

Puma 560 Robot Arm Manipulator

B. Durmus¹, H. Temurtas², N. Yumusak³, F. Temurtas³

¹Sakarya Üniversitesi, Elektrik - Elektronik Mühendisliği Bölümü, 54187 Adapazarı,
²Dumlupınar Üniversitesi, Elektrik - Elektronik Mühendisliği Bölümü, 41470 Kütahya,
³Sakarya Üniversitesi, Bilgisayar Mühendisliği Bölümü, 54187 Adapazarı,

Abstract: In this thesis study, GPC (Generalized Predictive Control) and Newton-Raphson implemented NGPC (Neural Generalized Predictive Control) algorithms belong to the class of MBPC (Model Based Predictive Control) are investigated and each of them is applied to a six joint robotic arm as SISO (Single Input Single Output) and MIMO (Multiple Inputs Multiple Outputs) for the joint based trajectory control. Dynamics modeling of the robotic arm is made by using the Lagrange-Euler equations. The frictional effects, the state of carrying and falling load are added to dynamics model. Dynamics model obtained is transformed into robotic arm simulator by using 4th degree Runge-Kutta integration method. The trajectory planning for the joints of the robotic arm is designated according to the cubic and sinusoidal trajectories principles. The control algorithms are compared with themselves for different examples and cases. The simulation program included all of these is prepared by using Borland Delphi 6.0 programming language.

Keywords: Generalized Predictive Control, NGPC, robot, control.

Puma 560 Robot Kolu Manipülâtörü

Özet: Bu çalışmada, Genelleştirilmiş Öngörülü Kontrol (GPC – Generalized Predictive Control) ve Newton-Raphson Uyarlamalı Yapay Sinir Ağı Genelleştirilmiş Öngörülü Kontrol (NGPC – Neural Generalized Predictive Control) algoritmaları incelenmiş olup her biri Tek Giriş Tek Çıkış (SISO – Single Input Single Output) ve Çok Giriş Çok Çıkış (MIMO – Multiple Inputs Multiple Outputs) olmak üzere iki şekilde altı eklemlili bir robot koluna eklem esaslı yörünge kontrolü için uygulanmıştır. Robot kolunun dinamik olarak modellenmesinde Lagrange-Euler yöntemi kullanılmıştır. Dinamik modellemeye sürtünme etkileri, yük tasıma ve tasınan yükün tasıma esnasında düşmesi durumları da ayrıca ilave edilmiştir. Elde edilen dinamik model, 4. mertebeden Runge-Kutta bütünleştirme yöntemi kullanılarak robot kolu simülâtörüne dönüştürülmüştür. Robot kolu eklemlerinin yörünge takibi kübik ve sinüzoidal yörünge esaslarına göre belirlenmiştir. Kontrol algoritmaları farklı örnek ve durumlar için kendi aralarında kıyaslanmıştır. Gerekli bütün yazılımlar tek bir paket program halinde Borland Delphi 6.0 programlama dili kullanılarak gerçekleştirilmiştir.

Anahtar Kelimeler: Genelleştirilmiş öngörülü kontrol, NGPC, robot, kontrol.

Reference to this paper should be made as follows (bu makaleye aşağıdaki şekilde atıfta bulunulmalı):
B. Durmus et al, 'Puma 560 Robot Arm Manipulator', Elec Lett Sci Eng, vol. 2(2), (2006), 21-30

1 Giriş

Robot kolunun dinamik kontrolü, robot kolu eklemlerinin istenilen pozisyon ve hız referansları doğrultusunda hareketlerini sağlayacak giriş bilgilerini (tork / voltaj) üreterek eklemlere vermek şeklindedir. Ancak, robot kolunun dinamik davranışını veren denklemlerin ikinci dereceden doğrusal olmayan diferansiyel denklemler olmaları ve aralarında yüksek oranda etkileşimler bulunması robot kolunun kontrolünü zorlaştırmaktadır. Bu yüzden klasik kontrol sistemlerini

* Corresponding author; Tel.: +(90) 533 549 41 70, email:bdurmus@sakarya.edu.tr

kullanan endüstriyel robot kolları belirli bir hız limitinin üzerine kolayca çıkamamakta, sonuç olarak üretim verimliliği sınırlanmaktadır. Ayrıca, robot kolu kontrolörlerinden beklenen ve her geçen gün artan performans isteklerinden dolayı daha gelişmiş kontrol tekniklerine gereksinim duyulmaktadır.

