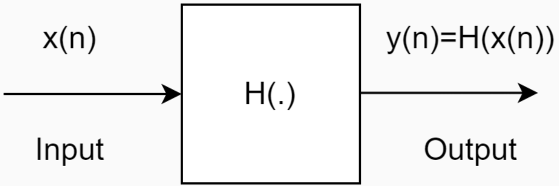
Discrete-time systems

Alberto Carini

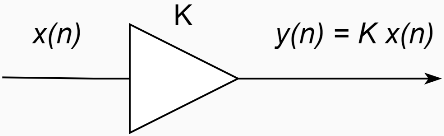
Department of Engineering and Architecture, University of Trieste, Trieste, Italy

- The function of a discrete-time system is to process one or more sequences, referred to as **input sequences,** with the aim of generating one or more sequences, known as **output sequences.**
- These output sequences are expected to exhibit certain desired properties or emphasize specific information from the input signals.
- In most cases, our systems have a single input and a single output.

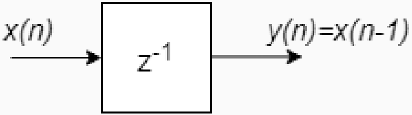


- In discrete-time systems of practical interest, all signals are digital signals (with discrete-time and discrete amplitude), and the operations on these signals also result in digital signals.
- Such systems are commonly referred to as **digital filters.** Throughout this discussion, we will interchangeably use the terms **discrete-time system**, **discrete system**, and **digital filter**.
- The term 'filter' originates from these systems' initial application in filtering the spectrum of a signal. The system was designed to leave certain frequency components of the signal unaltered while removing, or 'filtering,' other undesired frequencies—similarly to a mechanical filter.

- **Constant multiplier:**



- **Unit delay:**



$$y(n) = \sum_{l=-\infty}^{n} x(l)$$

The input-output relationship can also be expressed alternatively as:

$$y(n) = \sum_{l=-\infty}^{n-1} x(l) + x(n) = y(n-1) + x(n).$$

Another alternative form is the following:

$$y(n) = \sum_{l=-\infty}^{-1} x(l) + \sum_{l=0}^{n} x(l) = y(-1) + \sum_{l=0}^{n} x(l).$$

This form is used when the input signal $x(n)$ is a causal signal (i.e., defined only for $n \geq 0$) and $y(-1)$ is called **initial condition**.

$$y(n) = \frac{1}{M} \sum_{l=0}^{M-1} x(n-l)$$

It's worth noting that the expression for the moving average can be expressed in recursive form as follows:

$$\begin{aligned} y(n) &= \frac{1}{M} \left(\sum_{l=0}^{M-1} x(n-l) + x(n-M) - x(n-M) \right) = \\ &= \frac{1}{M} \left(\sum_{l=1}^{M} x(n-l) + x(n) - x(n-M) \right) = \\ &= y(n-1) + \frac{1}{M} \left(x(n) - x(n-M) \right). \end{aligned}$$

It's possible to describe the same system in different ways, corresponding to various implementations.

Further, we will observe that the moving average filter behaves like a low-pass filter, with a passband inversely proportional to M (larger M results in a lower passband).

$$y(n) = \alpha y(n-1) + x(n)$$

with $0 < \alpha < 1$.
This filter calculates the mean of past signal samples, with a greater emphasis on the most recent samples of $x(n)$.
Through successive substitutions, we find that

$$y(n) = \sum_{l=0}^{+\infty} \alpha^{n-l} x(n-l).$$

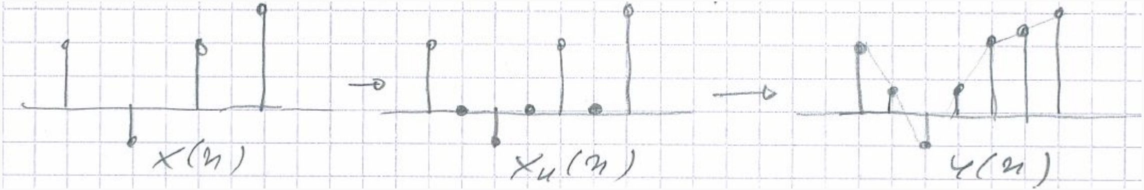
Here, the samples are multiplied by an exponential weight that gradually diminishes as we move away from $x(n)$.

