Going back to A sending bits in parallel to B:

Suppose A wishes to send 3 bit streams to B in parallel using three carrier frequencies f_1, f_2, and f_3

$\rightarrow$ say 3 bit streams: 1000100, 0101100, 1101101

$\rightarrow$ before: 3 single bits (1, 0, 1)

Does Fourier's framework work for parallel bit streams?

No.

Problem: to transmit 1000100 on carrier frequency f_1, we need to change its weight α_f over time

$\rightarrow$ large, small, small, small, large, small, small

$\rightarrow$ per period $T_1 = 1/f_1$

$\rightarrow$ weights don't change in Fourier's framework

$\rightarrow$ cannot use as is

Could try piecemeal approach: for carrier frequency f_1

Sender-side: encoding methods remains the same

$\rightarrow$ bit stream: 1000100

- during time interval $[0, T_1]$:

  $\rightarrow$ large sine amplitude (bit 1)

  $\rightarrow$ i.e., large α_{f_1}

- during time interval $[T_1, 2T_1]$:

  $\rightarrow$ small sine amplitude (bit 0)

  $\rightarrow$ i.e., small α_{f_1}

- during time interval $[2T_1, 3T_1]$:

  $\rightarrow$ small sine amplitude (bit 0)

  $\rightarrow$ i.e., small α_{f_1}

- etc.

  $\longrightarrow$ in parallel: carrier frequencies f_2 and f_3

Receiver-side:

- during time interval $[0, T_1]$: compute

$$\alpha_{f_1} = \int_0^{T_1} s(t) e^{-if_1 t} dt$$

$\rightarrow$ 1st bit: α_{f_1} is large hence 1

- during time interval $[T_1, 2T_1]$: compute

$$\alpha_{f_1} = \int_{T_1}^{2T_1} s(t) e^{-if_1 t} dt$$

$\rightarrow$ 2nd bit: α_{f_1} is small hence 0

- during time interval $[2T_1, 3T_1]$: compute

$$\alpha_{f_1} = \int_{2T_1}^{3T_1} s(t) e^{-if_1 t} dt$$

$\rightarrow$ 3rd bit: α_{f_1} is small hence 0

- etc.

$\longrightarrow$ problem solved!

$\longrightarrow$ not quite

$\longrightarrow$ 1st bit: α_{f_1} may not be large

When performing Fourier transform over finite time interval:

$\rightarrow$ bleeding or leakage effect

$\rightarrow$ between carrier waves f_1, f_2, f_3

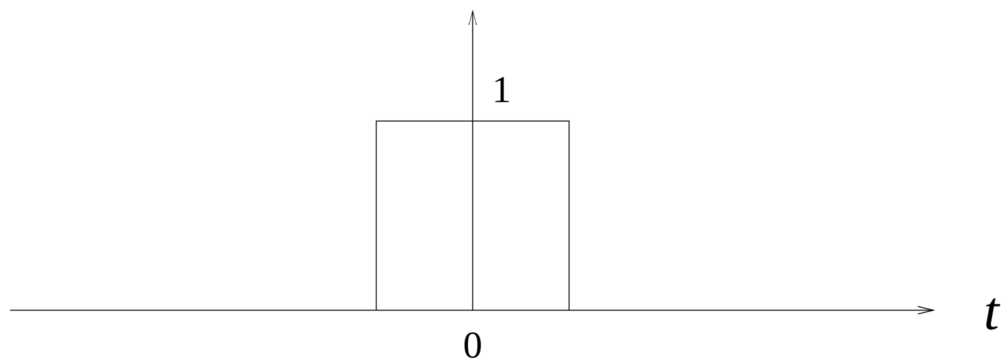
$\rightarrow$ interference

$\rightarrow$ weights α_{f_1}, α_{f_2}, and α_{f_3} may get corrupted

$\rightarrow$ hence bits may get corrupted

$\rightarrow$ why?

Consider square signal:

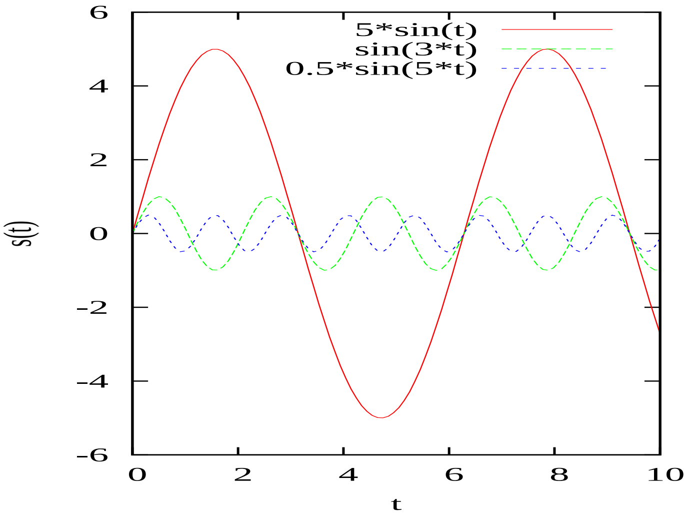


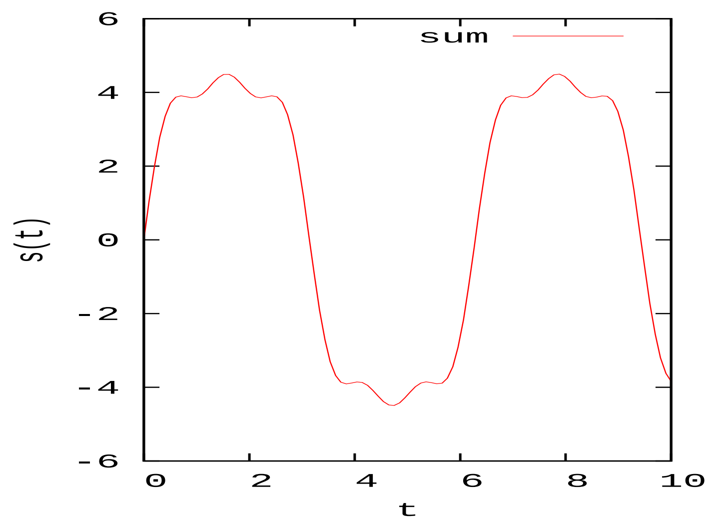
→ one of the signals considered difficult to synthesize using sinusoids

→ sharp transition or edge

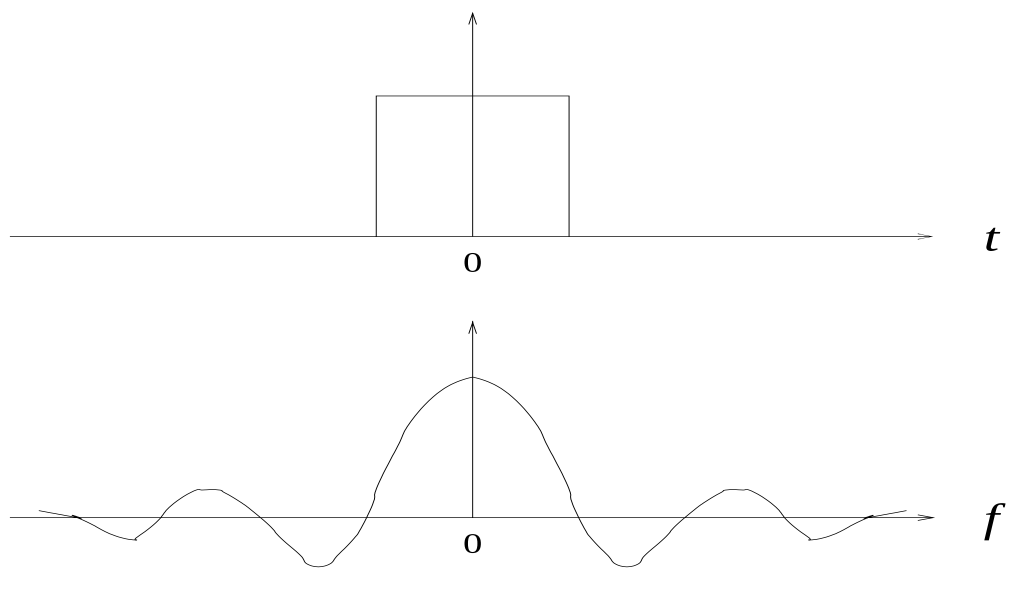
→ from 0 to 1, and 1 to 0

Three sine curves and their sum:





The sinusoids and their weights needed to create square wave:



$\rightarrow$ weights are called spectrum

$\rightarrow$ list of weight values

$\rightarrow$ ever higher frequency sinusoids required

$\rightarrow$ however weight values decrease: less important

$\rightarrow$ needed for fine detail: sharp edge

Why relevant for us?