Endüstride kullanılan robot kolu kontrolörleri yukarıda da belirtildiği gibi genellikle basit bir yapıya sahiptirler. Eklemler arasındaki etkileşimlerin önemsiz sayılabilecek düzeyde kalabilmesi için eklem hızları düşük tutulmakta ve her bir eklem ayrı bir PID tipi kontrolör ile denetlenmektedir. Robot kolunun çalışma hızını artırabilmek için eklem hızları arttırıldığında etkileşimler ve robot kolu modelindeki belirsizlikler nedeniyle istenilen yörünge ile gerçekleşen yörünge arasında önemli hatalar ortaya çıkmaktadır. Bu nedenle klasik kontrol sistemlerini kullanan robot kolları, yeterli hassasiyeti yakalamak için düşük hızlarda çalıştırılmaktadır.

Bu çalışmada, Clarke ve arkadaşları [1,2,3,4,5] tarafından 1987 yılında tanıtılan GPC ile Soloway ve Haley [6,7,8,9] tarafından 1997 yılında tanıtılan Newton-Raphson uyarlamalı Yapay Sinir Ağı Genelleştirilmiş Öngörülü Kontrol (NGPC – Neural Generalized Predictive Control) algoritmaları incelenmiş olup her birisi SISO ve MIMO olmak üzere iki şekilde tasarlanarak altı eklemlilik bir robot koluna uygulanmıştır. Robot kolu hareketlerinin dinamik olarak modellenmesinde Lagrange-Euler yöntemi kullanılmıştır. Dinamik modellemeye sürtünme, yük taşıma ve taşınan yükün taşıma esnasında düşmesi durumları da ayrıca ilave edilmiştir. Dinamik modelin robot kolu simülatorüne dönüştürülmesinde 4. mertebeden Runge-Kutta bütünleştirme yöntemi kullanılmıştır. Robot kolunun her bir eklemine ait takip etmesi istenilen konum referans ve hız referans yörüngeleri kübik ve sinüzoidal yörünge esaslarına göre belirlenmiştir. Gerekli bütün yazılımlar tek bir paket program halinde Borland Delphi 6.0 programlama dili kullanılarak gerçekleştirilmiştir.

2 Dinamik Model

Robot kolunun dinamik modellenmesinde daha önce yapılmış çalışmadan yararlanılmıştır. Çalışmada PUMA 560 robot kolunun dinamik model parametreleri hesaplanmıştır [10]. Manipülatör tasarımı dinamik model için bu parametre değerleri kullanılmıştır. Dinamik model için gerekli parametreler şu şekildedir: Modifiye edilmiş Denavit-Hartenberg parametreleri [11], motor ve sürücü parametreleri, maximum motor tork değerleri.

| i | α_{i-1} (degrees) | θ_i | a_{i-1} (meters) | d_i (meters) |
|-----|-----------------------------|------------|-----------------------|-------------------|
| 1 | 0 | q_1 | 0 | 0 |
| 2 | -90 | q_2 | 0 | .2435 |
| 3 | 0 | q_3 | .4318 | -.0934 |
| 4 | 90 | q_4 | -.0203 | .4331 |
| 5 | -90 | q_5 | 0 | 0 |
| 6 | 90 | q_6 | 0 | 0 |

Sekil 1 Denavit-Hartenberg Parametreleri

PUMA 560 robot koluna ait etiket özellikleri [12]:

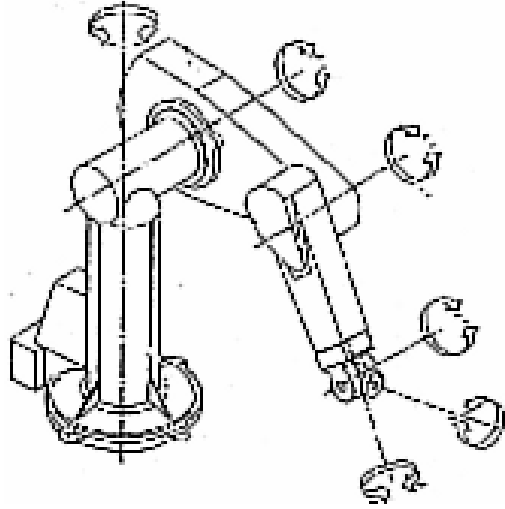
Eklem sayısı: 6

Yüklü kapasite: 4 kg

Servis uzunluğu: 0.95m radius

Maximum Hız: 1.0 m/sn

Sürücüler: DC motor



Sekil 2 PUMA 560 Robot Kolu Eklem Ekesenleri

Dinamik modelin türetilmesi:

Bu analizler için asagidaki dinamik model kullanılacaktır:

$$A(q)\ddot{q} + B(q)[\dot{q}'\dot{q}] + C(q)[\dot{q}'^2] + g(q) = \tau \quad (1)$$

Burada $A(q)$ $n \times n$ lik kinetik enerji matrisi;

