Sistemas Inteligentes: Filtragem Estocástica

Nelson Carlos de Sousa Campos

Maio de 2016

1 Introdução

Em estatística, o filtro de Kalman é um método matemático criado por Rudolf Kalman. Seu propósito é utilizar medições de grandezas realizadas ao longo do tempo (contaminadas com ruído e outras incertezas) e gerar resultados que tendam a se aproximar dos valores reais das grandezas medidas e valores associados.

O filtro de Kalman apresenta diversas aplicações e é uma parte essencial do desenvolvimento de tecnologias espaciais e militares. Talvez o tipo mais usado e simples do filtro de Kalman seja em phase-locked loop (malhas de captura de fase), bastante comuns em rádios FM e na maioria dos equipamentos de telecomunicações existentes. Extensões e generalizações do método também foram desenvolvidas.

O filtro de Kalman produz estimativas dos valores reais de grandezas medidas e valores associados predizendo um valor, estimando a incerteza do valor predito e calculando uma média ponderada entre o valor predito e o valor medido. O peso maior é dado ao valor de menor incerteza. As estimativas geradas pelo método tendem a estar mais próximas dos valores reais que as medidas originais pois a média ponderada apresenta uma melhor estimativa de incerteza que ambos os valores utilizados no seu cálculo. [3] [4]

Neste trabalho, o filtro de Kalman Estendido será aplicado a um problema de rastrear uma senóide,cujos parâmetros devem ser estimados. O filtro estendido é utilizado em sistemas não-lineares.[5]

2 Um modelo simples

Suponha o sinal senoidal da equação (1):

$$x(t) = Asin(\omega t) \tag{1}$$

Definindo $\phi = \omega t$, $\dot{\phi} = \omega$, e assumindo uma frequência e amplitude constantes para esta senóide, tem-se ainda: $\dot{\omega} = 0$ e $\dot{A} = 0$. A equação no espaço de estados para este sinal em função de ϕ,ω e A é dada pela equação (2), onde u_{s_1} e u_{s_2} são, respectivamente, as incertezas de ω e A serem constantes.

$$\begin{bmatrix} \dot{\phi} \\ \dot{\omega} \\ \dot{A} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \phi \\ \omega \\ A \end{bmatrix} + \begin{bmatrix} 0 \\ u_{s_1} \\ u_{s_2} \end{bmatrix} \tag{2}$$

Seja Q a matrix contínua das densidades espectrais de potência dos ruídos u_{s_1} e u_{s_2}. Admitindo um ruído branco e que eles são descorrelacionados, Q pode ser escrita como:

$$Q = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \Phi_{s_1} & 0 \\ 0 & 0 & \Phi_{s_2} \end{bmatrix} \tag{3}$$

Sendo F a matriz dinâmica do sistema, onde:

$$F = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{4}$$

e

$$F^2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{5}$$

fica evidente que a matriz $\Phi(t) = e^{Ft} = I + Ft$ é dada por:

$$\Phi(t) = \begin{bmatrix} 1 & t & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{6}$$

e sua versão discretizada com período de amostragem T_s é:

$$\Phi(kT_s) = \begin{bmatrix} 1 & T_s & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{7}$$

O vetor de medições do sinal é definido como sendo o produto escalar do gradiente do sinal em relação ao vetor de estados pela variação desses estados, contaminador por um ruído de medição, conforme a equação (8):

$$x^* =< \nabla x, \begin{bmatrix} \Delta\phi & \Delta\omega & \Delta A \end{bmatrix} > +v \tag{8}$$

onde v é uma variáel gaussiana com média nula e variância σ_v.

Linearizando as medições, obtém-se:

$$C = \begin{bmatrix} Acos(\phi) & 0 & sin(\phi) \end{bmatrix} \tag{9}$$

A estimação dos estados pelo filtro de Kalman estendido pode ser obtida iterativamente pela matriz dinâmica do sistema em sua forma discreta:

$$\begin{bmatrix} \bar{\phi_k} \\ \bar{\omega_k} \\ \bar{A_k} \end{bmatrix} = \begin{bmatrix} 1 & T_s & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{\phi}_{k-1} \\ \hat{\omega}_{k-1} \\ \hat{A}_{k-1} \end{bmatrix} \tag{10}$$

Uma vez definida a função de atualização de estados do sistema, o filtro de Kalman extendido pode ser aplicado.

