Fundamental Frequency Detection

Jan Černocký, Valentina Hubeika

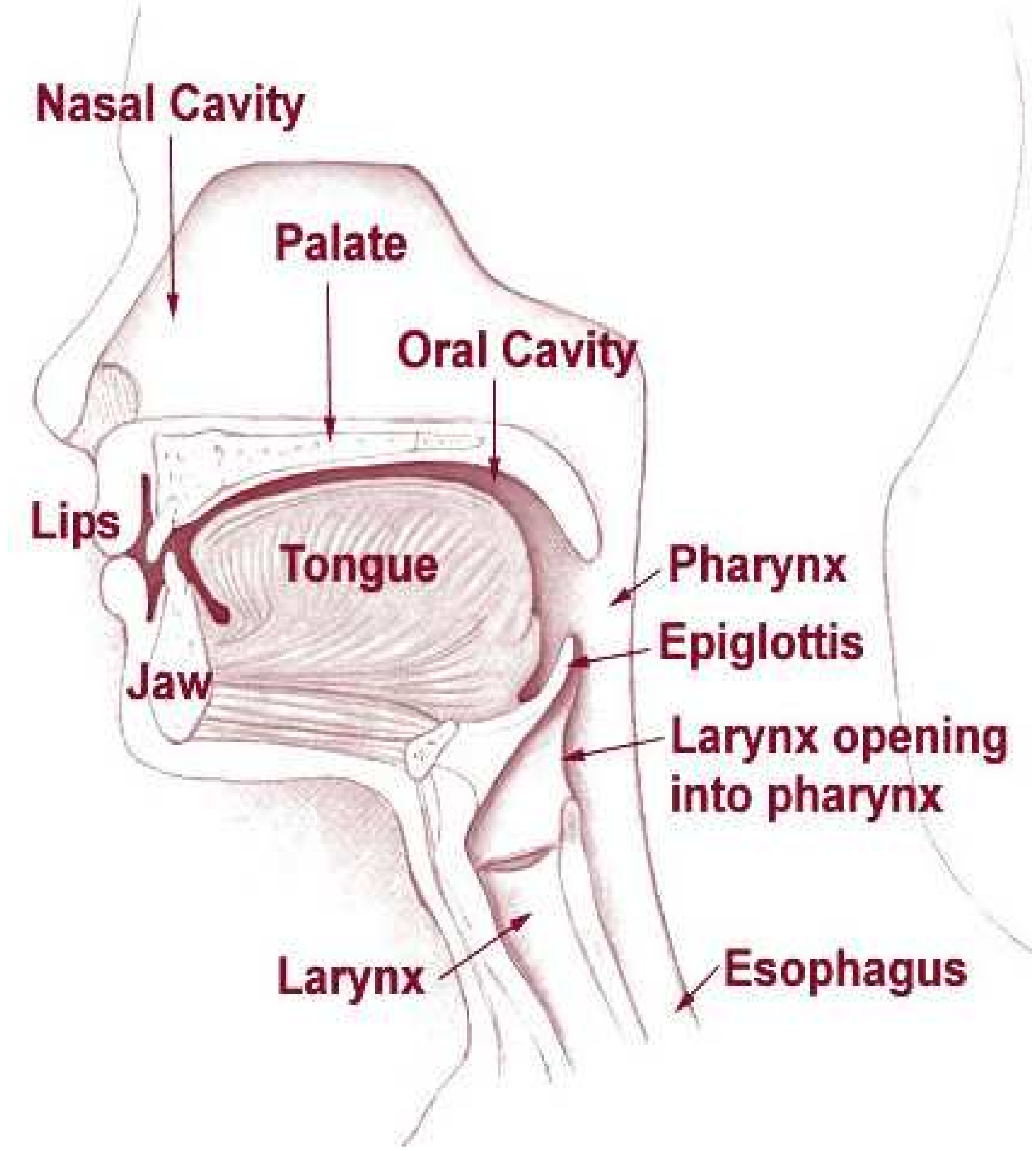
{cernocky|ihubeika}@fit.vutbr.cz

DCGM FIT BUT Brno

Agenda

- Fundamental frequency characteristics.
- Issues.
- Autocorrelation method, AMDF, NFFC.
- Formant impact reduction.
- Long time predictor.
- Cepstrum.
- Improvements in fundamental frequency detection.

Recap – speech production and its model



Introduction

- Fundamental frequency, *pitch*, is the frequency vocal cords oscillate on: F_0.
- The period of fundamental frequency, *(pitch period)* is $T_0 = \frac{1}{F_0}$.
- The term "*lag*" denotes the pitch period expressed in samples : $L = T_0 F_s$, where F_s is the sampling frequency.

Fundamental Frequency Utilization

- **speech synthesis** – melody generation.
- **coding**
  - in simple encoding such as LPC, reduction of the bit-stream can be reached by separate transfer of the articulatory tract parameters, energy, voiced/unvoiced sound flag and pitch F_0.
  - in more complex encoders (such as RPE-LTP or ACELP in the GSM cell phones) **long time predictor LTP** is used. LPT is a filter with a "long" impulse response which however contains only few non-zero components.

Fundamental Frequency Characteristics

- F_0 takes the values from 50 Hz (males) to 400 Hz (children), with F_s=8000 Hz these frequencies correspond to the lags L=160 to 20 samples. It can be seen, that with low values F_0 approaches the frame length (20 ms, which corresponds to 160 samples).
- The difference in pitch within a speaker can reach to the 2:1 relation.
- Pitch is characterized by a typical behaviour within different phones; small changes after the first period characterize the speaker ($\Delta F_0 < 10$ Hz), but difficult to estimate. In radio-techniques these small shifts are called “jitter”.
- F_0 is influenced by *many factors* – usually the melody, mood, distress, etc. Values of the changes of F_0 are higher (greater voice “modulation”) in case of professional speakers. Common people speech is usually rather monotonous.

Issues in Fundamental Frequency Detection

- Even voiced phones are never purely periodic! Only clean singing can be purely periodic. Speech generated with F_0=const is monotonous.
- Purely voiced or unvoiced excitation does not exist either. Usually, excitation is compound (noise at higher frequencies).
- Difficult estimation of pitch with low energy.
- High F_0 can be affected by the low formant F_1 (females, children).
- During transmission over land line (300–3400 Hz) the basic harmonic of pitch is not presented but its folds (higher harmonics). So simple filtering for purpose of capturing pitch would not work. . .

Methods Used in Fundamental Frequency Detection

- Autocorrelation + NCCF, which is applied on the original signal, further on the so-called clipped signal and linear prediction error.
- Utilization of the error predictor in linear prediction.
- Cepstral method.

