PITCH AND MULTIPITCH ESTIMATION

Anssi Klapuri, klap@cs.tut.fi

Esityksen sisältö:

- **Pitch estimation**
  - Definitions: fundamental frequency, pitch
  - Peculiar phenomena in human pitch perception
  - Different ways of calculating pitch
  - “Unitary” pitch algorithm
- **Multipitch estimation (MPE)**
  - Close relation to CASA in general
  - Problems in MPE, and review of MPE systems
  - Case study: klap’s music transcription system
  - Case study: ...problem solving strategy and results

DEFINITION OF PITCH

Definition of *pitch*

- **ANSI (1994): “Pitch is that attribute of auditory sensation in terms of which sounds may be ordered on a scale extending from low to high.” (verbal definition)**
- **Hartman (1996): “Sound has a certain pitch if it can be reliably matched by adjusting the frequency of a sine wave of arbitrary amplitude.” (operational definition)**

Fundamental frequency vs. pitch

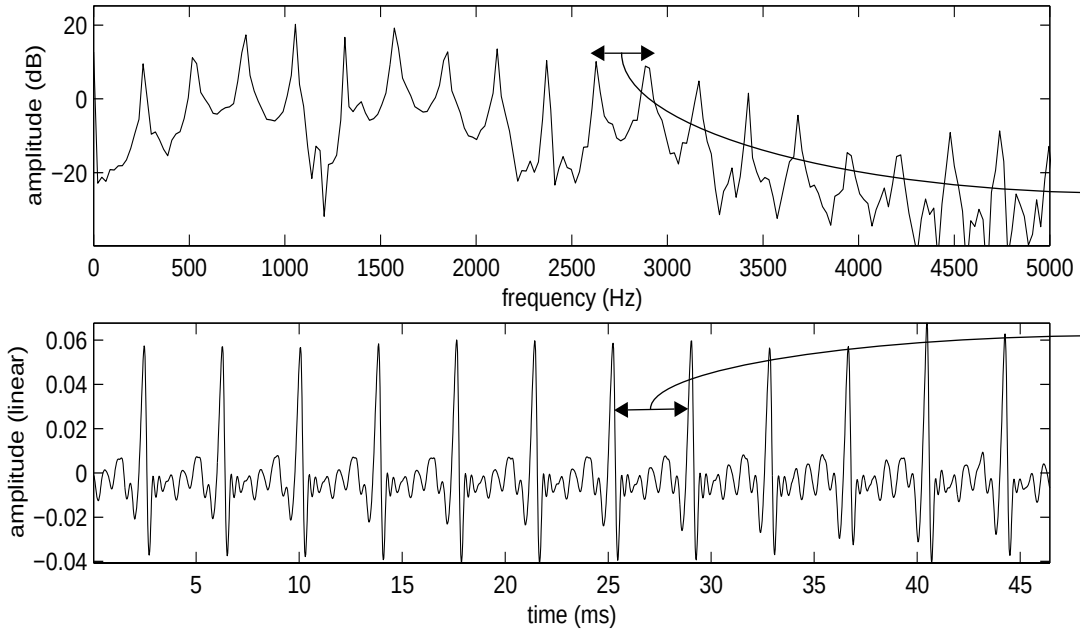
- **fundamental frequency (F0) is a *physical* term**
- **pitch is a *perceptual* term (perceived F0)**
- **both measured in Hertz (Hz)**
- **usually pitch (perceived F0) ~ F0 (approximately equal)**

HARMONIC SOUNDS

For sine waves:

- **F0 = frequency**
- **pitch ~ frequency**

Complex harmonic sounds

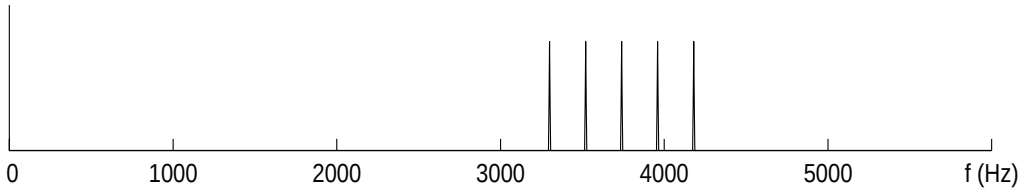


Trumpet sound

- **F0:**
  F = 262 Hz
- **wavelength**
  $1/F$ = 3.8 ms

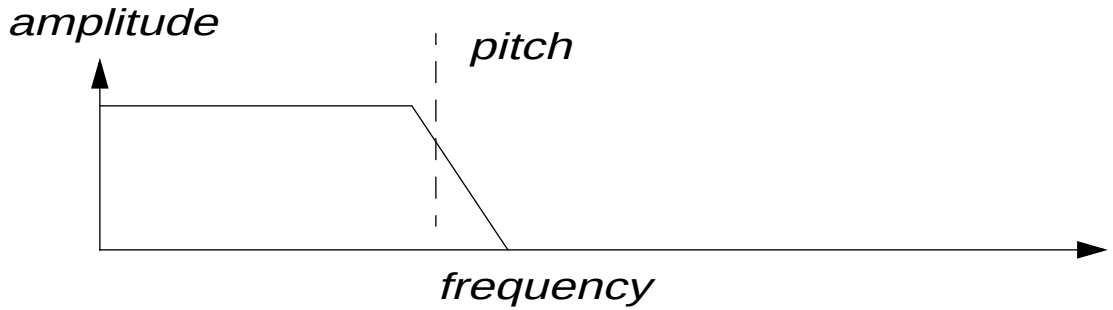
THE PECULIAR PITCH PERCEPTION

- **Pitch perception plays an important role in human hearing, and auditory system seems to be trying awfully hard to assign a pitch to anything that comes to its attention**
- **Harmonic sounds (the “normal” case)**
- **Missing fundamental (= virtual pitch = low p. = residue p.)**
  - a harmonic sound, where the fundamental component is missing



- **Shifted harmonics**
  - frequency components in the above figure are arbitrarily shifted so that their frequencies are no more multiples of a common F0
- **Strongly inharmonic sounds: bells, membranes, plates etc.**

- **Repetition pitch**
  - an arbitrary (e.g. noise) signal and its delayed version are summed
  - the delay is perceived, although the spectrum is flat
- **Edge pitch**
  - steep high- or lowpass filtering applied —> pitch at the edge frequency



- **AM-modulated noise —> pitch at the modulation rate**
  - amplitude of a noise signal oscillates at the pitch rate
- **Dichotic pitch: one frequency component to each ear**
  - —> perceived pitch is resulted from the mixture of the two sinusoids, somewhere in the brains

HOW TO CALCULATE PITCH?

Pitch is an *emergent* property, i.e., caused by the joint effect of several signal components that do not possess the same property alone

- **Mapping from acoustic features to pitch is complicated**
- **There is no single obvious way of calculating pitch**

> **Algorithms do not only differ in technical details, but in regard to the *information* that the calculations are based on**

In following, focus on more or less harmonic sounds

- **Peculiar signals can be used to reveal things about the mechanisms in human hearing**
- **Harmonic sounds are usually concerned in applications**
  - speech, music, several other vibrating systems
- **Keep in mind the plots of a harmonic sound on page 3**

“SPECTRAL PLACE” ALGORITHMS

Autocorrelation function (ACF) based algorithms

- **Among the most frequently used F0 estimators. Usually the maximum value in ACF is taken as 1/*F* period**
- **ACF *r*(*n*) for a discrete time domain signal *x*(*k*):**

