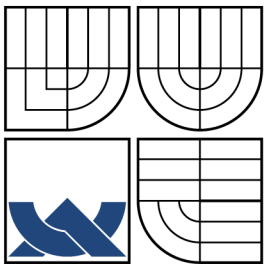
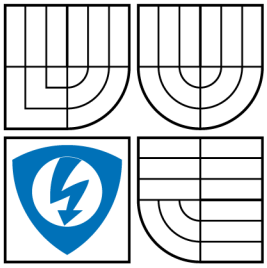


BRNO UNIVERSITY OF TECHNOLOGY
VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ



FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

MODERN METHODS OF TIME-FREQUENCY WARPING OF SOUND SIGNALS

MODERNÍ METODY BORCENÍ ČASOVÉ A KMITOČTOVÉ OSY ZVUKOVÝCH SIGNÁLŮ

DOCTORAL THESIS
DIZERTAČNÍ PRÁCE

AUTHOR
AUTOR PRÁCE
Ing. MICHAL TRZOS

SUPERVISOR
VEDOUCÍ PRÁCE
Ing. JIRI SCHIMMEL, PhD.

BRNO 2015

ABSTRACT

This thesis deals with representation of non-stationary harmonic signals with time-varying components. Its main focus is aimed at Harmonic Transform and its variant with subquadratic computational complexity, the Fast Harmonic Transform. Two algorithms using the Fast Harmonic Transform are presented. The first uses the gathered logspectrum as fundamental frequency change estimation method, the second uses analysis-by-synthesis approach. Both algorithms are used on a speech segment to compare its output. Further the analysis-by-synthesis algorithm is applied on several real sound signals to measure the increase in the ability to represent real frequency-modulated signals using the Harmonic Transform.

KEYWORDS

Harmonic Transform, Fan-Chirp Transform, FChT, QFFT, PTDFT

ABSTRAKT

Tato práce se zabývá reprezentací nestacionárních harmonických signálů s časově proměnnými komponentami. Primárně je zaměřena na Harmonickou transformaci a jeji variantu se subkvadratickou výpočetní složitostí, Rychlou harmonickou transformaci. V této práci jsou prezentovány dva algoritmy využívající Rychlou harmonickou transformaci. Prvni používá jako metodu odhadu změny základního kmitočtu sbírané logaritmické spektrum a druhá používá metodu analýzy syntézou. Oba algoritmy jsou použity k analýze řečového segmentu pro porovnání vystupů. Nakonec je algoritmus využívající metody analýzy syntézou použit na reálné zvukové signály, aby bylo možné změřit zlepšení reprezentace kmitočtově modulovaných signálů za použití Harmonické transformace.

KLÍČOVÁ SLOVA

Harmonická transformace, Fan-Chirp transformace, FChT, QFFT, PTDFT

TRZOS, Michal *Modern Methods of Time-Frequency Warping of Sound Signals*: doctoral thesis. Brno: Brno University of Technology, Faculty of Electrical Engineering and Communication, Department of Telecommunications, 2015. 107 p. Supervised by Ing. Jiri Schimmel, PhD.

DECLARATION

I declare that I have written my doctoral thesis on the theme of "Modern Methods of Time-Frequency Warping of Sound Signals" independently, under the guidance of the doctoral thesis supervisor and using the technical literature and other sources of information which are all quoted in the thesis and detailed in the list of literature at the end of the thesis.

As the author of the doctoral thesis I furthermore declare that, as regards the creation of this doctoral thesis, I have not infringed any copyright. In particular, I have not unlawfully encroached on anyone's personal and/or ownership rights and I am fully aware of the consequences in the case of breaking Regulation § 11 and the following of the Copyright Act No 121/2000 Sb., and of the rights related to intellectual property right and changes in some Acts (Intellectual Property Act) and formulated in later regulations, inclusive of the possible consequences resulting from the provisions of Criminal Act No 40/2009 Sb., Section 2, Head VI, Part 4.

Brno ....

author's signature

ACKNOWLEDGEMENT

Rád bych poděkoval vedoucímu disertační práce panu Ing. Jiřímu Schimmelovi, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Brno ....

author's signature

Faculty of Electrical Engineering
and Communication
Brno University of Technology
Purkynova 118, CZ-61200 Brno
Czech Republic

http://www.six.feec.vutbr.cz

ACKNOWLEDGEMENT

Research described in this doctoral thesis has been implemented in the laboratories supported byt the SIX project; reg. no. CZ.1.05/2.1.00/03.0072, operational program Výzkum a vývoj pro inovace.

Brno . . . . . .

. . . . . .

author's signature

EVROPSKÁ UNIE
EVROPSKÝ FOND PRO REGIONÁLNÍ ROZVOJ
INVESTICE DO VAŠÍ BUDOUCNOSTI

CONTENTS

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION

From all mechanisms of communication, sound communication is by far the most widely used by humans and at the same time easily processed using modern technology, namely digital signal processing. Most mammals, including humans, communicate using air stream modulation ranging in frequency from infrasound (whales) to ultrasound (bats). If the air stream modulation is constant, the produced sound can be approximated using an impulse train. In the frequency domain, the impulse train consists of a fundamenal frequency and harmonics at integer multiplies of the fundamental frequency. Such signal can therefore be effectively analyzed using traditional tools like the Fourier Transform. However if the air stream modulation changes in time, as is the case of most real signals, its frequency components also change in time. While frequency variance of the fundamental frequency may not be significant, it multiplies for each additional harmonic contained in the signal. When using Fourier Transform to analyze such singal, the higher harmonics may span over several frequency bins of the analyzed time interval depreciating the accuracy of harmonic parameters that can be acquired from the signal.

There are many applications that rely on the analysis of harmonic signals with time-varying components. Most of them deal with speech signals for speech coding, gender and age classification, detection of alcohol intoxication, emotion detection, or jitter estimation in Parkinsonian speech. Some musical instruments can be played in a way that causes fundamental frequency modulation like viola, violin, trombone, or guitar while some instruments create frequency modulation by their nature like the Theremin or the Leslie speaker. Also most synthesizers can be modulated using the pitch wheel which enables continous variation of the fundamental frequency. Analysing such signals may be performed with higher precision with a method that enables to take time-variant fundamental frequency into account.

This thesis therefore focuses on the representation of non-stationary signals with time-varying components. First it provides a summary of the state-of-the-art methods with main focus on Fan-Chirp Transform and Harmonic Transform. Then the focus turns solely on the Harmonic Transform and its computational demands which prevent its efficient use. A prerequisite to computing Harmonic Transform is knowledge of fundamental frequency change and an approach to decrease its estimation is presented. However the goal is decrease in computational complexity, which is presented as the Fast Harmonic Transform. This introduces some artifacts to the signal which is covered in the text. Then two algorithms for fundamental frequency estimation are presented. One is based on the gathered log-spectrum and the other on analysis-by-synthesis approach. Both algorithms are applied to a speech signal to compare their output. The thesis finishes with experiments on real signals.

1 STATE OF THE ART

1.1 Introduction

Many harmonic signals, including speech and music, exhibit frequency modulation caused by varying fundamental frequency. A traditional instrument for the analysis of speech and musical signals is Fourier Transform (FT) defined as

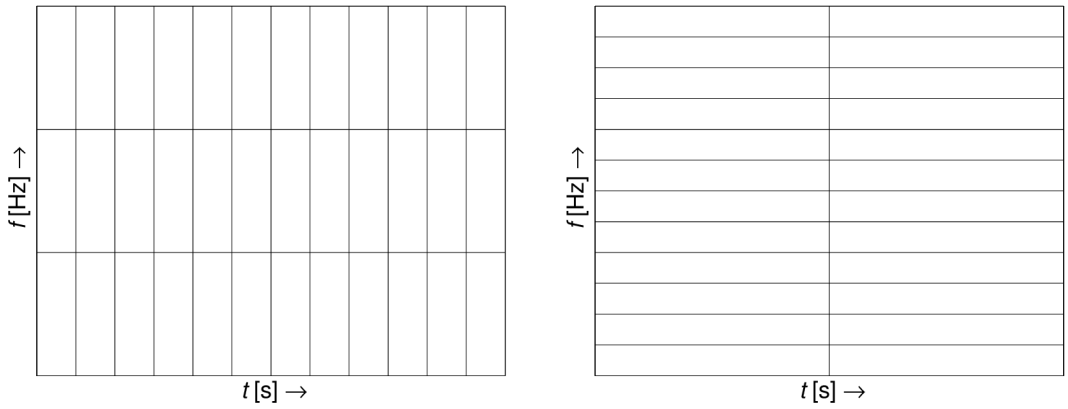
$$S(\omega) = \int_{-\infty}^{+\infty} s(t) \mathrm{e}^{-\mathrm{j}\omega t} \mathrm{d}t. \tag{1.1}$$

The original signal is recovered from the FT by inverse Fourier Transform (IFT)

$$s(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} S(\omega) \mathrm{e}^{\mathrm{j}\omega t} \mathrm{d}\omega. \tag{1.2}$$

The FT is generally able to represent frequency content of a signal, when the signal is composed of components with invariant frequency. Such signals can be called stationary harmonic signals and by using FT we can get their frequency representation with sufficient resolution in a specified frequency band. For shorter analysis windows the FT gives better time resolution while sacrificing frequency resolution, whereas for longer analysis windows the FT gives better frequency resolution at the cost of lower time resolution as seen in Fig. 1.1. The ability of the FT to represent frequency content of a signal diminishes if the signals contains components with varying frequency [1, 2]. This is represented in Fig. 1.2 showing a sinusoid with constant frequency over time and its FT which forms a clearly defined peak, whereas for the linearly modulated sinusoid, energy of the signal is spread between several frequency bins, which causes difficulties in harmonic parameter estimation. Especially if more harmonics are present in the signal. One solution of this problem is to use Warped Fourier Transform (WFT) [3], where the signal is frequency or time warped [4] before applying the FT, giving birth to warped wavelets [5, 6]. This operation can be interpreted as change of the signal's scale for the conversion of time-varying frequency components to frequency invariant components. The scaling operation can be generalized using the Scale Transform [7–9], where the scale is taken as a physical property of the signal, or the scaling operation can be integrated into the definition of transformation, as in Harmonic Transform [10]. Speech signals and other harmonic signals with a formant structure require a method to preserve the formant structure if modified. This can be done efficiently using frequency warping [11, 12].

There are other means of representation of signals with variable frequency components which are based on several models of speech. A family of transforms is based on the similarity of voiced speech to a chirp-periodic signal. Fan-Chirp Transform [13, 14] is suitable for signals with frequency components varying linearly on fan



(a) short analysis window (b) long analysis window

Fig. 1.1: Resolution of time-frequency analysis depending on the length of analysis window.

geometry, a property providing it with the best representation of chirp-like signals. The Fractional Fourier Transform (FrFT) [15–19] uses rotation of the time-frequency distribution to fit a signal with linearly changing frequency components, although similar to the Chirp Transform (CT) [20, 21] and Chirplet transform (ChT) [22], they cannot provide sufficient resolution for chirp-periodic signals both in lower and higher frequency bands at the same time as the Fan-Chirp Transform can.

A different approach is to consider the speech signal as a sum of periodic and aperiodic signals. This model takes into account that even the voiced part of speech contains some noise caused by air turbulence, thus making it quasi-periodic. This approach is used in Pitch Tracking Modified DFT (PTDFT) [23] and Time-Varying DFT (TVDFT) [24]. The PTDFT uses a pitch detection algorithm and analysis-by-synthesis approach in a closed loop to determine the fundamental frequency. When the fundamental frequency is known at least in two segments, the signal's harmonic components are estimated directly in the harmonic domain. The TVDFT also requires a pitch detection algorithm but the transformation kernel enables to perform tracking of the fundamental frequency and its partials.

1.2 Quasi-Harmonic Model

The Quasi-Harmonic Model (QHM) [25] is representation of a signal $s(t)$ consisting of K complex sinusoids defined as

$$s(t) = \sum_{k=1}^{K} c_k \mathrm{e}^{2\pi f_k t}, \tag{1.3}$$

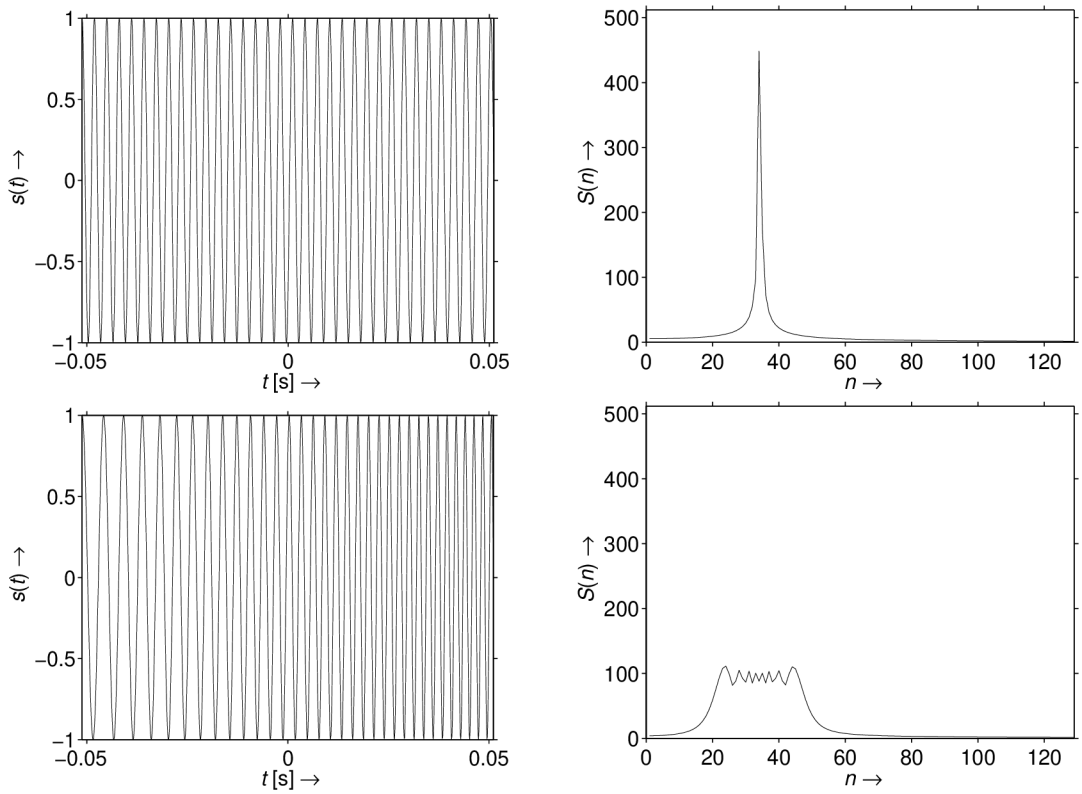


Fig. 1.2: Sinusoid with constant frequency over time and its FT (top); Linear chirp over time and its FT, showing that the signal's energy is spread over several frequency bins (bottom).

where f_k and c_k are the frequency and complex amplitude of the k-th sinusoid. To be able to compute complex amplitudes using least squares (LS) approach, estimates of frequencies $\{\hat{f}_k\}_{k=1}^{K}$ are needed. The frequencies f_k are defined from set of frequency estimates as

$$f_k = \hat{f}_k + \eta_k, \tag{1.4}$$

where η_k is the frequency error. If it is high, then the estimation of complex amplitudes c_k through LS will be biased. This has been addressed in [26] by the representation of the signal as

$$\hat{s}(t) = \sum_{k=1}^{K} (a_k + tb_k)\mathrm{e}^{\mathrm{j}2\pi\hat{f}_k t}, \tag{1.5}$$

where a_k and b_k denote complex amplitude and complex slope of the k-th component respectively. Parameters $\{a_k, b_k\}_{k=1}^{K}$ are then computed through iterative minimization of the LS criterion $\sum_{t=-T}^{T}((s(t) - \hat{s}(t))w(t))^2$, where $w(t)$ is the analysis window defined on interval $[-T, T]$. The QHM is further improved by adaptive Quasi-Harmonic Model [27] and enhanced adaptive Quasi-Harmonic Model [28], which have been shown to provide better results for AM-FM modulated speech signals [27, 28].

1.3 Sinusoidal Model

Sinusoidal model is based on the Fourier's theorem, which states that any periodic function can be modeled as a sum of sinusoids at various amplitudes and harmonically related frequencies [29]. In its most general expression it is a sum of complex exponentials (or partials)

$$s(t) = \sum_{k=1}^{K} a_k(t)\mathrm{e}^{\mathrm{j}\phi_k(t)}, \tag{1.6}$$

where $a_k(t)$ and $\phi_k(t)$ are the instantaneous amplitude and phase of the k-th sinusoid, respectively [30, 31] and where $\omega_k(t)$ is the frequency of the k-th sinusoid defined as the first derivative of phase $\phi_k(t)$.

Sinusoidal parameters of $s(t)$ from the observed signal $\hat{s}(t) = s(t) + v(t)$ in the non-stationary case, where $v(t)$ is an additive noise are modeled as [32]

$$\begin{aligned} s(t) &= a(t)\mathrm{e}^{\mathrm{j}\phi(t)}, \\ a(t) &= \exp\left(\textstyle\sum_{k=0}^{K} \Re\{\alpha_k\}t^k\right), \\ \phi(t) &= \textstyle\sum_{k=0}^{K} \Im\{\alpha_k\}t^k, \end{aligned} \tag{1.7}$$

where α_k are the $K+1$ complex non-stationary sinusoidal parameters. The amplitude is represented by $a(t)$ and frequency is represented by $\phi'(t)/2\pi$. The log-amplitude modulation parameters are given by the real time part of the parameters α_k,$\Re\{\alpha_k\}$, and the phase modulation parameters are given by the imaginary part, $\Im\{\alpha_k\}$.

There are several methods for estimating the sinusoidal parameters in (1.7) which have been generalized to the non-stationary case [32–37] and some of the significant methods are now presented.

1.3.1 Quadratically Interpolated FFT

The quadratically interpolated FFT (QIFFT) is a maximum likelihood method that has been used for sinusoidal parameter estimation in audio applications by means of quadratic peak interpolation in a zero padded FFT [31]. An improved QIFFT method to estimate first order amplitude and frequency rates of time-varying sinusoidal components has been presented in [38]. A sinusoid with first-order AM and FM can be written as

$$s(t) = \mathrm{e}^{\alpha_0 t+\lambda_0}\mathrm{e}^{\mathrm{j}(\beta_0 t^2+\omega_0 t+\phi_0)}, \tag{1.8}$$

where ω_0 is instantaneous frequency at $t=0$, λ_0 is instantaneous log-amplitude at $t=0$, ϕ_0 is instantaneous phase at $t=0$, α_0 is amplitude change rate, and β_0 is frequency change rate. The equation (1.8) is equivalent to (1.7) for $K=2$.

The QIFFT method for estimating sinusoidal parameters from peaks in spectral magnitude data can be summarized as follows [38]:

1. Calculate amplitude and phase spectrum of audio data, by using a zero-padded windowed FFT.
2. Find the maximum peak magnitude.
3. Quadratically interpolate log-amplitude of the peak using two neighboring samples.
4. Estimate the frequency and amplitude from the interpolation.
5. Estimate the phase, if needed, by quadratically interpolating the phase spectrum based on the interpolated frequency estimate.
6. Remove the peak from FFT data for subsequent processing.
7. Repeat steps $2-6$ above for each peak.

The QIFFT can be seen as approximating the nearly parabolic shape of the spectral peak of a non-Gaussian window with the truly parabolic shape of a Gaussian window. In practice, truncated Gaussian window is used on the observer signal $\hat{s}(t)$, so the log-magnitude and phase are not exactly quadratic. This is called the *direct method*. For other non-Gaussian windows, including Hann, Hamming, and Blackman-Harris,

an adapted method has been designed in [38]. An extension of this method for $t \neq 0$ can be found in [39].

1.3.2 Distributed Derivative Method

If we define the inner product of two signals $x(t)$ and $y(t)$ as

$$\langle x, y \rangle = \int_{-\infty}^{\infty} x(\tau) y^*(\tau) \mathrm{d}t, \tag{1.9}$$

then the distributed derivative method (DDM) [40] generates parameter estimators for α_k using the following system of equations

$$-\langle s, \gamma_i' \rangle = \sum_{k=1}^{K} \alpha_k \left\langle k\mathcal{T}^{k-1} s, \gamma_i \right\rangle, \quad i = 1, ..., L, \tag{1.10}$$

where $\mathcal{T}$ is an operator defined by $(\mathcal{T}x)(t) = tx(t))$ [32]. To solve for the K parameters, $L \geq K$ equations with L different atoms $\gamma_i(\tau)$ are needed to solve the linear system of L equations. Generalization of this method for non-stationary signals can be found in [41].

1.3.3 Generalized Derivative Method

In generalized derivative method (GDM) [42], we generate a linear system of L equations by applying L successive derivatives to $s(t)$ and taking the inner products with only one atom $\gamma(t)$ [32]. This results in the following system of equations

$$\left\langle s^{(i)}, \gamma \right\rangle = \sum_{k=1}^{K} \alpha_k \left\langle (k\mathcal{T}^{k-1} s)^{(i-1)}, \gamma \right\rangle, \quad i = 1, ..., L, \tag{1.11}$$

where superscript (i) denotes differentiation i-times. This requires signal derivatives up to order L, which, in practice, will be estimated with first-order differentiator filter. With L signal derivatives we have a linear system of L equations from which we can solve for K model parameters [32].

1.3.4 General Reassignment Method

Taking L derivatives of the signal in GDM can be avoided by the use of integration by parts to move the differentiation from the signal to the atom $\gamma(t)$ [32]. If we assume the chosen atom is sufficiently continuous and all its derivatives up to order $L-1$ go to zero at $t \pm \infty$ such that the identity $\langle x', y \rangle = -\langle x, y' \rangle$ holds for each successive derivative up to L. Then we get the following system of equations

$$-\left\langle s, \gamma^{(i)} \right\rangle = \sum_{k=1}^{K} \alpha_k \left\langle k\mathcal{T}^{k-1} s, \gamma^{i-1} \right\rangle, \tag{1.12}$$

which can be shown equivalent to (1.11), but here we use derivatives of the atom rather than derivatives of the signal [32]. This is called the generalized reassignment method (GRM) [42] and it is based on the earlier reassignment method found in [41].

1.3.5 Phase Vocoder for Non-Stationary Sinusoidal Modeling

This analysis method is based on the generalization of the phase vocoder approach using signal spectra derivatives [43] to non-stationary sinusoidal modeling presented in [44]. It is also the simplest possible method of non-stationary sinusoidal modeling [29]. If we consider consecutive signal segments x (e.g. by a sliding FFT) of length N from the signal s, their discrete spectra X can be obtained by zero-phase DFT. Then $X(\omega) = S_w(t, \omega)$ is the spectrum of the frame centered at the desired estimation time and $X_\pm(\omega) = S_w(t \pm (1/f_\mathrm{s}), \omega)$ be its left (one sample before) and right (one sample after) neighboring spectra. Then the derivative can be approximated by first-order difference. According to the model (1.7), the log-amplitude and phase differences correspond to the real and imaginary parts of the logarithm of spectral ratios

$$\begin{aligned} \Delta_\lambda(X_1, X_2) &= \log|X_1| - \log|X_2| \\ &= \Re(\log(X_1/X_2)), \\ \Delta_\phi(X_1, X_2) &= \angle X_1 - \angle X_2 \\ &= \Im(\log(X_1/X_2)), \end{aligned} \tag{1.13}$$

where X_1 and X_2 denote complex spectra. The amplitude modulation $\hat{\mu}_0$ can be estimated from the mean of the left and right estimates by

$$\begin{aligned} \mu_- &= \Delta_\lambda(X, X_-) f_\mathrm{s}, \\ \mu_+ &= \Delta_\lambda(X_+, X) f_\mathrm{s}, \\ \hat{\mu}_0 &= (\mu_- + \mu_+)/2. \end{aligned} \tag{1.14}$$

Similarly, the instantaneous frequency $\hat{\omega}_0$ can be estimated from the left and right phase spectra

$$\begin{aligned} \omega_- &= \mathrm{unwrap}(\Delta_\phi(X, X_-) f_\mathrm{s}), \\ \omega_+ &= \mathrm{unwrap}(\Delta_\phi(X_+, X) f_\mathrm{s}), \\ \hat{\omega}_0 &= (\omega_- + \omega_+)/2, \end{aligned} \tag{1.15}$$

where $\mathrm{unwrap}(\beta)$ is a function consisting of adding 2π to β if it is lower than 0.

Using left and right estimates of the frequency, we can estimate the frequency modulation by first-order difference

$$\hat{\psi}_0 = (\omega_+ - \omega_-) f_\mathrm{s}. \tag{1.16}$$

1.4 Estimation of Instantaneous Harmonic Parameters Using Frequency-Modulated Bandpass Filters

There are several methods for estimation of instantaneous harmonic parameters. Some of them are connected with the notion of analytic signal based on the Hilbert transform (HT) [45]. A unique complex signal $z(t)$ can be generated from a real one $s(t)$ using the Fourier transform [46]. This can be done as a time-domain procedure

$$z(t) = s(t) + \mathrm{j}\mathcal{H}[s(t)] = a(t)\mathrm{e}^{\mathrm{j}\varphi(t)}, \tag{1.17}$$

where $\mathcal{H}$ is the Hilbert transform defined as

$$\mathcal{H}[s(t)] = p.v. \int_{-\infty}^{\infty} \frac{s(t-\tau)}{\pi\tau}\mathrm{d}\tau, \tag{1.18}$$

where $p.v.$ denotes Cauchy principle value of the integral, $z(t)$ is referred to as Gabor's complex signal, $a(t)$ and $\varphi(t)$ can be considered the instantaneous amplitude and phase, respectively. Signals $s(t)$ and $\mathcal{H}[s(t)]$ are theoretically in quadrature. Signal $z(t)$ can be expressed in polar coordinates, therefore $a(t)$ and $\varphi(t)$ can be calculated as

$$a(t) = \sqrt{s^2(t) + \mathcal{H}^2[s(t)]}, \tag{1.19}$$

$$\varphi(t) = \arctan\left(\frac{\mathcal{H}[s(t)]}{s(t)}\right). \tag{1.20}$$

Another way of estimating the instantaneous harmonic parameters is discrete energy separation algorithm (DESA), which is based on the Teager energy operator [47]. The energy operator is defined as

$$\psi[s(n)] = s^2(n) - s(n-1)s(n+1), \tag{1.21}$$

where the derivative operation is approximated by the symmetric difference [45]. The instantaneous amplitude $a(n)$ and frequency $f(n)$ can be evaluated as

$$a(n) = \frac{2\psi[s(n)]}{\sqrt{\psi[s(n+1) - s(n-1)]}}, \tag{1.22}$$

$$f(n) = \arcsin\sqrt{\frac{\psi[s(n+1) - s(n-1)]}{4\psi[s(n)]}}. \tag{1.23}$$

The Hilbert transform and DESA can be applied only to monocomponent signals. For multicomponent signals, the signal should be split into single components

before using these techniques [45]. It is possible to use narrow-band filtering for this purpose [48].

Now with the presented methods for estimating instantaneous frequency, amplitude and phase of monocomponent signals, we will use a harmonic+noise representation of multicomponent speech [49] and audio signals [50] as a combination of sinusoids with slowly varying amplitudes and frequencies

$$s(n) = \sum_{k=1}^{K} A_k(n) \cos \varphi_k(n) + r(n), \tag{1.24}$$

where A_k is the instantaneous amplitude of k-th harmonic, K is the number of harmonics present in the signal, $r(n)$ is the noise component, φ_k is the instantaneous phase of k-th harmonic defined as

$$\varphi_k(n) = \sum_{i=0}^{n} \frac{2\pi f_k(i)}{f_\mathrm{s}} + \varphi_k(0), \tag{1.25}$$

where f_k is the instantaneous frequency of the k-th harmonic, f_s is the sampling frequency and $\varphi_k(0)$ is the initial phase of the k-th harmonic. The harmonic model assumes that the frequencies of the components are integer multiplies of fundamental frequency $f_k = kf_0$, where f_0 is the fundamental frequency.