$B(q)$ $n \times n(n-1)/2$ lik Coriolis torqlari matrisi

$C(q)$ $n \times n$ lik merkezkaç torku matrisi

$g(q)$ n vektör yerçekimi torku

\ddot{q} ivme(hizlandirma) n vektörü

τ genelleştirilmiş eklem kuvveti

$[\dot{q}\dot{q}]$ ve $[\dot{q}'^2]$ sembolleri hız ürünü $n(n-1)/2$ vektörü ve kare hız n vektörü için işaretlerdir. $[\dot{q}\dot{q}]$ ve $[\dot{q}'^2]$ şöyle verilir:

$$[\dot{q}'\dot{q}'] = \left[\dot{q}_1 \dot{q}_2, \dot{q}_1 \dot{q}_3, \dots, \dot{q}_1 \dot{q}_n, \dot{q}_2 \dot{q}_3, \dot{q}_2 \dot{q}_4, \dots, \dot{q}_{n-2} \dot{q}_n, \dot{q}_{n-1} \dot{q}_n \right]^T, [\dot{q}'^2] = \left[\dot{q}_1'^2, \dot{q}_2'^2, \dots, \dot{q}_n'^2 \right]^T \quad (2)$$

Dinamik modeli türetmek için su dört adım izlenir:

1. Lagrange veya Gibbs-Alambert formülasyonunun özetlenmesi ile icra edilen yerçekimi vektör elemanlari ve kinetik enerji matrisinin sembolik generasyonu.
2. Çogunlukla yaygin degisken ifadeler olan birlesik atalet sabitleri ile kinetik enerji matrisi elemanlarinin basitlestirilmesi.
3. Kinetik enerji matrisi elemanlarinin kısmi türetimi terimlerinde Coriolis ve merkezkaç matrisi elemanlarinin ifadelenmesi ve bu kısmi türetimleri tutan dört bağlantı ile bu ifadelerin indirgenmesi.
4. İhtiyaç duyulan kısmi türetimlerin olusturulmasi, türetilmiş terimlerdeki merkezkaç ve Coriolis matris elemanlarinin genişletilmesi ve 2 deki gibi atalet sabitlerinin bilesimi ile basitlestirme [13].

2.1 Sürtünme Etkisi

Sürtünme etkilerinin modellenmesi ve dengelenmesi ile robotların performansı artırılabilir. Dinamik modellemede robot kolunun sürtünmesi için statik, kinetik(Coulomb) ve akışkan sürtünmesini içeren bir sürtünme modellemesi yapılmıştır. Bu model aşağıda denklem ile gösterilmektedir.

$$\tau_{\text{friction}} = f_s \frac{\text{sgn}(\dot{q})}{1 + \left(\frac{\dot{q}}{\chi_s}\right)^2} + f_k \tanh(q') + k_{vn} (\dot{q}) \quad (3)$$

Burada f_s statik sürtünme, χ_s Stribeck etkisinden oluşan statik sürtünme sabiti, f_k kinetik sürtünme ve k_{vn} akışkan sürtünmesidir. Puma 560 robot koluna ait sürtünme parametreleri Tablo 2 de verilmektedir [14].

Tablo 2 Puma 560 Sürtünme Parametreleri

| Eklemler | f_s | f_k | k_{vn} | χ_s |
|----------|-------|-------|----------|----------|
| 1 | 5 | 2 | 1 | 0.1 |
| 2 | 5 | 2 | 1 | 0.1 |
| 3 | 2.5 | 1 | 1 | 0.1 |
| 4 | 0.3 | 0.1 | 0.05 | 0.1 |
| 5 | 0.2 | 0.1 | 0.05 | 0.1 |
| 6 | 0.2 | 0.1 | 0.05 | 0.1 |

Sürtünme etkisinden oluşan tork değeri eklemlere ait genel tork ifadesine eklenir. Bu durumda genel denklem aşağıdaki gibi elde edilir.

$$\mathbf{t}(t) = D(\mathbf{q})\ddot{\mathbf{q}} + H(\mathbf{q}, \dot{\mathbf{q}}) + G(\mathbf{q}) + \mathbf{t}_s(t) + \mathbf{t}_y(t) \quad (4)$$

2.2 Yük Etkisi

Robot kolu dinamik modellemesinde yük taşıma ve yükün taşıma esnasında düşmesi durumlarının incelenmesi için modele yük etkisi ilave edilmiştir. Yük durumu için 2004 de tamamlanan Hasan Temurtas'a ait doktora tezinden yararlanılmıştır [15].

Robot kolunun yük taşıma esnasında kontrolünü gözlemlemek için manipülatörde yük durumu ile ilgili kontrol paneli bulunmaktadır. Yük etkisi için yüklü ve yüksüz durumları kullanıcı tercihine bırakılmıştır. Kullanıcı robot kolunun taşıyacağı yük ağırlığını arayüzden belirleyebilmektedir. Öte yandan kontrol esnasında yükün düşmesi olayını incelemek için yükün düştüğü adım sayısı belirlenmelidir. Kullanıcı yükün düştüğü adım sayısını konsoldan girerek yük düşme esnasındaki kontrolün durumunu inceleyebilmektedir.