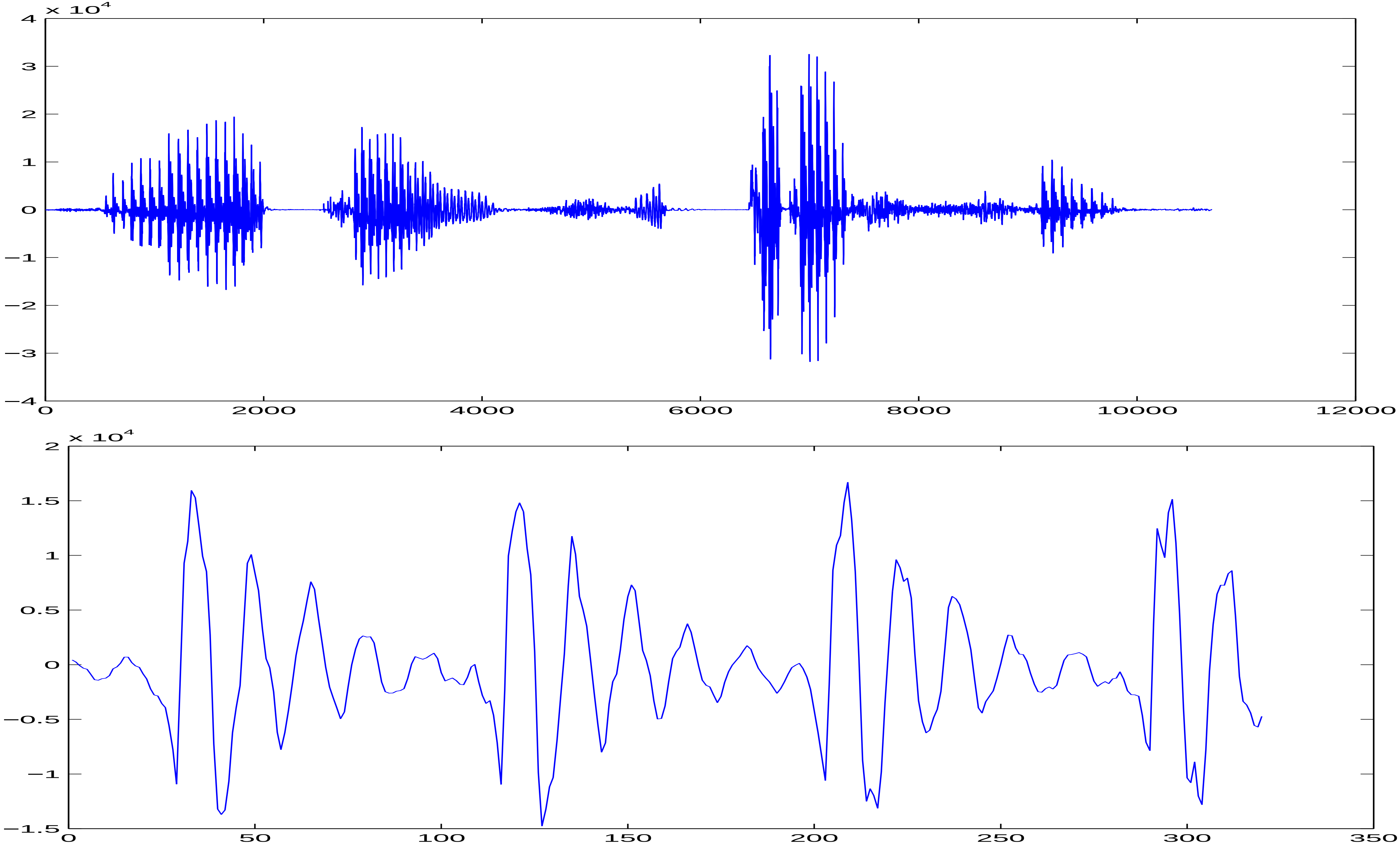
Autocorrelation Function – ACF

$$R(m) = \sum_{n=0}^{N-1-m} s(n)s(n+m) \tag{1}$$

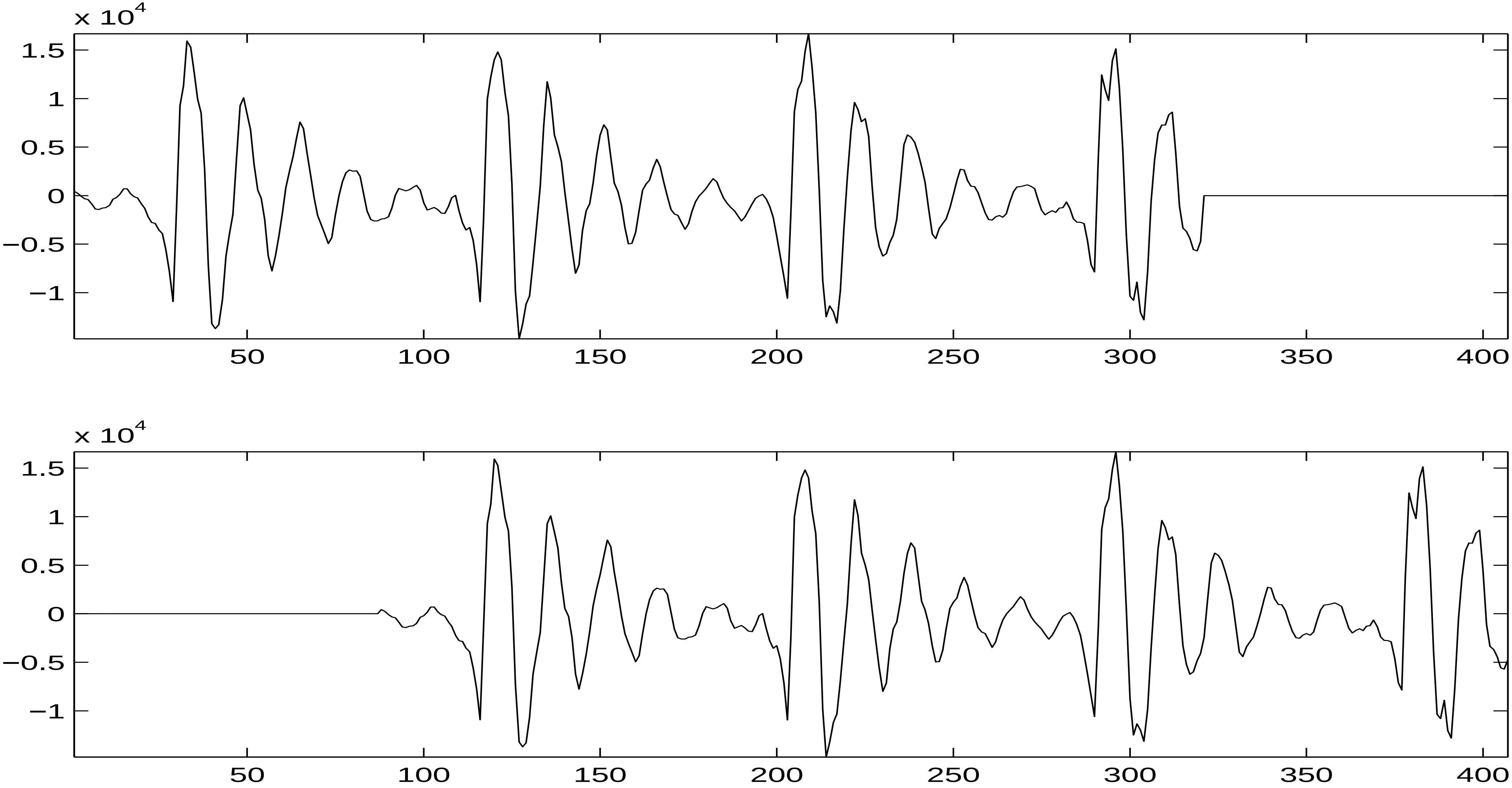
The symmetry property of the autocorrelation coefficients gives:

$$R(m) = \sum_{n=m}^{N-1} s(n)s(n-m) \tag{2}$$

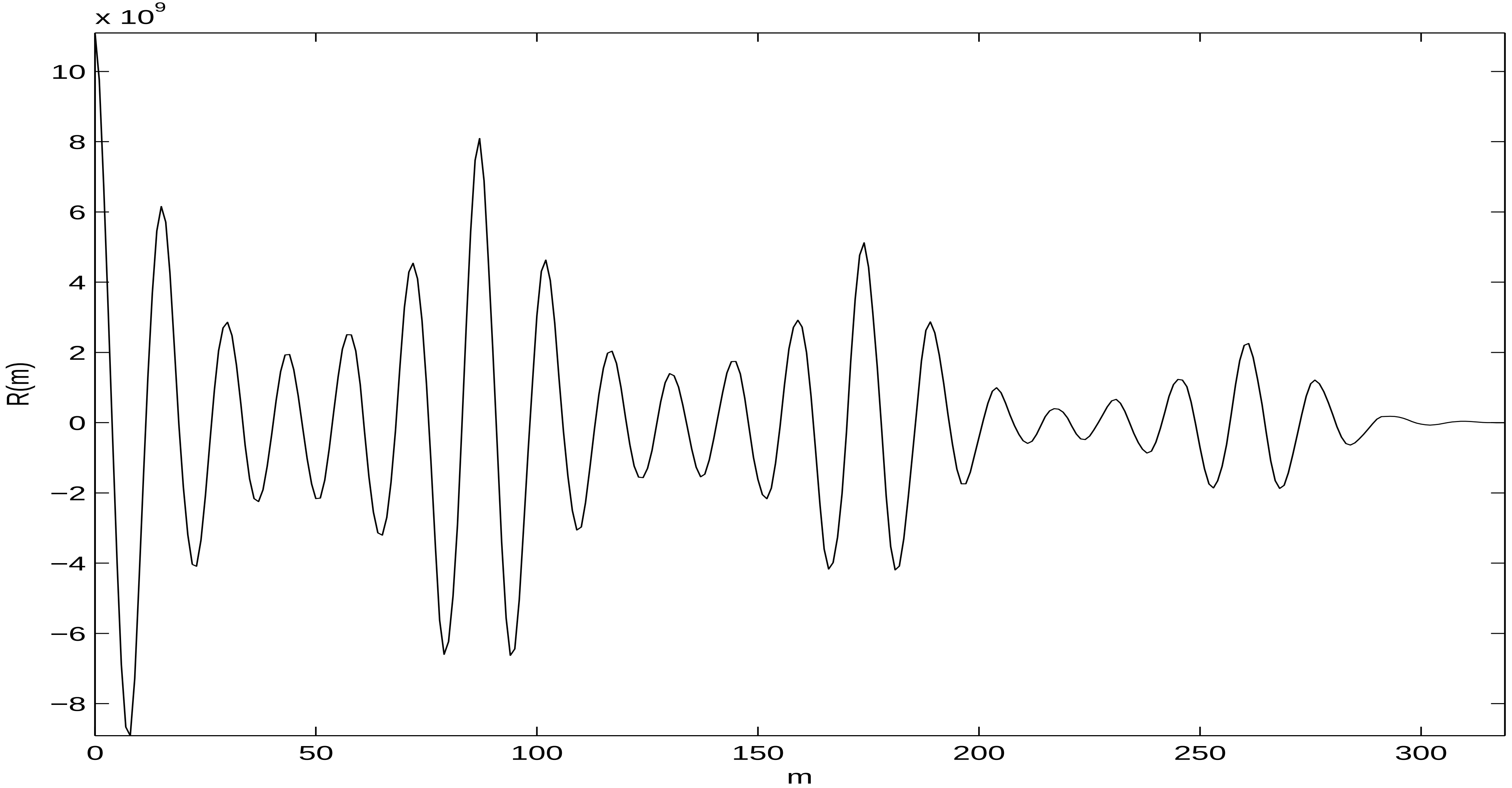
The whole signal and one frame of a signal



