

## ESE 531: Digital Signal Processing

Lec 10: February 19, 2019  
Non-Integer and Multi-rate Sampling



Penn ESE 531 Spring 2019 - Khanna

## Lecture Outline

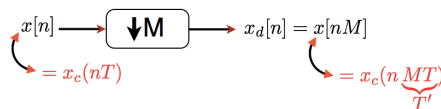
- Review: Downsampling/Upsampling
- Interpolation
- Non-integer Resampling
- Multi-Rate Processing
  - Interchanging Operations

Penn ESE 531 Spring 2019 - Khanna

2

## Downsampling

- Definition: Reducing the sampling rate by an integer number



$$X_d(e^{j\omega}) = \frac{1}{M} \sum_{i=0}^{M-1} X(e^{j(\frac{\omega}{M} - \frac{2\pi}{M}i)})$$

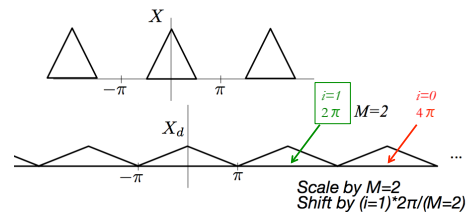
stretch by M      replicate

Penn ESE 531 Spring 2019 - Khanna

3

## Example

$$X_d(e^{j\omega}) = \frac{1}{M} \sum_{i=0}^{M-1} X(e^{j(\frac{\omega}{M} - \frac{2\pi}{M}i)})$$

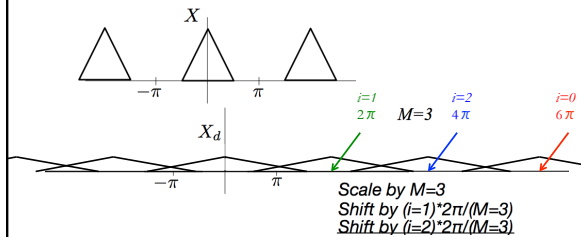


Penn ESE 531 Spring 2019 - Khanna

4

## Example

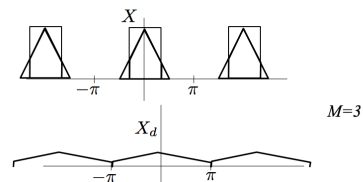
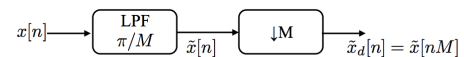
$$X_d(e^{j\omega}) = \frac{1}{M} \sum_{i=0}^{M-1} X(e^{j(\frac{\omega}{M} - \frac{2\pi}{M}i)})$$



Penn ESE 531 Spring 2019 - Khanna

5

## Example



Penn ESE 531 Spring 2019 - Khanna

6

## Upsampling

- Definition: Increasing the sampling rate by an integer number

$$x[n] = x_c(nT)$$

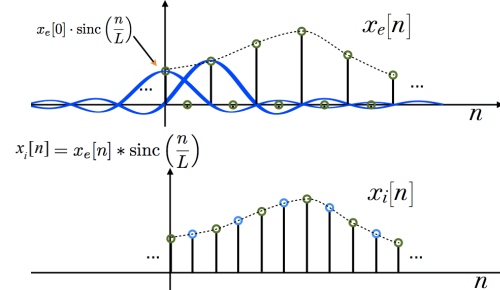
$$x_i[n] = x_c(nT') \quad \text{where } T' = \frac{T}{L} \quad L \text{ integer}$$

Obtain  $x_i[n]$  from  $x[n]$  in two steps:

$$(1) \text{ Generate: } x_e[n] = \begin{cases} x[n/L] & n = 0, \pm L, \pm 2L, \dots \\ 0 & \text{otherwise} \end{cases}$$

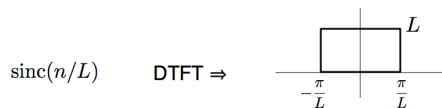
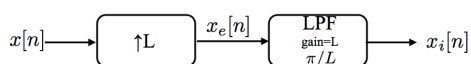
## Upsampling

