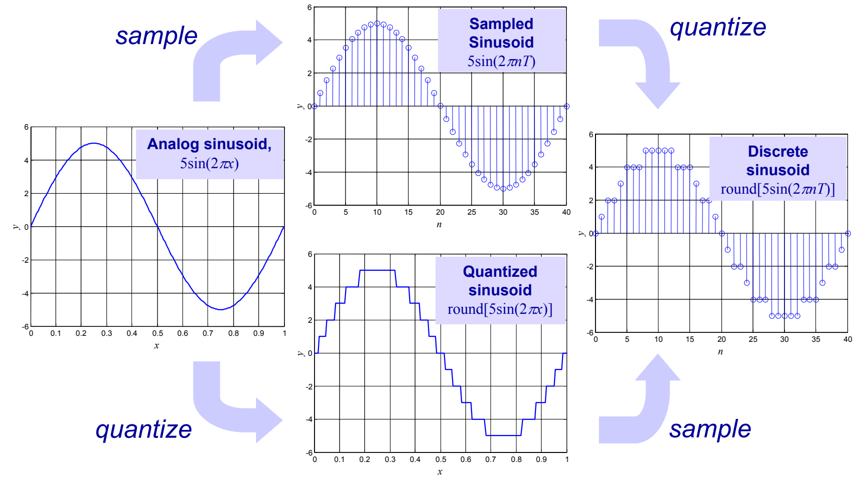
ECE 4270

Fundamentals of Digital Signal Processing

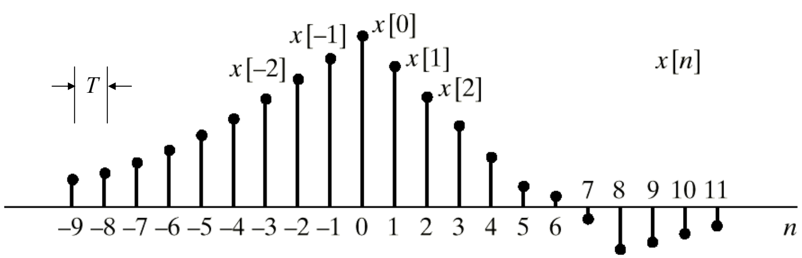
Lecture 2: Discrete-time Signals & Systems, & Properties of LTI Systems

School of Electrical and Computer Engineering
Georgia Institute of Technology
Summer 2004

Discrete Signals



Discrete-Time (DT) Signals are Sequences

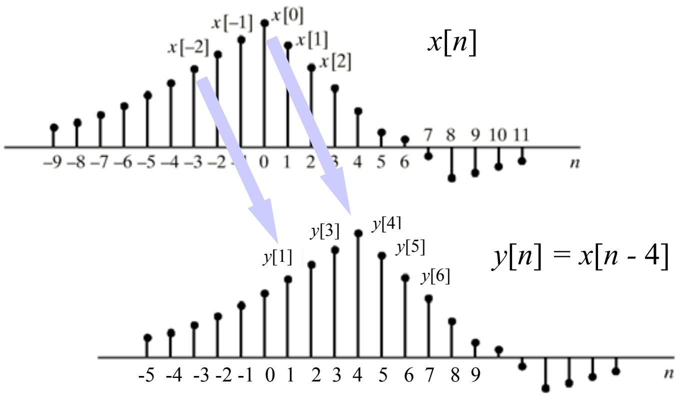


- $x[n]$ denotes the "sequence value at 'time' n"
- Sources of sequences:
  - Sampling a continuous-time signal
    $x[n] = x_c(nT) = x_c(t)|_{t=nT}$
  - Mathematical formulas – generative system
    e.g., $x[n] = 0.3 \bullet x[n\text{-}1]\ \text{-}1; \quad x[0] = 40$

Notation

- Continuous independent variables are enclosed in parentheses:
  $$x(t), \quad X(\omega)$$
- Discrete independent variables are enclosed in square brackets:
  $$x[n], \quad X[k]$$
- Discrete-index signal $x[n]$ – sometimes n may not represent time