Tasınan yükten dolayı oluşan yük tork vektörü $\mathbf{t}_y(t)$ 'yi bulmak için önce uç eleman vektörü $\mathbf{r}(t)$ elde edilir. Uç eleman vektörü $\mathbf{r}(t)$ ile eklem açılı vektörü $\mathbf{q}(t)$ arasındaki ilişki aşağıda verilmektedir.

$$\mathbf{r}(t) = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \Lambda(\mathbf{q}) \quad (5)$$

$\Lambda(\mathbf{q})$, 6×1 boyutunda robot kolu dinamiklerini içeren ve düz kinematik simgeleyen bir vektördür. Eklem uzayı $\{\mathbf{q}\}$ ile Kartezyen uzay $\{r\}$ arasındaki dönüşümü sağlayan bağıntılar

$$\begin{aligned} \dot{\mathbf{r}}(t) &= \mathbf{J}(\mathbf{q})\dot{\mathbf{q}} \\ \ddot{\mathbf{r}}(t) &= \mathbf{J}(\mathbf{q})\ddot{\mathbf{q}} + \dot{\mathbf{J}}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} \end{aligned}$$

şeklinde dir. Burada, $\mathbf{J}(\mathbf{q}) = \partial\Lambda(\mathbf{q})/\partial\mathbf{q}$, $3 \times n$ boyutlarında robot kolu Jacobian matrisidir. $\dot{\mathbf{J}}(\mathbf{q}, \dot{\mathbf{q}})$ ise Jacobian matrisinin birinci türevidir.

Robot kolu uç elemanının taşıdığı noktasal yükün kütlesi m olsun. Bu durumda yükün $\ddot{\mathbf{r}}(t)$ ivmesi ile hareket edebilmesi için uç eleman yüküne 6×1 'lik $\mathbf{f}(t)$,

$$\mathbf{f}(t) = m \cdot (\ddot{\mathbf{r}}(t) - \bar{\mathbf{g}}) \quad (6)$$

kuvvetini uygulayacaktır. $\bar{\mathbf{g}}$, 6×1 'lik yerçekimi ivme vektörü, temel koordinat sistemi eksenleri cinsinden ifade edilmelidir. Temel koordinat sisteminde \bar{z}_0 ekseninin yönü yerçekimi ivmesinin zıt yönünde alınırsa bu durumda $\bar{\mathbf{g}}$ vektörü,

$$\bar{\mathbf{g}} = \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} \quad (7)$$

şekilde gösterilir.

Jacobian matrisi $\mathbf{J}(\mathbf{q})$ 'nin transpozesi uç eleman kuvvet vektörü $\mathbf{f}(t)$ ile çarpılırsa m kütlelerinden dolayı oluşan ilave eklem tork vektörü

$$\mathbf{t}_y(t) = \mathbf{J}^T(\mathbf{q})\mathbf{f}(t) \quad (8)$$

elde edilir. $\mathbf{f}(t)$ yerine konulursa

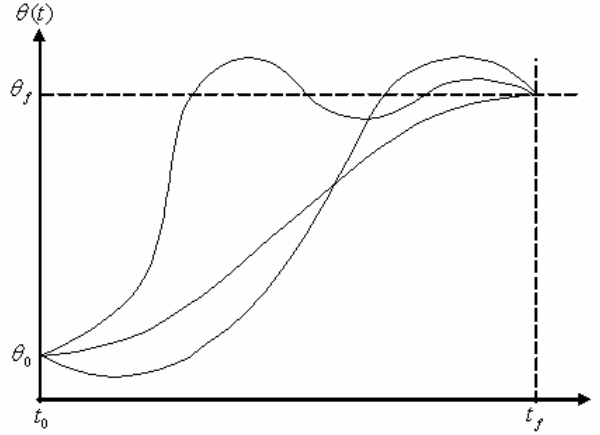
$$\mathbf{t}_y(t) = m\mathbf{J}^T(\mathbf{q})[\mathbf{J}(\mathbf{q})\ddot{\mathbf{q}} + \dot{\mathbf{J}}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} - \bar{\mathbf{g}}] = m\mathbf{L}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \quad (9)$$

elde edilir. Burada $\mathbf{L}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$, birim yük başına düşen ilave tork vektörüdür.

2.3 Yörünge planlamada fonksiyon kullanımı

Yol tanımlamada detayları içine alan bir yöntem; noktalar yoluyla istenen hareket sırasını göstermek yani başlangıç noktasından bitiş noktasına kadar yörüngeyi ara noktalarla temsil etmektir. Hareketin sarsıntısız ve düzenli olabilmesi için bu ara noktalar uygun bir düzgünleştirme fonksiyonunu takip edecek şekilde seçilmelidir. Noktalar denmesine rağmen bunlar pozisyon ve yönelmeyi gösteren koordinatlardır.