Shift illustration



Calculated autocorrelation function



Lag Estimation. Voiced/Unvoiced Phones

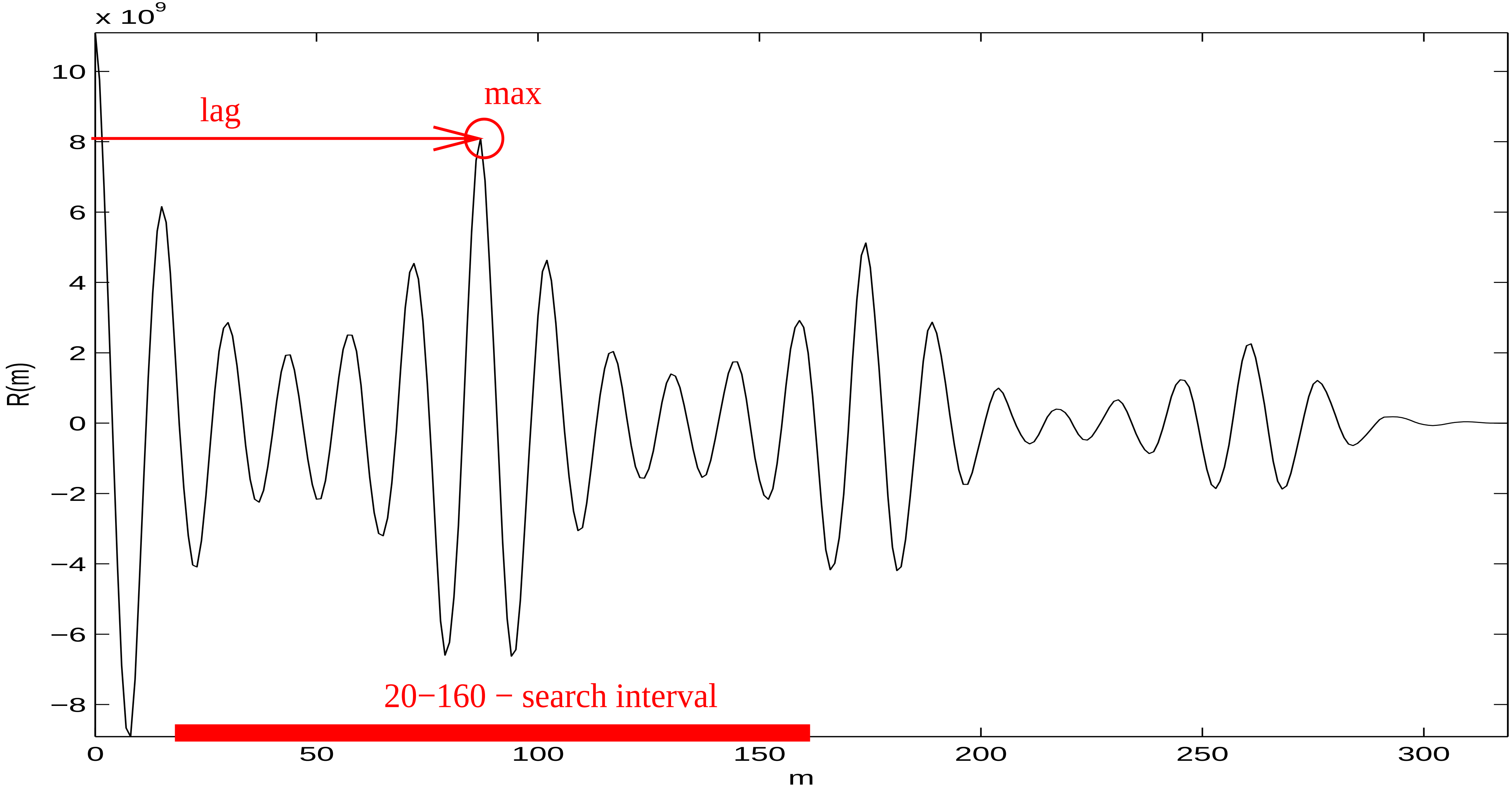
Lag estimation using ACF. Looking for the maximum of the function:

$$R(m) = \sum_{n=0}^{N-1-m} c[s(n)]c[s(n+m)] \qquad (3)$$

Phones can be determined as voice/unvoiced by comparing the found maximum to the zero's (maximum) autocorrelation coefficient. The constant α must be chosen experimentally.

$$\begin{array}{lcr} R_{max} < \alpha R(0) & \Rightarrow & \text{unvoiced} \\ R_{max} \geq \alpha R(0) & \Rightarrow & \text{voiced} \end{array} \qquad (4)$$

ACF Maximum estimation ⇒ lag (L=87 for the given figure):

