

AMATÖR DÖRT ROTORLU HAVA ARAÇLARI İÇİN PYTHON İLE KAYAN KİPLİ KONTROL GELİŞTİRİLMESİ

Abidin Sefa Aslan* ve Oktay Baş†
19 Mayıs Üniversitesi, Samsun

Nurdan Bilgin‡
19 Mayıs Üniversitesi, Samsun

ÖZET

Dört rotor ile tahrik edilen dikey iniş kalkış yapabilen hava araçları günlük hayatta birçok yeni işlev kazanmıştır. Basit mekaniği ve kontrol kolaylığı nedeniyle, sözü edilen araçlar, mühendislik öğrencilerine kontrol uygulamalarını sınamaları açısından olanak tanımaktadır. YÖK'ün tez katologu incelendiğinde neredeyse tüm uygulamaların MATLAB tabanlı olduğu görülmüştür. Oysa ki MATLAB ticari bir ürün olarak fiyat açısından, günlük işlerde kullanılmak üzere üretilecek için araç için uygun değildir. Bu nedenle açık kaynak kodlu yazılımlara yönelmek zorunluluk olarak düşünülmektedir. Bu nedenle PYTHON ile kontrolcü geliştirme fikri ortaya çıkmıştır. Sözü edilen dört rotorlu hava araçları, dış bozuculardan ve belirsizliklerden fazlası ile etkilendikleri için bu etkenlerle başa çıkmak üzere gürbüz kontrolcüler arasında sayılan kayan kipli kontrol tercih edilmiştir. Bu çalışmada, PYTHON ile geliştirilen benzetim algoritması ile önerilen kontrolcünün hava aracını kararlı olarak tutabildiği gösterilmektedir.

GİRİŞ

Dört rotor ile tahrik edilen dikey iniş kalkış yapabilen hava araçları (DRHA) hareketini ve yönelimini dört rotorun farklı oranlarla dönmesi sayesinde gerçekleştirmektedir. Şekil 1'de şematik gösterimi sunulan araçlarda, iki pervane saat yönünde diğer ikisi ise saat yönünün tersine dönmektedir. Bu tip hava araçlarının en önemli özelliği eksik eyletimli (under-actuated) olmalarıdır, yani altı eksenle serbest hareket edebilmesine (6DOF) karşın sadece dört eyleticiye (motor) sahip olmalarıdır. Bu nedenle klasik kontrollerinden farklı arayışlara yönelmek gerekmektedir.

DRHA kontrolü için çok çeşitli kontrol yöntemleri önerilmiştir. Örneğin, [Önkol ve Efe, 2009] oransal integral ve türevsel (PID) kontrol yöntemi, kayan kipli kontrol (KKK), geriadım lamalı kontrol yöntemi ve geribeslemeli doğrusallaştırmalı kontrol yöntemlerinin bir karşılaştırmasını sunmaktadır. YÖK Tez merkezi üzerinden yapılan tarama sonucunda son 6 yıl içerisinde 9 adet doğrusal karesel regülatör (LQR) ve 7 adet de KKK kullanan çalışmaya rastlanmıştır.

