

## Contents

- [Kalman filtresi uygulamaları](#)
- [Kalman filtresinin temelleri](#)
- [DC motor modeli üzerinde Kalman filtresi](#)
- [Kalman filtresi için tümleşik sistem oluşturma](#)
- [Kalman filtresi için parametre seçimi](#)
- [Kalman filtresini oluşturma](#)
- [Oluşturulan filtreyi deneme](#)
- [Farklı filtre parametreleri için sonuçlar](#)
- [Farklı filtre parametreleri için sonuçlar 2](#)
- [Model hatası olduğunu durumda Kalman filtresi kullanımına örnek](#)

## Kalman filtresi uygulamaları

Sistemlerin matematiksel modellemesi ile ilgili olarak pratikte karşılaşılan iki temel sorun vardır:

1) Sistem modelinin mükemmel olmaması, yani gerçek sistemi tam olarak temsil etmemesi. Örnekler:

- Sistemde gerçekte var olan bazı dinamiklerin modelde dikkate alınmamış olması.
- Bazı parametrelerin tam doğru seçilememiş olması.
- Sistem parametrelerini bir süre sonra değişmesi, böylece en başta modellemede kullanılmış olan değerlerinden artık farklılık göstermesi.

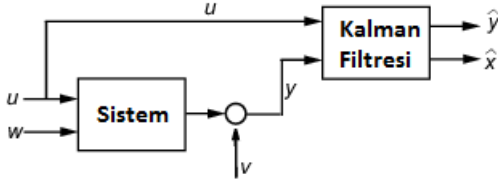
2) Sistem çıkışından alınana ölçümlerin mükemmel olmaması. Örnekler:

- Sensörlerin ölçümlerinde bir miktar hata payı olması.
- Ölçümlere gürültü karışması.

Bu durumlarda ölçülen çıkışları daha temiz bir hale getirmek ve sistemin dinamiklerini yöneten durum değişkenlerini tahmin etmek için **Kalman filtresi** kullanılabilir.

## Kalman filtresinin temelleri

Kalman filtresi sisteme verdiğimiz girişe ( $u$ ) ve ölçülen gürültülü çıkışa ( $y$ ) bakarak sistemin gerçek çıkışını ve durumunu tahmin etmeye çalışır. Bu durum şematik olarak aşağıdaki gibi ifade edilebilir:



Burada  $w$ 'ya proses gürültüsü denir ve sistem modelinde yukarıda sayılan nedenlerden dolayı meydana gelen idealden sapmaları temsil eder. Ayrıca  $v$ 'ye ölçüm gürültüsü denir de ölçümde meydana gelen sıkıntıları temsil eder. Kalman filtresinde bu gürültülerin sisteme aşağıdaki gibi etki ettiği varsayılır

$$\dot{x} = Ax + Bu + Gw$$

$$y = Cx + Du + Hw + v$$

Kalman filtresinin sistemin gerçek çıkışı için yaptığı tahmin  $\hat{y}$  ile, ve sistemin durumu için yaptığı tahmin  $\hat{x}$  ile gösterilir.

## DC motor modeli üzerinde Kalman filtresi

% DC motorunun modellenmesini geçen haftaki dersimizde görmüştük. Motor parametrelerini aşağıdaki gibi alalım:

```

R = 2.0; % Direnç (Ohm)
L = 0.5; % Endüktans (H)
Km = 0.1; % Armatür sabiti
Kb = 0.1; % EMK sabiti
Kf = 0.2; % Sürtünme sabiti (Nms)
J = 0.02; % Yük eylemsizlik momenti (kg.m^2)

```

Sistemin durum uzayı gösterimini oluşturalım:

```

Am = [-R/L -Kb/L; Km/J -Kf/J];
Bm = [1/L; 0];
Cm = [0 1];
Dm = 0;

G = ss(Am,Bm,Cm,Dm);

```

Bir kare dalga giriş altında sistemin tepkisini görelim:

```

T = 5; % Kare dalganın periyodu
t = linspace(0,5*T,1000); % Beş periyot uzunluğunda zaman vektörü oluştur
u = square(2*pi*1/T*t)'; % Kare dalgayı oluştur

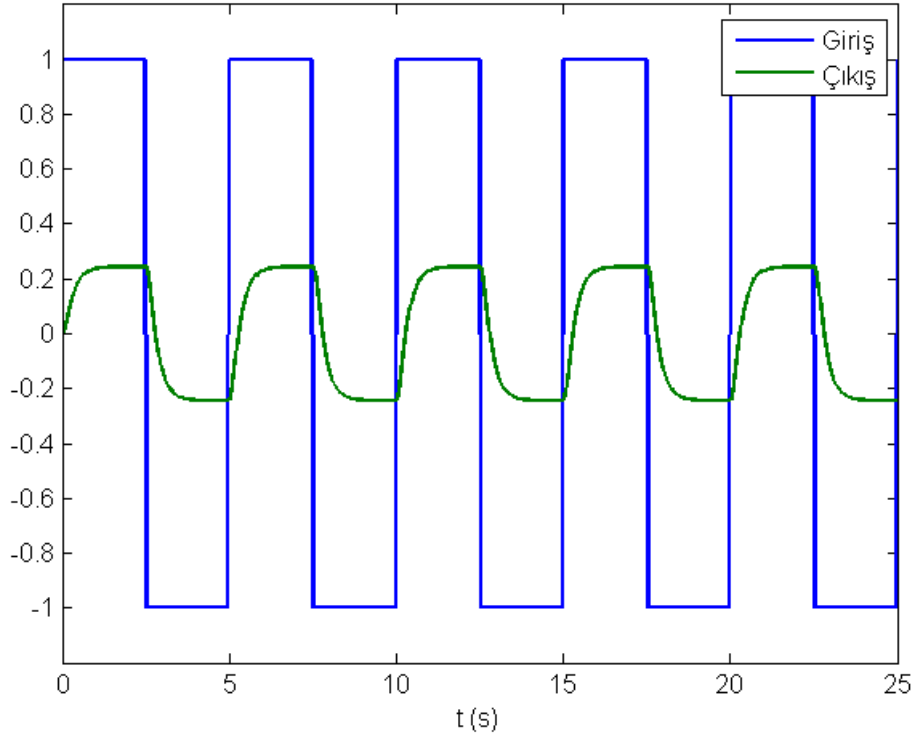
```

```

y = lsim(G,u,t); % Sistemin kare dalgaya olan tepkisini
hesaplat

plot(t,u,t,y,'LineWidth',1.5); % Kare dalgayı ve sistemin ona
tepkisini çizdir
ylim([-1.2 1.2]); % Daha güzel görünüm için y-ekseni
limitlerini ayarla
legend('Giriş','Çıkış'); % Lejand koy
xlabel('t (s)'); % x-eksenini isimlendir