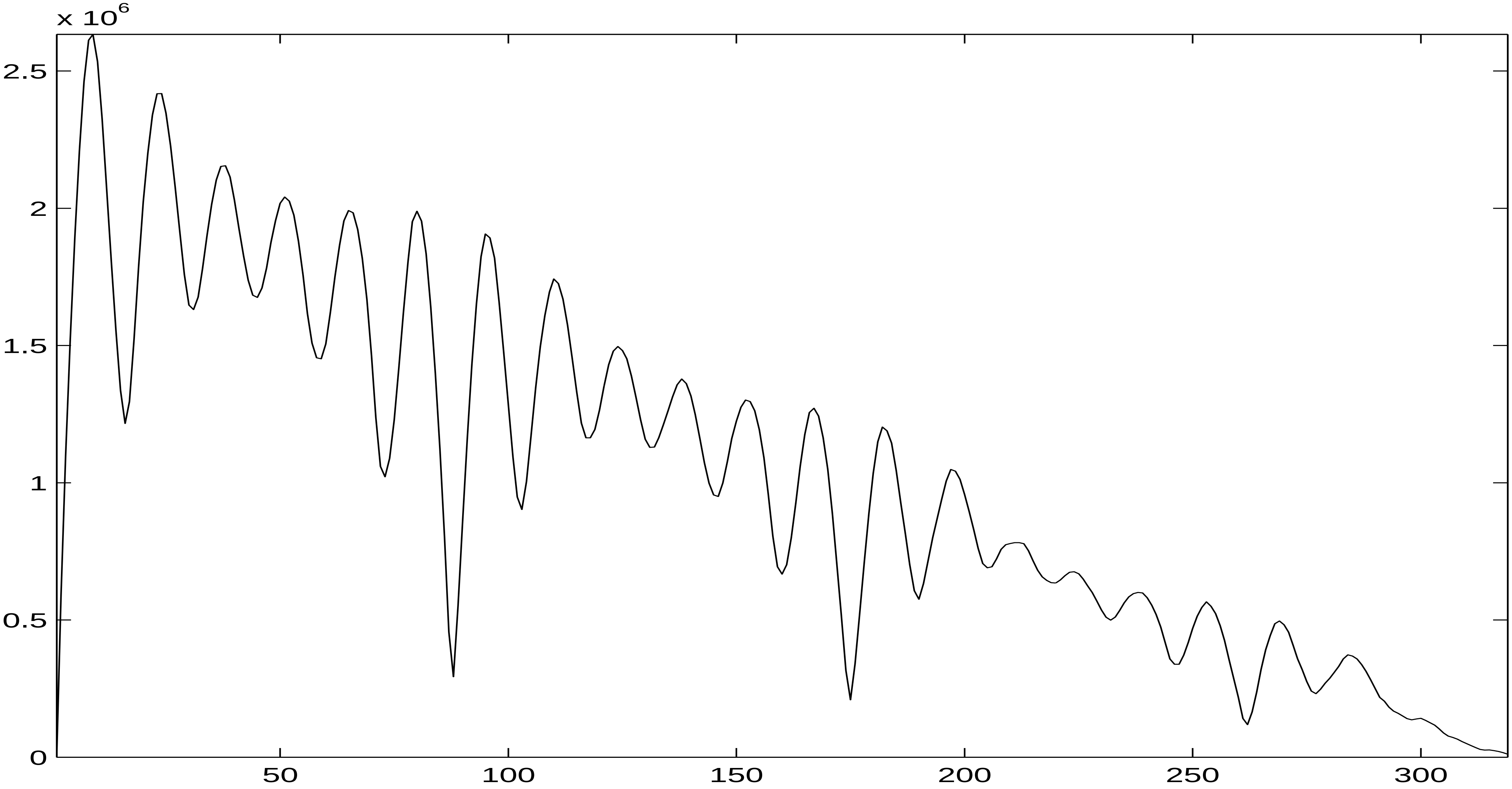


AMDF

Earlier, when multiplication was computationally expensive, the autocorrelation function was substituted with AMDF (Average Magnitude Difference Function):

$$R_D(m) = \sum_{n=0}^{N-1-m} |s(n) - s(n+m)|, \qquad (5)$$

where on the contrary the *minimum* had to be identified.

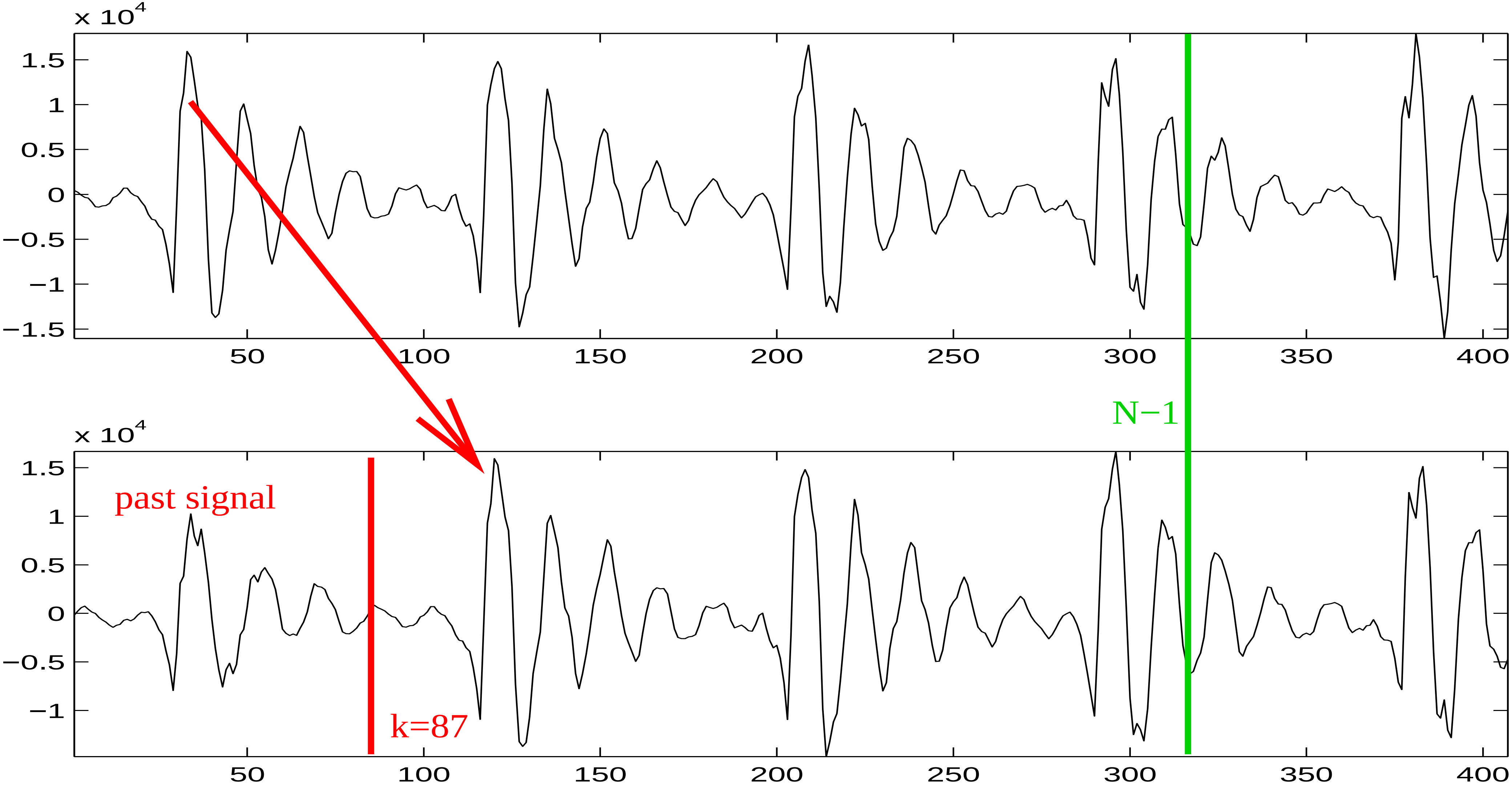

× 10^6
2.5
2
1.5
1
0.5
0
50
100
150
200
250
300

Cross-Correlation Function

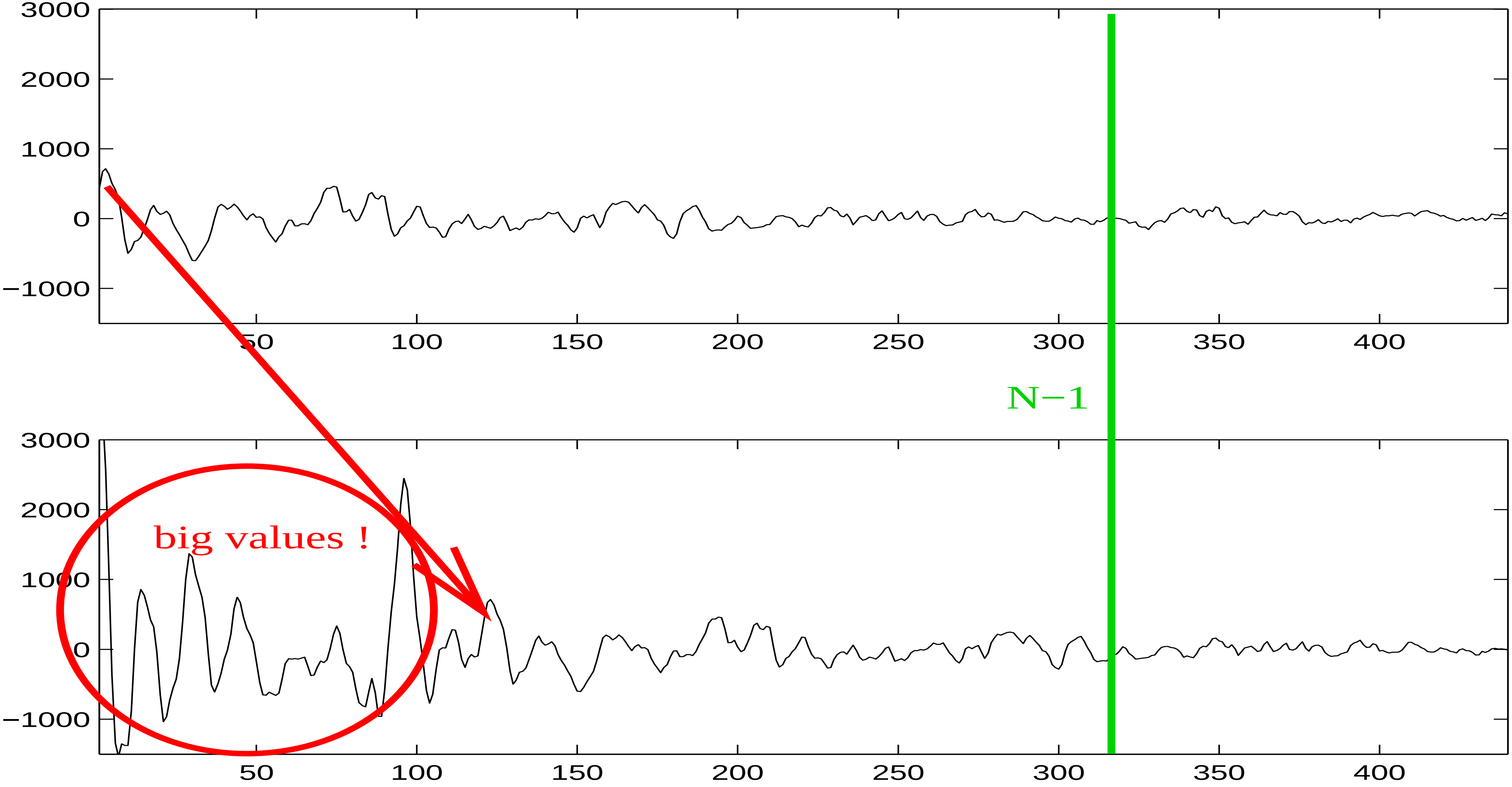
The drawback of the ACF is a stepwise "shortening" of the segment, the coefficients are computed from. Here, we want to use the whole signal ⇒ **CCF**. b indicates the beginning of the frame:

$$CCF(m) = \sum_{n=b}^{b+N-1} s(n)s(n-m) \tag{6}$$

The shift in the CCF calculation:



The CCF shift is a problem as the shifted signal has much higher energy!