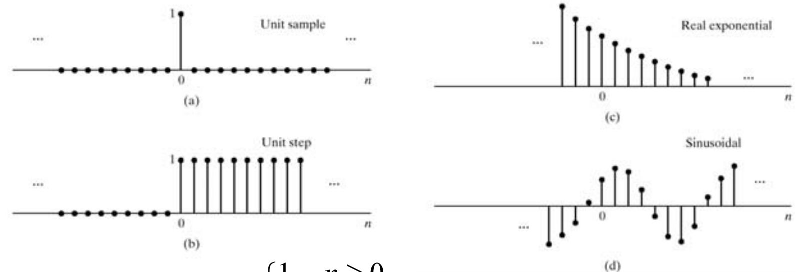
Delay & Shift of Sequence



Useful Sequences

unit sample $\delta[n] = \begin{cases} 1, & n = 0 \\ 0, & n \neq 0 \end{cases}$

real exponential $x[n] = \alpha^n$



unit step $u[n] = \begin{cases} 1, & n \geq 0 \\ 0, & n < 0 \end{cases}$

sine wave $x[n] = A\cos(\omega_0 n + \phi)$

Complex DT Sinusoid

$$x[n] = Ae^{j\omega n}$$

- Frequency ω is in radians (per sample), or just radians
  - not radians per second
  - because “time” index n is dimensionless
  - once sampled, $x[n]$ is a sequence that **relates to time only through sampling period** T
- Important property: periodic in ω with period 2π:

$$Ae^{j\omega_0 n} = Ae^{j(\omega_0 + 2\pi r)n}$$

  - Only unique frequencies are 0 to 2π (or $-\pi$ to $+\pi$)
  - Same applies to real sinusoids

Aside: Euler’s Formulae

$$e^{\pm j\theta} = \cos\theta \pm j\sin\theta$$

$$\cos\theta = \frac{1}{2}\left\{e^{+j\theta} + e^{-j\theta}\right\}$$

$$\sin\theta = \frac{1}{2j}\left\{e^{+j\theta} - e^{-j\theta}\right\}$$

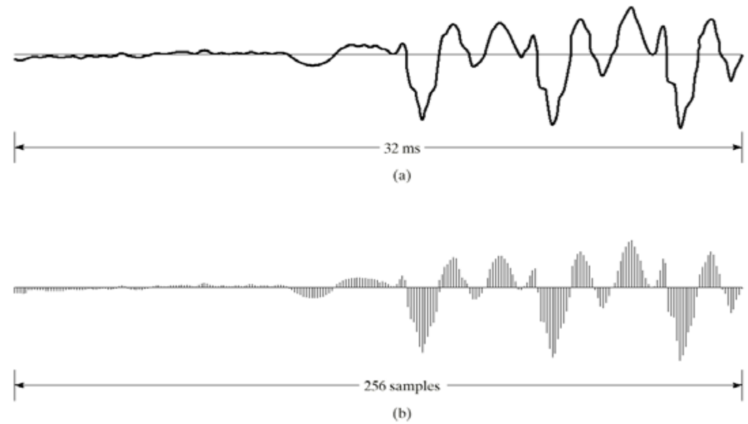
Periodic DT Signals

- A signal is periodic with period N if $x[n] = x[n+N]$ for all n
- For the complex exponential this condition becomes

$$Ae^{j\omega_0 n} = Ae^{j(\omega_0 n + \omega_0 N)}$$

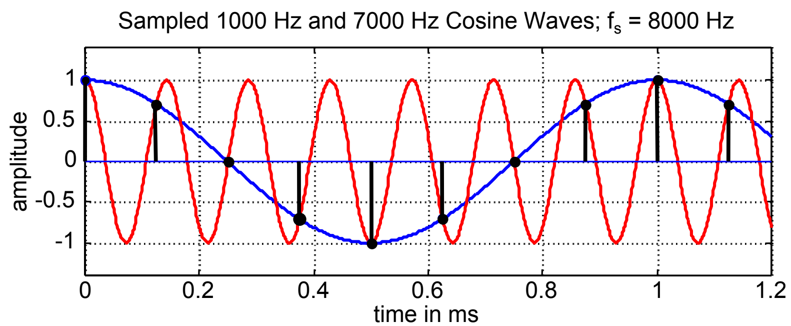
  which requires $\omega_0 N = 2\pi k$ for some integer k
- Thus, not all DT **sinusoids** are periodic!
- Consequence: there are N distinguishable frequencies with period N
  - *e.g.*, $\omega_k = 2\pi k/N,\ k=0,1,\ldots,N\text{-}1$