→ square wave example: timelimited

→ zero outside of finite time interval

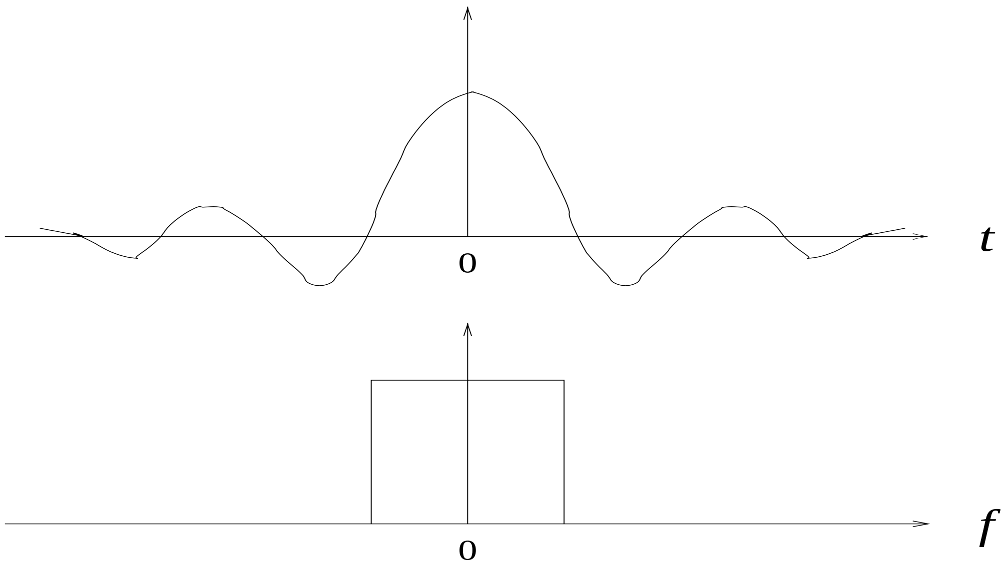
→ but spectrum: infinite

→ if finite: called bandlimited

Fact:

If a signal is timelimited, its spectrum is not bandlimited; and vice versa

Bandlimited signal that is not timelimited:



→ opposite from before

Connection to FDM:

When performing Fourier transform for finite time interval $[0, T_1]$

$\rightarrow$ similar to timelimited signal

$\rightarrow$ view as zero outside $[0, T_1]$

$\rightarrow$ sinusoid f_1 is not "pure" anymore

$\rightarrow$ since timelimited its spectrum is not bandlimited

$\rightarrow$ hence Fourier transform has non-zero α_{f_2} and α_{f_3}

$\rightarrow$ can cause distortion when performing Fourier transform for f_2 and f_3

$\rightarrow$ interference!

$\rightarrow$ inter-channel or inter-carrier interference (ICI)

Example: IEEE 802.11 WLAN

→ U.S.: 11 channels for 2.4 GHz systems

→ channel: similar to carrier frequency

→ 2.412, 2.417, 2.422, 2.427, 2.432, 2.437, 2.442, 2.447, 2.452, 2.457, 2.462 GHz

→ channel separation must be at least by 5 channel to avoid interference

→ three hot spots in neighboring coffee houses: 1, 6, 11

→ same in office buildings, residential areas

Traditional way to combat ICI in FDM: use guard bands

→ insert sufficient gaps between carrier frequencies

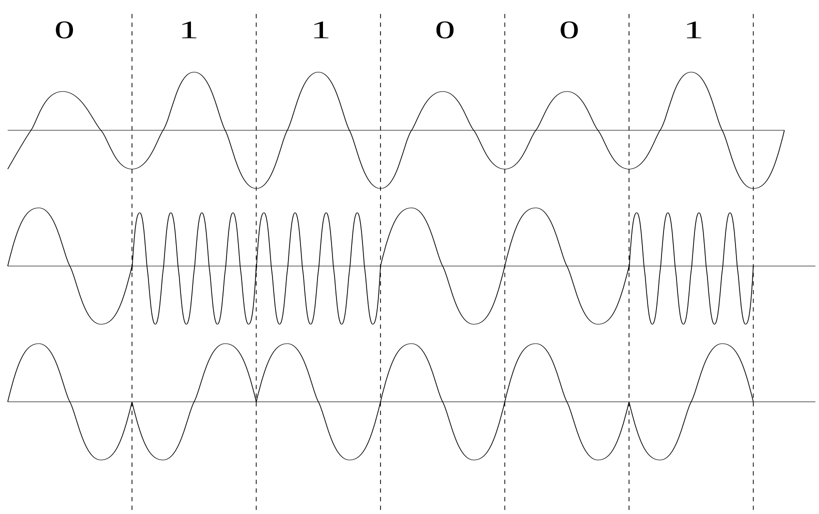
→ overhead can be significant

→ reduces how many carrier frequencies can be squeezed into a given frequency band

General picture:

Amplitude modulation (AM) is but one way to encode bits using sinusoid carrier waves

- Amplitude modulation (AM): encode bits using amplitude levels
- Frequency modulation (FM): encode bits using frequency changes
- Phase modulation (PM): encode bits using phase shifts



Also called amplitude, frequence, phase shift keying (ASK, FSK, PSK)

→ e.g., BPSK: binary PSK

→ or their combination

→ QAM (quadrature amplitude modulation): amplitude and phase

→ e.g., 16-QAM: 4 amplitudes, 4 phases

→ constellation diagram

Back to FDM:

Clearly if frequency modulation (FM) is used then carrier frequency f is not simply f but $f \pm \delta$

- $f + \delta$: bit 1
- $f - \delta$: bit 0
- f: called center frequency

$\rightarrow$ thus: carrier separation must be at least 2δ (Hz) plus guard band

Other factors:

Frequency distortion

→ amplitude degradation—called attenuation—varies by frequency

Doppler shift

→ mobile communication

→ carrier frequency appears faster or slower depending on direction and speed of movement