(2) Obtain  $x_i[n]$  from  $x_e[n]$  by bandlimited interpolation:

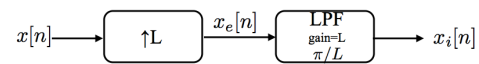


## Frequency Domain Interpretation

$$x_i[n] = x_e[n] * \text{sinc}(n/L)$$



## Frequency Domain Interpretation

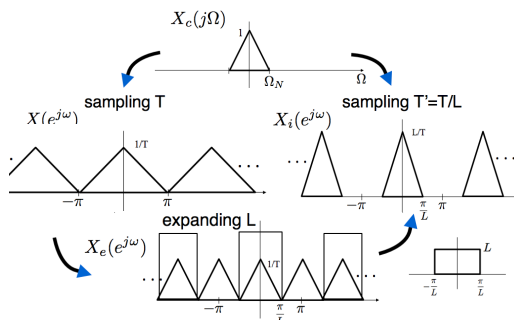
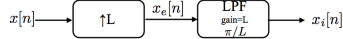


$$\begin{aligned} X_e(e^{j\omega}) &= \sum_{n=-\infty}^{\infty} x_e[n] e^{-j\omega n} \\ &= \sum_{m=-\infty}^{\infty} x_e[mL] e^{-j\omega mL} = X(e^{j\omega L}) \end{aligned}$$

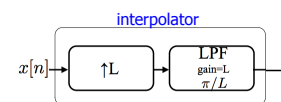
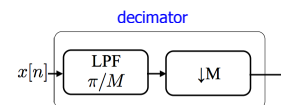
(Note: The original image has a red box around  $X(e^{j\omega L})$  in the final equation.)

Compress DTFT by a factor of L!

## Example

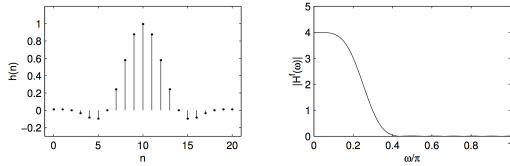


## Interpolation and Decimation



### Interpolation Filter Example 1

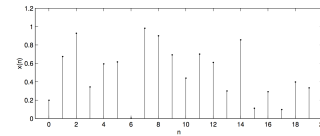
- In this example, we interpolate a signal  $x(n)$  by a factor of 4.
- We use a linear phase Type I FIR lowpass filter of length 21.



