Digital Systems & Transforms

Professor Marie Roch
San Diego State University

Huang: 5-5.2.1, 5.2.4, 5.3.1, 5.3.5, 5.4.2, 5.5-5.5.2

San Diego State University

Digital Systems

When

- x[n] is an input signal varying with discrete time index n, and
- T is a system that transforms the input x[n] to an output y[n],

we call this a *digital system* and write:

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

San Diego State University

Filtering

- Transforms are called filters
- Some useful filters
  – low pass
  – band pass
  – high pass
  – homomorphic [later in the semester]

Linear Time Invariant (LTI) Systems

- A system is *linear* iff
  $T\{ax_1[n]+bx_2[n]\} = aT\{x_1[n]\}+bT\{x_2[n]\}$
  (the superposition & linear scaling properties)

- A system is *time-invariant* iff
  $y[n-n_0] = T\{x[n-n_0]\}$

- We will be primarily concerned with a type of transform called a finite impulse response filter.

Finite Impulse Response (FIR) Filters

- FIR filter

h[0]=1 h[1]= -.95

x[2] x[1] x[0]

x[n] = 0 outside 0,1,2

1*0 -.95*0

Initial conditions

y[-1]=1*0+-.95*0

x[2] x[1]

1*x[0] -.95*0

x[0] enters system

y[0]=1*x[0]+-.95*0

x[2]

1*x[1] -.95*x[0]

y[1]=1*x[1]+-.95*x[0]

SAN DIEGO STATE UNIVERSITY

FIR Filters

1*x[2] -.95*x[1]

y[2]=1*x[2]+-.95*x[1]

1*0 -.95*x[2]

y[3]=1*0+-.95*x[2]

1*0 -.95*0

y[4]=1*0+-.95*0

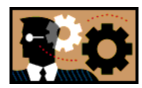
SAN DIEGO STATE UNIVERSITY

Convolution

- The operations of the previous slides can be written as:

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

where h[n-k] is the $(n\text{-}k)^{th}$ filter tap.



SAN DIEGO STATE UNIVERSITY

7

Convolution

- An input signal is *convolved* with filter {h[n]} to produce output: y[n]=x[n]*h[n]

- Convolution is commutative, associative, and distributive

SAN DIEGO STATE UNIVERSITY

8

Convolution in Matlab

```
>> % Input
>> x = [17, 22, 25, 12, -8, 10];
>> % Filter impulse response (tap weights)
>> h = [1, -.95];
>> conv(x, h)        % x * h – convolve x with h
ans =
   17.0000    5.8500    4.1000  -11.7500  -19.4000
    17.6000   -9.5000
```


Complex exponentials

- One signal of special interest is the complex exponential

$$
\begin{aligned}
x[n] &= Ae^{j\omega n} \\
&= A\cos(\omega n) + jA\sin(\omega n)
\end{aligned}
$$

where ω represents the frequency in radians per second and *j* denotes imaginary

[Matlab phasor demo]

This presentation follows that of fred harris's "Trigonometric Transformations," TR DSP-005, 8/1981, Scientific-Atlanta.

Exponentials & sinusoids

- Suppose we add two exponentials which differ only by rotational direction:

$$
\begin{aligned}
x[n] &= Ae^{j\omega n} + Ae^{-j\omega n} \\
&= A\cos(\omega n) + jA\sin(\omega n) + A\cos(\omega n) + jA\sin(-\omega n) \\
&= 2A\cos(\omega n) + jA\sin(\omega n) - jA\sin(\omega n) \\
&= 2A\cos(\omega n)
\end{aligned}
$$

$$A\cos(\omega n) = \frac{Ae^{j\omega n} + Ae^{-j\omega n}}{2}$$

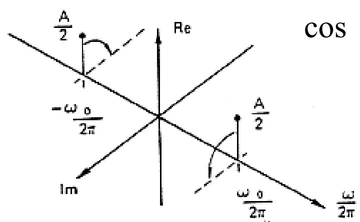
Exponentials and sinusoids

- Similarly, subtraction of the same two exponentials can be shown to be related to sine waves:

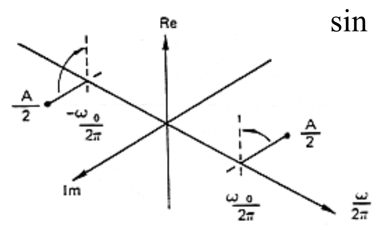
$$A\sin(\omega n) = \frac{Ae^{j\omega n} - Ae^{-j\omega n}}{2j}$$

harris's scoreboard representation

- Cosines/sines are linear combinations of exponentials of half the magnitude



cos



sin

these differ only by phase

harris TR DSP-005

Convolving a filter with an exponential

- Let $x[n] = e^{j\omega_0 n}$

- Then:

$$
\begin{aligned}
y[n] &= x[n] * h[n] \\
&= h[n] * x[n] && \text{commutative} \\
&= \sum_{k=-\infty}^{\infty} h[k] x[n-k] && \text{defn } *
\end{aligned}
$$

Convolving a filter with an exponential

$$y[n] = \sum_{k=-\infty}^{\infty} h[k] e^{j\omega_0 (n-k)} \qquad \text{defn } x[n-k] = e^{j\omega_0 (n-k)}$$

$$= e^{j\omega_0 n} \sum_{k=-\infty}^{\infty} h[k] e^{-j\omega_0 k} \qquad j\omega_0 \text{n is constant}$$

$$= e^{j\omega_0 n} H(e^{j\omega_0}) \qquad \text{where } H(e^{j\omega_0}) = \sum_{k=-\infty}^{\infty} h[k] e^{-j\omega_0 k}$$

- The output y[n] is another exponential of the same frequency scaled by $H(e^{j\omega_0})$

Eigenvalues & Eigensignals

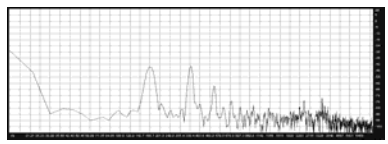
- Complex exponentials are *eigensignals* of LTI systems. An eigensignal convolved with an LTI system produces another eigensignal.
- $H(e^{j\omega_0})$, a complex quantity, is an *eigenvalue* of an LTI system. It indicates the quantity by which the eigensignal is scaled.

Fourier Transform

- The continuous frequency Fourier Transform of x[n] is:

$$X(e^{j\omega_0}) = \sum_{k=-\infty}^{\infty} x[k] e^{-j\omega_0 k}$$

- This enables us to evaluate the frequency content of x, at *any* frequency ω_0.



dB

log Hz

SAN DIEGO STATE UNIVERSITY

Notes of the Fourier Transform

- The Fourier transform for aperiodic signals exists when: $\sum_{n=-\infty}^{\infty} |x[n]| < \infty$

- It is sometimes convenient to normalize the frequency range to [-π, π]

- To move from a normalized [-π, π] frequency axis to Hz:

$$f(\omega) = \frac{\omega}{2\pi} \times SamplingRate, \;\; \omega \in [0, 2\pi]$$

SAN DIEGO STATE UNIVERSITY

Notes on the Fourier transform

- Output of Fourier transform is a set of complex sinusoids, which are eigensignals for LTI systems.
- Recall that LTI systems only *scale* eigensignals, so an LTI system can *only* boost or attenuate existing frequencies in a signal, not add new ones.

SAN DIEGO STATE UNIVERSITY

19

Notes on the Fourier Transform The Convolution Property

- Time domain convolution ↔
  Frequency domain multiplication
  $x[n]*h[n] = X(\omega)H(\omega)$
- Time domain multiplication ↔
  Frequency domain convolution
  $x[n]h[n] = X(\omega)*H(\omega)$

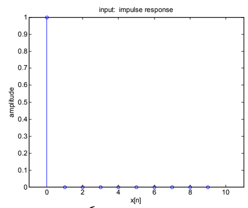
SAN DIEGO STATE UNIVERSITY

20

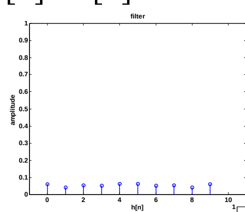
Unit impulse & impulse response

- Consider unit impulse (aka Kronecker delta)

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

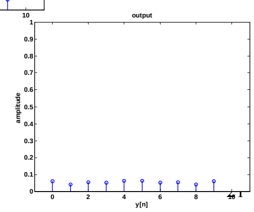


*



$$\delta[n] = \begin{cases} 1 & n = 0 \\ 0 & \text{elsewhere} \end{cases}$$

=



SAN DIEGO STATE UNIVERSITY

Significance of the impulse response

- Completely characterizes an LTI system
- The Fourier transform of the impulse response lets us understand what the system does.

