

# Introduction to Audio Content Analysis

Module B.6: Frequency Resolution & Instantaneous Frequency

alexander lerch

# introduction

overview

corresponding textbook section

- Section 7.3.1
- Section B.6

## ■ lecture content

- frequency detection error for sampled signals
- instantaneous frequency/frequency reassignment

## ■ learning objectives

- list the factors influencing frequency resolution in time and frequency domains
- explain the frequency error in Cent
- implement an instantaneous frequency estimate



# introduction

overview

corresponding textbook section

- Section 7.3.1
- Section B.6

## ■ lecture content

- frequency detection error for sampled signals
- instantaneous frequency/frequency reassignment

## ■ learning objectives

- list the factors influencing frequency resolution in time and frequency domains
- explain the frequency error in Cent
- implement an instantaneous frequency estimate



# pitch detection resolution

## introduction

- (fundamental) frequency detection on digital signals (discrete in time and frequency)  
⇒ quantized result

error being made due to *discrete* signal processing

- time domain:
  - detection of *period length*  
⇒ maximum error depends on distance between two samples (sample rate)
- frequency domain:
  - detection of *bin frequency*  
⇒ maximum error depends on distance between two frequency bins (block length and sample rate)

# pitch detection resolution

## introduction

- (fundamental) frequency detection on digital signals (discrete in time and frequency)
  - ⇒ quantized result

error being made due to *discrete* signal processing

- **time domain:**
  - detection of *period length*
    - ⇒ maximum error depends on distance between two samples (sample rate)
- **frequency domain:**
  - detection of *bin frequency*
    - ⇒ maximum error depends on distance between two frequency bins (block length and sample rate)

# pitch detection resolution

## introduction

- (fundamental) frequency detection on digital signals (discrete in time and frequency)

⇒ quantized result

error being made due to *discrete* signal processing

- **time domain:**

- detection of *period length*

⇒ maximum error depends on distance between two samples (sample rate)

- **frequency domain:**

- detection of *bin frequency*

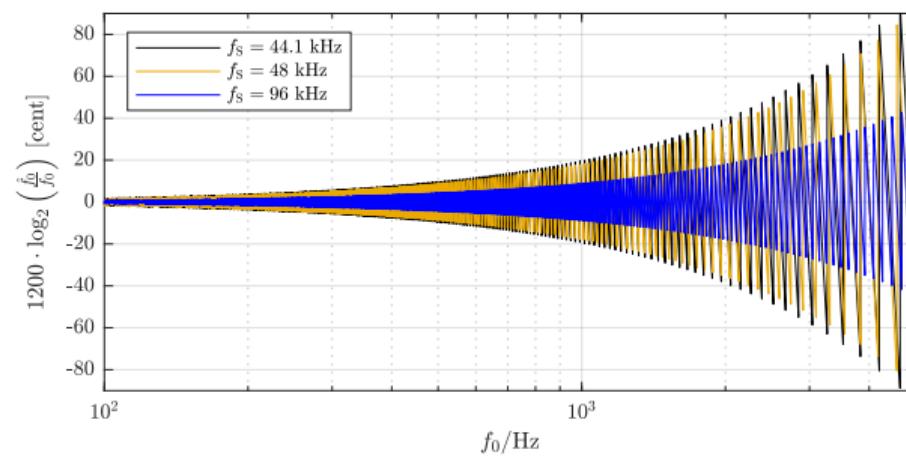
⇒ maximum error depends on distance between two frequency bins (block length and sample rate)

# pitch detection resolution

time domain (e.g., ACF)

period length quantized to multiple of inter-sample interval  $T_S$

$$\begin{aligned} T_Q &= j \cdot T_S \\ \Rightarrow f_Q &= \frac{1}{j \cdot T_S} \end{aligned}$$



