



# DC Motor Sürücüsü İçin Model Öngörülü Denetleyici Tasarımı

**Ammar BAWDAKA\***

Dicle Üniversitesi, Elektrik Elektronik Mühendisliği Bölümü, Diyarbakır  
[15805005@dicle.edu.tr](mailto:15805005@dicle.edu.tr) ORCID: 0000-0001-7297-7441, Tel: 0 505 155 02 14

**İbrahim KAYA**

Dicle Üniversitesi, Elektrik Elektronik Mühendisliği Bölümü, Diyarbakır  
[ikaya@dicle.edu.tr](mailto:ikaya@dicle.edu.tr) ORCID: 0000-0002-8393-1358, Tel: (412) 241 10 00 (3507)

Geliş: 12.07.2019, Revizyon: 20.08.2019, Kabul Tarihi: 20.09.2019

## Öz

*Model Öngörülü Kontrol çok değişkenli kısıtlı sistemlerin denetiminde kullanılan çok güçlü bir kontrol stratejisidir. Ayrıca, bu denetim metodu klasik kontrolden çok farklı olup yüksek tasarım becerileri gerektirir. Bu makale, Model Öngörülü Kontrol stratejilerinden birinin bir DC motorun matematiksel modeline dayalı uygulanmasının sonuçlarını sunmaktadır. Bu çalışmanın amacı, bu stratejinin gelişmiş endüstriyel sistemlerde nasıl çalıştığını ve bu denetleyicilerin üstlenebileceği aktif rolü göstermektir. Bu amaçla, Model Öngörülü Kontrol ile klasik PI-P denetleyici arasında benzetim yoluyla karşılaştırmalar yapılarak Model Öngörülü Kontrolün yüksek kapalı çevrim performansı gösterilmiştir.*

**Anahtar Kelimeler:** Model Öngörülü Kontrol (MPC), DC Motor, PI-P Denetleyici, Matlab, Simulink, MPC Araç Kutusu.

\* Yazışmaların yapılacağı yazar:

## Giriş

Model Öngörülü Kontrol (Model Predictive Control, MPC), diğer gelişmiş kontrol tekniklerinde olduğu gibi, bir insanın düşünme yöntemine çok yakın davranan denetleyiciler üretirek sistemdeki hataları hızlı ve etkin bir şekilde yok eder. Başka bir deyişle, MPC denetleyiciler sistemdeki hataları daha oluşmadan önce öngörmeye çalıştıklarından bu ismi almışlardır.

MPC denetleyici, FPGA ve ARM veya çok hızlı işlem yapabilen mikroişlemcilerde gerçek zamanlı uygulamalar için kullanılabilir (Gopi vd., 2013). Ancak, yakın zamanda yapılan bir çalışmada, Arduino gibi daha yavaş olan çiplerde de MPC denetleyicilerin çevrimdışı çalışabileceği bildirilmiştir (Prasetyo vd., 2018). MPC denetleyiciler, endüstriyel uygulamalarda yaygın araştırılan çalışma alanlarından olmuştur (Qin ve Badgwell, 2003). Bunun nedeni, geleneksel PID denetleyicilerin aksine, MPC algoritması ile tasarlanmış denetleyicilerin kontrol algoritmasının içine sistem kısıtlarını da katılabilmesi, -hızlı cevap verebilmeleri ve performanslarındaki dayanımdır. Bu çalışmada, MPC denetleyici ile serbest uyarımlı DC motorun matematiksel modeli kullanılarak söz konusu motorun hız kontrolüne dair sonuçlar verilmiştir.

Literatürde, farklı MPC denetleyiciler hakkında yapılmış çok sayıda çalışma mevcuttur. Örneğin, Gonzarolli vd. (2010) doğrusal-olmayan Model Öngörülü Kontrol ile bir DC motorun hız denetimini çalışmıştır. Bangia vd. (2015) doğrusal hale getirilmiş bir DC motorun çalışma zamanı ile alakalı problemleri iyileştirmeye yönelik analiz ve tasarımı MPC stratejisi ile çalışmıştır. Alkurawy ve Khamas (2018) MPC denetleyici yöntemlerinden olan Dinamik Matris Kontrol (DMC) ile klasik PI denetleyici kullanılarak denetlenen bir DC motorun performans karşılaştırmasını vermiştir.

Syaichu-Rohman ve Sirius (2011) bir DC

motorun hız kontrolünü MPC denetleyicinin Programlanabilir Mantık Denetleyici (PLC) ile uygulamasını vermiştir. Klasik PID denetleyici, Lineer Kuadratik Regülatör (LQR) ve MPC ile yapılan bir DC motorun hız kontrolüne ait karşılaştırmalar Dani vd. (2017) tarafından sağlanmıştır. Durmaz vd. (2018) aktif bir süspansiyon sisteminin LQR ve MPC ile kontrol edilmesi ve performanslarının karşılaştırılmasını çalışmışlardır. Çalışmada, MPC denetleyici için Matlab MPC araç kutusu kullanmışlardır. Šlapák vd. (2016) DC-DC çevirici ile beslenen fırçalı sürekli mıknatıslı DC motorun sonlu kontrol set MPC ile denetlenmesine ait tasarım prosedürünü açıklamışlardır. Çalışmada önerilen MPC denetleyicinin performansı PI-PWM kaskad hız kontrol ile karşılaştırılmış ve 250 W'lık bir laboratuvar prototipinde test edilmiştir.

MPC tasarımı karmaşık olduğu için, bu denetleyicilerin tasarımı için genellikle bu amaç için hazırlanmış olan araç kutuları kullanılmaktadır. Currie ve Wilson (2009) MPC uygulamalarının test edilebilmesi için bir grafik kullanıcı arayüzü geliştirmişlerdir. Çalışmada, ayrıca, gömülü sistemler için oto-kod üreten ve farklı derlemelere imkan veren MPC'nin Simulink'te gerçekleştirilmesi de verilmiştir. Qin ve Badgwell (2003) dinamik bir modelden başlayarak, tamamen otomatik bir şekilde, Matlab ile bir gömülü donanım üzerinde MPC'nin gerçekleştirilmesini çalışmışlardır. Kayan noktalı aritmetik birimi olmayan genel amaçlı mikrodenetleyicilerde kullanılabilen MPC'nin geliştirilmesi Boshkovski vd. (2017) tarafından verilmiştir.

