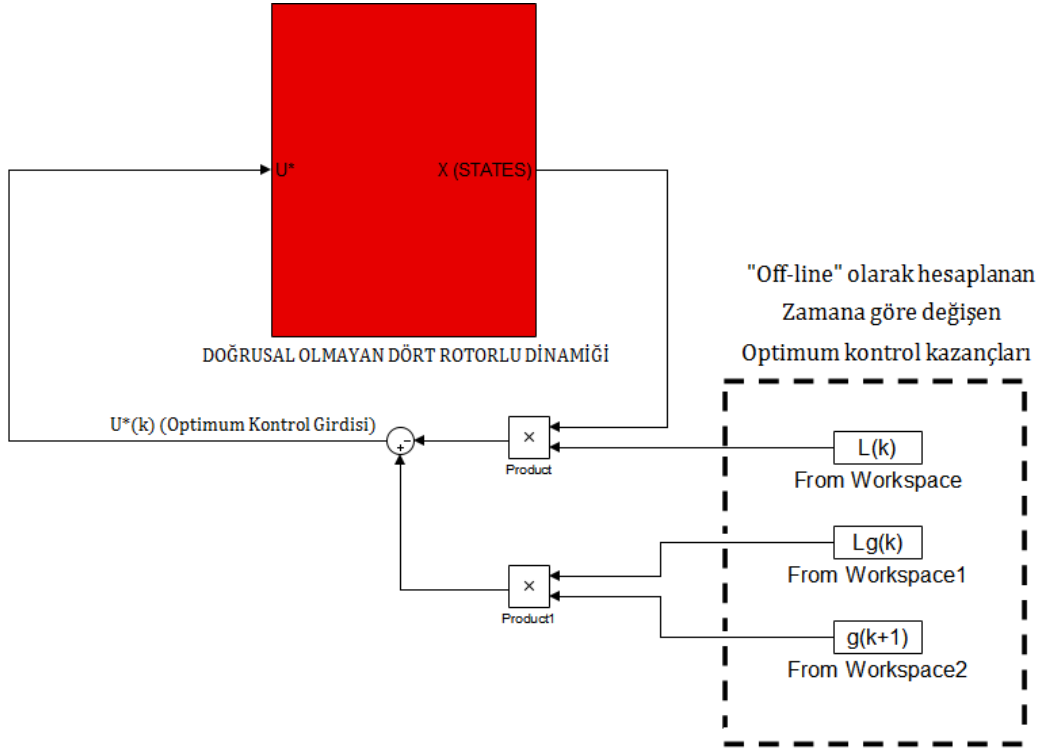
Denklem (24)'te tanımlanan LQT algoritması, MATLAB ortamında zamanda tersine integral alınarak çözülmüştür. Çözüm, 0.01 saniyelik sabit zaman aralığı kullanılarak elde edilmiştir. Başlangıç noktası olarak, denklem (25)'te tanımlanan, son zamandaki( $k_f$ ) sınır koşulları kullanılmıştır [Naidu, 2013].

$$\left. \begin{aligned} P(k_f) &= C_d^T F C_d \\ g(k_f) &= C_d^T F r(k_f) \end{aligned} \right\} \quad (25)$$

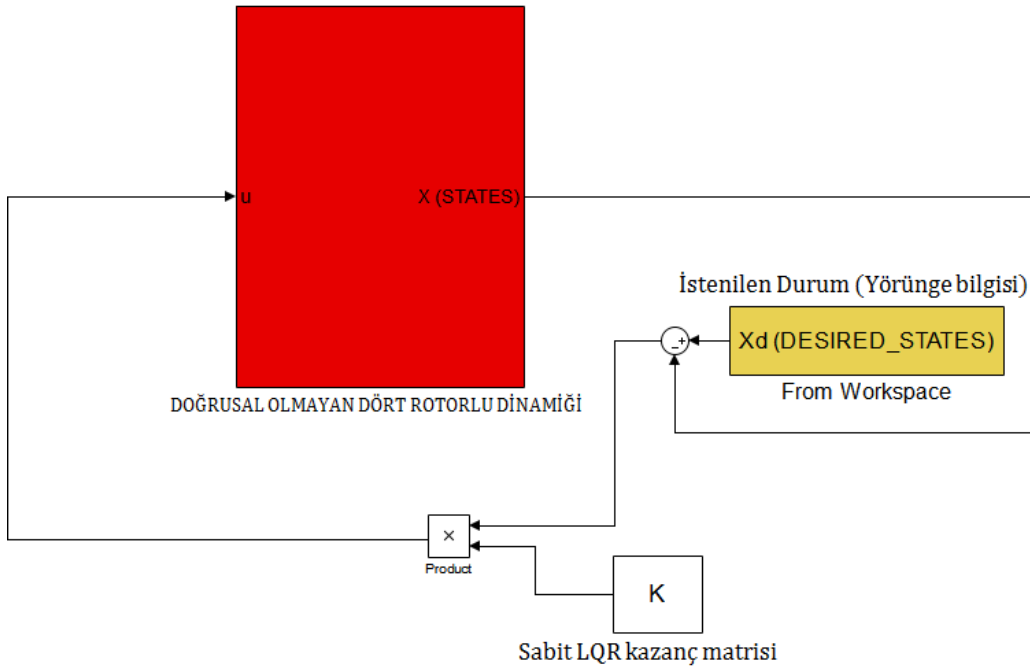
Denklem (24)'teki  $L(k)$ ,  $L_g(k)$ ,  $g(k)$  matrisleri, her bir zaman adımı için elde edilen optimum kontrol kazançlarıdır ve çevrimdışı olarak elde edilmiştir. Denklem (24)'ün 4. eşitliğinde görüldüğü üzere, yörünge bilgisini içeren  $r(k)$  terimi,  $g(k)$  eşitliğinde doğrudan yer almaktadır. Bu nedenle,  $g(k)$ , istenilen yörüngeyi takip edilmesi için optimize edilmektedir.