O algoritmo do filtro é dado como segue:

- Determine a matriz de covariância $P = F * P * F^T + Q$
- Determine o fator de ajuste $K = P * C^T * (C * P * C^T + R)^{-1}$;
- Determine as equações dinânimicas das variáveis de estados: $x(k) = f(x(k-1), v)$
- Atualize as variáveis de estados estimadas: $x(k) = x(k-1) + K * (ym(k) - ye(k))$
- Atualize a matriz de covariância: $P(k) = P(k) - K * C * P(k-1)$

A Figura 2 mostra a estimação de uma senóide, contaminada por ruído e também a estimação de sua frequência ω.

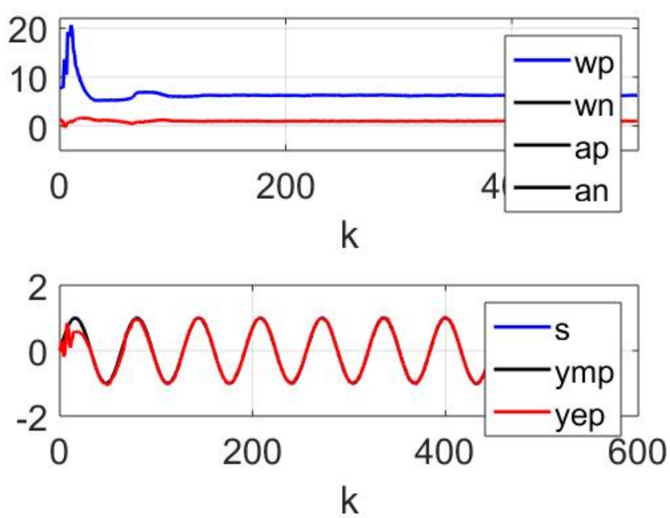

Figura 1: Estimação de uma senóide e de sua frequência angular - Modelo 1

3 Um modelo ainda mais simples

Supondo que no modelo da equação (1) a incerteza está presente apenas na consância da frequência e ignorando o ruído associado à derivada da amplitude, a dimensão das matrizes das equações (2) à (7) são reduzidas para ordem 2, e os elementos dessas matrizes são compostos pelas matrizes de ordem 3, eliminando-se a terceira linha e a terceira coluna. As equações (8) e (9) são adaptadas removendo-se a terceira coluna das matrizes presentes nestas equações. Sendo assim, a dinâmica de evolução das variáveis de estados são dadas pela equação (11).

$$\begin{bmatrix} \bar{\phi_k} \\ \bar{\omega_k} \end{bmatrix} = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{\phi}_{k-1} \\ \hat{\omega}_{k-1} \end{bmatrix} \tag{11}$$

A Figura **??** mostra a estimação de uma senóide, contaminada por ruído e também a estimação de sua frequência ω, na modelagem do sistema, ignorando-se qualquer informação em relação à amplitude.

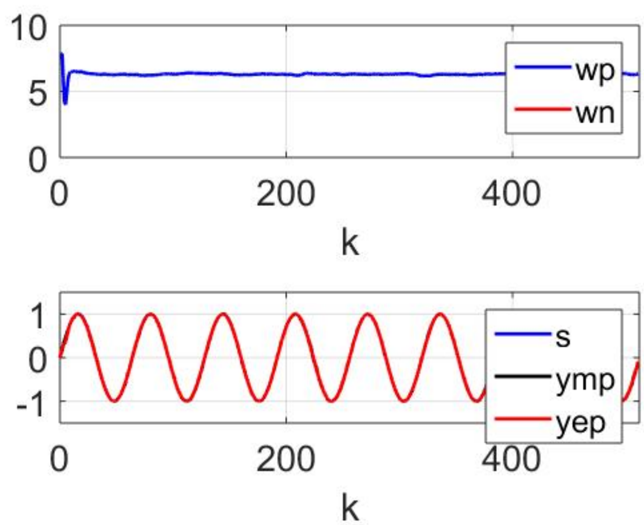

Figura 2: Estimação de uma senóide e de sua frequência angular - Modelo 2

4 Uma modelagem alternativa

As derivadas de primeira e segunda ordem da equação (1) são dadas pelas equações (12) e (13) respectivamente:

$$\dot{x} = A\omega cos(\omega t) \tag{12}$$

$$\ddot{x} = -A\omega^2 sin(\omega t) = -\omega^2 x \tag{13}$$

Procedento de maneira similar à seção 2, são obtidas as seguinte equações:

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ -\omega^2 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \dot{\omega} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ u_s \end{bmatrix} \tag{14}$$

A equação (14) não é linear. A matriz dinâmica do sistema é obtida de acordo com a equação (15).

$$F = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} = \begin{bmatrix} 0 & 1 & 0 \\ -\hat{\omega}^2 & 0 & -2\hat{\omega}\hat{x} \\ 0 & 0 & 0 \end{bmatrix} \tag{15}$$

E a matriz $\Phi(t)$ fica:

$$\Phi(t) \approx I + Ft = \begin{bmatrix} 1 & t & 0 \\ -\hat{\omega}^2 t & 1 & -2\hat{\omega}\hat{x}t \\ 0 & 0 & 1 \end{bmatrix} \tag{16}$$

As equações de evolução das variáveis de estados são:
$\dot{x}_1 = x_2$
$\dot{x}_2 = x_1 x_3^2$
$\dot{x}_3 = u_s$
Aplicando este modelo no algotimo do filtro de Kalman extendido, obteve-se o resultado da Figura 3.

5 Modificação da modelagem alternativa

Suponha agora que a equação (13) seja descrita como:

$$\ddot{x} = -zx \tag{17}$$

com $\dot{z} = u_s$. Dito isto, as equações (14) à (16) se tornam as equações (18) à (20):

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ -z & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ z \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ u_s \end{bmatrix} \tag{18}$$

$$F = \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} = \begin{bmatrix} 0 & 1 & 0 \\ -\hat{z} & 0 & \hat{x} \\ 0 & 0 & 0 \end{bmatrix} \tag{19}$$

$$\Phi(t) \approx I + Ft = \begin{bmatrix} 1 & t & 0 \\ -\hat{z}t & 1 & \hat{x}t \\ 0 & 0 & 1 \end{bmatrix} \tag{20}$$

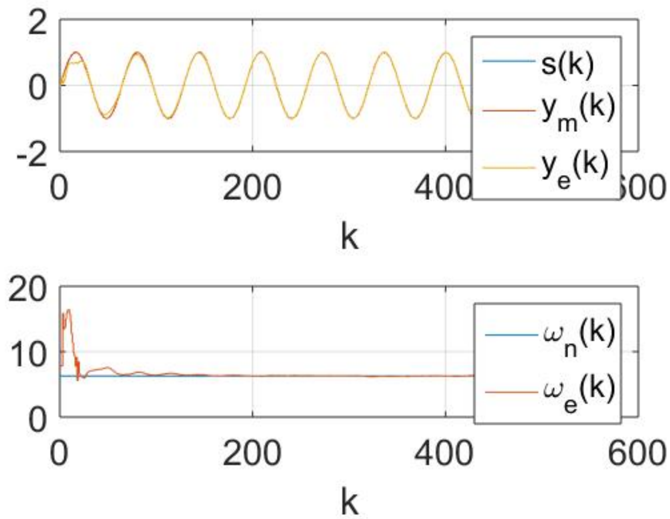

Figura 3: Estimação de uma senóide e de sua frequência angular - Modelo 3

A dinâmica de evolução das variáveis de estado neste modelo é descrita por:
$\dot{x}_1 = x_2$
$\dot{x}_2 = x_1 x_3$
$\dot{x}_3 = u_s$
Aplicando-se o filtro de Kalman estendido, os resultados são apresentandos na Figura 4

6 Análise dos Resultados e Conclusão

O filtro de Kalman extendido proposto pelo método 1 é eficaz quando se quer estimar frequências e amplitudes positivas quando as condições iniciais no algoritmo são positivas. Quando as condições iniciais são negativas, o filtro não consegue estimar estes parâmetros, conforme Figura 5.

O filtro proposto pelo método 2 consegue estimar a frequência do sinal mesmo quando a estimativa inicial é negativa conforme Figura 6.

O filtro proposto pelo método 3 não consegue estimar a frequência do sinal quando a estimativa inicial é negativa conforme Figura 7.