# pitch detection resolution

frequency domain (e.g., HPS)

frequency quantized to multiple of inter-bin interval

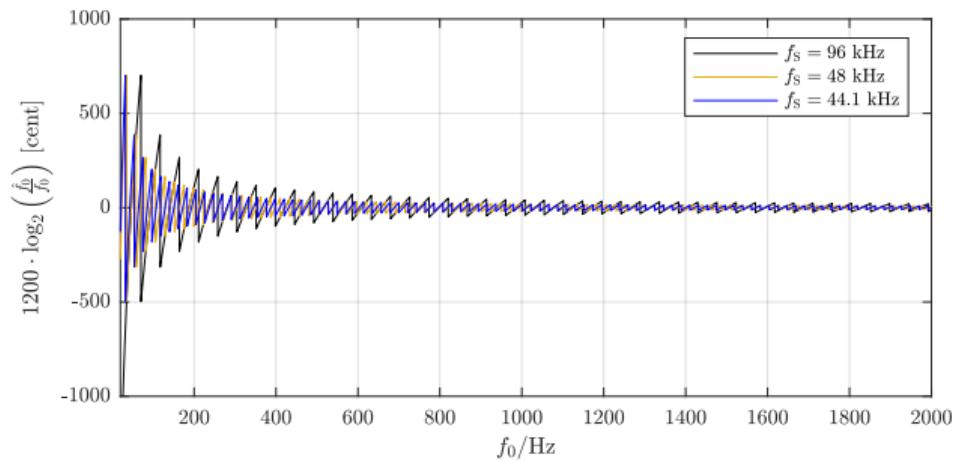
$$f_Q = k \cdot \frac{f_S}{\mathcal{K}}$$

| $\mathcal{K}$ | $\Delta f$ [Hz] | $k_{ST}$ | $f(k_{ST})$ [Hz] |
|---------------|-----------------|----------|------------------|
| <b>256</b>    | 187.5           | 35       | 6562.5           |
| <b>512</b>    | 93.75           | 35       | 3281.25          |
| <b>1024</b>   | 46.875          | 35       | 1640.625         |
| <b>2048</b>   | 23.4375         | 35       | 820.3125         |
| <b>4096</b>   | 11.7188         | 35       | 410.1563         |
| <b>8192</b>   | 5.8594          | 35       | 205.0781         |
| <b>16384</b>  | 2.9297          | 35       | 102.5391         |

# pitch detection resolution frequency domain (e.g., HPS)

frequency quantized to multiple of inter-bin interval

$$f_Q = k \cdot \frac{f_S}{\mathcal{K}}$$



# pitch detection resolution

## simple fix

■ **assumption:** pitch is stationary with minor deviations over time

■ **simple solution:**

- average pitch observations over blocks
- the more blocks are averaged, the more result might approximate the *real* (population) mean

■ **problems:**

- 1 adds significant latency (non-realtime)
- 2 will not work for time-variant signals (speech, music)

# pitch detection resolution

## simple fix

■ **assumption:** pitch is stationary with minor deviations over time

■ **simple solution:**

- average pitch observations over blocks
- the more blocks are averaged, the more result might approximate the *real* (population) mean

■ **problems:**

- 1 adds significant latency (non-realtime)
- 2 will not work for time-variant signals (speech, music)

# pitch detection resolution

## simple fix

■ **assumption:** pitch is stationary with minor deviations over time

■ **simple solution:**

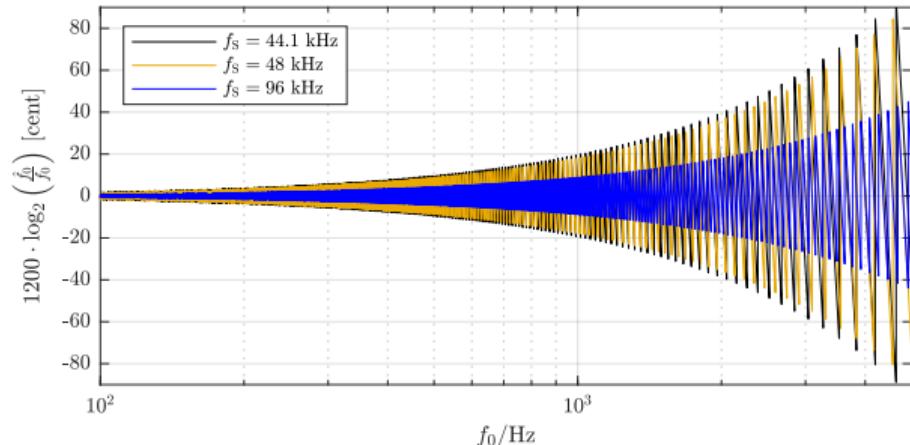
- average pitch observations over blocks
- the more blocks are averaged, the more result might approximate the *real* (population) mean

