1 IIR Filters

The goal is to introduce

- IIR filters,
- their system function,
- their poles and zeros.

The idea of the system function and poles and zeros has stronger motivation than it does for FIR filters.

Also, for IIR filters, students see that the impulse response and filter coefficients really are distinct concepts.

1.1 The definition of an IIR Filter

From now on, the **IIR filters** we consider are exclusively those defined

(a) by a difference equation of the following form:

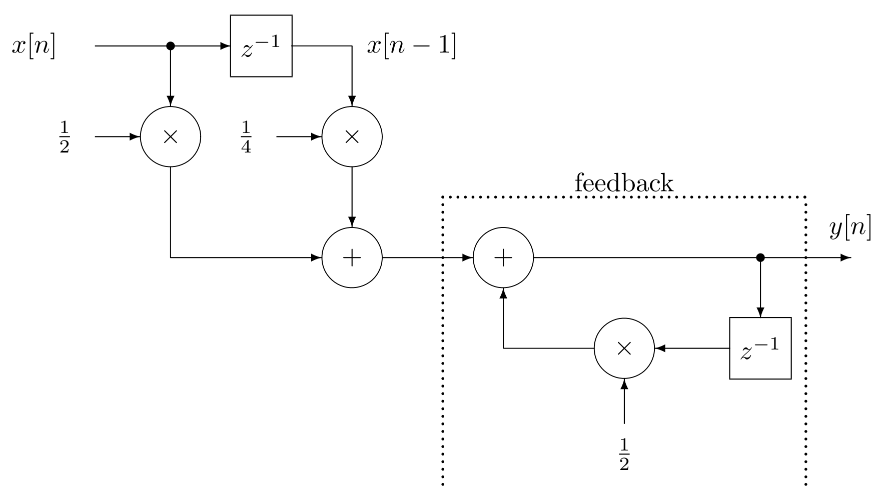
$$y[n] = a_1 y[n-1] + a_2 y[n-2] + \cdots + a_N y[n-N] + b_0 x[n] + b_1 x[n-1] + \cdots + b_M x[n-M],$$

where $N \geq 1$ and $M \geq 0$,

(b) and by the **initial rest condition**:
If $x[n] = 0$ for $n < k$, then $y[n] = 0$ for $n < k$.

Example 1: $y[n] = \frac{1}{2} y[n-1] + x[n]$.

Example 2: $y[n] = \frac{1}{2} y[n-1] + \frac{1}{2} x[n] + \frac{1}{4} x[n-1]$



Regarding the initial rest condition:

(i) When the system is "powered on", e.g., at $n = 0$, the output at the time

$$y[0] = a_1 y[-1] + a_2 y[-2] + \cdots + a_N y[-N] + b_0 x[0] + b_1 x[-1] + \cdots + b_M x[-M]$$

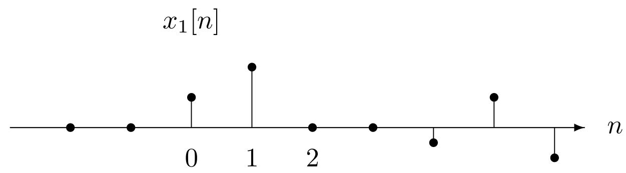
will depend on the "previous" output values. What are these previous values, when the system is "just turned on"? The initial rest condition addresses this issue:

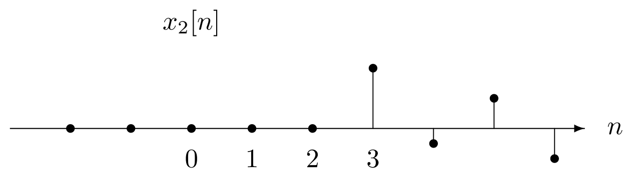
These previous values are all zero.

(ii) How do we know when the system is "turned on"?

The individual input signal $x[n]$ determines the "turn-on" time.

The "turn-on" time is the first time at which $x[n]$ becomes nonzero.





The "trun-on" time of $x_1[n]$ is $n = 0$ while that of $x_2[n]$ is $n = 3$

(iii) The IRC states that

all the output values are zero prior to the "turn-on" time.

