UNIVERSIDADE ESTADUAL DE CAMPINAS

FACULDADE DE ENGENHARIA ELÉTRICA E DE COMPUTAÇÃO DEPARTAMENTO DE SISTEMAS E ENERGIA



# IA–856 Exercício Computacional

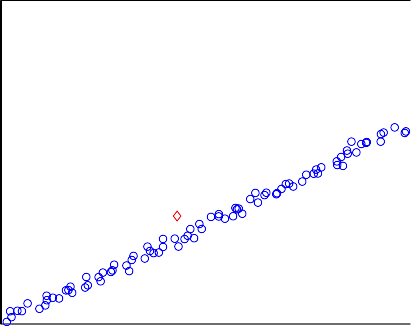
# Filtro de Kalman

# Giovani Chemello Caprio

**O problema de rastreamento**

Suponha que um dispositivo (como um radar) meça as duas posições coordenadas de objetos (avião, navio, etc) realizando movimentos dentro do alcance deste sensor. A Fig [1](#_bookmark0) apresenta três movimentos observados através de medidas sucessivas a cada 4s., tomando-se a posição do radar como referência.

x 104

6

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  | radar |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

5

4

3

2

1

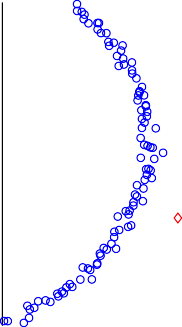
0

−1

−2

x 104

6



radar

5

4

3

2

1

0

−1

−2

−3

−3 −2 −1 0 1 2 3 4

x 104

1. Movimento retilíneo

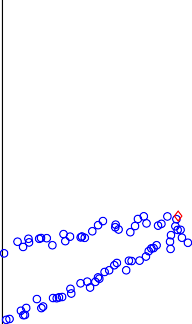
−3

−3 −2 −1 0 1 2 3 4

x 104

1. Curva suave

x 104

6

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  | radar |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

5

4

3

2

1

0

−1

−2

−3

−3 −2 −1 0 1 2 3 4

x 104

1. Manobra

Figura 1: Trajetórias observadas – escala em metros.

As trajetórias observadas não tem precisão suficiente para acompanhamento destes movimentos para, por exemplo, emitir alerta de colisões, detetar ataques em situação de guerra, etc. Além disso, é desejável estimar a velocidade do objeto móvel. Para este fim iremos utilizar um filtro de Kalman e adotaremos um modelo cinemático simples para a descrição dos movimentos. Identificam-se quatro variáveis de estados:

*x*1 : posição do objeto na abcissa

*x*2 : componente da velocidade do objeto na direção da abcissa

*x*3 : posição do objeto na ordenada

*x*4 : componente da velocidade do objeto na direção da abcissa

O modelo cinemático, conhecido como modelo de velocidade (quase) constante em cada uma das componentes ortogonais é descrito por:

*xi*(*k* + 1) = 1 *dT*  *x*1(*k*) + 0

*xi*+1(*k* + 1)

para *i* = 1*,* 2. Aqui temos que:

0 1 *x*2(*k*)

*εi*(*k*)

* *dT* = 4s é a taxa de amostragem/intervalo de tempo entre as medidas;
* *εi*(*k*) *i* = 1 ou *i* = 2 representa a alteração imposta à velocidade do móvel durante o intervalo de amostragem na direção coordenda correspondente. Como esta ação é desconhecida, deve-se modelar a aceleração imposta pelo navegador como uma amostra de ruído gaussiano a cada instante, com média nula e certa variância fixa, adotada *a priori*. Esta variância a princípio é desconhecida, e terá que ser estimada através de tentativas de ajustes do filtro de Kalman, variando-se este dado no modelo utilizado até se encontrar um valor representativo.

São disponíveis a cada instante medidas com ruído das posições cartesianas na forma seguinte:

*z*1(*k*) = *x*1(*k*) + *ν*1(*k*)

*z*2(*k*) = *x*3(*k*) + *ν*2(*k*)*, k* = 0*,* 1*, . . .*

o número total de amostras *k* depende da permanência do objeto no alcance do sensor. Aqui *νj*(*k*)*, j* = 1*,* 2 representam os erros cometido pelo sensor, modelados por ruído

gaussiano com média nula e variância conhecida, dada por *σ*2 2 = 12002, sendo

*ν*1

= *σ*

*ν*2

não-correlacionados.