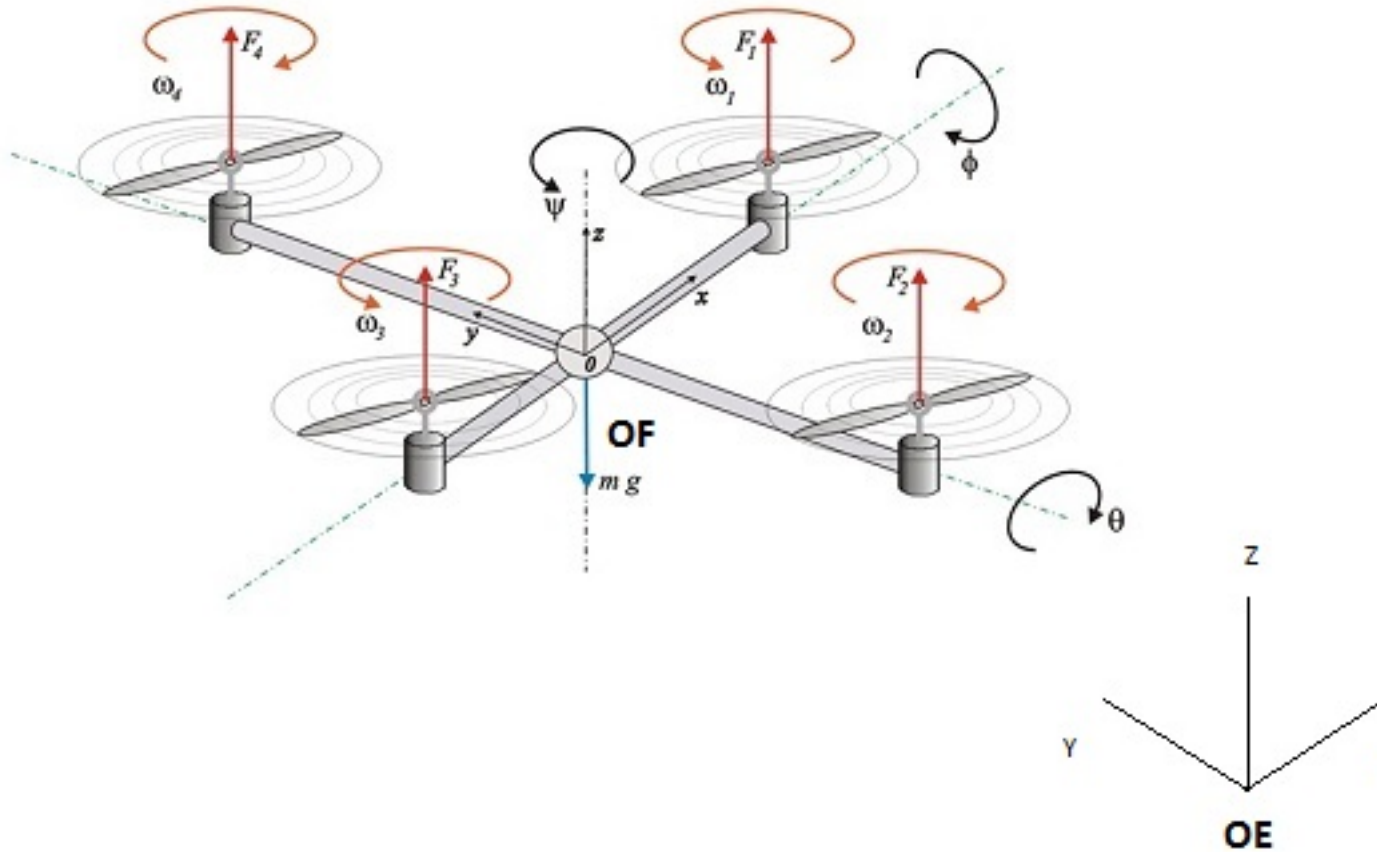
*4. Sınıf Öğrencisi, Mak. Müh. Böl., E-posta: abidinsefa@gmail.com

†4. Sınıf Öğrencisi, Mak. Müh. Böl., E-posta: oktaybas6@gmail.com

‡Yrd. Doç. Dr., Mak. Müh. Böl., E-posta: nurdan.bilgin@omu.edu.tr

Çalışmalardan biri dışında tümünde MATLAB kullanıldığı görülmüştür. Çoğunluğun tercihini etkileyen sözü edilen yazılımın yetkinliğinin yanında; üniversitelerde, eğitim lisanslı sürümlerinin bulunmasıdır. Ancak üretilen ürünün ticarileşme durumunda bu yazılımın maliyeti, ürün maliyetini çok yukarılara çekecektir.

Bu çalışmada [Xu ve Özgüner , 2006]'da verilen KKK modeli esas alınmış ve PYTHON kullanılarak benzetim çalışmaları yapılmıştır. Bu çalışma ilgili referansın yeniden üretimi olarak ele alınmalıdır. Çalışmanın paylaşmaya değer bulunmasının nedeni, açık kaynak kod kullanılarak yetkin kontrol algoritmalarının geliştirilebileceğini gösteren iyi bir örnek oluşudur. [Xu ve Özgüner , 2006]'da önerilen kontrol kuralı, (DRHA) modelini tam-eyletimli (fully-actuated) ve eksik-eyletimli (under-actuated) iki alt gruba bölmektedir. Ardından eksik-eyletimli sistem uygun bir koordinat dönüşümü ile iç içe geçmiş eksik-eyletimli alt sistemlere dönüştürülmektedir. İlgili referans, elde edilen bu eksik-eyletimli alt sistemlerin kararlı hale getirmesi için KKK kullanılabileceğini kanıtlamaktadır. PYTHON ile yapılan benzetim çalışması uyarınca, önerilen KKK ile tam eyletimli alt sistemin kontrolcülerinin, yunuslama (pitch) ve yuvarlanma (roll) açılarını sıfırda tutarken, DRHA'nın istenen bir yan dönme (yaw) açısı ile istenen bir konuma ulaşmasını mümkün kıldığı söylenebilmektedir. KKK yönteminin avantajı, model hatalarına, parametrik belirsizliklere ve diğer bozucuların etkisine karşı kapalı olması olarak ifade edilmektedir.



Şekil 1: Dört rotorlu hava aracı (DRHA) şematik gösterimi [Web Sayfası Erişim-, 13.03.2018].