Normalized cross-correlation function

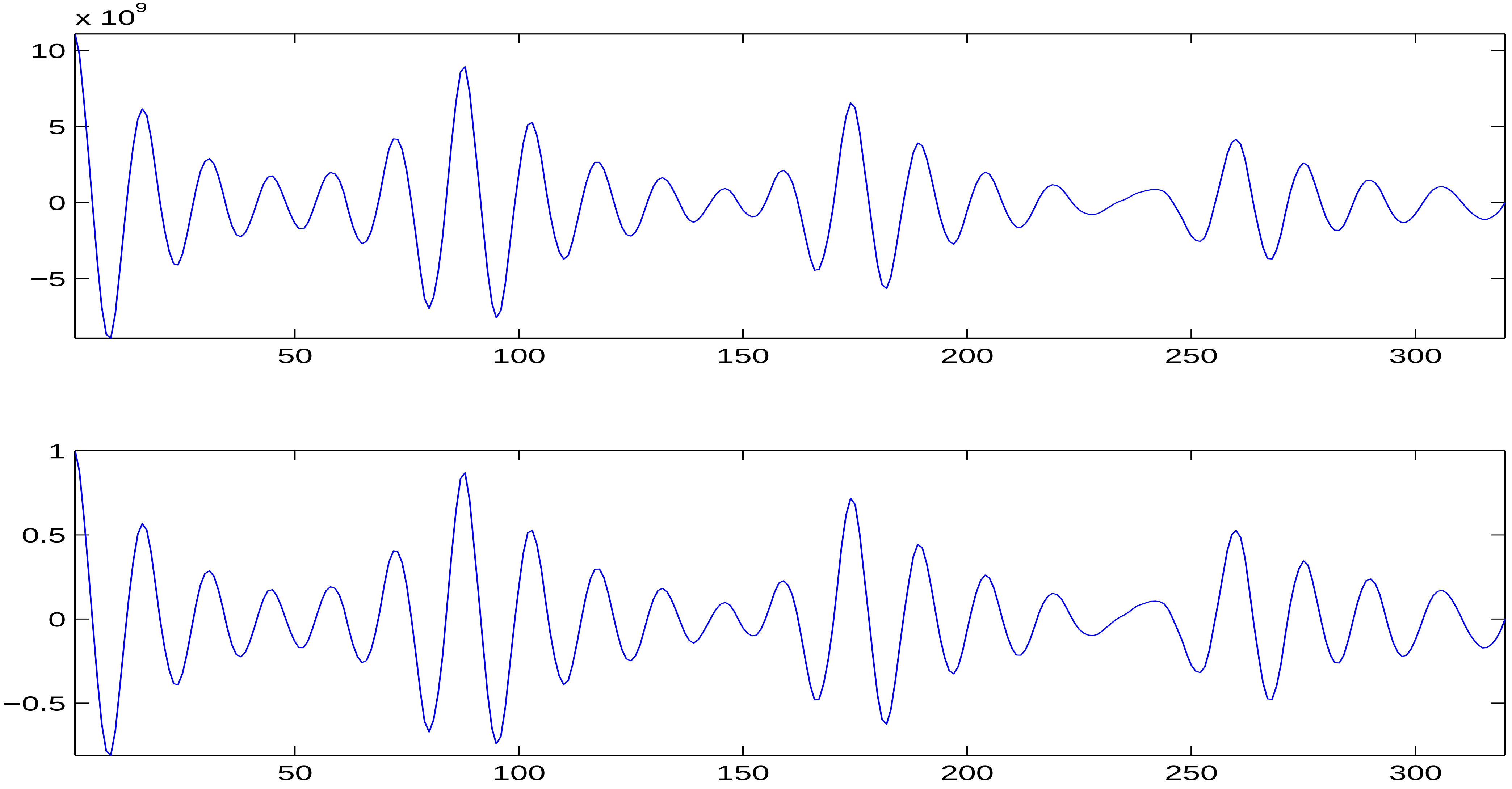
The difference of the energy can be compensated by normalization: **NCCF**

$$CCF(m) = \frac{\sum_{n=zr}^{zr+N-1} s(n)s(n-m)}{\sqrt{E_1 E_2}} \qquad (7)$$

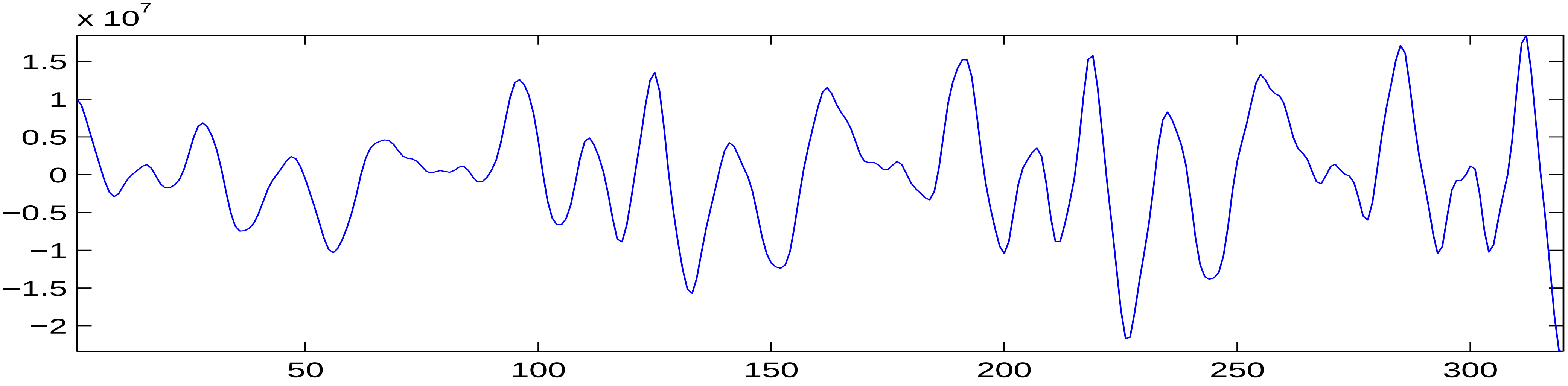
E_1 a E_2 are the energies of the original and the shifted signal:

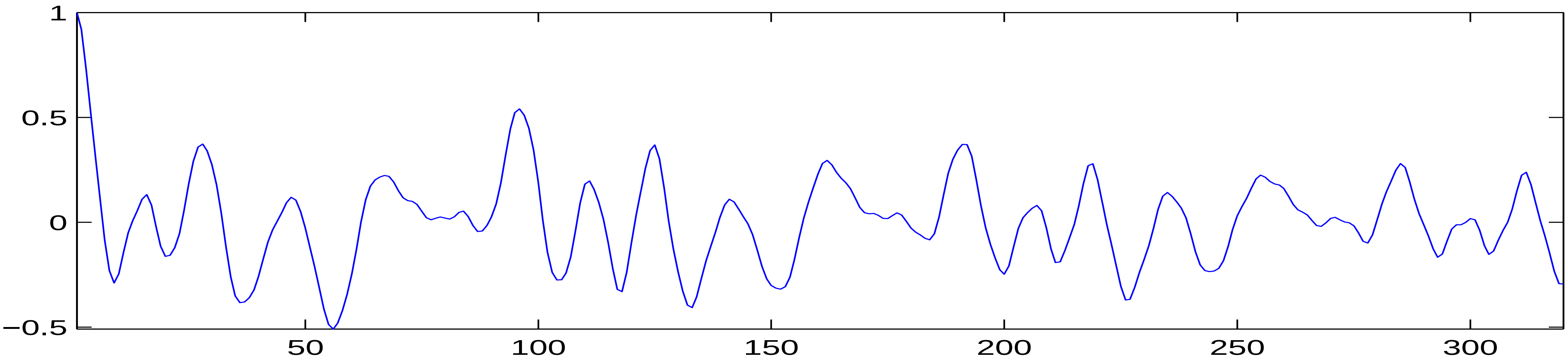
$$E_1 = \sum_{n=zr}^{zr+N-1} s^2(n) \qquad E_2 = \sum_{n=zr}^{zr+N-1} s^2(n-m) \qquad (8)$$

CCF and NCCF for a "good example"



CCF and NCCF for a “bad example”





Drawback: the methods do not suppress the formants influence (results additional maxima in ACF or AMDF).