The instantaneous frequencies can deviate from the multiples of the fundamental frequency for the value less than some specified f_tr as

$$|f_k - kf_0| < f_\mathrm{tr}. \tag{1.26}$$

To separate a certain harmonic from the others, it is necessary to use a bandpass filter [47, 51]. The band-pass filters can be used for signal decomposition into non-stationary periodic components with instantaneous frequency, amplitude and phase. The method can be used for processing of frequency-modulated signals such as voiced speech. Stationary filter can provide accurate results for estimation of the fundamental frequency, but it is not suitable for high-order harmonics [52]. The impulse response of such filter for k-th signal component can be written as [52]

$$h_k(n) = \begin{cases} 2f_\Delta^k, & n = 0, \\ \frac{f_\mathrm{s}}{n\pi} \cos\left(\frac{2\pi n}{f_\mathrm{s}} f_c^k\right) \sin\left(\frac{2\pi n}{f_\mathrm{s}} f_\Delta^k\right), & n \neq 0, \end{cases} \tag{1.27}$$

where $f_c^k = (f_{k-1} + f_k)/2$, $F_\Delta^k = (f_k - f_{k-1})/2$ and $[f_{k-1}, f_k]$ is the band-pass filter's pass band. Parameters f_c^k and f_Δ^k correspond to the center frequency of the pass band and the half of the filter's bandwidth, respectively. The convolution of signal $s(n)$ and the impulse response $h_k(n)$ produces a band-limited output signal $s_k(n) = s(n) * h_k(n)$ which can be rewritten as [52]

$$s_k(n) = A(n) \cos(\frac{2\pi}{f_\mathrm{s}} n f_c) + B(n) \sin(\frac{2\pi}{f_\mathrm{s}} n f_c), \tag{1.28}$$

where

$$\begin{aligned} A(n) &= \sum_{i=0}^{N-1} \frac{2s(i)}{\pi(n-i)} \sin\left(\frac{2\pi(n-i)}{f_s} f_\Delta\right) \cos\left(\frac{2\pi(n-i)}{f_s} f_c\right), \\ B(n) &= \sum_{i=0}^{N-1} \frac{-2s(i)}{\pi(n-i)} \sin\left(\frac{2\pi(n-i)}{f_s} f_\Delta\right) \sin\left(\frac{2\pi(n-i)}{f_s} f_c\right). \end{aligned} \tag{1.29}$$

From (1.29), the instantaneous magnitude, phase and frequency can then be calculated as [52]

$$a(n) = \sqrt{A^2(n) + B^2(n)}, \tag{1.30}$$

$$\varphi(n) = \arctan\left(\frac{-B(n)}{A(n)}\right), \tag{1.31}$$

$$f(n) = \frac{\varphi(n+1) - \varphi(n)}{2\pi} f_s. \tag{1.32}$$

Using instantaneous pitch contour obtained by stationary band-pass filters it is possible to ensure appropriate processing of high-order harmonics [45] by using frequency-modulated band-pass filters. The band-pass filters have a closed form impulse response that can be adjusted according to instantaneous frequencies of the harmonics and the fundamental frequency modulations of speech. The frequency modulated filter has a warped pass band, aligned to the given frequency contour $f_c^k(n)$, that provides adequate analysis of periodic components with rapid frequency alterations can be defined as [52]

$$\begin{aligned} A(n) &= \sum_{i=0}^{N-1} \frac{2s(i)}{\pi(n-i)} \sin\left(\frac{2\pi(n-i)}{f_s} f_\Delta\right) \cos\left(\frac{2\pi(n-i)}{f_s} \varphi_c(n,i)\right), \\ B(n) &= \sum_{i=0}^{N-1} \frac{-2s(i)}{\pi(n-i)} \sin\left(\frac{2\pi(n-i)}{f_s} f_\Delta\right) \sin\left(\frac{2\pi(n-i)}{f_s} \varphi_c(n,i)\right) \end{aligned} \tag{1.33}$$

$$\varphi_c(n,i) = \begin{cases} \sum_{j=n}^{i} f_c^k(j), & n < i, \\ -\sum_{j=i}^{n} f_c^k(j), & n > i, \\ 0, & n = i. \end{cases} \tag{1.34}$$

This approach is an alternative to time warping that is used in Harmonic and Fan-Chirp transforms (see below). It has been used for the improvement of the RAPT [53] pitch estimation algorithm [54]. In [55] it has been used for pitch, timbre and time-scale modifications, which has been improved and generalized in [56]. Further applications involve real-time speech conversion [57], estimation of spectral envelopes by means of linear prediction [58], parametric coding of audio and speech [59], and sinusoidal, transient and noise modeling [60].

1.5 Harmonic Transform

Time-varying harmonic signal generally contains higher harmonics whose nominal instantaneous frequencies are expressed by

$$c_k(t) = (k+1)c_0(t), \quad k = 1, 2, 3, \ldots \tag{1.35}$$

where $c_0(t)$ is the frequency of the fundamental and $c_k(t)$ is the frequency of the k-th harmonic component.

Harmonic transform has been introduced in [10] and it is based on [61] [62]. Its main difference from Fourier transform is the integrated time-warping function. It is defined as

$$S_{\phi_{\mathrm{u}}(t)}(\omega) = \int_{-\infty}^{+\infty} s(t)\phi'_{\mathrm{u}}(t)\mathrm{e}^{-\mathrm{j}\omega\phi_{\mathrm{u}}(t)}\mathrm{d}t, \tag{1.36}$$

where $\phi_{\mathrm{u}}(t)$ is a unit phase function, which is the phase of the fundamental harmonic component divided by its nominal instantaneous frequency [10], and $\phi'_{\mathrm{u}}(t)$ is first derivation of $\phi_{\mathrm{u}}(t)$. The $\phi_{\mathrm{u}}(t)$ is required to be invertible and differentiable on $(-\infty, +\infty)$. When the $\phi_{\mathrm{u}}(t) = t$, the HT reverts to the FT. The inverse harmonic transform (IHT) is defined as [10]

$$s(t) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} S_{\phi_{\mathrm{u}}(t)}(\omega)\mathrm{e}^{\mathrm{j}\omega\phi_{\mathrm{u}}(t)}\mathrm{d}\omega. \tag{1.37}$$

The relationship between $\phi_{\mathrm{u}}(t)$ and the nominal instantaneous frequency $c_k(t)$ as represented by a harmonic signal $f_h(t)$ with instantaneous frequencies given in (1.35) is

$$f_h(t) = \sum_{k=0}^{+\infty} a_k \mathrm{e}^{\mathrm{j}(k+1)\alpha(t)}, \tag{1.38}$$

where a_k is the amplitude of the k-th harmonic and $\alpha(t)$ is the phase function of the fundamental. The relationship between $\alpha(t)$ and $c_0(t)$ is

$$c_0(t) = \alpha'(t), \tag{1.39}$$

assuming the nominal instantaneous frequency of the fundamental is normalized to be one. When $\phi_{\mathrm{u}}(t) = \alpha(t)$, the HT of $f_h(t)$ is

$$\begin{aligned} S_{\alpha(t)}(\omega) &= \int_{-\infty}^{+\infty}\sum_{k=0}^{+\infty} a_k \mathrm{e}^{\mathrm{j}(k+1)\alpha(t)}\alpha'(t)\mathrm{e}^{-\mathrm{j}\omega\alpha(t)}\mathrm{d}t, \\ &= \sum_{k=0}^{+\infty} a_k \int_{-\infty}^{+\infty} \mathrm{e}^{\mathrm{j}(k+1-\omega)\alpha(t)}\mathrm{d}\alpha(t), \\ &= \sum_{k=0}^{+\infty} 2\pi a_k \delta(\omega - k - 1), \end{aligned} \tag{1.40}$$

which is an impulse-train for arbitrary $c_0(t)$. Therefore with a certain unit phase function, the HT can provide an impulse-train spectrum for a time-varying harmonic signal.

1.5.1 Short-Time Harmonic Transform

Time-frequency transforms describe signal in the time-frequency plane. STFT is one of the widely used time-frequency transforms. Many speech and music analysis applications based on sinusoidal modeling use the STFT spectrum for estimation of the harmonic parameters at one instant, providing instantaneous parameters and assuming the signal is stationary over the length of the segment. A window function $w(t)$ is usually used to emphasize the signal around the instant and to suppress artifacts caused by spectral leakage

$$\mathrm{STFT}(\omega, t) = \int_{-\infty}^{+\infty} s(\tau) w(\tau - t) \mathrm{e}^{-\mathrm{j}\omega\tau} \mathrm{d}\tau. \tag{1.41}$$

The STFT of a time-varying harmonic signal has poor resolution in theme and/or frequency domain although there are ways to improve the resolution [10].

HT can be used to improve resolution of the STFT for time-varying harmonic signals. After replacing the FT in (1.41) with the HT we get the short-time Harmonic transform (STHT) as

$$\mathrm{STHT}_{\phi_u(t)}(\omega, t) = \int_{-\infty}^{+\infty} s(\tau) w(\tau - t) \phi_u'(\tau) \mathrm{e}^{-\mathrm{j}\omega\phi_u(\tau)} \mathrm{d}\tau, \tag{1.42}$$

where $s(t)$ is the signal and $w(t)$ is the window function. Linear change of fundamental frequency in a given segment is presumed, which is sufficiently satisfied by selecting an analysis window of appropriate length [10].

Instantaneous phase $\varphi(t)$ of a sinusoid with linear change of frequency [63] is defined as

$$\varphi(t) = 2\pi \left(f_0 t + \frac{\epsilon t^2}{2} \right), \tag{1.43}$$

where f_0 is fundamental frequency and $\epsilon = \Delta f_0 / T$ is the change of fundamental frequency divided by length of the segment. Assuming discrete signal segment of the length N, where $T = N/f_\mathrm{s}$, the discrete phase $\varphi(n)$ of a sinusoid with linear frequency variation [49] can be written as

$$\varphi(n) = 2\pi \left(\frac{f_0 n}{f_\mathrm{s}} + \frac{\Delta f_0 n^2}{2 N f_\mathrm{s}} \right), \tag{1.44}$$

where f_0 is discrete instantaneous frequency, N is length of the analysis window, and f_s is sampling frequency (initial phase is disregarded for simplicity).

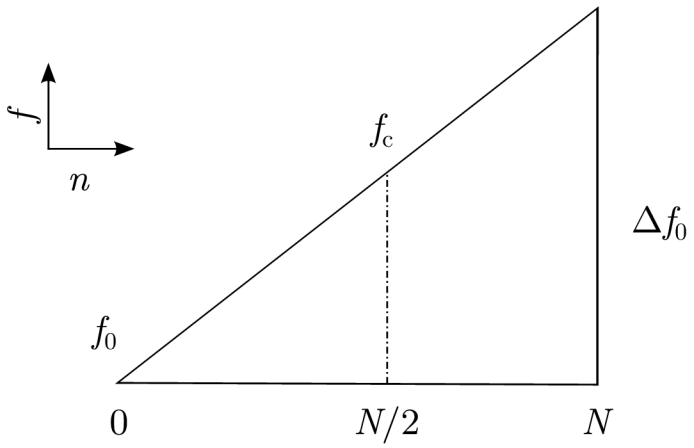


Fig. 1.3: Illustration of the relationship between fundamental frequency f_0, central fundamental frequency f_c, and fundamental frequency change Δf_0 in a segment of length N.

Initial fundamental frequency in a given segment can be written as

$$f_0 = f_\mathrm{c} - \frac{a f_\mathrm{c}}{2},\ a = \frac{\Delta f_0}{f_\mathrm{c}}, \tag{1.45}$$

where f_c is the central fundamental frequency within a segment and a is the slope of fundamental frequency change within the segment, see Fig. 1.3. Substituting (1.45) to (1.44) we get [49]

$$\varphi(n) = \frac{2\pi}{N}\alpha(n),\ \alpha_a(n) = n\left(1 - \frac{a}{2} + \frac{an}{2N}\right), \tag{1.46}$$

which is a non-linear relationship between the original and time-warped axis which depends on the fundamental frequency slope a as shown in Fig. 1.5.

Frequencies of spectral lines of the Fourier transform are given as

$$f_\mathrm{c} = \frac{f_\mathrm{s}}{N}, \tag{1.47}$$

and from the equation (1.46) it is obvious that the instantaneous phase is

$$\varphi(n) = \frac{2\pi}{N}\alpha(n). \tag{1.48}$$

Discrete harmonic transform (DHT) of signals with linear fundamental frequency variation [49] is defined as

$$S_a(k) = \frac{1}{N}\sum_{k=0}^{N-1} s(n)\alpha'(n)\mathrm{e}^{\mathrm{j}\frac{2\pi k}{N}\alpha(n)}, \tag{1.49}$$

where

$$\alpha'_a(n) = 1 - \frac{a}{2} + \frac{an}{N}, \tag{1.50}$$

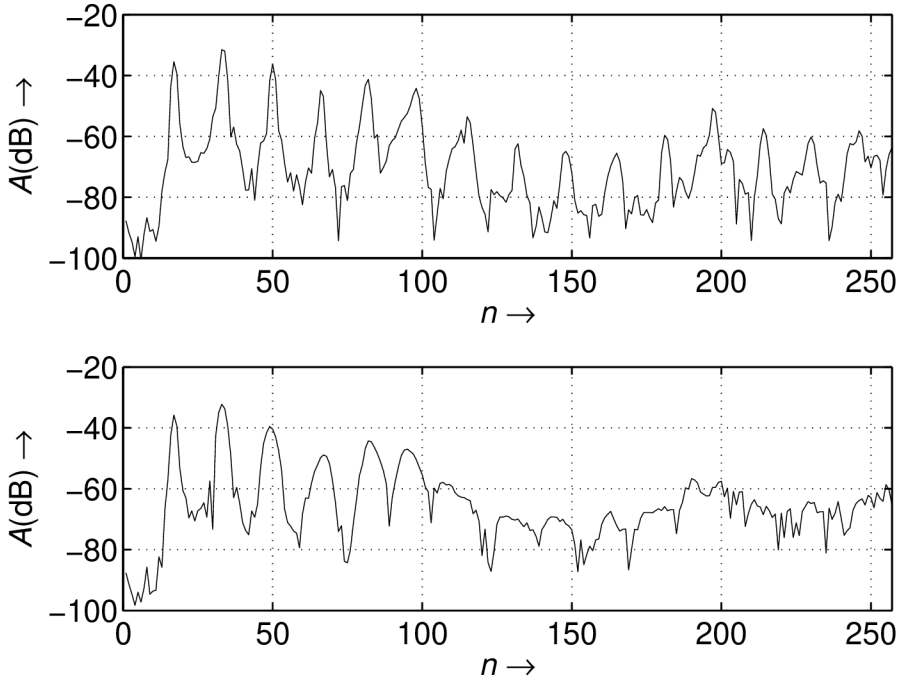


Fig. 1.4: Fourier transform spectrum (bottom); Harmonic transform spectrum with parameter $a = 0.2$ (top); of sound sample *happy child*.

is first-order derivation of (1.46).

The difference between DFT and DHT at analysis of non-stationary harmonic signals can be seen on a part of speech uttering from the PTDB-TUG[1] database with frequency modulation. In Fig. 1.4 (bottom) we can see that the higher frequencies are smoothed due to frequency modulation. In Fig. 1.4 (top) even high frequency peaks are clearly visible.

1.5.2 Estimation of Fundamental Frequency Change

Pitch estimation algorithm proposed in [64] consists of three stages. First, the fundamental frequency change rate within a frame is computed, then the best pitch candidate is selected by analysis of harmonic spectrum and finally the pitch estimation from several consecutive frames is analyzed in order to correct estimation errors.

The algorithm starts from finding fundamental frequency change by minimizing

[1]Pitch Tracking Database from Graz University of Technology available at: `http://www.spsc.tugraz.at/tools/ptdb-tug`

Spectral Flatness Measure (SFM)

$$\arg\min SFM(a) = \frac{\sqrt[N]{\prod_{k=0}^{N-1} |S_a(k)|}}{\frac{1}{N}\sum_{k=0}^{N-1} |S_a(k)|}, \tag{1.51}$$

where S_a is harmonic spectrum of given segment for given parameter a and $|.|$ denotes absolute value. Minimal spectral flatness value indicated highest concentration, which means optimal fit of signal and DHT kernel [64].

After determining f_0 change rate DHT is computed using the estimated pitch rate change a and a peak-picking algorithm is used to find local maxima of the harmonic spectrum (ideally on an interval suited to the spectral characteristic of the analyzed signal). Then the confidence function is computed

$$r(f) = \frac{\sum_{k=1}^{K} |S_a(kf)|^2}{\sum_{n=0}^{N-1} |S_a(n)|^2}, \tag{1.52}$$

where N is the length of the segment. The confidence function is designed to show how much energy of the frame is carried by particular pitch and its several first harmonics. Procedure of selecting best fundamental frequency candidate is as follows: take the highest local maximum of $r(f)$, if there is no corresponding local maxima in harmonic spectrum, discard it and repeat the procedure, otherwise the frequency corresponding to the current local maximum of $r(f)$ is initial fundamental frequency estimation. Further, the pitch frequency is refined using method similar to the one presented in [65]

$$f_r = \frac{\sum_{n-1}^{n_{k\max}} \frac{f_n}{n}}{n_{k\max}}, \tag{1.53}$$

where f_n is frequency of local maximum of harmonic spectrum corresponding to n-th harmonic of the selected candidate in previous step, f_r is refined pitch and $n_{k\max}$ is number of possible harmonics. Fundamental frequency is considered slowly time-varying and cannot change rapidly between consecutive segments. In the presence of noise, there are possible estimation errors. The estimates are held in a buffer and tracking algorithm gives the final estimate. In the proposed approach, median filtering is used as it has proved robust in the presence of noise [64].

1.5.3 Harmonic Parameters Estimation

The harmonic parameters are estimated on the basis of the harmonic+noise model (1.24), which here is defined for the periodic component as [64]

$$\hat{h}(n) = \sum_{k=1}^{K} A_k \cos(k\varphi(n) + \varphi_k(0)), \tag{1.54}$$

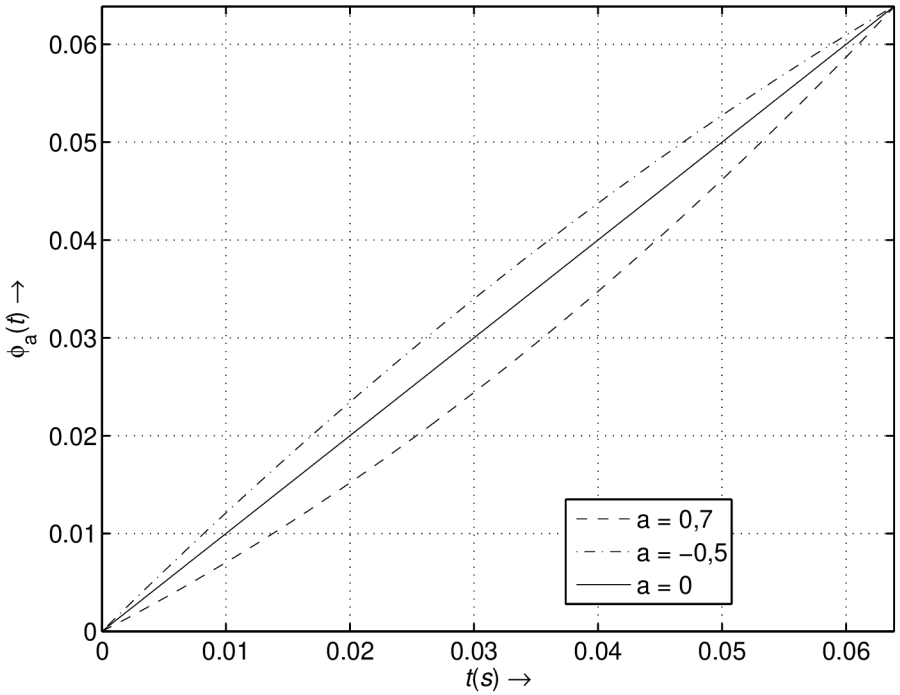


Fig. 1.5: Influence of parameter a on the phase function $\alpha_{\mathrm{a}}(t)$.

where A_k is the amplitude of the k-th harmonic, $k\varphi(n)$ is the instantaneous phase of the k-th harmonic component defined in (1.44) with f_c as the fundamental frequency and $\varphi(0)$ is the initial phase of the k-th harmonic component. The pitch harmonics are not aligned with the spectral lines and cannot be directly estimated from the HT spectrum. The DHT variant aligned with the fundamental frequency is defined as [64]

$$S(k) = \sum_{n=0}^{N-1} s(n)\alpha'(n)\mathrm{e}^{-j\frac{2\pi k f_r}{f_s}\alpha(n)}, \tag{1.55}$$

where f_r is the fundamental frequency and $k = 1, ..., K$ is the number of harmonics. The amplitudes and phases of the harmonics can be computed directly from $S(k)$ coefficients

$$\begin{aligned} A_k &= \sqrt{\Re S(k)^2 + \Im S(k)^2}, \\ \varphi_k(0) &= -\arctan\left(\frac{\Im S(k)^2}{\Re S(k)^2}\right), \end{aligned} \tag{1.56}$$

where $\Re$ and $\Im$ stands for the real and the imaginary parts of $S(k)$, respectively. The periodic component can then be generated using (1.54) and the noise component can be calculated from the input signal $s(n)$ as

$$\hat{r}(n) = s(n) - \hat{h}(n). \tag{1.57}$$

1.6 Fan-Chirp Transform

The Fan-Chirp Transform (FChT) has been introduced in [14]. A variation of FChT with different properties has been developed independently of the original work in [66]. In the FChT, the basis functions are a set of linear frequency modulated sinusoids with *chirp rate* tuned to the given signal [13, 14]. The term *fan* comes from the FChT's property of projecting the Wigner-Ville distribution according to a fan geometry. The set of basis functions in FChT is consistent with a harmonic signal whose fundamental frequency is changing linearly in time [67]. There have been attempts at making sinewave analysis consistent with time-varying sinewave models. This is particularly important in high-frequency speech regions where harmonic frequency modulation can be significant. A sinewave analysis/synthesis system based on the FChT has been presented in [67], where the short-time FChT is compared to a STFT estimation of sinusoidal parameters of time-varying synthetic speech-like signal. In [66] the FChT is used for melody extraction from polyphonic music and shows combination of FChT with Constant-Q Transform (CQT) [68]. Performance of the presented extraction system is compared to STFT using MIREX [2] and RWC [3] databases. The system has been improved further using spectral clustering [69] of local f_0 candidates to form pitch contours [70]. In [71] it has been enhanced with automatic detection of singing voice in polyphonic recordings, extraction of harmonic sounds from the audio and their classification. And [72] presents an application of the FChT based F0gram [73] to musicology. In [74] FChT is used for estimation of pitch and pitch rate analysis of Vietnamese speech and points out an undesired spectral envelope smoothing caused by the FChT. Another application is in [75], where FChT has been used for hybrid sinusoidal plus noise modeling of polyphonic audio. In presence of several musical instruments with different pitch variation simultaneously, the spectrum will show sharp peaks for instruments with the same chirp rate. The estimates of individual chirp rates of individual harmonic partials follow a multi-modal distribution that is approximated by a Gaussian mixture model. In [76] FChT has been used in an algorithm for monoaural speech separation.

A fast version of FChT reduces computation [13] but this algorithm presents two factors that affect sinewave parameter estimation. The phase of the fast FChT does not match the phase of the original continuous-time transform and this interferes with the estimation of sinewave phases. This has been solved in [77].

[2] available at `http://www.music-ir.org/mirex`

[3] available at `http://staff.aist.go.jp/m.goto/RWC-MDB/`

1.6.1 Continuous Fan-Chirp Transform

The FChT is defined as

$$X(f,\alpha) = \int_{-\infty}^{\infty} x(t)\sqrt{|\phi'_\alpha(t)|}\, \mathrm{e}^{-\mathrm{j}2\pi f \phi_\alpha(t)} \mathrm{d}t, \tag{1.58}$$

where α is the chirp rate,

$$\phi_\alpha(t) = \left(1 + \frac{1}{2}\alpha\left(t - \frac{T}{2}\right)\right)\left(t + \frac{T}{2}\right) - \frac{T}{2}, \tag{1.59}$$

is the time warping function and $\phi'_\alpha(t)$ denotes first derivative of $\phi_\alpha(t)$. The parameter T in (1.59) is the length of the interval centered at $t = 0$ where the mapping takes effect. Assuming the mapping interval is $(-T/2; T/2)$ the time warping (1.59) can written as

$$\phi_\alpha(t) = (1 + \frac{1}{2}\alpha t)t. \tag{1.60}$$

In order to prevent the derivative of the phase function $\phi_\alpha(t)$ (frequency of the basis functions) from becoming zero the chirp rate α is constrained to

$$|\alpha| < \frac{2}{T}. \tag{1.61}$$

The computation of FChT involves the inner product between $x(t)$ and the complex signals

$$\xi(t,f,\alpha) = \sqrt{|1+\alpha t|}\mathrm{e}^{\mathrm{j}2\pi f(1+(1/2)\alpha t)t}, \tag{1.62}$$

which are chirps whose instantaneous frequency, defined as the time derivative of the exponent, varies linearly over time as

$$v(t) = \phi_\alpha^{-1}(t)f = (1+\alpha t)f, \tag{1.63}$$

where f refers to the instantaneous frequency at $t = 0$. The signal $x(t)$ can be recovered from its FChT as

$$x(t) = \int_{-\infty}^{\infty} X(f,\alpha)\sqrt{|\phi'_\alpha(t)|}\, \mathrm{e}^{\mathrm{j}2\pi f \phi_\alpha(t)} \mathrm{d}f. \tag{1.64}$$

However, there is another condition that has to be met in (1.64) for perfect reconstruction. According to (1.63), the sign of the instantaneous frequency of all basis components switches at the instant $t = -1/\alpha$ which is called 'focal point' instant. Therefore $x(t)$ has to fulfill

$$x(t) = 0 \quad \text{for } t < -\frac{1}{\alpha}, \tag{1.65}$$

otherwise the synthesized signal will be overlaid with itself mirrored around the focal time instant [13]. Also the chirp rate has to be in the range defined by (1.61) otherwise the quadratic mapping $\phi_\alpha(t)$ would not be bijective, i.e. one-to-one mapping.

1.6.2 Discrete-Time Fan-Chirp Transform

For a signal $x(n)$, which is a discrete-time version of the signal $x(t)$ at sampling frequency f_s, the discrete-time FChT is defined as [14]

$$X(f,\alpha) = \sum_{n=0}^{N-1} x(n)\sqrt{\phi'_{\hat{\alpha}}(n)}\mathrm{e}^{-\mathrm{j}2\pi\frac{k}{N}\phi_{\hat{\alpha}}(n)}, \tag{1.66}$$

where k is the frequency bin index, N is the number of segment samples, $\hat{\alpha}$ is related to its continuous-time counterpart $\hat{\alpha} = \alpha/f_\mathrm{s}$, and $\phi_{\hat{\alpha}}$ is the following mapping, bijective in $[0, N]$

$$\phi_{\hat{\alpha}} = \left(1 + \frac{1}{2}\hat{\alpha}(n-N)\right)n. \tag{1.67}$$

The bijectivity of $\phi_{\hat{\alpha}}(n)$ results in the following limits for the chirp rate $\hat{\alpha}$

$$-\frac{2}{N} < \hat{\alpha} < \frac{2}{N}. \tag{1.68}$$

The computational load required to implement is N^2 complex multiplications [13].