Şekil 3, LQT kontrolcününün basitleştirilmiş MATLAB/Simulink modelini göstermektedir. Şekil 3'te görüldüğü üzere, çevrimdışı("off-line") olarak elde edilen optimum kontrol kazançları





Şekil 3: LQT kontrolcünün MATLAB/Simulink modeli.



Şekil 4: LQR kontrolcünün MATLAB/Simulink modeli.

$L(k)$ ,  $L_g(k)$ ,  $g(k)$ , simulink modeline ekstrapolasyon kullanılarak aktarılmaktadır. Optimum kontrol kazançları istenilen yörüngeyi takip etmek amacıyla çevrimdışı olarak hesaplandığı için, LQT kontrolcünde istenilen yörünge bilgisi sisteme girilmemektedir. Öte yandan, LQR kontrolcünün MATLAB/Simulink modelini gösteren Şekil 4'te görüldüğü üzere, sabit kazançlı LQR kontrolcünün, istenilen yörüngeyi takip etmesi için, yörünge bilgisinin sisteme girdi olarak verilmesi gerekmektedir.

## LQR Kontrolcü

Sabit kazançlı LQR kontrolcü, bu çalışmada kullanılan ve sabit kazançlı LQR'a göre daha avantajlı sonuç vermesi beklenen LQT yöntemi ile elde edilen sonuçları karşılaştırmak amacıyla kullanılmıştır. LQR kontrolcününün sabit kazanç matrisi MATLAB'ın "lqr" fonksiyonu [The MathWorks, 2014] kullanılarak elde edilmiştir ve ayrıca ARE(Algebraic Riccati Equations) çözümlere teyit edilmiştir. LQR kontrolcü ile ilgili ayrıntılı bilgi [The MathWorks, 2014]'te yer almaktadır.

Denklem (26)'da,  $u = -KX$  kontrol girdisi kullanılarak minimize edilmeye çalışılan performans indeksi tanımlanmıştır [The MathWorks, 2014]. Denklem (26)'daki  $Q$  ve  $R$  matrisleri, karşılaştırmaların eşit koşullarda yapılabilmesi için, LQT kontrolcünde kullanılan ve denklem (23)'de tanımlanan değerlerle aynıdır. Aynı zamanda, LQT ve LQR kontrolcülerin elde edilmesinde kullanılan ve dört rotorlu'nun doğrusallaştırılmış dinamik modelini belirten durum-uzay denklemleri de birbiriyle aynıdır.

$$J_u = \int_0^{\infty} (X^T Q X + u^T R u) dt, \quad | \quad \dot{X} = AX + Bu, \quad Y = CX, \quad u = -KX \quad (26)$$

Denklem (26)'daki  $u = -KX$  kontrol girdisi, istenilen bir yörüngeyi takip etmek yerine, sistemi regule ederek denge noktasına ulaştırmaktadır. Bu nedenle, LQR kontrolcünde kullanılan kontrol girdisi( $u$ ), denklem (27)'deki şekilde değiştirilmiştir. Denklem 27'deki  $X_d$  terimi, istenilen durum vektörünü temsil etmektedir ve takip edilmesi istenilen yörünge bilgisini içermektedir.