$$r(n) = \frac{1}{K} \cdot \sum_{k=0}^{K-n-1} x(k)x(k+n)$$

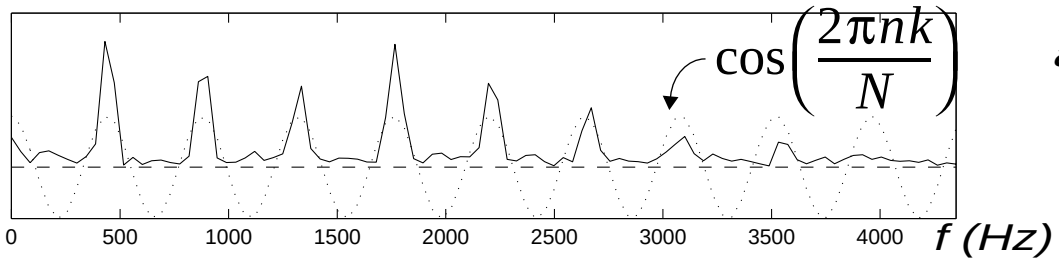
Let us analyze what ACF calculates in frequence domain terms

- **Here equations are not important, but the idea**
- **ACF can be calculated in frequency domain, because convolution in time domain is multiplication if freq-domain**
  - $r(n) = ifft\{[fft(x(k))]^2\}$, (...an instance of Matlabism(tm))
    where fft and ifft are the Fourier transform and its inverse

- **Writing *X*(*k*)=fft[*x*(*k*)], and substituting ifft for real signals**

$$r(n) = \frac{1}{K} \cdot \sum_{k=0}^{K-1} \left[|X(k)|^2 \cdot \cos\left(\frac{2\pi nk}{K}\right) \right]$$

- **Calculations are illustrated below for the case when *n* corresponds to the true pitch period**



Power spectrum and the weights of ACF calculation, when n corresponds to true pitch period.

- **<u>Conclusion</u>: ACF pitch algorithm emphasizes harmonically related spectral components according to their *places***

Cepstrum pitch detection is closely analogous to ACF

- **Cepstrum:** $c(n) = ifft\{\log|fft\{x(k)\}|\}$
- **Simply replace ()^2 in ACF with log() in cepstrum**

Difference is quantitative

- **log() gives dynamic compression to spectrum**
  - flattens the spectra of exotic sounds —> robustness for formants etc. :)
  - rises the noise level :(
- **()^2 emphasizes spectral peaks in relation to noise**
  - —> noise robustness :)
  - further strengthens spectral pecularities of sounds :(

“SPECTRAL INTERVAL” METHODS

Spectrum autocorrelation has been successfully used in several pitch estimators

- **Periodic but non-sinusoidal signal has a periodic magnitude spectrum, the period of which is F0**
- **Mathematically, ACF *R*(*k*) over the positive frequencies of a *K*-length magnitude spectrum *X*(*k*):**

$$R(n) = \frac{2}{K} \cdot \sum_{k=0}^{K/2-n-1} |X(k)||X(k+n)|$$

(again, equations are not important, but the idea)

Conclusion: spectral components are not picked according to their place, but any two components with a *spectral interval n* give an increase to *R*(*k*)