Sampled Speech Waveform



Sampling period (interval) is T = 0.125 msec

The Sampling Theorem



- A bandlimited signal can be reconstructed exactly from samples taken with sampling frequency

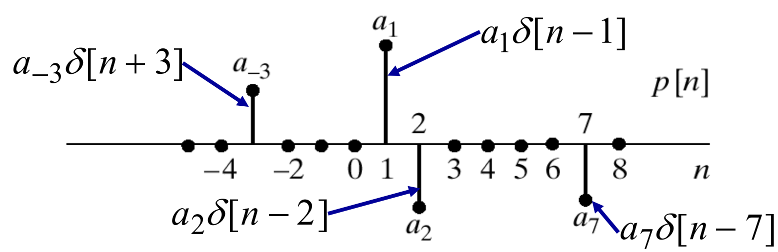
$$\frac{1}{T} = f_s \ge 2 f_{\max} \quad \text{or} \quad \frac{2\pi}{T} = W_s \ge 2W_{\max}$$

Impulse Representation of Sequences

A sequence, a function

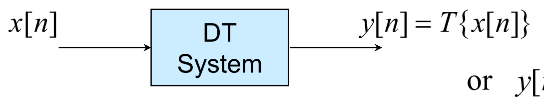
$$x[n] = \sum_{k=-\infty}^{\infty} x[k]\delta[n-k]$$

Value of the function at k



$$p[n] = a_{-3}\delta[n+3] + a_1\delta[n-1] + a_2\delta[n-2] + a_7\delta[n-7]$$

Discrete-Time Systems



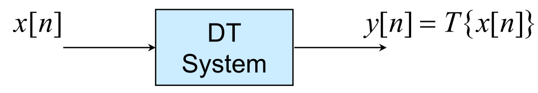
or $y[n] = T\{\{x[n]\}\}$

A system transforms an input into an output.

- Delay: $x[n] \mapsto y[n] = x[n - n_d]$
- Modulator: $x[n] \mapsto y[n] = x[n]\cos(\omega_0 n)$
- Squarer: $x[n] \mapsto y[n] = (x[n])^2$
- Compressor: $y[n] = x[Mn]$ (aka downsampler)
- Expander: (aka upsampler)

$$x[n] \mapsto y[n] = \begin{cases} x[n/L], & n = 0, \pm L, \ldots \\ 0, & \text{otherwise} \end{cases}$$

More Discrete-Time Systems



- L-point moving average system:

$$\begin{aligned} y[n] &= \frac{1}{L}\sum_{k=0}^{L-1} x[n-k] \\ &= \frac{1}{L}(x[n] + x[n-1] + \cdots + x[n-L+1]) \end{aligned}$$

- Accumulator: $y[n] = \sum_{k=-\infty}^{n} x[k]$

Properties of DT Systems: I

- A system is linear if and only if

$$T\{ax_1[n] + bx_2[n]\} = aT\{x_1[n]\} + bT\{x_2[n]\}$$

- A system is time-invariant if and only if

$$x_1[n] = x[n - n_d] \quad \Rightarrow \quad y_1[n] = y[n - n_d]$$

- A system is causal if and only if

$y[n]$ depends only on $x[k]$ for $k \le n$

Properties of DT Systems: II

- A system is BIBO stable if every bounded input produces a bounded output; *i.e.*,

when $|x[n]| < B_x < \infty \ \forall\ n$, then $|y[n]| < B_y < \infty \ \forall\ n$

- A system is memoryless if

$y[n]$ depends only on $x[n]$ at the same value of n

Moving Averager: $y[n] = (1/L)\sum_{k=0}^{L-1} x[n-k]$

- Linear? *Yes*

$$\frac{1}{L}\sum_{k=0}^{L-1}(ax_1[n-k]+bx_2[n-k]) = a\left(\frac{1}{L}\sum_{k=0}^{L-1}x_1[n-k]\right)+b\left(\frac{1}{L}\sum_{k=0}^{L-1}x_2[n-k]\right)$$