As seqüências de dados de medidas para as três trajetórias estão disponíveis nos arquivos reta.mat e curva\_suave.mat e manobra.mat (variável x\_m), além dos dados exatos de posição dos objetos (variável x\_c) disponíveis para efeito de comparação e avaliações de desempenho do filtro para cada trajetória.

# **Questões**

1. Implemente o filtro de Kalman para estimar a posição no plano e a velocidade do móvel, assumindo-se que *ε*(*k*) e *ν*(*k*) são processos estocásticos independentes brancos com média nula, com matriz de covariância Σ*ε* = *σ*2*I* e Σ*ν* = 12002*I*

*ε*

sendo *σ*2 desconhecido.

*ε*

Utilize os dados nos três arquivos e estabeleça condições iniciais apropriadas para o estado do filtro e para a covariância inicial, lembrando que ela deve ser suficientemente grande para representar o desconhecimento inicial que temos sobre a velocidade e o valor correto de posição.

Faça experimentos com as três trajetórias e verifique a adequação de ajustes do valor de *σ*2 para cada caso, através do erro que o filtro comente (vide os itens 2. e 4. em seguida, para adotar uma forma sistemática de avaliar o erro). Adote em seguida um único valor para *σ*2 que deverá ser mantido em todos experimentos

*ε*

*ε*

subseqüentes.

1. Estime o erro quadrático médio existente entre a posição real (coordenadas (*x, y*)) e os valores estimados, a partir dos dados nos arquivos e o resultado do filtro. Verifique a compatibilidade dos erros observados com o erro médio quadrático de estimação de estados de posição teóricos do filtro, dados por ele- mentos correspondentes da matriz de covariância do filtro de Kalman *Px*(*k*|*k*). Verifique a compatibilidade destes valores com a variância dos ruídos do sistema

*σ*2 e de observação *σ*2.

*ε ν*

1. De forma similar, estime a covariância da saída (erro da medida, avaliada através da inovação) e compare com a covariância de erro de medida teórica do filtro de Kalman, dada pela matriz *Pz*(*k*|*k*). Em seguida plote a verossimilhança das medidas ao longo do intervalo de observação através da distribuição condicional do erro de previsão da observação, dada por

*pk*|*k*−1(*z*(*k*)|*x*ˆ(*k*|*k* − 1)*, Px*(*k*|*k* − 1)) =

1

1

.|2*πPz*(*k*|*k* − 1)|

exp .−2

(*z*(*k*) − *z*(*k*|*k* − 1))*Pz*(*k*|*k* − 1)−1(*z*(*k*) − *z*(*k*|*k* − 1))′

Aqui, *z*(*k*|*k* − 1) é a previsão da medida para o instante *k* fornecida pelo filtro no passo anterior; *Pz*(*k*|*k* − 1) covariância da medida no instante *k*, prevista no instante anterior e | · | representa o determinante.

1. Analise a influência no comportamento do filtro na evolução do estado estimado, alterando os seguintes aspectos:
   * a inicialização do filtro,
   * as variâncias dos ruídos estado e da medida.

Comente os resultados. Sugestão: aumente e diminue a variância da ordem de uma grandeza.

1. Determine pela teoria se a estabilidade do filtro estacionário é garantida. Ve- rifique pela implementação do filtro se ele atinge uma solução estacionária. Se positivo, verifique também
   * se a solução estacionária depende da condição inicial utilizada, a matriz

*Px*(0).

* + se a solução encontrada de fato torna o filtro estacionário estável.

1. Verifique através do acompanhamento de erros se o filtro têm deficiências em curvas fechadas, e avalie a razão deste comportamento, explicando o observado.
2. Utilizando a informação sobre o erro previsto, proponha uma estratégia de adap- tação do ganho do filtro de Kalman, para que ele rastreie melhor não estaciona- riedades do estado. Faça experimentos numéricos com a modificação proposta, e compare com a solução do item anterior.

# **1º Experimento**

# Primeiramente, foi realizado um experimento gerando apenas uma filtragem para os eixos x e y, mas não houve uma razoável filtragem para o caso de manobra. Assim, foi decidido a filtragem em x e a filtragem em y ,separadamente, gerando resultados satisfatórios e muito melhores em relação a primeira tentativa.