- **Spectrum can be arbitrarily shifted without affecting the output value**

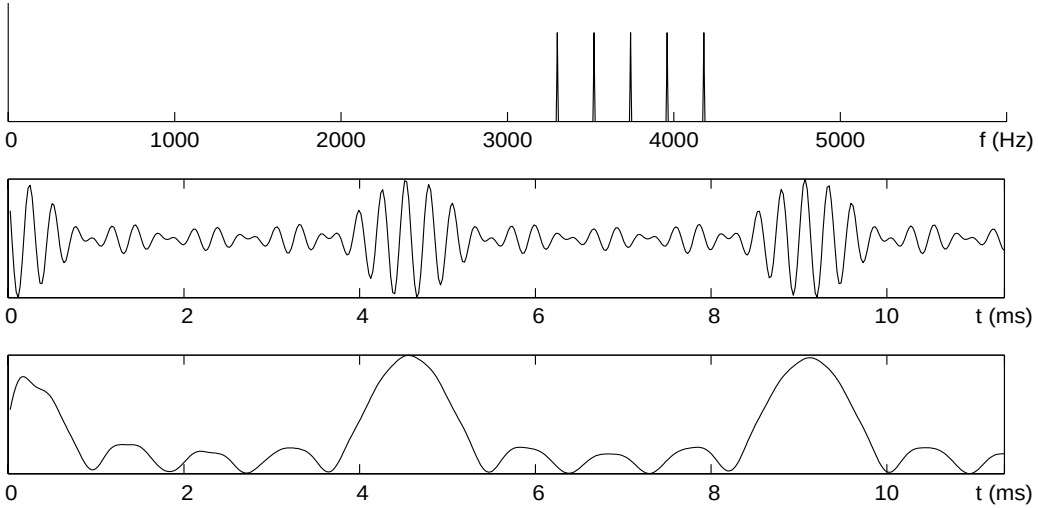
ENVELOPE PERIODICITY ALGORITHMS

A third fundamentally different algorithm

- **Succesfully used in several recent pitch algorithms**
- **Especially used in the field of psychoacoustic research**

Idea

- **Any signal *x*(*k*) with more than one frequency component exhibits periodic fluctuations, *beating*, in its time domain amplitude envelope $\xi(k)$**
- **Rate of beating depends on the amplitude difference between each two frequency component**
- **In the case of a harmonic sound, interval F0 will dominate, and the period is clearly visible in the amplitude envelope, see the figure below**



A signal containing the harmonics 15-19 of a 220 Hz fundamental. Reading top-down: (1) magnitude spectrum, (2) time domain signal, (3) amplitude envelope of the signal.

- **Usually ACF is used to detect periodicity in the amplitude envelope**
- **<u>Conclusions</u>:**
  - more spectral interval oriented (rate of beating depends on the frequency difference), also places (freq.) of low components affect
  - implicit spectral smoothing: amplitude of the beating caused by two sinusoids is determined by the minimum of the two amplitudes —> for a harmonic sound, each two neighboring harmonics contribute to the beating at F0 rate, but single clearly higher amplitude partials are "filtered out"

FINALLY: “UNITARY” ALGORITHM

Psychoacoustically the most relevant method

- **“unitary model”: one algorithm is able to reproduce practically all phenomena observed in human pitch perception**

Algorithm

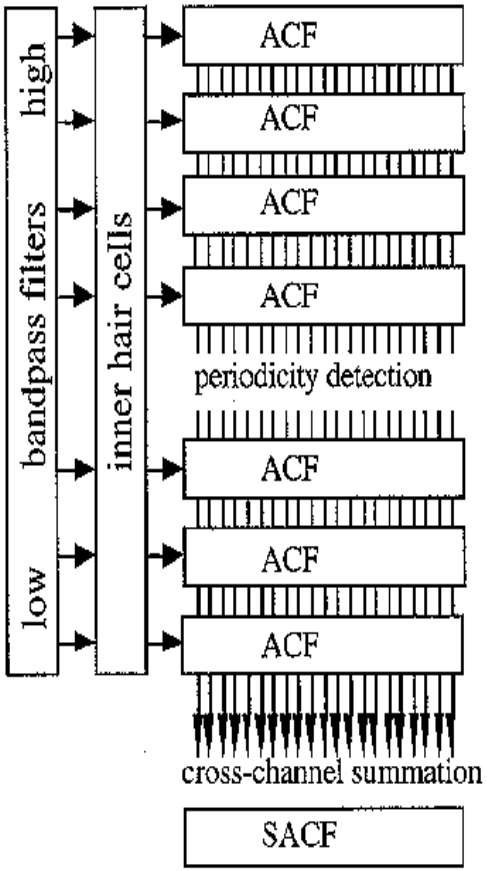
- **Input signal is passed through a bandpass filter bank**
- **At each band, the amplitude envelope $\xi_{band}(k)$ is calculated**
  - half-wave rectify: $x(k)=\max\{x(k),0\}$
  - lowpass filter, retain frequencies below about 1000 Hz —> $\xi_{band}(k)$
- **At each band, ACF_{band} (periodicity) of $\xi_{band}(k)$ is calculated**
- **Calculate summary autocorrelation function**