Penn ESE 531 Spring 2019 - Khanna

13

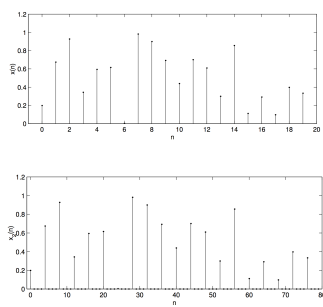
### Interpolation Filter Example 1



Penn ESE 531 Spring 2019 - Khanna

14

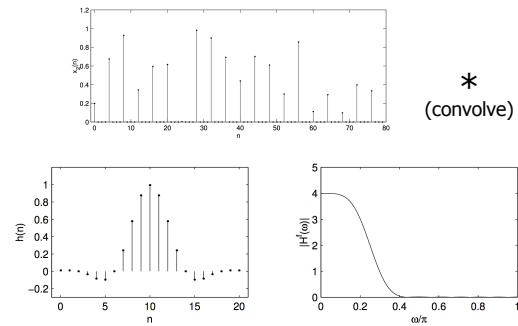
### Interpolation Filter Example 1



Penn ESE 531 Spring 2019 - Khanna

15

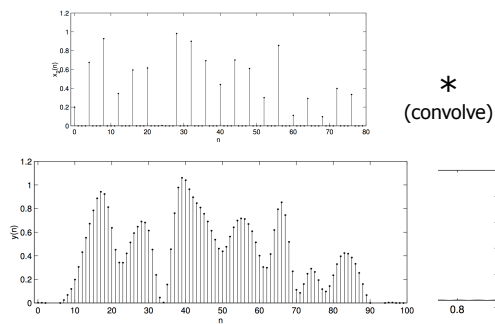
### Interpolation Filter Example 1



Penn ESE 531 Spring 2019 - Khanna

16

### Interpolation Filter Example 1

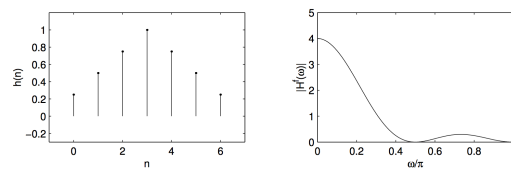


Penn ESE 531 Spring 2019 - Khanna

17

### Interpolation Filter Example 2

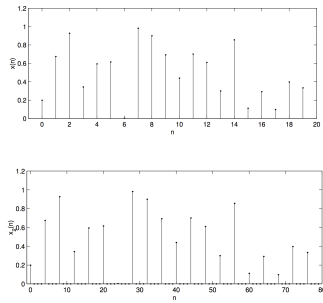
- This time we use a filter of length 7 with the effect of linear interpolation



Penn ESE 531 Spring 2019 - Khanna

18

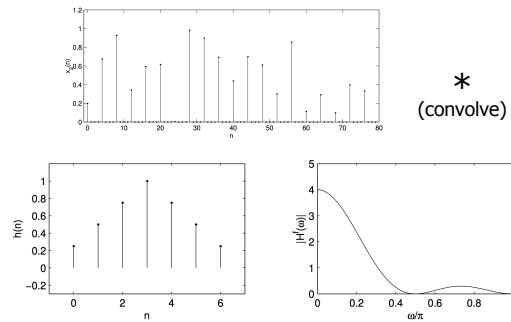
### Interpolation Filter Example 2



Penn ESE 531 Spring 2019 - Khanna

19

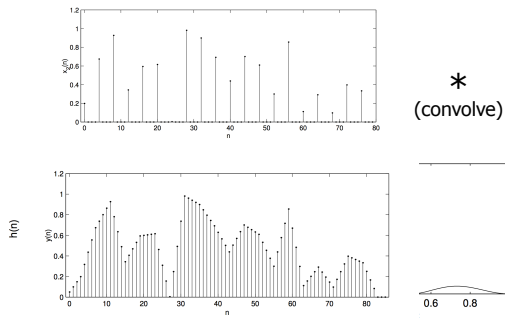
### Interpolation Filter Example 2



Penn ESE 531 Spring 2019 - Khanna

20

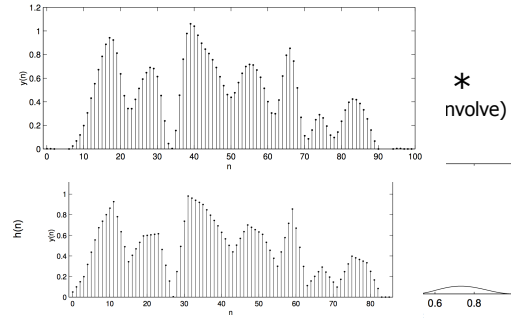
### Interpolation Filter Example 2



Penn ESE 531 Spring 2019 - Khanna

21

### Interpolation Filter Example 2



Penn ESE 531 Spring 2019 - Khanna

22

### Interpolation Filter Example 3

- When interpolating a signal  $x(n)$ , the interpolation filter  $h(n)$  will in general change the samples of  $x(n)$  in addition to filling in the zeros.
- Can a filter be designed so as to preserve the original samples  $x(n)$ ?