$$u = -K(X - X_d) \quad (27)$$

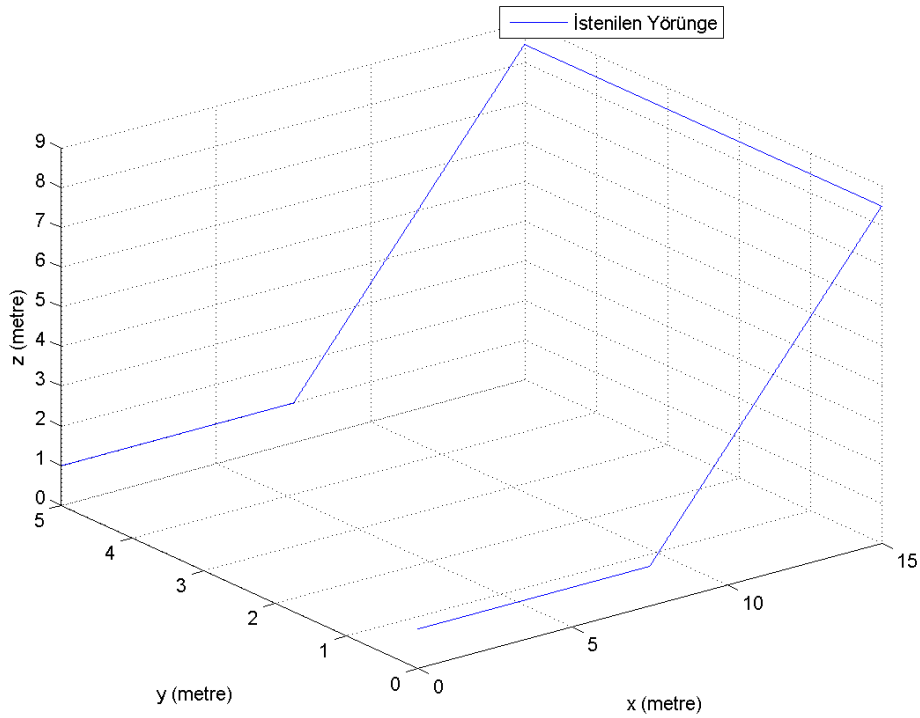
LQR kontrolcününün MATLAB/Simulink modelini gösteren Şekil 4'te de görüldüğü üzere, takip edilmesi istenilen yörünge bilgisi LQR kontrolcüye girdi olarak verilmektedir. LQT kontrolcünde ise, kontrol kazanç matrisleri( $L, L_g, g$ ) istenilen yörüngeyi takip etmek amacıyla optimize edilerek çevrimdışı olarak hesaplanmıştır ve yörünge bilgisini içermektedir. Bu nedenle, Şekil 3'te de görüldüğü üzere, LQT kontrolcüye takip edilmesi istenilen yörünge bilgisi girilmemektedir.

## SONUÇLAR VE ANALİZ

Bu bölümde, elde edilen LQT kontrolcü MATLAB/Simulink aracılığıyla denenmiştir ve simülasyon sonuçları sabit kazançlı LQR kontrolcüyle karşılaştırılmıştır. LQT ve LQR kontrolcülerin basitleştirilmiş MATLAB/Simulink modelleri sırasıyla Şekil 3 ve 4'te gösterilmiştir. İlk olarak takip edilmesi istenilen yörünge MATLAB ortamında oluşturulmalıdır.

### Takip Edilmesi İstenilen Yörünge

Takip edilmesi istenilen yörünge 0.01 saniyelik zaman aralığı kullanılarak oluşturulmuştur. Şekil 5'te görüldüğü üzere, dört rotorlu 0-7 saniyeleri arasında sadece x ekseninde doğrusal hareket etmektedir. 7-15 saniyeleri arasında yörüngeye z ekseninde tırmanma eklenmektedir. 15. saniyede ise, y ekseninde 5 metrelik bir "step input" verilmektedir. 15-23 saniyeleri arasında z ekseninde alçalma gerçekleşmektedir. Son olarak 23-30 saniyeleri arasında x ekseninde doğrusal hareket yapılarak, 30. saniyede takip edilmesi istenilen yörünge tamamlanmaktadır.



Şekil 5: Takip edilmesi istenilen yörünge.

Tablo 2 : LQT ve LQR kontrolcülerin denge durumundaki hata değerleri.