$$SACF(n) = \sum_{band=0}^{B-1} [ACF_{band}(n)]$$

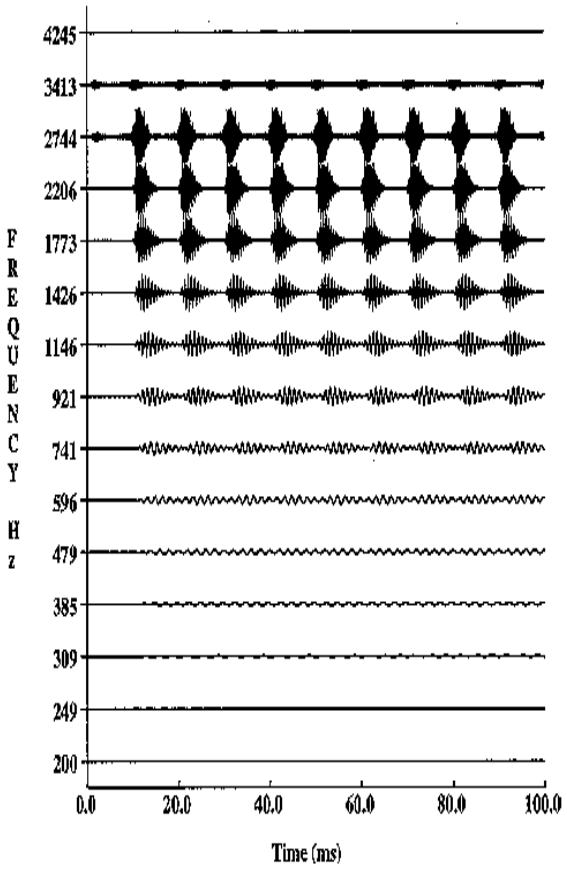
The maximum value of SACF indicates the pitch

Illustrations of the unitary model

Algorithm



Outputs of the filterbank



ACFs and SACF at the bottom

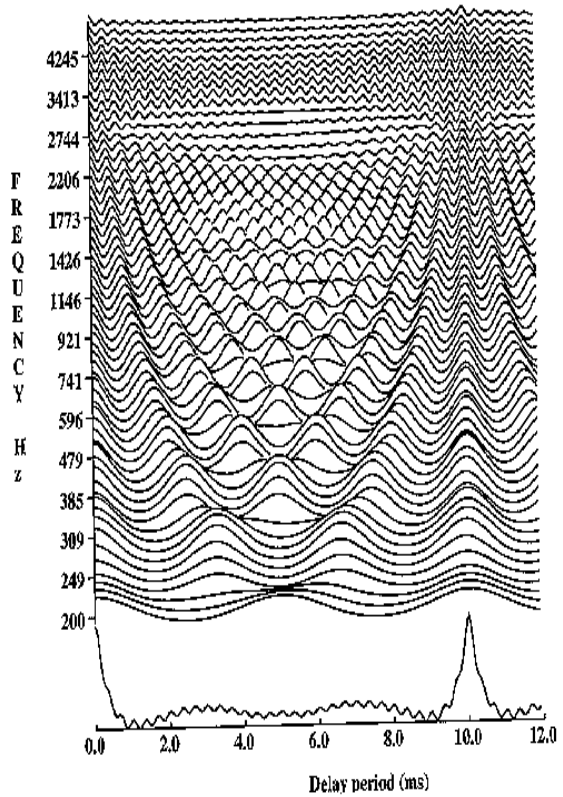
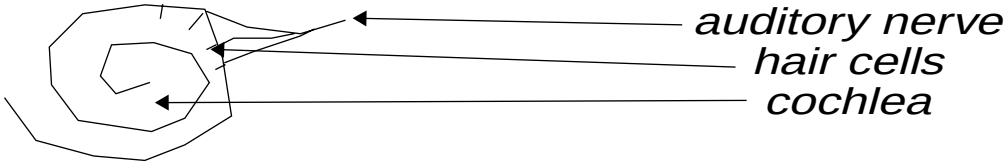


FIG. 4.3. Autocorrelation functions and summary autocorrelation functions produced by the model shown in Figure 4.2. The input is a 30 harmonic complex with a fundamental frequency of 100 Hz.