Bu çalışmada öncelikle, tasarlanacak MPC denetleyicinin performansını test etmek için, bir DC motor modeli DC motoru ifade eden matematiksel denklemlerin kullanılması ile elde edilmiştir. Daha sonra DC motorun hız ve akımını kontrol etmek için bir PI-P denetleyici tasarlanmıştır. Son olarak, DC motorun performansının iyileştirilmesi için, farklı ayar ağırlıkları ve sistem kısıtları göz önüne alınarak Matlab/Simulink ortamında bir MPC denetleyici

tasarlanmıştır. Tasarlanan MPC denetleyicinin sergilediği performans PI-P denetleyiciler ile karşılaştırılmıştır. Ayrıca, farklı çalışma koşulları altında tasarlanan MPC'nin performansı test edilmiştir.

## Model Öngörülü Kontrol:

Model Öngörülü Kontrol, kimya mühendisliği ve gıda işlemede yaygın olarak kullanılan gelişmiş bir kontrol stratejisidir. MPC çok değişkenli bir kontrol stratejisi olup, sadece süreç çıkışını referans giriş değerinde tutmaya çalışmakla kalmaz, aynı zamanda süreci önceden belirlenmiş sistem limitleri içinde tutmaya çalışır. Her bir örnekleme zamanında kısıtlı bir optimizasyon problemi çözülerek, gelecekteki kontrol dizisi elde edilir. Gelecekteki kontrol dizisi, gelecekteki kontrol yönü üzerinde, tahmin edilen çıkış ve ayar noktası ve kontrol hareketleri arasındaki karesel sapmaların ağırlıklı toplamının en aza indirileceği şekilde ilerler (Camacho & Bordons, 2007).

$$J = \sum_{i=1}^{N_p} \|y(k+i|k) - r(k+i|k)\|_0^2 + \sum_{i=0}^{N_c-1} \|\Delta u(k+i|k)\|_r^2 \quad (1)$$

Denklem (1), MPC denetleyiciye ait standart quadratik maliyet fonksiyon denklemidir. Burada,  $N_p$  ve  $N_c$ , sırasıyla, öngörü ufkunu ve kontrol ufkunu ifade etmektedir. Ayrıca;  $y$ ,  $r$  ve  $u$  sırasıyla, öngörülen çıkışı, referans girişi ve gelecekteki kontrol hareketleri dizisini ifade etmektedir. Maliyet fonksiyonu doğrusal süreç dinamikleri, giriş ve çıkışlar ve giriş değişim kısıtları ile sınırlandırılır. Süreç açık çevrim kararlı olmak zorunda değildir. Bazı matematiksel işlemler sonucunda, yukarıda verilen maliyet fonksiyonu aşağıdaki gibi düzenlenebilir (Maciejowski, 2002; Rossiter, 2016):

$$\min_{\Delta u} J = \frac{1}{2} \Delta u^T H \Delta u + F^T \Delta u \quad (2)$$

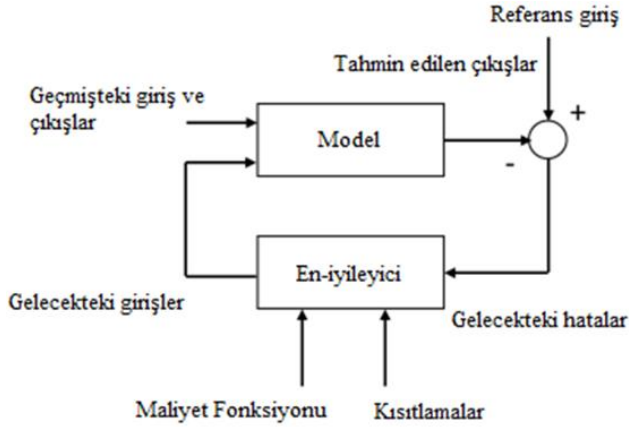
subject :  $A \Delta u \leq b$

Denklem (2)'nin türetilmesi için ilgili referanslara bakılabilir. MPC denetleyici, tek-girişli tek-çıkışlı, çok-girişli çok-çıkışlı ve kare olmayan (farklı giriş ve çıkış sayılarına sahip) sistemleri kontrol etmede kullanılabilir. Ayrıca, MPC denetleyicinin bir diğer önemli avantajı sisteme uygulanacak en-iyi kontrol sinyalinin hesaplanmasında sistem kısıtlarının göz önüne alınmasına olanak vermesidir. Dolayısıyla, MPC denetleyiciler 1970'lerden günümüze kadar süreç denetiminde tercih edilen kontrol yöntemi olmuştur.

Model Öngörülü Kontrol (MPC) ifadesi, belirli bir kontrol stratejisini değil, bir sürece ait bir modeli kullanıp maliyet fonksiyonunu minimize ederek kontrol sinyalini elde eden çok çeşitli kontrol yöntemlerini ifade eder. Doğrusal olan ve Doğrusal-olmayan MPC, Genelleştirilmiş Öngörülü Kontrol (Generalized Predictive Control, GPC), Öngörülü Fonksiyonel Kontrol (Predicted Functional Control, PFC) ve Dinamik Matris Kontrol (Dynamic Matrix Control, DMC) gibi farklı MPC algoritmaları mevcuttur. Ancak, hepsinde olan genel bir özellik mevcuttur: - Sürecin gelecekteki çıkışını tahmin edebilmek için süreç modeline ihtiyaç duyulmaktadır. Süreç modeli kullanılarak, sürecin gelecekteki çıkışı tahmin edilir. Böylece, denetleyicinin sistem dinamikleri, ölü-zaman ve kısıtların ihlali gibi durumları önceden tahmin etmesi sağlanır ve bir sonraki kontrol hareketi hesaplanır.