Sabit bir zaman aralığında robot kolunu bir başlangıç konumdan bir hedef konuma getirme problemini göz önüne alalım. Sistem kinematigi kullanılarak başlangıç ve hedef konuma karşılık gelen eklem açıları hesaplanabilir. Burada istenen, sabit bir zaman aralığında eklem başlangıç konumu ile hedef konumu arasında her bir eklem için zamana bağlı bir $\mathbf{q}(t)$ fonksiyonu elde etmektir. Bu amaçla kullanılacak bir çok düzgün fonksiyon mevcuttur (Sekil 3).



Sekil 3 Her bir eklem için uygun yörünge planlari

Robot kolu düzgün bir hareket yaparken eklemlere ait $\mathbf{q}(t)$ fonksiyonu üzerinde en az dört sınır değerinin olması gerekmektedir. İki sınır değeri, başlangıç ve bitiş pozisyonlarının belirlenmesinden gelmektedir.

$$\mathbf{q}(0) = \mathbf{q}_0, \quad \mathbf{q}(t_f) = \mathbf{q}_f \quad (10)$$

İlave iki sınır değeri ise fonksiyonun açısal hız bakımından sürekli olmasından gelir. Başlangıç ve bitiş pozisyonundaki açısal hızlar sıfırdır.

$$\dot{\mathbf{q}}(0) = 0, \quad \dot{\mathbf{q}}(t_f) = 0 \quad (11)$$

Robot koluna birbirini takip edecek şekilde pes pes birden fazla hareket yaptırmak istenirse hareketlerin toplamı tek bir hareketmiş gibi düşünülebilir. Bu durumda hareketlerin birleşme noktalarına ara noktalar denir. Ara noktalarda açısal hızlar 0 olmak zorunda değildir. Fakat hareketin sarsintisiz olması için bu ara noktalarda hız ve ivme sürekliliği aranır.

2.3.1 Kübik yörünge

Kübik yörünge üçüncü dereceden bir fonksiyon olup denklemini

$$\mathbf{q}(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 \quad (12)$$

biçimindedir. Bu durumda yörünge boyunca açısal hız ve ivme :

$$\dot{\mathbf{q}}(t) = a_1 + 2a_2 t + 3a_3 t^2, \quad \ddot{\mathbf{q}}(t) = 2a_2 + 6a_3 t \quad (13)$$

olarak elde edilir. Kübik yörünge yukarıdaki dört şartı rahatlıkla sağlar. Yörünge denklemleri bu dört şart ile birleştirilip çözümlerse katsayılar

$$a_0 = \mathbf{q}_0, \quad a_1 = 0, \quad a_2 = \frac{3}{t_f^2} (\mathbf{q}_f - \mathbf{q}_0), \quad a_3 = \frac{-2}{t_f^3} (\mathbf{q}_f - \mathbf{q}_0) \quad (14)$$

olarak elde edilir. Bu katsayılar yörünge denklemlerinde yerlerine konularsa

$$\mathbf{q}(t) = \mathbf{q}_0 + \frac{3}{t_f^2} (\mathbf{q}_f - \mathbf{q}_0) t^2 - \frac{2}{t_f^3} (\mathbf{q}_f - \mathbf{q}_0) t^3 \quad (15)$$

$$\dot{\mathbf{q}}(t) = \frac{6}{t_f^2} (\mathbf{q}_f - \mathbf{q}_0) t - \frac{6}{t_f^3} (\mathbf{q}_f - \mathbf{q}_0) t^2 \quad (16)$$

denklemleri, sonra bu denklemler n adimli kesikli hale dönüştürülürse

$$\mathbf{q}[i] = \mathbf{q}_0 + (\mathbf{q}_f - \mathbf{q}_0) \left(\frac{i}{n} \right)^2 \left(3 - \frac{2i}{n} \right), \quad i = 0 \dots n \quad (17)$$

$$\dot{\mathbf{q}}[i] = \frac{6(\mathbf{q}_f - \mathbf{q}_0)}{t_f} \frac{i}{n} \left(1 - \frac{i}{n} \right), \quad i = 0 \dots n \quad (18)$$

denklemleri elde edilir.