- Time-invariant? *Yes*

$$\frac{1}{L}\sum_{k=0}^{L-1}x[n-k-n_d] = \frac{1}{L}\sum_{k=0}^{L-1}x[(n-n_d)-k] = y[n-n_d]$$

- Causal? *Yes* $\quad y[n] = \frac{1}{L}(x[n]+x[n-1]+\cdots+x[n-L+1])$

- Stable? *Yes* $\quad |y[n]| = \left|\frac{1}{L}\sum_{k=0}^{L-1}x[n-k]\right| \le \frac{1}{L}\sum_{k=0}^{L-1}|x[n-k]| \le B_x$

Down Sampler: $y[n] = x[Mn]$

- Linear? *Yes*

$$ax_1[Mn]+bx_2[Mn] = ay_1[n]+by_2[n]$$

- Time-invariant? *No*

$$y_1[n] = x_1[Mn] = x[Mn-n_d] \ne y[n-n_d] = x[M(n-n_d)]$$

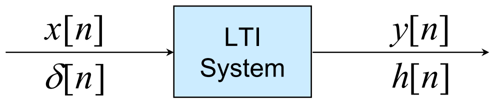
- Causal? *No*

$$y[-1] = x[-M], \text{ but } y[+1] = x[M]$$

- Stable? *Yes*

$$|y[n]| = |x[Mn]| \le B_x$$

LTI Discrete-Time Systems



- Linearity (superposition):

$$\mathrm{T}\{ax_1[n]+bx_2[n]\} = a\mathrm{T}\{x_1[n]\}+b\mathrm{T}\{x_2[n]\}$$

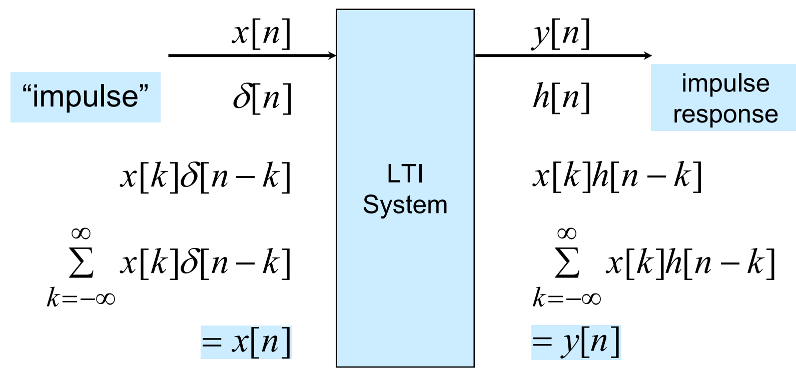
- Time-Invariance (shift-invariance):

$$x_1[n] = x[n-n_d] \quad \Rightarrow \quad y_1[n] = y[n-n_d]$$

- LTI implies discrete convolution:

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k] = x[n]*h[n] = h[n]*x[n]$$

LTI Discrete-Time Systems



convolution sum $\quad y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k] = x[n]*h[n] = h[n]*x[n]$