SAN DIEGO STATE UNIVERSITY

Discrete Frequency & Short-Time Fourier Transforms

- Computational methods require a discrete frequency domain.
- Speech signals change over time and we want to analyze a "static" portion of the signal.

DFT - Discrete Fourier Transform

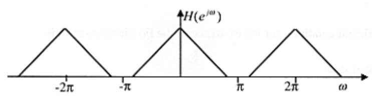
- Assume $x_N[n] = x_N[n+N]$
  - $x_N[n]$ periodic with period N.
- We define the DFT as

$$X_N[k] = \sum_{n=0}^{N-1} x_N[n] e^{-j2\pi nk/N} \text{ where } 0 \le k < N$$

and will refer to this as Fourier analysis

Notes on the DFT

- The DFT is periodic with period 2π.
- $X_n[k]$ samples the frequency domain.



Huang et al. p 209

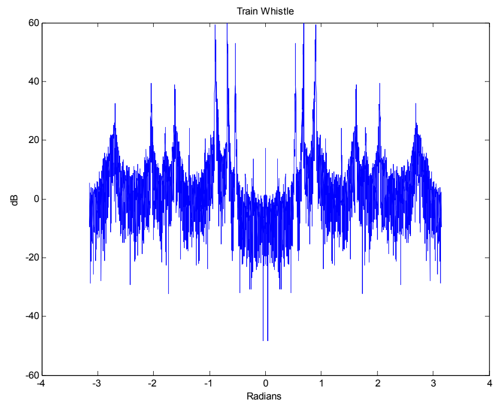
- An alternate form exists for analyzing periodic signals.

Generating a DFT Plot in Matlab

```
% train whistle into y, Fs
% y=signal, Fs=Sample Rate
% train is a standard demo
load train;
sound(y, Fs)   % play it!

% discrete Fourier transform
% (even though name is fft)
Y = fft(y);    % 0 -> 2pi
Y = fftshift(Y); % -pi->pi

% Take magnitude^2
YMagSq = Y .* conj(Y);

% Convert to dB
YdB = 10 * log10(YMagSq);
```

```
% number of frequency bins
N = length(Y);

% Create a normalized freq
% range [-pi, pi)

NormFreq = -pi : 1/N * 2*pi : pi -
   1/N;

% Create a new figure &
% name the window
figure('Name', 'Choo-choo');

plot(NormFreq, YdB);
xlabel('Radians')
ylabel('dB')
title('Train Whistle')
```

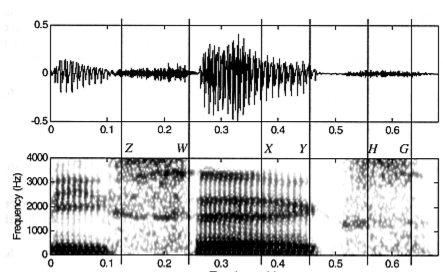
DFT Plot



SAN DIEGO STATE UNIVERSITY

Short Time Fourier Analysis

- Analyzing the whole signal doesn't make sense, not all regions are similar:

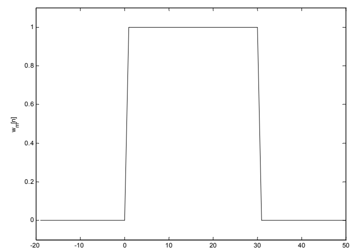


Huang et al. p 277

SAN DIEGO STATE UNIVERSITY

Short-Time Fourier Analysis

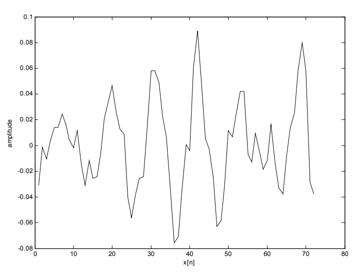
- Segment the signal into small *frames* and then perform an analysis on each frame.
- Let $w_m[n]$ be a rectangular window for the m'th frame.