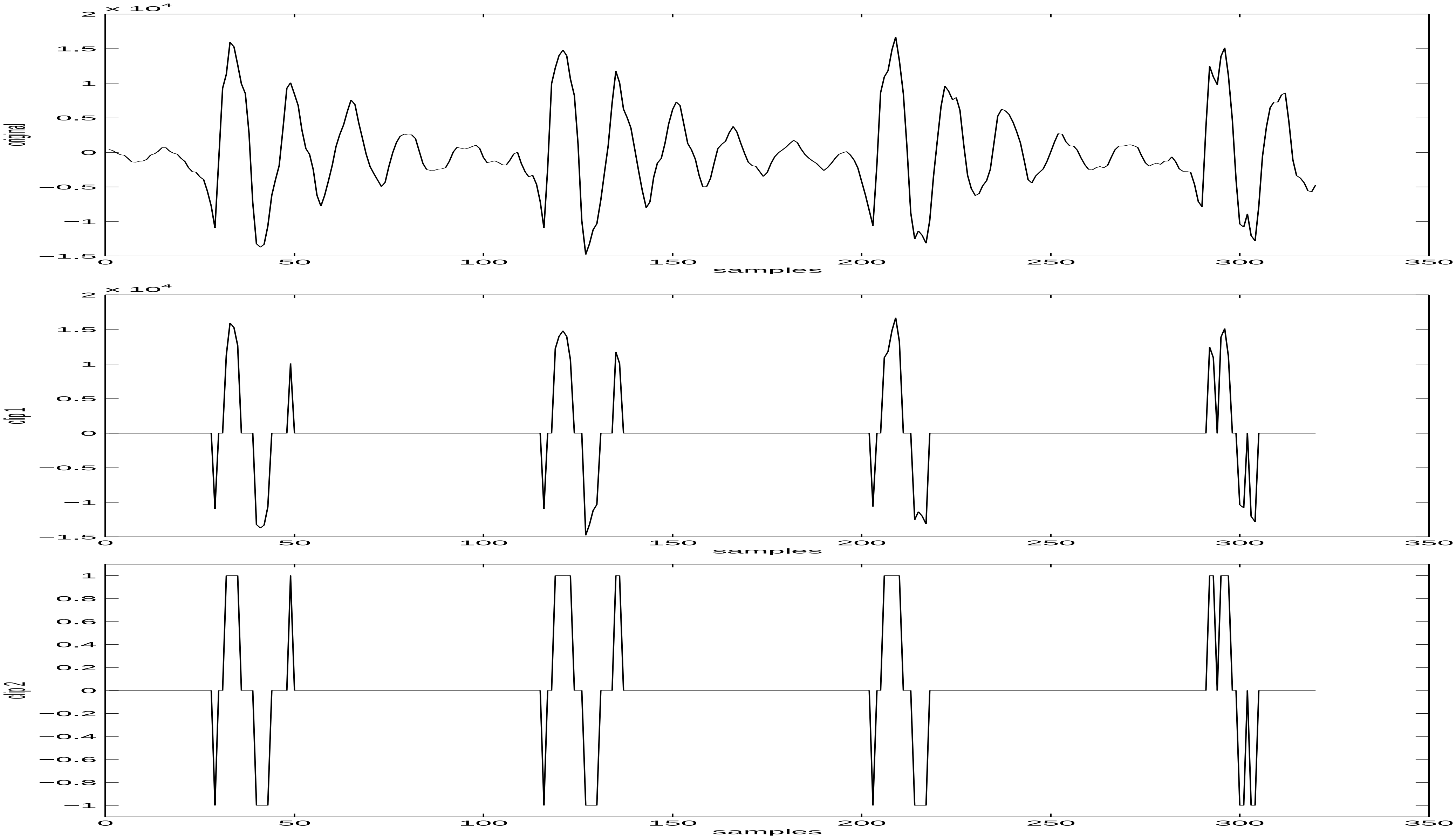
Center Clipping

- a signal preprocessing before ACF: we are interested only in the *signal picks*. Define so called clipping level c_L. We can either "leave out" the the values from the interval $< -c_L, +c_L >$. Or we can substitute the values of the signal by 1 and -1 where it crosses the levels c_L and $-c_L$, respectively:

$$c_1[s(n)] = \begin{cases} s(n) - c_L & \text{pro} \quad s(n) > c_L \\ 0 & \text{pro} \quad -c_L \leq s(n) \leq c_L \\ s(n) + c_L & \text{pro} \quad s(n) < -c_L \end{cases} \tag{9}$$

$$c_2[s(n)] = \begin{cases} +1 & \text{pro} \quad s(n) > c_L \\ 0 & \text{pro} \quad -c_L \leq s(n) \leq c_L \\ -1 & \text{pro} \quad s(n) < -c_L \end{cases} \tag{10}$$

Figures illustrate clipping into frames on a speech signal with the clipping level 9562:



Clipping Level Value Estimation

As a speech signal $s(n)$ is a nonstationary signal, the slipping level changes and it is necessary to estimate it for every frame, for which pitch is predicted. A simple method is to estimate the clipping level from the absolute maximum value in the frame:

$$c_L = k \max_{n=0\dots N-1} |x(n)|, \tag{11}$$

where the constant k is selected between 0.6 and 0.8. Further, subdivision into several micro-frames can be done, for instance $x_1(n)$, $x_2(n)$, $x_3(n)$ of one third of the original frame length. The clipping level is then given by the lowest maximum from the micro-frames:

$$c_L = k \min \left\{ \max |x_1(n)|, \max |x_2(n)|, \max |x_3(n)| \right\} \tag{12}$$

Issue: clipping of noise in pauses, where subsequently can be detected pitch. The method therefore should be preceded by the silence level s_L estimation. In the maximum of the signal is $< s_L$, then the frame is not further processed.