2.3.2 Sinüzoidal yörünge

Sinüzoidal yörünge denklemi

$$\mathbf{q}(t) = a + b \cos(\omega t) \quad (19)$$

biçimindedir. Bu durumda yörünge boyunca açısal hız ve ivme :

$$\dot{\mathbf{q}}(t) = -b \omega \sin(\omega t), \quad \ddot{\mathbf{q}}(t) = -b \omega^2 \cos(\omega t) \quad (20)$$

olarak elde edilir. Sinüzoidal yörünge yukarıdaki dört şartı rahatlıkla sağlar. Yörünge denklemleri bu dört şart ile birleştirilip çözülürse katsayılar

$$\omega = \frac{p}{t_f}, \quad a = \frac{\mathbf{q}_f + \mathbf{q}_0}{2}, \quad b = -\left(\frac{\mathbf{q}_f - \mathbf{q}_0}{2} \right) \quad (21)$$

olarak elde edilir. Bu katsayılar yörünge denklemlerinde yerlerine konulursa

$$\mathbf{q}(t) = \left(\frac{\mathbf{q}_f + \mathbf{q}_0}{2} \right) - \left(\frac{\mathbf{q}_f - \mathbf{q}_0}{2} \right) \cos\left(\frac{p t}{t_f} \right) \quad (22)$$

$$\dot{\mathbf{q}}(t) = \left(\frac{\mathbf{q}_f - \mathbf{q}_0}{2} \right) \left(\frac{p}{t_f} \right) \sin\left(\frac{p t}{t_f} \right) \quad (23)$$

denklemleri, sonra bu denklemler n adimli kesikli hale dönüştürülürse

$$\mathbf{q}[i] = \left(\frac{\mathbf{q}_f + \mathbf{q}_0}{2} \right) - \left(\frac{\mathbf{q}_f - \mathbf{q}_0}{2} \right) \cos\left(\frac{p i}{n} \right), \quad i = 0 \dots n \quad (24)$$

$$\dot{\mathbf{q}}[i] = \left(\frac{\mathbf{q}_f - \mathbf{q}_0}{2} \right) \left(\frac{p}{t_f} \right) \sin\left(\frac{p i}{n} \right), \quad i = 0 \dots n \quad (25)$$

denklemleri elde edilir.

2.3.3 Doğrusal yörünge

Doğrusal yörünge denklemi

$$\mathbf{q}(t) = a + b t \quad (26)$$

biçimindedir. Bu durumda yörünge boyunca açısal hız ve ivme :

$$\dot{\mathbf{q}}(t) = b, \quad \ddot{\mathbf{q}}(t) = 0 \quad (27)$$

olarak elde edilir. Doğrusal yörüngede açısal hız sabittir. Bu yüzden sadece başlangıç ve bitis açıları denklemde yerlerine konularak çözümlerse katsayılar

$$a = \mathbf{q}_0, \quad b = \frac{\mathbf{q}_f - \mathbf{q}_0}{t_f} \quad (28)$$

olarak elde edilir. Bu katsayılar yörünge denklemlerinde yerlerine konularsa

$$\mathbf{q}(t) = \mathbf{q}_0 + \left(\frac{\mathbf{q}_f - \mathbf{q}_0}{t_f}\right)t, \quad \dot{\mathbf{q}}(t) = \frac{\mathbf{q}_f - \mathbf{q}_0}{t_f} \quad (29)$$

denklemleri, sonra bu denklemler n adimli kesikli hale dönüştürülürse

$$\mathbf{q}[i] = \mathbf{q}_0 + (\mathbf{q}_f - \mathbf{q}_0) \left(\frac{i}{n}\right), \quad i = 0 \dots n \quad (30)$$

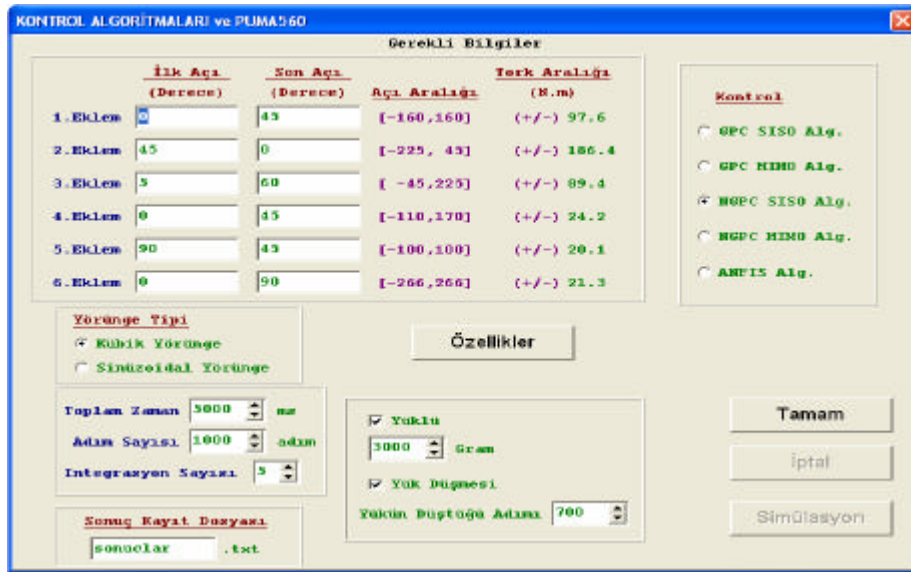
$$\dot{\mathbf{q}}[i] = \frac{\mathbf{q}_f - \mathbf{q}_0}{t_f}, \quad i = 0 \dots n \quad (31)$$

denklemleri elde edilir.