```



### Kalman filtresi için tümleşik sistem oluşturma

MATLAB'da kalman filtresi oluşturmak için **kalman** fonksiyonu kullanılır. Bu fonksiyonu kullanabilmek için öncelikle girişleri  $u$  (normal sistem girişi) ve  $w$  (proses gürültüsü) olan tümleşik sistemi oluşturmalıyız. Eğer elde ekstra bir bilgi yoksa genelde proses gürültüsünün sistem girişinin üzerine eklenerek sisteme etki ettiği var sayılır. Bu durumda sistem denklemleri aşağıdaki gibi olur:

$$\dot{x} = Ax + B(u + w) = Ax + Bu + Bw$$

$$y = Cx + D(u + w) + v = Cx + Du + Dw + v$$

Bu durumda DC motorumuz için girişleri  $[u \ w]$  olan tümleşik sistem aşağıdaki gibi oluşturulabilir:

```

sysTum = ss(Am,[Bm Bm],Cm,[Dm Dm]);

```

## Kalman filtresi için parametre seçimi

Kalman filtresi oluştururken ayrıca, ilk başlıkta anlatılan iki sıkıntıdan (1- Modelin mükemmel olmaması, 2- Ölçümlerin mükemmel olmaması) hangisinin daha baskın olduğunu beklediğimizin bilgisi de kullanılır. Birinci durumu temsil etmek için  $w$  gürültüsünün  $v$ 'ye göre daha yüksek olduğu kabul ederiz. İkinci durumda ise  $v$  gürültüsünün  $w$ 'ye göre daha yüksek olduğunu kabul ederiz. Bu beklentimizi  $Q$  ve  $R$  parametreleri ile temsil ederiz:

- Eğer  $Q > R$  alıyorsak modelimizle ilgili sıkıntıların daha baskın olduğunu varsayıyoruz demektir.
- Eğer  $R > Q$  alıyorsak o zaman ölçümlerle ilgili sıkıntıların daha baskın olduğunu varsayıyoruz demektir.
- Eğer hangi sıkıntının daha baskın olduğunu bilmiyorsak gürültülerin etkilerini eşit varsayabiliriz, yani  $Q=R$  alırız.

Filtrenin performansına göre daha sonra  $Q$  ve  $R$ 'yi değiştirip tekrar deneyebiliriz. Teoride  $Q$  ve  $R$  aslında gürültü kovaryans matrisleri olarak isimlendirilir ve gürültü kovaryanslarının tam olarak bilindiği durumlarda bu değerler kullanılırsa Kalman filtresinin çok iyi sonuç verdiği ispatlanabilir. Ancak gerçek problemlerde bu değerleri bilmek ve hesaplamak genelde mümkün değildir, pratikte bu değerler genellikle deneyerek, sonuçlara bakıp  $Q$  ve  $R$ 'yi değiştirip tekrar deneyerek ve bu şekilde birkaç iterasyon yaparak bulunur.

## Kalman filtresini oluşturma

Tümleşik sistemi oluşturduktan seçtiğimiz  $Q$  ve  $R$  parametrelerine göre Kalman filtresini oluşturmak için **kalman** komutunu kullanırız. Örnek olarak  $Q=R=1$  alarak (dolayısıyla modelde ve ölçümlerde sıkıntı olma şansını eşit farzederek bir Kalman filtresi tasarlayalım:

```
Q = 1; R = 1; % Filtre parametreleri
Kest1 = kalman(sysTum,Q,R) % Kalman filtresini tasarla
```

```
a =
      x1_e      x2_e
x1_e      -4    -0.3719
x2_e       5    -10.09
```

```
b =
      u1      y1
x1_e      2    0.1719
x2_e      0    0.08559
```

```
c =
```

	<b>x1_e</b>	<b>x2_e</b>
<b>y1_e</b>	0	1
<b>x1_e</b>	1	0
<b>x2_e</b>	0	1

d =

	<b>u1</b>	<b>y1</b>
<b>y1_e</b>	0	0
<b>x1_e</b>	0	0
<b>x2_e</b>	0	0

Input groups:

<b>Name</b>	<b>Channels</b>
KnownInput	1
Measurement	2

Output groups:

<b>Name</b>	<b>Channels</b>
OutputEstimate	1
StateEstimate	2,3

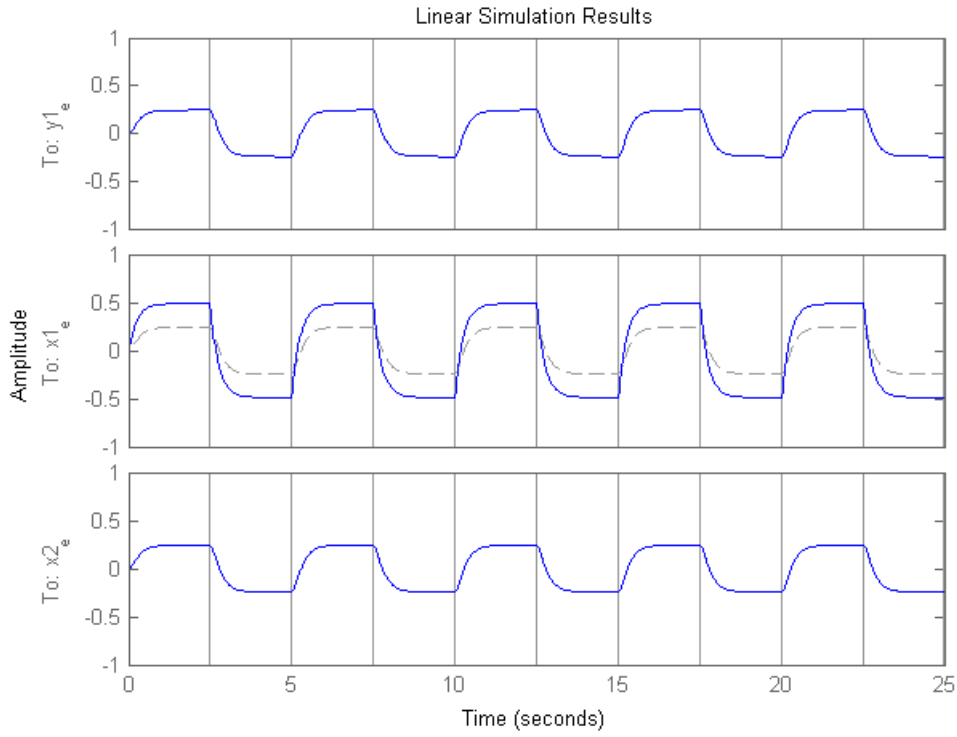
Continuous-time state-space model.

Oluşan filtrenin bir durum uzayı sistemi olduğu görülmektedir. Filtrenin iki girişi (u1=Sisteme uygulanan giriş, y1:Sistemden alınan gürültülü ölçümler) ve üç çıkışı (Sistemin temiz çıkışının ne olduğu ile ilgili tahmin, durum vektörülerinin değerlerinin ne olduğu ile ilgili tahmin) olduğu görülebilir.

### **Oluşturulan filtreyi deneme**

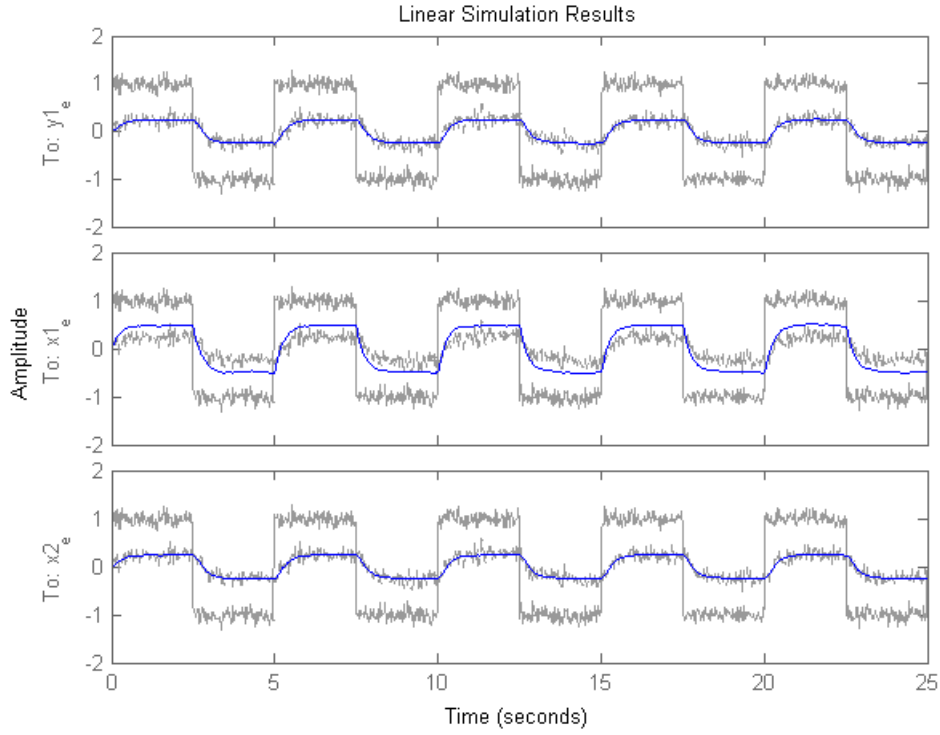
Oluşturduğumuz filtreyi denemek için önce temiz giriş ve çıkışlarla elde ettiğimiz sonuca bakalım.

```
lsim(Kest1,[u y],t); % Kalman filtresinin tepkisini gör
```