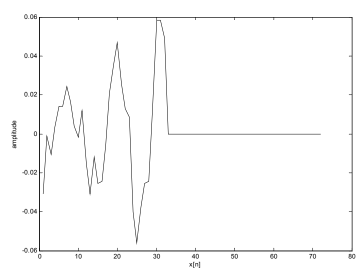


SAN DIEGO STATE UNIVERSITY

Extracting the short-time signal

- x[n]
- $x_m[n]=x[n]w_m[n]$

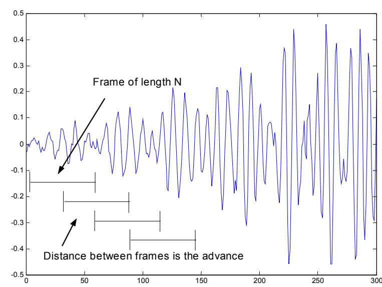




SAN DIEGO STATE UNIVERSITY

Short-time Fourier Analysis

- By shifting the region where the window is non-zero we can control what portion of the signal remains for each frame.
- We must choose:
  - Frame length
  - Frame advance



San Diego State University

31

Short-Time Fourier Analysis

- Fourier analysis of the framed signal:

$$X_m(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n]w_m[n]e^{-j\omega n}$$

- Analysis with discrete frequencies:

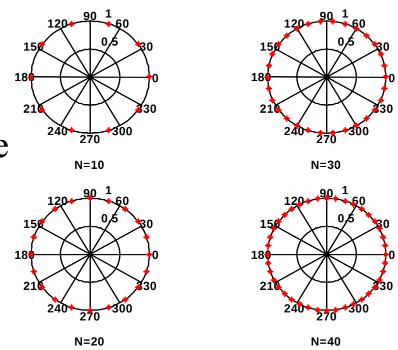
$$X_m[k] = \sum_{n=0}^{N-1} x[n]w_m[n]e^{-j2\pi nk/N} \text{ where } 0 \le k < N$$

San Diego State University

32

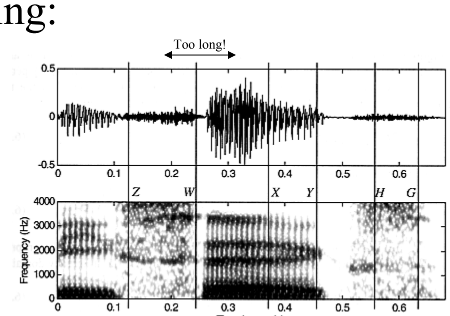
The time frequency trade-off

- As the window length increases, the number of frequencies at which we approximate increases:



Time-Frequency Trade-Off

- but a longer window increases the chance of smearing:



Huang et al. p 277

Window Considerations

- The approximation does not work very well if we do not have at least one pitch period in the window.
- Accommodating lower pitched voices means longer windows.

Window Considerations

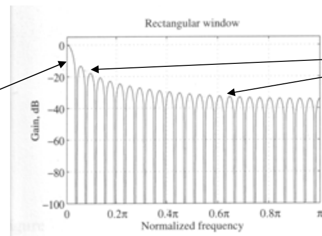
- Recall: Multiplication in the time domain is convolution in the frequency domain

$$x[n]w[n] \leftrightarrow X(\omega)*W(\omega)$$

- Ideally, we would like a window whose transforms was narrow with no side band information.

Spectral leakage

- Consider W(ω) when w[n] is a rectangular window:



Narrow main lobe

side lobes relatively high

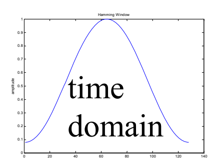
Mitra 2001, p. 453

- During the convolution, the side lobes will result in the DFT having additional energy to either side of a peak. This is called spectral leakage.

SAN DIEGO STATE UNIVERSITY

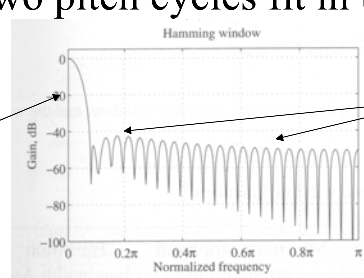
37

Spectral leakage



time domain

- Hamming window has smaller side lobes at the expense of a wider main lobe.
- Requires two pitch cycles fit in the window.