Bildirinin sunumu şu şekilde gerçekleştirilmiştir. Yöntem bölümünde, DRHA için basitleştirilmiş bir matematiksel modeli verilmekte ve ispatına girilmeksizin ilgili kaynaktan direkt alınarak kontrol yöntemi tanıtılmaktadır. Uygulamalar bölümünde, Kısaça Python tanıtılmakta ardından Python ile yapılan benzetim çalışması ile, önerilen kontrolcünün DRHA'yı kontrol edebildiği gösterilmektedir. Bildiri sonuç bölümüyle bitirilmektedir.

YÖNTEM

Bu bölüm, iki alt başlık altında sunulmaktadır. İlk başlık altında, DRHA'nın basitleştirilmiş bir matematiksel modeli verilmektedir. İkinci başlık altında kontrol yöntemi tartışılmaktadır. Bu alt başlık altında [Xu ve Özgüner , 2006]'de sunulan kontrol yaklaşımı, ispatlarına yer verilmeksizin kontrol algoritması içerisinde kullanıldığı şekliyle anlatılmaktadır.

Matematiksel Model

DRHA'nın dinamik modeli [Altuğ, Ostrowski ve Mahony , 2002]'de Lagrange yaklaşımıyla aşağıdaki gibi elde edilmiştir. Xu ve Özgüner [2006]'de aynı modeli kullandığı için karşılaştırma amacıyla model direkt ilgili kaynaktan alınmıştır.

$$\begin{aligned}\ddot{x} &= u_1(\cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi) - K_1 \dot{x}/m \\ \ddot{y} &= u_1(\sin \phi \sin \theta \cos \psi - \cos \phi \sin \psi) - K_2 \dot{y}/m \\ \ddot{z} &= u_1(\cos \phi \cos \psi) - g - K_3 \dot{z}/m \\ \ddot{\theta} &= u_2 - lK_4 \dot{\theta}/I_1 \\ \ddot{\psi} &= u_3 - lK_5 \dot{\psi}/I_2 \\ \ddot{\phi} &= u_4 - K_6 \dot{\phi}/I_3\end{aligned}\quad (1)$$

Burada (x, y, z) üç doğrusal eksen; (θ, ψ, ϕ) yunuslama (pitch), yuvarlanma (roll) ve yan dönme (yaw) açılarını ifade etmektedir. g yerçekimi ivmesidir. l rotorların merkeze olan uzaklıklarıdır. m aracın toplam kütlesi; I_i 'ler eksenlere göre atalet momentleri; K_i 'ler sürtünme katsayıları ve u_i 'ler aşağıda karşılıkları ifade edilmiş kontrol girişleridir.

$$\begin{aligned}u_1 &= (F_1 + F_2 + F_3 + F_4)/m \\ u_2 &= l(-F_1 - F_2 + F_3 + F_4)/I_1 \\ u_3 &= l(-F_1 + F_2 + F_3 - F_4)/I_2 \\ u_4 &= C(F_1 - F_2 + F_3 - F_4)/I_3\end{aligned}\quad (2)$$

Burada, F_i 'ler dört rotor tarafından üretilen itki kuvvetleridir ve sistemin gerçek kontrol girdileri olarak düşünülebilirler. C kuvveti momente çeviren bir ölçekleme faktörüdür. DRHA'nın altı çıkışı $(x, y, z, \theta, \psi, \phi)$ buna karşılık dört bağımsız girişi vardır. Bu nedenle DRHA'lar eksik-eyletimli sistemlerdir. Bütün durum değişkenleri aynı anda kontrol edilemez. Hedeflenen konuma ulaşmak için, (x, y, z) ve (ϕ) şeklinde keyfi olarak belirlenecek kontrol edilecek çıkışların olası bir kombinasyonu seçilebilir. Hedeflenen konuma uygun olarak keyfi olarak seçilen açılar dışındaki iki açı sabit tutularak sistemin kararlı kılınması sağlanır [Altuğ, Ostrowski ve Mahony , 2002]. Kontrolcünün başarımlı kriteri, yunuslama ve yuvarlanma açılarını sabit tutarken hedeflenen konuma ve hedeflenen yan dönme açısına ulaşabilmesidir.

Kontrolcü Tasarımı

DRHA modeli (1) tam-eyletimli ve eksik-eyletimli iki alt sisteme ayrılabilir: Tam-eyletimli alt sistem aşağıdaki gibi ifade edilebilir.

$$\begin{bmatrix} \ddot{z} \\ \ddot{\phi} \end{bmatrix} = \begin{bmatrix} u_1 \cos \theta \cos \psi - g \\ u_4 \end{bmatrix} + \begin{bmatrix} -K_3 \dot{z}/m \\ -K_6 \dot{\phi}/m \end{bmatrix}\quad (3)$$

Eksik-eyletimli alt sistem ise

$$\begin{aligned}\begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} &= \begin{bmatrix} u_1 \cos \phi & u_1 \sin \phi \\ u_1 \sin \phi & -u_1 \cos \phi \end{bmatrix} \begin{bmatrix} \sin \theta \cos \psi \\ \sin \psi \end{bmatrix} + \begin{bmatrix} -K_1 \dot{x}/m \\ -K_2 \dot{y}/m \end{bmatrix} \\ \begin{bmatrix} \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} &= \begin{bmatrix} u_2 \\ u_3 \end{bmatrix} + \begin{bmatrix} -lK_4 \dot{\theta}/I_1 \\ -lK_5 \dot{\psi}/I_2 \end{bmatrix}\end{aligned}\quad (4)$$

şeklinde ifade edilmektedir. Düşük hızlarda sürtünme çok küçük olduğu için, sürüklenme terimleri bozucu giriş olarak modellenmiştir.

