Introdução – Filtragem de Kalman

Conceitos básicos

Introdução

Filtro de Kalman:

- Um observador de estados que atenua (filtra) da melhor maneira possível, o efeito de ruídos/incertezas existentes no modelo e sensores

Conceitos importantes:

- Sistemas discretos e discretizados
- Probabilidade e processos estocáticos

Sistemas dinâmicos podem ser contínuos ou discretos. Sistemas contínuos podem ser representados de forma discreta.

O filtro de Kalman pode ser implementado na forma discreta (caso mais comum) ou de forma contínua (Kalman-Bucy)

Introdução

Caso discreto é o mais utilizado:

- Usualmente o FK é executado em um computador/microcontrolador
- Não há vantagem em simular a versão contínua em um computador que executa tarefas discretas

Devido a isso, estudaremos brevemente como discretizar um sistema contínuo.

Como envolve o efeito de variáveis aleatórias, estudaremos brevemente probabilidade e processos estocásticos

Discretização

Sistema contínuo, linear e invariante no tempo:

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t), \qquad \boldsymbol{y}(t) = \boldsymbol{C}x(t)$$

Sistema discreto, linear e invariante no tempo:

$$\boldsymbol{x}_{k+1} = \boldsymbol{F}\boldsymbol{x}_k + \boldsymbol{G}\boldsymbol{u}_k, \qquad \boldsymbol{y}(t) = \boldsymbol{H}\boldsymbol{x}_k$$

Em que $\boldsymbol{x}_k = \boldsymbol{x}(t)$, $\boldsymbol{x}_{k+1} = \boldsymbol{x}(t+T)$, e T é a taxa de amostragem

Exemplo:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

O modelo é chamado de duplo integrador, pois $y = \int\int u dt$. O modelo representa, por exemplo, um carro ou foguete andando/voando em linha reta.

u é uma entrada aceleração, x_2 é a velocidade e x_1 a posição

Como discretizar o modelo?

Discretização - exemplo

Viu-se na disciplina que

$$\boldsymbol{x}(t+T) = \boldsymbol{\phi}(T)\boldsymbol{x}(t) + \int_0^T \boldsymbol{\phi}(\tau)\mathbf{B}\mathbf{u}(T+t-\tau)d\tau$$

Então, se $\boldsymbol{u} = \mathbf{0}$:

$$\boldsymbol{F} = \boldsymbol{\phi}(T)$$

No exemplo:

$$A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \rightarrow \boldsymbol{\phi}(T) = \mathcal{L}^{-1}[(s\boldsymbol{I} - \boldsymbol{A})^{-1}] \rightarrow \boldsymbol{F} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}$$

Já $\boldsymbol{G}$ não é tão fácil, pois envolve uma integral de convolução. Entretanto, supondo que um computador fornece um valor constante de $\boldsymbol{u}$ entre dois períodos de amostragem (ou então, que o período de amostragem é tão pequeno que $\boldsymbol{u}$ é aproximadamente constante:

$$\boldsymbol{u}(t) = \boldsymbol{u}_k, \qquad (k-1)T < t < kT$$

Então:

$$\int_0^T \boldsymbol{\phi}(\tau)\mathbf{B}\mathbf{u}(T+t-\tau)d\tau = \int_0^T \boldsymbol{\phi}(\tau)\mathbf{B}d\tau \; \boldsymbol{u}_\mathrm{k} = \boldsymbol{G}\boldsymbol{u}_\mathrm{k}$$

Discretização - exemplo

No exemplo:

$$\boldsymbol{G} = \int_0^T \begin{bmatrix} 1 & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} dt = \begin{bmatrix} \frac{T^2}{2} \\ T \end{bmatrix}$$

Calculando $\boldsymbol{H}$:

Como o cálculo de $\boldsymbol{y}$ é algébrico, não há mudança no cálculo - $\boldsymbol{H} = \boldsymbol{C}$

Assim:

$$\begin{bmatrix} x_{1,k+1} \\ x_{2,k+1} \end{bmatrix} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_{1,k} \\ x_{2,k} \end{bmatrix} + \begin{bmatrix} T^2/2 \\ T \end{bmatrix} u_k$$

$$y_k = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_{1,k} \\ x_{2,k} \end{bmatrix}$$

Veja que o sistema discreto obtido é a fórmula do movimento uniformemente acelerado: $s = s_0 + v_0 t + \frac{at^2}{2}, \;\; v = v_0 + at$