■ **problems:**

- 1 adds significant latency (non-realtime)
- 2 will not work for time-variant signals (speech, music)

# pitch detection resolution

## time domain observations

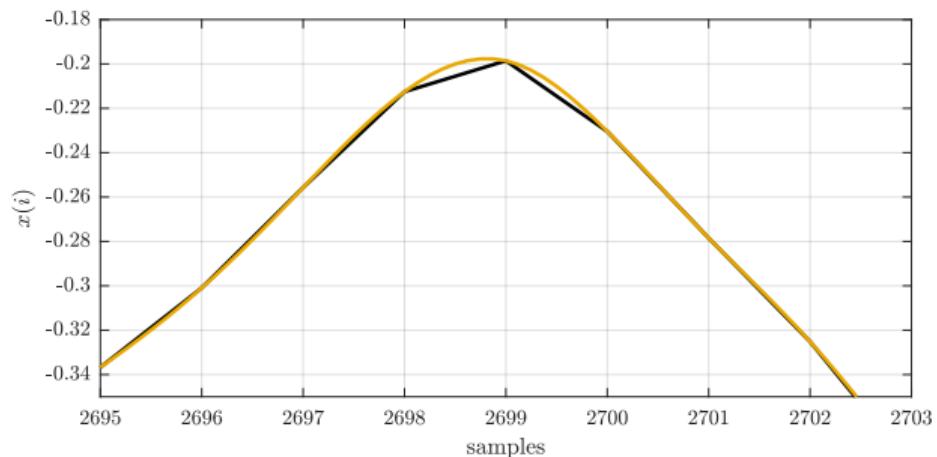


- error depends on fundamental frequency
- error depends on sample rate

# pitch detection resolution

## time domain workarounds

virtually increase time resolution by upsampling



- + higher virtual resolution
- significant workload increase

# pitch detection resolution

## frequency domain workarounds

different ways of increasing frequency resolution in the frequency domain

- 1 increasing the FFT window length (decreases time resolution)
- 2 interpolating the spectrum
- 3 applying frequency reassignment

# pitch detection resolution

## frequency domain workarounds

different ways of increasing frequency resolution in the frequency domain

- 1 increasing the FFT window length (decreases time resolution)
- 2 interpolating the spectrum
- 3 applying frequency reassignment

# pitch detection resolution

## frequency domain workarounds

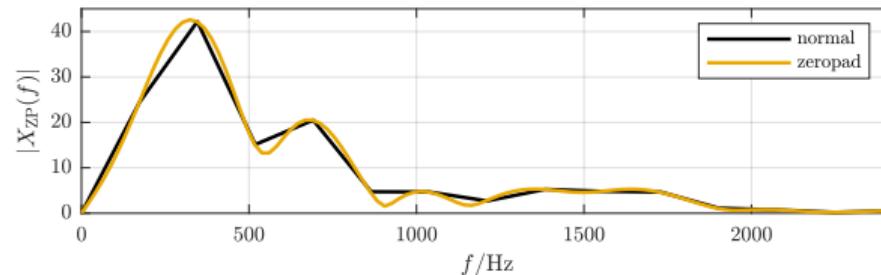
different ways of increasing frequency resolution in the frequency domain