A. Tam-eyletimli alt sistem için kontrolcü tasarımı:

Tam-eyletimli alt sistem (3) için durum değişkenleri z ve ϕ 'yi hedeflenen konumları z_d ve ϕ_d 'ye hareket ettirmek üzere hız sınırlamalı PID ve KKK tasarlanmıştır. z için istenilen kontrol girişi aşağıdaki gibi verilmektedir.

$$u_{1d} = \frac{k_{z1}(z_d - z) + k_{z2} \int (z_d - z) dt - k_{z3} \dot{z} + g}{\cos \theta \cos \psi} \quad (5)$$

Hız sınırlı kontrol u_1 , u_{1d} 'ye yakınsar [Xu ve Özgüner, 2006].

$$\dot{u}_1 = k_{sat} \left(\frac{k_0 \int (u_{1d} - u_1) dt + k_1 (u_{1d} - u_1)}{\varepsilon} \right) \quad (6)$$

Burada $sat(\cdot)$ doygunluk işlevini (saturation function) göstermektedir. Yaklaşma fazını ortadan kaldırmak amacıyla başlangıç koşulları $u_1(0) = u_{1d}(0)$ olarak seçilmiştir. Bu durumda aşağıdaki ifade elde edilir.

$$|\dot{u}_1| \leq k \quad (7)$$

ϕ için ϕ 'yi hedeflenen değeri ϕ_d 'ye mümkün olduğunca çabuk ulaştırmak üzere KKK tasarlanır.

$$u_4 = -c_\phi \dot{\phi} - M_\phi \text{sgn}(s_\phi) - k_\phi s_\phi \quad (8)$$

Burada, (k_{z1}, k_{z2}, k_{z3}) ve (c_ϕ, M_ϕ, k_ϕ) tasarımcı tarafından belirlenen kontrol parametreleridir ve tümü pozitifdir. ϕ için kararlı kayma yüzeyi $s_\phi = c_\phi(\phi - \phi_d) + \dot{\phi}$ şeklinde tanımlanmıştır. $\text{sgn}(\cdot)$ ile ifade edilen fonksiyon işaret fonksiyonunu sürekli kılacak şekilde aşağıdaki gibi tasarlanmıştır.

$$\text{sgn}(x) = \frac{2}{\pi} \arctan(\mu x) \quad (9)$$

Burada $\mu > 0$ pozitif bir sabittir ve yaklaşma hatası μ artırılarak azaltılabilmektedir. Bu yaklaşma hatasının azaltılması, kayan mod kontrolünün en temel sorunu olan çatırtı probleminden kaçınmak için kullanılmaktadır.

B. Eksik-eyletimli alt sistem için kontrolcü tasarımı:

[Xu ve Özgüner, 2006], (4)'de verilen eksik-eyletimli alt sistemin KKK ile kontrolüne geçmeden aşağıda verilen iç içe geçmiş eksik eyletimli sistemin KKK ile kontrol edilebilirliğini kanıtlamış ardından da (10)'un bir dizi koordinat dönüşümü ile (4) şekline getirilebileceğini göstermiştir. İspat için ilgili kaynağa bakılabileceği için biz burada sadece kontrol algoritması içerisinde kullanılacak temel formülleri ifade etmekle yetiniyoruz.

$$\begin{aligned} \dot{x}_1 &= x_2 + d_1 \\ \dot{x}_2 &= f(x_3) + d_2 \\ \dot{x}_3 &= x_4 \\ \dot{x}_4 &= u + d_3 \end{aligned} \quad (10)$$

Kontrol hedefi, tüm durum değişkenlerinin herhangi bir başlangıç koşulundan sıfıra getirilmesidir. Bu nedenle ilk olarak hatalar şu şekilde tanımlanır:

$$e_1 = x_1, e_2 = x_2, e_3 = f(x_3), e_4 = J(f)x_4 \quad (11)$$

Daha sonra kayma yüzeyi şu şekilde tanımlanır.

$$S = c_1 e_1 + c_2 e_2 + c_3 e_3 + e_4 \quad (12)$$

Burada c_i 'ler $i = (1, 2, 3)$ kontrol parametreleridir. Bozucu girişleri dikkate almaksızın kayma yüzeyi S üzerindeki eşdeğer kontrol aşağıdaki gibi ifade edilmektedir.

