Department of Systems Engineering

George Mason University

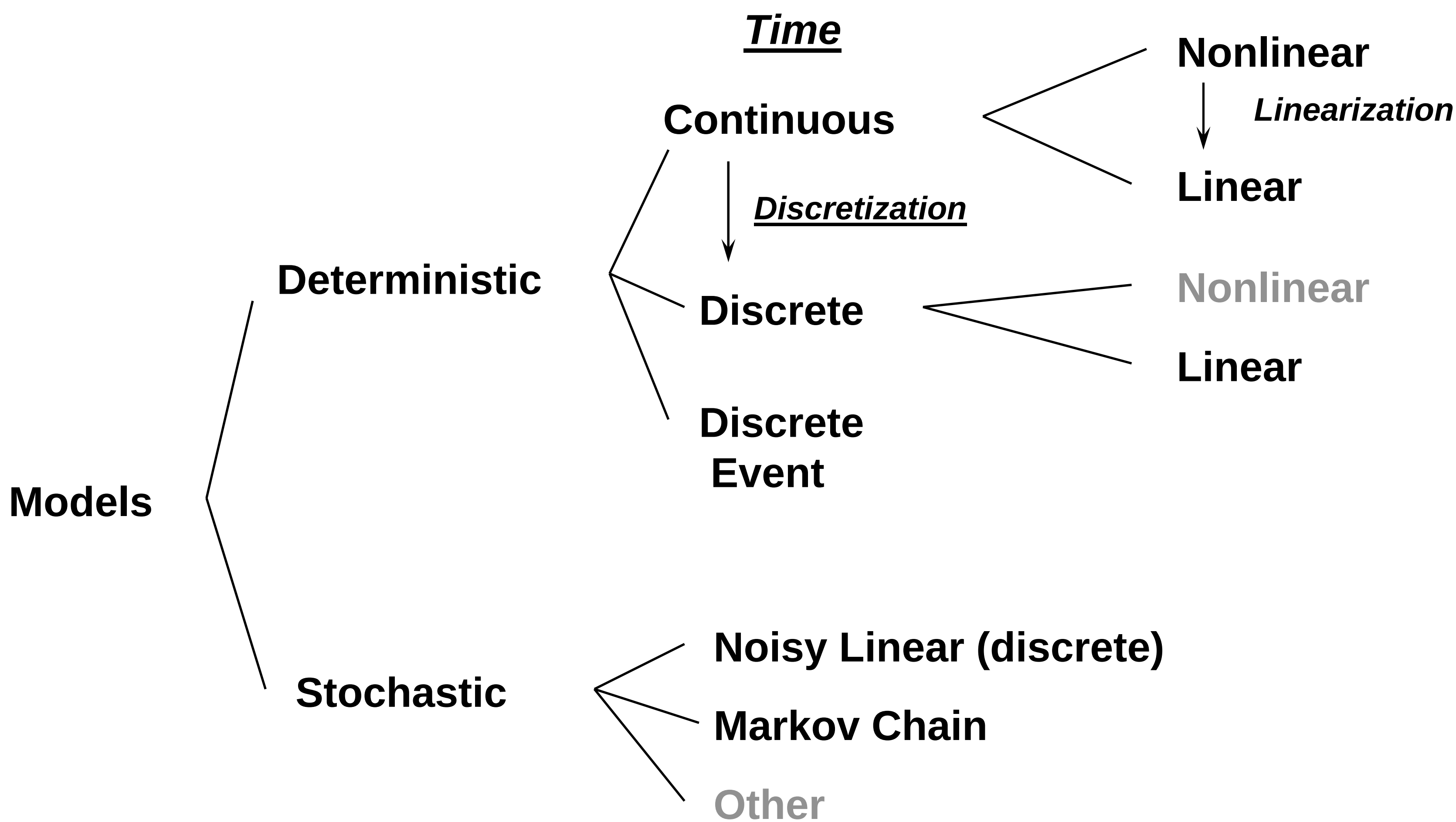
SYST611: Systems Methodology and Modeling #4

Kuo-Chu Chang
Fairfax, Virginia

Outline

- Discretization of Continuous Systems
- Sampling Theorem
- Analysis of Discretization Methods
- Nonlinear Systems
- Iterative Solution to Nonlinear Systems

Taxonomy of Models



Discretization of Continuous Systems

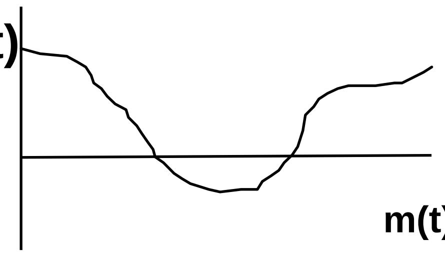
- **Motivation**
  - System simulation and analysis using digital computers and numerical methods
  - Digital signal processing and control for real-time applications
- **Sampling Theorem**
  - Uniform sampling of signals
  - Nyquist sampling rate allows perfect reconstruction of signals (R > 2 W)
- **Discretization Methods**
  - Sample and hold
  - Forward and backward difference
  - Trapezoidal method