Suppose we have a signal sampled at a frequency of f_c.
To obtain the samples of the same signal at a sampling frequency of $2f_c$, we can

- take the sequence sampled at f_c,
- insert a zero between each pair of samples, and then
- filter the resulting sequence with an interpolation filter.

The interpolation filter replaces all zero samples with the mean value of the preceding and succeeding samples:

$$y(n) = x_u(n) + \frac{1}{2}\left(x_u(n-1) + x_u(n+1)\right)$$



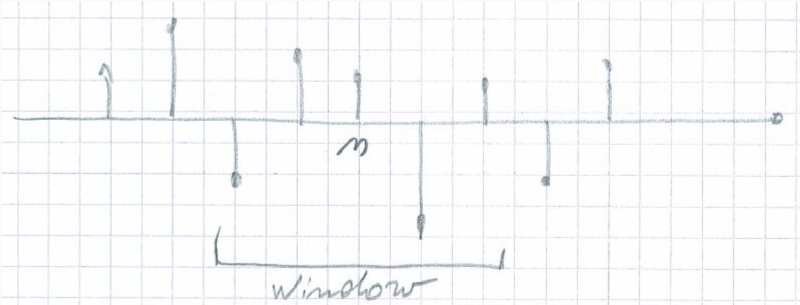
The technique can be easily extended for interpolation factors of 3, 4, or even higher.
For an interpolation factor of 3, the formula becomes:

$$y(n) = x_u(n) + \frac{2}{3}\left(x_u(n-1) + x_u(n+1)\right) + \frac{1}{3}\left(x_u(n-2) + x_u(n+2)\right)$$

These filters find applications in image processing, particularly for enlarging images. For example, they are used to transition from an image with $N \times N$ pixels to an enlarged image with $2N \times 2N$ pixels.

Consider a set of $2K+1$ numbers. Ordering these numbers by their values, the **median** is the number at the central position, precisely at position K when counted from 0.
Therefore, there are K numbers lower than or equal to the median and K numbers greater than or equal to the median.
The median filter is created by sliding a window of length $2K+1$ over the signal $x(n)$ and selecting the median value within this window:

$$y(n) = \text{med}\left\{x(n-K), x(n-K+1), \dots, x(n-1), x(n), x(n+1), \dots, x(n+K)\right\}.$$



If the signal has a finite length, it is extended with zeros in both directions.

$$\{\ldots, 0, 1, 2, 1, 0, \ldots\} \overset{\text{med}_3}{\longleftrightarrow} \{\ldots, 0, 1, 1, 1, 0, \ldots\}$$

The median filter is widely employed in image processing to eliminate impulsive noises.
Notably, it possesses the property of preserving edges, a characteristic that contrasts with the smoothing effect on borders when using low-pass filters like the moving average.

- A discrete-time system is termed **static** or **without memory** if, for every input sequence $\{x(n)\}$ and at every time instant n, the output $y(n)$ depends solely on the input sample at that time, $x(n)$. It does not depend on past or future output samples.
  An example of a static system is the multiplier for a constant.
- In contrast, a discrete-time system, where the output signal depends on both past and future input samples, is termed **dynamic**.
- A discrete-time system is termed **linear** if it satisfies the **superposition principle**:
  for any pair of input signals $x_1(n)$ and $x_2(n)$, and for any arbitrary constants a_1 and a_2, if $y_1(n)$ and $y_2(n)$ are the responses to $x_1(n)$ and $x_2(n)$, then the response to the input signal $x(n) = a_1 x_1(n) + a_2 x_2(n)$ is $a_1 y_1(n) + a_2 y_2(n)$.

$$x_1(n) \longrightarrow y_1(n)$$

$$x_2(n) \longrightarrow y_2(n)$$

$$a_1 x_1(n) + a_2 x_2(n) \longrightarrow a_1 y_1(n) + a_2 y_2(n)$$

- The superposition principle can be separated into two parts:
  - **Multiplicative property:** It the response to $x(n)$ is $y(n)$, then for all constants K the response to $Kx(n)$ is $Ky(n)$:

$$x(n) \longrightarrow y(n)$$

$$Kx(n) \longrightarrow Ky(n)$$

  - **Additive property:** If the responses to $x_1(n)$ and $x_2(n)$ are $y_1(n)$ and $y_2(n)$, respectively, then the response to $x_1(n) + x_2(n)$ is $y_1(n) + y_2(n)$:

$$x_1(n) \longrightarrow y_1(n)$$

$$x_2(n) \longrightarrow y_2(n)$$

$$x_1(n) + x_2(n) \longrightarrow y_1(n) + y_2(n)$$

- Every system that does not satisfy the superposition principle is called **nonlinear**.