- 1 increasing the FFT window length (decreases time resolution)
- 2 interpolating the spectrum
- 3 applying frequency reassignment

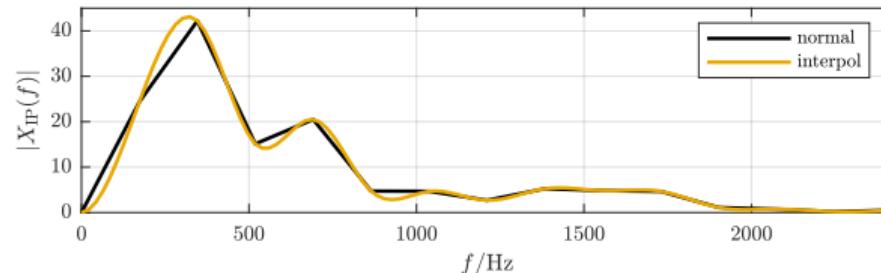
# pitch detection resolution

## spectrum interpolation

1 zeropad in time domain



2 use standard interpolation on magnitude spectrum



# pitch detection resolution

frequency reassignment: relation of phase and frequency 1/2

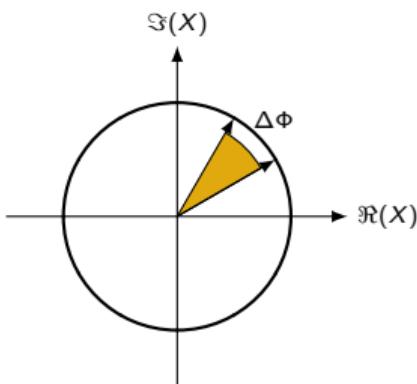
## ■ phasor representation:

- 1 sine value is defined by magnitude and phase
- 2 decreasing the amplitude  $\Rightarrow$  shorter vector
- 3 increasing the frequency  $\Rightarrow$  increasing speed



# pitch detection resolution

frequency reassignment: relation of phase and frequency 2/2



## ■ relation of frequency and phase change:

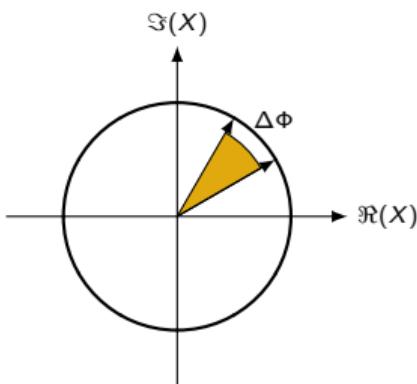
- time for full rotation is period length  $T$  with
$$f = \frac{1}{T}$$
- time for fractional rotation  $\Delta\Phi$  is corresponding fraction of period length
$$f = \frac{\Delta\Phi}{\Delta t}$$

- in other words:

$$\begin{aligned}\Phi(t) &= \omega \cdot t \\ \Rightarrow \frac{d\Phi(t)}{dt} &= \omega = 2\pi f\end{aligned}$$

# pitch detection resolution

frequency reassignment: relation of phase and frequency 2/2



## ■ relation of frequency and phase change:

- time for full rotation is period length  $T$  with

$$f = \frac{1}{T}$$

- time for fractional rotation  $\Delta\Phi$  is corresponding fraction of period length

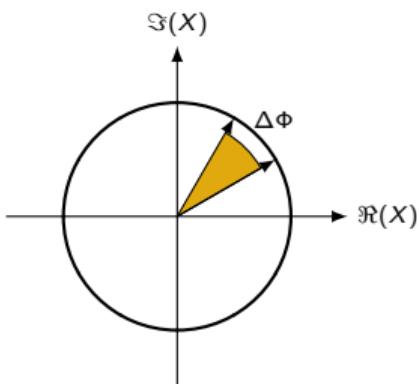
$$f = \frac{\Delta\Phi}{\Delta t}$$

- in other words:

$$\begin{aligned}\Phi(t) &= \omega \cdot t \\ \Rightarrow \frac{d\Phi(t)}{dt} &= \omega = 2\pi f\end{aligned}$$