Sampling Theorem

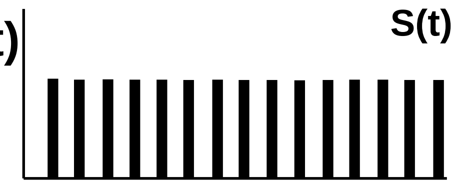
$$S(t) = \frac{dt}{T_s} + \frac{2\,dt}{T_s}\left(\cos\frac{2\pi t}{T_s} + \cos 2\frac{2\pi t}{T_s} + \cdots\right)$$ **(Fourier Series)**

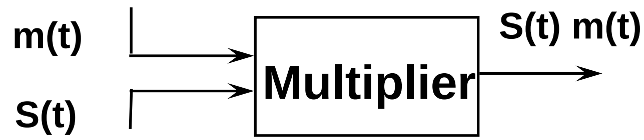
$$S(t)m(t) = \frac{dt}{T_s}m(t) + \frac{2\,dt}{T_s}\left(\cos\frac{2\pi t}{T_s}m(t) + \cos 2\frac{2\pi t}{T_s}m(t) + \cdots\right)$$

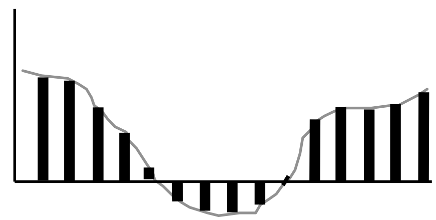
m(t)

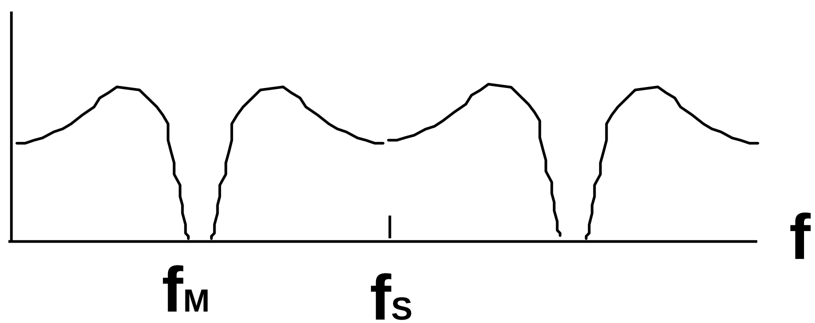


S(t)









$f_S > 2f_M$

Discretization Methods
Sample and Hold

- **System input stays constant between sampling instants:** $u(t) = u(kh), \quad kh \le t < (k+1)h$
- **Take sampled values of the state and output as approximations over the times between sampling instances**

$$\dot{x}(t) = Ax(t) + Bu(t) \Rightarrow x(t) = e^{A(t-t_0)}x(t_0) + \int_{t_0}^{t} e^{A(t-\tau)}Bu(\tau)d\tau$$

$$\Rightarrow x((k+1)h) = e^{Ah}x(kh) + \int_{kh}^{(k+1)h} e^{A((k+1)h-\tau)}Bu(\tau)d\tau$$

$$\Rightarrow x((k+1)h) = e^{Ah}x(kh) + \left(\int_{0}^{h} e^{A\tau'}Bd\tau'\right)u(kh), \text{where } \tau' = (k+1)h - \tau$$

$$x((k+1)h) = e^{Ah}x(kh) + A^{-1}(e^{Ah} - I)Bu(kh) \equiv Fx(kh) + Gu(kh), \text{ where } F = e^{Ah}$$

Sample and Hold
Example

$$\dot{\boldsymbol{x}} = \begin{bmatrix} \dot{q} \\ \dot{a} \end{bmatrix} = \begin{bmatrix} -0.6 & 18 \\ 1 & -1.2 \end{bmatrix} \begin{bmatrix} q \\ a \end{bmatrix} + \begin{bmatrix} m_e \\ k_e \end{bmatrix} e(t) = A\boldsymbol{x} + \boldsymbol{B}e(t)$$

q : pitch rate, a :angle of attack, e :elevator control

$$\Rightarrow \boldsymbol{x}(k+1) = F\boldsymbol{x}(k) + Ge(k) = e^{Ah}\boldsymbol{x}(k) + A^{-1}(e^{Ah} - I)Be(k)$$

$$\det[A - \lambda I] = 0 \Rightarrow \lambda_1 = 3.35, \lambda_2 = -5.15; x_1 = \begin{bmatrix} 1 \\ 0.22 \end{bmatrix}, x_2 = \begin{bmatrix} 1 \\ -0.25 \end{bmatrix}$$

$$F = e^{Ah} = Te^{\Lambda h}T^{-1} = \begin{bmatrix} 1 & 1 \\ 0.22 & -0.25 \end{bmatrix} \begin{bmatrix} e^{3.35h} & 0 \\ 0 & e^{-5.15h} \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 0.22 & -0.25 \end{bmatrix}^{-1}$$

$$G = A^{-1}(F - I)B = \begin{bmatrix} 0.07 & 1.04 \\ 0.06 & 0.03 \end{bmatrix} (F - I) \begin{bmatrix} m_e \\ k_e \end{bmatrix}$$