- Let us first consider the accumulator:

$$y_1(n) = \sum_{m=-\infty}^{n} x_1(m)$$

$$y_2(n) = \sum_{m=-\infty}^{n} x_2(m)$$

The response to $a_1 x_1(n) + a_2 x_2(n)$ is

$$y(n) = \sum_{m=-\infty}^{n} \left(a_1 x_1(m) + a_2 x_2(m)\right) =$$

$$= a_1 \sum_{m=-\infty}^{n} x_1(m) + a_2 \sum_{m=-\infty}^{n} x_2(m) =$$

$$= a_1 y_1(n) + a_2 y_2(n)$$

Thus, the accumulator in this form is linear.

- Let us now consider the alternative form of the accumulator:

$$y_1(n) = y_1(-1) + \sum_{m=0}^{n} x_1(m)$$

$$y_2(n) = y_2(-1) + \sum_{m=0}^{n} x_2(m)$$

The response to $a_1 x_1(n) + a_2 x_2(n)$ is

$$y(n) = y(-1) + \sum_{m=0}^{n} (a_1 x_1(m) + a_2 x_2(m)) = y(-1) + a_1 \sum_{m=0}^{n} x_1(m) + a_2 \sum_{m=0}^{n} x_2(m)$$

On the contrary, we have

$$a_1 y_1(n) + a_2 y_2(n) = a_1 y_1(-1) + a_2 y_2(-1) + a_1 \sum_{m=0}^{n} x_1(m) + a_2 \sum_{m=0}^{n} x_2(m)$$

The two expressions are equal if and only if:

$$a_1 y_1(-1) + a_2 y_2(-1) = y(-1).$$

- The two expressions are equal if and only if:

$$a_1 y_1(-1) + a_2 y_2(-1) = y(-1).$$

- The condition must be satisfied for all a_1, a_2, $x_1(n)$, $x_2(n)$, and for all $y_1(-1)$, $y_2(-1)$, $y(-1)$.
- Since $y_1(-1)$, $y_2(-1)$, $y(-1)$ are initialization constants, this condition is not generally satisfied unless we assume the system to be **initially at rest**, i.e., with $y_1(-1) = y_2(-1) = y(-1) = 0$.
- If the system has **zero initial conditions**, it is **linear**.
- Conversely, if it has an **initial condition different from zero**, it is a **nonlinear** system.

- Another example of nonlinear system is the median filter.
- Let us consider a median filter of length 3.

$$\{x_1(n)\} = \{3, 4, 5\} \longrightarrow \{y_1(n)\} = \{3, 4, 4\}$$

$$\{x_2(n)\} = \{2, -1, -1\} \longrightarrow \{y_2(n)\} = \{0, -1, -1\}$$

$$\{x_1(n)\} + \{x_2(n)\} = \{5, 3, 4\} \longrightarrow \{y_1(n)\} = \{3, 4, 3\}$$

But $\{3, 4, 3\} \neq \{y_1(n)\} + \{y_2(n)\} = \{3, 3, 3\}$.

- A system is termed **time-invariant** or **shift-invariant** if, for any input $x(n)$ with a response $y(n)$ and for any constant $k \in \mathbb{Z}$, the response to $x(n-k)$ is $y(n-k)$:

$$x(n) \longrightarrow y(n)$$

$$x(n-k) \longrightarrow y(n-k)$$

Note that this property must hold for every possible choice of $x(n)$ and k.

- In the following, we will particularly focus on **Linear Time-Invariant (LTI)** discrete-time systems.
- LTI systems exhibit both linearity and time-invariance properties.
- These characteristics make them straightforward to analyze and characterize, facilitating easy design. Consequently, LTI systems find widespread use in processing digital signals.
- For these systems, we can explicitly express the rule $H(\cdot)$ that maps the input signal to the output signal.
- In other words, for LTI systems, we can formulate a mathematical rule that computes the output signal samples based on the knowledge of the input signal samples.
- The concepts of impulse response and convolution sum play a crucial role in this context.