	Hata $x$ (metre)	Hata $y$ (metre)	Hata $z$ (metre)
LQT	0.0093	0	0.0101
LQR	0.6004	0	0.3102

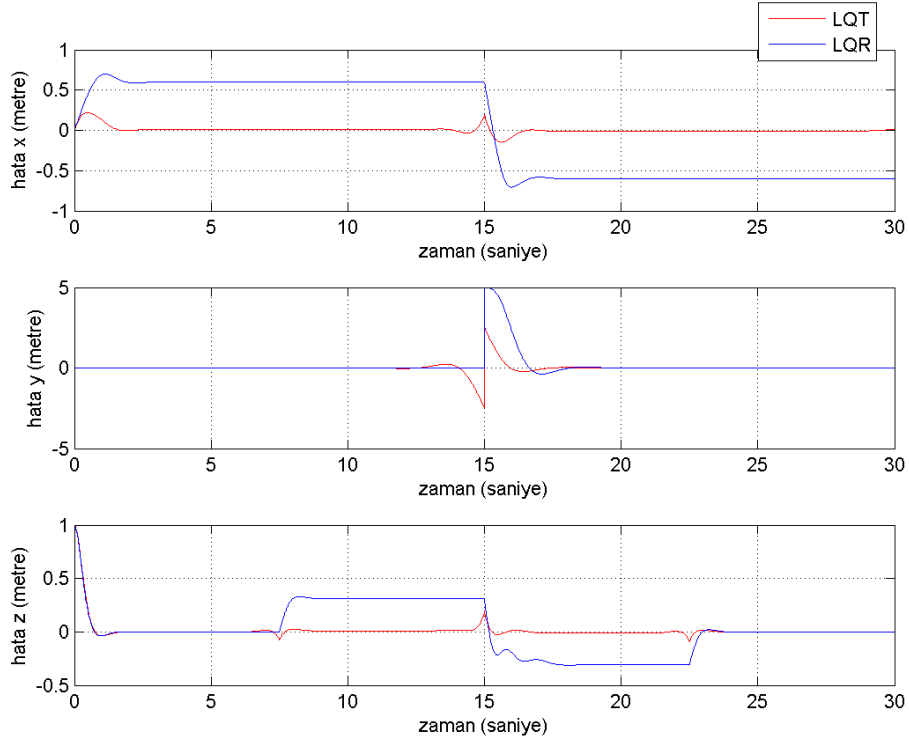
### Karşılaştırmalı Simülasyon Sonuçları

Bu kısımda, yörünge takibi için optimize edilmiş LQT kontrolcü ile sabit kazançlı LQR kontrolcü kullanılarak elde edilen simülasyon sonuçları karşılaştırılacaktır. LQT kontrolcünün zamana göre değişen kazançlarının, yörünge takibi için optimize edilmesi ve çevrimdışı hesaplanmasından dolayı, sabit kazançlı LQR'a göre çeşitli açılardan daha avantajlı olduğu gözlenmiştir.

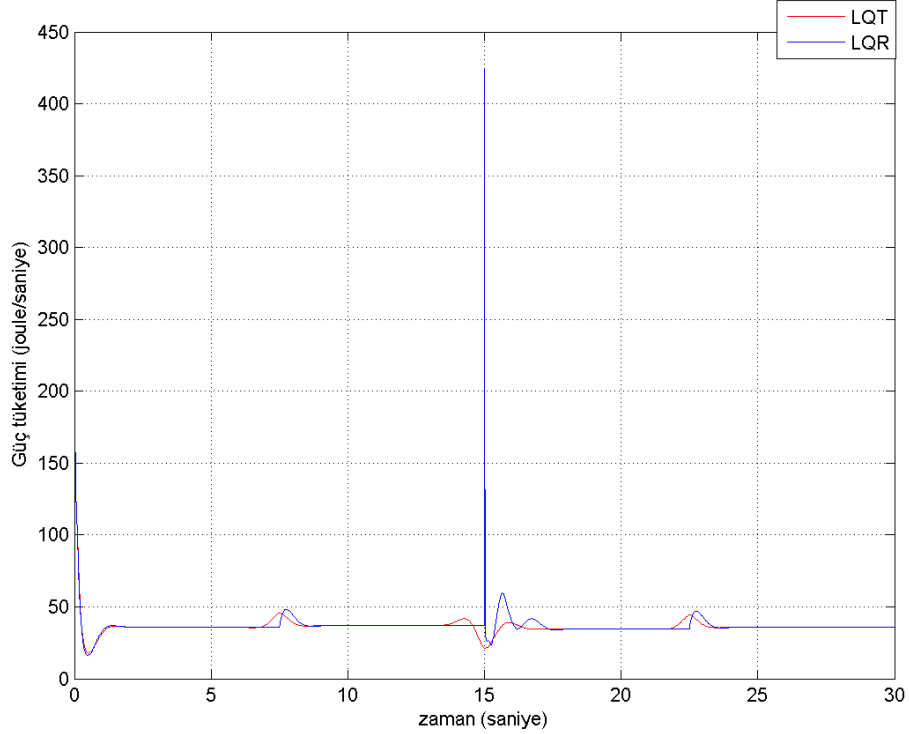
Hata değerleri: Şekil 6, takip edilmesi istenilen yörüngeyi  $x, y, z$  koordinatları ile LQT ve LQR kontrolcüler tarafından elde edilen  $x, y, z$  koordinatları arasındaki hata değerlerini göstermektedir. Şekil 6'da görüldüğü üzere, LQT kontrolcü istenilen yörüngeyi çok daha doğru bir şekilde takip etmektedir. Tablo 2'de görüldüğü üzere, LQT kontrolcünün denge durumundaki hata değerleri, yaklaşık olarak  $x$  ekseninde 60 kat,  $z$  ekseninde ise 30 kat daha azdır. Ayrıca, LQR kontrolcünün hata değerlerinin yörünge hızına göre arttığı da gözlenmiştir.