Por outro lado, o filtro proposto pelo método 4 praticamente não variou a estimativa dos parâmetros variando-se o sinal dos valores iniciais no algoritmo, como pode ser visto na Figura 8.

Todos os códigos das implmentações em Matlab do filtros encontram-se em anexos.

A aplicação do filtro de Kalman se dá de maneira eficiente quando se tem por objetivo o rastreamento de sinais. Essas aplicações podem ser estendidas

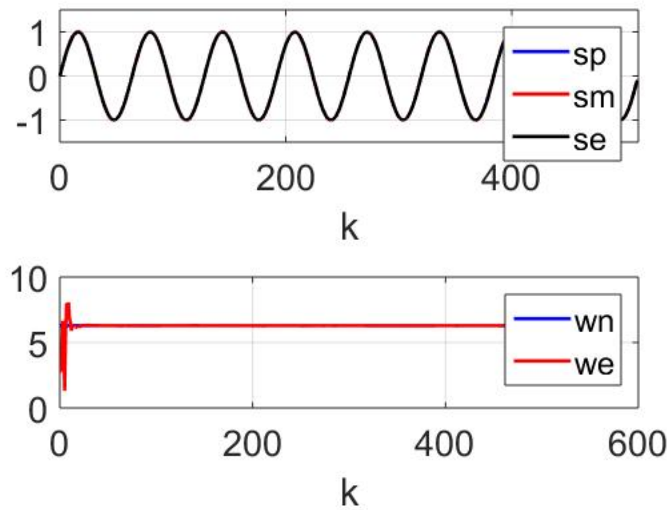

Figura 4: Estimação de uma senóide e de sua frequência angular - Modelo 4

para o Processamento Digital de Sinais, onde o filtro de Kalman pode rastrear as coordenadas de uma face detectada pelo algoritmo de Viola Jones. O "tracking"do objeto detectado, quando aplicado um filtro de Kalman reduz a taxa de falsos positios no sistema. [2] [1]

Referências

[1] Bikash Lamsal and Naofumi Matsumoto. Effects of the unscented kalman filter process for high performance face detector. *International Journal of Information and Electronics Engineering*, 5(6):454, 2015.

[2] Macharla Vinaykumar and Ravi Kumar Jatoth. Performance evaluation of alpha-beta and kalman filter for object tracking. In *Advanced Communication Control and Computing Technologies (ICACCCT), 2014 International Conference on*, pages 1369–1373. IEEE, 2014.

[3] Wikipedia. Filtro de kalman – wikipédia. `https://pt.wikipedia.org/wiki/Filtro_de_Kalman`, 2016.

[4] Wikipedia. Kalman filter – wikipédia. `https://en.wikipedia.org/wiki/Kalman_filter`, 2016.

[5] Paul Zarchan. *Progress In Astronautics and Aeronautics: Fundamentals of Kalman Filtering: A Practical Approach*, volume 208. Aiaa, 2005.

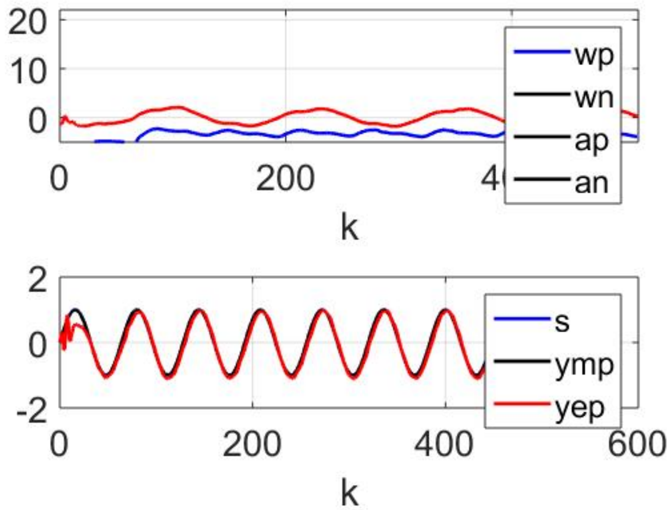

Figura 5: Estimação de uma senóide e de sua frequência angular - Modelo 1

7 Anexos

7.1 Código do Modelo 1

```
clear all
close all
clc

rng(1);
%
an=1;
wn=2*pi;
wa=2^(6)*wn;
h=wn/wa;
%
% x1=0;
% x2=1.25*an;
% x3=1.25*wn;

x1=0;
x2=1.25*wn;
x3=1.25*an;

x=[x1;x2;x3];
```