— this is an "bonus" slide, which you may ignore —

Mechanics of the inner ear (simplified)

- ***Cochlea* (simpukka) is tonotopically organized**
  - high and low frequency excite different parts of the cochlea
  - —> modeled by the filterbank in unitary model
- **Distributed along the cochlea are *hair cells* that transform vibrations to neural impulses (~extract amplit. envelopes)**



Psychoacoustic theories of pitch

- **Place theory: pitch is resulted from the tonotopic organization of cochlea, i.e., from the excitation of certain *place(s)* of the cochlea**
- **Timing theory: neural impulses from hair cells form a timing pattern which encodes frequency (~autocorrelation)**
- **—> Unitary model is a mixture of these two**

MULTIPITCH ESTIMATION (MPE)

Musical signals demonstrate that human listeners are able to hear multiple pitches at the same time

- **Computational modeling of this function has been little explored compared to single pitch estimation**

Single pitch estimation methods are not appropriate as such for MPE

- **Algorithms get confused by the other sounds**
- **Problem: sounds occupy same time-frequency regions —> it is difficult to organize partials to their due sources**

MPE has a strong connection to CASA

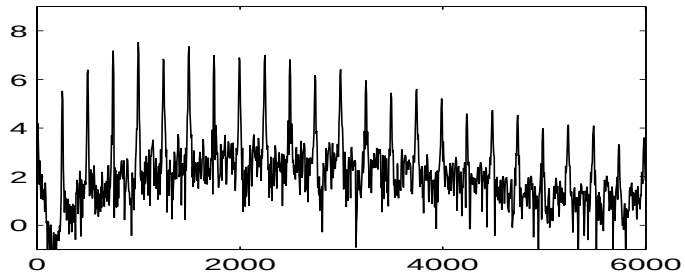
- **“Any theory of pitch computation that can compute the pitch of a sound and not get confused by other co-occurring souds would be, in effect, doing CASA by harmonicity principle” –*Bregman***

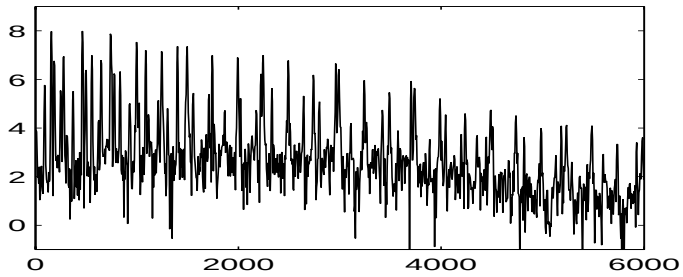
Double function in pitch perception

- **Group spectral components to their sources of production**
- **Assign a pitch value to each group**

Figure: spectrum of a single harmonic sound (left) and a mixture of four sounds (right)

- **Difficulty levels are different...**





SHORT REVIEW OF MPE ATTEMPTS

1. Music is “home field” for MPE, as speech processing is for single pitch estimation
   - First attempts in the automatic transcription of music
   - Martin ~1996: use of musical knowledge (4-voice piano)
   - Kashino ~1995: several instruments, 3-voice
   - Goto ~1999: melody and bass lines from CD-recordings
2. Psychoacoustic knowledge applied
   - Brown, Cooke ~1994: perceptual grouping of musical sounds
   - Godsmark ~1998: Bregman’s spectral organization principles applied
   - Cheveigne ~1999: iterative MPE approach, performance was poor
3. Purely mathematical (Sethares: “periodicity transforms”)

—> All these are nice attempts and models, but not accurate and reliable enough for real applications

CASE: KLAP'S TRANSCRIBER

Problems and requirements

- **Wide pitch range, rich spectra (exotic sounds)**
  - Pitch range in speech: 60-600 Hz; in music: 50-4000 Hz
  - Musical instruments: many sound producing mechanisms
- **Robustness in interference**
  - Noise and other sounds are usually present;
    Polyphony: *multi*pitch estimation
- **Inharmonic sounds**
  - For many physical vibrators, frequency partials are not exactly in harmonic relations (string instruments etc.)
    —> partials cannot be assumed at harmonic positions
    —> makes mixture spectra even more confusing
- **Computational efficiency**
  - Unitary model requires bandwise processing