Penn ESE 531 Spring 2019 - Khanna

23

### Interpolation Filter Example 3

- When interpolating a signal  $x(n)$ , the interpolation filter  $h(n)$  will in general change the samples of  $x(n)$  in addition to filling in the zeros.
- Can a filter be designed so as to preserve the original samples  $x(n)$ ?
- To be precise, if  $y(n) = h(n) * [\uparrow 2] x(n)$  then can we design  $h(n)$  so that  $y(2n) = x(n)$ ?

Penn ESE 531 Spring 2019 - Khanna

24

### Interpolation Filter Example 3

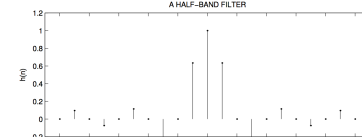
- When interpolating a signal  $x(n)$ , the interpolation filter  $h(n)$  will in general change the samples of  $x(n)$  in addition to filling in the zeros.
- Can a filter be designed so as to preserve the original samples  $x(n)$ ?
- To be precise, if  $y(n) = h(n) * [\uparrow 2] x(n)$  then can we design  $h(n)$  so that  $y(2n) = x(n)$ ?
- Or more generally, so that  $y(2n + n_0) = x(n)$ ?

Penn ESE 531 Spring 2019 - Khanna

25

### Interpolation Filter Example 3

- When interpolating by a factor of 2, if  $h(n)$  is a half-band filter, then it will not change the samples  $x(n)$ .
- A  $n_0$ -centered half-band filter  $h(n)$  is a filter that satisfies:
 
$$h(n) = \begin{cases} 1, & \text{for } n = n_0 \\ 0, & \text{for } n = n_0 \pm 2, 4, 6, \dots \end{cases}$$
- That means, every second value of  $h(n)$  is zero, except for one such value, as shown in the figure.

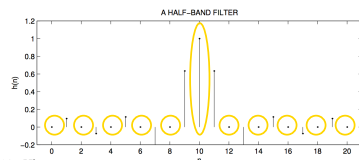


Penn ESE 531 Spring 2019 - Khanna

26

### Interpolation Filter Example 3

- When interpolating by a factor of 2, if  $h(n)$  is a half-band filter, then it will not change the samples  $x(n)$ .
- A  $n_0$ -centered half-band filter  $h(n)$  is a filter that satisfies:
 
$$h(n) = \begin{cases} 1, & \text{for } n = n_0 \\ 0, & \text{for } n = n_0 \pm 2, 4, 6, \dots \end{cases}$$
- That means, every second value of  $h(n)$  is zero, except for one such value, as shown in the figure.

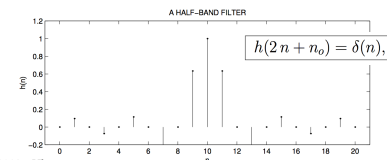


Penn ESE 531 Spring 2019 - Khanna

27

### Interpolation Filter Example 3

- When interpolating by a factor of 2, if  $h(n)$  is a half-band filter, then it will not change the samples  $x(n)$ .
- A  $n_0$ -centered half-band filter  $h(n)$  is a filter that satisfies:
 
$$h(n) = \begin{cases} 1, & \text{for } n = n_0 \\ 0, & \text{for } n = n_0 \pm 2, 4, 6, \dots \end{cases}$$
- That means, every second value of  $h(n)$  is zero, except for one such value, as shown in the figure.



Penn ESE 531 Spring 2019 - Khanna

28

### Interpolation Filter Example 4

- When interpolating a signal  $x(n)$  by a factor  $L$ , the original samples of  $x(n)$  are preserved if  $h(n)$  is a Nyquist-L filter.
- A Nyquist-L filter simply generalizes the notion of the halfband filter to  $L > 2$ .