- The **impulse response** of a LTI system is defined as the system's response to the unit impulse input signal:

$$x(n) = \delta(n) \longrightarrow y(n) = h(n)$$

- We have observed that every sequence $x(n)$ can be represented as the sum of an infinite number of impulses, appropriately scaled:

$$x(n) = \sum_{m=-\infty}^{+\infty} x(m)\delta(n-m)$$

But

$$\delta(n) \longrightarrow h(n)$$

$$\delta(n-m) \longrightarrow h(n-m)$$

$$x(m)\delta(n-m) \longrightarrow x(m)h(n-m)$$

$$\sum_{m=-\infty}^{+\infty} x(m)\delta(n-m) \longrightarrow \sum_{m=-\infty}^{+\infty} x(m)h(n-m)$$

- In an LTI system, the output sequence can be calculated from the input sequence using the following relation:

$$\boxed{y(n) = \sum_{m=-\infty}^{+\infty} x(m)h(n-m) = x(n) \circledast h(n)}$$

  This sum is known as the **convolution sum**. We also say that the signal $x(n)$ is **convolved** with $h(n)$.

- The impulse response is sufficient to completely describe LTI systems. Knowing $h(n)$ allows us to determine $y(n)$ for any input $x(n)$.

- **Commutative property**:

$$x(n) \circledast h(n) = h(n) \circledast x(n)$$

- Proof:

$$y(n) = \sum_{m=-\infty}^{+\infty} x(m)h(n-m)$$

Let us consider the change of variable $m' = n - m$, i.e., $m = n - m'$

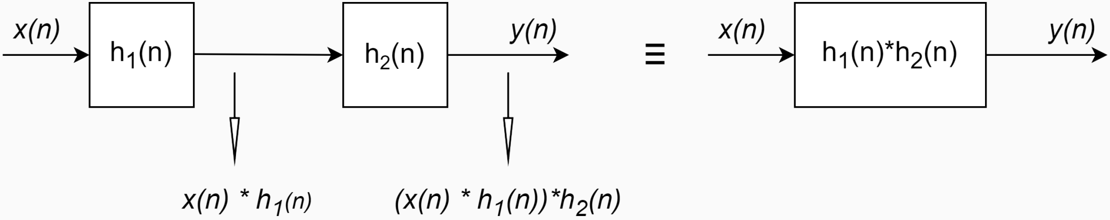
$$y(n) = \sum_{m'=-\infty}^{+\infty} x(n-m')h(m') = h(n) \circledast x(n)$$

- Physical interpretation:
The system with input $x(n)$ and impulse response $h(n)$ has the same response $y(n)$ as the system with input $h(n)$ and impulse response $x(n)$:

- **Associative property**:

$$[x(n) \circledast h_1(n)] \circledast h_2(n) = x(n) \circledast [h_1(n) \circledast h_2(n)]$$

- Physical interpretation:
  If we consider the cascade of two systems with impulse responses $h_1(n)$ and $h_2(n)$, respectively, the resulting system has an impulse response given by $h_1(n) \circledast h_2(n)$:



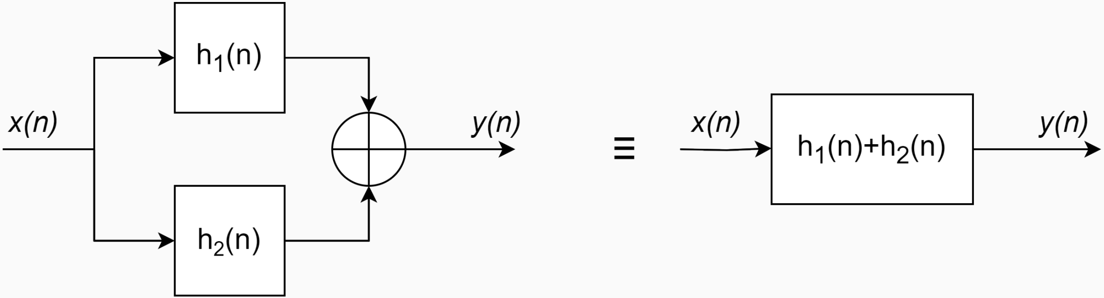
- **Distributive property**:

$$[x_1(n) + x_2(n)] \circledast h(n) = x_1(n) \circledast h(n) + x_2(n) \circledast h(n)$$