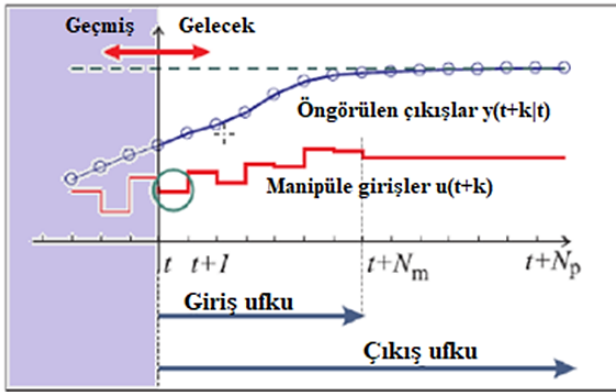
## Temel MPC Yapısı

MPC'nin temel yapısı Şekil 1'de gösterilmiştir. Geçmişteki değerler, hali hazırdaki değerler ve tasarımcı tarafından önerilen en-iyi gelecek kontrol eylemlerine dayalı olarak, gelecekteki süreç çıkışları süreç modeli kullanılarak tahmin edilir. Kontrol eylemleri, en-iyileyici tarafından maliyet fonksiyonu ve kısıtlar da göz önüne alınarak hesaplanır. Süreç modeli denetimde belirleyici bir rol oynamaktadır. Seçilen süreç modeli, gelecekteki çıkışları doğru bir şekilde tahmin edebilmek için, hem tüm süreç dinamiklerini kapsamalıdır hem de gerçekleştirilebilir ve anlaşılabilir olması için basit olmalıdır (Camacho ve Bordons, 2007).



Şekil 1. MPC'nin Temel Yapısı (Camacho ve Bordons, 2007)

MPC basık ufuk ilkesini kullanır. Basık ufuk ilkesinde, sistemin her bir  $t$  örnekleme zamanında, ilk en-iyilenmiş kontrol sinyali sisteme uygulanır. Sonra,  $t+1$  zamanında, yeni kontrol sinyalini elde etmek için optimal kontrol problemi çözülür. Bu ilke tüm MPC algoritmaları için ortak olup Şekil 2'de gösterilmiştir. Geribesleme bilgileri her örnekleme zamanında süreçten toplanırken, basık ufuk sürecin istenilen özelliklerde davranmasını sağlar.



Şekil 2. Basık ufuk stratejisi (Bemporad, 2016)

Özellikle çok-girişli ve çok-çıkışlı sistemlerde, her bir döngüde yer alan PID denetleyicilerin ayrı ayrı tasarlanması ve döngü etkileşimlerinin azaltılması veya yok edilmesi için farklı kontrol tekniklerine ihtiyaç duyulur. Bu dezavantajlar MPC ile kolayca aşılabildiğinden, MPC denetleyiciler ile PID denetleyicilere göre daha üstün bir performans elde edilir.

## DC Motorun Modellenmesi

Basitliği ve değişken hız kontrolünün uygulanması kolay olduğundan, DC motorlar endüstriyel uygulamalarda yaygın olarak kullanılmaktadır. Her fiziksel sistem, giriş ve çıkış arasındaki ilişkiyi veren transfer fonksiyonu formunda ifade edilebildiğinden, DC motorlar da dört denklem kullanılarak transfer fonksiyonu yapısında modellenebilir. Ayrıca, durum değişkenleri, durum değişkenlerinin türevleri, girişler ve çıkışlar arasındaki ilişkiyi veren durum uzayı denklemleri kullanılarak da modellenebilirler. DC motorun elektriksel kısmına ait denklem (Krishnani, 2010):

$$v_a = R_a i_a + L_a \frac{di_a}{dt} + e \quad (3)$$

ile verilir. Denklem (3) düzenlenirse

$$L_a \frac{di_a}{dt} + R_a i_a = v_a - e$$

elde edilir. Elde edilen son bağıntının Laplace dönüşümü alınırsa

$$(R_a + L_a(s))i_a = v_a - e$$

olur. Dolayısıyla, DC motorun elektrik kısmına ait transfer fonksiyonu aşağıdaki gibi olur.

$$\frac{i_a}{v_a - e} = \frac{1}{L_a s + R_a} \quad (4)$$

Mekanik kısma ait transfer fonksiyonu türetmek için Newton denklemlerinden faydalanılır:

$$J\omega + f\omega = t_{em} - t_L$$

Burada  $t_L = t_d + f\omega_m$  yukarıdaki denklemde yerine yazılıp Laplace dönüşümü alınırsa aşağıdaki bağıntı bulunur.

$$\frac{\omega}{t_{em} - t_d} = \frac{1}{Js + f} \quad (5)$$

Denklem (4) ve (5) kullanılarak, çıkış açısal hız değeri ile giriş gerilimi arasındaki transfer fonksiyonu bulunabilir:

$$G(s) = \frac{\omega(s)}{V_a(s)} = \frac{K / JL_a}{s^2 + [(JR_a + fL_a) / JL_a]s + (fR_a + K^2) / JL_a} \quad (6)$$

Benzer şekilde, açısal hız değeri ile yük torku arasındaki transfer fonksiyonu aşağıdaki gibi hesaplanabilir:

$$H(s) = \frac{\omega(s)}{t_d(s)} = \frac{(R_a + L_a)}{(R_a + L_a s)(f + Js) + K^2} \quad (7)$$

Sonuçta, DC motor açısal hız değeri aşağıdaki gibi hesaplanır:

$$\omega(s) = G(s)v_a - H(s)t_d \quad (8)$$

Bu çalışmada kullanılan DC motora ait fiziksel parametre değerleri Tablo 1’de verilmiştir (Nassani, 2012).