Discrete Convolution - I

- Definition

$$y[n]=\sum_{k=-\infty}^{\infty}x[k]h[n-k]=\sum_{k=-\infty}^{\infty}h[k]x[n-k]$$

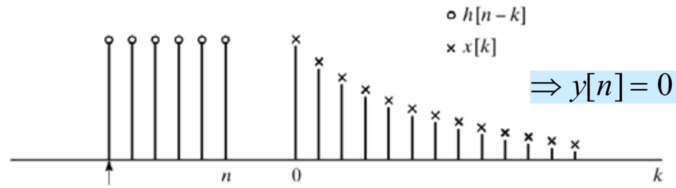
- Example

$$h[n]=\begin{cases}1, & 0\le n\le 5\\ 0, & \text{otherwise}\end{cases}$$

$$y[n]=\sum_{k=n-5}^{n}x[k]=\sum_{k=0}^{5}x[n-k]$$

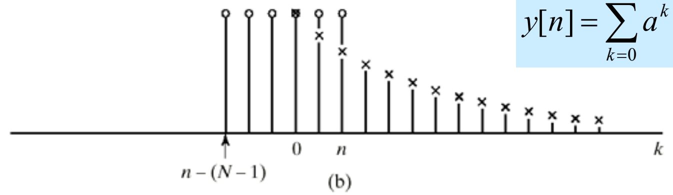
$$y[n]=x[n]+x[n-1]+\ldots+x[n-5]$$

Discrete Convolution - II



(a)

$$\Rightarrow y[n]=0 \quad \text{for } n<0$$



(b)

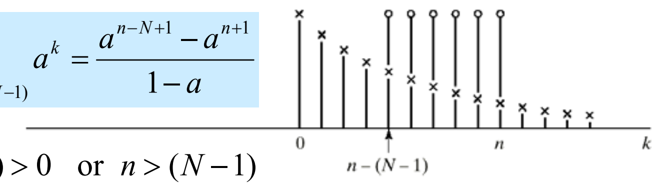
$$y[n]=\sum_{k=0}^{n}a^k=\frac{1-a^{n+1}}{1-a}$$

$$n-(N-1)\le 0 \text{ and } n\ge 0 \quad \text{or} \quad 0\le n\le (N-1)$$

Discrete Convolution - III

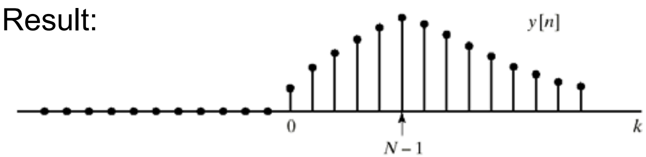
$$y[n]=\sum_{k=n-(N-1)}^{n}a^k=\frac{a^{n-N+1}-a^{n+1}}{1-a}$$

$$n-(N-1)>0 \quad \text{or} \quad n>(N-1)$$



(c)

Result:



(d)

Discrete Convolution

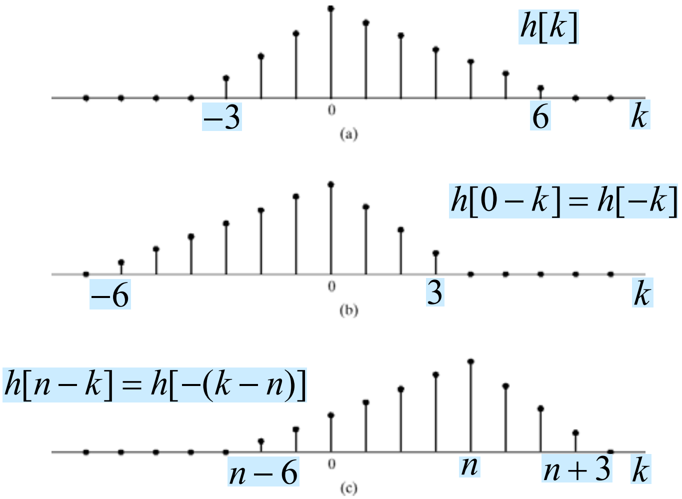
- Two ways to look at it:
  - As the representation of the output as a sum of delayed and scaled impulse responses.

$$y[n]=\sum_{k=-\infty}^{\infty}x[k]h[n-k]=x[0]h[n]+x[1]h[n-1]+\ldots$$

$$\ldots+x[-1]h[n+1]+\ldots$$

  - As a computational formula for computing $y[n]$ ("y at time n") from the entire sequences x and h.
    - Form $x[k]h[n\text{-}k]$ for $-\infty<k<+\infty$ for a fixed n
    - Sum over all k to produce $y[n]$
    - Repeat for all n

"Flipping and Shifting"



Examples

- Delay: $y[n]=x[n-n_d]$

$$h[n]=\delta[n-n_d] \Rightarrow x[n]*\delta[n-n_d]=x[n-n_d]$$

- Accumulator: $y[n]=\sum_{k=-\infty}^{n} x[k]$

$$h[n]=\sum_{k=-\infty}^{n} \delta[k]=\begin{cases}0 & n<0\\ 1 & n\geq 0\end{cases}=u[n]$$