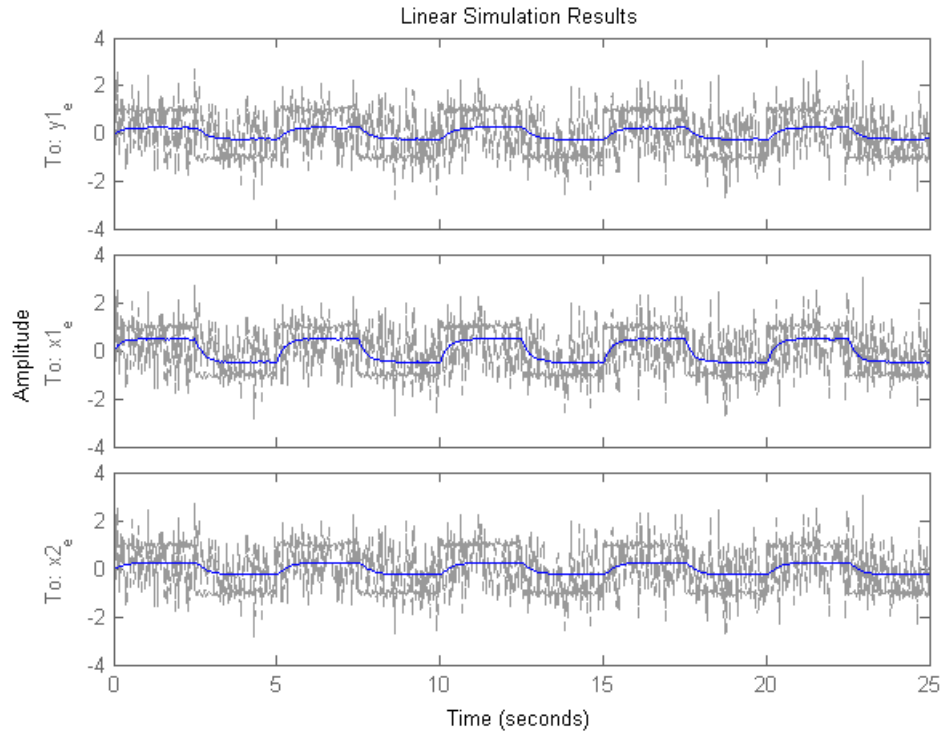
Daha sonra filtrenin gerçek kullanımına daha uygun bir duruma örnek olarak giriş ve çıkışları bir miktar gürültü ile kirleterek filtreye verelim. Örneğin hem girişe hem çıkışa 0.1 genlikli bir rastgele gürültü ekleyelim:

```
u2 = u + 0.1*randn(size(u)); % Girişe 0.1 genlikte gürültü  
ekle  
y2 = y + 0.1*randn(size(u)); % Çıkışa 0.1 genlikte gürültü  
ekle  
lsim(Kest1,[u2 y2],t); % Kalman filtresinin tepkisi gör
```



Filtreyi tasarlarken giriş ve çıkışta aynı miktarda gürültü olacağını varsaymıştık ve yukarıda da bu şekilde bir durumu denedik. Tabii ki pratikte gürültüler hiçbir zaman çok kesin bir şekilde tahmin edilmeyeceğinden giriş ve çıkış gürültüsünün de her zaman aynı olması beklenemez ve filtremiz zaman zaman tasarım koşullarından farklı durumlarda çalışmak zorunda kalabilir. Bu durumu temsil etmek için, örneğin çıkış gürültüsünün giriş gürültüsüne göre daha yüksek olduğu bir durum için filtreyi çalıştırıp sonuçları görelim

```
u2 = u + 0.1*randn(size(u)); % Girişe 0.1 genlikte gürültü ekle
y2 = y + 1*randn(size(u)); % Çıkışa 1 genlikte gürültü ekle
lsim(Kest1,[u2 y2],t); % Kalman filtresinin tepkisini gör
```