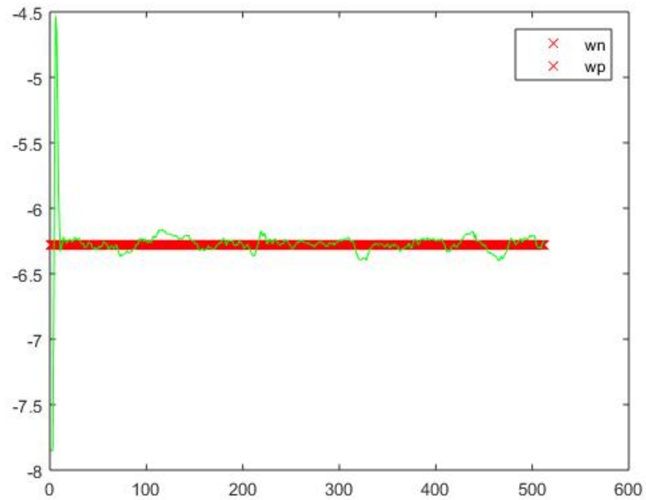

Figura 6: Estimação de uma senóide e de sua frequência angular - Modelo 2

```
%
N=2^(9)-1;
ks=zeros(N+1,1);
sm=zeros(N+1,1);
sp=zeros(N+1,1);
se=zeros(N+1,1);
we=zeros(N+1,1);
%
I3=eye(3,3);
P=1e6*I3;
sigv=1e-2;
R=sigv^2;
sigw=1e-2;
siga=1e-2;
%
for i=0:N,
    ks(i+1)=i;
    %
    s=an*sin(h*wn*i);
    sp(i+1) = s;

    w = x2;
    wp(i+1) = w;
    a = x3;
```

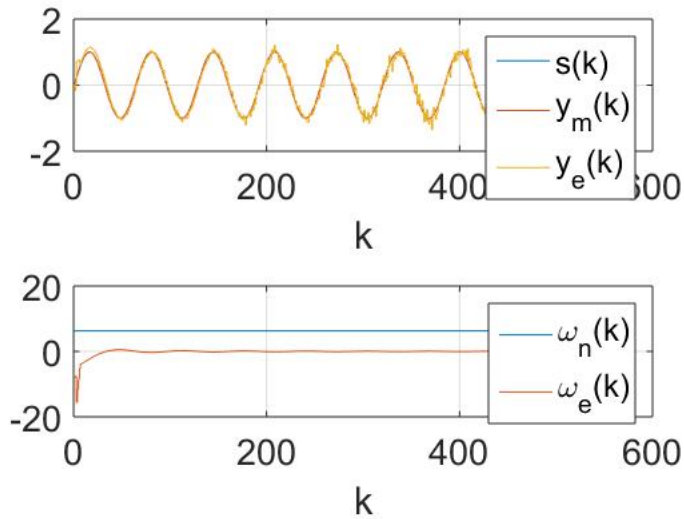

Figura 7: Estimação de uma senóide e de sua frequência angular - Modelo 3

```
ap(i+1) = a;
ye = x3*sin(x1);
yep(i+1) = ye;
ym = s + sigv*randn();
ymp(i+1) = ym;

x1=x(1);x2=x(2);x3=x(3);

F=[1,        h,          0
   0,        1,          0
   0,        0,          1];
% Q
Q=[0,   0,        0
   0,   sigw^2,  0
   0,   0,        siga^2];
% C
C=[x3*cos(x1),  0,  sin(x1)];
%
P=F*P*F'+Q;

K=P*C'*inv(C*P*C'+R);
% f(x)
x1=x1+h*x2;
x2=x2+sigw*randn();
```