- First difference: $y[n]=x[n]-x[n-1]$

$$h[n]=\delta[n]-\delta[n-1]$$

Stability of LTI Systems

- Stability: Every bounded input produces a bounded output (BIBO).

$$|y[n]|=\left|\sum_{k=-\infty}^{\infty} h[k]x[n-k]\right| \leq \sum_{k=-\infty}^{\infty} |h[k]||x[n-k]|$$

$$|y[n]| \leq \sum_{k=-\infty}^{\infty} |h[k]| B_x$$

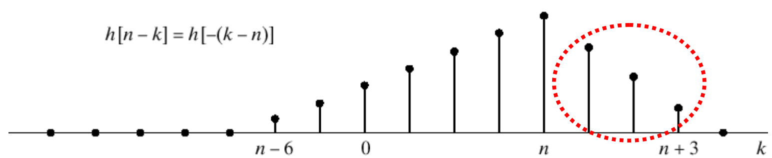
$$\text{Therefore, } |y[n]|<\infty \text{ if } \sum_{k=-\infty}^{\infty} |h[k]|<\infty$$

- This condition can also be shown to be necessary as well as sufficient for BIBO stability.

Causality of LTI Systems

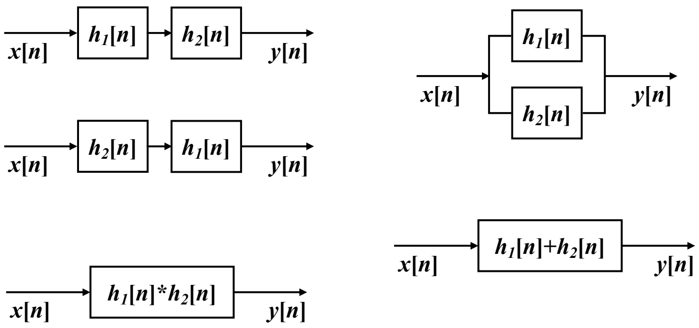
- A system is causal if $y[n]$ depends only on $x[k]$ for k less than or equal to n.

$$y[n]=\sum_{k=-\infty}^{\infty} x[k]h[n-k] \Rightarrow h[n-k]=0 \text{ for } k>n$$



$$\text{Causality requires } h[n]=0 \text{ for } n<0$$

Equivalent LTI Systems



$h_1[n]*h_2[n]= h_2[n]*h_1[n]$ $h_1[n]+h_2[n]= h_2[n]+h_1[n]$

Difference Equations

- For all computationally realizable LTI systems, the input and output satisfy a difference equation of the form

$$\sum_{k=0}^{N} a_k y[n-k] = \sum_{k=0}^{M} b_k x[n-k]$$

- This leads to the recurrence formula

$$y[n] = -\sum_{k=1}^{N}\left(\frac{a_k}{a_0}\right) y[n-k] + \sum_{k=0}^{M}\left(\frac{b_k}{a_0}\right) x[n-k]$$

which can be used to compute the "present" output from the present and M past values of the input and N past values of the output

Homogenous Equation

A linear system: $\sum_{k=0}^{N} a_k y[n-k] = \sum_{k=0}^{M} b_k x[n-k]$

Its output for a given input is not uniquely specified without additional constraints.

Suppose: $\sum_{k=0}^{N} a_k y_h[n-k] = 0$ ⟵ Zero-input (free, natural, or homogeneous) response; result of initial condition of the system

Then, $\sum_{k=0}^{N} a_k \left(y[n-k] + y_h[n-k]\right) = \sum_{k=0}^{M} b_k x[n-k]$

Total response
= forced (zero-state) response $y[n]$
+ homogeneous (zero-input) response $y_h[n]$

Homogeneous Solution

Solution to $\sum_{k=0}^{N} a_k y_h[n-k] = 0$ has the form:

$$y_h[n] = \sum_{m=1}^{N} A_m z_m^n$$

$$\sum_{k=0}^{N} a_k \sum_{m=1}^{N} A_m z_m^{n-k} = 0 \quad \Rightarrow \quad \sum_{m=1}^{N} A_m z_m^n \sum_{k=0}^{N} a_k z_m^{-k} = 0$$