# pitch detection resolution

frequency reassignment: relation of phase and frequency 2/2



## ■ relation of frequency and phase change:

- time for full rotation is period length  $T$  with

$$f = \frac{1}{T}$$

- time for fractional rotation  $\Delta\Phi$  is corresponding fraction of period length

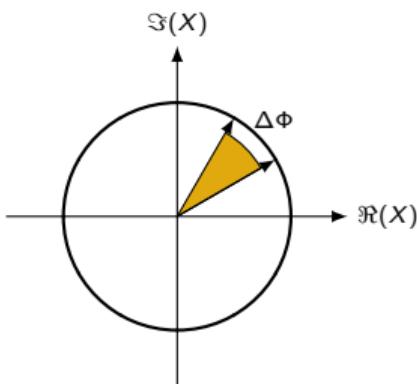
$$f = \frac{\Delta\Phi}{\Delta t}$$

- in other words:

$$\begin{aligned}\Phi(t) &= \omega \cdot t \\ \Rightarrow \frac{d\Phi(t)}{dt} &= \omega = 2\pi f\end{aligned}$$

# pitch detection resolution

frequency reassignment: relation of phase and frequency 2/2



- relation of frequency and phase change:

- time for full rotation is period length  $T$  with

$$f = \frac{1}{T}$$

- time for fractional rotation  $\Delta\Phi$  is corresponding fraction of period length

$$f = \frac{\Delta\Phi}{\Delta t}$$

- in other words:

$$\begin{aligned}\Phi(t) &= \omega \cdot t \\ \Rightarrow \frac{d\Phi(t)}{dt} &= \omega = 2\pi f\end{aligned}$$

# pitch detection resolution

## frequency reassignment: principles

frequency domain:

- instead of using the bin frequency

$$f(k) = k \cdot \frac{f_s}{K}$$

- we use the phase of each bin  $\Phi(k, n)$
- to compute the frequency from the phase difference of neighboring blocks

$$\omega_I(k, n) \propto \Phi(k, n) - \Phi(k, n - 1)$$

- $\omega_I(k, n)$  is called **instantaneous frequency** per block per bin

# pitch detection resolution

## frequency reassignment: principles

frequency domain:

- instead of using the bin frequency

$$f(k) = k \cdot \frac{f_s}{K}$$

- we use the phase of each bin  $\Phi(k, n)$
- to compute the frequency from the phase difference of neighboring blocks

$$\omega_I(k, n) \propto \Phi(k, n) - \Phi(k, n - 1)$$

- $\omega_I(k, n)$  is called **instantaneous frequency** per block per bin

# pitch detection resolution

frequency reassignment: scaling factor

- instantaneous frequency calculation has to take into account

- hop size  $\mathcal{H}$
- sample rate  $f_S$

$$\omega_I(k, n) = \frac{\Delta\Phi_u(k, n)}{\mathcal{H}} \cdot f_S$$

- problem: phase ambiguity

$$\Phi(k, n) = \Phi(k, n) + j \cdot 2\pi$$

⇒ *phase unwrapping*

# pitch detection resolution

frequency reassignment: scaling factor

- instantaneous frequency calculation has to take into account

- hop size  $\mathcal{H}$
- sample rate  $f_S$

$$\omega_I(k, n) = \frac{\Delta\Phi_u(k, n)}{\mathcal{H}} \cdot f_S$$

- problem: phase ambiguity

$$\Phi(k, n) = \Phi(k, n) + j \cdot 2\pi$$

⇒ *phase unwrapping*

# pitch detection resolution

## frequency reassignment: phase unwrapping

### 1 compute unwrapped phase $\Phi_u(k, n)$

- estimate unwrapped bin phase

$$\hat{\Phi}(k, n) = \Phi(k, n - 1) + \underbrace{2\pi k \cdot \frac{\mathcal{H}}{\mathcal{K}}}_{= \omega_k \cdot \frac{\mathcal{H}}{f_s}}$$