1.6.3 Fan-Chirp Transform as Time-warped Fourier Transform

While the discrete-time FChT can be computed directly using (1.66), computational load of the direct version is quadratic. A fast version of the FChT operates reformulating the FChT as the FFT of a time-warped signal, substituting $\tau = \phi_\alpha(t)$ thus significantly reducing computation [13]. The FChT with the variable substitution becomes [77]

$$X(f,\alpha) = \int_{\phi_\alpha(-\frac{T}{2})}^{\phi_\alpha(-\frac{T}{2})} \tilde{x}(\tau)\tilde{\rho}(\tau)\mathrm{e}^{-\mathrm{j}2\pi f\tau}d\tau, \tag{1.69}$$

where $\tilde{x}(\tau)$ is a time-warped version of the signal $x(t)$ and $\tilde{\rho}(\tau)$ is a scaling function on the time-warped axis. It can be seen, that the equation (1.69) is a Fourier transform of the product $\tilde{x}(\tau)\tilde{\rho}(\tau)$. To compute the time-warped input signal $\tilde{x}(\tau) = x(\phi_\alpha(\tau))$, we use inverse of the warping function $\psi_\alpha(\tau) = \phi_\alpha(\tau)^{-1}$. Since $\phi_\alpha(t)$ is a quadratic function, its inverse function has two solutions. The solution of interest is

$$\psi_\alpha(t) = -\frac{1}{\alpha} + \frac{\sqrt{1+2\alpha t}}{\alpha}. \tag{1.70}$$

The scaling function $\tilde{\rho}(\tau)$ can be shown to be [77]

$$\tilde{\rho}(\tau) = \sqrt{|\phi'_\alpha(\psi_\alpha(\tau))|}\psi'_\alpha(\tau), \tag{1.71}$$

which can be expressed as

$$\tilde{\rho}(\tau) = \frac{1}{\sqrt[4]{1+2\alpha\tau}}. \tag{1.72}$$

This concludes the computation of FChT by first time warping and scaling the input signal using (1.70) and (1.72), respectively and then computing the Fourier transform of the result. In discrete time equation (1.69) can be written as

$$X(k,\hat{\alpha}) = \sum_m \tilde{x}(m)\tilde{\rho}(m)\mathrm{e}^{-\mathrm{j}2\pi\frac{k}{K}m}, \tag{1.73}$$

where the range of m will be defined below. Again the equation (1.73) can be evaluated using FFT of the product $\tilde{x}(m)\tilde{\rho}(m)$. The discrete-time signal $\tilde{x}(m)$ is created by uniformly sampling the continuous time signal $\tilde{x}(\tau)$. Due to time warping, the signal $\tilde{x}(\tau)$ has greater bandwidth when $\hat{\alpha}$ is nonzero. In other words, the warping rule has a slope greater than $\phi'_\alpha(t) > 1$ for $\alpha t < 0$. This implies that signal $\tilde{x}(m)$ is undersampled on that region, leading to undesired aliasing effects [13]. This undesired aliasing is reduced or even suppressed by setting the length M to a proper value. It is clear that $M > N$. In order to have N aliasing-free bins (out of M), M has to be set as

$$M \geq \frac{1-|\hat{\alpha}|N/4}{1-|\hat{\alpha}|N/2}N. \tag{1.74}$$

The range of $\tilde{x}(\tau)$ is $\phi_\alpha(-\frac{T}{2}) \geq \tau \geq \phi_\alpha(\frac{T}{2})$ which can be shown to have duration T [77]. In [13] the time warped signal $\tilde{x}(\tau)$ is sampled at time instants

$$\tau_m = \phi_\alpha\left(-\frac{T}{2}\right) + \left(m+\frac{1}{2}\right)\frac{T}{M} \qquad \text{for } 0 \leq m \leq M, \tag{1.75}$$

where $\frac{T}{M}$ is the sampling period on the time-warped axis. The definition (1.75) of time instants τ_m at which the time-warped signal is sampled, is selected from the M samples symmetrically between the endpoints of the time-warped signal. Unfortunately this definition has the side effect of adding a delay to the discrete-time signal $\tilde{x}(m)$ [77]. The delay changes the phase of the FChT such that it does not match the phase of the continuous time FChT given in (1.69). This introduces a phase shift, which has to be removed from the FChT before it can be used to estimate phases of sinewave parameters properly [77]. An alternative method to avoid the need for phase correction is to redefine the samples at which the time-warped signal is sampled as

$$\tau_m = m\left(\frac{T}{M}\right). \tag{1.76}$$

The range of m in (1.76) is derived from the relationship $\phi_\alpha(-\frac{T}{2}) \geq \tau \geq \phi_\alpha(\frac{T}{2})$, which yields

$$M\left(\frac{1}{8}\hat{\alpha}N - \frac{1}{2}\right) \leq m \leq M\left(\frac{1}{8}\hat{\alpha}N + \frac{1}{2}\right). \tag{1.77}$$

1.6.4 Fundamental Frequency and Chirp Rate Estimation

The most important aspect of the FChT regards the adequacy of the law (1.70) to the actual time-frequency characteristics of the signal [13]. Considering segment-wise processing, the chirp rate $\alpha(t)$ that best matches time-frequency characteristics of the segment is doubtlessly the decisive factor on using FChT-based spectrogram instead of the STFT-based spectrogram. This estimation can be carried out using two methodologies: inter-frame and intra-frame.

Inter-frame

Assuming the signal shows a continuous evolution of its fundamental frequency $f_0(t)$ according to the instantaneous frequency of the fan geometry, the best estimation of the pitch rate is [13]

$$\alpha(t) = \frac{f_0'(t)}{f_0(t)}, \tag{1.78}$$

where $f_0'(t)$ is time derivative of $f_0(t)$. The intuitive approach would be to quantify the evolution of pitch $f_0(t)$ and then compute the chirp rate using (1.78). Many methodologies for estimating f_0 exist [78]. It is common in segment-wise processing, that the f_0 is estimated at instants $t = nS$, where S is a shift interval between segments. After the pitch has been estimated in the neighboring segments around the n-th segment, the pitch rate can be obtained as [13]

$$\alpha(n) = \frac{f_0(n+1) - f_0(n-1)}{2Sf_0(n)}, \tag{1.79}$$

where S is the shift interval between segments. Estimation of chirp rate for the n-th segment using (1.78) requires the pitch of the next segment. This non-causal method implies to step one segment back to recompute the FChT of the n-th segment once that the pitch of the $(n+1)$-th segment is available [13].

Intra-frame

While in the inter-frame approach used pitch information from adjacent segments, the intra-frame method uses only information from the current segment. One method of computing the chirp rate of the current segment is computing a dense (α, f) plane. The (α, f) shows spread in the shape of a bow tie, which is typical in

chirp-based transforms [13]. The chirp rate best representing the harmonic information contained in the signal will be in the place of highest concentration of peaks in the (α, f) plane. To skip the large computational load of computing the redundant (α, f) plane, chirp rate can be estimated from L FChT instances for different chirp rates α. It has been shown that three instances ($L = 3$) are sufficient for that purpose [13].

The pitch and pitch rate can be both estimated using pitch salience. Its aim is to build a continuous function that gives a prominence value for each fundamental frequency in a range of interest [66]. Ideally it shows pronounced peaks at the positions corresponding to the true pitches presented in a signal frame. The salience of a given fundamental frequency candidate f_0 can be obtained by gathering the log-spectrum at the positions of the corresponding harmonics [14]

$$\rho_0(f) = \frac{1}{n_{\mathrm{H}}} \sum_{i=1}^{n_{\mathrm{H}}} \log |S(if)|, \tag{1.80}$$

where $|S(f)|$ is the power spectrum, n_{H} is hypothetical number of harmonics in the Nyquist band and i is order of harmonic component. Linear interpolation from the discrete log-spectrum is applied to estimate the values at arbitrary frequency positions [66]. Gathering the linear spectrum was initially proposed [79] as a method for detecting periodic signal when the period is unknown. However, the same gathering procedure on the logarithmic power spectrum delivers higher accuracy and noise robustness than working on the linear spectrum, as well as robustness against the formant structure [14]. Since the harmonic accumulation when using (1.80) shows peaks not only at the position of the true pitch, but also at multiples and submultiples. To handle the ambiguity produced by multiples, a simple non-linear processing is proposed in [14]

$$\rho(f) = \rho_0(f) - \max_{q \in \mathbb{N}} \rho_0(f/q) \qquad q = 1, 2, 3, ... \tag{1.81}$$

This is effective in removing pitch candidates multiples of the actual one. Submultiple spurious peaks do not affect the estimation because their amplitude is necessarily lower than the true pitch for the monophonic case [66].

2 THESIS OBJECTIVES

From the methods for representation of non-stationary signals with time-variant frequency components presented in the previous chapter this thesis will deal with the Harmonic Transform. Specifically with decreasing its computational demands. Knowledge of fundamental frequency change is required before computing the Harmonic Transform. This is done by Spectral Flatness Measure and our first focus will be on optimizing its computation. Unfortunately, the Harmonic Transform computation still employs $\mathcal{O}(N^2)$ computational complexity. So the next focus will be on obtaining a Harmonic Transform which employs subquadratic computational complexity. This will be attempted substituting the time-warping kernel of Harmonic Transform with time-warping of the time axis. Since we usually only have discrete signals available, it is necessary to use interpolation which introduces noise into the signal. This renders Spectral Flatness Measure ineffective for the computation of fundamental frequency change as will be shown in the next chapter. Therefore a different method of fundamental frequency change estimation is needed. Two methods will be presented in this thesis. The first one computes fundamental frequency change using a method which has been used in computation of Fan-Chirp Trasnform, the gathered log-spectrum which performs gathering of the logarithm of the magnitude spectrum at the places of the fundamental frequency and its multiples. The second method selects the optimal fit of fundamental frequency change by comparing the reconstruction error of the harmonic part of the signal which is estimated using the Harmonic Transform centered on the fundamental frequency. Both of these methods will be tested on the same speech signal to compare their approach.

Since the Fast Harmonic Transform uses interpolation for its fast computation, there will inevitably be artifacts caused by the interpolation. This will be even more pronounced in the signal reconstructed using Inverse Fast Harmonic Transform from the harmonic domain. The reconstruction error will be measured for several interpolation methods. Another artifact present in the Fast Harmonic Transform image is aliasing and it will be addressed using oversampling and evaluated for different oversampling factors and interpolation methods.

All of the papers dedicated to Harmonic Transform present improvment of representation of non-stationary signals with time-varying components only on speech signals sampled at sampling frequency 8 kHz. One of the goals of this thesis is to attempt the application of Harmonic Transform on real signals such as vocals or instruments with significant frequency modulation which have been sampled at sampling frequency 44.1 kHz.

To summarize the goals of this thesis, they can be divided into these main areas:

- Fast Harmonic Transform Algorithm
- Fast Inverse Harmonic Transform Algorithm
- Computational load of the Fast Harmonic Transform
- Fundamental frequency change estimation using gathered log-spectrum
- Fundamental frequency change estimation using analysis-by-synthesis approach
- Aliasing artifacts and anti-aliasing by oversampling
- Experiments on real frequency-modulated signals

3 RESEARCH RESULTS

This chapter deals with efficient implementation of the Harmonic Transform. The original implementation requires $\mathcal{O}(N^2)$ operations. The first approach to reduce the number of computations required to compute HT is to reduce the number of operations for computation of spectral frequency measure. This is done by exploiting redundancy in its algorithm. Since the Harmonic Transform does not produce mirrored double-sided spectrum and only the left side spectrum represents the harmonic information of the analyzed signal, the SFM computation can be carried out on half of the spectrum, reducing the number of operations needed.

This however still leaves an algorithm with quadratic computational complexity, so the research is then focused on producing an algorithm with subquadratic computational complexity. This is achieved by time-warping the input signal, where the relationship between the warped axis and original axis is given by the transformation kernel of the HT. By this reduction of computational complexity, another problem emerges. The interpolation used in time-warping introduces some noise to the signal if spectral flatness is used for fundamental frequency change computation. Methods which can be used for fundamental frequency change estimation instead of SFM have been used with the Fan-Chirp Transform and are presented in chapter 1.6.4. The general approach is to compute several transformations with different values of fundamental frequency and fundamental frequency change where the most suitable parameters are picked from a 2D representation of the analyzed data.

Several audio samples are used for demonstration of the presented methods. Their list can be found in appendix A. In some cases a synthetic linear chirp signal (*test signal*) is used for demonstration. It is defined as

$$x(t) = \sum_i \sin(2\pi(f_0 i + i\frac{k}{2}t^2)) \quad i = 1, ..., 12 \tag{3.1}$$

where $t \in (0, T)$, T is length of the segment, f_0 is fundamental frequency, i is the number of harmonic, and k is the chirp rate defined as $k = \frac{\Delta f}{T}$.

3.1 Reducing The Number Of Computations Of The Harmonic Transform

One of the crucial steps in computation of the Harmonic Transform is to estimate the fundamental frequency change of the analyzed signal. So far, algorithm based on SFM has been used and can be found in section 1.5.2. When exploring this algorithm, several observations have been made. When using (1.51) the SFM has several minimums and if a search algorithm was used, it could fall into local minimum. It is

also noteworthy that it is possible the harmonic transform $|\mathrm{DHT}(a,k)|$ will be equal to zero for some values of k, which could mean that the spectral flatness will be zero for all a. Removing zero values solves this problem and leads to band-limited spectral flatness measure [80].

Harmonic spectrum of the Harmonic Transform computed using (1.49) is not complex conjugated even for real signals (which is true for Fourier transform). From the frequency axis point of view, the unit phase function $\phi_{\mathrm{u}}(t)$ shifts the spectrum towards lower frequencies if a is positive, and to higher frequencies if it is negative. Using the formula (1.49) we get only one-sided spectrum (see Fig. 3.2), the right part will not represent harmonic components of the analysed signal [80]. When estimating a, (1.51) has two minimums (see Fig. 3.1). For harmonic signal analysis, only left side of the spectrum is useful, because it appropriately represents non-stationary harmonic signal. Using the modified spectral flatness measure (MSFM)

$$\arg\min_{a} \mathrm{MSFM}(a) = \frac{\sqrt{\prod_{k=0}^{N/2} |\mathrm{DHT}(a,k)|}}{\frac{1}{N/2+1}\sum_{k=0}^{N/2} |\mathrm{DHT}(a,k)|} \tag{3.2}$$

we can get function of a which has clearly defined minimum [81]. This is caused by using only left side of the spectrum when computing SFM and it consequently leads to reducing the number of operations needed to compute spectral flatness by $\frac{N}{2} - 1$ [81].

While this approach reduces the number of operations needed to compute the Harmonic Transform, it does not reduce the asymptotic computational complexity and the computation has $\mathcal{O}(N^2)$ complexity. Further research on the reduction of the number of operations used to compute Harmonic Transform has been therefore focused on developing a subquadratic method of Harmonic Transform computation.

3.2 Fast Harmonic Transform

The number of operations in direct computation of the HT from (1.49) raises quadratically, similarly to direct computation of Fourier transform. The goal of this section is to present an algorithm to compute the HT which shows sub-quadratic complexity. When there is a transform with quadratic complexity, then its sub-quadratic form is referred to as the fast version of the transform. In this case it is the Fast Harmonic Transform (FHT). One of the ways to produce a fast transform of a transform which depends on time-warping of the time axis, is to separate it into a time-warping operation and a Fast Fourier Transform. This has been demonstrated in the case of Fan-Chirp Transform [14] and Mellin Transform [82]. And this principle is also used here for devising the FHT.

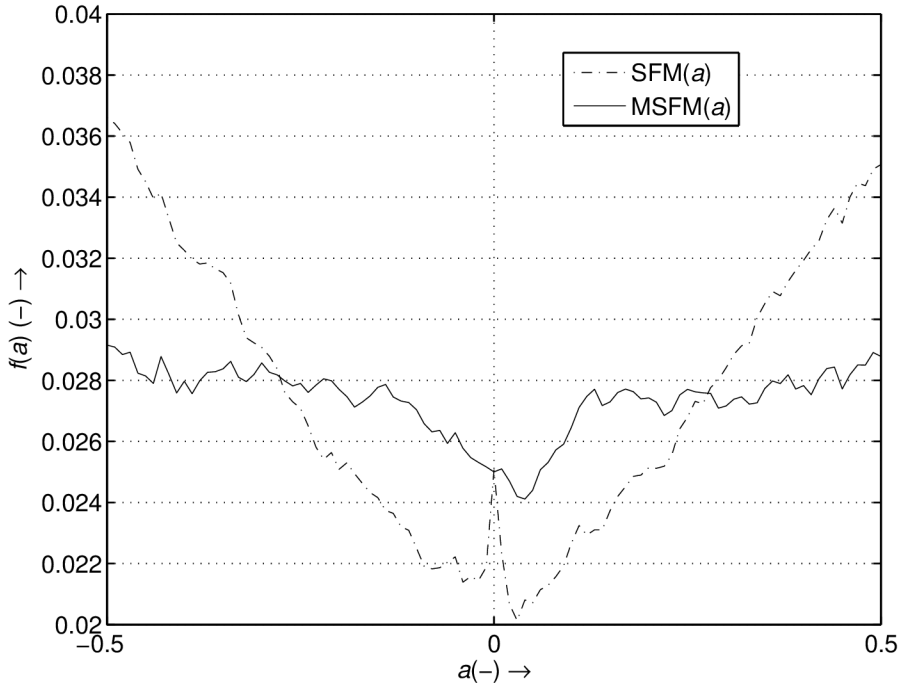


Fig. 3.1: Comparison of spectral flatness measure and modified spectral measure of the sound sample *sopranoshort*. The analysed segment is 1024 samples long.

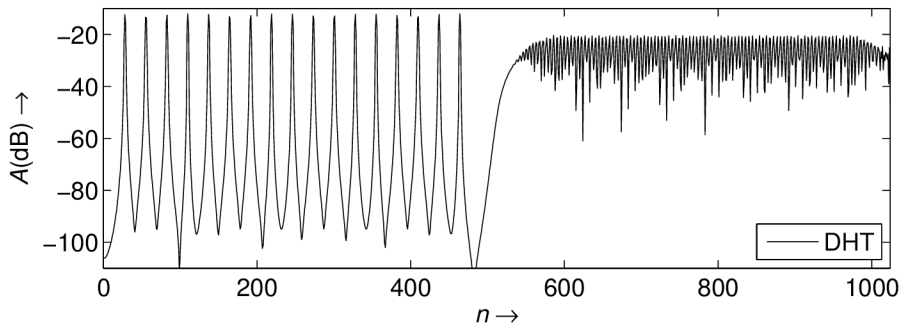


Fig. 3.2: Harmonic Transform image (harmonic spectrum) of a harmonic chirp signal showing one-sided spectrum.

By substituting $\tau = \alpha_a(t)$ in (1.36) it becomes

$$S(\omega, a) = \int_{\alpha_a(0)}^{\alpha_a(T)} \tilde{s}(\tau)\tilde{\rho}(\tau)\mathrm{e}^{-\mathrm{j}\omega\tau}\mathrm{d}\tau, \tag{3.3}$$

where $\tilde{s}(\tau)$ is a time-warped version of $s(t)$ and $\tilde{\rho}(\tau)$ is a scaling function on the time-warped axis. To compute the time-warped input signal $\tilde{s}(\tau)$ we need the inverse of the warping function $\psi_a(\tau) = \alpha_a^{-1}(\tau)$ which then gives $\tilde{s}(\tau) = s(\psi_a(\tau))$. The inverse of $\alpha_a(\tau)$ is a quadratic function which has two solutions. The solution of interest is

$$\psi_a(\tau) = \frac{T}{2} - \frac{T}{a} + \frac{T\sqrt{(\frac{a^2}{4} - a + \frac{2a\tau}{T} + 1)}}{a}, \tag{3.4}$$

where T is the length of the analyzed segment. The scaling function is then defined as

$$\tilde{\rho}(\tau) = \alpha_a'(\psi_a(\tau))\psi_a'(\tau). \tag{3.5}$$

Equation (3.3) can be seen as Fourier Transform of the product of $\tilde{s}(\tau)\tilde{\rho}(\tau)$. This enables efficient implementation in discrete time based on the FFT. Further analysis will therefore be focused on the discrete-time FHT.

Discrete-Time Fast Harmonic Transform

The equation (3.3) can be written in discrete time as

$$S(k, a) = \sum_{n=0}^{N} \tilde{s}(n)\tilde{\rho}(n)\mathrm{e}^{-\mathrm{j}2\pi\frac{k}{K}n} \tag{3.6}$$

which is a FFT of the product $\tilde{s}(n)\tilde{\rho}(n)$ which is the uniformly sampled product $\tilde{s}(\tau)\tilde{\rho}(\tau)$. Since we usually only have discrete signals available, we will use discrete-time intervals n even though its value can be non-integer. Any values at non-integer intervals will be enumerated using interpolation from the signal samples. Now to get a discrete-time counterpart of (3.4) we take $\alpha_a(n)$ which is a quadratic function and its inverse $\alpha_a^{-1}(n)$ yields two results. The result of interest is

$$\psi_a(n) = \frac{N}{2} - \frac{N}{a} + \frac{N\sqrt{(\frac{a^2}{4} - a + \frac{2an}{N} + 1)}}{a}, \tag{3.7}$$

where n is sample index and N is number of samples [83]. Plot of the warping function $\alpha_a(n)$ and its inverse warping function $\psi_a(n)$ can be seen in Fig. 3.3. A demonstration of the warping function (3.7) on a signal with linear frequency change can be seen in Fig. 3.4. The warping function is used to time-warp the signal with linear frequency change to a signal with stationary frequency f_c which corresponds to the frequency of the signal with linear frequency change at time $t = 0$, or $n = N/2$ for discrete-time signals.

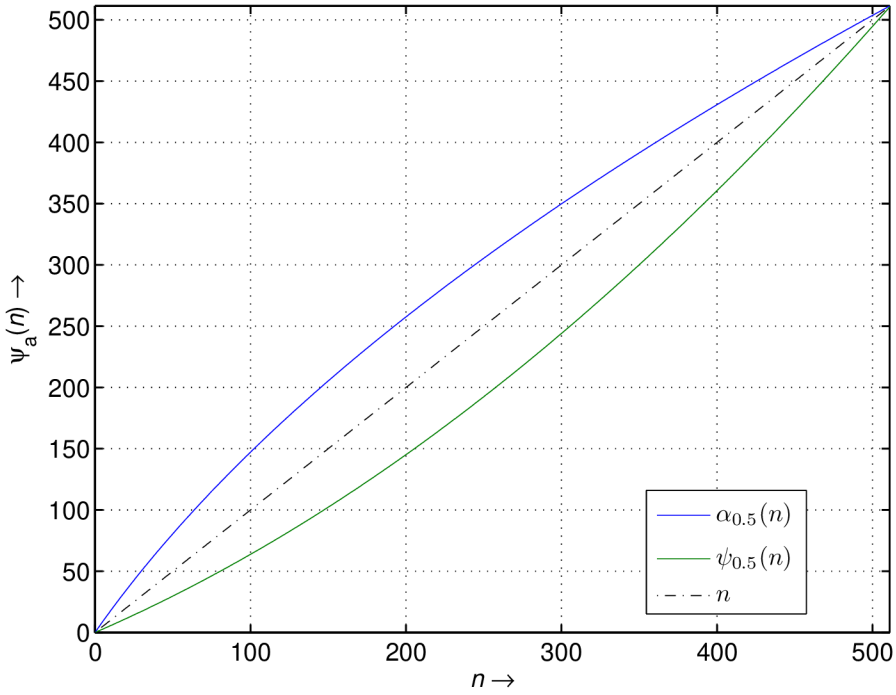


Fig. 3.3: Mapping from the original time axis to the time-warped axis given by the discrete-time warping function $\psi_a(n)$.

With (3.7) we can define the discrete-time time-warped signal from (1.49) as

$$s_a(n) = \tilde{\rho}(n)\tilde{s}(\psi_a(n)), \tag{3.8}$$

where $\tilde{\rho}(n) = \phi_a'(\psi_a(n))^{-1}$ is the scaling factor which can be written as

$$\tilde{\rho}(n) = \left(-\frac{a}{2} + \frac{\frac{N}{2} - \frac{N}{a} + \frac{N\sqrt{a^2/4 - a + \frac{2an}{N} + 1}}{a}}{N} + 1 \right)^{-1} \tag{3.9}$$

and $\tilde{s}(\psi_a(n))$ is the time-warped signal [83]. The last step to compute the HT is using FFT on the time-warped signal $s_a(n)$ as follows [83]

$$S(k, a) = \sum_{n=0}^{N-1} s_a(n) \mathrm{e}^{-\mathrm{j}2\pi \frac{k}{N} n}. \tag{3.10}$$

Now we have a Fast Harmonic Transform for harmonic signals with linear frequency change which in the next step will be turned into an algorithm which will enable its use for analysis and synthesis in the harmonic domain.

Harmonic Transform Algorithm

It has been stated in 1.5.2, that fundamental frequency change estimation is needed for correct representation of a signal using the HT. The fundamental frequency

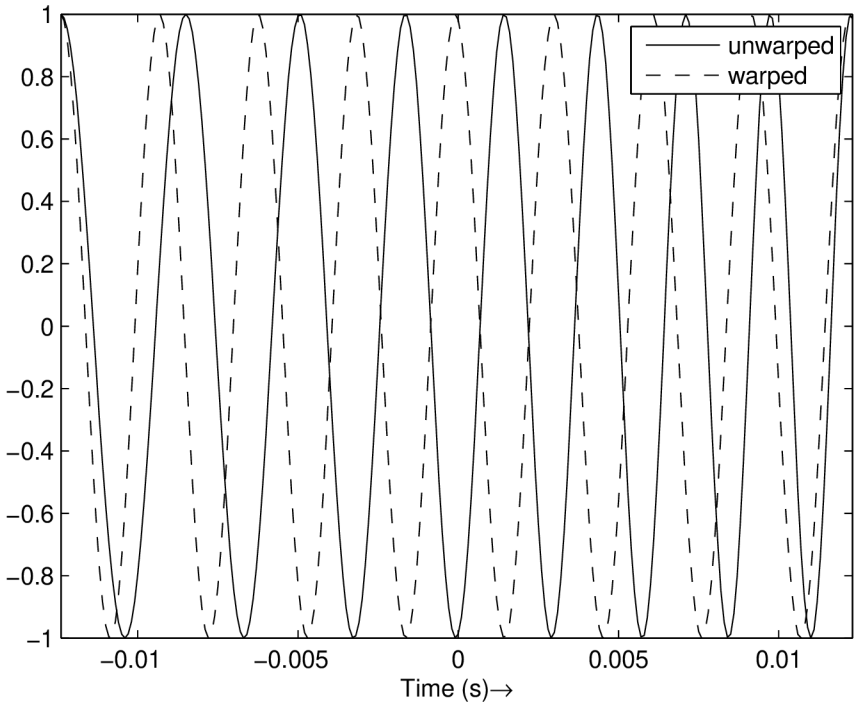


Fig. 3.4: The solid line represents a linear chirp and the dashed line represents a sinusoid with frequency equal to the linear chirp's frequency at $t = 0$ also called the central frequency f_c.

change estimation has been carried out using spectral flatness measure. The searching algorithm for fundamental frequency change estimation is following: It starts by searching the fundamental frequency change by measuring harmonic spectrum for different unit phase functions, i.e. phase functions with different fundamental frequency slope a. The optimal parameter a is defined by the harmonic spectrum with the lowest SFM as can be seen in Fig. 3.6. It indicates the highest concentration of spectral peaks, an optimal fit of the transformation kernel for the signal. This means the optimal fundamental frequency change also has been found. In FHT, we are using interpolation to obtain a time-warped version of the analyzed signal. The interpolation adds noise and errors to the signal, creating more peaks in the spectrum that do not represent the analyzed signal, even when using high-quality interpolation, which renders spectral flatness inaccurate as can be seen in Fig. 3.7. It is therefore necessary to use a different method for fundamental frequency change estimation which will be covered in sections 3.3 and 3.4. Fundamental frequency change can also be computed from fundamental frequencies obtained by another fundamental frequency estimating algorithm. If we use the estimation algorithm to estimate fundamental frequencies at the beginning and at the end of segment, the slope of linear frequency change can be computed from (1.45) as

$$a = \frac{f(N) - f(0)}{f(N/2)}, \tag{3.11}$$

where $f(0), f(N/2)$, and $f(N)$ is the instantaneous frequency of the fundamental frequency at the beginning, middle, and the end of the segment respectively [83].

Fast implementation of the Harmonic Transformation is based on (3.10), though its actual implementation employs several improvements. Block diagram of the Harmonic Transform algorithm is shown in Fig. 3.5. The algorithm consists of:

1. Upsampling - Since the interpolation introduces noise to the signal, which is most pronounced in higher frequencies, it may be advantageous, depending on the application, to introduce upsampling to increase the quality of the transformed signal. This operation increases the number of samples and operations by the upsampling factor. Chapter 3.6 deals with the aliasing problem.
2. Windowing - Hann window is used for windowing.
3. Normalization - When the analyzed signal has fundamental frequency change, the transformation can introduce energy leakage to neighboring spectral lines. To deal with this phenomenon, the window is adapted to the frequency change.
4. Interpolation - Since the phase function $\psi_a(t)$ will likely not fit the discrete-time signal sampled at uniform time intervals, interpolation of the signal values is necessary.
5. Zero-phase zero padding - If we want to be able to determine phases of the harmonics, we need a zero-phase Fourier Transform implementation. This

bounds the transformation length to odd number of samples. In this phase, zero samples are appended to the signal buffer.

6. FFTshift - This block carries out the actual zero-phase zero padding as defined by

$$f(n) = \begin{cases} n + N - \frac{(M-1)}{2} & \text{for } n \leq \frac{M-1}{2}, \\ n - \frac{(M-1)}{2} + 1 & \text{for } n > \frac{M-1}{2}, \end{cases} \tag{3.12}$$

   where M is the input buffer length and N is the total number of samples used for the transform including zero samples from zero-padding. The first $(M-1)/2$ samples of the windowed data is stored at the end of the buffer from sample $N-(M-1)/2$ to $N-1$. The remaining samples will be stored starting at the beginning of the buffer from sample 0 to sample $(M-1)/2$. All zero padding occurs in the middle of the FFT buffer.
7. FFT - Preforms the Fast Fourier Transform.

Now we should have an efficient algorithm to compute the FHT. It is noteworthy that the harmonic spectrum of the FHT is double-sided as seen in Fig. 3.8, whereas the harmonic spectrum of DHT is one-sided. This allows for modifications in the spectrum like linear filtration, convolution, or correlation. After we obtain the harmonic spectrum and perform some modifications, an algorithm to return the signal to the time domains is required. This will be the contents of the following section.

3.2.1 Inverse Fast Harmonic Transform

Inverse Fast Harmonic Transform (IFHT) is the inverse transform to the Fast Harmonic Transform. It can be used to obtain a time domain signal from a harmonic spectrum and its estimated fundamental frequency slope a. The IFHT is defined as

$$s(n) = \frac{1}{N} \sum_{n=0}^{N-1} S(k,a) \mathrm{e}^{\mathrm{j}2\pi \frac{k}{N} n}. \tag{3.13}$$

An algorithm to compute the IFHT is very similar to the algorithm of FHT with reversed block order. The block diagram is in Fig. 3.9. Description of the blocks follows.