Notes:

(i) An IIR filter is characterized by coefficients $a_1, \ldots, a_N$ and $b_0, \ldots, b_M$ of a difference equation.

(ii) $y[n]$ depends on present and past inputs $x[n], \ldots, x[n-M]$, just as for FIR filters.

(iii) $y[n]$ also depends on past outputs, $y[n-1], \ldots, y[n-N]$. This is the new feature.

(iv) The a_i's are called **feedback** coefficients; the terms $a_i y[n-i]$ are called **feedback** terms. The b_i's are called **feedforward** coefficients; the terms $b_i y[n-i]$ are called **feedforward** terms.

(v) An IIR filter is linear and time-invariant.

(vi) It is causal. (We only consider causal IIR filters.)

(vii) Its *order* is usually defined to be N, though this is not a universal convention. (Some might say the order is $M + N$.)

(viii) Motivation for IIR filters:

We'll be able to create frequency responses we cannot create with FIR filters, e.g. shaper cutoffs and peaks.

IIR Filters As the following bullets will illustrate, IIR filters are generally more difficult to deal with and analyze than FIR filters.

- **Stability:** IIR filters can be stable or unstable. It's the feedback that might cause instability. But it doesn't always. (Recall that FIR filters are always stable.)

  (i) Example of an unstable IIR filter:

  $$y[n] = y[n-1] + x[n].$$

  If $x[n] = u[n]$, then $y[n] = n$ which grows to infinity as n increases.
  This shows that *a bounded input can lead to an unbounded output*, which means the system is unstable.

  (ii) On the other hand, just so you don't think that all IIR filters are unstable, the following IIR filter is stable

  $$y[n] = \frac{1}{2}y[n-1] + x[n].$$

  (iii) We'll later find a nice z-domain criteria for checking stability.

  (iv) *Most applications demand stable filters.*

- **Impulse Response:** The impulse response sequence of an IIR filter has infinite support, which is why we use the name IIR.

  (i) The terms of the impulse response sequence can be found numerically simply by substituting an impulse as the input to the difference equation.

  (ii) However, since the impulse response has infinite length, this doesn't give us a complete description of the impulse response sequence.

  (iii) Instead we'd like a closed form expression for the impulse response.

  (iv) The latter is usually difficult to compute.

  (v) It does not usually have a simple relationship to the coefficients.

  (vi) The impulse response $h[n]$ of an IIR filter is not as important for IIR filters as for FIR filters since it is not ordinarily used to compute the output, i.e. we don't ordinarily implement a convolution of $h[n]$ with the input. Rather we implement the difference equation.

  (vii) Example: The following is the only kind of IIR filter for which the impulse response sequence can be found using elementary time-domain analysis. We'll later introduce z-domain analysis for finding the impulse response of a general IIR filter.

  $$y[n] = a_1 y[n-1] + b_0 x[n].$$

  To find the impulse response of this IIR filter, we choose the input to be an impulse: $x[n] = \delta[n]$, and we assume the initial rest condition holds: $y[n] = 0, n < 0$. Then the output produced in response to this input is the impulse response. Thus,

  $$\begin{aligned}
  h[n] &= y[n] = 0, \quad n < 0 \\
  h[0] &= y[0] = a_1 y[-1] + b_0 x[0] = a_1 \cdot 0 + b_0 \cdot 1 = b_0 \\
  h[1] &= y[1] = a_1 y[0] + b_0 x[1] = a_1 \cdot b_0 + b_0 \cdot 0 = a_1 \cdot b0 \\
  h[2] &= y[2] = a_1 y[1] + b_0 x[2] = a_1 \cdot a_1 b_0 + b_0 \cdot 0 = a_1^2 b_0 \\
  h[3] &= y[3] = a_1 y[2] + b_0 x[3] = a_1 \cdot a_1^2 b_0 + b_0 \cdot 0 = a_1^3 b_0
  \end{aligned}$$

  Continuing this, we see that, in general, for $n \geq 1$, $h[n] = h[n-1]a_0$. This implies

  $$h[n] = b_0 \cdot a_1^n u[n].$$

  This confirms that the impulse response is infinite.