- unwrap phase by shifting current phase to estimate's range

$$\Phi_u(k, n) = \hat{\Phi}(k, n) + \text{princarg} [\Phi(k, n) - \hat{\Phi}(k, n)]$$

### 2 compute unwrapped phase difference

$$\begin{aligned}\Delta\Phi_u(k, n) &= \Phi_u(k, n) - \Phi(k, n - 1) \\ &= \hat{\Phi}(k, n) + \text{princarg} [\Phi(k, n) - \hat{\Phi}(k, n)] - \Phi(k, n - 1) \\ &= \frac{2\pi k}{\mathcal{K}} \mathcal{H} + \text{princarg} [\Phi(k, n) - \Phi(k, n - 1) - \frac{2\pi k}{\mathcal{K}} \mathcal{H}]\end{aligned}$$

# pitch detection resolution

## frequency reassignment: phase unwrapping

### 1 compute unwrapped phase $\Phi_u(k, n)$

- estimate unwrapped bin phase

$$\hat{\Phi}(k, n) = \Phi(k, n - 1) + \underbrace{2\pi k \cdot \frac{\mathcal{H}}{\mathcal{K}}}_{= \omega_k \cdot \frac{\mathcal{H}}{f_s}}$$

- unwrap phase by shifting current phase to estimate's range

$$\Phi_u(k, n) = \hat{\Phi}(k, n) + \text{princarg} [\Phi(k, n) - \hat{\Phi}(k, n)]$$

### 2 compute unwrapped phase difference

$$\begin{aligned}\Delta\Phi_u(k, n) &= \Phi_u(k, n) - \Phi(k, n - 1) \\ &= \hat{\Phi}(k, n) + \text{princarg} [\Phi(k, n) - \hat{\Phi}(k, n)] - \Phi(k, n - 1) \\ &= \frac{2\pi k}{\mathcal{K}} \mathcal{H} + \text{princarg} \left[ \Phi(k, n) - \Phi(k, n - 1) - \frac{2\pi k}{\mathcal{K}} \mathcal{H} \right]\end{aligned}$$

# pitch detection resolution

## frequency reassignment: phase unwrapping

### 1 compute unwrapped phase $\Phi_u(k, n)$

- estimate unwrapped bin phase

$$\hat{\Phi}(k, n) = \Phi(k, n - 1) + \underbrace{2\pi k \cdot \frac{\mathcal{H}}{\mathcal{K}}}_{= \omega_k \cdot \frac{\mathcal{H}}{f_s}}$$

- unwrap phase by shifting current phase to estimate's range

$$\Phi_u(k, n) = \hat{\Phi}(k, n) + \text{princarg} [\Phi(k, n) - \hat{\Phi}(k, n)]$$

### 2 compute unwrapped phase difference

$$\begin{aligned}\Delta\Phi_u(k, n) &= \Phi_u(k, n) - \Phi(k, n - 1) \\ &= \hat{\Phi}(k, n) + \text{princarg} [\Phi(k, n) - \hat{\Phi}(k, n)] - \Phi(k, n - 1) \\ &= \frac{2\pi k}{\mathcal{K}} \mathcal{H} + \text{princarg} \left[ \Phi(k, n) - \Phi(k, n - 1) - \frac{2\pi k}{\mathcal{K}} \mathcal{H} \right]\end{aligned}$$

# pitch detection resolution

## frequency reassignment: problems

### ■ overlapping spectral components

- sinusoidal components often overlap (spectral leakage, several instruments playing the same pitch, ...)
  - ⇒ “incorrect” phase estimate
  - spectrum should be as sparse as possible, increase STFT length

### ■ inaccurate phase unwrapping

- unwrapping algorithm is based on assumption of similarity between predicted and measured phase
- decrease hop size