Inverse Harmonic Transform Algorithm

1. IFFT - Performs the inverse Fast Fourier Transform.
2. FFTshift - Returns the shifted samples in the buffer to their correct order. This is done simply by applying the formula (3.12) again. Any added zeroes are simply discarded if zero padding was used.

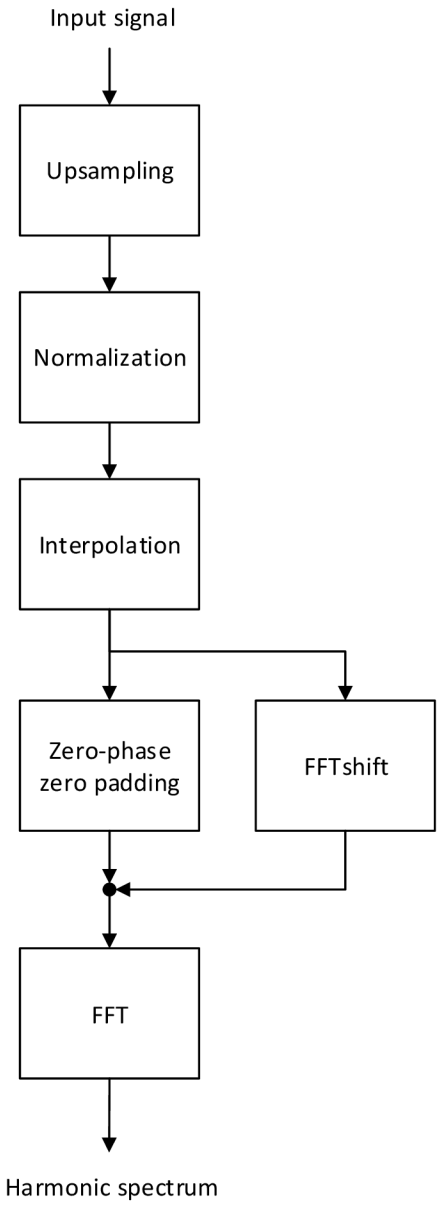


Fig. 3.5: Block diagram of the forward Fast Harmonic Transform.

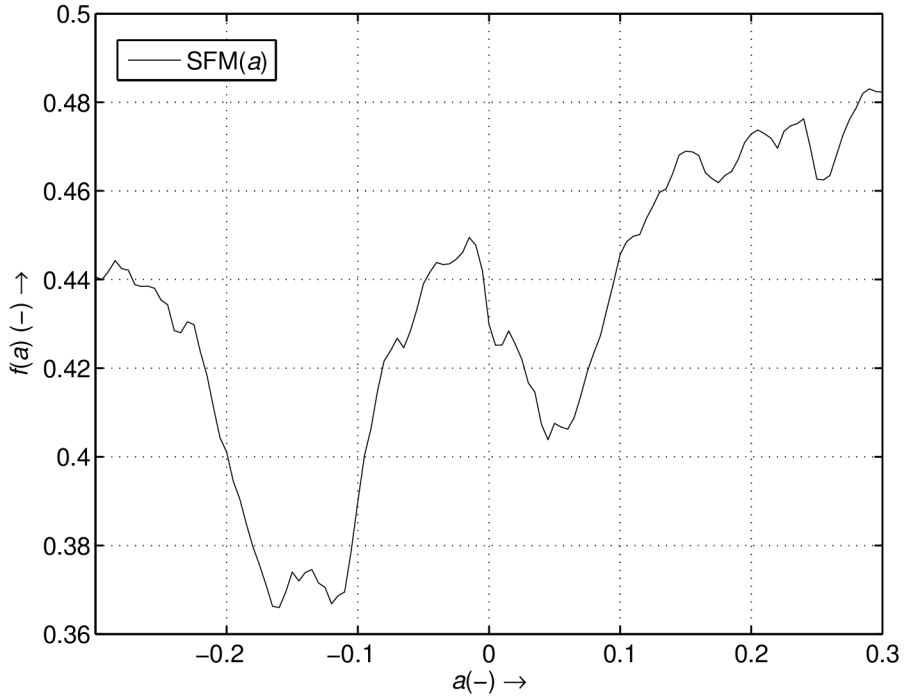


Fig. 3.6: Spectral flatness measure obtained using Harmonic Transform for a voiced speech segment *happychild* with frequency modulation.

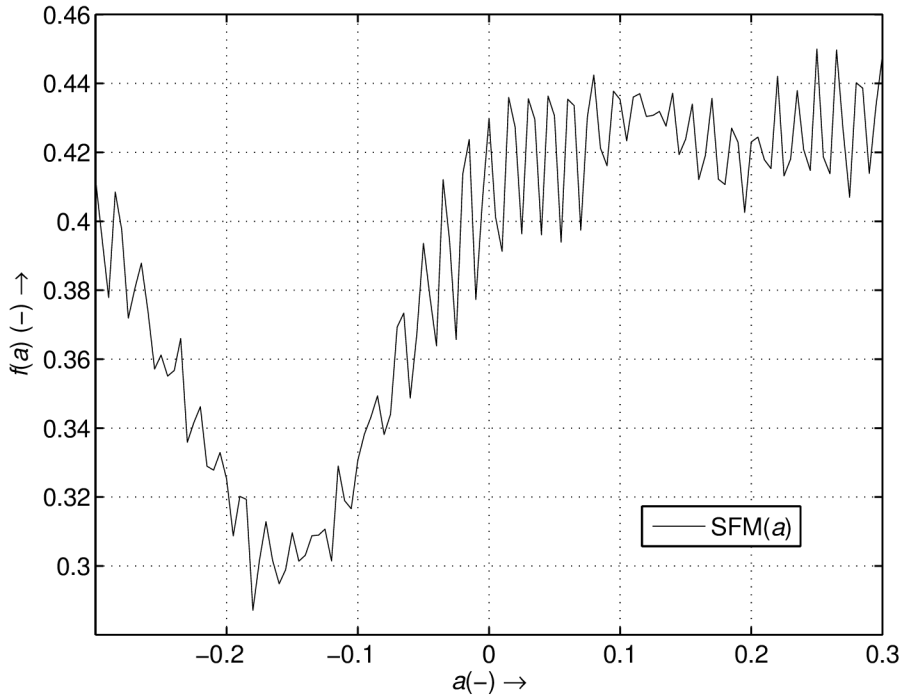


Fig. 3.7: Spectral flatness measure obtained using Fast Harmonic Transform with linear interpolation for a voiced speech segment *happychild* with frequency modulation.

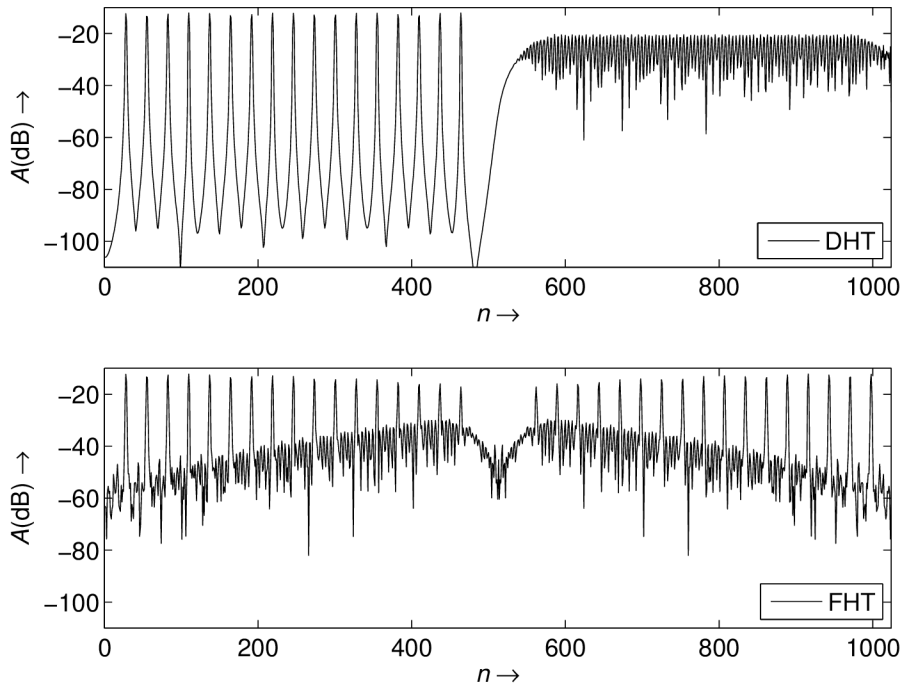


Fig. 3.8: Discrete Harmonic Transform of *happychild* sound sample shows that its double-sided harmonic spectrum is not mirrored, while the Fast Harmonic Transform with linear interpolation shows mirrored two-sided spectrum with aliasing.

3. Interpolation - Since mapping from the warped time axis $\psi_a(t)$ to the natural time axis t will likely not fit the discrete-time signal sampled at time-warped intervals, interpolation of the signal values is necessary.
4. Normalization - This is the inverse of the normalization operation in the forward Fast Harmonic Transform. It simply relieves the fundamental frequency-adapted windowing of the forward Fast Harmonic Transform.
5. Downsampling - If upsampling has been used in the forward harmonic transform, the time domain signal is downsampled.

Tab. 3.1: SNR (dB) of a speech signal *micf01sa02* reconstructed using IFHT from a harmonic spectrum obtained by FHT.

interpolation method	oversampling		
	1x	2x	4x
linear	17.9	28.0	37.1
cubic	22.0	37.7	42.6
spline	28.0	42.4	42.9

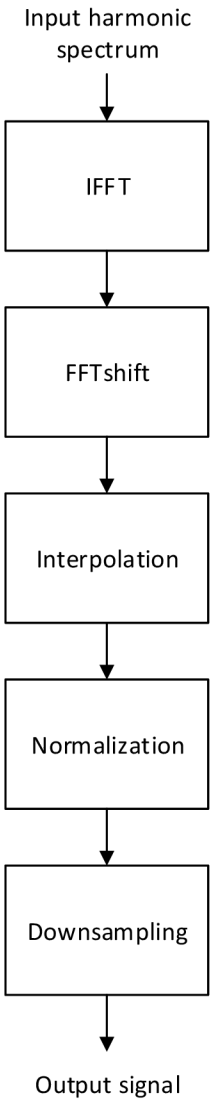


Fig. 3.9: Block diagram of the Inverse Fast Harmonic Transform computation.

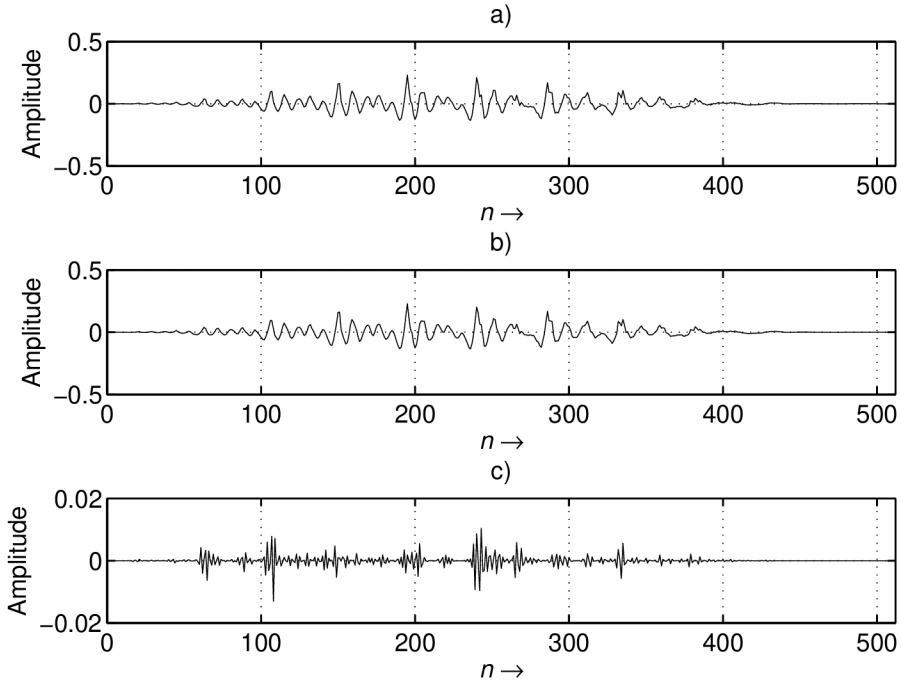


Fig. 3.10: Reconstruction error of a segment of a speech signal *micf01sa02*; a) original signal; b) reconstructed signal; c) residual signal.

The interpolation methods used for linear, cubic, and spline interpolation are the matlab `interp1` parameters 'linear', 'cubic', and 'spline', respectively. Since the IFHT contains a second interpolation (the first one is used in obtaining the harmonic spectrum using FHT), the reconstructed signal is quite likely to have more noise than the input signal. This is depicted in Fig. 3.10, where the input and reconstructed signals are subtracted to obtain the noise signal. To quantify the reconstruction error, we have enumerated the SNR of reconstructed signals using IFHT for different interpolations and several oversampling factors for the signal *micf01sa02* in Tab. 3.1. It can be seen from the table that we can use a cheaper interpolation method if we use oversampling. The relationship betwen noise in the reconstructed signal and oversampling factor is further explored in section 3.6.

The signal-to-noise ratio (SNR) is computed using

$$\mathrm{SNR} = 10 \log \left(\frac{P_\mathrm{s}}{P_\mathrm{n}} \right), \tag{3.14}$$

where P_s is the power of measured signal and P_n is the power of noise. The power is computed using

$$P = \frac{1}{N} \sum_{n-1}^{N-1} x^2(n), \tag{3.15}$$

where P is the power, $x(n)$ is the signal sample, N is the signal length. The actual implementation of signal power estimation employs the Matlab function `NORM` which

computes the signal power through $\mathcal{L}_2$

$$P = \frac{||x||_2^2}{N}, \tag{3.16}$$

with the norm defined as

$$||\mathrm{x}||_\mathrm{p} = \left(\sum_{n=0}^{N-1} |x(n)|^p \right)^{\frac{1}{p}}. \tag{3.17}$$

The $\mathcal{L}_2$ norm can also be viewed as the Euclidean distance.

3.3 Estimation of Fundamental Frequency Change Using Gathered Log-Spectrum

If the Harmonic Transform is to accurately represent a harmonic signal with linear frequency change, estimation of the signal's fundamental frequency is its indispensable part. This method for fundamental frequency estimation is inspired by the method used in Fan-Chirp Transform which is presented in Section 1.6.4. A block diagram of this method can be seen in Fig. 3.11. Its principle is computation of gathered log-spectrum (1.80) for a predefined range of fundamental frequencies and fundamental frequency changes based on the nature of the analyzed signal. Then (a, f_0) plane is constructed from the gathered log-spectrum values (as shown in Fig. 3.12) which represent pitch salience and the most likely candidates for fundamental frequency are represented as peak values. For signals with dominant first harmonic component the first candidate with highest value is usually equal to the fundamental frequency in the analyzed signal. The resulting fundamental frequency f_0 and its slope a is taken from the maximum value of the gathered log-spectrum.

The equation (1.80) computes logarithm of the magnitude spectrum, where the logarithm provides better results compared to the gathering of the linear spectrum making it more robust against formant structure [66]. In [66] p-norm with $0 < p < 1$ has been used to obtain similar results. Therefore the gathered log-spectrum used here is defined as

$$\rho_0(f) = \frac{1}{n_\mathrm{H}} \sum_{i=1}^{n_\mathrm{H}} \log \gamma |S(if)| + 1, \tag{3.18}$$

where higher γ tends to 0-norm and lower γ tends to 1-norm.

During experiments with this method, it has been observed that the number of harmonics n_H used in computation of (3.18) influences fundamental frequency estimation. At higher n_H, the f_0 estimation precision improved at the cost of increased noise sensitivity. At lower n_H, the f_0 estimation was less sensitive to noise at the cost of decreased f_0 estimation precision. The same conclusion can be reached by

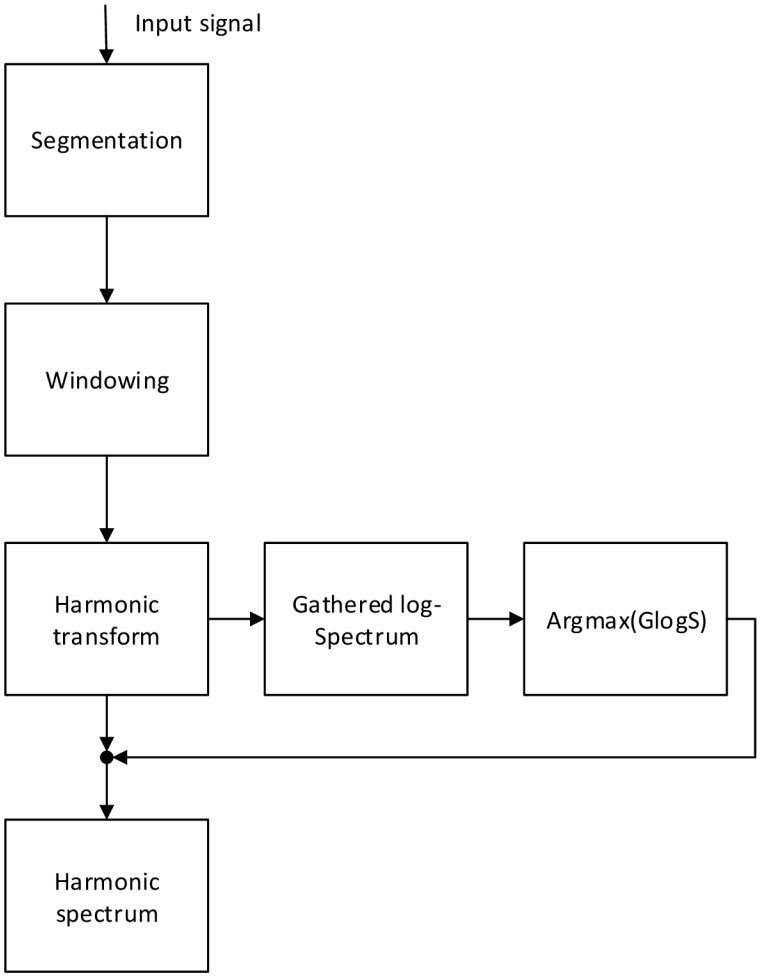


Fig. 3.11: Block diagram of Harmonic Transform computation with f_0 estimation using gathered log-spectrum.

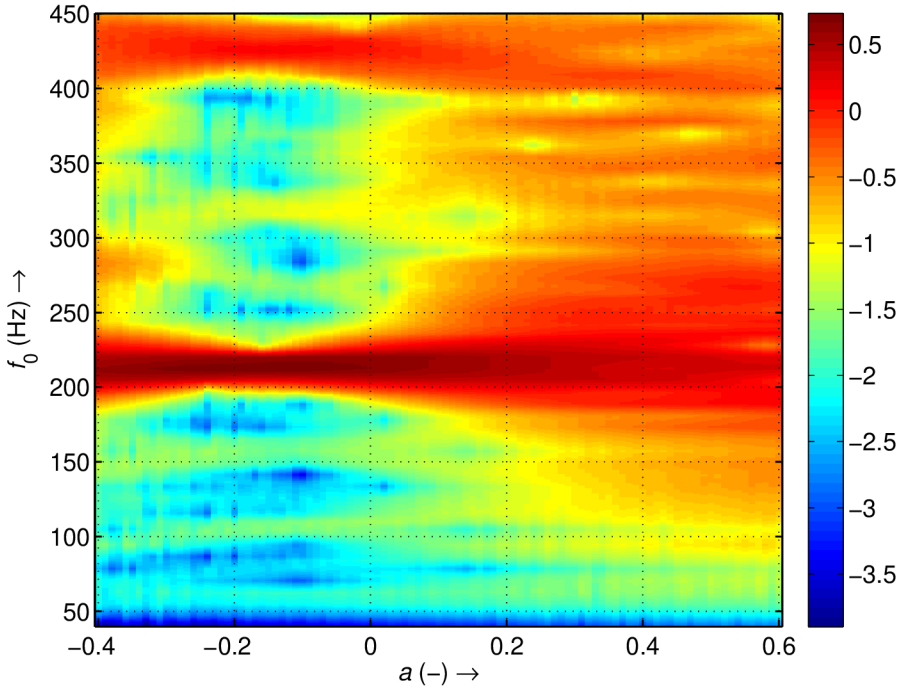


Fig. 3.12: Pitch salience on (a, f_0) plane for a speech signal *micf01sa02* at $t = 359.6$ ms, $M = 255$, $NFFT = 255$, $f_s = 8000$ Hz, $n_H = 4$, with estimated $f_0 = 212.8$ Hz and $a = -0.17$.

reasoning. Considering we are analyzing speech signals, the presence of a Gaussian noise in the signal could mask higher harmonic components as they tend to have lower energy, while the lower harmonic components can still be prominent in the magnitude spectrum since most of the speech signals' energy is concentrated in the lower frequencies. Now if we take the same speech signal with higher harmonic components unaffected by noise and take a sum of several harmonic components, the fundamental frequency estimate will have to be more precise to hit the peak in gathered log-spectrum plane, as the higher frequency components will have greater frequency fluctuation and will require more precise input fundamental frequency in (3.18) to fit the higher frequency component.

Algorithm outline

The algorithm consists of several steps defined as follows:

1. Segmentation - The audio signal is divided into segments for segment-wise processing. Length of the segments depends on the analyzed signal, specifically on its fundamental frequency change. It is necessary to adjust the length of the segment so that the fundamental change throughout the segment is approximately linear. This will make sure the harmonic transform will give a

fine representation of the frequency-modulated harmonic content of the signal.

2. Windowing - is usually used to suppress spectral leakage on the borders of the segment. Hann and Hamming windows have given good results.
3. Harmonic Transform - computes the harmonic transform using the algorithm presented in 3.2.
4. Gathered log-Spectrum - computes the pitch salinity on (a, f_0) plane. This method is based on the gathered log-spectrum introduced in section 1.6.4 for FChT, but uses the harmonic spectrum instead. The number of harmonics used for computation of gathered log-spectrum n_{H} has an impact on f_0 estimation. For lower n_{H}, the f_0 estimation is less precise but has a higher resistance to noise, whereas for higher n_{H} the f_0 estimation is more precise though the resistance to noise is lower.
5. Argmax - denotes an operation which chooses the highest pitch salinity as a maximum of the (a, f_0) plane, giving the most likely values of f_0 and a.
6. Harmonic Spectrum - is the output of the harmonic transform for the estimated fundamental frequency change a.

Using the estimated values a and f_0 we can compute the harmonic parameters of the fundamental frequency and its harmonics using (1.55) and (1.56). The harmonic part of the analyzed segment can then be constructed using (1.54).

To show a typical output of the presented algorithm, it has been run on a signal *micf01sa02* with parameters $M = 511$, $NFFT = 511$, overlap = 5 ms, $f_{\mathrm{s}} = 8$ kHz, $n_{\mathrm{H}} = 4$, for $f_0 \in < 80; 350 >$ and $a \in < -0.3; 0.3 >$ without oversampling. Fig. 3.13 shows the pitch salience where the fundamental frequency contour can be seen as peak values. The maximum values of pitch salience for each segment are shown in Fig. 3.14. It should be noted many of the values are indeed maximum values though they represent a non-voiced segment, which does not have any fundamental. Spectrogram constructed from the outputs of Harmonic Transform is shown in Fig. 3.15 and a STFT spectrogram is shown in Fig. 3.16 for reference. It is evident the Harmonic Transform based spectrogram has sharper peaks without spectral smearing where a harmonic structure is present in the signal, specifically in the higher frequencies. Fig. 3.17 represents fundamental frequency change a which is one of the input parameters of the HT. Only values corresponding to voiced segments are meaningful. Even though there are several octave errors in estimating the fundamental frequency as can be seen at time between 2.3 s and 2.5 s in Fig. 3.14, the fundamental frequency slope is correctly estimated as there are sharp continuous peaks at the same time interval in Fig. 3.15.

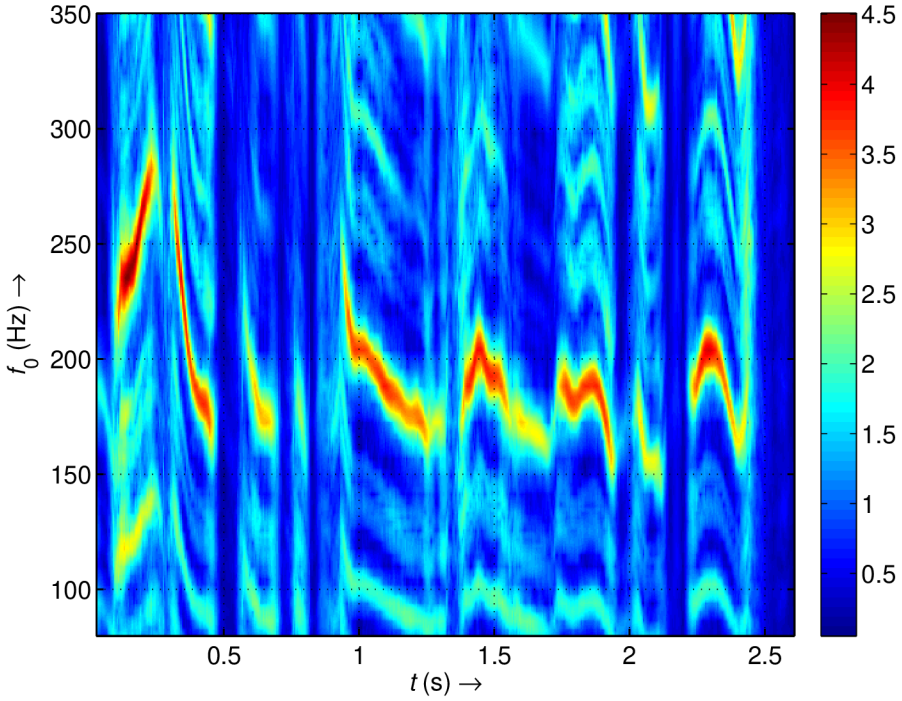


Fig. 3.13: Pitch salience shown on gathered log-spectrum of *micf01sa02* signal for a range of f_0's in time showing the most likely f_0 trajectory as peak values.

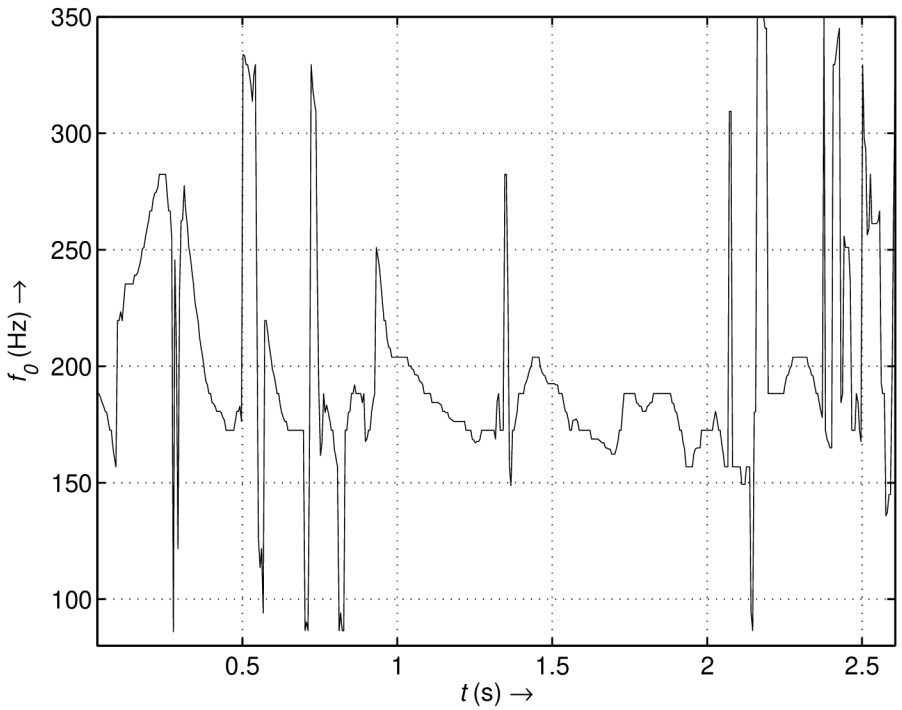


Fig. 3.14: Fundamental frequency of the speech segment *micf01sa02* obtained from maximum values of the gathered log-spectrum.