Sample and Hold
Issues

$$x((k+1)h) = e^{Ah}x(kh) + A^{-1}(e^{Ah} - I)Bu(kh)$$
$$\equiv Fx(kh) + Gu(kh),$$

where $F = e^{Ah}$

(1) Since negative λ implies $\left|e^{\lambda h}\right| < 1$, stability (as well as instability) is preserved

(2) If input agrees with the assumption (stays constant between sampling instants), there is no approximation

Discretization Methods
Forward Difference

- **Approximate the State Equations**
- **Replace Derivative by Difference Quotient**
- **Forward Difference (Euler method)**

$$\dot{x}(kh) \approx \frac{x((k+1)h) - x(kh)}{h}$$

$$x((k+1)h) \approx h\dot{x}(kh) + x(kh) = h(Ax(kh) + Bu(kh)) + x(kh)$$

$$= (I + hA)x(kh) + hBu(kh) \equiv Fx(kh) + Gu(kh)$$

- **Explicit Method**

$$\dot{x}(kh) \approx f(x(kh), u(kh)), \ \ \boldsymbol{kh \le t < (k+1)h}$$

$$x((k+1)h) = x(kh) + hf(x(kh), u(kh))$$

Forward Difference

Example

$$\dot{\boldsymbol{x}} = \begin{bmatrix} \dot{q} \\ \dot{a} \end{bmatrix} = \begin{bmatrix} -0.6 & 18 \\ 1 & -1.2 \end{bmatrix} \begin{bmatrix} q \\ a \end{bmatrix} + \begin{bmatrix} m_e \\ k_e \end{bmatrix} e(t) = A\boldsymbol{x} + \boldsymbol{B}e(t)$$

$$\Rightarrow \boldsymbol{x}(k+1) = F\boldsymbol{x}(k) + Ge(k) = (I + Ah)\boldsymbol{x}(k) + hBe(k)$$

$$F = I + Ah = \begin{bmatrix} 1-0.6h & 18h \\ h & 1-1.2h \end{bmatrix}, \; G = hB = \begin{bmatrix} hm_e \\ hk_e \end{bmatrix}$$

(1) This is the same as the Sample and Hold method with F replaced by I+Ah and the assumption that input is fixed for kh<t<(k+1)h

(2) It is a first-order approximation: exp(Ah) ~ I+Ah

Forward Difference
Stability Preservation

Forward difference method preserves instability but not stability

Note that with Z - Transform

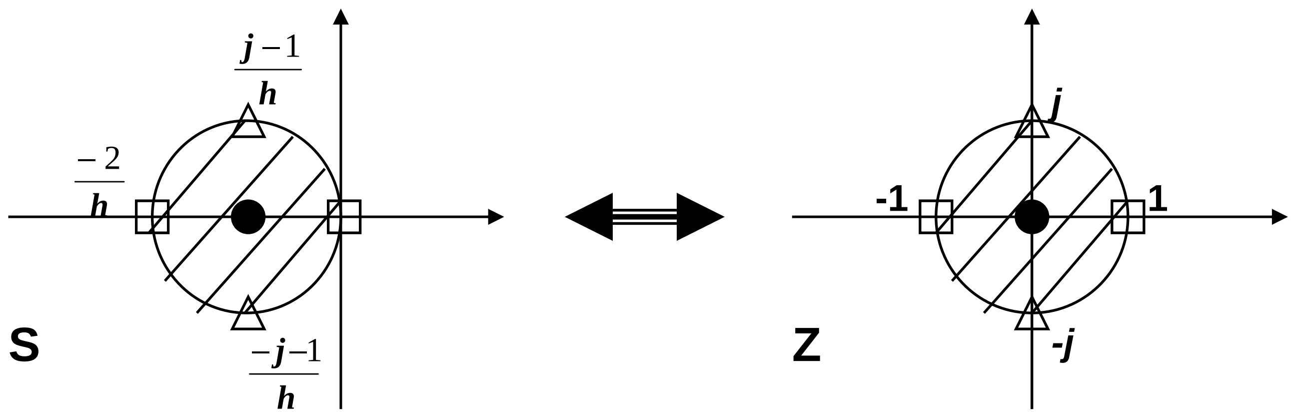
$$zX(z) = (I + Ah)X(z) + hBU(z), Y(z) = CX(z)$$

$$\Rightarrow H_z(z) = \frac{Y(z)}{U(z)} = C\left(\left(\tfrac{z-1}{h}\right)I - A\right)^{-1}B$$

But with Laplace Transform

$$sX(s) = AX(s) + BU(s),\ Y(s) = CX(s)$$

$$\Rightarrow H_s(s) = \frac{Y(s)}{U(s)} = C\left(sI - A\right)^{-1}B \Rightarrow H_z(z) = H_s\left(\frac{z-1}{h}\right)$$