# Com a realização do experimento de implementação do filtro de Kalman, para as observações geradas pelo radar. Obtivemos o seguinte resultado:

# Manobra

# Reta

# Curva Suave

# 

# Claramente, se percebeu que a reta teve uma resposta satisfatória, a curva uma boa resposta e no caso da manobra uma resposta ruim.

# Para obter a qualidade dessa aproximação da estimação de uma forma númerica, foi calculado o erro quadrático médio entre a curva real e estimado. Alem disso, obteus-se a velocidade média e sua variância para cada observação.

# Reta

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Erro em x | Erro em y | Erro Médio Quadrático | Velocidade Média | Variância(Velocidade) |
| 41.3895 | 50.8542 | 92.2437 | 492.3853 | 5.8312e+03 |

# Curva Suave

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Erro em x | Erro em y | Erro Médio Quadrático | Velocidade Média | Variância(Velocidade) |
| 39.5105 | 31.3760 | 70.8864 | 272.3832 | 1.9076e+04 |

# Manobra

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Erro em x | Erro em y | Erro Médio Quadrático | Velocidade Média | Variância(Velocidade) |
| 160.6278 | 89.7769 | 250.4046 | 500.31 | 1.5182e+04 |

# Portanto, percebe-se que o erro aumenta no caso de manobras brutas. Isto ocorre devido a matriz de covariância do erro(P) tentar acompanhar o estimado, mas não prever as mudanças bruscas.

# A fim de testes, foram mudados valores das matrizes Q e R, a variação de Q levou a estimação errada, devido ser do próprio sistema. Em mudanças de R, houve piora e melhora, menos significativas.

# Por fim, foi feito o gráfico de Covariância de erros da posição, percebe-se que todos tenderam a um valor. Como pode-se ver a seguir:

# Reta

# 

# Curva Suave

# 

# Manobra

# 

# A controlabilidade e observabilidade do sistema foram verdadeiras, assim conseguimos calcular o P previsto por Ricatti, e vimos que isso bateu com a tendência de P1 e P2 em cada caso. Assim, uma variação de P apenas faz com que o filtro se torne bom com mais iterações, mas tende a se estabilizar.

# Na tentativa de mudar parametros iniciais para verificar as mudanças no filtro. O código original obtinha a posição e velocidade em relação as observações. Em uma mudança, para zero em todos os quesitos, obtivemos:

# 

# 

# Na mudança do sigma na matriz R , de 1200 para 3000, houve uma piora das medições na manobra e curva. Isto faz sentido,pois devido a uma precisão pior do radar, teremos uma maior dificuldade de filtragem. Em uma mudança de 1200 para 100, as observações ficaram mais proximas da estimação. Portanto, sua mudança drástica foi piorada para mais e para menos.

# Na mudança de Q, a sua mudança obteve uma estimação melhorar em caso de dimunuição e piora em caso de aumento, como podemos ver a seguir:

# 

# Obviamente, o filtro apresentou dificuldades em relação a curva e manobras bruscas. Isto ocorre, pois é um filtro linear trabalhando com curvas, fugindo da trajetória real. Para a melhora, a adição da aceleração pode tornar a curva mais suave, como tambem uma precisão maior no sistema e no sensor.

# 

# **1º Parte do Código**

clear all ,clc,close all

%% Inicialização

%Dados(escolhendo um dos 3)

load('reta.mat')

load('curva\_suave.mat')

load('manobra.mat')

dT = 4; %Intervalo entre amostras

t = length(x\_m(1,:)); %números de amostras de observação

xc = x\_c'; %leitura curva real

% Planta do sistema

F = [ 1 dT; 0 1 ];

G = [ dT;(1/2)\*(dT)^2];

H = [1 0];

Q = 1000^2\*eye(2);%Erro do sistema

cov\_v = 1200; %sigma

R = cov\_v^2\*eye(2);%Erro no sensor

%% Filtro de Kalman

% Inicialização

P1 = [4\*10^5 10^4;10^4 4\*10^5]; % Covariância inicial

x = [x\_m(1,1);(x\_m(1,2)-x\_m(1,1))/dT]; % Estado inicial[pos x, velo em x]

xe = zeros(4, t);

errcov = zeros(4, 2\*t);

ycov = zeros(t,1);

ver = zeros(4,2\*t);

xx = zeros(2,t);