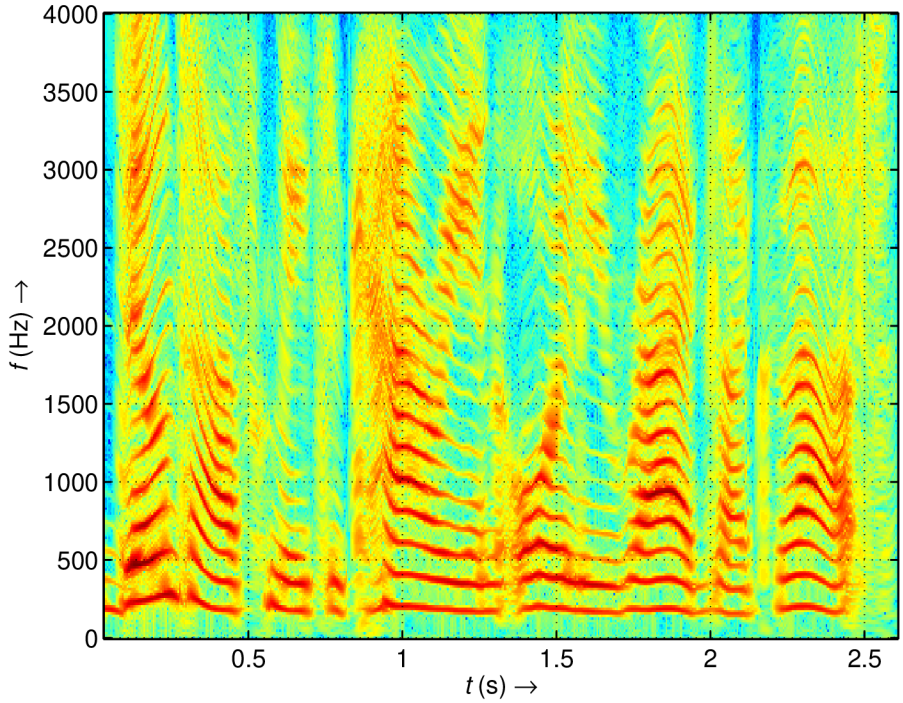


Fig. 3.15: Spectrogram of the *micf01sa02* signal obtained using Fast Harmonic Transform with gathered log-spectrum as the f_0 change estimation algorithm.

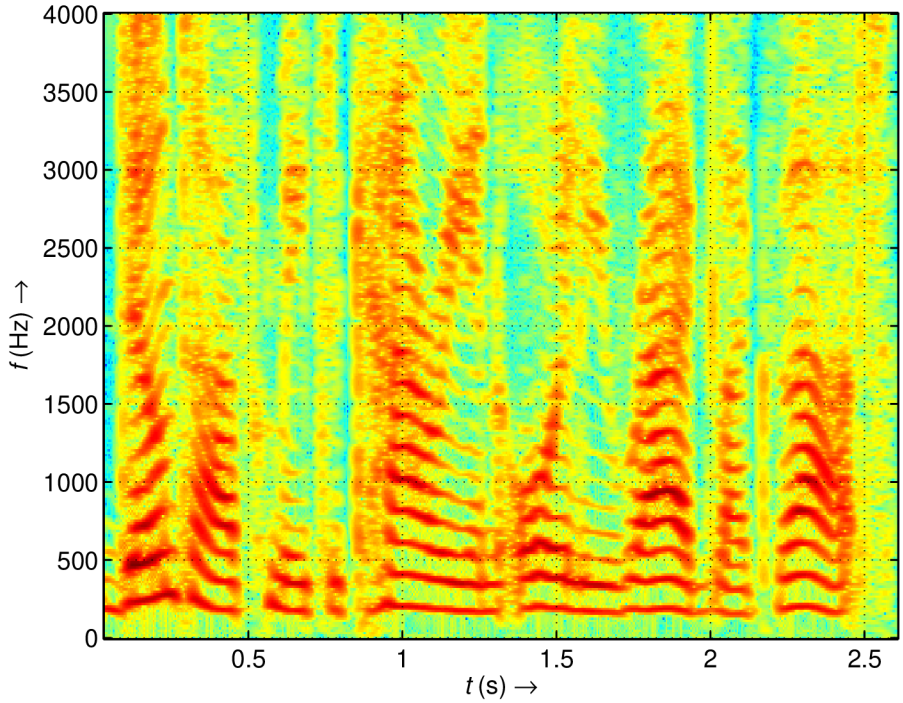


Fig. 3.16: Spectrogram of the signal *micf01sa02* obtained using STFT.

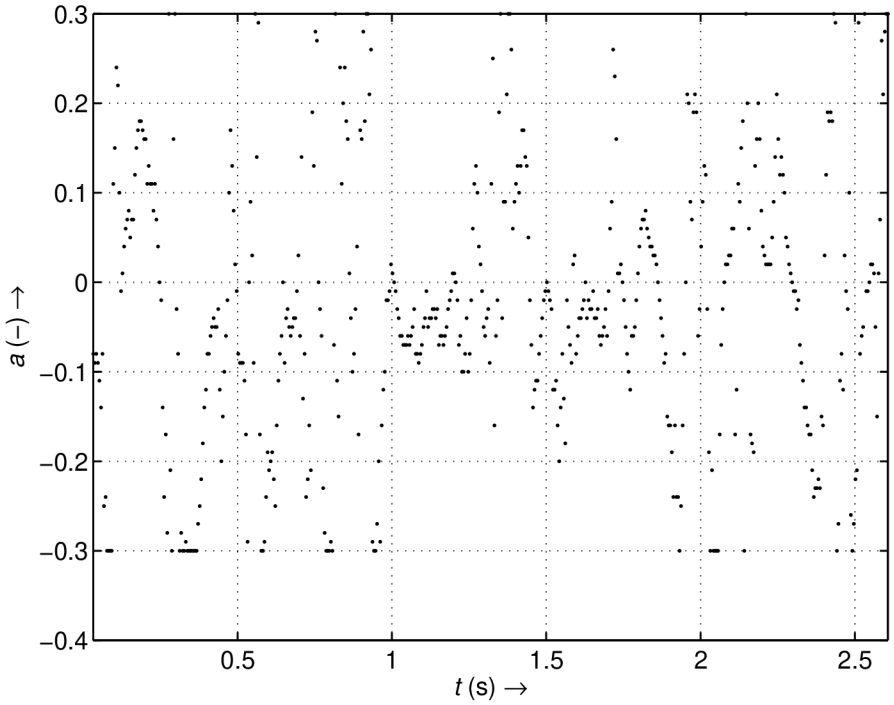


Fig. 3.17: Fundamental frequency slope of the signal *micf01sa02* obtained from each segment using the gathered log-spectrum based method.

3.4 Estimation of Fundamental Frequency Change Using Analysis-by-Synthesis Approach

In this approach we will use the (a, f_0) plane to estimate the fundamental frequency as in 3.3 but with harmonic-to-noise ratio instead of pitch salinity. This approach assumes analysis of signals which are composed of a fundamental frequency and its harmonics. We will try to estimate harmonic parameters of each harmonic of such signal using (1.55), where the hypothetical number of harmonics n_{H}, range of fundamental frequencies f_0 and range of fundamental frequency changes a is based on previous knowledge of the nature of the analyzed signal. After the harmonic parameters have been estimated, they are used to construct the harmonic part of the analyzed signal which is then subtracted from the analyzed signal to get the residual signal. Then harmonic-to-noise ratio is computed from the harmonic and residual signal for all values of a and f_0 which are then assembled on the (a, f_0) plane as can be seen in Fig. 3.18. For a and f_0 that match the analyzed signal there will be a peak in the (a, f_0) plane and these values are evaluated as the final values.

FFT cannot be used to compute (1.55) though its computational complexity is $\mathcal{O}(kN)$, where k is the number of harmonic components and N is length of the transformation. Computational requirements can be kept reasonable through suit-

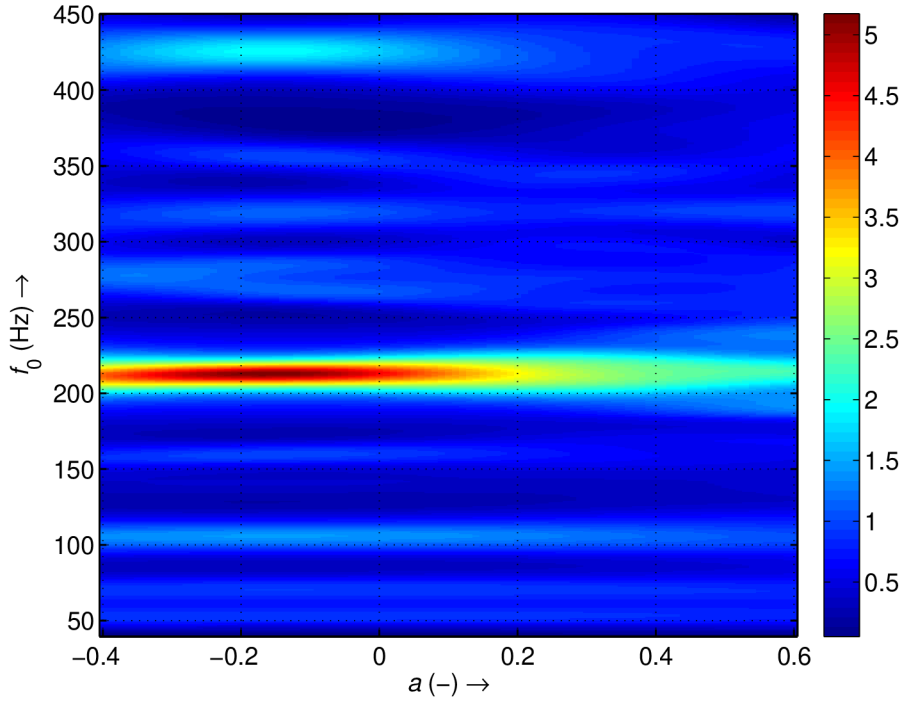


Fig. 3.18: HNR (dB) of speech signal *micf01sa02* estimated using at $t = 359.6$ ms, $M = 255$, $NFFT = 255$, $f_s = 8000$ Hz, $n_H = 4$, with estimated $f_0 = 212.4$ Hz, $a = -0.17$, and HNR = 5.16 dB at the peak.

able choice of input parameters.

Algorithm outline

Block diagram of the algorithm can be seen in Fig. 3.19. Block diagram description is as follows:

1. Segmentation - The audio signal is divided into segments for segment-wise processing. Length of the segments depends on the analyzed signal, specifically on its fundamental frequency change. It is necessary to adjust the length of the segment so that the fundamental change throughout the segment is approximately linear. This will make sure the harmonic transform will give a fine representation of the frequency-modulated harmonic content of the signal.
2. Windowing - is usually used to suppress spectral leakage on the borders of the segment. Hann and Hamming windows have given good results.
3. Harmonic transform aligned at f_0 - is performed by the DHT aligned with the fundamental frequency f_0 using (1.55). The transformation is performed several times for a range of fundamental frequencies and a range of fundamental frequency slopes. The fundamental frequency and fundamental frequency slope ranges are chosen so they are sensible to the analyzed data (e.g.

$f_0 \in < 80; 450 >$ Hz for adult speech). The transform is performed over fundamental frequency and a selected number of harmonics. The output consists of harmonic coefficients of each harmonic.

4. Sinusoidal generator - generates the harmonic signal from its harmonic parameters. Amplitudes and phases of the harmonics can be computed directly from the $S(k)$ coefficients using (1.56) and the periodic component of the signal is computed using (1.54). The noise signal $\hat{r}(n)$, required for the next step is computed by subtracting the reconstructed harmonic component $\hat{h}(n)$ from the input signal $s(n)$ as in (1.57).
5. Argmax(HNR) - denotes the maximum value on the (a, f_0) plane which is constructed using harmonic-to-noise ratios (HNR) where the harmonic signal is constructed using values from the previous step and the residual signal is computed using (3.19). This value should represent the best fit of fundamental frequency f_0 and its change a for the analyzed segment. If the fundamental frequency f_0 of the analyzed signal is absent, the fundamental frequency change a can still be estimated accurately. The succeeding steps are performed with the a and f_0 parameters found at the peak of the (a, f_0) plane.
6. Harmonic transform - performs FHT with a and f_0 parameters obtained from the previous step.
7. Harmonic parameters - outputs the harmonic parameters of the estimated harmonic signal with a and f_0 parameters from step 5.
8. Harmonic spectrum - outputs the amplitude-frequency and phase-frequency spectrum of the transform from step 6.

Harmonic-to-noise ratio (HNR) is a ratio between the energy of the harmonic component of a signal and its noise component. It can be computed as

$$\mathrm{HNR} = 10 \log \frac{E_{\hat{\mathrm{h}}}}{E_{\mathrm{n}}}, \tag{3.19}$$

where $E_{\hat{\mathrm{h}}}$ is energy of synthesized harmonic component $\hat{h}(n)$ and E_{n} is energy of the noise-like component $\hat{r}(n)$. The noise-like component or the residual signal $\hat{r}(n)$ is defined as the difference between original signal and the synthesized harmonic component.

The algorithm has been tested on a signal *micf01sa02* with the same parameters as in case of the method presented in Section 3.3: $M = 511$, $NFFT = 511$, overlap = 5 ms, $f_{\mathrm{s}} = 8$ kHz, $n_{\mathrm{H}} = 4$, for $f_0 \in < 80; 350 >$ and $a \in < -0.3; 0.3 >$ without oversampling. Fig. 3.20 shows HNR of each analyzed segment with the fundamental frequency slope selected for that segment. Maximum values of HNR of each analyzed segment form the fundamental frequency in Fig. 3.21. Compared to fundamental frequency obtained using gathered log-spectrum in Fig. 3.14 we can see

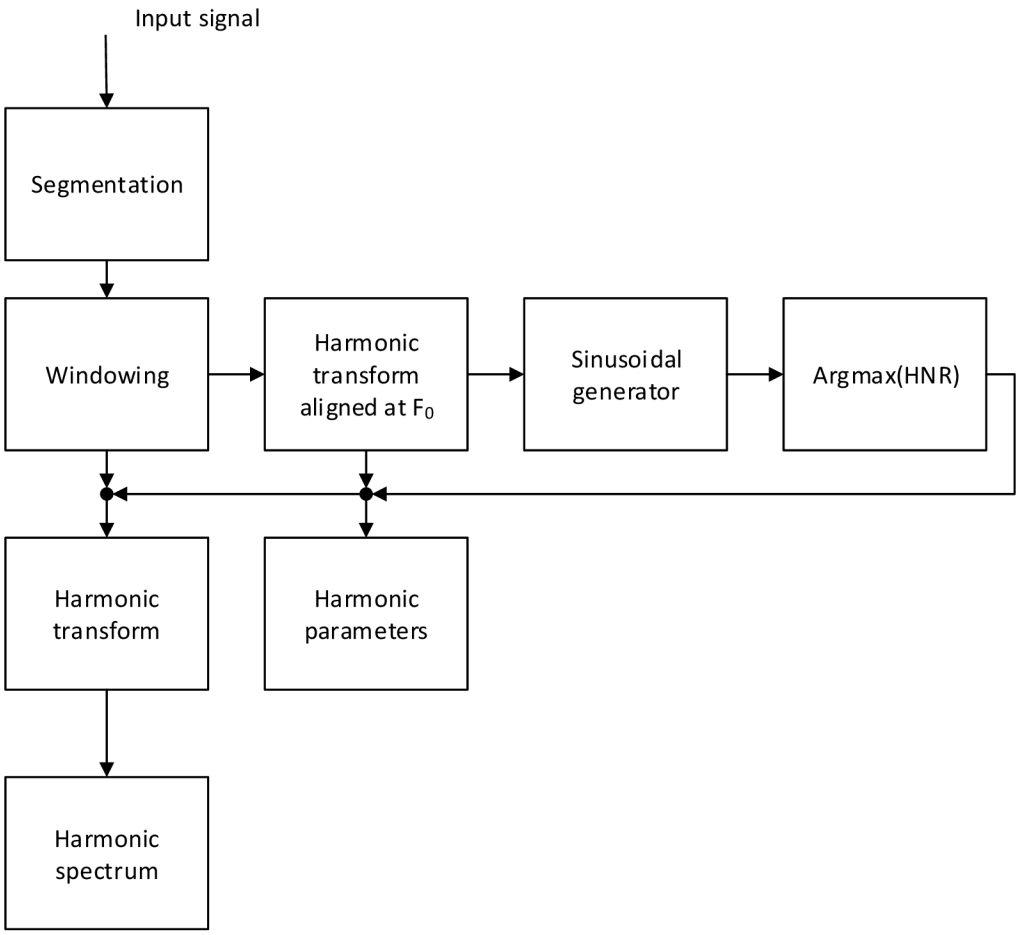


Fig. 3.19: Block diagram of Fast Harmonic Transform algorithm using harmonic parameters for f_0 change estimation.

the former has a smoother contour while the latter is usually much faster to compute. From Fig. 3.22 we can see the harmonic spectrogram provides much sharper peaks compared to the STFT spectrogram in Fig. 3.16 and it is very similar to the harmonic spectrogram obtained using gathered log-spectrum as can be seen in Fig. 3.15. There are also parts where the harmonic spectrogram provides doubtful results occuring usually at transients e.g. at time intervals (0.8 s;1 s) and (2.3 s; 2.5 s). Fig. 3.23 shows the fundamental frequency slope a selected for each segment from maximum values of the (a, f_0) plane for each segment. It is only meaningful for voiced segments.

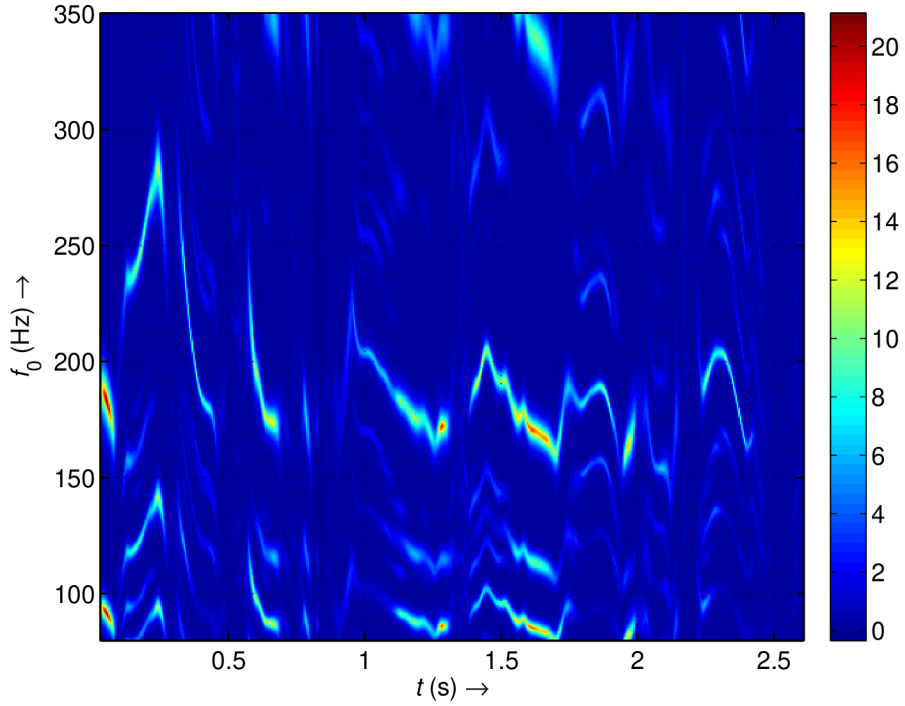


Fig. 3.20: HNR (dB) of the synthesized harmonic component from the signal *micf01sa02* with harmonic parameters extracted using the analysis-by-synthesis method.

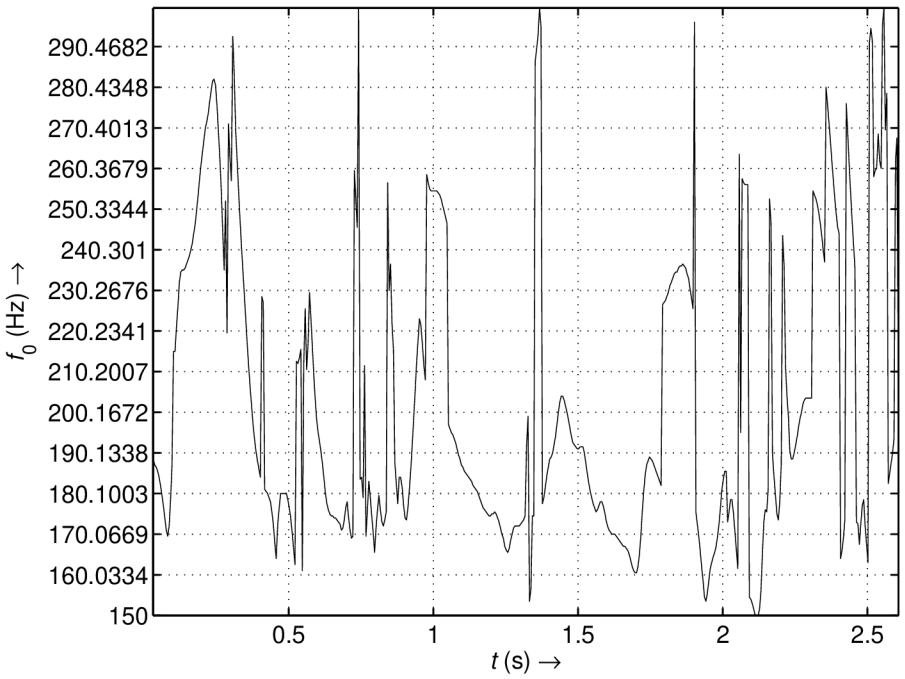


Fig. 3.21: Fundamental frequency of the signal *micf01sa02* extracted using the the analysis-by-synthesis method.

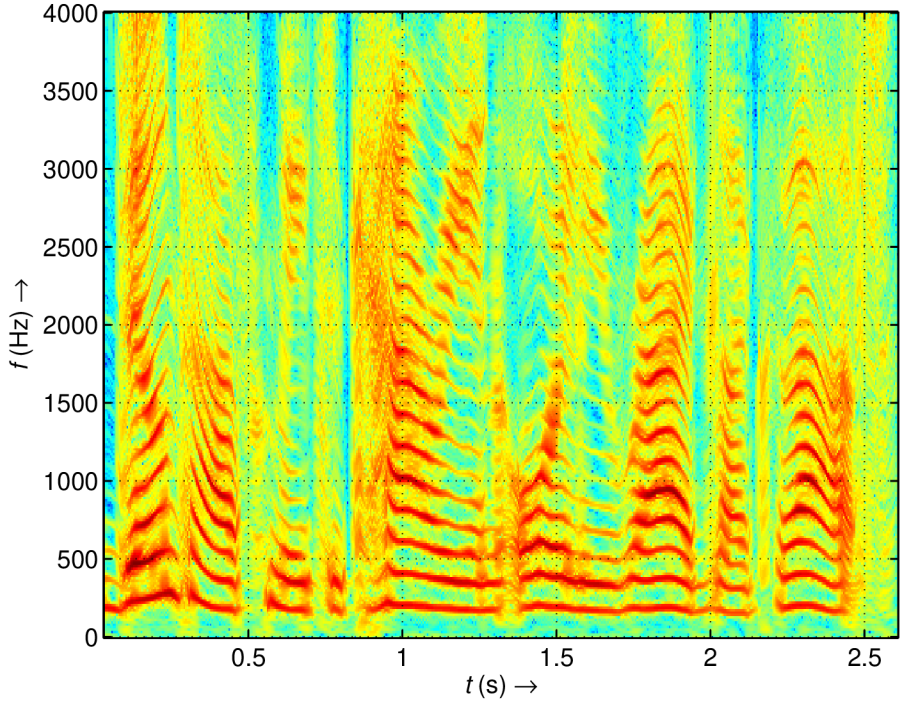


Fig. 3.22: Spectrogram of the signal *micf01sa02* obtained using the analysis-by-synthesis method.

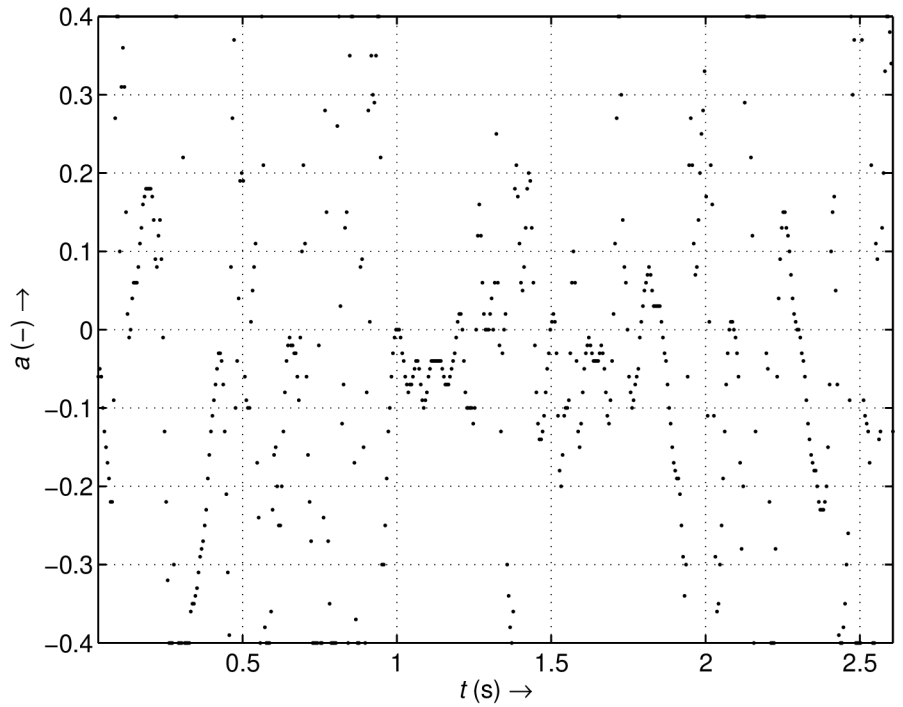


Fig. 3.23: Fundamental frequency slope of segments of the signal *micf01sa02* extracted using the analysis-by-synthesis method.

Tab. 3.2: Computational steps of the Fast Harmonic Transform algorithm

Operation	Description	Operations
Normalization	$z_a(n) = \frac{x(n)w(n)}{\phi'_a(n)}$	$2N$
Warped index	$\tau_a = \psi_a(n)$	N
Resampling	$s_a(n) = z_a(\tau_a)$	
	Hermite spline interpolation [13]	$4N$
	linear interpolation [13]	$2N$
DFT	$S(k,a) = \mathrm{DFT}(s_a(n))$	$N \log N$

3.5 Computational Load

The computational load of the fast algorithm can be enumerated using the number of operations required for analysis of one segment of length N. The algorithm can be divided into: normalization, warped index computation, resampling, and FFT. In normalization stage, the input signal $x(n)$ is multiplied by the window function which has been divided by the scaling factor $\phi'(n)$. Warped index computation estimates time instants of the signal time-warped according to the warping function $\psi_a(n)$. The time-warped discrete-time signal $s_a(n)$ is obtained using interpolation from the normalized input signal $z_a(n)$. Finally the output harmonic spectrum $S(k,a)$ is computed using FFT, assuming the length of analyzed segment N is power of two. Each step of the algorithm is summarized in Table 3.2 together with the number of operations involved for every length of analyzed segment. The resulting computational load is $N(\log N + 7)$ for the Hermite spline interpolation and $N(\log N + 5)$ for the linear interpolation.

Computational load of the gathered log-spectrum computation depends on the number of analyzed fundamental frequencies, the range of fundamental frequency change, length of the FFT, choice of interpolation method, and number of presumed harmonics in the signal. Analyzing the computational load of a fundamental frequency estimation algorithm is out of scope of this thesis.

3.6 Effect of Aliasing

Spectrum of a linear chirp signal *test signal* obtained using Harmonic Transform computed directly from the equation (1.49) is without artifacts (as can be seen in Fig. 3.8), except the usual artifacts caused by windowing. The Fast Harmonic Transform uses interpolation of the input signal which introduces errors, namely, aliasing. To demonstrate the effect of aliasing we have used *test signal* which is a

linear chirp with 17 harmonics. Its STFT spectrum can be seen in Fig. 3.25 where the spectral smearing can be seen as wider peaks with lower magnitude which blend together, specifically towards higher frequencies.

Time warping performed using (3.7) maps one axis with equidistant intervals to a time-warped axis where the intervals between samples get shorter towards one of the ends of analysis segment as shown in Fig. 3.24. This causes the signal on the warped axis to be undersampled. Aliasing can be seen in Fig. 3.26 as a noise floor which increases with frequency. Fig. 3.27 shows the contribution of each harmonic of the *test signal* to the noise floor caused by aliasing.

One of the straightforward means of diminishing aliasing is oversampling. Oversampling consists of increasing the sampling frequency by adding zeroes to the signal and then filtering the signal by a low-pass filter to eliminate mirroring artifacts. The resulting signal will have a multiple number of samples which in principle reduces the intervals between samples of the signal on the original axis and therefore the time-warped signal is interpolated with higher precision. This also allows us to use a cheaper interpolation method, if advantageous. A case where linear interpolation was used on the *test signal* with 2x and 4x oversampling is shown in Fig. 3.28. In the case of *test signal*, the 4x oversampling is performing close to DHT which can be seen in Fig. 3.26. A more thorough analysis has been performed on signal *micf01sa02* as shown in Tab. 3.1.