Discretization Methods

Backward Difference

- **Approximate the State Equations**
- **Replace Derivative by Difference Quotient**
- **Backward Difference**

$$\dot{x}((k+1)h) \approx \frac{x((k+1)h) - x(kh)}{h}$$

$$x((k+1)h) = h\dot{x}((k+1)h) + x(kh)$$

$$= h(Ax((k+1)h) + Bu((k+1)h)) + x(kh)$$

$$\Rightarrow \; x((k+1)h) = (I - hA)^{-1}x(kh) + h(I - hA)^{-1}Bu((k+1)h)$$

- **Implicit Method**

$$\dot{x}((k+1)h) \approx f(x((k+1)h), u((k+1)h))$$

$$x((k+1)h) = x(kh) + h\dot{x}((k+1)h)$$

$$= x(kh) + hf(x((k+1)h), u((k+1)h))$$

Backward Difference
Example

$$\dot{\boldsymbol{x}} = \begin{bmatrix} \dot{\boldsymbol{q}} \\ \dot{\boldsymbol{a}} \end{bmatrix} = \begin{bmatrix} -0.6 & 18 \\ 1 & -1.2 \end{bmatrix} \begin{bmatrix} \boldsymbol{q} \\ \boldsymbol{a} \end{bmatrix} + \begin{bmatrix} \boldsymbol{m_e} \\ \boldsymbol{k_e} \end{bmatrix} \boldsymbol{e}(\boldsymbol{t}) = \boldsymbol{Ax} + \boldsymbol{Be}(\boldsymbol{t})$$

$$\Rightarrow \boldsymbol{x}(\boldsymbol{k}+1) = \boldsymbol{Fx}(\boldsymbol{k}) + \boldsymbol{Ge}(\boldsymbol{k}) = (\boldsymbol{I} - \boldsymbol{Ah})^{-1} \boldsymbol{x}(\boldsymbol{k}) + \boldsymbol{h}(\boldsymbol{I} - \boldsymbol{Ah})^{-1} \boldsymbol{Be}(\boldsymbol{k})$$

$$F = (I - Ah)^{-1} = \begin{bmatrix} 1+0.6h & 18h \\ h & 1+1.2h \end{bmatrix}^{-1}$$

$$G = hFB = h(I - Ah)^{-1} B$$

(1) This is similar to the Sample and Hold method with F replaced by (I-Ah)^(-1) and the assumption that input is fixed for kh<t<(k+1)h
(2) It is a first-order approximation: exp(-Ah) ~ I-Ah

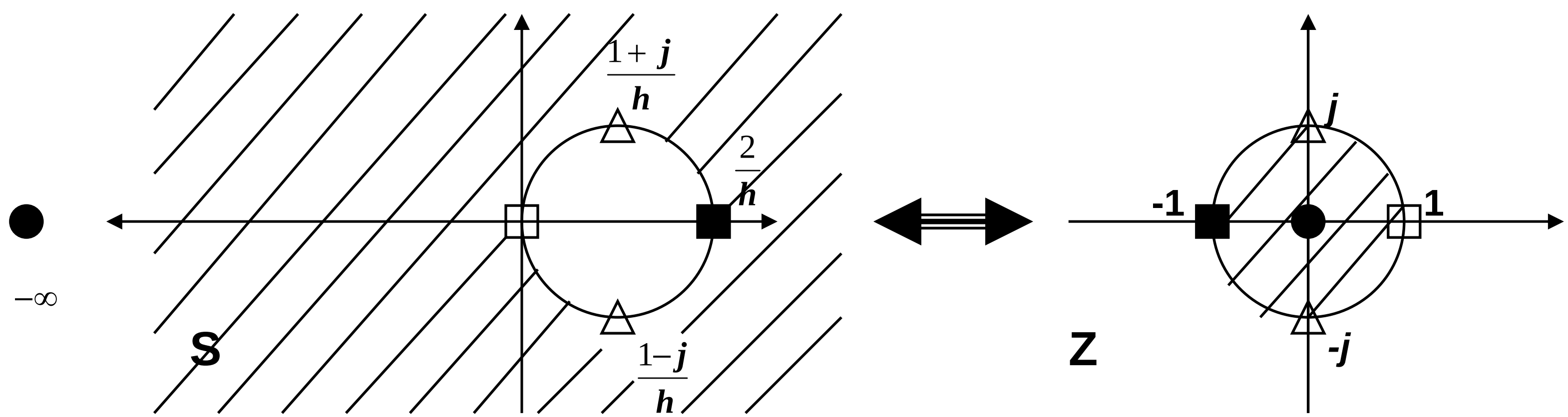
Backward Difference
Stability Preservation

Backward difference method preserves stability but not instability

Note:

$$zX(z) = (I - Ah)^{-1} X(z) + h(I - Ah)^{-1} BzU(z),\ Y(z) = CX(z)$$

$$\Rightarrow H_z(z) = \frac{Y(z)}{U(z)} = C\left(\left(\tfrac{z-1}{zh}\right)I - A\right)^{-1} B = H_s\left(\tfrac{z-1}{zh}\right)$$