Sonuç olarak, LQT kontrolcünün LQR'a göre, istenilen yörüngeyi daha yüksek doğrulukta takip ettiği gözlenmiştir. LQT kontrolcünün daha doğru sonuç vermesi, sabit kazanç matrisi kullanan LQR'ın aksine, çevrimdışı hesaplanmış ve yörünge takibi için optimize edilmiş kazanç matrisleri kullanmasına bağlanabilir.

Enerji tüketimi: Bu kısımda LQT ve LQR kontrolcüler kullanılarak, yörünge takibi sırasında motorlar tarafından harcanan enerji değerleri karşılaştırılacaktır. Şekil 7, LQT ve LQR kontrolcülerin anlık güç tüketimi göstermektedir. Her bir motor için güç tüketimi değerleri denklem (28) kullanılarak hesaplanmıştır. Denklem (28)'deki  $P$  joule/saniye cinsinden anlık güç tüketimini,  $T$



Şekil 6: Takip edilmesi istenilen yörünge ile kontrolcüler tarafından elde edilen yörünge arasındaki hata değerleri.



Şekil 7: Her bir zaman adımında, dört motor tarafından tüketilen toplam güç tüketimi (joule/saniye).

dönme momentini,  $\omega_i$  radyan/saniye cinsinden açısal hızı,  $i$  ise motor numarasını temsil etmektedir.

$$P_i = T_i \omega_i, \quad i = 1, 2, 3, 4 \quad (28)$$

Şekil 7'de görüldüğü üzere, 15.saniyede sisteme y ekseninde 5 metrelik bir "step input" verildiğinde, LQR kontrolcünün güç tüketimi değerleri anlık olarak çok yükselmektedir. Öte yandan, LQT kontrolcünün güç tüketimi değerleri çok daha düşük seviyelerde kalmaktadır. LQT ve LQR kontrolcüler arasındaki bu farklılık Şekil 8'de daha net görülebilir. Şekil 8'de görüldüğü üzere, 15. saniyede sisteme y ekseninde 5 metrelik bir "step input" verilmektedir. LQT kontrolcünün kazanç matrisleri, çevrimdışı hesaplanmış ve istenilen yörüngeye göre optimize edilmiş olduğu için, LQT kontrolcü 15. saniyede verilen "step input"'a daha önceden tepki verebilmektedir. LQR kontrolcü ise sabit kazançlı ve yörünge takibi için optimize edilmemiş kazanç matrisi kullandığı için, "step input"'a ancak 15. saniyede tepki verebilmektedir. Bu nedenle, Şekil 7'de görüldüğü üzere, LQR kontrolcünün güç tüketimi değerleri 15. saniyeden itibaren çok yüksek seviyelere çıkmaktadır.

LQT ve LQR kontrolcülerin, 15. saniyede sisteme y ekseninde verilen 5 metrelik keskin dönüşü("step input") tamamlamaları sırasında tükettikleri toplam enerji değerleri denklem (29) ve (30)'daki şekilde elde edilmiştir. Enerji değerleri, anlık güç tüketiminin 13-18 saniyeleri arasında integralinin alınmasıyla elde edilmiştir. Karşılaştırmaların anlamlı olabilmesi için, her iki kontrolcünün keskin dönüşü başlama ve bitirme zamanları dikkate alınarak, integralin zaman aralığı 13-18 saniyeleri olarak seçilmiştir.

$$E_{lqt} = \sum_{i=1}^4 \int_{13}^{18} P_{lqt,i} dt = 175.5069 \quad [joule] \quad (29)$$