3.7 PTDFT and HT

This section addresses the similarity between Pitch Tracking Modified DFT (PTDFT) and Harmonic Transform. The PTDFT is a modified DFT transform for analysis in harmonic domain. It is enumerated by direct computation and its computational complexity is therefore quadratic. It can be shown that the transformation kernel of PTDFT and HT is identical for linear frequency change over the length of analyzed segment. The PTDFT is defined as [23]

$$S_i(k) = \sum_{n}^{N-1} s_i(n) w_i(n) \mathrm{e}^{-\mathrm{j}\frac{2\pi nk}{f_\mathrm{s}}\left(f_0+\frac{\Delta f_0 n}{2N}\right)}, \tag{3.20}$$

where $s_i(n)$ is n-th sample of the i-th frame, f_0 fundamental frequency, Δf_0 fundamental frequency change, $w_i(n)$ time window of i-th frame. It can be seen, that after substitutions from (1.44), the phase of a linear chirp signal can be written as

$$\varphi(n) = \frac{2\pi n}{f_\mathrm{s}}\left(f_0 + \frac{\Delta f_0 n}{2N}\right), \tag{3.21}$$

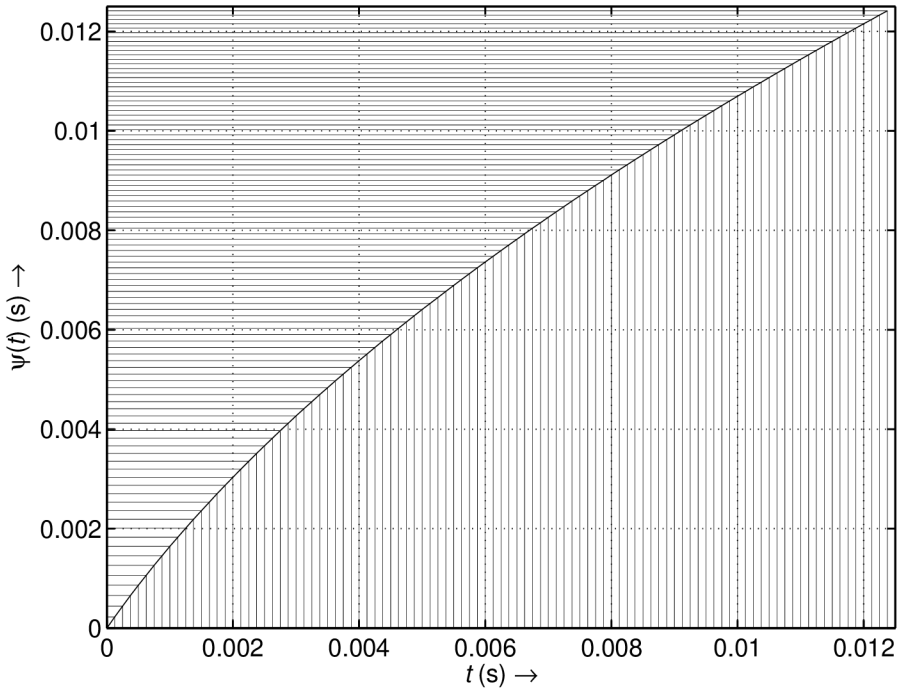


Fig. 3.24: Relationship between the original and warped axis showing the distance between samples gets smaller at the end of segment for $a = 0.9$.

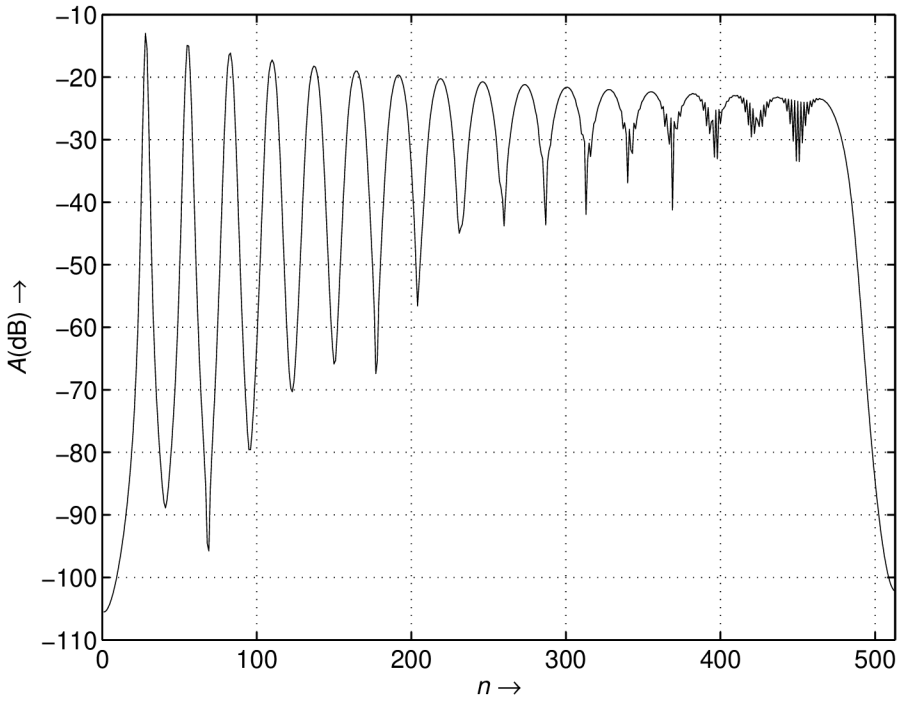


Fig. 3.25: Magnitude spectrum of the *test signal*, a linear chirp with 17 harmonics.

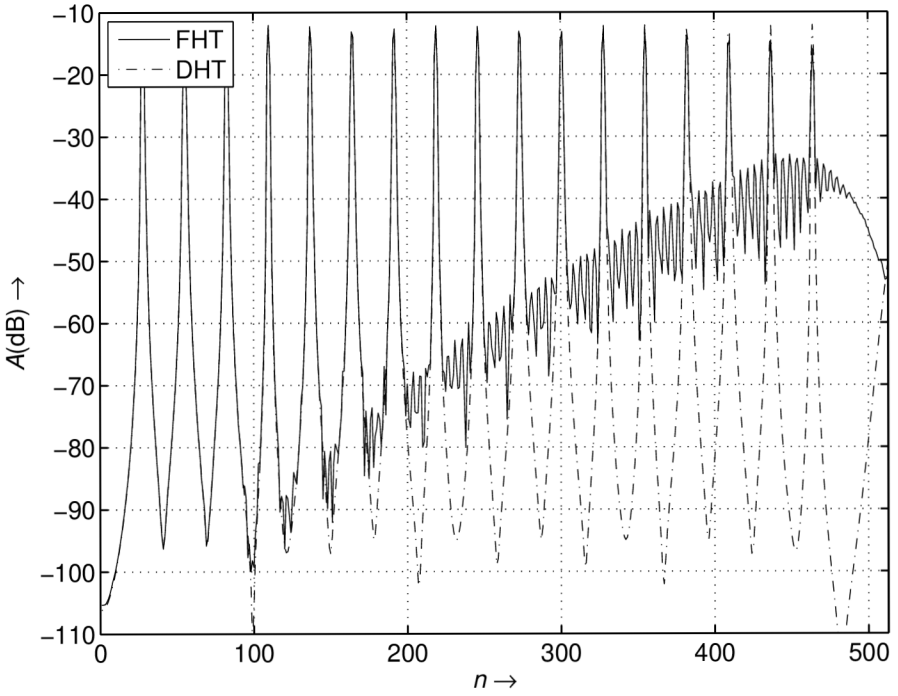


Fig. 3.26: Spectra comparison of linear chirp *test signal* between Fast Harmonic Transform with linear interpolation and aliasing and Discrete Harmonic Transform.

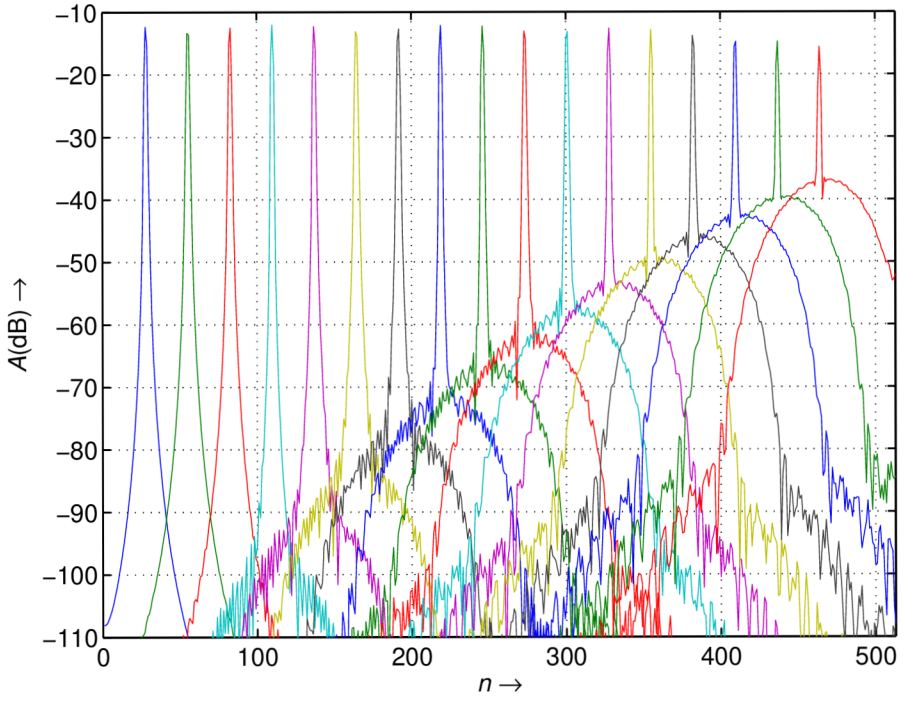


Fig. 3.27: Contribution of each harmonic to aliasing in Fast Harmonic Transform with linear interpolation to the spectrum of linear chrip *test signal*.

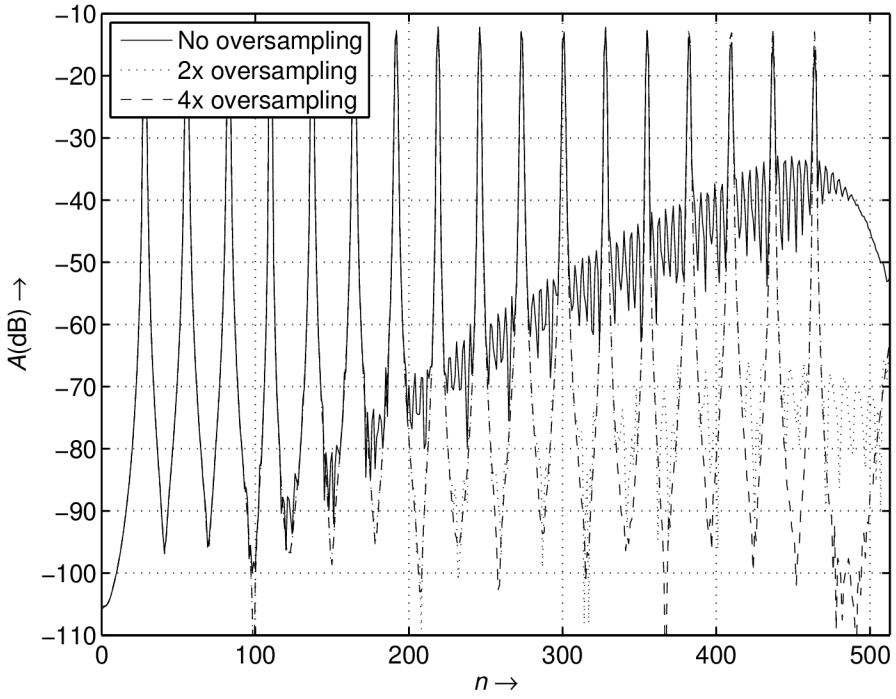


Fig. 3.28: The effect of oversampling on aliasing. Fast Harmonic Transform with linear interpolation was used on *test signal*.

which is the transformation kernel of PTDFT and with substitutions from (1.45),

$$\begin{aligned} \varphi(n) &= \frac{2\pi n}{f_\mathrm{s}} \left(f_c - \frac{a f_c}{2} + \frac{f_c n}{2N} \right), \\ \varphi(n) &= \frac{2\pi n f_c}{f_\mathrm{s}} \left(1 - \frac{a}{2} + \frac{an}{2N} \right), \end{aligned} \tag{3.22}$$

and since $f_c = \frac{N}{f_\mathrm{s}}$, which is shown in transition from (1.43) to (1.44), the result is

$$\varphi(n) = \frac{2\pi n}{N} \alpha(n), \tag{3.23}$$

which is the Harmonic transform kernel from (1.48).

This enables to apply methods presented in this thesis to increase performance of algorithms based on the PTDFT. Simultaneously, many of the improvements and applications of PTDFT such as time-varying Kaiser window design, fundamental frequency estimation based on cross-correllation, or real-time speech coding that have been successfully proven to work with PTDFT, can be applied to HT.

It should be noted, that the PTDFT is a special case of HT for linear frequency change of fundamental frequency. The HT is designed to represent any general form of continuous fundamental frequency change (e.g. quadratic).

3.8 Experiments

The purpose of this chapter is to present use of the algorithm presented in 3.4 on real audio signals. From now on it will be referred to as the ABS (analysis-by-synthesis) algorithm. Since we are analyzing real signals, there is no ground truth for the signal's harmonic parameters at each instant as opposed to analyzing synthesized signals, where the parameters are known and can be directly compared. Therefore we will analyze the signal using the ABS algorithm and use it to extract the fundamental frequency which will be used as input to harmonic parameter estimation. The signal will then be reconstructed using the harmonic parameters when using the knowledge of fundamental frequency slope and without this knowledge. This will produce a synthetic harmonic signal, an estimate of the input signal, with (further referred to as ABS-FM) and without frequency modulation (ABS-S). The ABS-S algorithm is essentially the same algorithm as ABS-FM with $a = 0$. This synthetic harmonic signal will then be subtracted from the input signal, leaving a residual signal. The better the harmonic parameter estimation, the lesser the residual signal energy. By measuring the harmonic-to-noise ratio for different signals with frequency modulation while using the knowledge of fundamental frequency change and without it, we can quantify the increase of harmonic parameter estimation accuracy which we get by using ABS-FM algorithm.

So far, the Harmonic Transform has been used on speech signals which are usually conveniently sampled at 8 kHz. Yet, for many applications higher sampling frequencies are required. Experiments in this section are done on audio signals with sampling frequency 44.1 kHz. This causes two effects that change the efficiency of Harmonic Transform. First, the analysis windows used for audio signals sampled at 44.1 kHz are only two to four times longer, while the sampling frequency is more than five times higher. The fundamental frequency change of the same signal in the analysis window will therefore be smaller. Second, energy of the audio signals is still mostly in the lower frequency region so the improvement in terms of energy will likely be subtle. These two factors are going to diminish the HNR gain of signals reconstructed using the knowledge of fundamental frequency slope and without it.

3.8.1 Viola

This experiment has been performed on a *viola* sound sample. It contains glissando and vibrato, which are both frequency modulation techniques on stringed instruments. Spectrogram of this sound is shown in Fig. 3.29 where we can see most of the signals energy lies bellow 5 kHz though there is more spectral content in higher frequencies. The improvement in resolution of the harmonic spectrogram can be

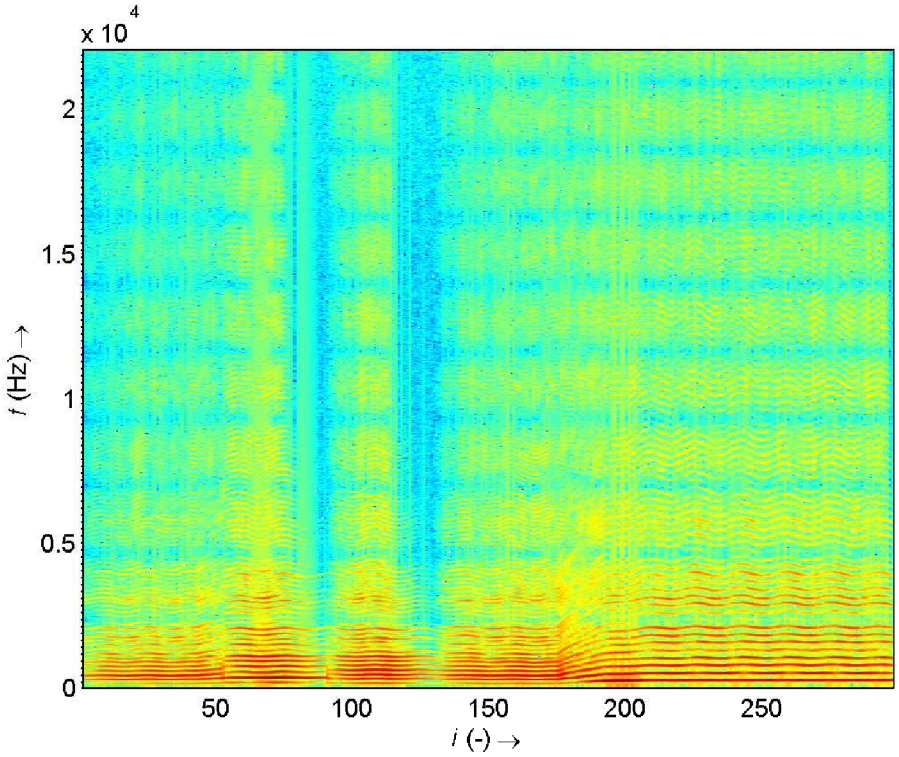


Fig. 3.29: Spectrogram of *viola* sound sample.

seen in Fig. 3.30, specifically between the 180-th and 190-th segment where the glissando takes place. In Fig. 3.29 the spectral lines around 5 kHz are considerably smeared while in Fig. 3.30 they follow the change of the fundamental frequency which can be seen in Fig. 3.31. Fundamental frequency change a shown in Fig. 3.32 has the shape of a fundamental frequency differential. Fig. 3.33 shows the HNR of the reconstructed harmonic part over the residual signal. The sharp notches in HNR are due to poor spectral content caused by string damping when using the glissando technique. And it can be seen from Fig. 3.34 the highest increase in HNR of the ABS-FM is at time intervals where glissando and vibrato (i.e. intervals with the highest frequency modulation) takes place.

3.8.2 Artificial vibrato

In this experiment we would like to apply frequency modulation on a harmonic signal with known and nearly stationary fundamental frequency to compare the ability to estimate harmonic parameters from a signal in our system from section 3.4 when using ABS-FM and ABS-S. The selected harmonic signal is a vocal excerpt. The frequency modulation is created using a vibrato audio effect.

Modulation frequency used in this experiment is 6 Hz, which is well within the

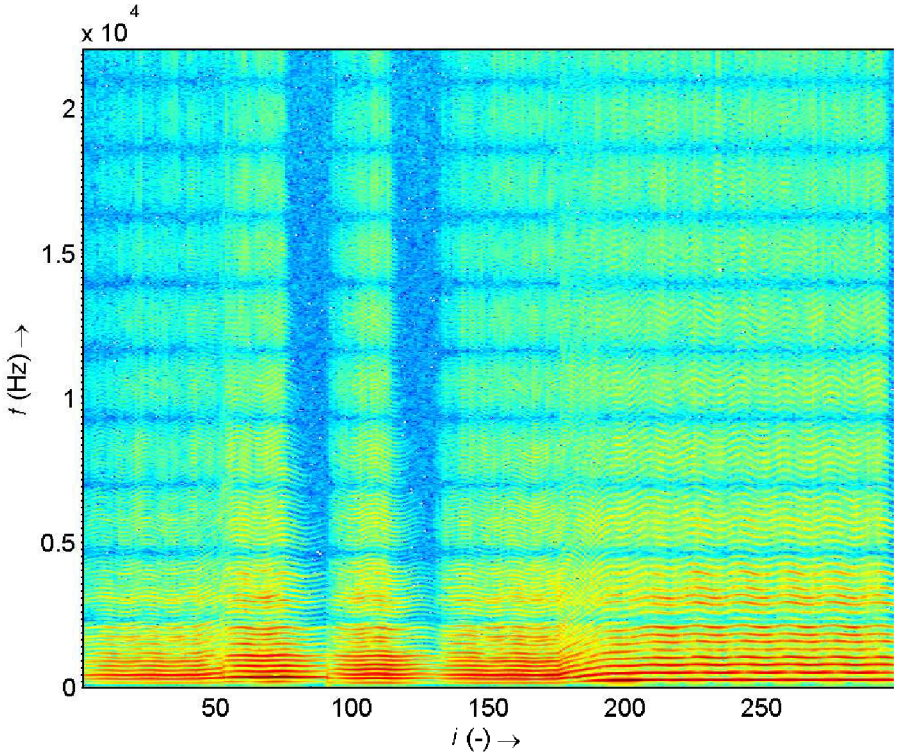


Fig. 3.30: Harmonic spectrogram of *viola* sound sample.

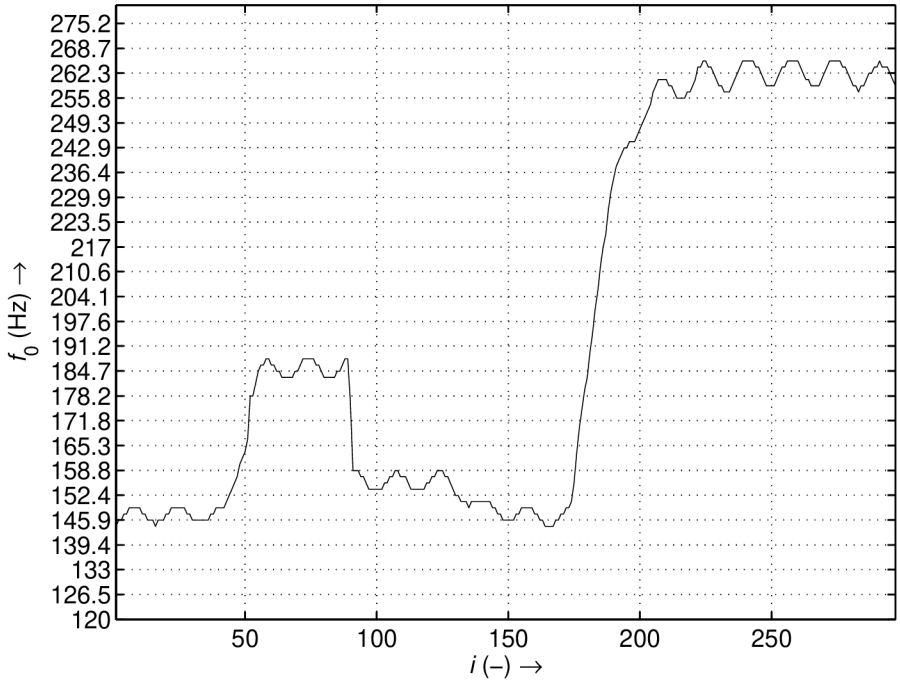


Fig. 3.31: Fundamental frequency of *viola* sound sample.

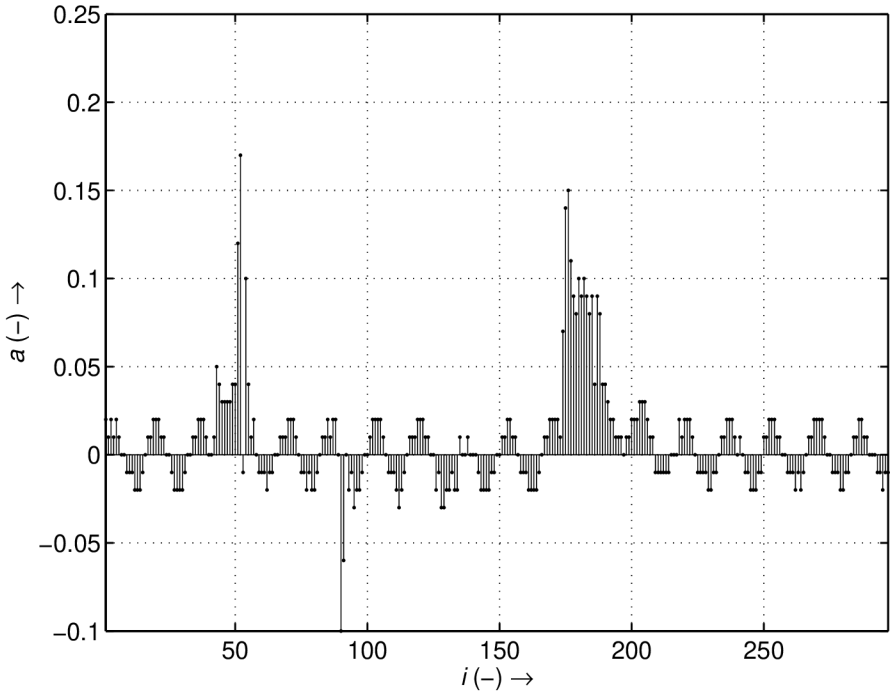


Fig. 3.32: Fundamental frequency change of *viola* sound sample.

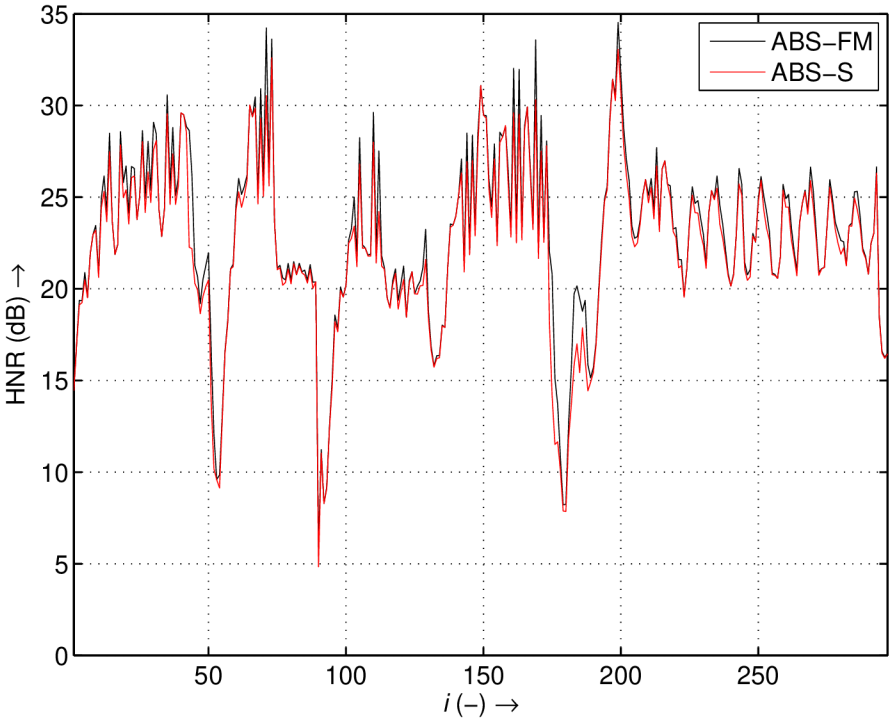


Fig. 3.33: HNR of *viola* sound sample for ABS-FM and ABS-S.

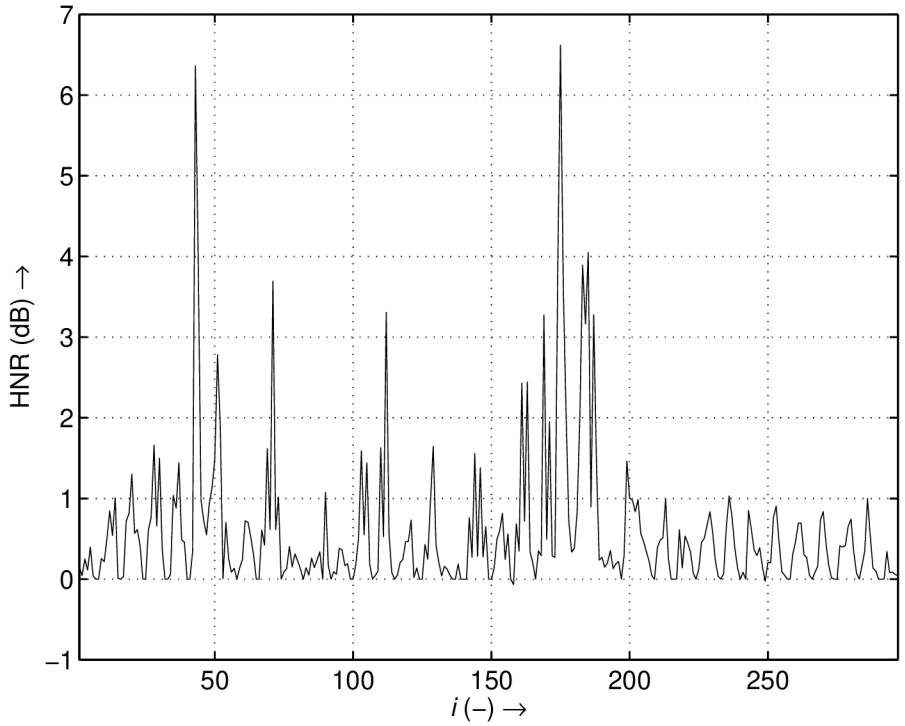


Fig. 3.34: Increase of HNR when using ABS-FM over ABS-S on sound sample *viola*.