(viii) The slightly more involved example:

$$y[n] = a_1 y[n-1] + a_2 y[n-2] + b_0 x[n].$$

This is no where near as easy as the previous example. (If $a_1 = a_2 = 1$, then $h[n]$ is a Fibonacci sequence.)

- **Usefulness of the difference equation**

(i) The difference equation is useful for computing the output only when the input signal is one-sided and the initial rest condition is assumed.

(ii) What is the output when the input is two-sided infinite?
If the filter is *stable*, then we can compute the output by convolving the input with the impulse response $h[n]$:

$$y[n] = x[n] * h[n]$$

If the filter is *unstable*, then the infinite convolution sum might not be defined (might not converge, might not exist). In such cases, there's no way to compute the response to the two-sided infinite sequence.

- **Frequency Response Function**

(i) The frequency response function is difficult to determine from the impulse response via the usual summation formula because the impulse response sequence is difficult to compute, and because even if we have an expression for the impulse response sequence, computing frequency response involves an infinite sum. We'd like and we'll find a better approach.

(ii) Unstable IIR filters don't even have frequency response functions.

(iii) We will later use z-domain analysis to show that for a stable IIR filter, the response to a suddenly applied sinusoid is a signal that is asymptotically sinusoidal. More specifically, the output is a sinusoid plus a decaying "transient" signal. For unstable filters the output in response to a suddenly applied sinusoid is a sinusoid plus a transient signal that does not decay to zero.

1.2 Determining the system function $H(z)$ directly from difference equation

This is another big payoff from z-domain analysis.

(a) Once we've found the system function, we'll be able to find the frequency response (assuming the filter is stable).

(b) We'll be able to check whether the system is stable.

(c) We'll see how to design filters to have desired magnitude frequency responses.

Procedure:

(i) To determine $H(z)$, we'll use the following form of the z-domain I/O law:

$$H(z) = \frac{Y(z)}{X(z)}.$$

(ii) Rewrite the difference equation as follows:

$$y[n] - a_1 y[n-1] - a_2 y[n-2] - \cdots - a_N y[n-N] = b_0 x[n] + b_1 x[n-2] + \cdots + b_M x[n-M].$$

(iii) Take the z-transform of both sides, using the linearity and time delay properties.

$$Y(z) - a_1 Y(z) z^{-1} - \cdots - a_N Y(z) z^{-N} = b_0 X(z) + b_1 X(z) z^{-1} + \cdots + b_M X(z) z^{-M}.$$

(iv) Collect terms:

$$Y(z)\big(1 - a_1 z^{-1} - a_2 z^{-2} - \cdots - a_N z^{-N}\big) = X(z)\big(b_0 + b_1 z^{-1} + \cdots + b_M z^{-M}\big).$$

(v) The system function is then

$$\begin{aligned} H(z) = \frac{Y(z)}{X(z)} &= \frac{b_0 + b_1 z^{-1} + \cdots + b_M z^{-M}}{1 - a_1 z^{-1} - \cdots - a_N z^{-N}} \\ &= \frac{\big(b_0 + b_1 z^{-1} + \cdots + b_M z^{-M}\big) z^M / z^M}{\big(1 - a_1 z^{-1} - \cdots - a_N z^{-N}\big) z^N / z^N} \\ &= \frac{\big(b_0 z^M + \cdots + b_{M-1} z + b_M\big) z^N}{\big(z^N - a_1 z^{N-1} - \cdots - a_N\big) z^M} \end{aligned}$$

We see that $H(z)$ is a *rational function* of z.

One should either memorize this formula (the relationship of $H(z)$ to the coefficients), or one should be ready to repeat the derivation when one needs to determine the system function for a given a difference equation.

Examples of finding $H(z)$ from a difference equation

(i) Example of finding $H(z)$ from a difference equation.

$$y[n] = y[n-1] + y[n-2] + x[n] \qquad \Longrightarrow \; H(z) = \frac{1}{1 - z^{-1} - z^{-2}}.$$

(ii) Example of finding the filter coefficients from an $H(z)$.

$$H(z) = \frac{1 + 0.8z^{-1}}{1 - 0.9z^{-1}} \qquad \Longrightarrow \; y[n] = 0.9y[n-1] + x[n] + 0.8x[n-1].$$