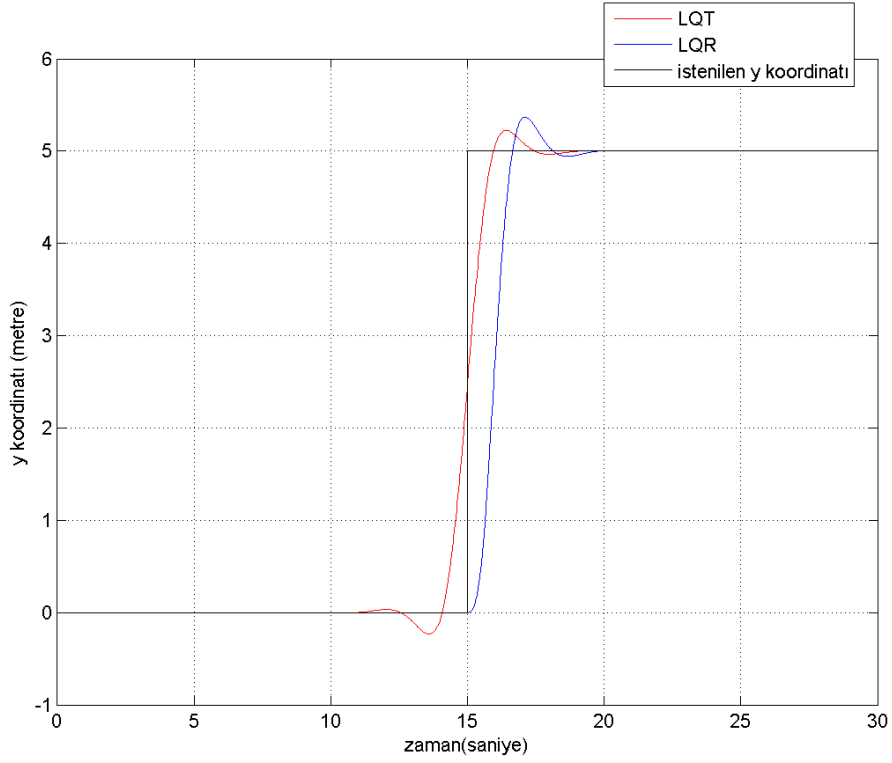
$$E_{lqr} = \sum_{i=1}^4 \int_{13}^{18} P_{lqr,i} dt = 193.0112 \quad [joule] \quad (30)$$

Denklem (29) ve (30)'da elde edilen değerlere göre, LQT kontrolcünün y eksenindeki keskin dönüşü("step input") LQR'a göre **9.9735 %** daha az enerji harcayarak gerçekleştirebildiği gözlenmiştir.

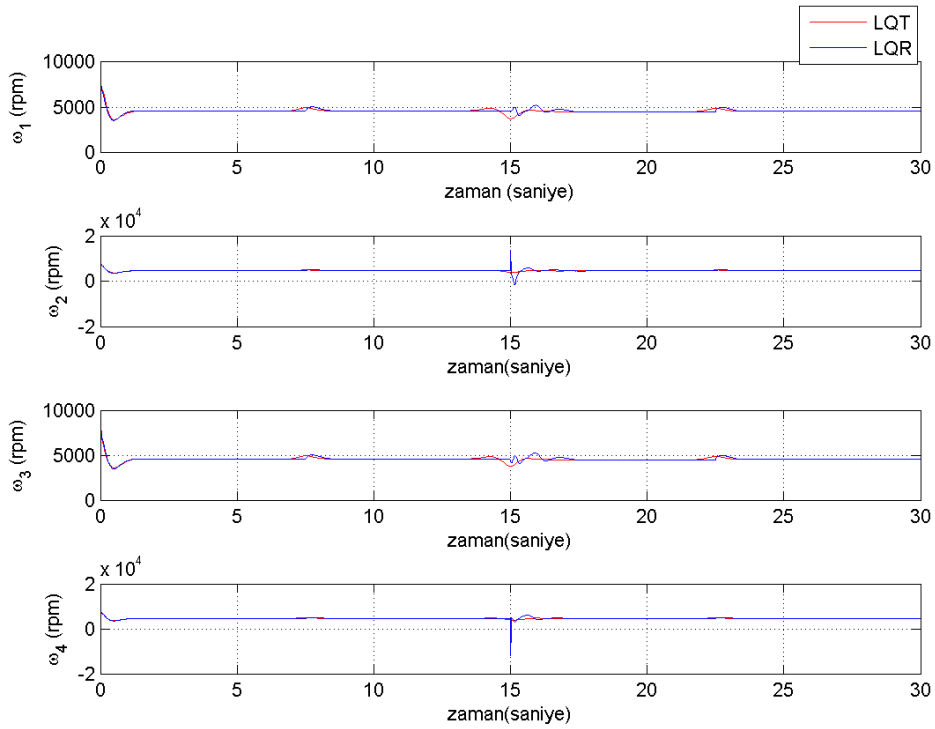
Sonuç olarak, özellikle keskin manevralar içeren yörüngelerin takibinde, LQT kontrolcünün enerji tüketimi açısından sabit kazançlı LQR'a göre daha verimli olduğu söylenebilir. LQT kontrolcünün verimi, keskin manevraların sayısına ve büyüklüğüne bağlı olarak daha yüksek seviyelere çıkabilmektedir.