Bu çalışmada, robot kolunun her bir eklemine ait takip etmesi istenen konum referans ve hız referans yörüngeleri bu bölümde anlatılan kübik ve sinüzoidal yörünge esaslarına göre belirlenmiştir. İki yörünge planlamasının da robot kolu kontrolü için uygun olduğu gözlenmiştir.

3 Manipülâtör

Manipülâtör arayüzü Delphi 6.0 ile yapılmıştır. Arayüzde eklemlerin başlangıç ve bitis açılarının istenildiği konsol mevcuttur. Kullanıcı uygulamayı başlatmadan önce 6 ekleme ait bu açı değerlerini girmelidir. Arayüz konsolunun sağ üst köşesinde robot koluna uygulanacak kontrol algoritmaları yer almaktadır. Kullanıcı bu algoritmalarından tercihini yapabilmektedir. Şekil 4'de arayüz gösterilmiştir.

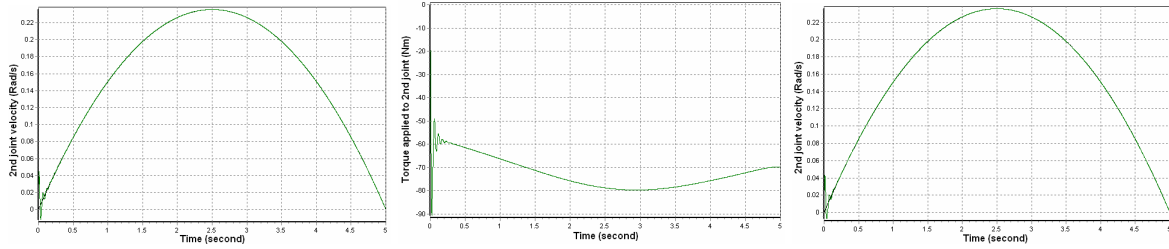


Şekil 4 Puma 560 Manipülâtör Arayüzü

Kontrolün toplam süresi ve adım sayısı kullanıcı tercihine bırakılmıştır. Kullanıcı kontrolün adım sayısını belirleyebilmektedir. Kullanıcı eklemlere ait başlangıç ve bitis açı değerleri,

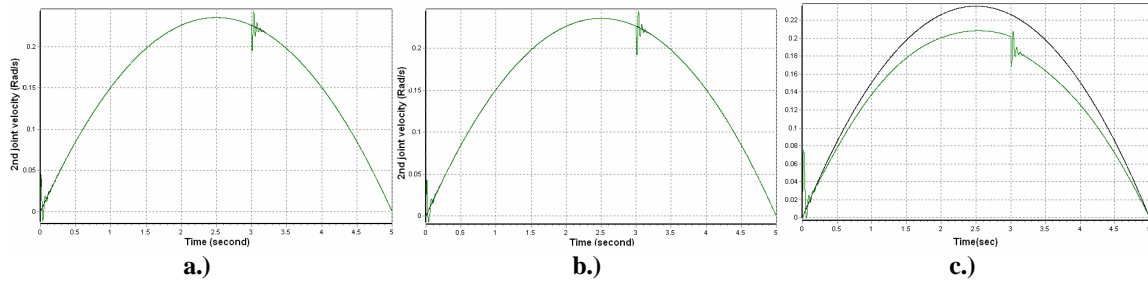
uygulanacak kontrol algoritması, adım sayısı ve yük durumu girdikten sonra simülasyonu başlatır. Simülasyon sonuçlarında kontrol grafik sonuçları yer almaktadır. Burada eklemlere uygulanan tork değerleri, eklemlerin açısal yörüngesi, eklemlerin açısal hızları, açı farkları, hız farkları grafikleri çizilmektedir. Kullanıcı herbir eklem ait grafikleri görebilir.

Manipülator eklemlere ait açı değerleri girildiğinde herbir eklem için açı, açısal yol ve eklem uygulanacak tork değerini hesaplayarak dinamik modele uygular. Uygulanan bu tork değerleri sonucunda eklemlerin takip ettiği yörüngelerin grafikleri manipülator tarafından çizilerek kullanıcıya sunulur. Şekil 5 de örnek grafikler verilmiştir.



Şekil 5 Eklemlere ait örnek yörünge ve tork grafikleri

Kontrolde GPC-SISO, GPC-MIMO, NGPC-SISO ve NGPC-MIMO olmak üzere dört adet kontrol algoritması kullanılmaktadır. Bu algoritmalar için açı, açısal yol ve son nokta koordinat hataları sonuçları kullanılarak kendi aralarında karşılaştırma yapılmıştır.