and also

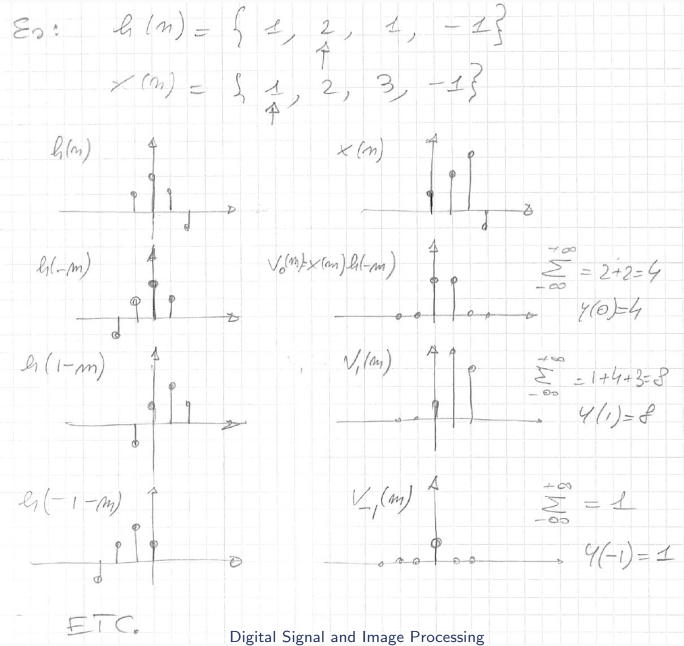
$$x(n) \circledast [h_1(n) + h_2(n)] = x(n) \circledast h_1(n) + x(n) \circledast h_2(n)$$

- Physical interpretation of the last relation:
If we consider the parallel connection of two systems with impulse responses $h_1(n)$ and $h_2(n)$, respectively, the resulting system has an impulse response given by $h_1(n) + h_2(n)$:



$$y(n) = \sum_{m=-\infty}^{+\infty} x(m)h(n-m)$$

- We can compute $y(n_0) = y(n)\Big|_{n=n_0}$ by means of the following operations:
    - Fold the sequences $h(m)$ to obtain $h(-m)$.
    - Time-shift $h(-m)$ of n_0 samples to the right for $n_0 > 0$ (time delay), or to the left by $|n_0|$ samples for $n_0 < 0$ (time advancement), resulting in $h(n_0 - m)$.
    - Perform element-wise multiplication between $x(m)$ and $h(n_0 - m)$ to get $V_{n_0}(m) = x(m)h(n_0 - m)$.
    - Sum of all the terms of $V_{n_0}(m)$.
- Conceptually, this method can be applied to any pair of sequences $x(n)$ and $h(n)$. However, in practice, this approach is feasible only when at least one of the two sequences has a finite length.


Es: h(n) = { 1, 2, 1, −1 }
x(n) = { 1, 2, 3, −1 }
h(n)
x(n)
h(−m)
$V_0(m) = x(m)h(-m)$
$\sum_{-\infty}^{+\infty} = 2+2=4$
y(0)=4
h(1−m)
$V_1(m)$
$\sum_{-\infty}^{+\infty} = 1+4+3=8$
y(1)=8
h(−1−m)
$V_{-1}(m)$
$\sum_{-\infty}^{+\infty} = 1$
y(−1)=1
ETC.

- A discrete-time system is termed **causal** if, at every time instant n, the output $y(n)$ depends solely on the present and past samples of $x(n)$ (i.e., $x(n)$, $x(n-1)$, $x(n-2)$, etc.), while it remains independent of the future samples of the signal ($x(n+1)$, $x(n+2)$, $x(n+3)$, etc.).
- Systems that are not causal are referred to as **noncausal**.
- In real-time digital signal processing systems, the observation of future samples of the signal is not possible, rendering noncausal systems unrealizable. Hence, the property of causality is also known as the **realizability property**.

- **Property:** An LTI system is causal if and only if its impulse response is zero for $n < 0$:

$$\boxed{\text{causality} \Longleftrightarrow h(n) = 0 \qquad \forall n < 0.}$$

- *Proof:*

$$y(n) = \sum_{m=-\infty}^{+\infty} h(m) \cdot x(n-m) =$$

$$= \sum_{m=-\infty}^{-1} h(m) \cdot x(n-m) + \sum_{m=0}^{+\infty} h(m) \cdot x(n-m)$$

The first term depends on the future samples of $x(n)$, while the second term depends only on the present and past samples of $x(n)$. Thus, the system is causal if and only if $h(n) = 0 \ \forall n < 0$.

Q.E.D.

- An example of a noncausal system is the linear interpolator.

