Discretization of Continuous Controllers

Thao Dang

VERIMAG, CNRS (France)

Discretization of Continuous Controllers

- One way to design a computer-controlled control system is to make a continuous-time design and then make a discrete-time approximation of this controller $\Rightarrow$ Analog Design Digital Implementation
- The computer-controlled system should now behave as the continuous-time system

Discretization of Continuous Controllers

- One way to design a computer-controlled control system is to make a continuous-time design and then make a discrete-time approximation of this controller $\Rightarrow$ Analog Design Digital Implementation
- The computer-controlled system should now behave as the continuous-time system

Discretization of Continuous Controllers

- One way to design a computer-controlled control system is to make a continuous-time design and then make a discrete-time approximation of this controller $\Rightarrow$ Analog Design Digital Implementation
- The computer-controlled system should now behave as the continuous-time system
- This is crucially dependent on choosing fairly short sampling periods.

Difference Approximations (1)

- When the continuous-time controller is specified as a transfer function $C(s)$, it is natural to look for methods that will transform the continuous transfer function $C(s)$ to a pulse transfer function $C_d(z)$ so that the corresponding behaviors of the two systems are close to each other.
- z and s are related as $z = exp(sT)$, where T is the sampling period.

Difference Approximations (2)

The difference approximations correspond to the series expansions

- $z = e^{sT} \approx 1 + sT$ (Forward difference or Euler's method)
- $z = e^{sT} \approx \frac{1}{1-sT}$ (Backward difference)
- $z = e^{sT} \approx \frac{1+sT/2}{1-sT/2}$ (Trapezoidal method, or Tustin's approximation)

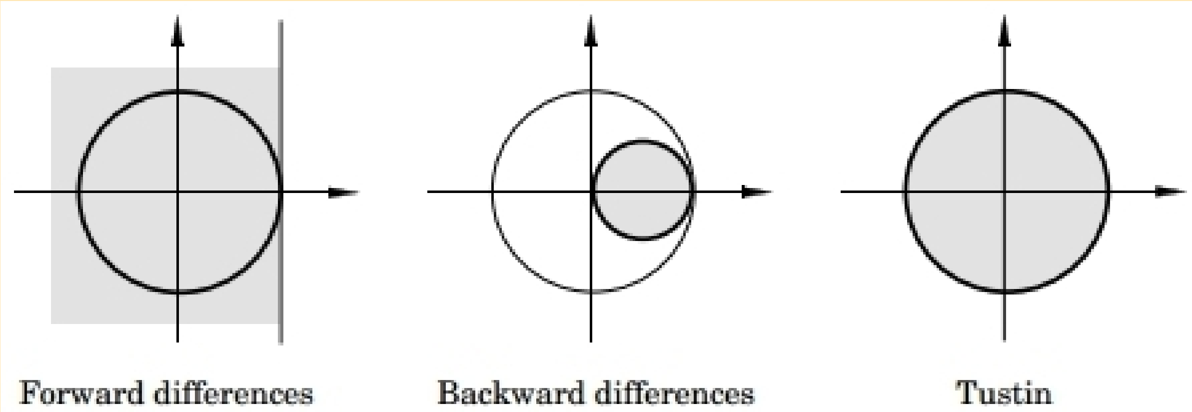
Computing transfer function $C_d(z)$

To calculate $C_d(z)$ we substitute s in $C(s)$ with the following:

- $s \approx \frac{z-1}{T}$ (Forward difference or Euler's method)
- $s \approx \frac{z-1}{zT}$ (Backward difference)
- $s \approx \frac{2}{T}\frac{z-1}{z+1}$ (Trapezoidal method, or Tustin's approximation)

Stability

the stability region (corresponding to the left half-plane $Re(s) \leq 0$) in the s-plane is mapped on the z-plane.



Stability - Remarks

- Forward-difference approximation: it is possible that a stable continuous-time system is mapped into an unstable discrete-time system.
- Backward approximation: a stable continuous-time system will always give a stable discrete-time system.
- Tustin's approximation: has the advantage that the left half s-plane is transformed into the unit disc in the z-plane.

Selection of Sampling Period and Anti-aliasing Filters

Choice of sampling rates and anti-aliasing flters are important

- Preserve stability
- Preserve performance

Anti-aliasing Filters: Example

Let ω_e be the sampling frequency.
The sampled system can be approximated by the hold circuit followed by the continuous-time system. The hold circuit can be written as:

$$\frac{1 - e^{sT}}{sT}$$

For small T, the hold can be approximated by a time delay of half a sampling interval.
We can use a filter of the form (where ω_f and ξ_f are parameters to choose)

$$G(s) = \frac{\omega_f^2}{s^2 + 2\xi_f \omega_f s + \omega_f^2}$$

The anti-aliasing filter will decrease the phase margin. We need to choose the parameters in order to achieve desired gain and phase margin decrease.
The gain of the filter at the Nyquist frequency ω_N $(= 2\omega_e)$ is approximately

$$g_N = (\frac{\omega_f}{\omega_N})^2$$

Hence, given a desired value of g_N, $\omega_f = \omega_N \sqrt{g_N}$.

Anti-aliasing Filters: Example (cont'd)

After calculations, the hold circuit and the anti-aliasing filter decrease the phase margin with

$$\phi = (0.5 + \frac{2\xi_f}{\pi\sqrt{g_N}})\omega_c T$$

where ω_c is the crossover frequency (in radians per second) of the continuous-time system. Note that the above 0.5 in the formula corresponds to the time delay of the hold (as mentioned earlier)

Assume that we choose $\xi_f = 0.707$ and $g_N = 0.1$ and the phase margin is allowed to be decreased by 5 to 15 degrees, **the following rule of thumb** is obtained: $\omega_c T$ is between 0.05 and 0.14

Application to LEGO robots

Compute the continuous-time transfer function of the closed loop $F_{BO}(s)$

Estimate the crossover frequency ω_c of $F_{BO}(s)$ using the function "margin" in matlab.

Choose sampling period T according to the rule: $\omega_c T$ **is between** 0.05 **and** 0.14

Discretize the controllers using one of the approximation methods and the chosen sampling period.

Note that instead of replacing the whole continuous-time controller with its discretized version, we can replace only its components containing continuous-time blocs, such as the integrators $\dfrac{1}{s}$