Discretizing a compensator transfer function

Integrator example

\[ v_c(s) = -\frac{1}{RC} v_e(s) \]  
\[ \text{(s domain)} \]

\[ i_c(t) = \frac{v_{el}(t)}{R} = \]

with \[ c \frac{dv_{el}(t)}{dt} = -i_c(t) = -\frac{v_{el}(t)}{R} \]

So \[ \frac{dv_c(t)}{dt} = -\frac{1}{RC} v_e(t) \]  
\[ \text{(derivative form)} \]

\[ v_c(t) = v_c(0) - \frac{1}{RC} \int_0^t v_e(u) du \]  
\[ \text{(integral form)} \]

\[ = v_c(0) - \frac{1}{C} \int_0^t v_e(u) du \text{ with } C = RC \]
Discretizing: Sample the waveforms with sampling period $T$

\[ u_c(nT) = u_c((n-1)T) - \frac{1}{T} \int_{(n-1)T}^{nT} u_e(u) \, du \]

Given the above input, approximate the sampled output:

Forward differencing rule:

\[ \int_{(n-1)T}^{nT} u_e(u) \, du \approx T \cdot u_e((n-1)T) \]

Then the integrator equation becomes

\[ u_c(nT) = u_c((n-1)T) - \frac{T}{C} \cdot u_e((n-1)T) \]
A hardware realization:

\[ V_e' \rightarrow \frac{T}{2} \rightarrow \triangle \rightarrow \text{latch} \rightarrow V_c \]

(output at \( nT \))

The \( z \) transform

The \( z \) operator is useful for modeling delays and uniformly-sampled systems.

The \( z \) transform of a sampled signal is

\[ V_c(z) = \sum_{n=0}^{\infty} v_c[nT] z^{-n} \]

with \( z = e^{sT} \)

Note \( z^{-1} = e^{-sT} \) represents a delay of length \( T \).
We can use the z-transform to represent difference equations. For the previous integrator example:

\[ v_c(nT) = v_c((n-1)T) - \frac{T}{C} v_e((n-1)T) \]

The z-transform is

\[ V_c(z) = z^{-1} \cdot V_c(z) - \frac{T}{C} z^{-1} V_e(z) \]

\[ V_c, \text{ delayed by one sampling period} \]

The z-domain transfer function is

\[ V_c(z) \cdot (1 - z^{-1}) = -\frac{T}{C} z^{-1} V_e(z) \]

\[ \Rightarrow \frac{V_c(z)}{V_e(z)} = -\frac{T}{C} \cdot \frac{z^{-1}}{1 - z^{-1}} = -\frac{T}{C} \cdot \frac{1}{z - 1} \]

System block diagram: use \( v_e(z) = z^{-1} \cdot (V_c(z) - \frac{T}{Z} V_e(z)) \)

```
input
\[ \frac{T}{Z} \]

\[ \text{multiply by constant} \]

\[ \text{subtract} \]

\[ \text{delay - label previous value} \]

\[ \text{output} \]
```

\[ v_c(z) \]

Another way to approximate the derivative: trapezoidal rule (bilinear or Tustin rule):

\[ v_e(t) \]

\[ v_e(\eta T) \]

\[ v_e((n-1)T) \]

\[ \text{area} \]

\[ (n-1)T \quad \eta T \quad t \]

We have

\[ v_c(\eta T) = v_c((n-1)T) - \frac{1}{T} \int_{(n-1)T}^{\eta T} v_e(u) du \]

\[ \text{area} \approx T \cdot \frac{v_e(\eta T) + v_e((n-1)T)}{2} \]

So

\[ v_c(\eta T) \approx v_c((n-1)T) - \frac{1}{2C} \left[ v_e(\eta T) + v_e((n-1)T) \right] \]

In the \( Z \) domain:

\[ v_c(\bar{z}) = \bar{z}^{-1} v_c(\bar{z}) - \frac{T}{2RC} \left[ v_e(\bar{z}) + \bar{z}^{-1} v_e(\bar{z}) \right] \]

The transfer function is

\[ \frac{v_c(\bar{z})}{v_e(\bar{z})} = - \frac{T}{2RC} \frac{1 + \bar{z}^{-1}}{1 - \bar{z}^{-1}} = - \frac{T}{2RC} \frac{\bar{z} + 1}{\bar{z} - 1} \]
A block diagram:

We can write

\[ V_c(z) = z^{-1} \left[ V_c(z) - \frac{T}{2RC} V_e(z) \right] - \frac{T}{2RC} V_e(z) \]

The block diagram illustrates a way to implement this integrator function in hardware.
Stability

If a function has a $z$-transform

$$V(z) = \frac{KN(z)}{(z-z_1)(z-z_2)\cdots}$$

with poles at $z_1, z_2, \ldots$,

then the inverse transform has terms of the form

$$v(nT) = K_1 z_1^n + K_2 z_2^n + \ldots$$

This represents a stable response provided that the poles have magnitude less than 1:

$$v(nT) \rightarrow 0 \text{ for large } n \text{ provided } |z_1| < 1, |z_2| < 1, \text{ etc.}$$

In other words, the poles must lie inside the unit circle in the complex $z$ plane.
Mapping from $s$ plane to $z$ plane

If $s = \sigma + j\omega$

than $z = e^{st} = e^{\sigma T} e^{j\omega T} = r e^{j\theta}$ with $r = e^{\sigma T}$ $\theta = \omega T$
Key observations

DC: \( S = 0 \) \( \rightarrow \) \( Z = 1 \)

Undamped: \( S = j\omega \) \( \rightarrow \) \( |Z| = 1 \) (unit circle)

Sinusoid (j\omega axis)

Real pole, high frequency: \( S = \sigma \rightarrow -\infty \) \( Z \rightarrow 0 \)

The mapping from \( S \) to \( Z \) depends on \( T \), the sampling period. And this mapping repeats for \( \omega > \frac{2\pi}{T} \): if \( S = j\omega \) then \( Z = e^{j\omega T} \)

High sampling frequencies (oversampling) causes s-domain poles to be mapped close to unit circle, with values such as 0.99998 etc. Rounding error then becomes significant, and the digital controller requires longer word lengths.
Low sampling frequencies (under-sampling) can lead to aliasing.

Nyquist criterion: must sample faster than twice the maximum signal frequency. Better to sample substantially faster than this.

Integral compensators

\[ v_c(s) = -\frac{1}{sT} v_e(s) \]

Forward differencing:

\[ v_c[n] = v_c[n-1] - \frac{T}{2} v_e[n-1] \]

\[
\frac{v_c(z)}{v_e(z)} = -\frac{T}{2 \frac{z-1}{z}}
\]

Backwards differencing:

\[ v_c[n] = v_c[n-1] - \frac{T}{2} v_e[n] \]

\[
\frac{v_c(z)}{v_e(z)} = -\frac{T}{2 \frac{z}{z-1}}
\]

Bilinear (trapezoidal):

\[ v_c[n] = v_c[n-1] - \frac{T}{2T} (v_e[n] + v_e[n-1]) \]

\[
\frac{v_c(z)}{v_e(z)} = -\frac{T}{2 \frac{z+1}{z-1}}
\]
Mapping an s-domain compensator transfer function into the z-domain

Based on trapezoidal integration: let \( s = K \frac{2t-1}{2+1} \)

(there are many approximate ways to approximate an s-domain transfer function in the z-domain. The bilinear transformation has the advantage of preserving stability: the \( s=j\omega \) axis is mapped into the \( |z|=1 \) unit circle). Note that the exact mapping \( z=e^{sT} \) leads to an infinite number of poles; the approximation \( z = K \frac{2t-1}{2+1} \) gives a reasonably accurate result with a finite number of poles and zeroes. The magnitude and phase of the continuous and discrete filters can be made to exactly match at one frequency point by the judicious choice of \( K \), as follows:

at \( s=j\omega T \), we want \( s = K \frac{2t-1}{2+1} \) to be equal to the actual value according to \( z=e^{sT} \). Hence

\[
j\omega T = K \frac{e^{j\omega T} - 1}{e^{j\omega T} + 1}.
\]

Note \( \frac{e^{j\omega} - 1}{e^{j\omega} + 1} = j \tan(\frac{\omega}{2}) \)
So we get

\[ j \omega_{\text{int}} = j K \tan \left( \frac{\omega_{\text{int}} T}{2} \right) \]

we should therefore choose

\[ K = \frac{\omega_{\text{c}}}{\tan \left( \frac{\omega_{\text{c}} T}{2} \right)} \]

to preserve the magnitude and phase at \( \omega = \omega_{\text{int}} \).