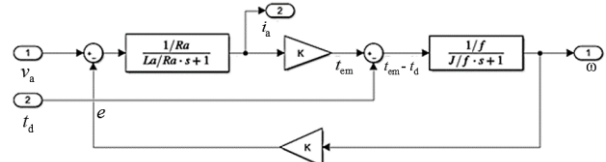
**Tablo 1.** DC Motor fiziksel parametreleri

Parametre	Nominal değeri
$R_a$ : Rotor direnci	11.8 ohm
$K$ : Elektriksel sabit	0.949 Nm/A
$L_a$ : Rotor endüktansı	0.2 H
$J$ : Eylemsizlik momenti	0.0086 kg/m <sup>2</sup>
$f$ : Viskoz sürtünme katsayısı	0.000574 Nm/rad.s
$\omega$ : Nominal açısal hız	209.4 rad/s
$t_d$ : Yük momenti (Bu çalışmada bozucu olarak varsayılmıştır).	1.58 Nm
$v_a$ : Armatür gerilimi (Sistemin kontrol girişi)	220 Volt

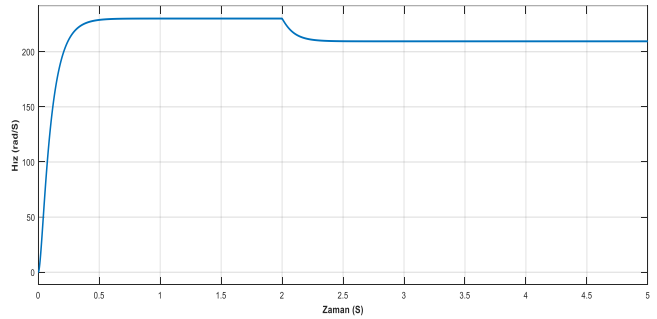
Denklem (4) ve (5) ile verilen transfer fonksiyonları ile modellenen bir DC motorun Matlab/Simulink ortamında gösterimi Şekil 3’te gösterilmiştir.

Şekil 3’te, DC motoru besleyen gerilimin doğrudan DC motora uygulanabildiği

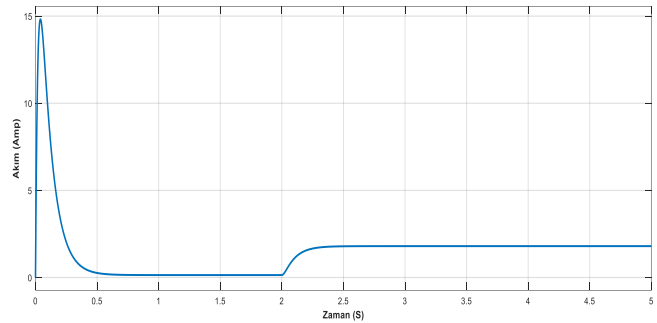
varsayılmıştır ve  $V_a = 220$  V ile verildiği kabul edilmiştir. Buna göre, motor açısal hız ve akım grafikleri, sırasıyla, Şekil 4 ve Şekil 5’te gösterilmiştir. DC motora uygulanacak gerilimi sağlayan DC-DC çevirici ideal çalıştığı varsayılarak modele dahil edilmemiştir.



Şekil 3. DC motorun Simulink gösterimi



Şekil 4. Nominal yük uygulanmadan ve uygulandıktan sonra motor açısal hız değişimi



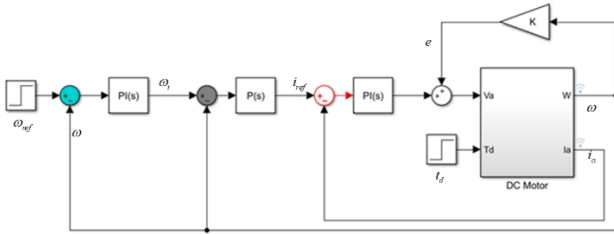
Şekil 5. Nominal yük uygulanmadan ve uygulandıktan sonra motor akımı değişimi

Motor 2 saniye boşa çalıştırıldıktan sonra 2. saniyede bir yük ile yüklenmiştir. Şekil 4'ten görüldüğü gibi, nominal yük uygulanmasında sistemin cevabında kalıcı durum hatası meydana gelmektedir. Ayrıca, Şekil 5'te motorun başlangıç akım değerinin çok büyük olduğu gözlenmektedir. Bu dezavantajlardan kurtulmak için uygun bir denetleyicinin tasarlanması gerekir. Klasik PI denetleyicinin kullanılması halinde, açısal hız değerinde meydana gelen

kalıcı durum hatasının yok edilmesi uzun zaman alacaktır. Dolayısıyla, bu çalışmada açılmal hız kalıcı durum hatasını hızlı bir şekilde yok etmek için, açılmal hız döngüsünde PI-P denetleyicinin kullanılması önerilmiştir. Önerilen PI-P denetleyicinin tasarımı bir sonraki bölümde anlatılmıştır.

## PI-P denetleyicilerin tasarımı

Bu bölümde öncelikle PI-P denetleyici ile DC motor açılmal hız denetimine yönelik tasarım verilecektir. PI-P denetleyici ile DC motor açılmal hız denetiminde kullanılan Simulink diyagramı Şekil 6'da gösterilmiştir.



Şekil 6. PI-P denetleyici ile DC motor açılmal hız denetimi (Zıt emk iç bozucu olarak kabul edilmiştir.)

Şekil 6'da, DC motor akımını kontrol etmek için kullanılan ideal PI denetleyicinin transfer fonksiyonunun

$$G_{PI,i} = K_{P-i} + \frac{K_{I-i}}{s} \quad (9)$$

ile verildiği kabul edilmiştir. Ayrıca, açılmal hız denetimi için dış döngüde transfer fonksiyonları

$$G_{P,\omega} = K_v \quad (10)$$

$$G_{PI,\omega} = K_{P-\omega} + \frac{K_{I-\omega}}{s} \quad (11)$$

ile verilen P ve PI denetleyicilerin kullanılması önerilmektedir. Şekilde görülen DC motor bloğunda, Şekil 3'te verilen DC motor eşdeğer blok diyagramı alt-sistem olarak yer almaktadır. Bu durumda, ölçülen açılmal hız ile referans açılmal hız değeri arasındaki transfer fonksiyonunun

$$\frac{i_a}{i_{a-ref}} = \frac{\frac{1}{R_a} K_{P-i} (s + \frac{K_{I-i}}{K_{P-i}})}{s(\frac{L_a}{R_a} s + 1) + \frac{1}{R_a} K_{P-i} (s + \frac{K_{I-i}}{K_{P-i}})} \quad (12)$$

ile verildiği kolaylıkla gösterilebilir. Bu çalışmada, PID tip denetleyicilerin ayar parametreleri kontrolör sentezi (Burns, 2001) yaklaşımı ile hesaplanacaktır.