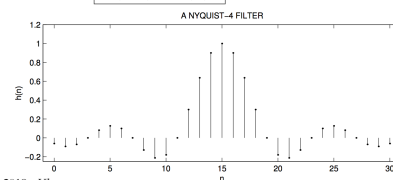
Penn ESE 531 Spring 2019 - Khanna

29

### Interpolation Filter Example 4

- When interpolating a signal  $x(n)$  by a factor  $L$ , the original samples of  $x(n)$  are preserved if  $h(n)$  is a Nyquist-L filter.
- A Nyquist-L filter simply generalizes the notion of the halfband filter to  $L > 2$ .
- A (0-centered) Nyquist-L filter  $h(n)$  is one for which

$$h(Ln) = \delta(n).$$



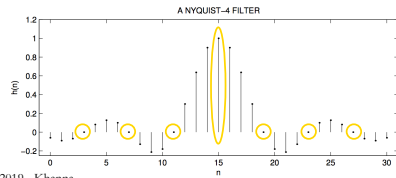
Penn ESE 531 Spring 2019 - Khanna

30

## Interpolation Filter Example 4

- When interpolating a signal  $x(n)$  by a factor  $L$ , the original samples of  $x(n)$  are preserved if  $h(n)$  is a Nyquist- $L$  filter.
- A Nyquist- $L$  filter simply generalizes the notion of the halfband filter to  $L > 2$ .
- A (0-centered) Nyquist- $L$  filter  $h(n)$  is one for which

$$h(Ln) = \delta(n).$$



Penn ESE 531 Spring 2019 - Khanna

31

## Non-integer Resampling



## Non-integer Resampling

- $T' = TM/L$

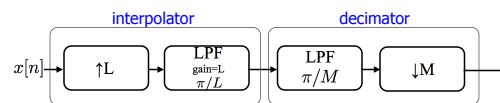
Penn ESE 531 Spring 2019 - Khanna

33

## Non-integer Resampling

- $T' = TM/L$

- Upsample by  $L$ , then downsample by  $M$



Penn ESE 531 Spring 2019 - Khanna

34

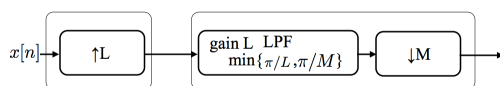
## Non-integer Resampling

- $T' = TM/L$

- Upsample by  $L$ , then downsample by  $M$



Or,

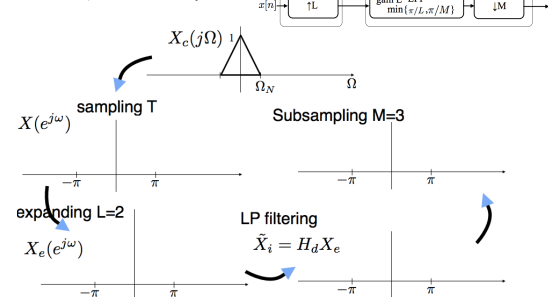


Penn ESE 531 Spring 2019 - Khanna

35

## Example

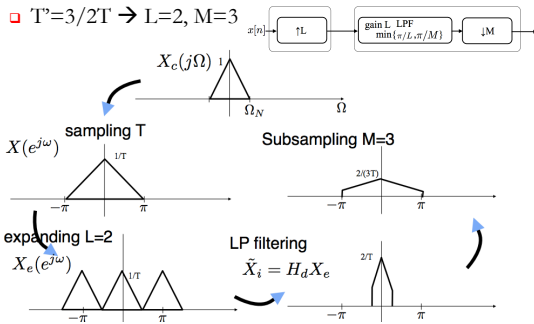
- $T' = 3/2T \rightarrow L=2, M=3$



Penn ESE 531 Spring 2019 - Khanna

36

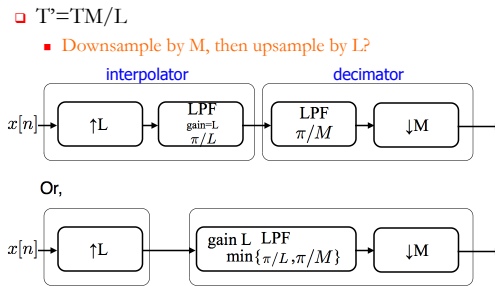
## Example



Penn ESE 531 Spring 2019 - Khanna

37

## Non-integer Sampling



Penn ESE 531 Spring 2019 - Khanna

38

## Example

- What if we want to resample by  $1.01T$ ?