Şekil 6 a.) GPC-SISO b.) NGPC-SISO c.) GPC-MIMO ya ait 2. eklem için yörünge grafikleri

4 Sonuçlar

Robot kol kontrolü için önemli olan robot kolu uç elemanının her hangi bir başlangıç pozisyonundan istenen bir bitiş pozisyonuna sarsintisiz bir şekilde en az hata ile gidebilmesidir. Her bir eklemin açısal hız hatalarının kareleri toplamı ne kadar az olursa robot kolundaki sarsinti o derece az olur. Yine, eklemlerin açısal hataları ne kadar az ise robot kolunun uç elemanının hedef noktaya uzaklık hatası o derece azdır. Kullanılan algoritmalar içerisinde NGPC SISO algoritması bu şartları diğer üç algoritmaya kıyasla çok daha fazla sağlamaktadır. Bunu sırasıyla GPC SISO ve GPC MIMO algoritmaları takip etmektedir.

Eklemler, açısal hız yörüngelerini büyük bir doğrulukla takip etmektedirler. Açısal hız sapmaları en çok kontrolün başlangıcında ve yük düşmesi durumunda olmaktadır. Simülasyon sonuçları algoritmaların duruma kendilerini çok çabuk adapte ettiklerini göstermektedir. Bunların haricindeki sapmalar ihmal edilecek kadar küçüktür.

References (Referanslar)

1. Clarke, D. W., Mohtadi, C., Tuffs, P. C., Generalized Predictive Control - Part 1: The Basic Algorithm, *Automatica*, vol. 23, pp. 137-148, 1987.
2. Clarke, D. W., Mohtadi, C., Tuffs, P. C., Generalized Predictive Control - Part 2: The Basic Algorithm, *Automatica*, vol. 23, pp. 149-163, 1987.
3. Clarke, D. W., Mohtadi, C., Tuffs, P. C., Generalized Predictive Control: A New Robust Self-Tuning Algorithm in Landau, I. D., and L. Dugard(Eds). *Comande Adaptive - Aspects Pratiques et Theoriques*, pp. 209-228, Masson, Paris, 1987.
4. Clarke, D. W., Zhang, L., Long-Range Predictive Control Using Weighting-Sequence Models, *Proc. IEE*, 134, Pt. D(3): 187-195, 1987.
5. Clarke, D. W., Generalized Predictive Control: A Robust Predictive Control: A Robust Self-Tuning Algorithm, *ACC* 1987.
6. Soloway, D., Neural Generalized Predictive Control for Real-Time Control, Masters Thesis, Old Dominion University, 1996.
7. Soloway, D., Haley, P. J., Neural Generalized Predictive Control: A Newton - Raphson Implementation, *Proceedings of the IEEE CCA/ISIC/CACSD*, IEEE Paper No. ISIAAC-TA5.2, Sept. 15-18, 1996.
8. Soloway, D., Haley, P. J., Neural Generalized Predictive Control: A Newton-Raphson Implementation, NASA Technical Memorandum 110244, Langley Research Center, Hampton, Virginia, Feb. 1997.
9. Haley, P. J., Soloway, D., Gold, B., Real-Time Adaptive Control Using Neural Generalized Predictive Control, 1999 American Control Conference, San Diego, California, June 1999.
10. B. Armstrong, O. Khatib and J.Burdick, "The explicit dynamic model and inertial parameters of the PUMA 560 arm", *IEEE Conf. on Robotics and Aotomation*, pp. 510-518, April 1986.
11. Denavit, J., Hartenberg, R. S., A Kinematics Notation for Lower-Pair Mechanisms, *J. Applied Mechanics*, Vol. 22, pp. 215-221, 1955.
12. <http://www.rpautomation.com/Robotspecs/560spec.htm>
13. Bezine, H., Derbel, N., Alimi, A. M., Fuzzy control of robot manipulators : some issues on design and rule base size reduction, *Engineering Applications of Artificial Intelligence*, Volume 15, Issue 5, pp. 401-416, Sept. 2002.
14. "The Operational Space Formulation Implementation to Aircraft Canopy Polishing Using a Mobile Manipulator", Rodrigo J., Marcelo H. Ang, Denny Oetomo, Oussama Khatib, Tao Ming Lim, Ser Yong Lim, *International Conference on Robotics&Automation*, May 2002, Whashington D C., p.p 400-4005.
15. " Üç Eklemlı Bir Robot Kolunun Nöro Genelleştirilmiş Öngörölü Kontrol İle Eklem Esaslı Yörünge Kontrolü", H. Temurtas, *Doktora Tezi*, 2004, Sakarya.