Erro\_x=0;Erro\_y=0;

for i=1:t

% Measurement update

Mn = P1\*H'/(H\*P1\*H'+H\*R\*H');

xx(1,i) = x\_m(1,i)-H\*x;

x = x + Mn\*(xx(1,i)); % x[n|n]

P1 = (eye(2)-Mn\*H)\*P1;

xe(1:2,i) = x;

errcov(1:2, (2\*i-1):2\*i) = P1;

% Time update

Erro\_x=Erro\_x+(x(1,1)-xc(1,i)).^2;

x = F\*x; % x[n+1|n]

P1 = F\*P1\*F' + Q;

ver(1:2, (2\*i-1):2\*i) = 1/sqrt(det(2\*pi\*P1))\*exp( -0.5\*xx(1,i)\*inv(P1)\*xx(1,i)' );

end

P2 = [4\*10^5 10^4;10^4 4\*10^5];

%x = [x\_m(2,1); 0]; % Condição inicial para o estado

x = [x\_m(2,1);(x\_m(2,2)-x\_m(2,1))/dT];%Estado inicial[pos y, velo em y]

for i=1:t

% Measurement update

Mn = P2\*H'/(H\*P2\*H'+H\*R\*H');

xx(2,i) = x\_m(2,i)-H\*x;

x = x + Mn\*(xx(2,i)); % x[n|n]

P2 = (eye(2)-Mn\*H)\*P2;

xe(3:4,i) = x;

errcov(3:4, (2\*i-1):2\*i) = P2;

Erro\_y=Erro\_y+(x(1,1)-xc(2,i)).^2;

% Time update

x = F\*x; % x[n+1|n]

P2 = F\*P2\*F' + Q;

ver(3:4, (2\*i-1):2\*i) = 1/sqrt(det(2\*pi\*P2))\*exp( -0.5\*xx(2,i)\*inv(P2)\*xx(2,i)' );

end

%Calculos erro médio

Erro\_x = sqrt(Erro\_x)/t

Erro\_y = sqrt(Erro\_y)/t

Erro\_MQ = (Erro\_x + Erro\_y) %Erro Médio Quadrático

%Validação Filtragem

plot(x\_m(1,:),x\_m(2,:),'-.'),hold on %plota observações

plot(xe(1,:),xe(3,:),'^k'),hold on %plota estimação

plot(xc(1,:),xc(2,:)) %plota real

plot(0,0,'or'); %Posicionamento do radar em coordenadas

legend('Observado','Estimado','Real','Radar')

xlabel('Coordenada X(m)'), ylabel('Coordenada Y(m)')

%erro

figure, plot(xx')

%Covariância do erro

figure, plot(errcov(1,1:2:end),'^k')

hold on, plot(errcov(3,1:2:end),'-.')

plot(errcov(1,2:2:end),'b')

plot(errcov(2,2:2:end),'\*')

plot(sqrt(errcov(1,1:2:end).^2+errcov(3,1:2:end).^2),'or')

title('Covariância do erro')

legend('Covariância em x','Covariância em y', 'Covariância entre posição e velocidade','Covariância de velocidade')

%% Controlabilidade e ricatti

%Controlabilidade de F e C onde Q = C'C

M = ctrb(F, chol(Q));

rank(M)

%Observabilidade

M = obsv(F,H)

rank(M)

%Ricattti

Pf = F\*(P1-P1\*H'\*inv(H\*P1\*H'+H\*R\*H')\*H\*P1)\*F' + Q

# **2º Experimento**

# Com o primeiro experimento, foi sugerido que o filtro fosse feito com a junção do eixo x e eixo y. Isso porque com a junção dos eixos conseguimos uma maior precisão nos ganhos de Kalman e na matriz de Covariância (P), gerando estimações melhores para o filtro de Kalman.

# No primeiro experimento, realizou-se duas filtragens, uma para o eixo x e uma para o y. Neste, a planta foi gerada em 2D e as matrizes de covariância foram geradas para obter a interferências entre eixos, sendo uma matriz 4x4.