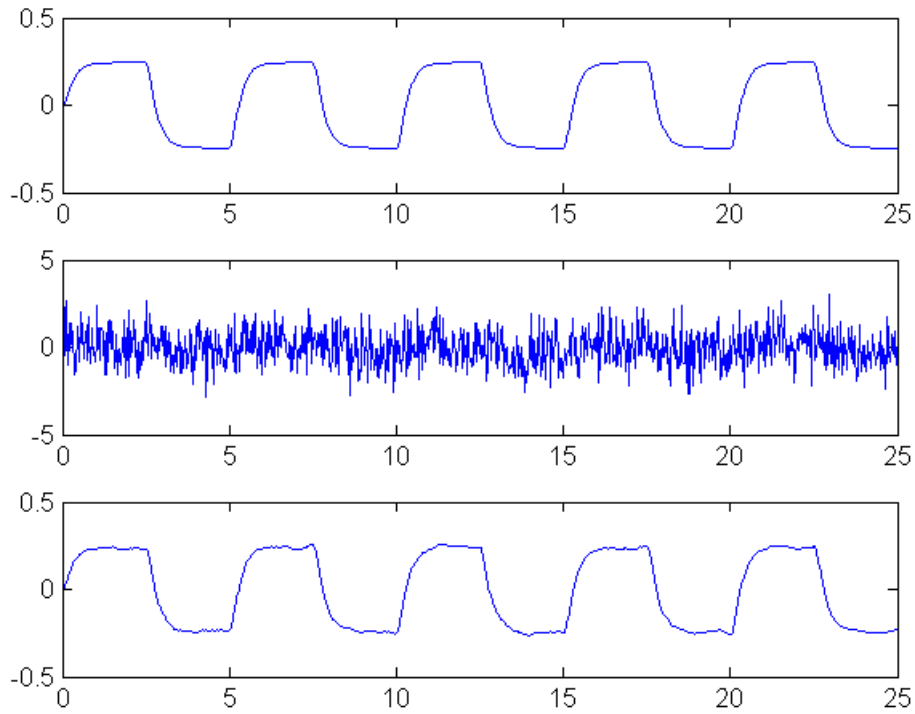


Gürültüsüz çıkışı, gürültülü çıkışı, ve Kalman filtresinin çıkış için oluşturduğu tahmini bir arada görelim:

```
figure; % Yeni şekil aç
yest = lsim(Kest1,[u2 y2],t); % Kalman filtresi tepkisini
hesaplat

subplot(3,1,1); % Alt şekli seç
plot(t,y); % Gürültüsüz çıkış
subplot(3,1,2); % Alt şekli seç
plot(t,y2); % Gürültülü çıkış
subplot(3,1,3); % Alt şekli seç
plot(t,yest(:,1)); % Kalman filtresinin tahmini
```





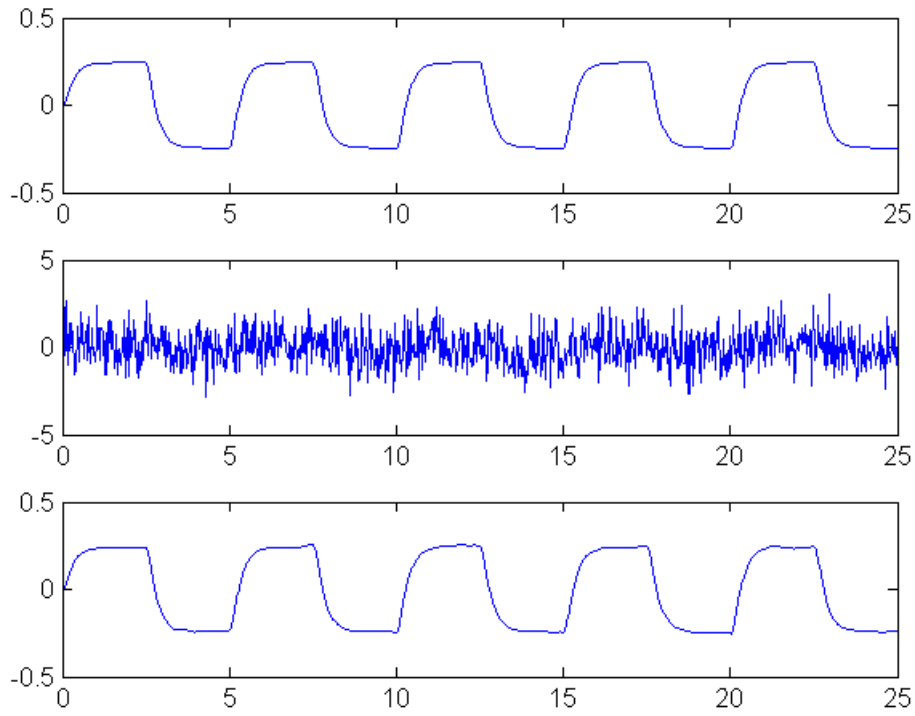
### Farklı filtre parametreleri için sonuçlar

Eğer nispeten güvenilir bir modelle çalıştığımızı ama ölçüm yapan sensörlerde hata/gürültü olma ihtimalinin yüksek olduğunu düşünüyor olsaydık, Kalman filtresi tasarımı yaparken bu durumu temsil etmek için  $R > Q$  seçerdik. Örnek olarak  $Q=1$ ,  $R=1000$  alalım, filtreyi tasarlayalım ve yukarıdaki giriş ve çıkışa verilen tepkiyi görelim

```
Q = 1; R = 1000; % Filtre parametreleri
Kest2 = kalman(sysTum,Q,R); % Kalman filtresini tasarla

figure; % Yeni şekil aç
yest = lsim(Kest2,[u2 y2],t); % Kalman filtresi tepkisini hesaplat

subplot(3,1,1); % Alt şekli seç
plot(t,y); % Gürültüsüz çıkış
subplot(3,1,2); % Alt şekli seç
plot(t,y2); % Gürültülü çıkış
subplot(3,1,3); % Alt şekli seç
plot(t,yest(:,1)); % Kalman filtresinin tahmini
```