$$u_{eq} = -[J(f)]^{-1}\{c_1x_2 + c_2f(x_3) + c_3J(f)x_4 + \frac{d}{dt}[J(f)]x_4\} \quad (13)$$

Kontrol girişi u kayma yüzeyini $S = 0$ yapmak üzere tasarlanmalıdır.

$$u = u_{eq} + u_{sw} \quad (14)$$

Burada u_{sw} KKK'nın işarete göre değişen kontrol parçasıdır. Kontrol hedefinin ikinci parçası, daha önce bahsedildiği gibi x_3 ve x_4 'ü kararlı kılmaktır. Bunun için aşağıdaki gibi sıfıra eşitlenirler.

$$\begin{aligned} e_3 &= f(x_3) = 0 \\ e_4 &= J(f)x_4 = 0 \end{aligned} \quad (15)$$

Şimdi, yukarıda açıklanan bu KKK (4)'de verilen eksik eyletimli sisteme uygulanacaktır.

Tam-eyletimli alt sistemin kontrolcüsü, eksik-eyletimli alt sistem için geliştirilecek kontrolcüden bağımsız olduğu için, ϕ 'yi hedeflenen değerine hızlıca ulaştıracak hızlı tepki veren bir kontrolcü tasarlanabilir. Dolayısıyla, bunu varsaymak mantıklıdır, çünkü bir süre sonra ϕ hedeflenen değerine ulaştığında neredeyse zamanla değişmez hale gelir. Ayrıca, u_1 bir süre sonra sabit durum değerine $g/(\cos \theta \cos \psi)$ yaklaşır.

$$\left| u_1 - \frac{g}{\cos \theta \cos \psi} \right| \leq \eta \leq \frac{\eta}{|\cos \theta \cos \psi|} \quad (16)$$

Burada $\eta > 0$ küçük bir sabittir. Böylece alt sınır $|u_1|$ şu şekilde elde edilir.

$$|u_1| \geq \left| \frac{g - \eta}{\cos \theta \cos \psi} \right| \quad (17)$$

Eksik-eyletimli alt sistem için kontrol geliştirmek amacıyla aşağıdaki koordinat dönüşümü matrisine ihtiyaç vardır.

$$T = u_1 \begin{bmatrix} \cos \phi & \sin \phi \\ \sin \phi & -\cos \phi \end{bmatrix} = u_1 \hat{T}(\phi) \quad (18)$$

ϕ zamanla değişmez olduğu için, $\hat{T}(\phi)$ 'de zamanla değişmezdir ve basitce $\hat{T}$ olarak gösterilebilir. $u_1 \neq 0$ dışındaki her durumda transformasyon matrisinin tekil olmadığına dikkat çekmek gerekmektedir. u_1 yer çekimine dik yönde onu yenmek üzere aracın üzerine etkiyen net kuvvet toplamı olduğu için durma durumu dışında sıfır olması beklenmez. Böylece

$$x_1 = T^{-1} \begin{bmatrix} x \\ y \end{bmatrix} \quad x_2 = T^{-1} \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} \quad x_3 = \begin{bmatrix} \psi \\ \dot{\psi} \end{bmatrix} \quad x_4 = \begin{bmatrix} \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (19)$$

Yeni tanımlanan durum değişkenlerinin türevi alınırsa

$$\dot{x}_1 = T^{-1} \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} + \frac{d}{dt} \left(\frac{1}{u_1} \right) \hat{T}^{-1} \begin{bmatrix} x \\ y \end{bmatrix} = x_2 - \frac{\dot{u}_1}{u_1} x_1 \quad (20)$$

$$\dot{x}_2 = T^{-1} \begin{bmatrix} \ddot{x} \\ \ddot{y} \end{bmatrix} + \frac{d}{dt} \left(\frac{1}{u_1} \right) \hat{T}^{-1} \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} \sin \theta \cos \psi \\ \sin \psi \end{bmatrix} - \hat{T}^{-1} \begin{bmatrix} K_1/m & 0 \\ 0 & K_2/m \end{bmatrix} \hat{T} x_2 - \frac{\dot{u}_1}{u_1} x_2 \quad (21)$$

$$\dot{x}_3 = x_4 \quad (22)$$

$$\dot{x}_4 = \begin{bmatrix} u_2 \\ u_3 \end{bmatrix} + \begin{bmatrix} -lK_4/I_1 & 0 \\ 0 & -lK_5/I_2 \end{bmatrix} x_4 \quad (23)$$

Böylece aşağıdaki tanımlamalar yapıldığında (4) ile gösterilen eksik eyletimli alt sistem (10)'da verilen eksik eyletimli sisteme benzetilmiş olmaktadır.