Motorların doygunluğa ulaşması: Bu çalışmada kullanılan "AscTech Hummingbird" dört rotorlu aracı için rotorlar tarafından ulaşılabilen maksimum açısal hız değerleri 8000 rpm civarındadır [AscTech Hummingbird quadrotor, 2014]. Bu nedenle, elde edilen kontrolcülerin ileriki aşamalarda gerçek zamanlı deneylerle doğrulanabilmesi için, açısal hız değerlerinin 8000 rpm'i geçmemesi gerekmektedir. Simülasyonlardaki motor dinamiğinde saturasyon kullanılarak, açısal hız değerleri sınırlandırılabilir. Fakat LQT ve LQR yöntemleri doğrusal kontrol teknikleri olduğu için, sisteme eklenen doğrusal olmayan etkiler(örnek:saturasyon), kontrolcülerin performansını düşürmektedir. Bu nedenle, LQT ve LQR kontrolcülerin motor dinamiğinde saturasyon kullanılmamıştır.

Şekil 9'da görüldüğü üzere, LQT kontrolcünün maksimum açısal hız değerleri 7947 rpm iken, LQR kontrolcü 13603 rpm değerlerine ulaşmaktadır. Bu nedenle, LQT kontrolcünün, motor dinamiğini otomatik olarak sınırlayabildiği söylenebilir. Maksimum açısal hız değerleri, denklem (23)'de tanımlanan kontrol ağırlık matrisi ( $R$ ) değiştirilerek daha düşük seviyelere çekilebilir. Fakat bu



Şekil 8: İstenilen ve kontrolcüler tarafından elde edilen y koordinatı.



Şekil 9: Pervane açısal hızları (rpm).

durum kontrolcülerin daha yavaş performans göstermesine sebep olmaktadır. LQT ve LQR kontrolcüler, aynı  $R$  matrisi kullanılarak karşılaştırıldığı için, LQT kontrolcününün bu noktada uygulanabilirlik açısından daha avantajlı olduğu söylenebilir.

## SONUÇ

Bu çalışmada, dört rotorlu bir İHA'nın yörünge takibi, LQT optimum kontrol yöntemi kullanılarak elde edilmiştir [Naidu, 2013]. Ayrıca, MATLAB/Simulink ortamında modellenen simülasyonlar aracılığıyla, LQT kontrolcü denenmiş ve literatürde sıkça kullanılan sabit kazançlı LQR kontrolcüyle karşılaştırılmıştır.

Kullanılan LQT yöntemi, takip edilmesi istenilen yörüngeye göre optimize edilen ve çevrimdışı hesaplanan kontrol kazançları kullanılmaktadır. Bu nedenle LQT'nin, literatürde yaygın olarak kullanılan sabit kazançlı LQR yöntemine göre, daha avantajlı sonuçlar vermesi beklenilmiştir.

Elde edilen simülasyon sonuçları, LQT 'nin LQR'a göre, özellikle çok sayıda keskin manevra içeren yörüngelerin takibinde, çeşitli avantajlarının olduğunu göstermiştir. LQT kontrolcünün LQR'a göre, istenilen yörüngeyi daha doğru ve az enerji harcayarak takip edebildiği görülmüştür. Ayrıca, LQT kontrolcünün, motor dinamiğini otomatik olarak sınırladığı (saturasyon) ve LQR'a göre uygulanabilirlik açısından avantajlı olabileceği gözlenmiştir. LQT kontrolcünün en belirgin dezavantajı olarak, optimum kontrol kazançlarının çevrimdışı şekilde, her bir yörünge için hesaplanması gerektiği söylenebilir. Fakat, Şekil 4'de görüldüğü üzere, LQR kontrolcü için de yörünge bilgisi çevrimdışı olarak kontrolcüye girilmektedir. LQT kontrolcünde ise çevrimdışı hesaplanan optimum kontrol kazançları yörünge bilgisini dolaylı olarak içermektedir. Bu nedenle, Şekil 3 ve 4'te görüldüğü üzere, LQT kontrolcünün çevrimdışı hesaplanan optimum kontrol kazançları ( $L, L_g, g$ ), LQR kontrolcüye çevrimdışı olarak girilen yörünge bilgisi ile aynı anlamda değerlendirilebilir.