Penn ESE 531 Spring 2019 - Khanna

39

## Example

- What if we want to resample by  $1.01T$ ?
- Upsample by  $L=100$
  - Filter  $\pi/101$
  - Downsample by  $M=101$

Penn ESE 531 Spring 2019 - Khanna

40

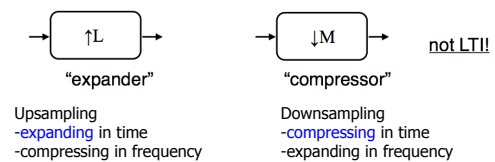
## Example

- What if we want to resample by  $1.01T$ ?
- Upsample by  $L=100$
  - Filter  $\pi/101$  (\$\$\$\$)
  - Downsample by  $M=101$
- Fortunately there are ways around it!
- Called multi-rate signal processing
  - Uses compressors, expanders and filtering

Penn ESE 531 Spring 2019 - Khanna

41

## Interchanging Operations



Penn ESE 531 Spring 2019 - Khanna

42

### Interchanging Operations - Expander

"expander"

Upsampling  
 -expanding in time  
 -compressing in frequency

$x[n] \rightarrow H(z) \rightarrow \uparrow L \rightarrow y[n]$  ?  $x[n] \rightarrow \uparrow L \rightarrow H(z) \rightarrow y[n]$

Penn ESE 531 Spring 2019 - Khanna 43

### Interchanging Operations - Expander

"expander"

Upsampling  
 -expanding in time  
 -compressing in frequency

$x[n] \rightarrow H(z) \rightarrow \uparrow L \rightarrow y[n]$  ?  $x[n] \rightarrow \uparrow L \rightarrow H(z) \rightarrow y[n]$

$H(e^{j\omega})X(e^{j\omega})$   $H(e^{j\omega L})X(e^{j\omega L})$

Penn ESE 531 Spring 2019 - Khanna 44

### Interchanging Operations - Expander

"expander"

Upsampling  
 -expanding in time  
 -compressing in frequency

$x[n] \rightarrow H(z) \rightarrow \uparrow L \rightarrow y[n]$   $\neq$   $x[n] \rightarrow \uparrow L \rightarrow H(z) \rightarrow y[n]$

$H(e^{j\omega})X(e^{j\omega})$   $H(e^{j\omega L})X(e^{j\omega L})$   $X(e^{j\omega L})H(e^{j\omega})X(e^{j\omega L})$

Penn ESE 531 Spring 2019 - Khanna 45

### Interchanging Operations - Expander

"expander"

Upsampling  
 -expanding in time  
 -compressing in frequency

$x[n] \rightarrow H(z) \rightarrow \uparrow L \rightarrow y[n]$   $=$   $x[n] \rightarrow \uparrow L \rightarrow H(z^L) \rightarrow y[n]$

$H(e^{j\omega})X(e^{j\omega})$   $H(e^{j\omega L})X(e^{j\omega L})$   $X(e^{j\omega L})H(e^{j\omega L})X(e^{j\omega L})$

Penn ESE 531 Spring 2019 - Khanna 46

### Interchanging Operations - Compressor

"compressor"

Downsampling  
 -compressing in time  
 -expanding in frequency

$x[n] \rightarrow \downarrow M \rightarrow H(z) \rightarrow y[n]$   $=$   $x[n] \rightarrow H(z^M) \rightarrow \downarrow M \rightarrow \tilde{y}[n]$

$v[n]$

not LTI!!