Discretization Methods
Trapezoidal Method

- **Approximate the State Equations**
- **Replace Derivative by Average Value**
- **Trapezoidal Method**

$$\dot{x}(kh) \approx \frac{1}{2}[\,f(x(kh),u(kh)) + f(x((k+1)h),u((k+1)h))]$$

$$= \frac{1}{2}[Ax(kh) + Bu(kh) + Ax((k+1)h) + Bu((k+1)h)]\text{(for linear system)}$$

$$x((k+1)h) = x(kh) + h\dot{x}(kh)$$

$$\Rightarrow x((k+1)h) = \left(I - \frac{hA}{2}\right)^{-1}\left(I + \frac{hA}{2}\right)x(kh) + \frac{1}{2}\left(I - \frac{hA}{2}\right)^{-1} Bh\big(u((k+1)h) + u(kh)\big)$$

$$= Fx(kh) + G\big(u((k+1)h) + u(kh)\big)$$

Trapezoidal Method

Example

$$\dot{x} = \begin{bmatrix} \dot{q} \\ \dot{a} \end{bmatrix} = \begin{bmatrix} -0.6 & 18 \\ 1 & -1.2 \end{bmatrix} \begin{bmatrix} q \\ a \end{bmatrix} + \begin{bmatrix} m_e \\ k_e \end{bmatrix} e(t) = Ax + Be(t)$$

$$\Rightarrow x(k+1) = Fx(k) + G[e(k) + e(k+1)]$$

$$= (I - \tfrac{1}{2}Ah)^{-1}(I + \tfrac{1}{2}Ah)x(k) + \tfrac{1}{2}h(I - \tfrac{1}{2}Ah)^{-1}B[e(k) + e(k+1)]$$

$$F = (I - \tfrac{1}{2}Ah)^{-1}(I + \tfrac{1}{2}Ah) = \begin{bmatrix} 1+0.3h & -9h \\ -0.5h & 1+0.6h \end{bmatrix}^{-1} \begin{bmatrix} 1-0.3h & 9h \\ 0.5h & 1-0.6h \end{bmatrix},$$

$$G = \tfrac{1}{2}(I - \tfrac{1}{2}Ah)^{-1}hB = \tfrac{1}{2}\begin{bmatrix} 1+0.3h & -9h \\ -0.5h & 1+0.6h \end{bmatrix}^{-1} \begin{bmatrix} hm_e \\ hm_k \end{bmatrix}$$

(1) It is a second-order approximation:
exp(Ah) ~ I+Ah+1/2(Ah)^2

(2) This method preserves stability as well as instability

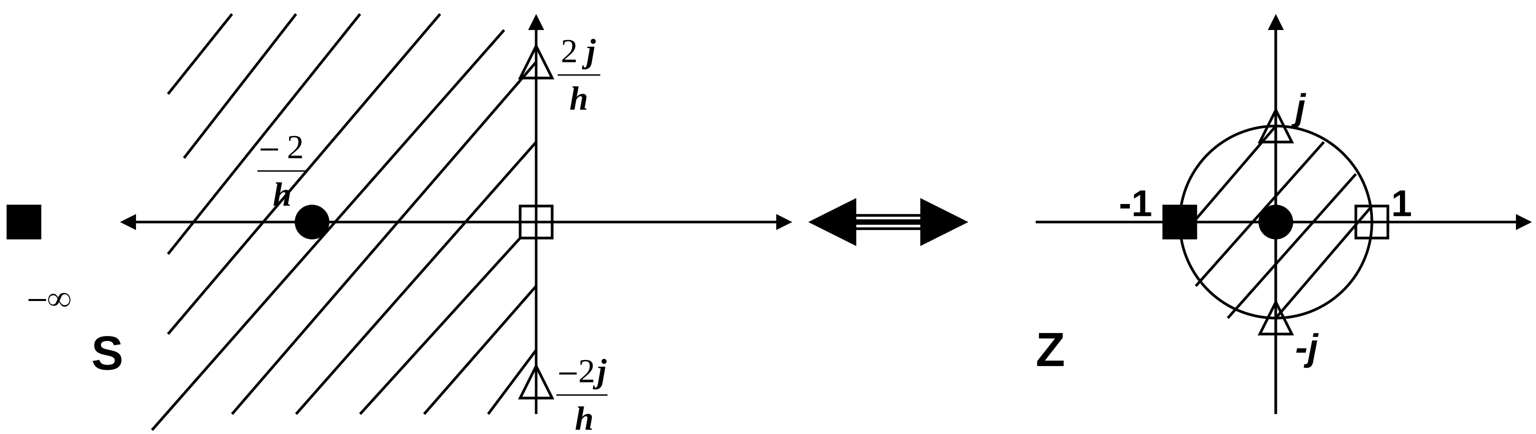
Trapezoidal Method
Stability Preservation