For example, we could select \( \omega_{\text{c}} = \omega_c \) the crossover frequency, so that the bilinear transformation does not alter the crossover frequency or phase margin.

The bilinear transformation is built into many software tools such as MATLAB.

For \( \omega_{\text{int}} \ll \frac{2 \pi}{T} \), \( K \rightarrow \frac{2}{T} \) and

The bilinear transformation becomes

\[ s = \frac{2}{T} \left( \frac{2 - 1}{2 + 1} \right) \]
Mapping common terms using the bilinear transformation

1. Integrator

\[ G(s) = \frac{\omega_0}{s} \]

Then

\[ G(z) = \frac{\omega_0}{K} \frac{z+1}{z-1} = \frac{\omega_0 T}{2} \frac{z+1}{z-1} \quad \text{for} \quad K \to \frac{2}{T} \]

**S plane:** pole at \( s = 0 \)

**Z plane:** pole at \( z = +1 \)
zero at \( z = -1 \)

**Impulse response:**

\[ g(nT) = 1 \quad \text{for} \quad n \geq 0 \]
2. Real pole

\[ G(s) = \frac{1}{1 + \frac{s}{\omega_p}} \]

Apply the bilinear transformation

\[ s = \frac{2}{T} \frac{z-1}{z+1} \]

So

\[ G(z) = \frac{1}{1 + \frac{2}{\omega_p T} \frac{z-1}{z+1}} \]

after some simplification, we obtain

\[ G(z) = G_0 \frac{z+1}{z-r} \quad \text{with} \quad G_0 = \frac{\omega_p T}{\omega_p T + 2} \]

\[ r = \frac{2 - \omega_p T}{2 + \omega_p T} \]

The inverse transform of \( G = \frac{z}{z-r} \) is \( g(n) = r^n \)
3. Complex poles

\[ G(s) = \frac{1}{1 + \frac{s}{\omega_0} + \left(\frac{s}{\omega_0}\right)^2} \]

Upon substitution \( s \rightarrow \frac{2}{t} \frac{z-1}{\beta+1} \)

one obtains

\[ G(z) = \frac{(z+1)^2}{az^2 + bz + c} \]

with

\[ a = 1 + \frac{2}{Q \omega_0 T} + \frac{4}{Q \omega_0^2 T^2} \]

\[ b = 2 - \frac{8}{Q \omega_0^2 T^2} \]

\[ c = 1 - \frac{2}{Q \omega_0 T} + \frac{4}{Q \omega_0^2 T^2} \]

\[ g(nT) \] has terms of the form \( r^n \cos(n \theta) \)

Poles at \( z_1, z_2 \); \( z_1 = r e^{j \phi} \)

Settling time determined by \( r \)

Oscillation time determined by \( \theta \)
Application of the bilinear transformation to a compensator gain

Example: \[ G_c(s) = G_{CM} \frac{(1 + \frac{\omega_1}{s})(1 + \frac{s}{\omega_2})}{(1 + \frac{s}{\omega_1})(1 + \frac{s}{\omega_2})} \]
(compensator gain - digital part)

bilinear transformation: \[ s = K \frac{z-1}{z+1} \]

with \[ K = \frac{20 \text{ fs}}{\tan(\text{fs} \cdot T)} \]
\[ T = \text{sampling period} = \frac{1}{f_s} \]
\[ f_s = \text{critical frequency at which bilinear transformation preserves mag & phase} \]

Define \[ K = 2 \xi f_s \]
\[ \xi = \frac{f_s \tan(\text{fs} / f_s)}{f_s \tan(\text{fs} / f_s)} \]

\[ \text{typically } \xi \text{ is close to 1} \]

so \[ s = 2 \xi f_s \prod \frac{z-1}{z+1} \]
Plug into transfer function

\[ G_c(z) = G_{CM} \left[ 1 + \frac{\omega_L}{2\pi f_s \left( \frac{z-1}{z+1} \right)} \right] \left[ \frac{1 + \frac{2\pi f_s}{\omega p_1}}{\frac{z-2}{z+2}} \right] \]