Penn ESE 531 Spring 2019 - Khanna 47

### Interchanging Operations - Compressor

$x[n] \rightarrow \downarrow M \rightarrow H(z) \rightarrow y[n]$   $=$   $x[n] \rightarrow H(z^M) \rightarrow \downarrow M \rightarrow \tilde{y}[n]$

$v[n]$

$$\begin{aligned}
 Y(e^{j\omega}) &= H(e^{j\omega}) \left( \frac{1}{M} \sum_{i=0}^{M-1} X \left( e^{j \left( \frac{\omega}{M} - \frac{2\pi i}{M} \right)} \right) \right) \\
 &= \frac{1}{M} \sum_{i=0}^{M-1} \underbrace{H \left( e^{j(\omega - 2\pi i)} \right)}_{H(e^{j\omega})} X \left( e^{j \left( \frac{\omega}{M} - \frac{2\pi i}{M} \right)} \right) \\
 &= \frac{1}{M} \sum_{i=0}^{M-1} H \left( e^{jM \left( \frac{\omega}{M} - \frac{2\pi i}{M} \right)} \right) X \left( e^{j \left( \frac{\omega}{M} - \frac{2\pi i}{M} \right)} \right)
 \end{aligned}$$

Penn ESE 531 Spring 2019 - Khanna 48



## Interchanging Operations - Compressor

$$x[n] \rightarrow \downarrow M \rightarrow H(z) \rightarrow y[n] = x[n] \rightarrow H(z^M) \rightarrow \downarrow M \rightarrow \tilde{y}[n]$$

$v[n]$

$$Y(e^{j\omega}) = \frac{1}{M} \sum_{i=0}^{M-1} H(e^{jM(\frac{\omega}{M} - \frac{2\pi i}{M})}) X(e^{j(\frac{\omega}{M} - \frac{2\pi i}{M})})$$

$$V(e^{j\omega}) = H(e^{j\omega M}) X(e^{j\omega})$$

$$\tilde{Y}(e^{j\omega}) = \frac{1}{M} \sum_{i=0}^{M-1} V(e^{j(\frac{\omega}{M} - \frac{2\pi i}{M})})$$

Penn ESE 531 Spring 2019 - Khanna

49

## Interchanging Operations - Summary

**Filter and expander**      **Expander and expanded filter\***

$$x[n] \rightarrow H(z) \rightarrow \uparrow L \rightarrow y[n] \equiv x[n] \rightarrow \uparrow L \rightarrow H(z^L) \rightarrow y[n]$$

**Compressor and filter**      **Expanded filter\* and compressor**

$$x[n] \rightarrow \downarrow M \rightarrow H(z) \rightarrow y[n] \equiv x[n] \rightarrow H(z^M) \rightarrow \downarrow M \rightarrow y[n]$$

\*Expanded filter = expanded impulse response, compressed freq response

Penn ESE 531 Spring 2019 - Khanna

50

## Multi-Rate Signal Processing

- What if we want to resample by 1.01T?
  - Expand by L=100
  - Filter  $\pi/101$  (\$\$\$\$)
  - Compress by M=101
- Fortunately there are ways around it!
  - Called multi-rate
  - Uses compressors, expanders and filtering

Penn ESE 531 Spring 2019 - Khanna

51

## Big Ideas

- Downsampling/Upsampling
- Practical Interpolation
- Non-integer Resampling
- Multi-Rate Processing
  - Interchanging Operations

$$x[n] \rightarrow H(z) \rightarrow \uparrow L \rightarrow y[n] \equiv x[n] \rightarrow \uparrow L \rightarrow H(z^L) \rightarrow y[n]$$

$$x[n] \rightarrow \downarrow M \rightarrow H(z) \rightarrow y[n] \equiv x[n] \rightarrow H(z^M) \rightarrow \downarrow M \rightarrow y[n]$$

Penn ESE 531 Spring 2019 - Khanna

52

## Admin

- HW 4 due Sunday
  - Typo, homework problem 4.28 removed, homework handout fixed.

Penn ESE 531 Spring 2019 - Khanna

53