Trapezoidal method preserves both stability and instability

Note:

$$zX(z) = (I - \tfrac{1}{2}Ah)^{-1}(I + \tfrac{1}{2}Ah)X(z) + \tfrac{1}{2}(I - \tfrac{1}{2}Ah)^{-1}Bh(z+1)U(z), Y(z) = CX(z)$$

$$\Rightarrow H_z(z) = \frac{Y(z)}{U(z)} = C\left(\tfrac{2}{h}\left(\tfrac{z-1}{z+1}\right)I - A\right)^{-1}B = H_s\left(\tfrac{2}{h}\left(\tfrac{z-1}{z+1}\right)\right)$$



MATLAB Example

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix} = \dot{x}(t) = Ax(t) + Bu = \begin{bmatrix}0 & 1\\ 0 & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix} + \begin{bmatrix}0\\ 1\end{bmatrix}u$$

with $u = 1$ $for\, t \geq 0$ and $x(0) = \begin{bmatrix}0\\ 0\end{bmatrix}$, $ICBEST$ $(see\ Lec\#1, p.28-29)$

$$\begin{bmatrix}x_1(t)\\ x_2(t)\end{bmatrix} = \begin{bmatrix}t^2/2\\ t\end{bmatrix}$$

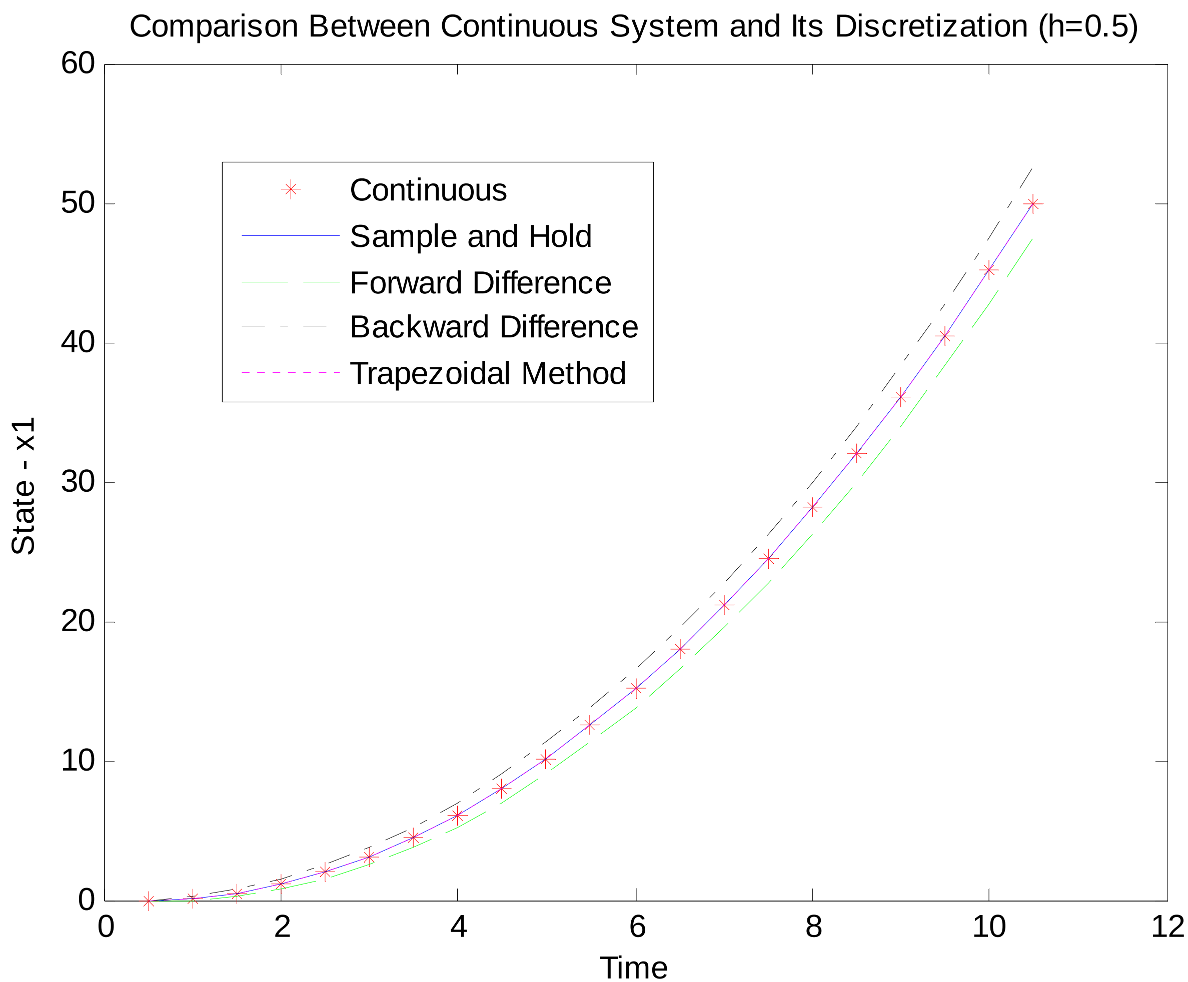
$$Sample\,\&\,Hold: F = e^{Ah} = \begin{bmatrix}1 & h\\ 0 & 1\end{bmatrix}, G = \int_0^h e^{A\tau}Bd\tau = \int_0^h \begin{bmatrix}1 & \tau\\ 0 & 1\end{bmatrix}\begin{bmatrix}0\\ 1\end{bmatrix}d\tau = \begin{bmatrix}h^2/2\\ h\end{bmatrix}$$