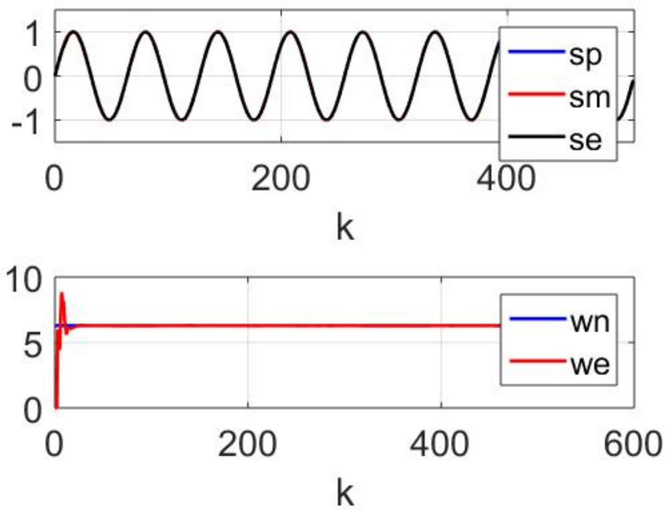

Figura 8: Estimação de uma senóide e de sua frequência angular - Modelo 4

```
    x3=x3+siga*randn();
    %
    x=[x1;x2;x3];

    x=x+K*(ym-ye);
    P=P-K*C*P;
end
%
figure(1)
subplot(2,1,1)
plot(ks',wp,'b',ks',wn,'k',ks',an,'k--',ks',ap,'r','
    Linewidth',2)
set(gca,'FontSize',20)
xlabel('k')
grid
axis([0 length(ks) -5 22])
legend('wp','wn','ap','an')
%
subplot(2,1,2)
plot(ks',sp,'b',ks',ymp,'k',ks',yep,'r','Linewidth',2)
set(gca,'FontSize',20)
xlabel('k')
grid
legend('s','ymp','yep')
```

7.2 Código do Modelo 2

```
clear all
close all
clc

rng(1);
%
an=1;
wn=2*pi;
wa=2^(6)*wn;
h=wn/wa;

%
x1=0;
x2=1.25*wn;

x=[x1;x2];

%
N=2^(9)-1;
ks=zeros(N+1,1);
sm=zeros(N+1,1);
sp=zeros(N+1,1);
se=zeros(N+1,1);
we=zeros(N+1,1);

%
I2=eye(2,2);
P=1e6*I2;
sigv=1e-2;
R=sigv^2;
sigw=1e-2;

%
for i=0:N,
    ks(i+1)=i;
    %
    s=an*sin(h*wn*i);
    sp(i+1) = s;

    w = x2;
    wp(i+1) = w;

    ye = an*sin(x1);
    yep(i+1) = ye;
```

```
    ym = s + sigv*randn();
    ymp(i+1) = ym;

    x1=x(1);
    x2=x(2);

    F=[1,   h
       0,   1];

    % Q
    Q=[0,        0
       0,   sigw^2];

   % C
    C=[an*cos(x1),  0];

    %
    P=F*P*F'+Q;

    K=P*C'*inv(C*P*C'+R);

    % f(x)
    x1=x1+h*x2;
    x2=x2+sigw*randn();

    %
    x=[x1;x2];

    x=x+K*(ym-ye);
    P=P-K*C*P;
end
%
figure(1)
subplot(2,1,1)
plot(ks',wp,'b',ks',wn,'r','Linewidth',2)
set(gca,'FontSize',20)
xlabel('k')
grid
axis([0 length(ks) 0 10])
legend('wp','wn')
%
subplot(2,1,2)
plot(ks',sp,'b',ks',ymp,'k',ks',yep,'r','Linewidth',2)
set(gca,'FontSize',20)
xlabel('k')
```

```
grid
axis([0 length(ks) -1.5 1.5])
legend('s','ymp','yep')
```