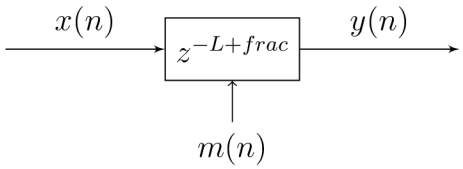


Fig. 3.35: Phase modulation by delay line modulation [85].

range of what the combination of vocalis muscle and diaphragm allows and it is also within the range of modulation frequencies for phase modulation algorithms in terms of musicality [84].

The vibrato used in this experiment is a simple delay line modulation based phase modulator [85]. This technique periodically changes the frequency of the fundamental and its harmonics. While it is easy to implement, it has some disadvantages. If the signal has spectral content close to half of the fundamental frequency, it may cause aliasing after applying the vibrato. It also changes the frequency of harmonic components without regard for spectral envelope of the instrument or vocal tract. Accordingly, the resulting modulated signal may not sound natural. This, however, should not interfere with this experiment.

As stated above, the vibrato used in this experiment is implemented using phase

modulation (see Fig. 3.35 for block diagram) by a modulation signal $m(n)$ and

$$y(n) = x(n - m(n)), \tag{3.24}$$

where $y(n)$ is the vibrato output and $m(n)$ is a continuous variable, which changes for every signal sample [85]. It is therefore decomposed into integer and fractional part [86] where the integer part is implemented using unit delays and the fractional part is implemented using interpolation. For sinusoidal modulation, the modulation signal is defined as

$$m(n) = L + \text{DEPTH} \cdot \sin(\omega_{\text{M}} nT), \tag{3.25}$$

where DEPTH is modulation depth in samples, ω_{M} is angular modulation frequency, L is the number of unit delays, and T is the sampling period. The resulting fundamental freuqency is a product of the fundamental frequency and fundamental frequency ratio

$$\beta(n) = \frac{\omega_{\text{I}}}{\omega} = 1 - \text{DEPTH} \cdot \omega_{\text{M}} T \cos(\omega_{\text{M}} nT), \tag{3.26}$$

where ω_{I} is instantaneous phase, where $\beta(n)$ is also the resampling factor of the fractional part of the modulation signal.

If we now apply the vibrato on a harmonic signal with quasi-stationary fundamental frequency, we can predict the expected fundamental frequency at each moment using (3.26) and compare it with the output of our algorithm. Spectrogram of the *salvation* sound sample without modulation is shown in Fig. 3.36. It is a decaying vocal sample of average fundamental frequency 392 Hz. When using depth of modulation 1 ms, the resulting maximal and minimal frequency ratio is 1.04 and 1/1.04 , giving high an low extreme of the vibrato which is at 407.9 Hz and 376.8 Hz respectively. This can also be observed from Fig. 3.37 which shows the computed vibrato as the predicted sinusoid and the estimated sinusoid shows the estimated fundamental frequency from the ABS-FM algorithm. Spectrogram of the sound sample with vibrato is shown in Fig. 3.38. The harmonic spectrogram in Fig. 3.39 has more clearly defined peaks with less noise, especially around 1 kHz when compared to the STFT spectrogram. The HNR in Fig. 3.40 is copying the decaying tendency of the sound sample by decreasing in time. Again, the difference between ABS-FM and ABS-S as shown in Fig. 3.41 shows increase in harmonic component separation at intervals whith frequency modulation.

3.8.3 Soprano

This is an analysis of the *soprano* sound sample. It is a sound of a female opera singer singing a vowel /i:/. From spectrogram in Fig. 3.43 it is clear most of the signals energy is concentrated below 5 kHz. The harmonic spectrogram (see Fig. 3.42)

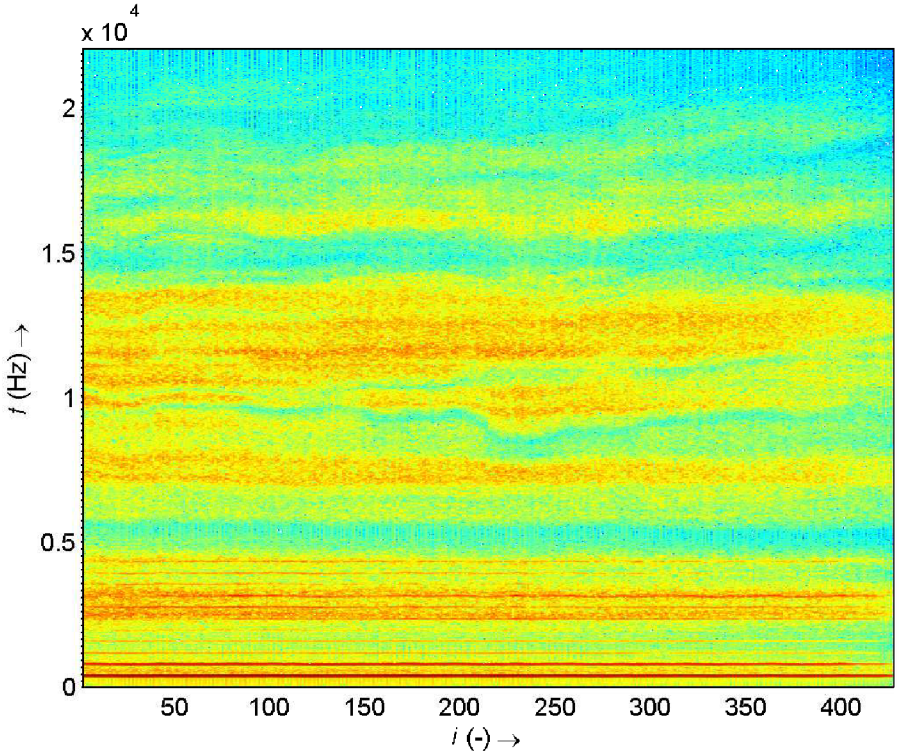


Fig. 3.36: Spectrogram of the sound sample *salvation* without modulation.

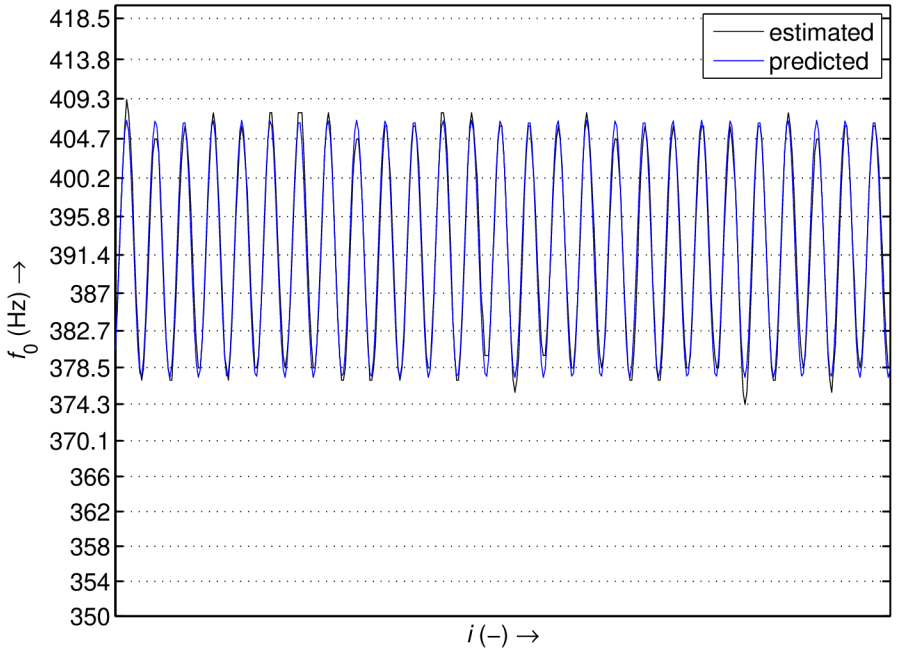


Fig. 3.37: Fundamental frequency of vocal sample *salvation* with artificial vibrato.

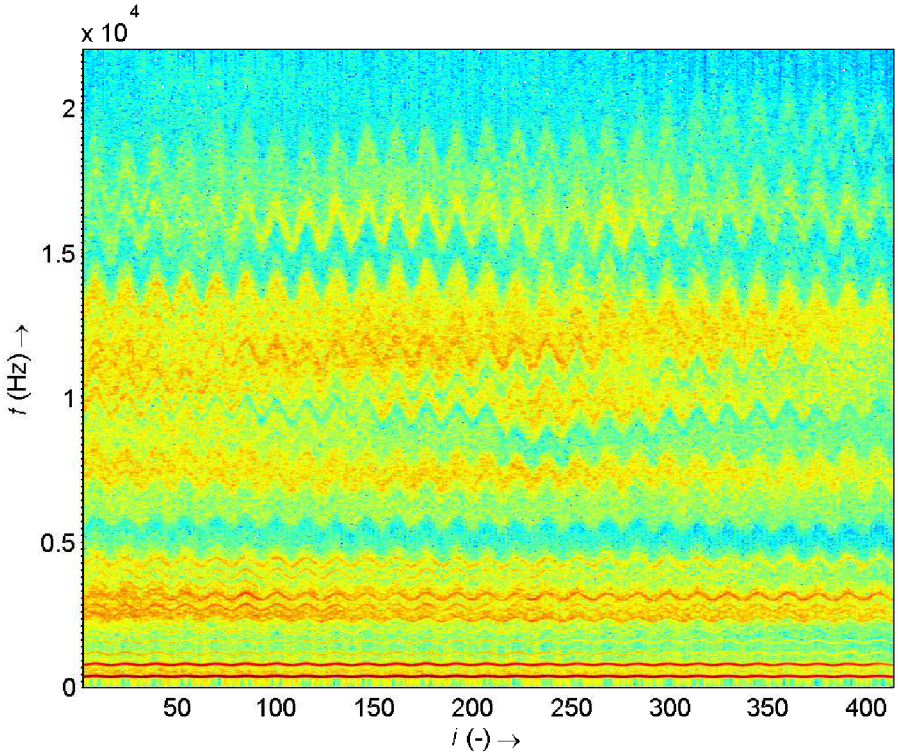


Fig. 3.38: Spectrogram of the sound sample *salvation* with frequency modulation.

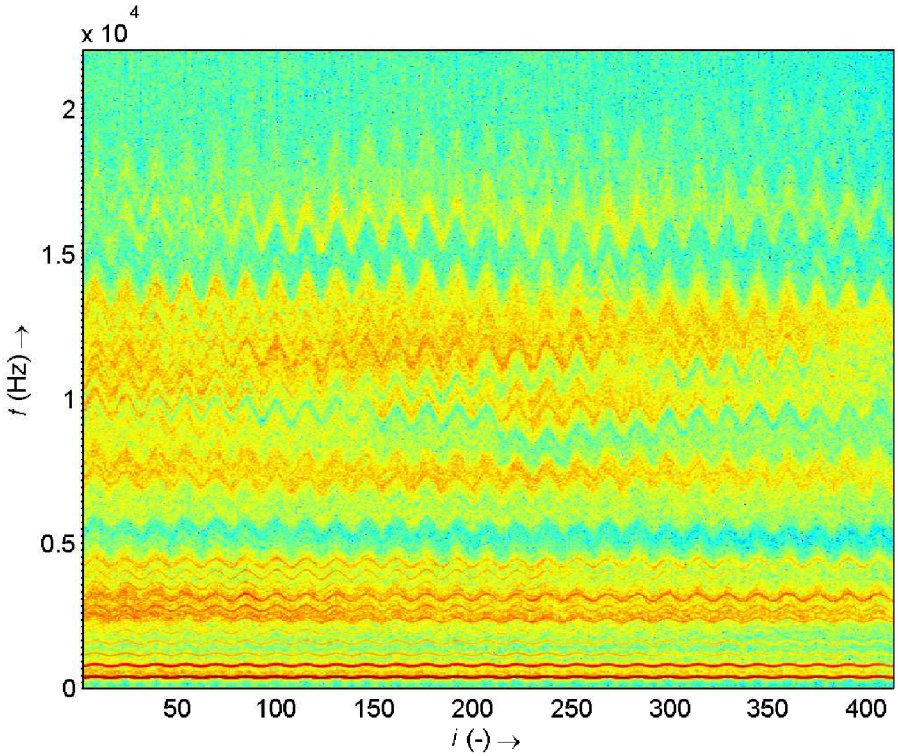


Fig. 3.39: Harmonic spectrogram of the sound sample *salvation* with frequency modulation.

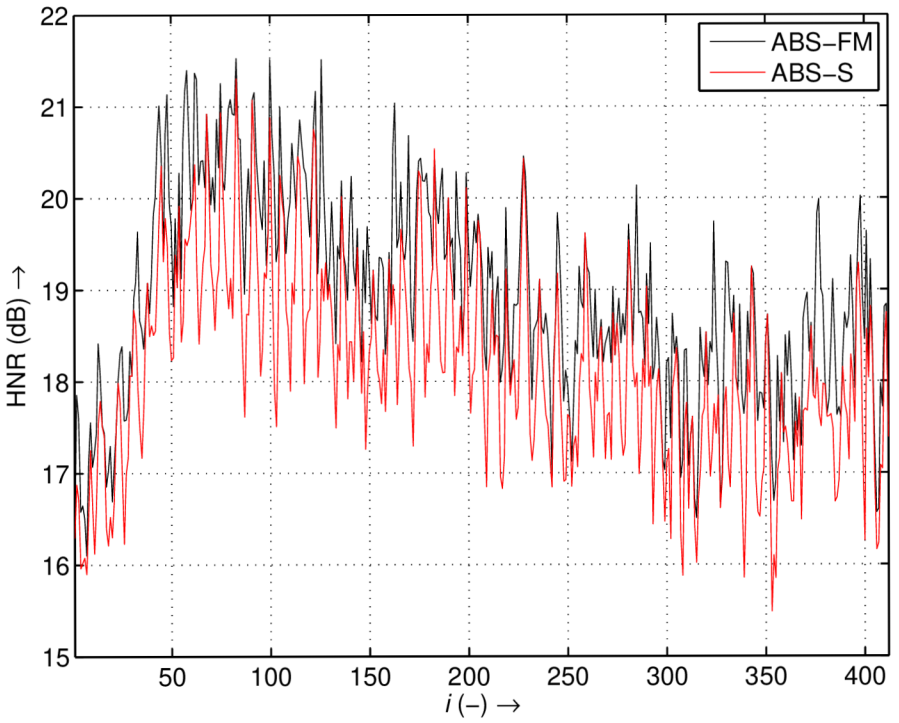


Fig. 3.40: HNR of reconstructed harmonic part of the sound sample *salvation* for ASB-FM and ABS-S.

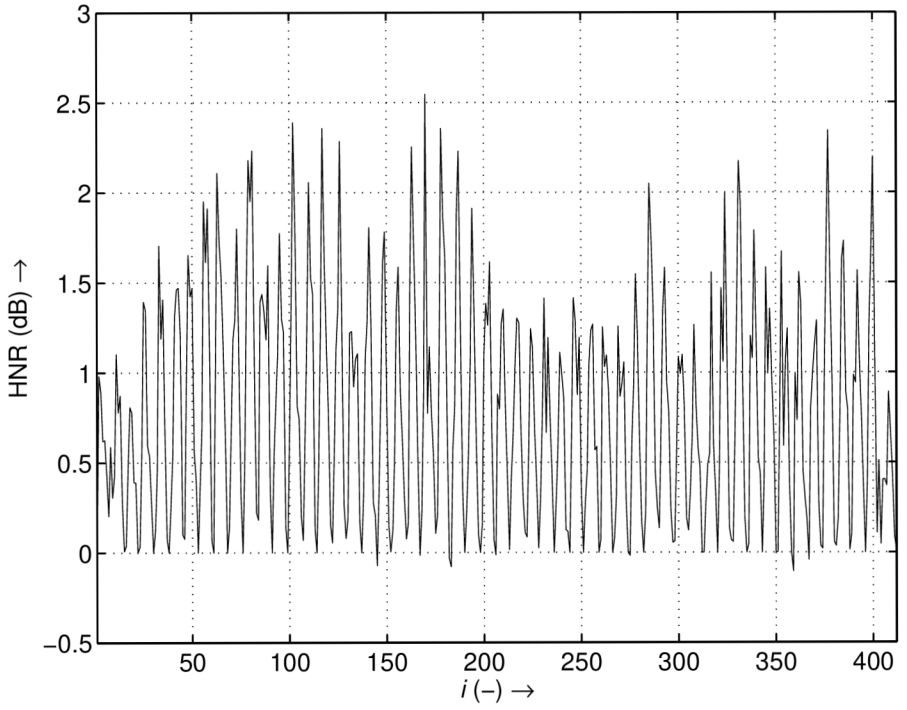


Fig. 3.41: HNR increase of ABS-FM over ABS-S of reconstructed harmonic part of the sound sample *salvation*.

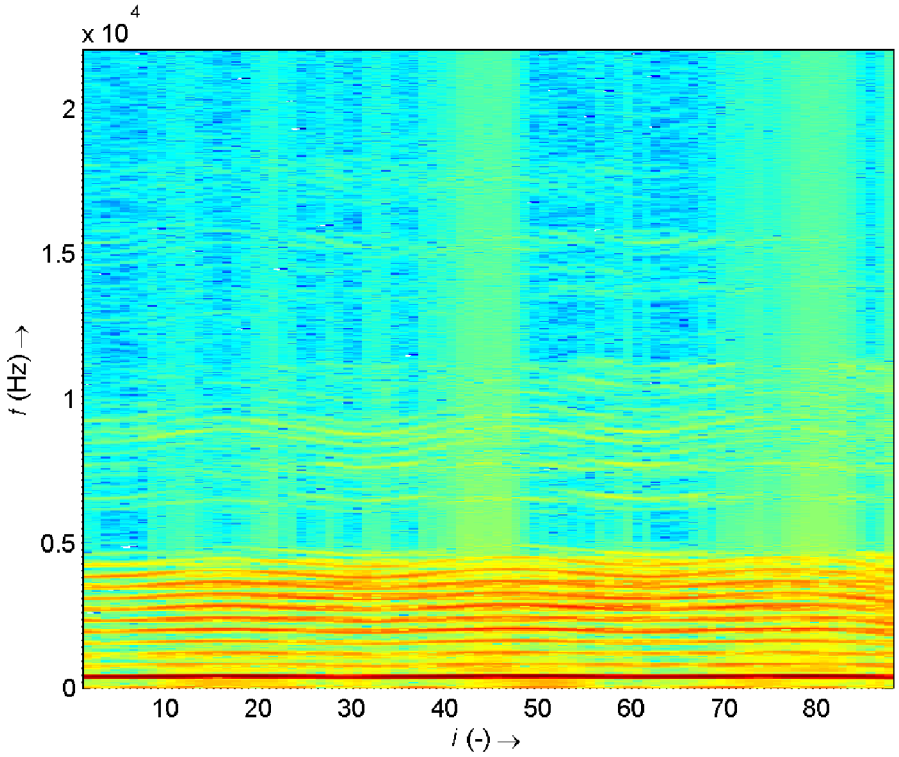


Fig. 3.42: Spectrogram of the sound sample *soprano*.

in this case provides some reduction of noise which can be seen as vertical lines in Fig. 3.43. However at around 44-th and 75-th segment, there is an error in fundamental frequency estimation which can be seen in Fig. 3.44 as a sharp spikes in fundamental frequency at around 44-th and 75-th segment. This error can also be seen in fundamental frequency change estimation in Fig. 3.45 where the fundamental frequency change at 44-th and 75-th segment is too steep for a voice signal. This error of fundamental frequency estimation is most likely due to amplitude modulation. Fig. 3.48 shows the 44-th analyzed segment and its reconstruction. The reconstruction does not represent the amplitude modulation well which decreases the HNR of the reconstructed harmonic part. The HNR is shown in Fig. 3.46 which shows that both algorithms perform similarly well which is confirmed in Fig. 3.47 where the increase of HNR for the ABS-FM algorithm is mostly under 1 dB. This is most likely due to lack of high frequency content in the analyzed signal.

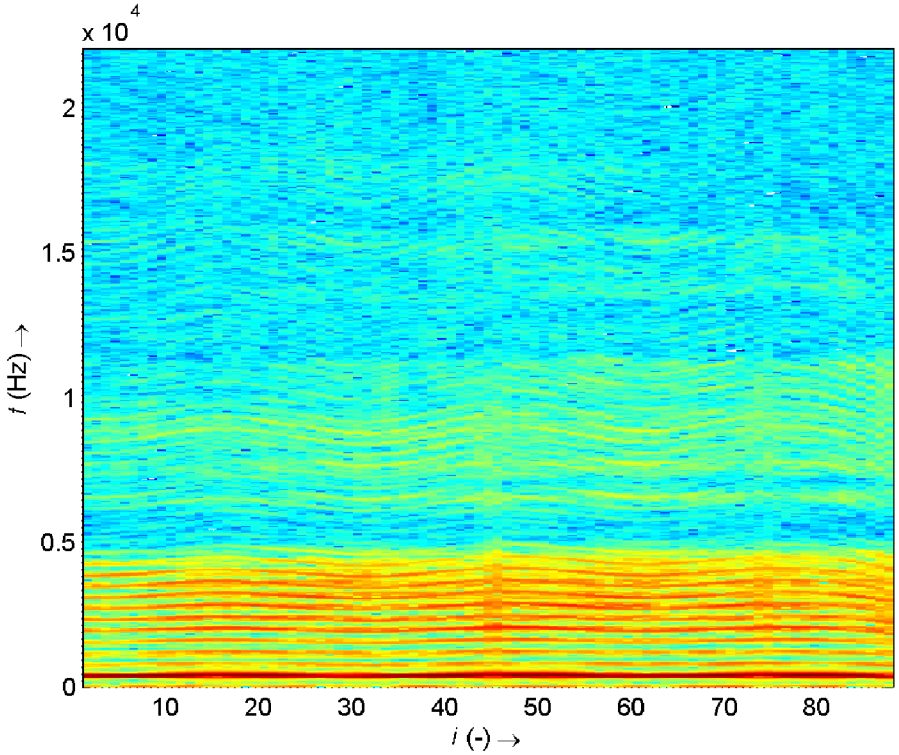


Fig. 3.43: Harmonic spectrogram of the sound sample *soprano*.

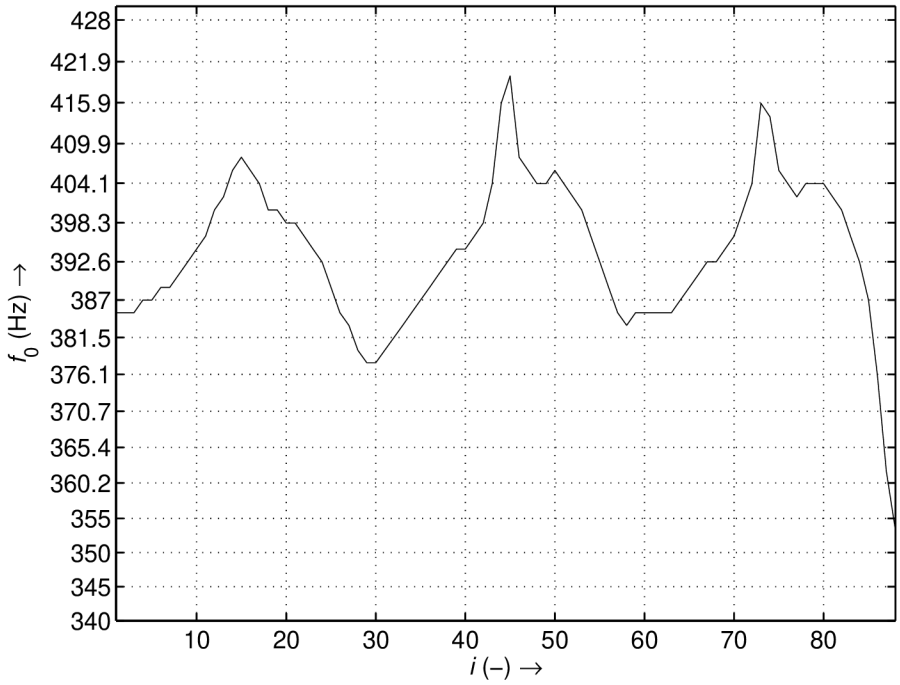


Fig. 3.44: Fundamental frequency of the sound sample *soprano* extracted using ABS-FM.

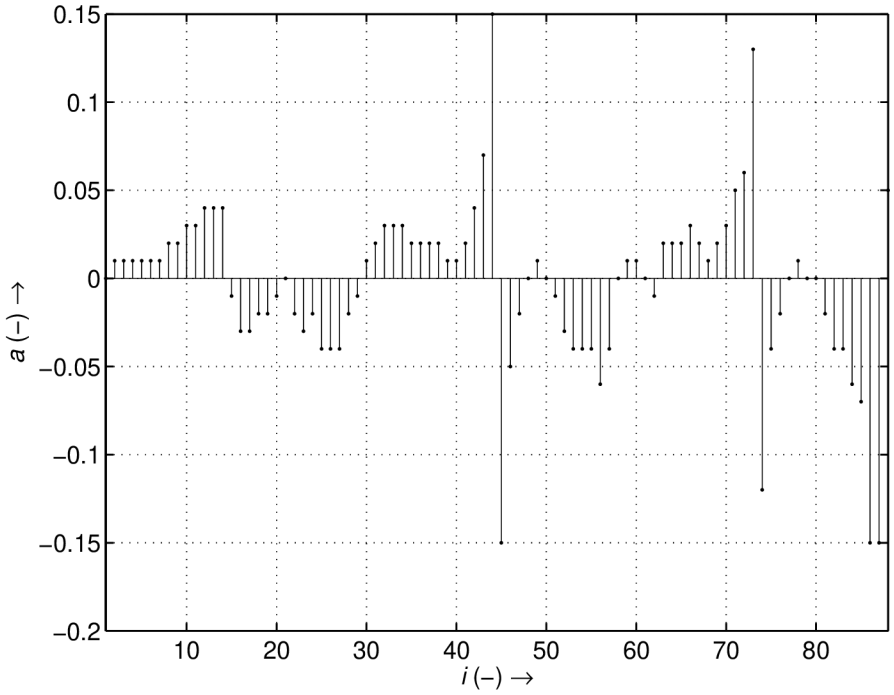


Fig. 3.45: Fundamental frequency slope of the sound sample *soprano* estimated using ABS-FM.

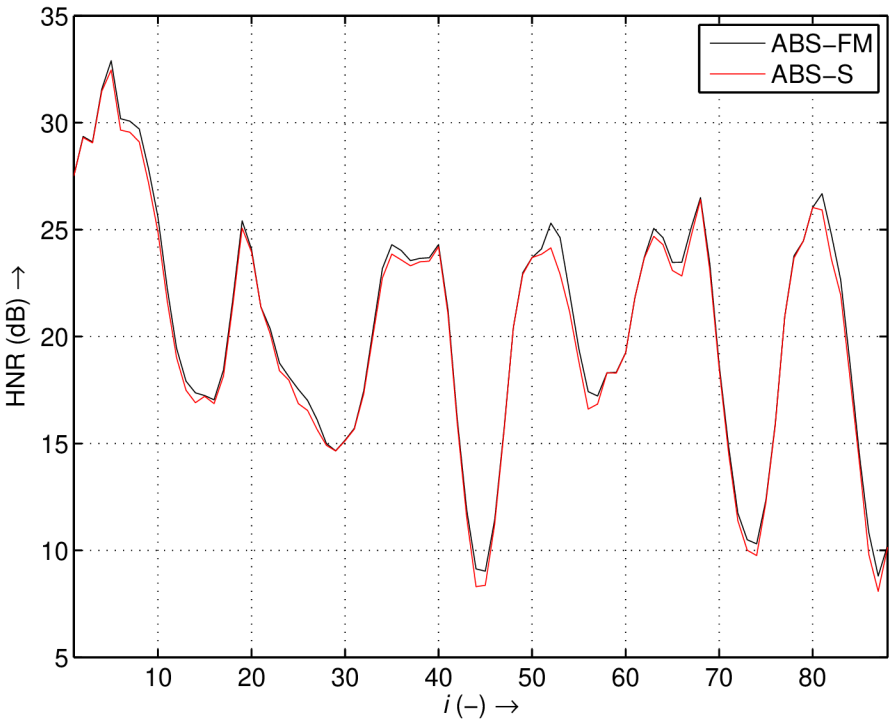


Fig. 3.46: HNR of the reconstructed harmonic part of the sound sample *soprano* for ABS-FM and ABS-S.