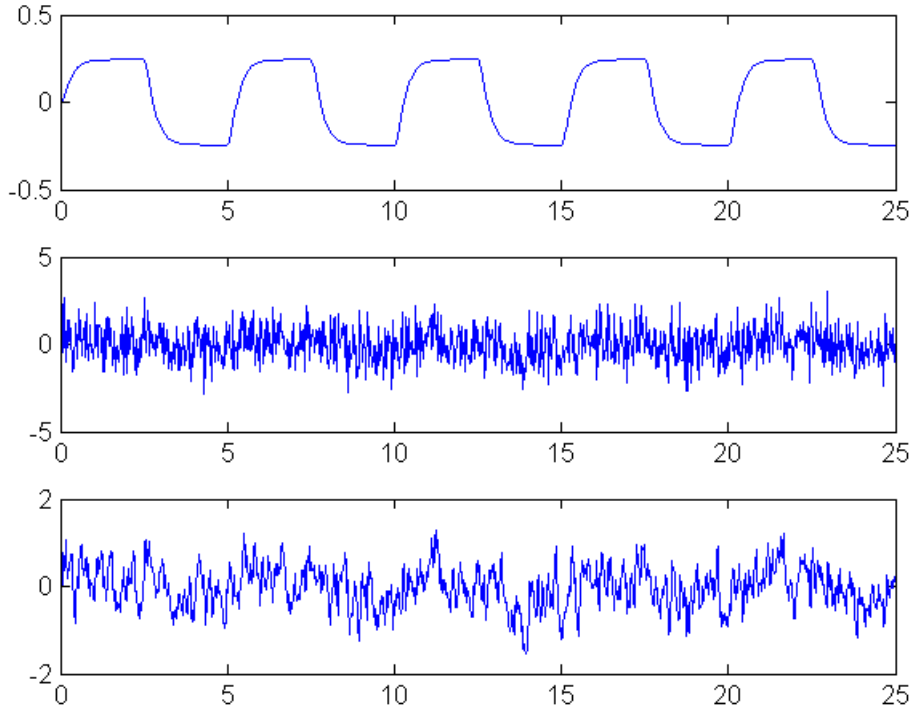
## Farklı filtre parametreleri için sonuçlar 2

Eğer nispeten güvenilir sensörlerle çalıştığımızı düşünseydik ama modelle ilgili sıkıntılar olma ihtimalinin yüksek olduğuna inansaydık, Kalman filtresi tasarımı yaparken bu durumu temsil etmek için  $Q > R$  seçerdik. Örnek olarak  $Q=1000$ ,  $R=1$  alalım, filtreyi tasarlayalım ve yukarıdaki giriş ve çıkışa verilen tepkiyi görelim:

```
Q = 1000; R = 1; % Filtre parametreleri
Kest3 = kalman(sysTum,Q,R); % Kalman filtresini tasarla

figure; % Yeni şekil aç
yest = lsim(Kest3,[u2 y2],t); % Kalman filtresi tepkisini hesaplat

subplot(3,1,1); % Alt şekli seç
plot(t,y); % Gürültüsüz çıkış
subplot(3,1,2); % Alt şekli seç
plot(t,y2); % Gürültülü çıkış
subplot(3,1,3); % Alt şekli seç
plot(t,yest(:,1)); % Kalman filtresinin tahmini
```



### Model hatası olduğunu durumda Kalman filtresi kullanımına örnek

Kalman filtresini tasarlarken kullandığımız model ile, gerçekteki model arasında farklılıklar olduğunu durumu inceleyelim. Böyle bir durum pratikte sıkça karşımıza çıkabilir çünkü:

- Fiziksel sistemlerin parametreleri her zaman tam doğrulukla belirlenemeyebilir.
- Sistem parametreleri aşınma, yıpranma, yaşlanma vb. sebeplerden zaman içinde ilk değerlerinden farklılık gösterebilir.
- Sistemin bazı karmaşık dinamikleri (örneğin doğrusal olmayan dinamikler) işlemleri basitleştirmek için ihmal edilmiş olabilir.

Bu gibi durumlarda Kalman filtresini oluştururken kullandığımız model ile, gerçekte sistemin davranışını belirleyen dinamikler arasında farklılıklar oluşacaktır. Model ile ilgili sıkıntıları temsil etmek için proses gürültüsünü ( $w$ ) temsil eden  $Q$  parametresini yüksek tutabiliriz. Bu durumda Kalman filtresi ölçümlere daha çok, filtrenin içindeki modelin tahminlerine daha az ağırlık verecektir. Böylece, ölçümlerin  $sw$  yeterinde iyi olduğunu varsayarsak, nispeten temiz çıkış ve durum tahminleri oluşturabilecektir.

Örnek olarak DC motor sistemin gerçek parametreleri aşağıdaki gibi olsun:

```
R = 3.0; % Direnç (Ohm)
L = 1.5; % Endüktans (H)
Km = 0.5; % Armatür sabiti
Kb = 0.4; % EMK sabiti
Kf = 0.1; % Sürtünme sabiti (Nms)
J = 0.2; % Yük eylemsizlik momenti (kg.m^2)
```