CASE...: OBJECTIVES

Robustness

- **In rich polyphonies, no upper limit**
- **For noise**
- **For corrupted signals (only part of the spectrum available)**
- **Non-ideal harmonic sounds handled without problems**

Generality

- **26 musical instruments, sung vowels**
- **Regularities of music are not utilized**
  —> applies to CASA in general

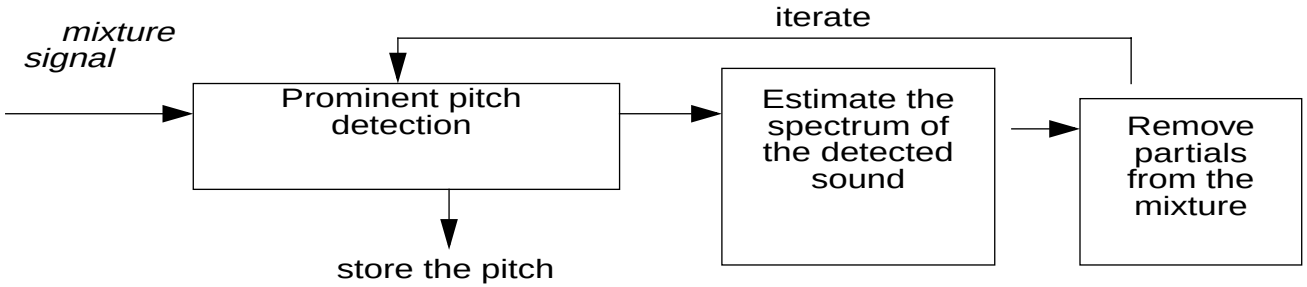
Computational practicality

In a single time frame —> based on harmonicity

PROBLEM SOLVING STRATEGY

(I) *Iterative* estimation–separation–estimation

- **Provides robustness in rich polyphonies**
- **At least a couple of most prominent sounds are detected even in 10-voice polyphonies**
- ***Predominat pitch detection*: find one (any) of the correct F0s in the presence of several sounds**



Psychoacoustically motivated

- **Human ear is somehow able to group components, remove them from the mixture, and continue with the residual**
- **Also used by some other researchers (e.g. Cheveigne)**

(II) Predominant pitch algorithm: unitary model with some crucial modifications

- **Benefits of bandwise processing**
  - enables the analysis of inharmonic sounds
  - bandlimited noise can be handled flexibly
- **Modification 1: Bandwise processing is done in the frequency domain**
  —> local regions of the spectrum are separately processed
  —> computationally efficient
- **Modification 2: Pitch calculations are based *only* on the harmonically related spectrum components, not on the overall spectrum**
  - —> does not get confused by other co-occurring sounds

In brief: probabilities of each F0 value are calculated independently at 18 distinct frequency bands, and the results are then combined to a global choice for F0

(III) *Machine learning* used to find optimal parameters

- **Well posed learning problem consists of**
  - *Task* : to transcribe musical polyphonies from signal to score
  - *Performance measure* : percentage of correct notes, or more complicated
  - *Experience* from which to learn : automatic compilation of semi-random polyphonic signals —> {signal, score} pairs are available
- **Sample database: sung vowels + musical instruments**

piano	clarinets (bass, bb, eb)	trumpets	flute, alto, bass
guitar		bassoons	piccolo
violin, viola	oboe	trombones (bass,tenor,alto)	saxophones (sop, alto, tenor, barit, bass)
cello	bassoons		
double	english horn	tuba	

- **—> System transcribes a piece using certain parameters, compares results to correct ones, and refines the params**
- **—> Optimal performance of a model can be checked**

VALIDATION EXPERIMENTS

Note error rate (NER) metric

- **F0 estimate is correct, if it deviates less than half a semitone (+/- 3%) from the true value (=musical note found)**
- **NER = <number of errors> / <number of F0s in reference>**
- **Errors: missing F0s, wrong F0s, extraneous F0s**