Discretização - exemplo

Foi visto que:

$$\boldsymbol{F} = \boldsymbol{\phi}(T) = e^{\boldsymbol{A}T}$$

Considerando um intervalo de amostragem T suficientemente curto, a exponencial acima pode ser aproximada por uma série de Taylor, ignorando os termos de ordem 2 ou maior:

$$\boldsymbol{F} \approx \boldsymbol{I} + \boldsymbol{A}T$$

No exemplo dado:

$$\boldsymbol{F} \approx \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} T = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}$$

Obs: nesse caso a resposta deu exata porque o valor de $\boldsymbol{A}$ fez com que a solução fosse um polinômio, e não uma exponencial

A matriz $\boldsymbol{G}$ também pode ser computada de modo aproximado:

$$\boldsymbol{G} \approx \boldsymbol{B}T$$

No exemplo:

$$\boldsymbol{G} \approx \begin{bmatrix} 0 \\ 1 \end{bmatrix} T = \begin{bmatrix} 0 \\ T \end{bmatrix}$$

Veja que o termo de segunda ordem $T^2/2$ não apareceu nessa aprox.

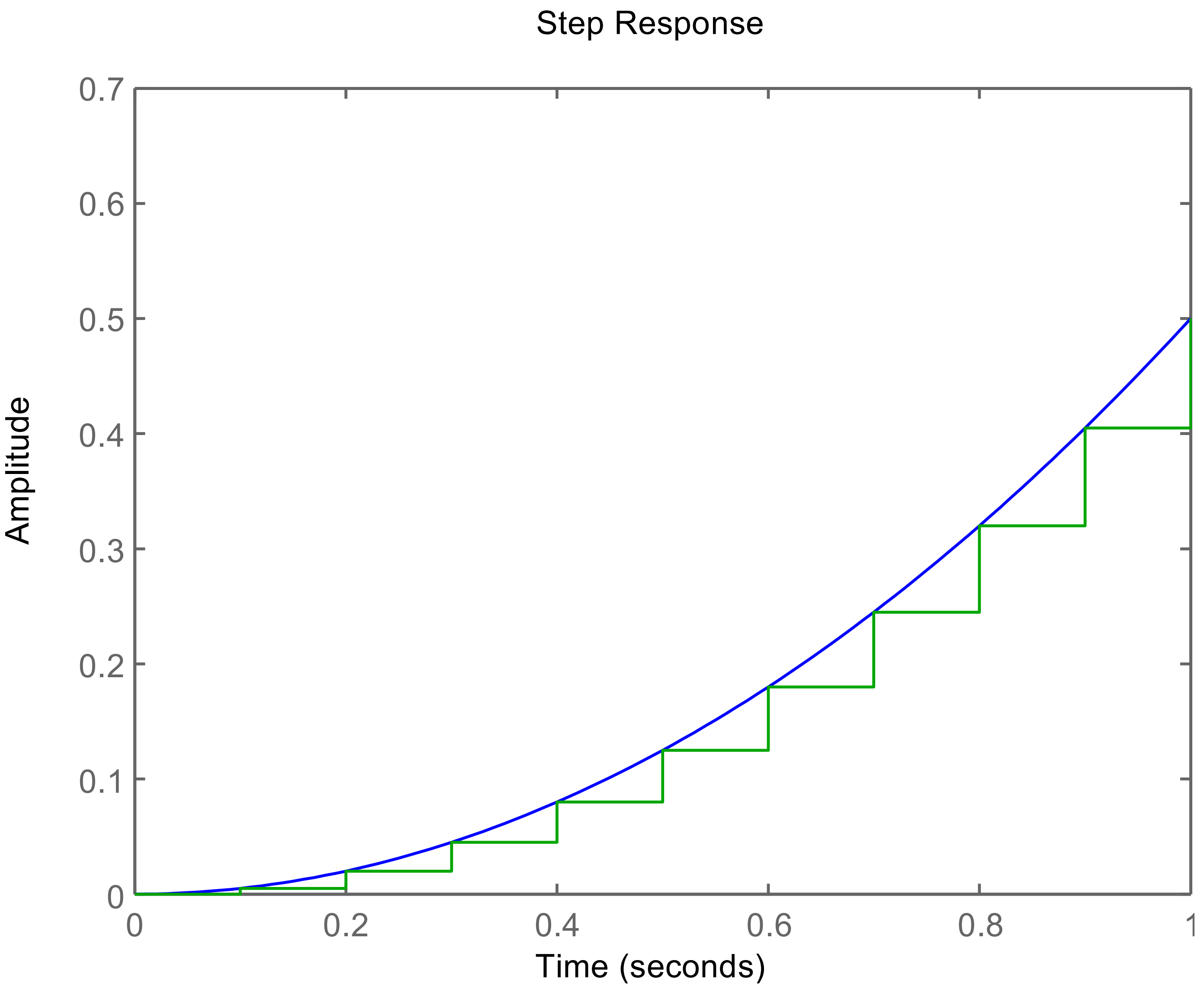
Discretização - exemplo

Usando o MATLAB:

```
>> G_cont = ss(A,B,C,0)
G_cont =
 a =
     x1  x2
  x1  0   1
  x2  0   0
 b =
     u1
  x1  0
  x2  1
 c =
     x1  x2
  y1  1   0
 d =
     u1
  y1  0
Continuous-time state-space model.
```

```
>> G_disc = c2d(G_cont,0.1)
G_disc =
 a =
       x1   x2
  x1    1  0.1
  x2    0    1
 b =
        u1
  x1  0.005
  x2    0.1
 c =
      x1  x2
  y1   1   0
 d =
      u1
  y1   0
Sample time: 0.1 seconds
Discrete-time state-space model.
```

Comparação de respostas

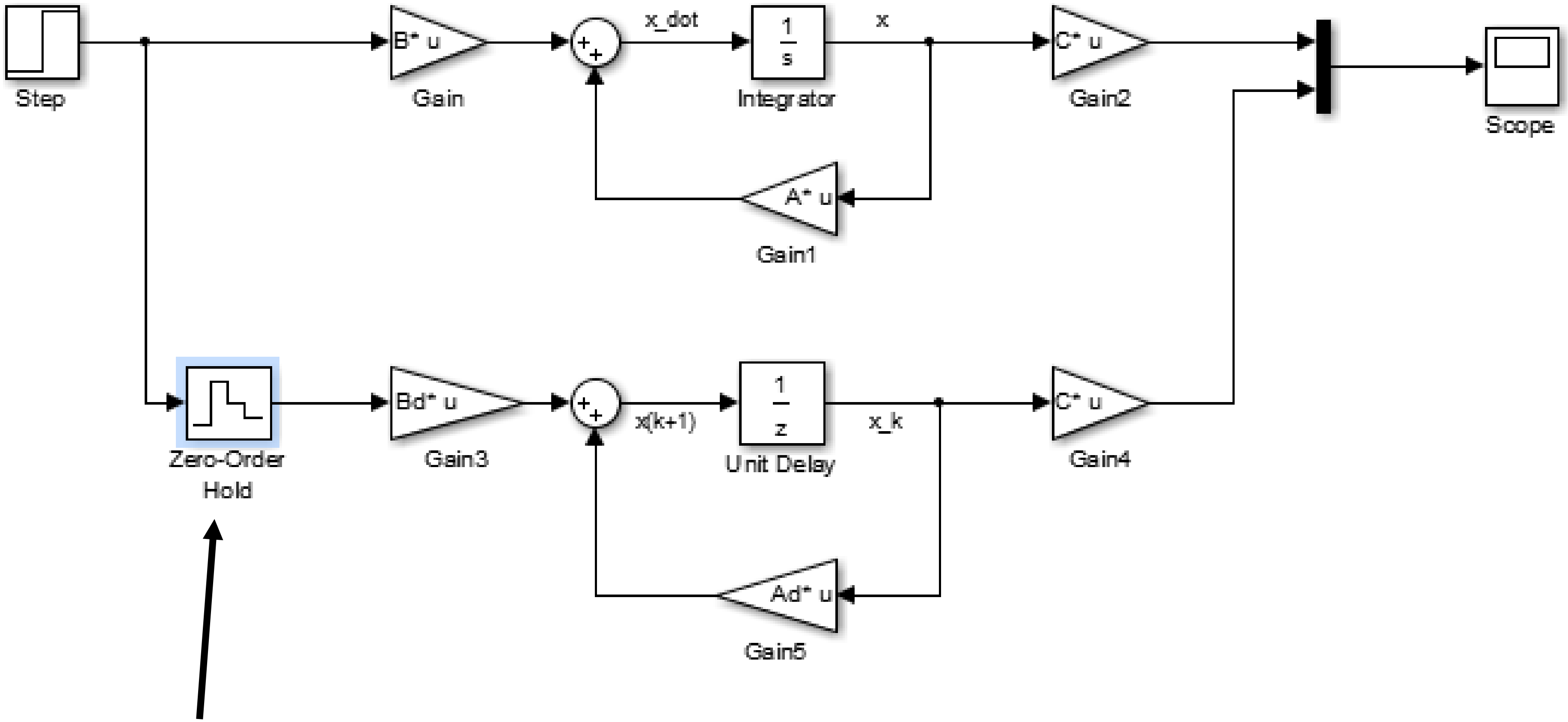



```
>> step(G_cont, 1)
>> hold on
>> step(G_disc, 1)
```