Wider main lobe

side lobes much lower

Mitra 2001, p. 453

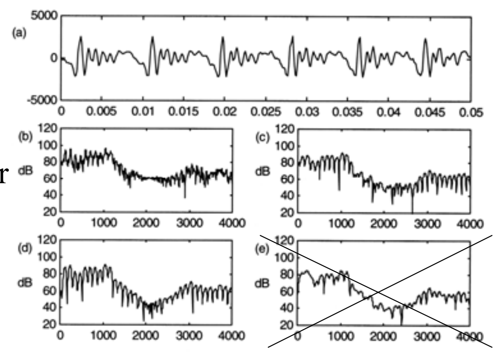
SAN DIEGO STATE UNIVERSITY

38

Spectral leakage comparison

Rectangular

Hamming



Time domain
F0 110 Hz
(9 ms period)

too short!
2 cycles = 18 ms

30 ms 15 ms

window length

Huang et al. 2001, p. 279

Typical framing parameters

- Window: Hamming.
- Frame/Window length: 20-30 ms.
- Frame advance: 10-20 ms.

Inverse DFT

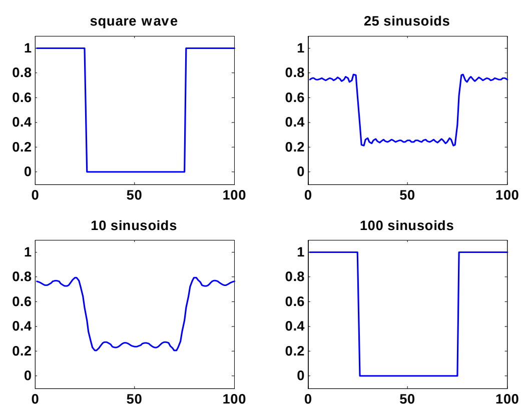
- Given a set of complex exponential coefficients X[n], we can recover the signal x[n]:

$$x_N[n] = \frac{1}{N}\sum_{k=0}^{N-1} X_N[k]e^{j2\pi k/N} \text{ where } 0 \le n < N$$

- Signal recovery is called Fourier synthesis.

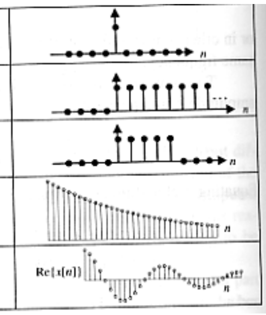
Approximate Synthesis Retaining <N Fourier Coefficients

- Periodic square wave N=100
- Signal synthesis using varying numbers of sinusoids



A few useful signals

Kronecker delta, or unit impulse	$\delta[n] = \begin{cases} 1 & n = 0 \\ 0 & otherwise \end{cases}$	
Unit step	$u[n] = \begin{cases} 1 & n \geq 0 \\ 0 & n < 0 \end{cases}$	
Rectangular signal	$\text{rect}_N[n] = \begin{cases} 1 & 0 \leq n < N \\ 0 & otherwise \end{cases}$	
Real exponential	$x[n] = a^n u[n]$	
Complex exponential	$x[n] = a^n u[n] = r^n e^{jn\omega_0} u[n]$ $= r^n (\cos n\omega_0 + j \sin n\omega_0) u[n]$	



Huang et al. p. 206

SAN DIEGO STATE UNIVERSITY

43

Convolution & Periodic Signals

- When a signal is periodic, the convolution operator

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

is undefined as the sum would be infinite.

- For periodic signals, *circular convolution* is used which sums over one period only:

$$x[n] \otimes h[n] = \sum_{k=<N>} x[k] h[n-k]$$

where <N> is any period of x (i.e. 0:N-1)