Utilization of the Linear Prediction Error

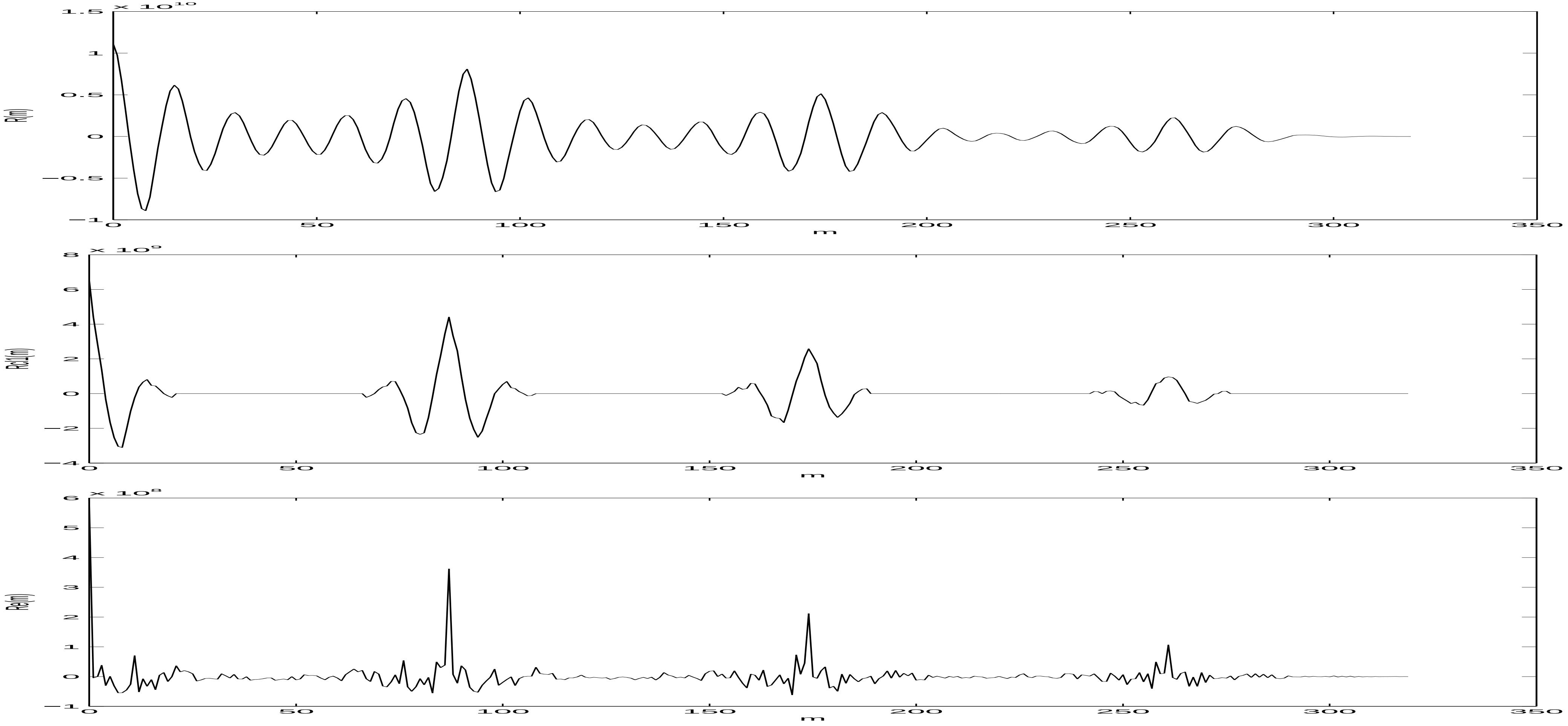
Preprocessing method (not only for the ACF, used as well in other pitch estimation algorithms). Recap: the linear prediction error is given as the difference between the true sample and the estimated sample:

$$\begin{aligned}
e(n) &= s(n) - \hat{s}(n) && (13)\\
E(Z) &= S(z)[1-(1-A(z))] = S(z)A(z) && (14)\\
e(n) &= s(n) + \sum_{i=1}^{P} a_i s(n-i) && (15)\\
& && (16)
\end{aligned}$$

The signal $e(n)$ contains no information about formants, thus is more suitable for the estimation. Lag estimation from the error signal can be done using the ACF method, etc..

Autocorrelation Functions Comparison

The following figure presents the autocorrelation functions calculated from the original signal, from the clipped signal and from the linear prediction error signal.



Prediction Error Long Time Predictor for Pitch Estimation

The aim is to estimate the nth sample from two samples distanced by the assumed lag. The distance with the minimum prediction error energy determines the lag. The predicted error of prediction is:

$$\hat{e}(n) = -\beta_1 e(n-m+1) - \beta_2 e(n-m) \tag{17}$$

Then the predictor error of the prediction error is:

$$ee(n) = e(n) - \hat{e}(n) = e(n) + \beta_1 e(n-m+1) + \beta_2 e(n-m) \tag{18}$$

The aim is to minimize the energy of the signal:

$$\min E = \min \sum_{n=0}^{N-1} ee^2(n) \tag{19}$$

The approach is similar as for LPC coefficients, the coefficients β_1 and β_2 are:

$$\begin{aligned}\beta_1 &= [r_e(1)r_e(m) - r_e(m-1)]/[1-r_e^2(1)] \\ \beta_2 &= [r_e(1)r_e(m-1) - r_e(m)]/[1-r_e^2(1)],\end{aligned} \tag{20}$$

where $r_e(m)$ are normalized autocorrelation coefficients of the error signal $e(n)$. After substituting these coefficients to the energy estimation equation 19, the energy can be estimated as a function of the shift m:

$$E(m) = 1 - K(m)/[1 - r_e^2(1)] \tag{21}$$

$$\text{kde} \quad K(m) = r_e^2(m-1) + r_e^2(m) - 2r_e(1)r_e(m-1)r_e(m) \tag{22}$$

The lag can be determined by identifying either the minimum energy or the maximum of the function $K(m)$ (notice, the nominator $1 - r_e^2(1)$ does not depend on m).

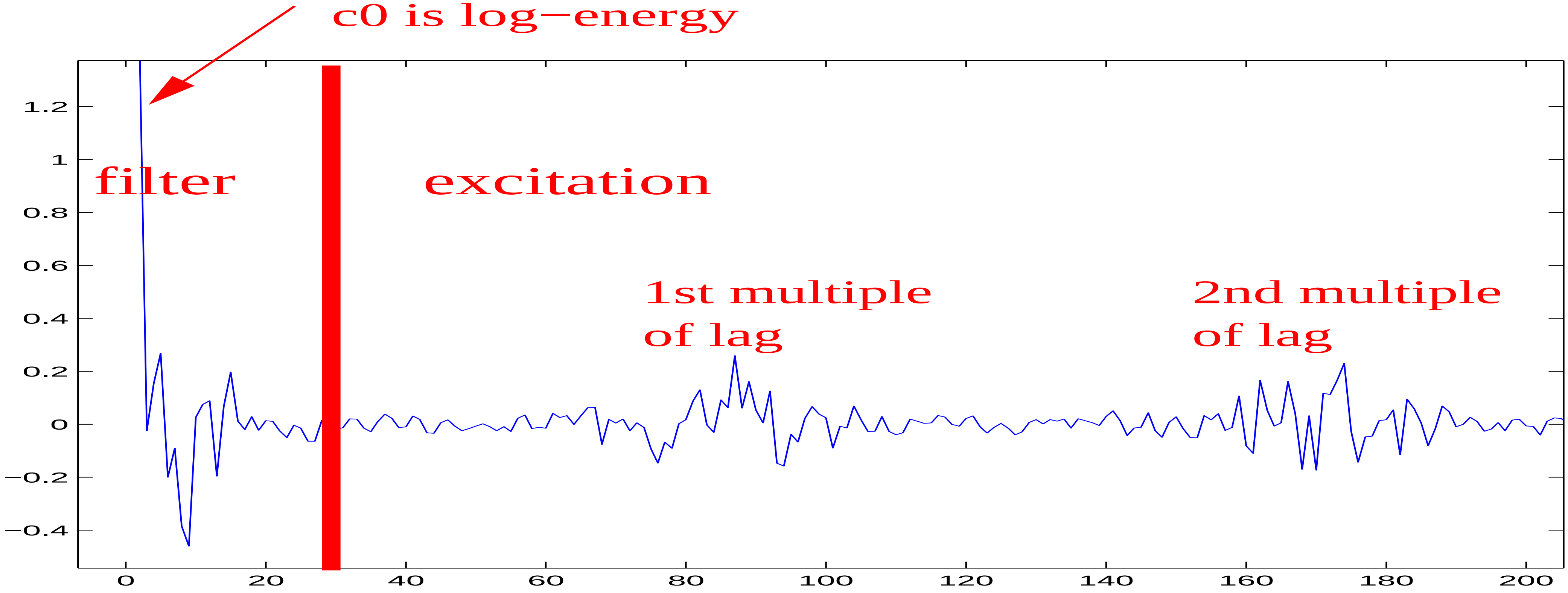
$$L = \arg \min_{m \in [L_{min}, L_{max}]} E(m) = \arg \max_{m \in [L_{min}, L_{max}]} K(m) \tag{23}$$

Cepstral Analysis in Fundamental Frequency Detection

The cepstral coefficients can be acquired using the following relation:

$$c(m) = \mathcal{F}^{-1}\left[ln|\mathcal{F}s(n)|^2\right] \tag{24}$$

In cepstrum, it is possible to separate the coefficients representing the vocal tract (low indices) and the coefficients carrying the information on the fundamental frequency, pitch, (high indices). The lag can be predicted by identifying the maximum of $c(m)$ in the potential range of lag values.


c0 is log−energy
filter
excitation
1st multiple
of lag
2nd multiple
of lag
1.2
1
0.8
0.6
0.4
0.2
0
−0.2
−0.4
0
20
40
60
80
100
120
140
160
180
200

Robust Fundamental Frequency Estimation

Often, the half-lag or the multiple of the lag is found instead of the true lag. Assume, we have the values 50, 50, 100, 50, 50 estimated from the sequence of five neighboring frames. Obviously, the third estimate is incorrect: we have found the double of the true lag. Such defects can be corrected in several ways.