Simplify via algebra to obtain

\[ G_c(z) = G_{CD} \frac{(z-2z_L)(z-2z_{21})}{(z-z_p_1)(z-1)} \]

with

\[ z_L = \frac{1 - \frac{\pi f_L}{\pi f_s}}{1 + \frac{\pi f_L}{\pi f_s}} \]

\[ z_{21} = \frac{1 - \frac{\pi f_{21}}{\pi f_s}}{1 + \frac{\pi f_{21}}{\pi f_s}} \]

\[ z_{p_1} = \frac{1 - \frac{\pi f_{p_1}}{\pi f_s}}{1 + \frac{\pi f_{p_1}}{\pi f_s}} \]

\[ G_{CD} = G_{CM} \frac{(1 + \frac{\pi f_L}{\pi f_s})(1 + \frac{\pi f_{p_1}}{\pi f_{21}})}{(1 + \frac{\pi f_{p_1}}{\pi f_{p_1}})} \]
For the values

\[ G_{Cm} = 5.45 \]
\[ f_{21} = 33 \text{kHz} \]
\[ f_L = 8 \text{kHz} \]
\[ f_{p1} = 300 \text{kHz} \]

Then

\[ \bar{\eta} = \frac{f_{cut}}{f_s \tan \left( \frac{f_{cut}}{f_s} \right)} = 0.967 \]

and

\[ z_L = 0.9493 \] (low frequency zero at \( f_L \rightarrow \) zero near \( z = 1 \))
\[ z_{21} = 0.8063 \]
\[ z_{p1} = 0.01278 \] (high frequency pole at \( f_{p1} \rightarrow \) pole near \( z = 0 \))

So,

\[ G_c(z) = \frac{(28.5016)(z-0.9493)(z-0.8063)}{(z-0.01278)(z-1)} \]

\[ \text{300kHz pole} \quad \text{integrator pole} \]
Realization of the transfer function

\[
\frac{V_c(z)}{V_e(z)} = G_c(z) = G_{cd} \frac{(z-z_2)(z-z_{21})}{(z-z_p)(z-1)}
\]

Since multiplication by \(z^n\) represents a delay of one sampling period, let's manipulate into forms involving delays \(z^{-1}\), \(z^{-2}\), etc. Divide the bottom by \(z^2\).

\[
\frac{V_c(z)}{V_e(z)} = G_{cd} \frac{(1-zLz^{-1})(1-z_{21}z^{-1})}{(1-z_pz^{-1})(1-z^{-1})}
\]

Express as polynomials:

\[
V_c(z) \cdot (1-z_{21}z^{-1})(1-z^{-1}) = G_{cd} V_e(z) (1-z_{21}z^{-1})(1-z_{21}z^{-2})
\]

Multiply out:

\[
V_c(z) \cdot \left[1 - (z+z_p)z^{-1} + z_p z^{-2}\right] =
\]

\[
G_{cd} V_e(z) \left[1 - (z+2z_{21})z^{-1} + 2zLz_{21}z^{-2}\right]
\]

Now invert transform:

\[
V_c(nT) - (1+z_p) V_c((n-1)T) + z_p V_c((n-2)T) =
\]

\[
G_{cd} \left[V_e(nT) - (zL+z_{21}) V_e((n-1)T) + zLz_{21} V_e((n-2)T)\right]
\]
A hardware realization:

\[ V_e(nT) \xrightarrow{G_{cd}} G_{cd} V_e(nT) \xrightarrow{\text{adder}} V_c(nT) \]

\[ G_{cd} V_e((n-1)T) \xrightarrow{\text{delay}} G_{cd} V_e((n-2)T) \xrightarrow{\text{delay}} \]

\[-(Z_2 + Z_{21}) \xrightarrow{\text{gain}}\]

\[ V_c((n-1)T) \xrightarrow{\text{delay}} V_c((n-2)T) \xrightarrow{\text{delay}} \]

\[ 1 + Z_{21} \xrightarrow{\text{gain}}\]

\[ -Z_{21} \xrightarrow{\text{gain}}\]