SAN DIEGO STATE UNIVERSITY

44

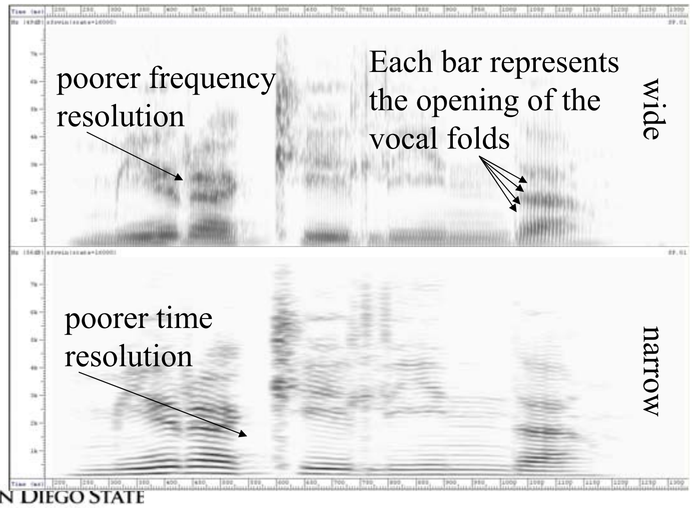
Spectrograms

- Spectrograms are a series of short-time DFTs on windowed segments of a signal
- There are two types of spectrograms:
  - Narrow-band spectrogram
  - Wide-band spectrogram
- The type depends upon the window length.

Window length

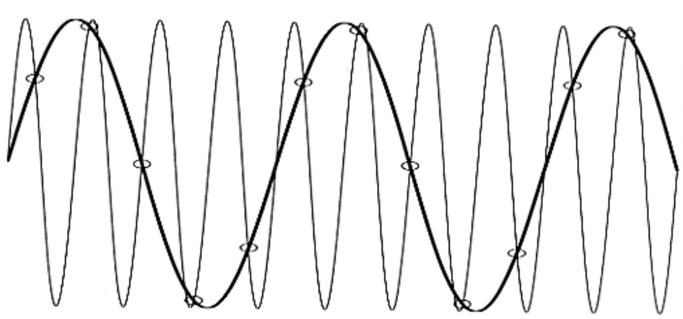
- Short ≈ 10 ms or less
  - Better time analysis
  - Frequency bands are widely spaced.
  - F0 can be estimated by determining time between vertical striations
- Long ≈ 20 ms or more
  - Poor time analysis
  - Frequency bands narrowly spaced.

Narrow vs. Wide-band Spectrogram



Sampling

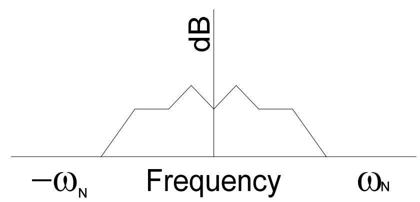
- More than one way to interpolate a set of samples:



Karu p. 98

Nyquist Sampling Theorem

- If an analog signal x(t) is bandlimited between $\pm\omega_N$:



- Then x(t) can be accurately determined by samples x[n] whenever the sampling rate ω_S is > 2 ω_N.

Nyquist Sampling Theorem

- ω_N is called the Nyquist rate.
- Analog signals frequently contain frequencies higher than the rate at which we want to sample.
- Consequently, frequencies higher than ω_N must be eliminated before sampling. This is done through low-pass filtering.

Parseval's Theorem

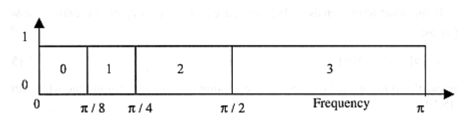
- The total energy in a signal can be useful

- Parseval's Theorem states that energy can be calculated in either the time or frequency domain and that the two are equivalent:

$$\sum_{n=-\infty}^{\infty} \left| x[n] \right|^2 = \frac{1}{2\pi} \left| X(\omega) \right|^2 \delta\omega$$

Filterbanks

- A filterbank is a set of filters than span the spectrum.
- The filterbank decomposes the signal into multiple signals, each of which covers some subset of the total bandwidth:



Huang p. 255

Filterbanks

- Constructing a filterbank is beyond the scope of this class, but we will use the idea later in the semester when we speak about Mel-filtered cepstral coefficients.
- A filterbank can be constructed from a DFT.

SAN DIEGO STATE UNIVERSITY

Discrete Cosine Transform (DCT)

- The DCT is a transform similar to the Fourier transform.
- The DCT decomposes a signal into multiples of cosines.

$$C[k] = \sum_{n=0}^{N-1} x[n]\cos(\pi k(n+1/2)/N) \quad \text{where } 0 \le k < N$$

SAN DIEGO STATE UNIVERSITY

DCT vs. DFT

- The DCT represents more of the signal with fewer coefficients than the DFT (called energy compaction).
- We will use the DCT later in the semester when we compute a feature known as the *cepstrum*. (Matlab: function dct)