Nonlinear Filtering using Median Filter

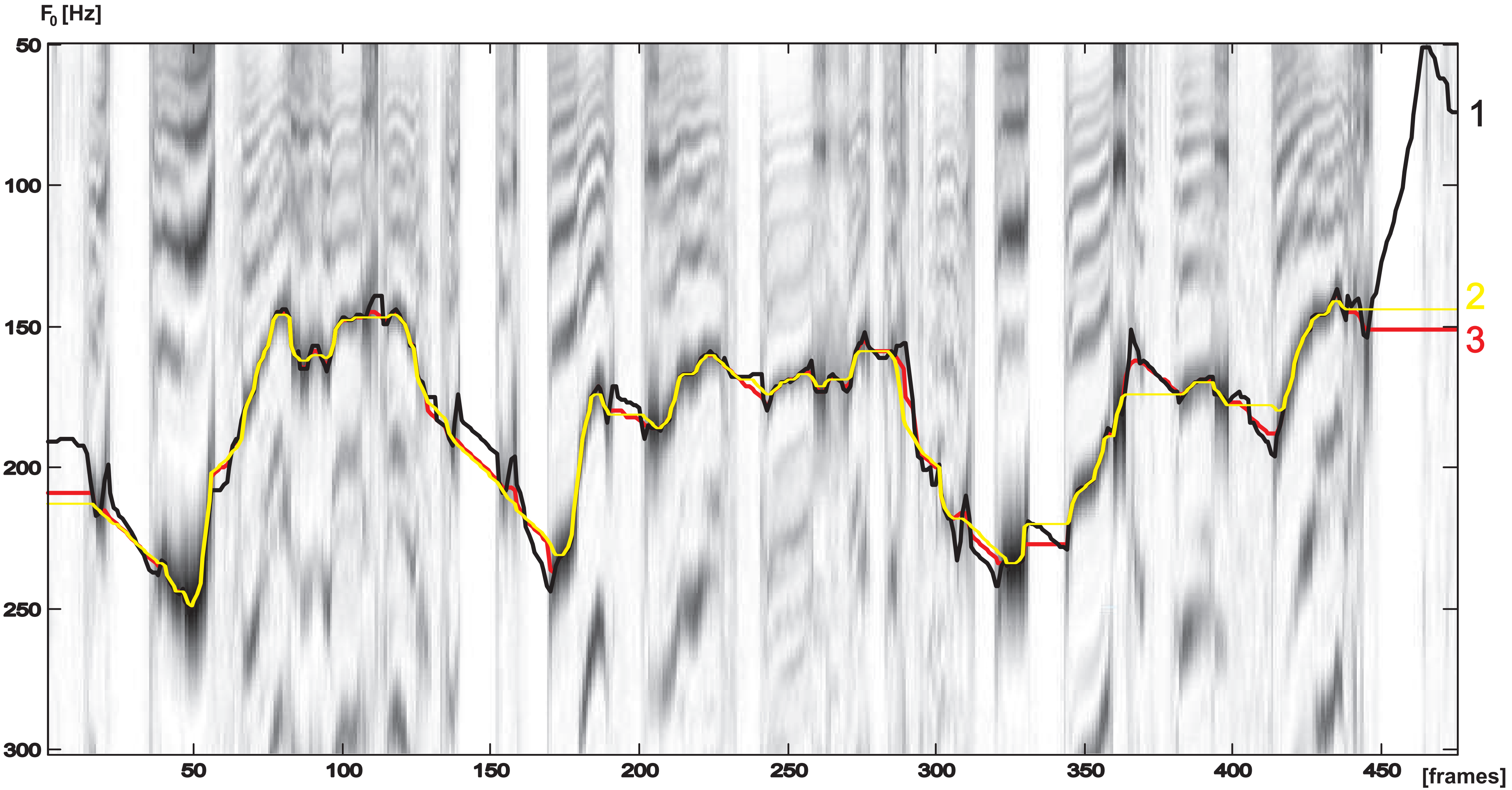
$$L(i) = \text{med}\left[L(i-k), L(i-k+1), \ldots, L(i), \ldots, L(i+k)\right] \tag{25}$$

Sort the items by their value and pick up the middle item. The lag values from the above example are therefore corrected to the sequence: 50, 50, 50, 50, 50.

Optimal Path Approach

In the previously introduced methods the lag was predicted by finding *one* maximum, eventually minimum per frame. The extreme estimation can be extended into searching in several neighboring frames: we are not interested in the value itself but in the "path" which minimizes (maximizes) a given criterium. The criterium can be defined as a function of $\frac{R(m)}{R(0)}$ or the prediction error energy for the given lag. Further, hypotheses on the path course have to be defined (floor in changes of the value between two neighboring frames. . .). The algorithm is defined as follows:

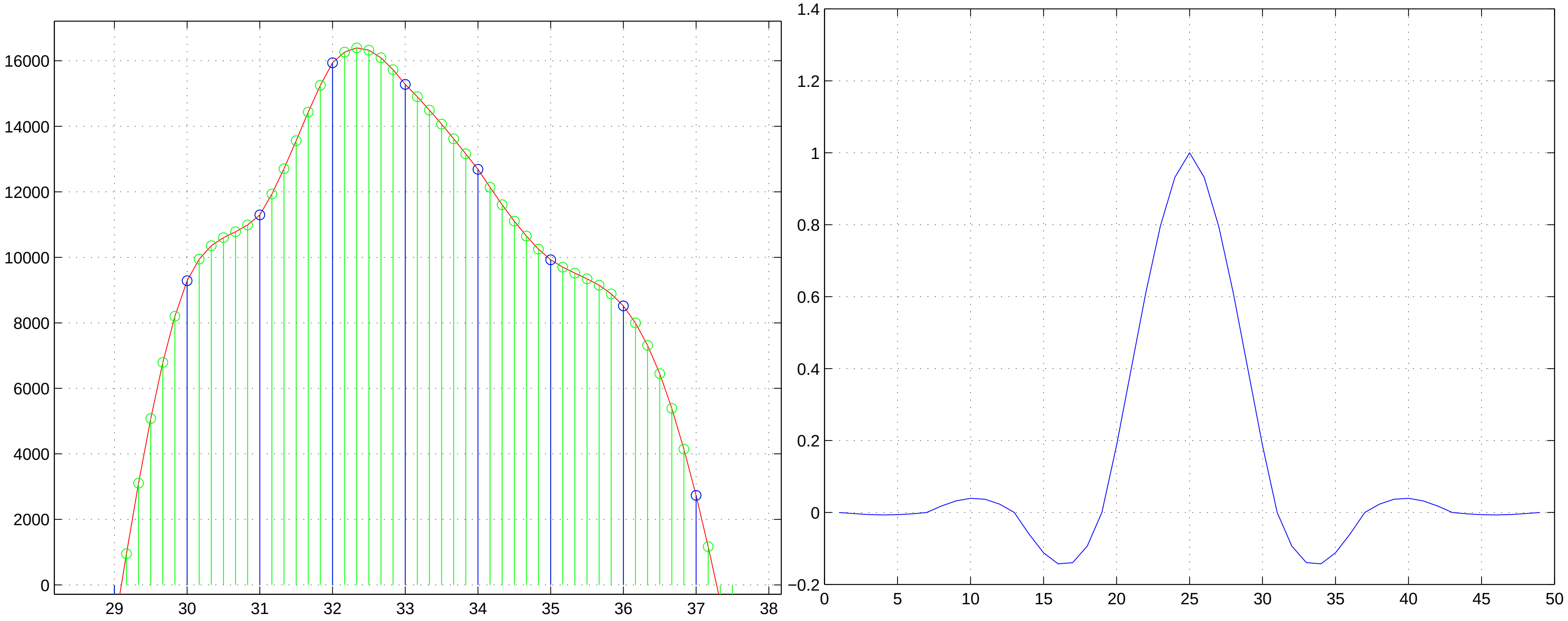
1. finding all possible paths — for instance, the lag value difference between two neighboring frames can't be larger than the set constant ΔL.
2. estimation of the overall criterium for the given path.
3. choose the optimal path.


F_0 [Hz]
50
100
150
200
250
300
50
100
150
200
250
300
350
400
450
[frames]
1
2
3

Decimal Sampling

To improve the F_0 detection accuracy we can apply super-sampling onto the signal and consequently filter it. This operation doesn't have to be implemented "physically" but rather can be projected into the autocorrelation coefficient estimation. Super-sampling often prevents detection of the double lag value.

An example of an interpolated signal and an interpolated filter:


16000
14000
12000
10000
8000
6000
4000
2000
0
29
30
31
32
33
34
35
36
37
38
1.4
1.2
1
0.8
0.6
0.4
0.2
0
−0.2
0
5
10
15
20
25
30
35
40
45
50