- In a causal system, the convolution sum is given by

$$y(n) = \sum_{m=0}^{+\infty} h(m)x(n-m)$$

$$= \sum_{m=-\infty}^{n} x(m)h(n-m)$$

- In analogy to the causality property of LTI systems, sequences that are zero for all $n < 0$ are referred to as **causal**.
- If the input of a causal LTI system is a causal sequence, the boundaries of the convolution sum are further reduced since we have:

$$y(n) = \sum_{m=0}^{n} h(m)x(n-m)$$

$$= \sum_{m=0}^{n} x(m)h(n-m)$$

- A discrete-time system is defined **BIBO stable** (Bounded Input Bounded Output stable) if, for every bounded input signal $x(n)$, the output signal $y(n)$ is bounded.
- If the input signal is bounded, there exists a constant M_x such that

$$|x(n)| \leq M_x < +\infty \qquad \forall n.$$

  If the system is BIBO stable, there must exist a constant M_y such that, for any $|x(n)| \leq M_x$,

$$|y(n)| \leq M_y < +\infty \qquad \forall n.$$

- **Property:** A LTI discrete-time system is BIBO stable if and only if

$$\boxed{\sum_{n=-\infty}^{+\infty} |h(n)| < +\infty,}$$

  i.e., if and only if the impulse response is absolutely summable.

UNIVERSITÀ DEGLI STUDI DI TRIESTE

- **Proof:** First, let's prove that this condition is sufficient for BIBO stability. If $x(n)$ is bounded, there exists M_x such that

$$|x(n)| \leq M_x < +\infty \qquad \forall n.$$

Thus,

$$|y(n)| = \left| \sum_{m=-\infty}^{+\infty} h(m)x(n-m) \right| \leq$$

$$\leq \sum_{m=-\infty}^{+\infty} |h(m)x(n-m)| \leq$$

$$\leq \sum_{m=-\infty}^{+\infty} |h(m)|\, M_x$$

If we define $M_y = M_x \sum_{m=-\infty}^{+\infty} |h(m)|$, it is proved that, if $h(n)$ is absolutely summable, there exists a constant M_y such that

$$|y(n)| \leq M_y < +\infty \qquad \forall n.$$

- **Proof:**
  First, let's prove that this condition is sufficient for BIBO stability.
  If $x(n)$ is bounded, there exists M_x such that

$$|x(n)| \leq M_x < +\infty \qquad \forall n.$$

Thus,

$$|y(n)| = \left| \sum_{m=-\infty}^{+\infty} h(m)x(n-m) \right| \leq$$

$$\leq \sum_{m=-\infty}^{+\infty} |h(m)x(n-m)| \leq$$

$$\leq \sum_{m=-\infty}^{+\infty} |h(m)|\, M_x$$

If we define $M_y = M_x \sum_{m=-\infty}^{+\infty} |h(m)|$, it is proved that, if $h(n)$ is absolutely summable, there exists a constant M_y such that

$$|y(n)| \leq M_y < +\infty \qquad \forall n.$$

- Now, let's prove that it is also a necessary condition.
  To this purpose, let us assume
  $$\sum_{m=-\infty}^{+\infty} |h(m)| = +\infty$$
  and let us demonstrate that it is always possible to find a bounded input whose output is not bounded.
- If $h(n) \in \mathbb{R}$, one of such signals is
  $$x(n) = \text{sign}[h(-n)] = \begin{cases} +1 & h(-n) \geq 0 \\ -1 & h(-n) < 0 \end{cases}$$
  Surely, $x(n)$ is bounded since $|x(n)| = 1$. If we consider the output of our system for $n = 0$:
  $$y(0) = \sum_{m=-\infty}^{+\infty} h(m)x(0-m) = \sum_{m=-\infty}^{+\infty} h(m)\text{sign}[h(m)] = \sum_{m=-\infty}^{+\infty} |h(m)| = +\infty$$
- Here, for simplicity, we have considered a real $h(n)$. However, everything holds true for a complex $h(n)$ as well. It is sufficient to consider
  $$x(n) = \frac{h^*(-n)}{|h(-n)|}.$$

- An LTI discrete-time system is termed a **Finite Impulse Response (FIR)** system if its impulse response has a finite length, i.e.

$$h(n) = 0 \qquad \forall n < N_1 \text{ or } n > N_2,$$

with $N_1 \leq N_2$. $h(n)$ has only $N = N_2 - N_1 + 1$ elements different from 0.