$$\begin{aligned} f(x_3) &= \begin{bmatrix} \sin \theta \cos \psi \\ \sin \psi \end{bmatrix} \\ u &= \begin{bmatrix} u_2 \\ u_3 \end{bmatrix} \\ d_1 &= -\frac{\dot{u}_1}{u_1} x_1 \\ d_2 &= -\frac{\dot{u}_1}{u_1} x_2 - \hat{T}^{-1} \begin{bmatrix} K_1/m & 0 \\ 0 & K_2/m \end{bmatrix} \hat{T} x_2 \\ d_3 &= \begin{bmatrix} -lK_4/I_1 & 0 \\ 0 & -lK_5/I_2 \end{bmatrix} x_4 \end{aligned} \quad (24)$$

(7) ve (17) dikkate alındığında aşağıdaki sınır fonksiyonları yazılabilmektedir.

$$\begin{aligned} \|d_1\| &\leq \frac{|\dot{u}_1 \cos \theta \cos \psi|}{|g-\eta|} \|x_1\| \\ &\leq \frac{|\dot{u}_1|}{|g-\eta|} \|x_1\| \\ &\leq \frac{k}{|g-\eta|} \|x_1\| \\ &= \bar{d}_1 \|x_1\| \end{aligned} \quad (25)$$

$$\begin{aligned} \|d_2\| &\leq \frac{k}{|g-\eta|} \|x_2\| + \left\| \begin{bmatrix} K_1/m & 0 \\ 0 & K_2/m \end{bmatrix} \right\| \|x_2\| \\ &\leq \left(\frac{k}{|g-\eta|} + \max \left\{ \frac{K_1}{m}, \frac{K_2}{m} \right\} \right) \|x_2\| \\ &= \bar{d}_2 \|x_2\| \end{aligned} \quad (26)$$

$$\begin{aligned} \|d_3\| &\leq \max \left\{ \frac{lK_4}{I_1}, \frac{lK_5}{I_2} \right\} \|x_4\| \\ &= \bar{d}_3 \|x_4\| \end{aligned} \quad (27)$$

Burada,

$$\bar{d}_1 = \frac{k}{|g-\eta|}, \quad \bar{d}_2 = \max \left\{ \frac{K_1}{m}, \frac{K_2}{m} \right\}, \text{ ve } \bar{d}_3 = \max \left\{ \frac{lK_4}{I_1}, \frac{lK_5}{I_2} \right\} \quad (28)$$

Bunların hepsi pozitif sabit sayılardır. Ek olarak $f(x_3)$ fonksiyonunun jakobiyen matrisi aşağıdaki gibi elde edilir.

$$J(f) = \frac{\partial f}{\partial x_3} = \begin{bmatrix} \cos \theta \cos \psi & -\sin \theta \sin \psi \\ 0 & \cos \psi \end{bmatrix} \quad (29)$$

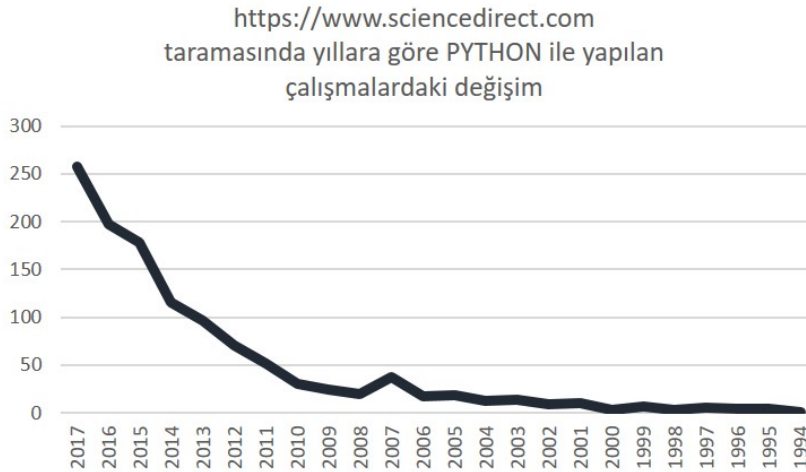
$J(f)$ pozitif tanımlı ve daima tersi alınabilir bir matristir. Bununla beraber normu daima 2'den küçüktür.

$$\|J(f)\|_{\infty} < 2 \quad (30)$$

Özetle, (14)'de verilen KKK, (6)'da verilen hız sınırlanmış PID denetleyicisiyle birlikte (8)'de ϕ açısı için verilen KKK, yunuslama ve yuvarlanma açılarını sıfır olarak tutarken istenen bir sapma açısı ile istenen bir konuma DRHA'nın sürülmesini mümkün kılmaktadır.