# pitch detection resolution

frequency reassignment: problems

## ■ overlapping spectral components

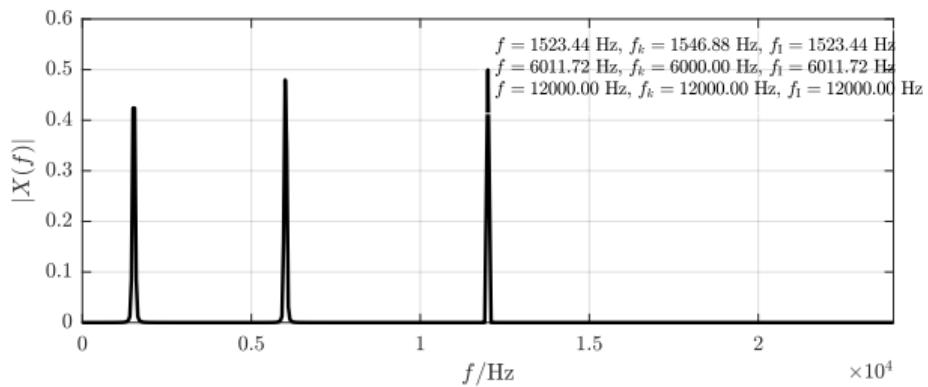
- sinusoidal components often overlap (spectral leakage, several instruments playing the same pitch, ...)
  - ⇒ “incorrect” phase estimate
  - spectrum should be as sparse as possible, increase STFT length

## ■ inaccurate phase unwrapping

- unwrapping algorithm is based on assumption of similarity between predicted and measured phase
- decrease hop size

# pitch detection resolution

## frequency reassignment: example



- FFT length: 1024
- sample rate: 48 kHz

- selected frequencies:
  - between bins (0.5)
  - between bins (0.25)
  - on bin

# pitch detection resolution

## frequency reassignment: applications

### ■ improving frequency resolution

- e.g., for detecting signal frequencies when using a filter bank

### ■ improving phase extrapolation

- e.g., for accurate phase estimation in the *phase vocoder*

### ■ grouping spectral bins

- spectral leakage sidelobes have the same instantaneous frequency

### ■ tonalness detection

- the instantaneous frequency should be reasonably close to the bin frequency for the component to be considered tonal

# pitch detection resolution

## frequency reassignment: applications

### ■ improving frequency resolution

- e.g., for detecting signal frequencies when using a filter bank

### ■ improving phase extrapolation

- e.g., for accurate phase estimation in the *phase vocoder*

### ■ grouping spectral bins

- spectral leakage sidelobes have the same instantaneous frequency

### ■ tonalness detection

- the instantaneous frequency should be reasonably close to the bin frequency for the component to be considered tonal

# pitch detection resolution

## frequency reassignment: applications

### ■ improving frequency resolution

- e.g., for detecting signal frequencies when using a filter bank

### ■ improving phase extrapolation

- e.g., for accurate phase estimation in the *phase vocoder*

### ■ grouping spectral bins

- spectral leakage sidelobes have the same instantaneous frequency

### ■ tonalness detection

- the instantaneous frequency should be reasonably close to the bin frequency for the component to be considered tonal

# pitch detection resolution

## frequency reassignment: applications

### ■ improving frequency resolution

- e.g., for detecting signal frequencies when using a filter bank

### ■ improving phase extrapolation

- e.g., for accurate phase estimation in the *phase vocoder*

### ■ grouping spectral bins

- spectral leakage sidelobes have the same instantaneous frequency

### ■ tonalness detection

- the instantaneous frequency should be reasonably close to the bin frequency for the component to be considered tonal

**■ frequency resolution of sampled signals depends on**

- time domain: sample rate
- freq domain: sample rate, block size

**■ pitch detection error in Cent also depends on input frequency**

- time domain: high error at high frequencies
- freq domain: high error at low frequencies

**■ possible solutions**

- time domain:
  - ▶ upsampling/interpolation
- freq domain:
  - ▶ zeropadding/interpolation
  - ▶ frequency reassignment (instantaneous frequency)