(12) nolu denklemde, transfer fonksiyonuna ait sıfırın yok edilebilmesi için  $K_{I-i} / K_{P-i} = R_a / L_a$  kabul edilirse (12) nolu denklem aşağıdaki gibi sadeleştirilebilir:

$$\frac{i_a}{i_{a-ref}} = \frac{\frac{1}{R_a} K_{P-i} (s + \frac{K_{I-i}}{K_{P-i}})}{s(\frac{L_a}{R_a} s + 1) + \frac{1}{R_a} K_{P-i} (s + \frac{K_{I-i}}{K_{P-i}})}$$

$$\frac{i_a}{i_{a-ref}} = \frac{K_{P-i}}{L_a} \frac{1}{s + \frac{K_{P-i}}{L_a}} \quad (13)$$

Dolayısıyla, kapalı çevrim sistemin birinci derece bir transfer fonksiyonuna dönüştürüldüğü (13) nolu denklemde görülmektedir. Buna göre, iç döngüdeki PI denetleyicinin ayar parametrelerinin hesaplanması için, ayar parametrelerinden  $K_{P-i}$  biliniyor kabul edilip diğer ayar parametresi aşağıdaki denklemde hesaplanabilir:

$$K_{I-i} = \frac{R_a}{L_a} K_{P-i} \quad (14)$$

Dış döngüde yer alan P denetleyicinin oransal kazanç değerinin hesaplanması yine kontrolör sentezi ile yapılacaktır. Şekil 6'da  $\omega$  ile  $\omega_i$  arasındaki kapalı çevrim transfer fonksiyonunun

$$\frac{\omega}{\omega_i} = \frac{(KK_v K_{P-i} / JL_a)}{s^2 + (K_{P-i} / L_a + f / J)s + (fK_{P-i} + KK_v K_{P-i} / JL_a)} \quad (15)$$

ile verildiği hesaplanabilir. Bu transfer fonksiyonu standart ikinci derece bir transfer fonksiyonuna benzetilmek istenirse, aşağıdaki bağıntılara ulaşılabilir.

$$\omega_{n1} = \frac{1}{2\zeta} \left( \frac{K_{P-i}}{L_a} + \frac{f}{J} \right) = \frac{JK_{P-i} + fL_a}{2JL_a} \quad (16)$$

$$K_v = \frac{\omega_{n1}^2 JL_a - fK_{P-i}}{K_{P-i}K} \quad (17)$$

(16) ve (17) nolu denklemlerden,  $K_v$  değeri  $\omega_{n1}$  değerine bağlıdır.  $\omega_{n1}$  ise  $\xi$  değerine bağlıdır. Sonuçta,  $K_v$  değerinin elde edilebilmesi için  $\xi$  değerine ihtiyaç duyulmaktadır. Bu çalışmada, basamak cevapta aşımaların olmaması için  $\xi = 1$  kabul edilmiştir.

Dış döngüde yer alan PI denetleyici ayar parametreleri  $K_{P-\omega}$  ve  $K_{I-\omega}$  değerlerinin hesaplanması için:

$$\frac{\omega}{\omega_{ref}} = \frac{(KK_v K_{P-i} K_{P-\omega} / JL_a)}{s^2 + \omega_{n1}s + (KK_v K_{P-i} K_{P-\omega} / JL_a)} \quad (18)$$

ile verildiği gösterilebilir.  $K_{I-\omega} / K_{P-\omega} = \omega_{n1}$  kabul edilip, (18) nolu denklemde yerine yazılırsa ve standart ikinci derece bir sisteme benzetilirse:

$$\omega_{n2} = \frac{\omega_{n1}}{2\zeta_2} \quad (19)$$

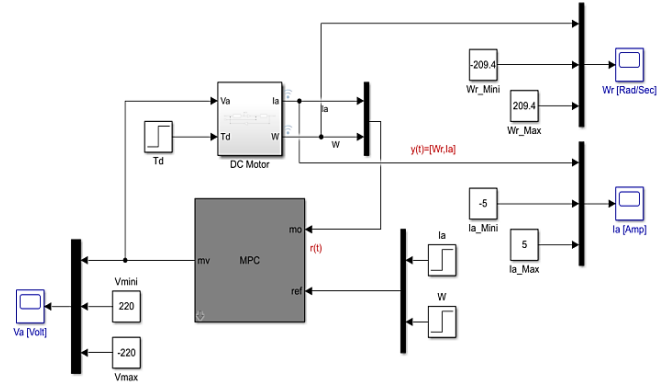
$$K_{P-\omega} = \frac{\omega_{n2}^2 JL_a}{KK_v K_{P-i}} \quad (20)$$

bağıntılarına ulaşılabilir. Bu çalışmada, (21) nolu denklemde  $\zeta_2 = 1/\sqrt{2}$  alınmıştır.

### MPC denetleyici tasarımı:

Bu bölümde, MPC denetleyici ile DC motorun açısal hız ve akım kontrolü verilecektir. Bu duruma ait Simulink diyagramı Şekil 7’de verilmiştir. Şekilde görülen DC motor bloğunda, Şekil 3’te verilen DC motor eşdeğer blok

diyagramı alt-sistem olarak yer almaktadır. Şekilde, giriş ve çıkış değişkenlerine ait kısıtlar da gösterilmiştir.