Sistemin durum uzayı gösterimini oluşturalım:

```
Am2 = [-R/L -Kb/L; Km/J -Kf/J];
Bm2 = [1/L; 0];
Cm2 = [ 0 1 ];
Dm2 = 0;

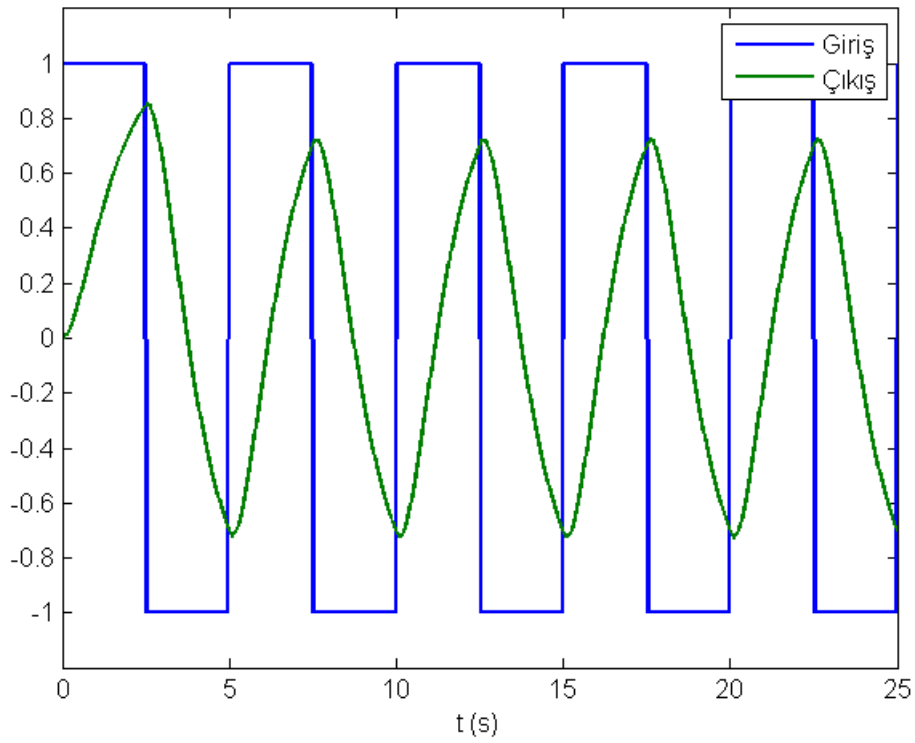
G2 = ss(Am2,Bm2,Cm2,Dm2);
```

Bir kare dalga giriş altında sistemin tepkisini görelim:

```
T = 5; % Kare dalganın periyodu
t = linspace(0,5*T,1000); % Beş periyot uzunluğunda zaman
vektörü oluştur
u = square(2*pi*1/T*t)'; % Kare dalgayı oluştur

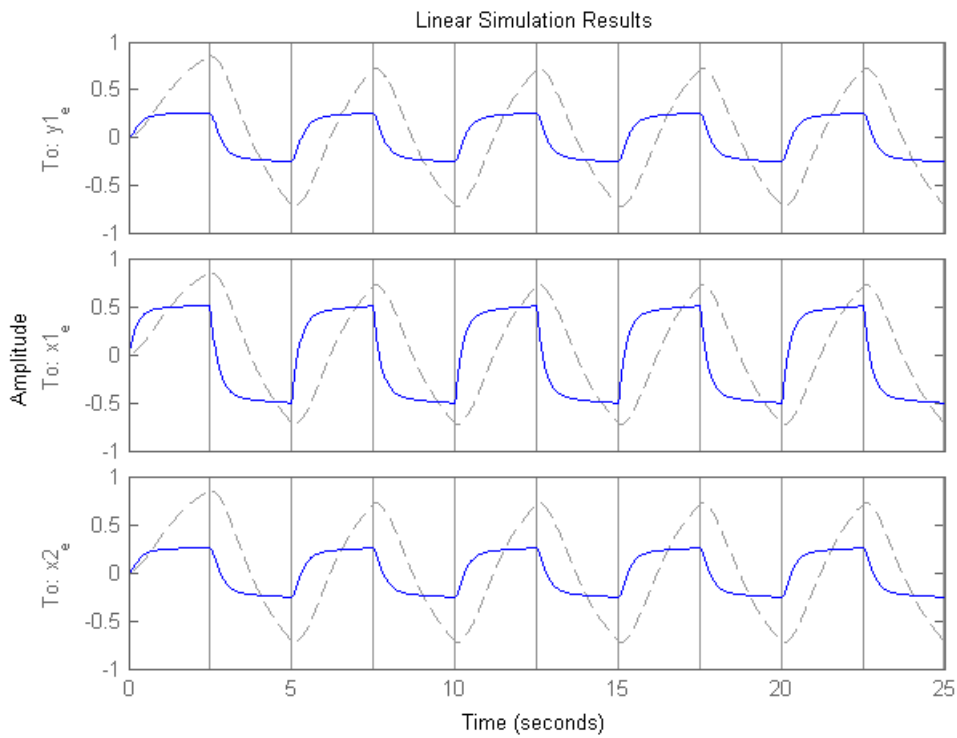
y = lsim(G2,u,t); % Sistemin kare dalgaya olan tepkisini
hesaplat

figure;
plot(t,u,t,y,'LineWidth',1.5); % Kare dalgayı ve sistemin ona
tepkisini çizdir
ylim([-1.2 1.2]); % Daha güzel görünüm için y-ekseni
limitlerini ayarla
legend('Giriş','Çıkış'); % Lejand koy
xlabel('t (s)'); % x-eksenini isimlendir
```