UYGULAMALAR

Bu bölümde, bildirinin ana fikri olan açık kaynak kodlu bir yazılımla, ileri bir kontrol uygulamasının gerçekleştirilme düşüncesinin sonuçları görülebilir. Öncelikle kısaca, PYTHON'dan bahsetmekte yarar görülmektedir. PYTHON açık kaynak kod lisanslı özgür ve ücretsiz bir yazılımdır. Nesneye yönelimli programlama, fonksiyonel ya da yapısal programlama gibi birden fazla programlama paradigmasını desteklemektedir. Windows, Linux/ Unix ve Mac-OS üzerinde çalışabilmesinin yanında Java ve .NET sanal makinelerine de port edilmiş durumdadır [Web Sayfası Erişim-, 15.03.2018]. Ek olarak tüm dünyada yeni bir eğilim olarak PYTHON'un bilimsel çalışmalarda kullanılmaya başladığını iddia etmek mümkündür. "ScienceDirect" web sitesinde yapılan aramanın sonuçları aşağıdaki grafikte görülebilir.



Şekil 2:

<https://www.sciencedirect.com> taramasında yıllara göre PYTHON ile yapılan çalışmalardaki değişim

PYTHON'a yönelmemizin başka bir nedeni de Python ile Arduino ve benzeri programlamanabilir elektronik kartlar haberleşebilmektedir. Günlük faaliyetlerde kullanılacak araçların üretim maliyetlerini düşürmek için bu yönelim bir zorunluluktur.

PYTHON ile yapılan benzetim çalışması için gerekli parametre tanımları ve DRHA için başlangıç koşulları karşılaştırma kolaylığı açısından [Xu ve Özgüner, 2006]'da verilenler ile aynı alınmıştır.

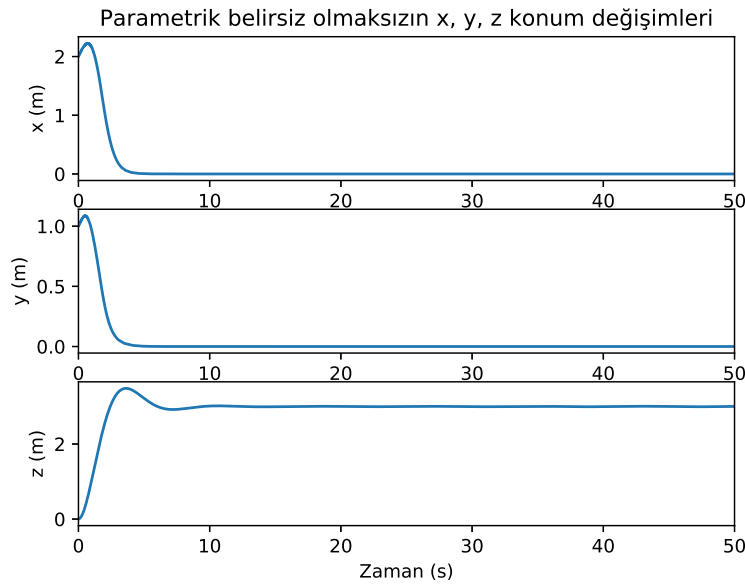
$$\begin{aligned}
I_1 &= I_2 = 1.25 \text{Ns}^2/\text{rad}, I_3 = 2.5 \text{Ns}^2/\text{rad}, \\
K_1 &= K_2 = K_3 = 0.010 \text{Ns/m}, \\
K_4 &= K_5 = K_6 = 0.012 \text{Ns/rad}, \\
m &= 2 \text{kg}, l = 0.2 \text{m}, g = 9.8 \text{m/s}^2 \\
x(0) &= 2, y(0) = 1, z(0) = 0, \\
\theta(0) &= 0, \psi(0) = 0, \phi(0) = 0
\end{aligned}$$

Aşağıdaki kontrol parametreleri ve hedeflenen değerler benzetimde, [Xu ve Özgüner , 2006]'de kullanıldığı gibi kullanılmıştır.

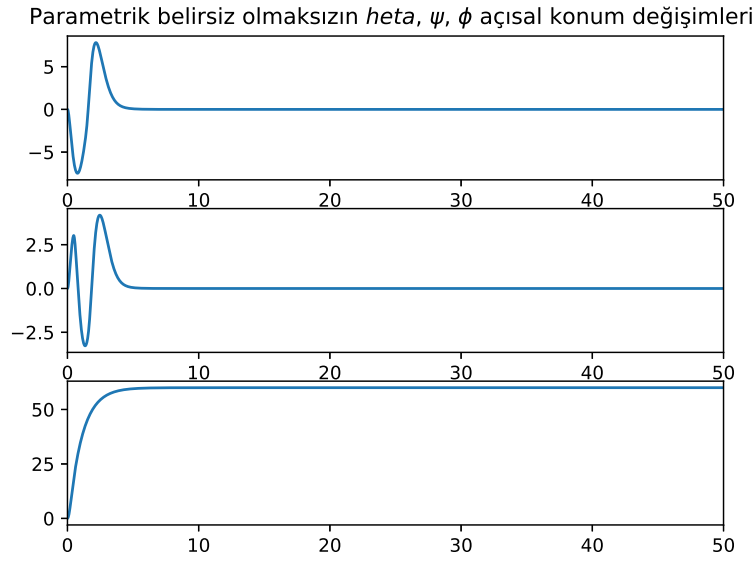
$$\begin{aligned}
k_{z1} &= 1, k_{z2} = 0.1, k_{z3} = 1, \\
k_0 &= 1, k_1 = 5, k = 5, \varepsilon = 0.1, \\
c_\phi &= 1, k_\phi = 0.1, M_\phi = 5, k_\phi = 0.1, \mu = 10, \\
c_1 &= 20, c_2 = 22, c_3 = 8, \rho = 1, \nu = 0.1, \\
\beta &= 2, \eta = 1, z_d = 3, \phi_d = \pi/3
\end{aligned}$$