# Os resultados obtidos foram:

# Manobra

# 

# Curva Suave

# 

# Reta

# 

# 

|  |  |  |
| --- | --- | --- |
| Errro Médio Quadrático | Velocidade Média | Variância Média |
| 3.1423e+03 | 2.3170e+03 | 7.4888e+05 |

# Com isso, fica registrado uma melhora significativa em relação a curvas no segundo método. Sendo essa conclusão visual e numérica.

# **2º Parte do Código**

%% Inicialização

%Dados

load('reta.mat')

load('curva\_suave.mat')

load('manobra.mat')

dT = 4; %Intervalo entre amostras

t = length(x\_m(1,:)); %números de amostras de observação

xc = x\_c'; %leitura curva real

% Planta do sistema

F = [ 1 dT; 0 1 ];

G = [ dT;(1/2)\*(dT)^2];

H = [1 0];

Q = 10^2\*eye(2);%Erro do sistema

cov\_v = 1200; %sigma

R = cov\_v^2\*eye(2);%Erro no sensor

%% Filtro de Kalman

% Inicialização

P1 = [4\*10^5 10^4;10^4 4\*10^5]; % Covariância inicial

x = [x\_m(1,1);(x\_m(1,2)-x\_m(1,1))/dT]; % Estado inicial[pos x, velo em x]

xe = zeros(4, t);

errcov = zeros(4, 2\*t);

ycov = zeros(t,1);

ver = zeros(4,2\*t);

xx = zeros(2,t);

Erro\_x=0;Erro\_y=0;

for i=1:t

% Measurement update

Mn = P1\*H'/(H\*P1\*H'+H\*R\*H');

xx(1,i) = x\_m(1,i)-H\*x;

x = x + Mn\*(xx(1,i)); % x[n|n]

P1 = (eye(2)-Mn\*H)\*P1;

xe(1:2,i) = x;

errcov(1:2, (2\*i-1):2\*i) = P1;

% Time update

Erro\_x=Erro\_x+(x(1,1)-xc(1,i)).^2;

x = F\*x; % x[n+1|n]

P1 = F\*P1\*F' + Q;

ver(1:2, (2\*i-1):2\*i) = 1/sqrt(det(2\*pi\*P1))\*exp( -0.5\*xx(1,i)\*inv(P1)\*xx(1,i)' );

end

P2 = [4\*10^5 10^4;10^4 4\*10^5];

%x = [x\_m(2,1); 0]; % Condição inicial para o estado

x = [x\_m(2,1);(x\_m(2,2)-x\_m(2,1))/dT];%Estado inicial[pos y, velo em y]

for i=1:t

% Measurement update

Mn = P2\*H'/(H\*P2\*H'+H\*R\*H');

xx(2,i) = x\_m(2,i)-H\*x;

x = x + Mn\*(xx(2,i)); % x[n|n]

P2 = (eye(2)-Mn\*H)\*P2;

xe(3:4,i) = x;

errcov(3:4, (2\*i-1):2\*i) = P2;

Erro\_y=Erro\_y+(x(1,1)-xc(2,i)).^2;

% Time update

x = F\*x; % x[n+1|n]

P2 = F\*P2\*F' + Q;

ver(3:4, (2\*i-1):2\*i) = 1/sqrt(det(2\*pi\*P2))\*exp( -0.5\*xx(2,i)\*inv(P2)\*xx(2,i)' );

end

%Calculos erro médio

Erro\_x = sqrt(Erro\_x)/t

Erro\_y = sqrt(Erro\_y)/t

Erro\_MQ = (Erro\_x + Erro\_y) %Erro Médio Quadrático

%Validação Filtragem

plot(x\_m(1,:),x\_m(2,:),'-.'),hold on %plota observações

plot(xe(1,:),xe(3,:),'^k'),hold on %plota estimação

plot(xc(1,:),xc(2,:)) %plota real

plot(0,0,'or'); %Posicionamento do radar em coordenadas

legend('Observado','Estimado','Real','Radar')

xlabel('Coordenada X(m)'), ylabel('Coordenada Y(m)')

title('Q=1000')

%erro

figure, plot(xx')

%% Controlabilidade e ricatti

%Controlabilidade de F e C onde Q = C'C

M = ctrb(F, chol(Q));

rank(M)

%Observabilidade

M = obsv(F,H)

rank(M)

%Ricattti

Pf = F\*(P1-P1\*H'\*inv(H\*P1\*H'+H\*R\*H')\*H\*P1)\*F' + Q