- In this case, the convolution sum simplifies to

$$y(n) = \sum_{m=N_1}^{N_2} h(m)x(n-m).$$

- As this sum is finite, it can be directly used to compute $y(n)$.
- For a causal FIR system of length N:

$$h(n) = 0 \qquad \forall n < 0 \text{ or } n \geq N,$$

and the output is given by

$$\boxed{y(n) = \sum_{m=0}^{N-1} h(m)x(n-m).}$$

- A system whose impulse response has infinite length is referred to as an **Infinite Impulse Response (IIR)** system.
- In the case of a causal IIR system, the output is given by

$$y(n) = \sum_{m=0}^{+\infty} h(m)x(n-m).$$

- While FIR systems can be directly implemented using the convolution sum, IIR systems cannot be realized through the convolution sum due to the requirement of an infinite number of additions, multiplications, and memory elements.
- In practice, in digital signal processing, we often focus on a subclass of LTI and causal discrete-time systems.
- This subclass comprises all systems that can be represented by a **finite difference equation with constant coefficients**, i.e., they can be expressed in the following form:

$$\boxed{y(n) = \sum_{i=0}^{M} b_i x(n-i) - \sum_{i=1}^{N} a_i y(n-i)}$$

for all $n \geq 0$.

$$y(n) = \sum_{i=0}^{M} b_i x(n-i) - \sum_{i=1}^{N} a_i y(n-i)$$

- For this class of systems, the output can be computed directly from some past samples of the input and output signals.
- These systems are causal, as they involve only the past samples and the present sample of the input signal.
- To compute $y(n)$ from the input signal $x(n)$, knowledge of the N initial conditions, $y(-1)$, $y(-2)$, ..., $y(-N)$, is required.
- If these N initial conditions are all zero,

$$y(-1) = y(-2) = \ldots = y(-N) = 0,$$

  the system is termed **initially at rest.**
- The class of systems that can be described by a finite difference equation with constant coefficients includes all causal FIR systems and a subset of causal LTI IIR systems.

- There are causal IIR systems that cannot be described by a finite difference equation.
- For instance, the system with impulse response:

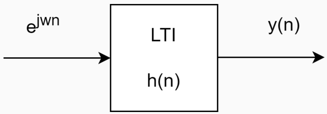
$$h(n) = \begin{cases} \dfrac{1}{n^2} & n > 0 \\ 0 & n \leq 0 \end{cases}$$

is a causal, BIBO-stable system, but it lacks a representation in terms of a finite difference equation.

- We have seen that every sequence can be represented in the time domain as the weighted sum of an infinite number of unit pulses, shifted in time:

$$x(n) = \sum_{m=-\infty}^{+\infty} x(m)\delta(n-m).$$

- This representation leads to an important consequence – the characterization of LTI systems using the impulse response and the convolution sum.
- We have also seen that sequences can be represented by means of a weighted sum of an infinite number of complex exponential sequences $\{e^{j\omega n}\}$.
- This representation leads to another description of LTI systems through the so-called **Frequency Response**.
- Let us consider an LTI system with impulse response $h(n)$ and let us excite the system with a complex exponential sequence $e^{j\omega n}$ with $-\infty < n < +\infty$.



$$y(n) = \sum_{m=-\infty}^{+\infty} h(m) e^{j\omega(n-m)} = \sum_{m=-\infty}^{+\infty} h(m) e^{-j\omega m} e^{j\omega n} = H(e^{j\omega}) \cdot e^{j\omega n}.$$

- If we apply a complex exponential sequence $e^{j\omega n}$ to the input of our system, the output is the same exponential sequence multiplied by the complex constant $H(e^{j\omega})$.
- $H(e^{j\omega})$ is the Discrete-Time Fourier Transform (DTFT) of the impulse response $h(n)$ and is referred to as the **Frequency Response** of the LTI system.
- $|H(e^{j\omega})|$ is termed the **amplitude response** (or **magnitude response**), and $\arg\{H(e^{j\omega})\}$ is termed the **phase response** of the LTI system.