Üç doğrusal eksenlerdeki (x, y, z) değişim; Dönme eksenlerinde (θ, ψ, ϕ) yunuşlama (pitch), yuvarlanma (roll) ve yan dönme (yaw) açılarındaki değişim ve kontrol girişlerindeki u_i 'ler değişimler grafik olarak gösterilmiştir. Benzetim çalışmasının sonuçları grafik olarak aşağıda verilmektedir. Ek olarak, ileride yapılabilecek olası benzer çalışmalara kaynaklık etmesi açısından aşağıdaki linkten erişilebilir.

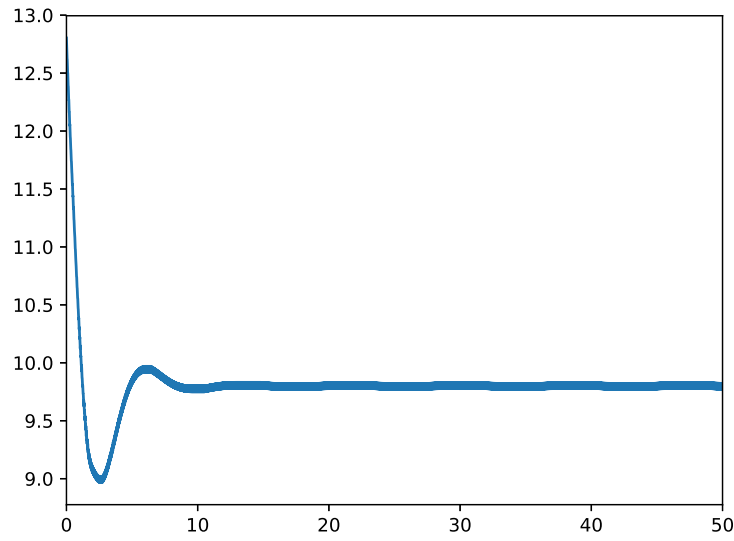
Benzetim Kodunun Linki: <https://www.dropbox.com/s/q5kcrx1cvqbf55d/quadrator.py?dl=0>



Şekil 2: DRHA'nın doğrusal eksenlerdeki konum (x, y, z) değişimi;



Şekil 3: DRHA'nın dönme eksenlerindeki (θ, ψ, ϕ) açısal konum değişimi;



Şekil 4: Hız sınırlı kontrolün u_1 değişimi;

SONUÇ

Bu bildiride, DRHA için geliştirilmiş, geçerliliği kabul edilmiş bir kontrol algoritmasını ilk defa açık kaynak kodlu bir yazılım olan PYTHON ile yeniden ürettik. Grafikler arasında bir takım parametrelerin bulunamaması nedeniyle (bu parametreler keyfi olarak belirlenmiştir) küçük farklılıklar dışında yeniden üretim başarı ile gerçekleşmiştir. Çalışmanın bundan sonraki aşaması yazılan kodun fiziksel sisteme uygulanması olarak hedeflenmektedir.

Kaynaklar

Altug, E., Ostrowski J. P. ve Mahony, R., 2002. *Control of a Quadrotor Helicopter using Visual Feedback.*, Proceedings of the 2002 IEEE International Conference on Robotics and Automation, cilt:1, s.72-77

Önkol, M. ve Efe. M.Ö., 2009. *Experimental Model Based Nonlinear Control Algorithms for a Quadrotor Unmanned Vehicle.*, 2 nd International Symposium on Unmanned Aerial Vehicles, Reno, USA, Haziran

Xu, R. ve Özgüner, Ü., 2006. *Sliding Mode Control of a Quadrotor Helicopter.*, Proceedings of the 45th IEEE Conference on Decision and Control, s.4957-4962

Web Sayfası Erişim-(13.03.2018) , <http://bolubeyi.net/quadrotorquadcopter-nedir.html>

Web Sayfası Erişim-(15.03.2018) , <https://www.python.tc/python-nedir/>