7.3 Código do Modelo 3

```
clear all
close all
rng(1);
%
an=1;
wn=2*pi;
wa=2^(6)*wn;
h=wn/wa;
%
x1=0;
x2=1.25*an;
x3=1.25*wn;
x=[x1;x2;x3];
%
N=2^(9)-1;
ks=zeros(N+1,1);
sm=zeros(N+1,1);
sp=zeros(N+1,1);
se=zeros(N+1,1);
we=zeros(N+1,1);
%
I3=eye(3,3);
P=1e6*I3;
sigv=1e-2;
R=sigv^2;
sigw=1e-2;
%
for i=0:N,
    ks(i+1)=i;
    %
    s=an*sin(h*wn*i);
    sp(i+1)=s;
    ym=s+sigv*randn();
    sm(i+1)=ym;
    %
    x1=x(1);x2=x(2);x3=x(3);
    ye=x3*x1;
    se(i+1)=ye;
    we(i+1)=x3;
```

```
    % F
    F=[1,          h,           0
       -h*x3^2,  1,       -2*h*x1*x3
       0,          0,           1];
    % Q
    Q=[0,0,0
       0,0,0
       0,0,sigw^2];
    % C
    %C=[x3,0,x1];
     %C=[1,0,0];
     C=[x3,0,x1];
    %
    P=F*P*F'+Q;
    K=P*C'*inv(C*P*C'+R);
    % f(x)
    x1=x1+h*x2;
    x2=x2-h*x1*x3^2;
    x3=x3+sigw*randn();
    %
    x=[x1;x2;x3];
    x=x+K*(ym-ye);
    P=P-K*C*P;
end
%
figure(1)
subplot(2,1,1)
stairs([ks ks ks],[sp sm se])
set(gca,'FontSize',20)
set(gca,'defaultLineLineWidth',2)
xlabel('k')
grid
legend('s(k)','y_{m}(k)','y_{e}(k)')
%
subplot(2,1,2)
stairs([ks ks],[wn*ones(size(we)) we])
set(gca,'FontSize',20)
set(gca,'defaultLineLineWidth',2)
xlabel('k')
grid
legend('\omega_{n}(k)','\omega_{e}(k)')
```

7.4 Código do Modelo 4

```
clear all
close all
rng(1);
%
an=1;
wn=2*pi;
wa=2^(6)*wn;
h=wn/wa;
%
x1 = 0;
x2 = an*wn;
x3 = 1.25*wn;
x  = [x1;x2;x3];

%
N=2^(9)-1;
ks=zeros(N+1,1);
sm=zeros(N+1,1);
sp=zeros(N+1,1);
se=zeros(N+1,1);
we=zeros(N+1,1);

%
I3=eye(3,3);
P=1e6*I3;
sigv=1e-2;
R=sigv^2;
sigw=1e-2;
%
for i=0:N,
    ks(i+1)=i;
    %
    s=an*sin(h*wn*i);
    sp(i+1)=s;

    ym=s+sigv*randn();
    sm(i+1)=ym;

    x1=x(1);
    x2=x(2);
    x3=x(3);

    ye=x1;
    se(i+1)=ye;

    z = x3;
```

```
    zp(i+1) = z;

    % F
    F=[1,        h,            0
      -z*h,      1,         -x1*h
        0,       0,              1];

   % Q
    Q=[0,0,0
        0,0,0
        0,0,sigw^2];


    %C=[sqrt(x3),0,x1];
    C=[1 0 0];
    %
    P=F*P*F'+Q;
    K=P*C'*inv(C*P*C'+R);

    % f(x)
    x1=x1+h*x2;
    x2=x2-h*x3*x1;
    x3=x3+sigw*randn();


    x=[x1;x2;x3];
    x=x+K*(ym-ye);
    P=P-K*C*P;
end


figure(1)
subplot(2,1,1)
plot(ks',sp,'b',ks',sm,'r',ks',se,'k','Linewidth',2)
set(gca,'FontSize',20)
xlabel('k')
grid
axis([0 length(ks) -1.5 1.5])
legend('sp','sm','se')
%
subplot(2,1,2)
plot(ks,wn*ones(size(we)),'b',ks',zp.^0.5, 'r','Linewidth
    ',2)
set(gca,'FontSize',20)
xlabel('k')
grid
```

```
legend('wn','we')



%%
% figure(1)
% subplot(2,1,1)
% stairs([ks ks ks],[sp sm se])
% set(gca,'FontSize',20)
% set(gca,'defaultLineLineWidth',2)
% xlabel('k')
% grid
% legend('s(k)','y_{m}(k)','y_{e}(k)')
%%
% subplot(2,1,2)
% stairs([ks ],[wn*ones(size(we))])
% set(gca,'FontSize',20)
% set(gca,'defaultLineLineWidth',2)
% xlabel('k')
% grid
% legend('\omega_{n}(k)','\omega_{e}(k)')
```