- The frequency response completely characterizes the response of an LTI system in the frequency domain.
- Indeed,

$$y(n) = x(n) \circledast h(n) \overset{DTFT}{\longleftrightarrow} Y(e^{j\omega}) = H(e^{j\omega})X(e^{j\omega})$$

- *Proof:*

$$\begin{aligned} Y(e^{j\omega}) &= \sum_{n=-\infty}^{+\infty} y(n)e^{-j\omega n} = \\ &= \sum_{n=-\infty}^{+\infty} \left(\sum_{m=-\infty}^{+\infty} h(m)x(n-m) \right) e^{-j\omega n} = \\ &= \sum_{m=-\infty}^{+\infty} h(m) \sum_{n=-\infty}^{+\infty} x(n-m)e^{-j\omega(n-m)}e^{-j\omega m} = \\ &= \sum_{m=-\infty}^{+\infty} h(m)X(e^{j\omega})e^{-j\omega m} = \\ &= X(e^{j\omega}) \sum_{m=-\infty}^{+\infty} h(m)e^{-j\omega m} = \\ &= X(e^{j\omega})H(e^{j\omega}) \end{aligned}$$

Q.E.D.

UNIVERSITÀ DEGLI STUDI DI TRIESTE

- The Discrete-Time Fourier Transform (DTFT) transforms the convolution sum of two sequences into the product of their respective DTFTs.
- If we know the frequency response of an LTI system, we can calculate the output sequence, denoted as $y(n)$ and representing the response to the input sequence $x(n)$, through the following steps:
  1. $X(e^{j\omega}) = \text{DTFT}\ \{x(n)\}$
  2. $Y(e^{j\omega}) = X(e^{j\omega})H(e^{j\omega})$
  3. $y(n) = \text{IDTFT}\ \{Y(e^{j\omega})\}$
- The main inconvenience is represented by the fact that the IDTFT requires the computation of the integral of a continuous function.
- We will address this inconvenience later by introducing the Discrete Fourier Transform (DFT).

- An application of LTI systems is to allow certain frequency components of a sequence to pass without distortions while blocking any other frequency component.
- Such systems are referred to as **digital filters**.
- For example, let us consider the low pass filter with frequency response:

$$H(e^{j\omega}) \simeq \begin{cases} 1 & 0 \leq |\omega| \leq \omega_c \\ 0 & \omega_c \leq |\omega| \leq \pi \end{cases}$$

Let the system input be

$$x(n) = A\cos(\omega_1 n) + B\cos(\omega_2 n)$$

with $0 < \omega_1 < \omega_c < \omega_2 < \pi$. Since $cos(\omega n) = \frac{1}{2}[e^{j\omega n} + e^{-j\omega n}]$, it can be easily proved that:

$$y(n) \simeq A|H(e^{j\omega_1})|cos\left(\omega_1 n + \arg\{H(e^{j\omega_1}\}\right) \simeq Acos(\omega_1 n).$$

Thus, the output comprises only the first cosine component, which lies within the passband of the filter, while the second component outside the passband is eliminated.

- Let us consider the system

$$\begin{cases} y(n) = ay(n-1) + x(n) \\ y(-1) = 0. \end{cases}$$

We aim to calculate the frequency response of this system.
If $x(n) = \delta(n)$ then it is easy to observe through successive substitutions that

$$y(n) = h(n) = a^n \qquad \forall n \geq 0.$$

$$H(e^{j\omega}) = \sum_{n=0}^{+\infty} a^n e^{-j\omega n} = \sum_{n=0}^{+\infty} \left(ae^{-j\omega}\right)^n = \frac{1}{1 - ae^{-j\omega}},$$

provided that $|a| < 1$.

$$|H(e^{j\omega})| = \frac{1}{|1 - a\cos(\omega) + ja\sin(\omega)|} = \frac{1}{\sqrt{(1 - a\cos(\omega))^2 + a^2\sin^2(\omega)}} = \frac{1}{\sqrt{1 - 2a\cos(\omega) + a^2}}$$

$$\arg\left\{H(e^{j\omega})\right\} = \arg\left\{\frac{e^{j\omega}}{e^{j\omega} - a}\right\} = \omega - \arctan\left(\frac{\sin(\omega)}{\cos(\omega) - a}\right).$$

When $a > 0$, the filter exhibits a low-pass behavior, while for $a < 0$, it demonstrates a high-pass behavior.
This is a first-order system as it involves only one delayed sample of the output.
With first-order systems, we can implement either low-pass or high-pass filters.

- For more information study:

S. K. Mitra, "Digital Signal Processing: a computer based approach," 4th edition, McGraw-Hill, 2011
Chapter 4.1-4.5, pp. 141-163
Chapter 4.7.1, pp. 173-174
Chapter 4.8.1-4.8.2, pp. 175-177

Unless otherwise specified, all images have either been originally produced or have been taken from S. K. Mitra, "Digital Signal Processing: a computer based approach."