Random note combinations

1. **Select an instrument randomly**
2. **Allot a note from the whole playing range of that instrument, however, restrict between 65 Hz and 2000 Hz**
3. **Repeat until the desired number of sounds are allotted**

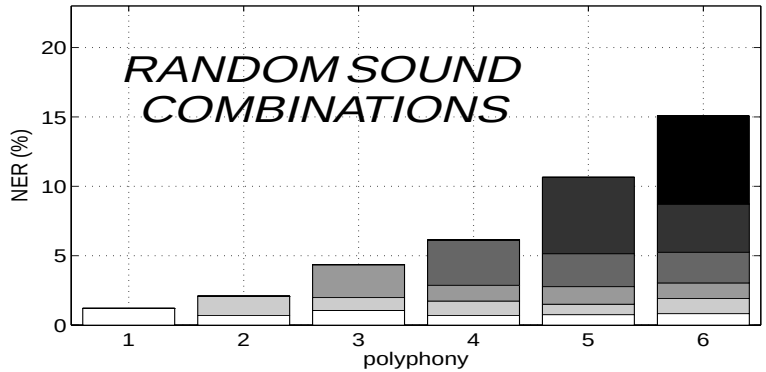
Semirandom musical combinations (more difficult)

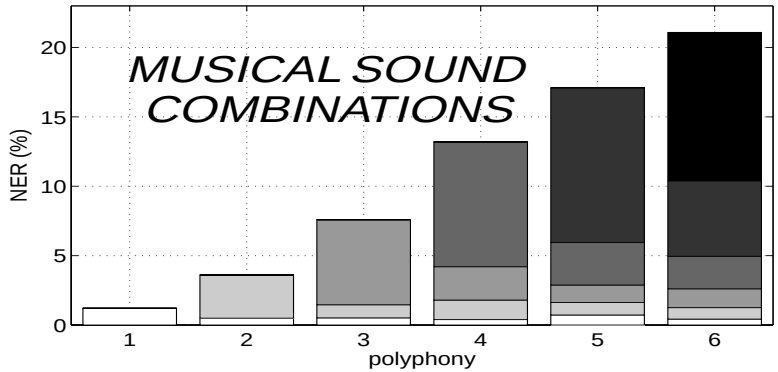
- **More musical intervals (octaves, harmonic chords etc.)**

RESULTS

Note error rates (%) as a function of polyphony

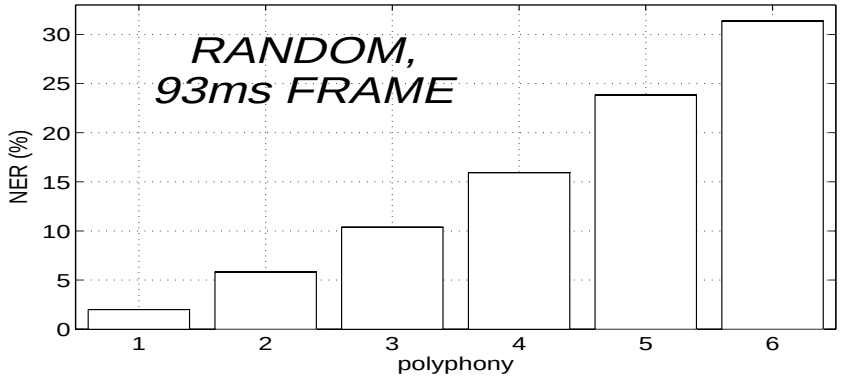
- **Polyphony = number of simultaneous sounds**
- **Mixtures are resolved in a single 190 ms time frame —> long frame is needed to resolve mixtures of low F0s**
- **Bars represent the overall NER, and their different colors the error cumulation in iteration, NER of the first detection at bottom**





- **—> last found sound accounts for almost half of all errors (= there is often one sound that is buried under others)**

Results using shorter (93 ms) time frame



Sound examples:

- **http://www.cs.tut.fi/~klap/iiro/**