Gerando resposta discreta manualmente

```
T = 0.1;
t = 0:T:1;
x = zeros(2,length(t));
u = 1;
for i = 1:(length(t)-1)
    x(:,i+1) = G_disc.a*x(:,i) + G_disc.b*u;
end
stairs(t,x(1,:))
```

Simulink



Mantém sinal de entrada constante a cada intervalo de tempo

Sistemas discretos – algumas propriedades

Controlabilidade e observabilidade:

Exatamente a mesma definição:

$$\mathcal{C} = [\boldsymbol{G} \quad \boldsymbol{F}\boldsymbol{G} \quad \dots \quad \boldsymbol{F}^{n-1}\boldsymbol{G}], \qquad \mathcal{O} = \begin{bmatrix} \boldsymbol{H} \\ \boldsymbol{H}\boldsymbol{F} \\ \vdots \\ \boldsymbol{H}^{n-1}\boldsymbol{F} \end{bmatrix}$$

Conversão de espaço de estados discreto para transformada Z (equivalente à transformada de Laplace, mas discreta) – mesma fórmula

$$\frac{\mathrm{Y}(z)}{U(z)} = G(z) = \mathbf{H}(z\boldsymbol{I} - \boldsymbol{F})^{-1}\mathbf{G}$$

z^{-1}: atraso de um intervalo de tempo.

Exemplo: usando, no MATLAB, G_tf_disc = tf(G_disc), obtém-se

$$\frac{\mathrm{Y}(z)}{U(z)} = \frac{0.005z + 0.005}{z^2 - 2z + 1} = \frac{0.005z^{-1} + 0.005z^{-2}}{1 - 2z^{-1} + z^{-2}}$$

Sistemas discretos – algumas propriedades

$$Y(z)(1 - 2z^{-1} + z^{-2}) = U(z)(0.005z^{-1} + 0.005z^{-2})$$

$$y_k - 2y_{k-1} + y_{k-2} = 0.005(u_{k-1} + u_{k-2})$$

$$y_k = 0.005(u_{k-1} + u_{k-2}) + 2y_{k-1} - y_{k-2}$$

Veja que saída atual depende apenas de entradas e saídas passadas.
Simulando manualmente no MATLAB:

```
T = 0.1;
u = 1;
y_m2 = 0;
y_m1 = 0;

t = 0:T:1;
y = zeros(1,length(t));
```

```
y(1) = 0;
y(2) = 0.005*u + 2*y(1);

for i = 3:length(t)
    y(i) = 0.01*u + 2*y(i-1) - y(i-2);
end
stairs(t,y);
```

Sistemas discretos – algumas propriedades

Slides a seguir foram desenvolvidos pelo Prof. João Paulo Lustosa, da UnB, para a disciplina de pós AASP.

Destaca-se que foram inseridas apenas páginas específicas da aula de AASP que possuem relevância para a disciplina de Projeto de Sistemas de Controle

Adaptive & Array Signal Processing
AASP

Prof. Dr.-Ing. João Paulo C. Lustosa da Costa
University of Brasília (UnB)
Department of Electrical Engineering (ENE)
Laboratory of Array Signal Processing
PO Box 4386
Zip Code 70.919-970, Brasília - DF

Homepage: http://www.pgea.unb.br/~lasp

Mathematical Background: Z Transform (1)

- Z Transform
  - also seen as the discrete form of the Laplace transform
  - The Laplace transform is defined as

$$\boxed{H(s) = \int_{t=0}^{\infty} h(t)e^{-st}dt}$$

    - ✓ $h(t)$ is continuous.
  - In digital processing devices, samples are used. Therefore, the sampled version of $h(t)$ is given by

$$h[n] = h(n \cdot T_s)$$

    where T_s is **sampling period**.
  - By replacing $h[n]$ in the Laplace transform

$$H(s) = \sum_{n=0}^{\infty} h(n \cdot T_s)e^{-n \cdot s \cdot T_s}$$

Mathematical Background: Z Transform (2)

- Z Transform
  - defining:

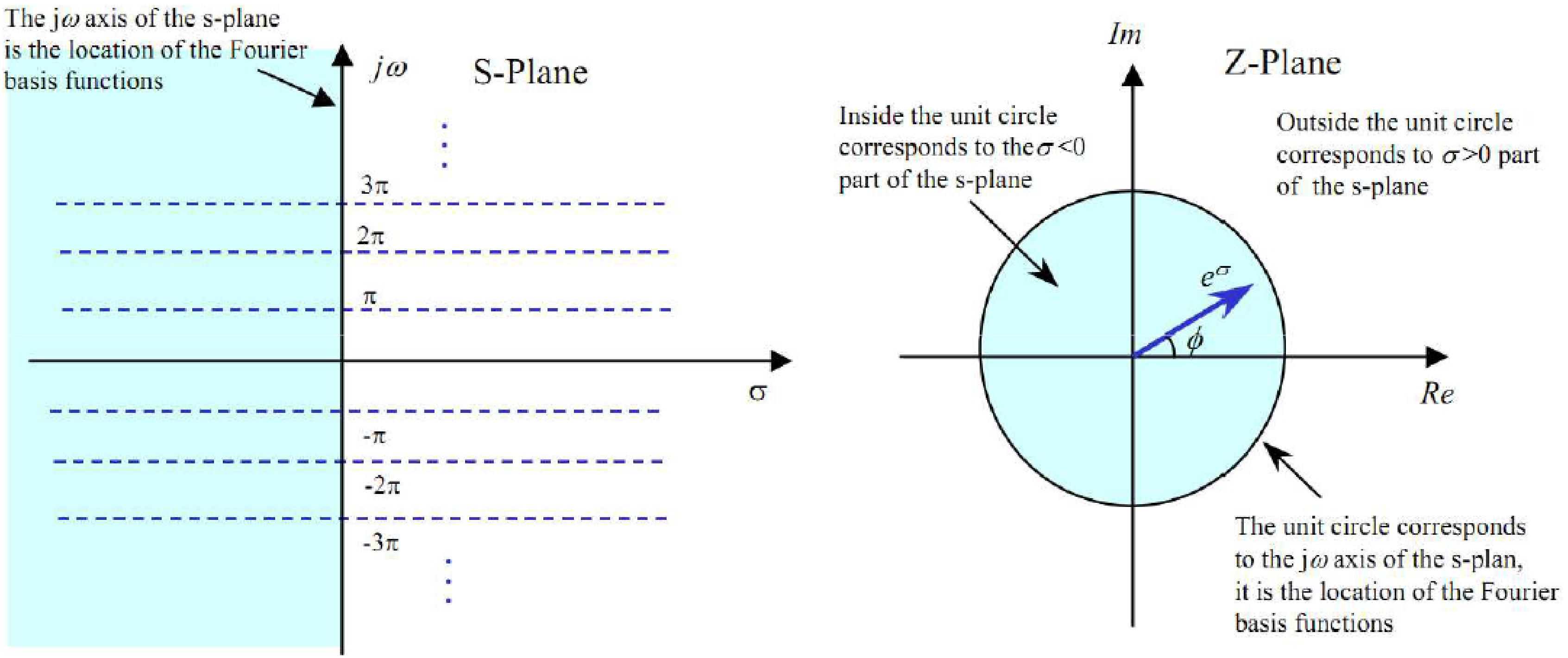
$$\boxed{z = e^{s \cdot T_s}}$$

  - Therefore:

$$H(z) = \sum_{n=0}^{\infty} h(n \cdot T_s) z^{-n}$$

Mathematical Background: Z Transform

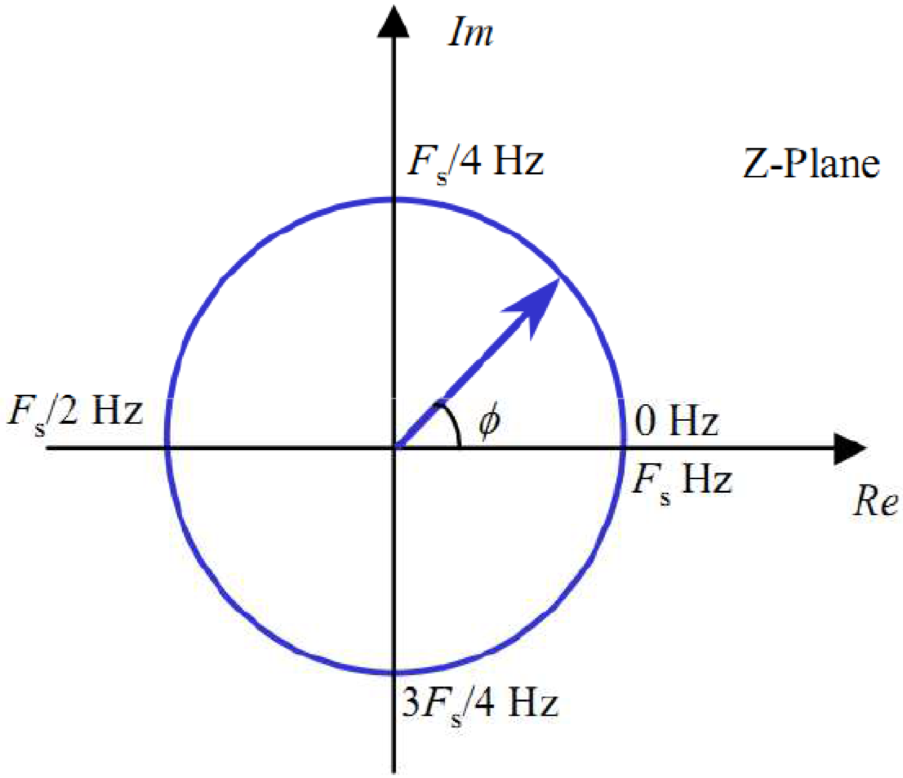
- Z Transform: the **z** plane and the unit circle



Reference: Saeed Vaseghi, Communication and Multimedia Signal Processing group, Brunel University

Mathematical Background: Z Transform (4)

- Z Transform: Frequency to angle mapping



$$\boxed{\phi = \frac{2\pi}{F_s} f}$$

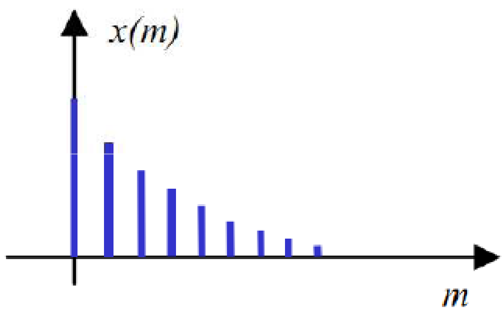
If $F_s = 40$ kHz and $f = 5$ kHz, then $\phi = 0.05\pi$ rad $= 45°$

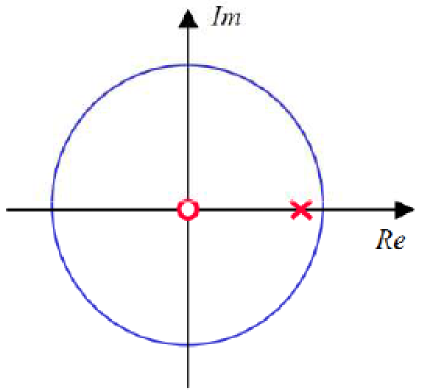
Reference: Saeed Vaseghi, Communication and Multimedia Signal Processing group, Brunel University

Mathematical Background: Z Transform

- Z Transform: Example 1 – Find the **z** transform and pole-zero diagram

$$x(m) = \begin{cases} \alpha^m & m \geq 0 \\ 0 & m < 0 \end{cases}$$





$$X(z) = \sum_{m=0}^{\infty} \alpha^m z^{-m} = \sum_{m=0}^{\infty} \left(\alpha z^{-1}\right)^m$$

$$\boxed{X(z) = \frac{1}{1 - \alpha z^{-1}} = \frac{z}{z - \alpha}}$$

Sistemas discretos – algumas propriedades

Da mesma forma que os sistemas contínuos, nos sistemas discretos há a equivalência entre autovalor e polos do sistema.

Porém, em sistemas discretos, os autovalores/polos devem estar situados dentro de um círculo unitário para sistema ser estável.

Exemplo:

```
>> eig(A)
ans =
    0
    0
```

```
>> eig(G_disc.a)
ans =
    1
    1
```

Frequência de oscilação relacionada ao ângulo (maior ângulo, maior a frequência)

- O círculo unitário é capaz de representar o efeito de aliasing de amostragem.

Velocidade de convergência relacionado à distância: mais rápido quanto mais perto do centro

Processos estocásticos

Slide Fundamentos de probabilidade e estatística do site:

http://www2.ene.unb.br/gaborges/disciplinas/efe/index.htm