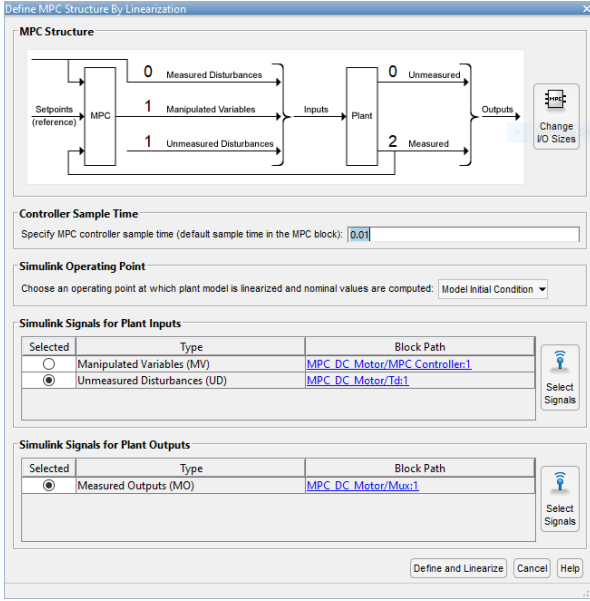
Şekil 7. MPC denetleyici ile DC motor denetimi için Simulink diyagramı

Bu çalışmada, MATLAB MPC Araç Kutusu kullanılarak MPC denetleyici tasarımı gerçekleştirilmiştir. MATLAB MPC Araç Kutusu ile tasarım üç adımda gerçekleştirilir: Birinci adımda sürece ait bir modelin tanımlanması gerekmektedir. İkinci adımda, MPC’ye ait giriş ve çıkışların, üçüncü adımda ise MPC’ye ait parametrelerin tanımlanması gerekmektedir (Bemporad, Morari ve Ricker, 2018). Dolayısıyla, öncelikle DC motorun durum uzayı gösterimi bağıntı (22)’te verildiği gibi elde edilmiştir. Öngörü süreci kullanılan modelin doğruluğuna bağlı olduğundan, modellemenin mümkün olduğu kadar hatasız olması MPC tasarımında oldukça önemlidir.

$$\begin{bmatrix} \frac{dI_a}{dt} \\ \frac{d\omega}{dt} \end{bmatrix} = \begin{bmatrix} -\frac{R_a}{L_a} & -\frac{K}{L_a} \\ \frac{K}{J} & -\frac{f}{J} \end{bmatrix} \begin{bmatrix} I_a \\ \omega \end{bmatrix} + \begin{bmatrix} \frac{1}{L_a} & 0 \\ 0 & -\frac{t_d}{J} \end{bmatrix} v_a(t) \quad (22)$$

$$y = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} I_a \\ \omega \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} D$$

MPC denetleyici tasarımında lineer süreç modeline gereksinim duyulduğundan, MPC Araç Kutusunda, Şekil 8’de gösterildiği gibi, “define and linearize” tuşuna basılarak, süreç modeli lineer olmasa bile, araç kutusu tarafından doğrusal hale dönüştürülmüş olur.



Şekil 8. MPC araç kutusunda süreç modelinin tanıtılması ve doğrusal hale getirilmesi

Süreç modelinin, Şekil 8’de gösterildiği gibi, araç kutusuna tanıtılmasından sonra, araç kutusu kontrol sisteminin giriş ve çıkış sayılarını otomatik olarak belirler. Ancak, giriş veya çıkış değişkenlerinin ölçülebilir olması veya olmaması tasarımcı tarafından değiştirilebilir.

Daha sonra, giriş ve çıkış değişkenlerinin özelliklerinin ve birimlerinin belirlenmesi gerekir. Bu durum Şekil 9’da gösterilmiştir.

Channel	Type	Name	Unit	Nominal Value	Scale Factor
u(1)	MV	va	Volt	0	1
u(2)	UD	td	N.m	0	1

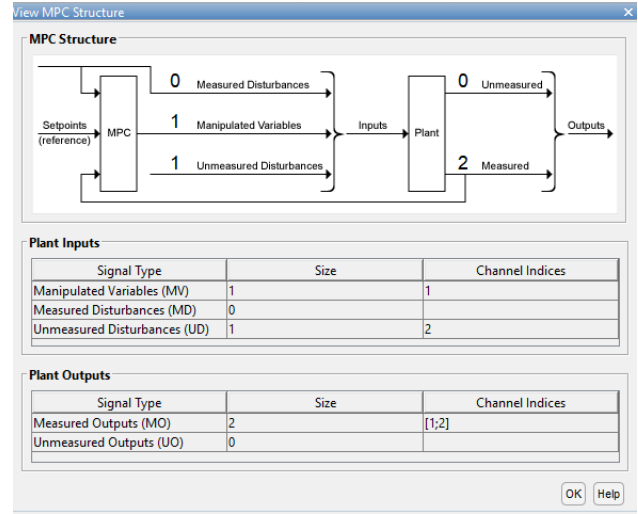
  

Channel	Type	Name	Unit	Nominal Value	Scale Factor
y(1)	MO	Current	Amp	0	1
y(2)	MO	Speed	Rad/Sec	0	1

Şekil 9. MPC IO özellikleri.

Bu çalışmada, DC motorun açısız hızını kontrol etmek ve armatür akım değerini ayarlamak için motor armatür gerilimi giriş değişkeni olarak kullanılmaktadır. Dolayısıyla,  $v_a$  kontrol değişkeni olarak, yük momenti,  $t_d$ , ise

ölçülemeyen bozucu olarak ele alınmıştır. DC motor açısız hız ve armatür akım değerleri ise çıkış değişkenleri olarak seçilmiştir. Bu durum Şekil 10’da gösterilmiştir.



Şekil 10. MPC denetleyici yapısı

Gerçek sisteme ait giriş ve çıkış değişkenlerine ait kısıtların araç kutusuna tanıtılması Şekil 11’de verilmiştir.

Channel	Type	Min	Max	RateMin	RateMax
u(1)	MV	-220	220	-Inf	Inf

Channel	Type	Min	Max
y(1)	MO	-5	5
y(2)	MO	-209.4	209.4

Şekil 11. MPC denetleyici kısıtları

DC motora ait hız ve akım cevapları MPC denetleyicinin ayarlanması ile oldukça yakından ilgilidir. MPC denetleyicinin ayarlanması giriş ve çıkış değişkenlerine atanan ağırlıklar ile belirlenir. Bu çalışmada ağırlıklar şu şekilde belirlenmiştir:

- Giriş gerilim değişkenine 0 ağırlık atanarak denetleyiciye tam bağımsızlık verilmiştir. Böylece, çıkış cevabını istediği gibi, izin



verilen aralıkta, değiştirmesine müsaade edilmiştir.