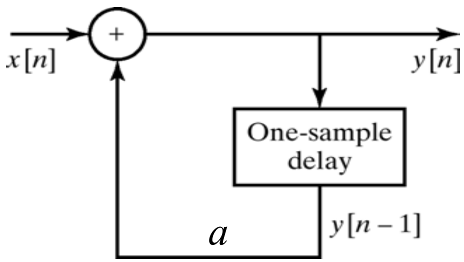
z_m must be roots of polynomial $\sum_{k=0}^{N} a_k z_m^{-k}$

First-Order Example

- Consider the difference equation

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

- We can represent this system by the following block diagram:



Recursive Computation of Output

Let $x[n] = K\delta[n]$ and $y[-1] = c$.

n	$x[n]$	$y[n]=ay[n-1]+x[n]$
…	…	…
-1	0	c
0	K	$ac + K$
1	0	$a(ac+K) = a^2c + Ka$
2	0	$a(a^2c+Ka) = a^3c + Ka^2$
3	0	$a(a^3c+Ka^2) = a^4c + Ka^3$

General Solution

- By induction, we see that if

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

$$\text{with } x[n] = K\delta[n] \text{ and } y[-1] = c.$$

then the solution is

$$y[n] = ca^{n+1} + Ka^n u[n] \quad \text{for all } n$$

- If $x[n] = K\delta[n-n_d]$, then the output will be

$$y[n] = ca^{n+1} + Ka^{n-n_d} u[n-n_d]$$

← Implies *not* TI

- If $x[n] = 0$ then the output will be

$$y[n] = ca^{n+1}$$

← Implies *not* linear

LTI Recursive Implementation

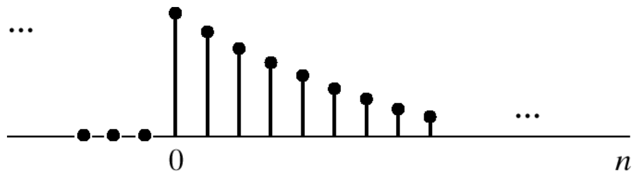
- We say that an input is *suddenly applied at time n_d* if $x[n]=0$ for all $n<n_d$.
- If the input is suddenly applied and we assume that $y[n]=0$ for all $n<n_d$, then the iterative computation will be both linear and time-invariant. This assumption provides the set of auxiliary conditions $\{y[n_d-1], y[n_d-2], , \ldots, y[n_d-N]\}$ that is required to get the recursion going. Identically zero auxiliary conditions are called *initial rest conditions* (zero-state).

Exponential Impulse Response

- With initial rest conditions, the difference equation

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

has impulse response $h[n] = a^n u[n]$



Equivalence to First-Order DE

- If we assume initial rest conditions, then the difference equation

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

has impulse response $h[n] = a^n u[n]$

- In other words, $y[n]$ is also given by the convolution

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]a^{n-k}u[n-k] = \sum_{k=-\infty}^{n} x[k]a^{n-k}$$

IIR Systems

- Under conditions of initial rest, a system whose input and output satisfy a difference equation of the form

$$y[n] = \underbrace{\sum_{k=1}^{N} a_k y[n-k]}_{\text{feedback}} + \underbrace{\sum_{k=0}^{M} b_k x[n-k]}_{\text{feedforward}}$$

Note the redefinition of the coefficients.

is LTI and its impulse response is of the form

$$h[n] = \sum_{k=1}^{N} A_k \alpha_k^n u[n] = \begin{cases} A_k \alpha_k^n, & n \geq 0 \\ 0, & n < 0 \end{cases}$$

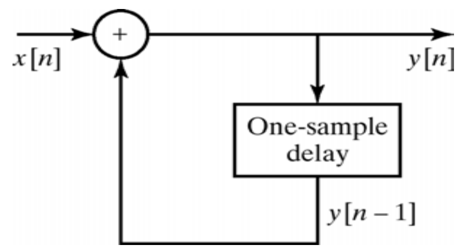
$\Rightarrow$ Infinite duration Impulse Response (IIR)