$$Forward\ Difference: F = I + hA = \begin{bmatrix}1 & h\\ 0 & 1\end{bmatrix}, G = hB = \begin{bmatrix}0\\ h\end{bmatrix}$$

$$Backward\ Difference: F = \left(I - hA\right)^{-1} = \begin{bmatrix}1 & -h\\ 0 & 1\end{bmatrix}^{-1}, G = h\left(I - hA\right)^{-1}B = \begin{bmatrix}1 & -h\\ 0 & 1\end{bmatrix}^{-1}\begin{bmatrix}0\\ h\end{bmatrix}$$

$$Trapezoidal: F = \begin{bmatrix}1 & -h/2\\ 0 & 1\end{bmatrix}^{-1}\begin{bmatrix}1 & h/2\\ 0 & 1\end{bmatrix}, G = \begin{bmatrix}1 & -h/2\\ 0 & 1\end{bmatrix}^{-1}\begin{bmatrix}0\\ h/2\end{bmatrix}$$

MATLAB Example – con't

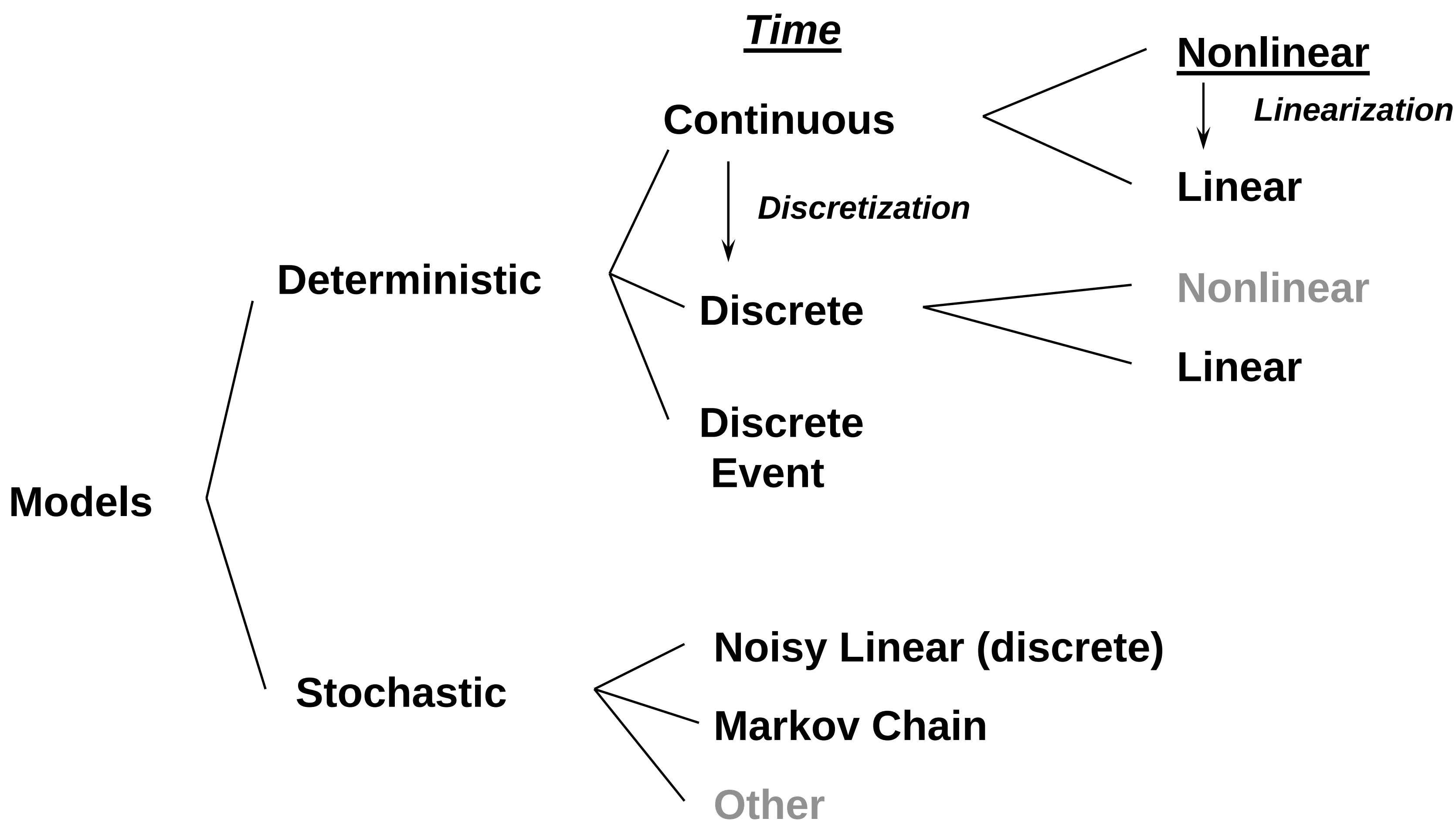


See MATLAB code - lec41.m

Issues in Discretization

- **Accuracy**
  - Sample-and-hold is exact when system input remains constant between sampling times
  - Forward and Backward difference provide first-order approximation
  - Trapezoidal method provides second-order accuracy
- **Stability**
  - Sample-and-hold preserves stability and instability
  - Forward discretization preserves instability but not stability
  - Backward discretization preserves stability but not instability
  - Trapezoidal method preserves both stability and instability
- **Computational Requirements**
  - Smaller sampling interval requires more computation
  - Sample-and-hold is most expensive due to the exact form
  - Forward and Backward difference are cheapest but inaccurate
  - Trapezoidal method balances the trade-off

Taxonomy of Models



Nonlinear Systems

- **Nonlinear System Equations:**

$$\dot{x}(t) = f(x(t), u(t)); \qquad x(0) = x_0$$
$$y(t) = h(x(t))$$

- **Approximated by Linear systems**

$$f(x(t), u(t)) = Ax(t) + Bu(t)$$
$$y(t) = Cx(t)$$

  - **Are only idealizations that are valid over certain limited ranges of the physical variables involved**
  - **Can be used as an approximation to a nonlinear one over some valid range**
  - **Good for "smooth" nonlinear systems**
  - **Typically, we apply linearization at "singular point", i.e.,**

$$\dot{x}(t) = f(x(t), u(t)) = 0$$

Solution of Nonlinear Equations

- **Given a General Vector Equation**