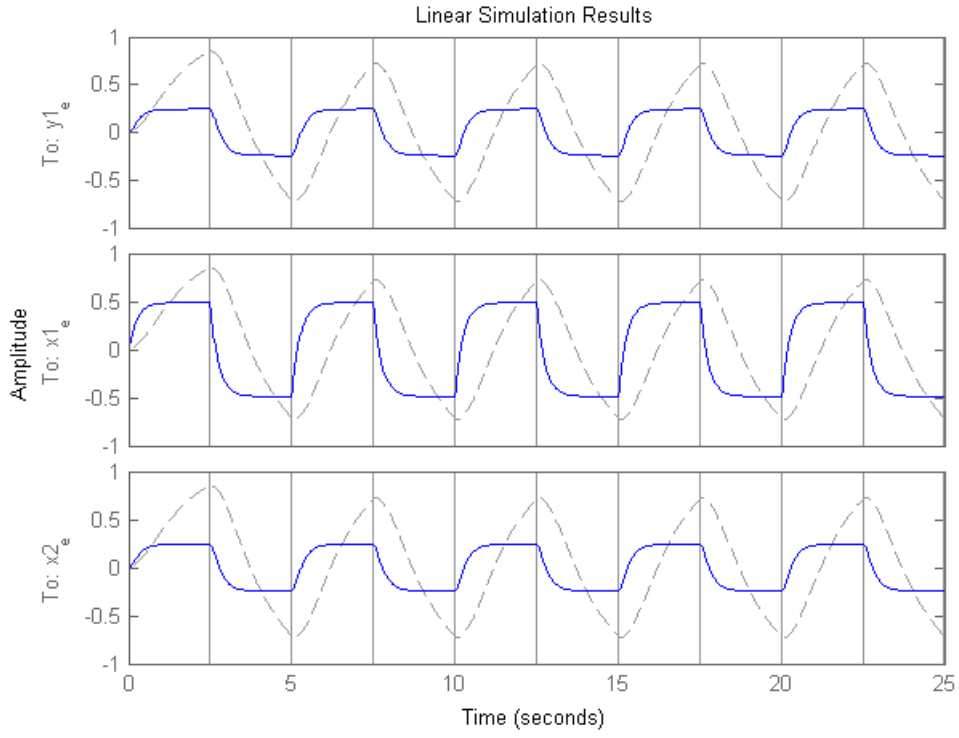
Oluşturduğumuz bu sistemin giriş ve çıkışlarını, daha önceden farklı model parametreleri ile oluşturmuş olduğumuz Kalman filtrelerine sokup sonuçları görelim. Önce  $Q=1$ ,  $R=1$  olarak tasarlanmış filtre için sonuçları görelim:

```
lsim(Kest1,[u y],t);
```



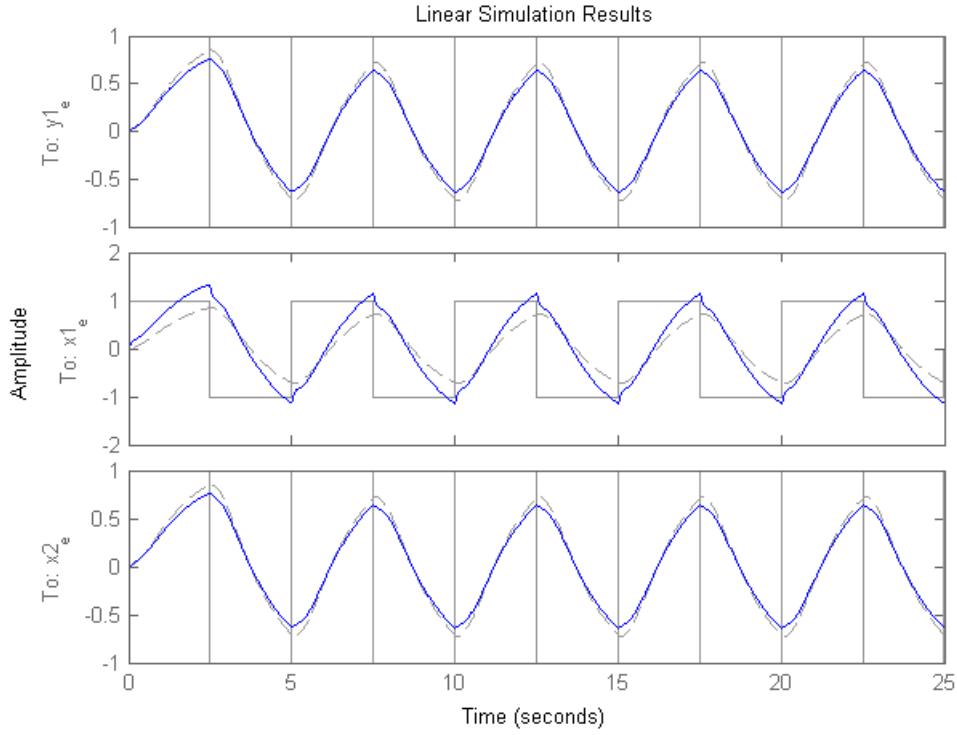
Q=1, R=1000 olarak tasarlanmış filtre için sonuçlar:

```
lsim(Kest2,[u y],t);
```



Q=1000, R=1 olarak tasarlanmış filtre için sonuçlar:

```
lsim(Kest3,[u y],t);
```



Görüldüğü gibi son fitre gerçeğe en yakın tahminleri vermektedir. Bunun sebebi, bu filtrenin modelde meydana gelebilecek sıkıntıların daha fazla olacağı beklentisi altında tasarlanmış olmasıdır ( $Q > R$ ). Bunun sonucunda filtre, kendi içindeki nispeten kötü olan modele çok fazla güvenmemekte ve daha çok almış olduğu ölçümlere dayarak çıkışlarını oluşturmaktadır. Model sıkıntılı, ölçümler daha sıkıntısız olduğu için filtrenin bu yaklaşımı olumlu sonuçlar doğurmaktadır.