İleriki çalışmalarda, elde edilen LQT kontrolcü, "AscTech Hummingbird" dört rotorlu İHA kullanılarak gerçek zamanlı deneyler aracılığıyla da doğrulanabilir. Ayrıca, LQT kontrolcü modifiye edilerek, daha kompleks ve denge durumundan (hover) çok daha fazla sapan yörüngelerin takibi denenebilir.

## Kaynaklar

- [1] Achtelik M., 2010, "Nonlinear and adaptive control of a quadcopter", Dipl.-Ing. Dissertation, Lehrstuhl für Flugsystemdynamik, Technische Universität München, Garching, Almanya.
- [2] AscTech Hummingbird quadrotor, 2014, <http://www.asctc.de>, son ziyaret tarihi Haziran 2014
- [3] Al-Younes Y. M., Al-Jarrah M. A. ve Jhemi A. A., 2010, "Linear vs. nonlinear control techniques for a quadrotor vehicle", *Proceedings of the ISMA10 International Symposium on Mechatronics and its Applications*, Sharjah, UAE, 20-22 Nisan, s.1-10.
- [4] Bouabdallah S., 2007, "Design and control of quadrotors with application to autonomous flying", Ph.D. tezi, Ecole Polytechnique Federale de Lausanne, Lausanne, Fransa.
- [5] Etkin B., 2005, *Dynamics of Atmospheric Flight*, Dover Publications, New York, USA.
- [6] Li Y. ve Song S., 2012, "A survey of control algorithms for quadrotor unmanned helicopter", *Proceedings of the IEEE fifth International Conference on Advanced Computational Intelligence (ICACI)*, Nanjing, Jiangsu, Çin, 18-20 Ekim, s.365-369.
- [7] Madani T. ve Benallegue A., 2007, "Sliding mode observer and backstepping control for a quadrotor unmanned aerial vehicles", *Proceedings of the 2007 American Control Conference*, New York City, USA, 11-13 Temmuz, s.5887-5892.

- [8] Mellinger D. ve Kumar V., 2011, "Minimum Snap Trajectory Generation and Control for Quadrotors", *Proceedings of the IEEE International Conference on Robotics and Automation*, Shanghai, Çin, 9-13 Mayıs, s.2520-2525.
- [9] Minh L. D. ve Ha C., 2010, "Modeling and Control of Quadrotor MAV Using Vision based Measurement", *Proceedings of the International Forum on Strategic Technology*, Ulsan, Güney Kore, 13-15 Ekim, s.70-75.
- [10] Naidu D. S., 2013, *Optimal Control Systems*, CRC Press LLC, Florida, USA.
- [11] Raffo G. V., Ortega M. G. ve Rubio F. R., 2008, "Backstepping/Nonlinear  $H_\infty$  Control for Path Tracking of a QuadRotor Unmanned Aerial Vehicle", *Proceedings of the American Control Conference*, Seattle, Washington, USA, 11-13 Haziran, s.3356-3361.
- [12] Suiçmez E. C. ve Kutay A. T., 2014, "Optimal path tracking control of a quadrotor UAV", *Proceedings of the International Conference on Unmanned Aircraft Systems (ICUAS)*, Orlando, FL, USA, 27-30 Mayıs 2014, s.115-125.
- [13] Suiçmez E. C. ve Kutay A. T., 2014, "Dört rotorlu bir insansız hava aracının geri-adımlama yöntemi ile yol takibi kontrolü", *Havacılık ve Uzay Teknolojileri Dergisi*, cilt:7, sayı:2, Temmuz 2014, ISSN:1304-0448.
- [14] The MathWorks, Inc., 2014, <http://www.mathworks.com/help/control/ref/lqr.html>, son ziyaret tarihi Temmuz 2014
- [15] Zhou Q. L., Zhang Y., Rabbath C. A. ve Theilliol D. , 2010, "Design of Feedback Linearization Control and Reconfigurable Control Allocation with Application to a Quadrotor UAV", *Proceedings of the Conference on Control and Fault Tolerant Systems*, Nice, Fransa, 6-8 Ekim, s.371-376.