Accumulator System

$$y[n] = \sum_{k=-\infty}^{n} x[k] = x[n] * u[n]$$

$$y[n] = \underbrace{\left(\sum_{k=-\infty}^{n-1} x[k] \right)}_{y[n-1]} + x[n] = y[n-1] + x[n]$$

$h[n] = u[n]$



FIR Systems

- If there are no feedback terms, then the difference equation becomes

$$y[n] = \sum_{k=0}^{M} b_k x[n-k]$$

- This system is LTI and its impulse response is

$$h[n] = \sum_{k=0}^{M} b_k \delta[n-k] = \begin{cases} b_n, & 0 \le n \le M \\ 0, & \text{otherwise} \end{cases}$$

$\Rightarrow$ Finite duration Impulse Response (FIR)

Moving Average System

$$y[n] = \frac{1}{M_2+1} \sum_{k=0}^{M_2} x[n-k]$$

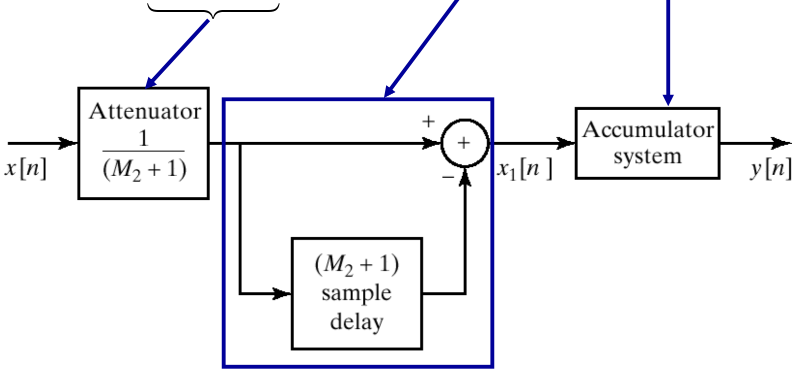
$$h[n] = \frac{1}{M_2+1} \sum_{k=0}^{M_2} \delta[n-k] = \frac{1}{M_2+1} \begin{cases} 1 & 0 \le n \le M_2 \\ 0 & \text{otherwise} \end{cases}$$

$$= \frac{1}{M_2+1}(u[n] - u[n-M_2-1])$$

$$= \frac{1}{M_2+1}(\delta[n] - \delta[n-M_2-1]) * u[n]$$

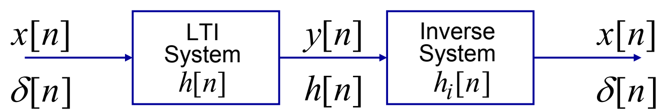
Moving Average System

$$h[n] = \left(\frac{1}{M_2+1}\right) \bullet (\delta[n] - \delta[n-M_2-1]) * u[n]$$



Inverse Systems

- An inverse system compensates (undoes) the effects of another system.



$$\Rightarrow h[n] * h_i[n] = \delta[n]$$

- The accumulator and first-difference systems are inverses of each other.

$$(\delta[n] - \delta[n-1]) * u[n] = u[n] - u[n-1] = \delta[n]$$

- Understanding inverse systems is greatly facilitated by transform methods.

MATLAB and LTI Systems

»help conv
CONV Convolution and polynomial multiplication.
Y = CONV(X, H) convolves vectors X and H. The resulting vector is length LENGTH(X)+LENGTH(H)-1.
If X and H are vectors of polynomial coefficients, convolving them is equivalent to multiplying the two polynomials.

»help filter
FILTER One-dimensional digital filter.
Y = FILTER(B,A,X) filters the data in vector X with the filter described by vectors A and B to create the filtered data Y. The filter is a "Direct Form II Transposed" implementation of the standard difference equation:

MATLAB and LTI Systems

- The moving average system

  \>> h=ones(1,M+1)/(M+1);

  \>> y=conv(x,h);

- The accumulator system:

  \>> b=1; a=[1,-1];

  \>> y=filter(b,a,x);