$$y = F(x) \quad \Rightarrow \quad G(x) = F(x) - y = 0$$

- **Newton's Method**
  - **An iterative method for generating a convergent sequence**
  - **Use derivatives to provide a sequence of linear approximations to the nonlinear equation**

$$G(x) \approx G(x_0) + G'(x_0)(x - x_0)$$

$$\text{take } x_0 \text{ as a guess} \qquad \Rightarrow \quad G(x_0) + G'(x_0)(x - x_0) = 0$$

$$G'(x_0)x = -G(x_0) + G'(x_0)x_0$$

$$\Rightarrow \quad x = x_0 - (G'(x_0))^{-1} G(x_0)$$

$$\Rightarrow \quad x_{k+1} = x_k - (G'(x_k))^{-1} G(x_k), \qquad k \geq 0$$

Example

(1) To find $\sqrt{2}$, let $G(x) = x^2 - 2 = 0$, then $G'(x) = 2x$

and $x_{k+1} = x_k - \dfrac{x_k^2 - 2}{2x_k}$

$x_0 = 1,\ x_1 = 1.5,\ x_2 = 1.4666..,\ x_3 = 1.414215..$

(2) To find π, let $G(x) = \sin x = 0$, then $G'(x) = \cos x$

and $x_{k+1} = x_k - \tan x$

$x_0 = 3,\ x_1 = 3.1425\ ..,\ x_2 = 3.1415\ \ldots$

(3) To find e, let $G(x) = \ln x - 1 = 0$, then $G'(x) = 1/x$

and $x_{k+1} = x_k - x_k(\ln x_k - 1) = x_k(2 - \ln x_k)$

$x_0 = 3,\ x_1 = 2.70416\ ..,\ x_2 = 2.71824\ \ldots$

Successive Approximation

- **Consider the nonlinear equation**

$$\dot{x}(t) = f(x(t), u(t)) \text{ for } t \in [0, T],\ x(0) = x^*$$

- **Successive Method**

$$x(t) = x^* + \int_0^t f(x(t), u(t))\,dt$$

$$x(t) = K(x(t)) \to x^{[k+1]}(t) = K(x^{[k]}(t))$$

$$\Rightarrow x^{[k+1]}(t) = x^* + \int_0^t f(x^{[k]}(t), u(t))\,dt$$

- **Example**

$$\dot{x}(t) = Ax(t),\ x(0) = x^*$$

$$x^{[1]}(t) = x^* + \int_0^t f(x^{[0]}(t), u(t))\,dt = (I + At)x^*$$

$$x^{[2]}(t) = (I + At + \frac{A^2 t^2}{2})x^* \Rightarrow x^{[k]}(t) = \left[\sum_{i=0}^{k} \frac{A^i t^i}{i!}\right] x^*$$

Example:
Midpoint Discretization

$$\dot{x}(t) = f(x(t))$$

$$x((k+1)h) \approx x(kh) + h\dot{x}(kh)$$

$$\approx x(kh) + hf\left(\frac{x((k+1)h) + x(kh)}{2}\right)$$

if the system is linear, then $\Rightarrow$ *trapezoidal discretization*

for nonlinear systems

$$\mathrm{x} = K(\mathrm{x}) = \mathrm{x}(kh) + h\mathrm{f}\left(\frac{\mathrm{x} + \mathrm{x}(kh)}{2}\right)$$

use successive approximation $x^{[i+1]}(t) = K(x^{[i]}(t))$ to find $\mathrm{x}((k+1)h)$