- Birinci çıkış değişkenine (akım) ikinci akım değişkenine (hız) göre daha büyük bir ağırlık atanmıştır.

Bu çalışmada giriş ve çıkış değişkenleri için seçilen ağırlıklara ait durum Şekil 12’de gösterilmiştir.

Input Weights (dimensionless)				
Channel	Type	Weight	Rate Weight	Target
u(1)	MV	0	0.5	nominal

Output Weights (dimensionless)		
Channel	Type	Weight
y(1)	MO	1
y(2)	MO	0.1

ECR Weight (dimensionless)  
Weight on the slack variable: 33201.1692

Şekil 12. MPC denetleyici ağırlıkların ayarlaması

Sistemin cevabı araç kutusunda yer alan “Performance Tuning” kısmı kullanılarak, Şekil 13’de gösterildiği gibi, Robust (Gürlüz)-Aggressive (Agresif) ve Slower (Yavaş)-Faster (Hızlı) koşulları göz önünde bulundurularak ayarlanır. Ayrıca, örnekleme periyodu, öngörü ve kontrol ufuk değerleri de bu aşamada belirlenmelidir.

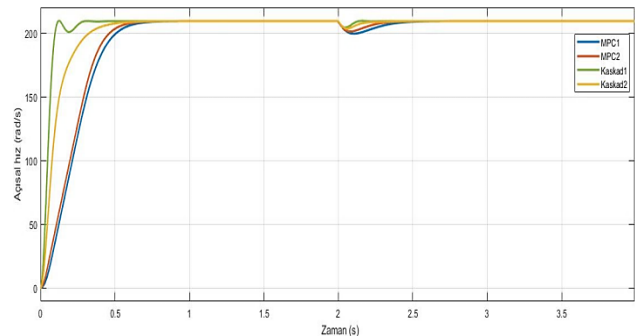


Şekil 13. MPC performans ayarı

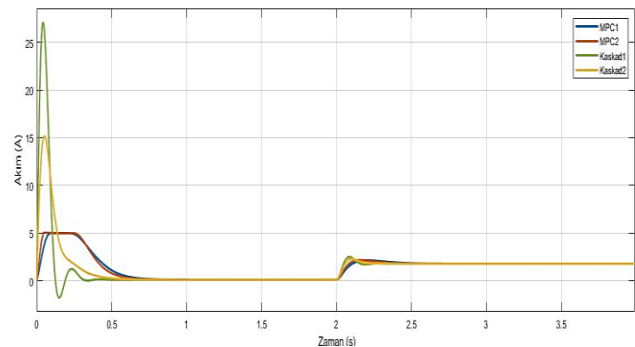
## Sonuçlar ve Tartışma

Tablo 1’de verilen DC motor parametreleri ve bir önceki bölümde anlatılan MATLAB MPC araç kutusu için gerekli değerler kullanılarak MPC denetleyici stratejisine ait benzetim sonuçları

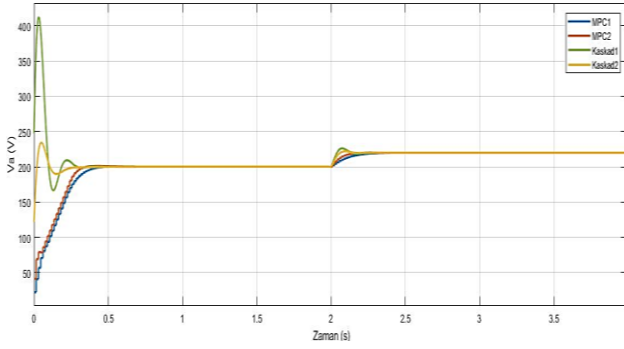
Simulink ile elde edilmiştir. Ayrıca, PI-P denetleyici tasarıma ait benzetim sonuçları da Simulink ile elde edilmiş ve her iki tasarıma ait kapalı çevrim cevaplar Şekil 14 ve 15’te gösterilmiştir. Şekillerde, MPC denetleyici için kapalı çevrim performans ayar bölümünde, orta agresif, agresif ve aşırı agresif durumlar için de cevaplar verilmiştir. PID tip denetleyici ile karşılaştırıldığında MPC denetleyicinin hem açılmal hız hem de akım denetiminde daha iyi sonuçlar verdiği gözlenmektedir. MPC denetleyici ile hem gerilim hem de akım değerlerin daha önce Şekil 11’de belirlenen sınırlarda kaldığı görülmektedir. Ancak, aşırı agresif durum için akım değeri az da olsa belirlenen sınır aşmıştır. Dolayısıyla, kapalı çevrim performans değerinin orta agresif olarak belirlenmesi gerektiği gözlenmektedir.



Şekil 14. Kaskad ve MPC denetleyiciler için elde edilen açısal hız sonuçlarının karşılaştırılması



Şekil 15. Kaskad ve MPC denetleyiciler için elde edilen akım sonuçlarının karşılaştırılması



Şekil 16. Kaskad ve MPC denetleyiciler için elde edilen kontrol sinyallerinin sonuçlarının karşılaştırılması

Tasarlanan denetleyicilere ait kapalı çevrim kontrol performansları Tablo 2’de özetlenmiştir. Kaskad yaklaşım için her ne kadar yerleşme zaman değerleri daha küçük olsa da, MPC ile denetimde aşmanın olmaması, akım değerlerinin belirlenen kısıtların dışına çıkmaması ve kontrol sinyal genliklerinin küçük olması MPC’nin avantajlarını oluşturmaktadır.

**Tablo 2.** MPC ve Kaskad denetleyici kapalı çevrim performans değerleri

	Yerleşme zamanı	Maksimum yüzde aşım
MPC (Çıkış ağırlıklarını akım için 0 ve hız 0.1 seçilmesi durumu)	0.570	0
MPC (Çıkış ağırlıklarını akım için 1.5 ve hız 0.2 seçilmesi durumu)	0.525	0
Kaskad ( $K_{p_i} = 10$ $\zeta_1 = 0.7$ ve $\zeta_2 = 0.7$ durumu)	0.240	0.131
Kaskad ( $K_{p_i} = 10$ $\zeta_1 = 0.7$ ve $\zeta_2 = 1$ durumu)	0.413	—

## Kaynaklar

Alkurawy, J., and Khamas, N. (2018) ‘Model Predictive Control for DC Motors’, in *1st International Scientific Conference of Engineering Sciences*, pp. 56–61.

- Bangia, S., Jain, S., & Neha. (2015). ‘DC Motor Control System Using Model Predictive Controller’, (02), pp. 1010–1014.
- Bemporad, A. (2016). ‘Recent Advances in Model Predictive Control’. *Institute For Advanced Studies Lucca*.
- Bemporad, A., Morari, M., & Ricker, N. L. (2018). ‘Model Predictive Control Toolbox TM User’s Guide’.
- Boshkovski, G., Stojanovski, G., & Stankovski, M. (2017). ‘Development of Embedded Model Predictive Controller’. In *IEEE International Conference on Control and Automation, ICCA* (pp. 76–81).
- Burns, S. (2001). ‘Advanced Control Engineering’. London, England: Butterworth-Heinemann.
- Camacho, E. F., & Bordons, C. (2007). ‘Model Predictive Control’. Springer.
- Currie, J., & I. Wilson, D. (2009). ‘A Model Predictive Control Toolbox Intended for Rapid Prototyping’. In *16th Electronics New Zealand Conference* (Vol. 2, pp. 7–12).
- Dani, S., Sonawane, D., Ingole, D., & Patil, S. (2017). ‘Performance evaluation of PID, LQR and MPC for DC Motor Speed Control’. In *2nd International Conference for Convergence in Technology* (Vol. 2017–Janua, pp. 348–354).
- E. Durmaz, B., Kaçmaz, B., Mutlu, İ., & Söylemez, M. (2018). ‘Implementation and Comparison of LQR-MPC on Active Suspension System’. In *10th International Conference on Electrical and Electronics Engineering* (pp. 828–835).
- Gopi, S., Vaidyan, V. M., & Vaidyan, M. V. (2013). ‘Implementation of FPGA Based Model Predictive Control For MIMO Systems’. In *Proceedings, IEEE Conference on Systems, Process and Control, Kuala Lumpur*(pp. 21–24).
- Joe Qin, S., & A. Badgwell, T. (2003). ‘A Survey of Industrial Model Predictive Control Technology’. In *Control Engineering Practice* 11 (2003) 733–764 (Vol. 11, pp. 733–764). USA: Elsevier Science.
- Krishnani, R. (2010). ‘Electric Motor Drives: Modeling, Analysis and Control’. Blacksburg, VA: Prentice Hall.
- Maciejowski, J. (2002). ‘Predictive Control With Constraints’. Prentice Hall.
- Nassani, A.,A., Nassif, M. 2012. ‘Automatic Drive Systems (1) Practical Part’.
- Prasetyo, H. F., Rohman, A. S., & Santabudi, M. R. A. R. (2018). ‘Implementation of Model Predictive Control Using Algorithm-3 on Arduino Mega 2560 for Speed Control of BLDC Motor’. In *Proceeding in 3rd International Conference on*

*Science in Information Technology: Theory and Application of IT for Education, Industry and Society in Big Data Era, ICSITech 2017 (Vol. 2018–Janua, pp. 642–647).*

Šlapák, V., Kyslan, K., Lacko, M., Fedák, V., & Ďurovský, F. (2016). 'Finite Control Set Model Predictive Speed Control of a DC Motor'. Hindawi Publishing, 2016, 1-11.

Syaichu-Rohman, A., & Sirius, R. (2011). 'Model Predictive Control Implementation On A Programmable Logic Controller For DC Motor Speed Control'. In Proceedings of *the International Conference on Electrical Engineering and Informatics, ICEEI 2011* (pp.1–4).

## Model Predictive Controller Design for DC Motor Drive

### Extended Abstract

DC Motor is one of the most popular motors used in industry field and control engineering. Due to its simplicity and easy of implementing variable speed control, DC motors have been widely used in industrial applications. In addition to being an educational system used to apply the bases of control theory, it is also a simple mechanism to find its model. DC Motor can be modelled both by a transfer function or state space representation. For the transfer function representation electrical and mechanical relations can be used to derive it. For the state space representation, relations between state variables can be used.

The purpose behind of this paper is to analyze and design a linearized control system of DC Motor and applying Model Predictive Control (MPC) strategy to investigate and improve the performance of DC motor by adjusting the tuning weights and taking into account system constraints.

The performance of designed MPC has been compared with classical PID type controllers. Hence, first of all, the transfer function of the DC motor from the electrical and mechanical equations describing it was derived. Secondly, a new controller structure called PI-P was suggested. In this structure, a PI controller in an inner loop was used to control armature current of the DC motor. In the outer loop a PI-P controller was used to control the angular velocity of the DC motor. Controller parameters were determined based on the synthesis method.

Simulation results showed that, under the PI-P controller the response for speed and current of the DC motor exceeded the nominal physical limits of the DC motor which are  $I_a = 5A$  and  $V_a = 209.4 V$ . On other hand, the results of the MPC controller were much better and showed a respect to the constraints of the DC motor system in the normal running of the simulation.

To see the effects of the robust and aggressive slider in the MPC toolbox three cases were studied. It has been observed that the mid-aggressiveness case the response for speed and current of the DC motor was stayed in the nominal physical limits of the DC motor. However, for aggressiveness and hard aggressiveness cases very slightly faster responses

for the angular velocity of the DC motor were obtained. Nevertheless, it has been seen that the armature current of the DC motor slightly exceeds the physical constraint of DC motor. Therefore it can be concluded that for the best performance, the Matlab MPC toolbox should be used with mid-aggressiveness.

This study provided results of applying one of the Model Predictive Control (MPC) strategies, which is relying on mathematical models of the DC motor, to show how this strategy works, and prove the active role that can be done by these controllers within advanced industrial systems. A comparison with classical PID controllers was given to show the high performance of MPC and test its strength.

**Keywords:** Model Predictive Control (MPC), DC Motor, PI-P Controller, Matlab, Simulink, MPC Toolbox.