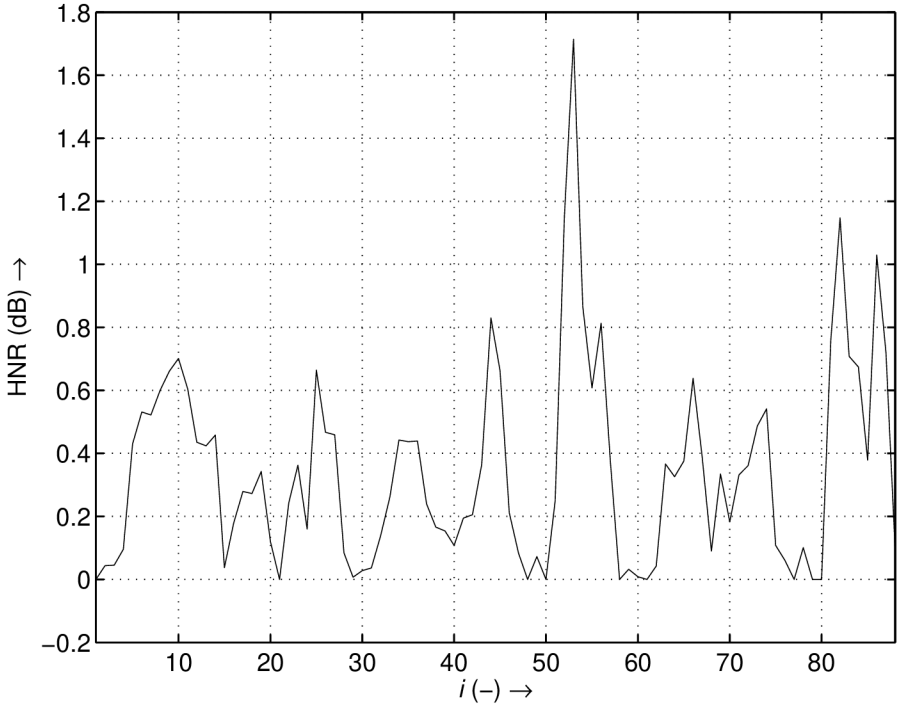


Fig. 3.47: HNR increase of ABS-FM over ABS-S of the reconstructed harmonic part of the sound sample *soprano*.

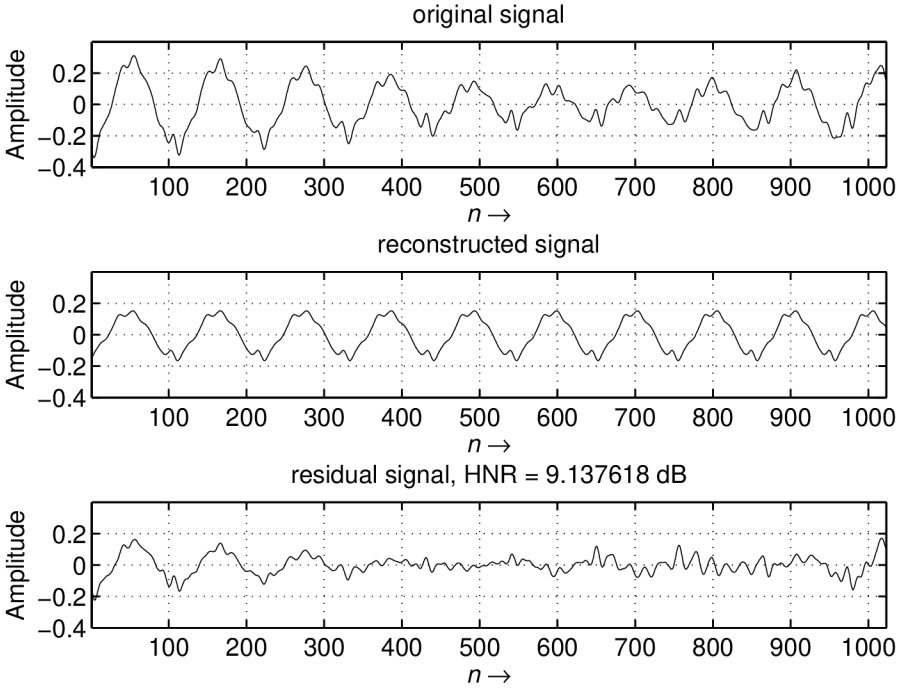


Fig. 3.48: Reconstruction of the 44-th segment of the *soprano* sound sample using ABS-FM.

4 CONCLUSION

This thesis was focused on methods for representation of harmonic signals with time-varying frequency components. In section 1.1 a problem which occurs when such signals are analyzed using traditional methods is presented as well as state-of-the-art methods which aim at accurate representation of these signals. Most of the focus of this section is on the Fan-Chirp Transform and Harmonic Transform which are both generalizations of the Fourier Transform for harmonic signals with time-varying frequency components and therefore they share some resemblances.

The chapter 3.1 is dedicated to Harmonic Transform and its computation speed. Fundamental frequency estimation is a prerequisite to computing the Harmonic Transform which has so far been computed using Spectral Flatness Measure. An algorithm to decrease the number of operations needed for SFM computation is presented based on the fact that the Harmonic Transform's image is one-sided. However the Harmonic Transform is enumerated using direct computation from (1.49) which employs $\mathcal{O}(N^2)$ computational complexity. Therefore further research was aimed at decreasing the computational complexity of Harmonic Transform.

Section 3.2 introduces the Fast Harmonic Transform. The fast transform has been designed by splitting the Harmonic Transform into time-warping of the input signal and performing FFT. This allows for subquadratic computational complexity. Analysis of the number of operations requred to compute FHT can be found on Table 3.2. However the time-warping operation involves interpolation which introduces noise to the signal and renders SFM ineffective as fundamental frequency change estimation algorithm. It also introduces aliasing which is dealt with in section 3.6 using oversampling and different interpolation methods.

Since SFM cannot be used as a fundamental frequency change algorithm for FHT, we have introduced two methods of its estimation. First method is based on computing a gathered log-spectrum on a range of fundamental frequencies and its changes. This method is rather fast though it suffers in fundamental frequency resolution. Second method is based on reconstruction error of harmonic part of the signal using harmonic parameter estimation. This method is slower than the first method, though it offers better resolution in fundamental frequency estimation. Both of these methods have been run on a speech signal *micf01sa02* to compare their results.

Finally, since until now all papers published on the HT have been applied on speech signals sampled at 8 kHz, we wanted to analyze real signals with frequency modulation sampled at 44.1 kHz, which is a common sampling frequency in digital audio. The selected signals are composed of vocal and instrument samples and the results can be seen in section 3.8. Generally it can be said that the HT decreases re-

construction error (i.e. the ability to represent the signal) for signals with frequency modulation.

Another transform, Pitch Tracking Modified DFT, introduced concurrently with the HT is analyzed in this thesis and section 3.7 provides proof it is equivalent to the HT.

BIBLIOGRAPHY

[1] M. Goodwin and M. Vetterli, "Time-frequency signal models for music analysis, transformation, and synthesis", in *Proceedings of the IEEE-SP International Symposium on Time-Frequency and Time-Scale Analysis*, 1996, pp. 133–136.

[2] G. Wakefield, "Time-pitch representations: acoustic signal processing and auditory representations", in *Time-Frequency and Time-Scale Analysis, 1998. Proceedings of the IEEE-SP International Symposium on*, Oct. 1998, pp. 577–580.

[3] A. Makur and S. Mitra, "Warped discrete-fourier transform: theory and applications", *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, vol. 48, no. 9, pp. 1086–1093, Sep. 2001, ISSN: 1057-7122.

[4] R. Sluijter and A. Janssen, "A time warper for speech signals", in *Speech Coding Proceedings, 1999 IEEE Workshop on*, 1999, pp. 150–152. DOI: 10.1109/SCFT.1999.781514.

[5] R. Baraniuk and D. L. Jones, "Warped wavelet bases: unitary equivalence and signal processing", in *in Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing — ICASSP '93*, 1993, pp. 320–323.

[6] R. Baraniuk and D. Jones, "Unitary equivalence: a new twist on signal processing", *Signal Processing, IEEE Transactions on*, vol. 43, no. 10, pp. 2269–2282, 1995, ISSN: 1053-587X. DOI: 10.1109/78.469861.

[7] L. Cohen, "The scale representation", *Signal Processing, IEEE Transactions on*, vol. 41, no. 12, pp. 3275–3292, Dec. 1993, ISSN: 1053-587X.

[8] T. Irino and R. D. Patterson, "Segregating information about the size and shape of the vocal tract using a time-domain auditory model: the stabilised wavelet-mellin transform", *Speech Commun.*, vol. 36, pp. 181–203, 3 Mar. 2002, ISSN: 0167-6393.

[9] A. De Sena and D. Rocchesso, "A study on using the mellin transform for vowel recognition", in *SMC05, International conference on Sound Music and Computing*, Salerno, Italy, 2005.

[10] F. Zhang, G. Bi, and Y. Chen, "Harmonic transform", *Vision, Image and Signal Processing, IEE Proceedings*, vol. 151, no. 4, pp. 257–263, Aug. 2004, ISSN: 1350-245X.

[13] L. Weruaga and M. Képesi, "The fan-chirp transform for non-stationary harmonic signals", *Signal Processing*, vol. 87, no. 6, pp. 1504–1522, 2007, ISSN: 0165-1684.

[14] M. Képesi and L. Weruaga, "Adaptive chirp-based time–frequency analysis of speech signals", *Speech Communication*, vol. 48, no. 5, pp. 474–492, 2006.

[15] L. Almeida, "The fractional fourier transform and time-frequency representations", *Signal Processing, IEEE Transactions on*, vol. 42, no. 11, pp. 3084–3091, 1994, ISSN: 1053-587X. DOI: 10.1109/78.330368.

[16] D. Bailey and P. Swarztrauber, "The fractional fourier transform and applications", *SIAM Review*, vol. 33, no. 3, pp. 389–404, Sep. 1995.

[17] H. Ozaktas, A. Kutay, and Z. Zalevsky, *The Fractional Fourier Transform: With Applications in Optics and Signal Processing*, ser. Wiley Series in Pure and Applied Optics. Wiley, 2001, ISBN: 9780471963462. [Online]. Available: http://books.google.cz/books?id=1TQbAQAAIAAJ.

[18] J. Vargas-Rubio and B. Santhanam, "An improved spectrogram using the multiangle centered discrete fractional fourier transform", in *Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP '05). IEEE International Conference on*, vol. 4, 2005, iv/505–iv/508 Vol. 4. DOI: 10.1109/ICASSP.2005.1416056.

[19] R. Tao, Y.-L. Li, and Y. Wang, "Short-time fractional fourier transform and its applications", *Signal Processing, IEEE Transactions on*, vol. 58, no. 5, pp. 2568–2580, May 2010, ISSN: 1053-587X.

[20] L. Weruaga and M. Képesi, "Speech analysis with the short-time chirp transform", in *Eurospeech*, Geneve, Sep. 2003, pp. 53–56.

[21] M. Képesi and L. Weruaga, "Speech analysis with the fast chirp transform", 2004.

[22] S. Mann and S. Haykin, "The chirplet transform: physical considerations", *Signal Processing, IEEE Transactions on*, vol. 43, no. 11, pp. 2745–2761, 1995, ISSN: 1053-587X. DOI: 10.1109/78.482123.

[23] A. Pavlovets and A. A. Petrovsky, "Robust hnr-based closed-loop pitch and harmonic parameters estimation", in *INTERSPEECH*, 2011, pp. 1981–1984.

[24] P. Zubrycki and A. Petrovsky, "Analysis/synthesis speech model based on the pitch-tracking periodic-aperiodic decomposition", *Information Processing and Security Systems*, pp. 33–42, 2005.

[25] Y. Pantazis, O. Rosec, and Y. Stylianou, "Iterative estimation of sinusoidal signal parameters", *Signal Processing Letters, IEEE*, vol. 17, no. 5, pp. 461–464, 2010.

[26] ——, "On the properties of a time-varying quasi-harmonic model of speech", in *INTERSPEECH*, 2008, pp. 1044–1047.

[27] ——, "Adaptive am-fm signal decomposition with application to speech analysis", *IEEE Transactions on Audio, Speech & Language Processing*, vol. 19, no. 2, pp. 290–300, 2011.

[28] G. Kafentzis, Y. Pantazis, O. Rosec, and Y. Stylianou, "An extension of the adaptive quasi-harmonic model", in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, 2012, pp. 4605–4608.

[29] S. Marchand, "The simplest analysis method for non-stationary sinusoidal modeling", in *Proc. of the 15th Int. Conference on Digital Audio Effects (DAFx-12)*, York, UK, Sep. 2012.

[30] R. McAulay and T. Quatieri, "Speech analysis/synthesis based on a sinusoidal representation", *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 34, no. 4, pp. 744–754, 1986.

[31] J. Smith and X. Serra, *PARSHL: an analysis/synthesis program for non-harmonic sounds based on a sinusoidal representation*, ser. Report no. 43. CCRMA, Dept. of Music, Stanford University, 1987.

[32] B. Hamilton and P. Depalle, "Comparisons of parameter estimation methods for an exponential polynomial sound signal model", in *Audio Engineering Society Conference: 45th International Conference: Applications of Time-Frequency Processing in Audio*, Mar. 2012. [Online]. Available: `http://www.aes.org/e-lib/browse.cfm?elib=16180`.

[33] M. Betser, "Modélisation sinusoidale et applications a l'indexation sonore", PhD thesis, École Nationale Supérieure des Télécommunications, Paris, France, Apr. 2008.

[34] B. Hamilton, P. Depalle, and S. Marchand, "Theoretical and practical comparisons of the reassignment method and the derivative method for the estimation of the frequency slope", in *Applications of Signal Processing to Audio and Acoustics, 2009. WASPAA '09. IEEE Workshop on*, 2009, pp. 345–348. DOI: `10.1109/ASPAA.2009.5346513`.

[35] S. Musevic and J. Bonada, "Comparison of non-stationary sinusoid estimation methods using reassignment and derivatives", in *Sound and Music Computing Conference*, 2010. [Online]. Available: `http://smcnetwork.org/files/proceedings/2010/14.pdf`.

[36] ——, "Generalized reassignment with an adaptive polynomial-phase for fourier-kernel for the estimation of non-stationary sinusoidal parameters", in *Proc. of the 14th International Conference on Digital Audio Effects (DAFx-11)*, Paris, France, Sep. 2011.

[37] B. Hamilton and P. Depalle, "A unified view of non-stationary sinusoidal parameter estimation methods using signal derivatives", in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, 2012, pp. 369–372. DOI: 10.1109/ICASSP.2012.6287893.

[38] M. Abe and J. Smith, "Am/fm rate estimation for time-varying sinusoidal modeling", in *Acoustics, Speech, and Signal Processing, 2005. Proceedings. (ICASSP '05). IEEE International Conference on*, vol. 3, 2005, iii/201–iii/204 Vol. 3.

[39] B. Hamilton, "Non-stationary sinusoidal parameter estimation", Master's thesis, McGill University, Montreal, Canada, 2011.

[40] M. Betser, "Sinusoidal polynomial parameter estimation using the distribution derivative", *Signal Processing, IEEE Transactions on*, vol. 57, no. 12, pp. 4633–4645, 2009, ISSN: 1053-587X. DOI: 10.1109/TSP.2009.2027401.

[41] S. Marchand and P. Depalle, "Generalization of the derivative analysis method to non-stationary sinusoidal modeling", in *Proc. of the Int. Conf. on Digital Audio Effects (DAFx-08)*, 2008, pp. 281–288. [Online]. Available: http://www.acoustics.hut.fi/dafx08/papers/dafx08_48/slides_281.pdf.

[42] X. Wen and M. Sandler, "Notes on model-based non-stationary sinusoid estimation methods using derivatives", in *Proc. of the 12th Int. Conference on Digital Audio Effects (DAFx-09)*, Como, Italy, 2009.

[43] S. Marchand, "Improving spectral analysis precision with an enhanced phase vocoder using signal derivatives", in *In Proc. DAFX98 Digital Audio Effects Workshop*, MIT Press, 1998, pp. 114–118.

[44] M. Betser, P. Collen, G. Richard, and B. David, "Estimation of frequency for am/fm models using the phase vocoder framework", *Signal Processing, IEEE Transactions on*, vol. 56, no. 2, pp. 505–517, 2008, ISSN: 1053-587X. DOI: 10.1109/TSP.2007.906768.

[45] A. Petrovsky, E. Azarov, and A. Petrovsky, "Hybrid signal decomposition based on instantaneous harmonic parameters and perceptually motivated wavelet packets for scalable audio coding", *Signal Processing*, vol. 91, no. 6, pp. 1489 –1504, 2011, <ce:title>Fourier Related Transforms for Non-Stationary Signals</ce:title>, ISSN: 0165-1684.

[46] D. Gabor, "Theory of Communication", *J. IEE*, vol. 93, no. 26, pp. 429–457, Nov. 1946.

[47] P. Maragos, J. Kaiser, and T. Quatieri, "Energy separation in signal modulations with application to speech analysis", *Signal Processing, IEEE Transactions on*, vol. 41, no. 10, pp. 3024–3051, 1993, ISSN: 1053-587X. DOI: 10.1109/78.277799.

[48] T. Abe, T. Kobayashi, and S. Imai, "Harmonics tracking and pitch extraction based on instantaneous frequency", in *Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on*, vol. 1, 1995, 756–759 vol.1. DOI: 10.1109/ICASSP.1995.479804.

[49] P. Zubrycki and A. Petrovsky, "Accurate speech decomposition into periodic and aperiodic components based on discrete harmonic transform", in *15th European Signal Processing Conference (EUSIPCO 2007)*, Poznań, Poland, 2007, pp. 2336–2340.

[50] B. Yegnanarayana, C. d'Alessandro, and V. Darsinos, "An iterative algorithm for decomposition of speech signals into periodic and aperiodic components", *Speech and Audio Processing, IEEE Transactions on*, vol. 6, no. 1, pp. 1–11, 1998, ISSN: 1063-6676. DOI: 10.1109/89.650304.

[51] P. Maragos, J. Kaiser, and T. Quatieri, "On amplitude and frequency demodulation using energy operators", *Signal Processing, IEEE Transactions on*, vol. 41, no. 4, pp. 1532–1550, 1993, ISSN: 1053-587X. DOI: 10.1109/78.212729.

[52] E. Azarov, A. Petrovsky, and M. Parfieniuk, "Estimation of the instantaneous harmonic parameters of speech", in *16th European Signal Processing Conference (EUSIPCO 2008)*, Lausanne, Switzerland, Audust 25-29 2008.

[53] D. Talkin, "A robust algorithm for pitch tracking (rapt)", *Speech coding and synthesis*, vol. 495, p. 518, 1995.

[54] E. Azarov, M. Vashkevich, and A. Petrovsky, "Instantaneous pitch estimation based on rapt framework", in *Signal Processing Conference (EUSIPCO), 2012 Proceedings of the 20th European*, 2012, pp. 2787–2791.

[55] E. Azarov and A. Petrovsky, "Instantaneous harmonic analysis for vocal processing", in *Proc. of the 12th Int. Conference on Digital Audio Effects (DAFx-09)*, Como, Italy, Sep. 2009.

[56] E. Azarov, A. Petrovsky, and M. Parfieniuk, "High-quality time stretch and pitch shift effects for speech and audio using the instantaneous harmonic analysis", *EURASIP Journal on Advances in Signal Processing*, vol. 2010, no. 1, p. 712 749, 2010, ISSN: 1687-6180. DOI: 10.1155/2010/712749. [Online]. Available: http://asp.eurasipjournals.com/content/2010/1/712749.

[57] E. Azarov and A. Petrovsky, "Real-time voice conversion based on instantaneous harmonic parameters", in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, 2011, pp. 5140–5143. DOI: 10.1109/ICASSP.2011.5947514.

[58] ——, "Linear prediction of deterministic components in hybrid signal representation", in *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on*, 2010, pp. 2662–2665. DOI: 10.1109/ISCAS.2010.5537055.

[59] A. Petrovsky, E. Azarov, and A. Petrovsky, "Harmonic representation and auditory model-based parametric matching and its application in speech/audio analysis", in *Audio Engineering Society Convention 126*, May 2009. [Online]. Available: http://www.aes.org/e-lib/browse.cfm?elib=14901.

[60] ——, "Combining advanced sinusoidal and waveform matching models for parametric audio/speech coding", in *17th European Signal Processing Conference (EUSIPCO 2009)*, Glasgow, Scotland: 17th EUSIPCO, Aug. 2009.

[61] F. Zhang, Y.-Q. Chen, and G. Bi, "Adaptive harmonic fractional fourier transform", *Signal Processing Letters, IEEE*, vol. 6, no. 11, pp. 281–283, 1999, ISSN: 1070-9908. DOI: 10.1109/97.796288.

[62] ——, "Adaptive harmonic fractional fourier transform", in *Circuits and Systems, 2000. Proceedings. ISCAS 2000 Geneva. The 2000 IEEE International Symposium on*, vol. 5, 2000, 45–48 vol.5. DOI: 10.1109/ISCAS.2000.857359.

[63] P. Zubrycki and A. Petrovsky, "Accurate estimation of harmonic amplitudes in voiced speech based on harmonic transform", in *Signals and Electronic Systems, 2008. ICSES '08. International Conference on*, Sep. 2008, pp. 47 – 50.

[64] P. Zubrycki and A. Petrovsky, "Quasi-periodic signal analysis using harmonic transform with application to voiced speech processing", in *Proceedins of International Symposium on Circuits and Systems (ISCAS 2010)*, Paris, France, May 2010, pp. 2374–2377.

[65] P. Jackson and C. Shadle, "Pitch-scaled estimation of simultaneous voiced and turbulence-noise components in speech", *Speech and Audio Processing, IEEE Transactions on*, vol. 9, no. 7, pp. 713–726, Oct. 2001, ISSN: 1063-6676.

[66] P. Cancela, E. Lopez, and M. Rocamora, "Fan chirp transform for music representation", in *Proc. of the 13th Int. Conference on Digital Audio Effects (DAFx-10)*, Graz, Austria, Sep. 2010.

[67] R. Dunn and T. Quatieri, "Sinewave analysis/synthesis based on the fan-chirp tranform", in *Applications of Signal Processing to Audio and Acoustics, 2007 IEEE Workshop on*, 2007, pp. 247–250. DOI: 10.1109/ASPAA.2007.4393028.

[68] J. C. Brown, "Calculation of a constant q spectral transform", *The Journal of the Acoustical Society of America*, vol. 89, no. 1, pp. 425–434, 1991. DOI: 10.1121/1.400476.

[69] U. Luxburg, "A tutorial on spectral clustering", *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, Dec. 2007, ISSN: 0960-3174. DOI: 10.1007/s11222-007-9033-z.

[70] M. Rocamora and P. Cancela, "Pitch tracking in polyphonic audio by clustering local fundamental frequency estimates", in *Brazilian AES Audio Engineering Congress, 9th. S ao Paulo, Brazil*, May 2011. [Online]. Available: http://iie.fing.edu.uy/publicaciones/2011/RC11.

[71] M. Rocamora and A. Pardo, "Separation and classification of harmonic sounds for singing voice detection", in *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, ser. Lecture Notes in Computer Science, L. Alvarez, M. Mejail, L. Gomez, and J. Jacobo, Eds., vol. 7441, Springer Berlin Heidelberg, 2012, pp. 707–714, ISBN: 978-3-642-33274-6. DOI: 10.1007/978-3-642-33275-3_87.

[72] L. Jure, E. López, M. Rocamora, P. Cancela, H. Sponton, and I. Irigaray, "Pitch content visualization tools for music performance analysis", in *Proceedings of the 13th International Society for Music Information Retrieval Conference*, Porto, Portugal, Oct. 2012.

[73] P. Cancela, "Tracking melody in polyphonic audio", in *The 9th International Conference on Music Information Retrieval, ISMIR 2008*, Philadelphia, Pennsylvania, USA, Sep. 2008.

[74] H. Nguyen and L. Weruaga, "Time-frequency analysis of vietnamese speech inspired on chirp auditory selectivity", in *PRICAI*, 2008, pp. 284–295.

[75] M. Bartkowiak, "Application of the fan-chirp transform to hybrid sinusoidal+noise modeling of polyphonic audio", *16th European Signal Processing Confernce*, 2008.

[76] P. Zhao, Z. Zhang, and X. Wu, "Monaural speech separation based on multi-scale fan-chirp transform", in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, 2008, pp. 161–164. DOI: 10.1109/ICASSP.2008.4517571.

[77] R. B. Dunn, T. F. Quatieri, and N. Malyska, “Sinewave parameter estimation using the fast fan-chirp transform”, in *WASPAA*, 2009, pp. 349–352.

[78] A. Kondoz, *Digital Speech: Coding for Low Bit Rate Communication Systems*. Wiley, 2004, ISBN: 9780470870075. [Online]. Available: `http://books.google.cz/books?id=zKfz4uYgpmUC`.

[79] M. Hinich, “Detecting a hidden periodic signal when its period is unknown”, *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 30, no. 5, pp. 747–750, 1982, ISSN: 0096-3518. DOI: `10.1109/TASSP.1982.1163952`.

[82] A. De Sena and D. Rocchesso, “A fast mellin and scale transform”, *EURASIP J. Appl. Signal Process.*, vol. 2007, pp. 75–75, 1 Jan. 2007, ISSN: 1110-8657.

[84] W. Martens and A. Maurui, “Categories of perception for vibrato, flange, and stereo chorus: mapping out the musically useful ranges of modulation rate and depth for delay-based effects”, in *Proc. of the 9th Int. Conference on Digital Audio Effects (DAFx-06)*, Montreal, Canada, Sep. 2006, pp. 149–152.

[85] U. Zölzer, *DAFX - Digital Audio Effects*, 1st ed. New York: John Wiley & Sons, Ltd, 2002, p. 533, ISBN: 0-471-49078-4.

[86] J. Dattorro, “Effect design: part 2 delay-line modulation and chorus”, *Journal of the Acoustical Society of America*, vol. 45, no. 10, pp. 764–788, Oct. 1997.

AUTHOR'S PUBLICATIONS

[11] M. Trzos, "Frequency warping via warped linear prediction", in *Telecommunications and Signal Processing (TSP), 2011 34th International Conference on*, Aug. 2011, pp. 348–350.

[12] ——, "Determining the prediction order of warped linear prediction for frequency warping", *Elektrorevue - Internetový časopis (http://www.elektrorevue.cz)*, vol. 2, no. 4, pp. 51–57, Dec. 2011.

[80] ——, "Optimalizace odhadu fázové funkce harmonické transformace", *Elektrorevue - Internetový časopis (http://www.elektrorevue.cz)*, vol. 14, no. 4, pp. 1–4, 2012.

[81] M. Trzos and H. Khaddour, "Efficient spectral estimation of non-stationary harmonic signals using harmonic transform", *International Journal of Advances in Telecommunications, Electrotechnics, Signals and Systems*, vol. 1, no. 2-3, 2012, ISSN: 1805-5443.

[83] M. Trzos and H. Ladhe, "Fast implementation of harmonic transform", in *Telecommunications and Signal Processing (TSP), 2013 36th International Conference on*, Jul. 2013, pp. 498–501. DOI: 10.1109/TSP.2013.6613982.

LIST OF SYMBOLS, PHYSICAL CONSTANTS AND ABBREVIATIONS

FT Fourier Transform

IFT Inverse Fourier Transform

FrFT Fractional Fourier Transform

HT Harmonic Transform

FChT Fan-Chirp Transform

CT Chirp Transform

ChT Chirplet Transform

PTDFT Pitch Tracking Modified Fourier Transform

TVDFT Time-Varying Discrete Fourier Transform

QHM Quasi-Harmonic Model

LS Least Squares

AM Amplitude Modulation

FM Frequency Modulation

SM Sinusoidal Model

QIFFT Quadratically Interpolated Fast Fourier Transform

DDM Distributed Derivative Method

GDM Generalized Derivative Method

GRM General Reassignment Method

DESA Discrete Energy Separation Algorithm

IHT Inverse Harmonic Transform

STHT Short-Time Harmonic Transform

DHT Discrete Harmonic Transform

CQT Constant-Q Transform

IFHT Inverse Fast Harmonic Transform

WFT Warped Fourier Transform

GlogS Gathered log-Spectrum

HNR Harmonic to Noise Ratio

SFM Spectral Flatness Measure

MSFM Modified Spectral Flatness Measure

n, m discrete-time index

t time

T length of a segment in seconds

N, M length of a segment in samples

L delay line length in samples

f_s sampling frequency

T_s sampling period

ω angular frequency

ω_I instantaneous angular frequency

f_0 fundamental frequency

f_c central frequency

$s(t)$ continuous-time signal

$\hat{s}(t)$ estimation of the continuous-time signal $s(t)$

$S(\cdot)$ complex spectrum

c_k complex amplitude of the k-th sinusoid

f_k frequency of the k-th sinusoid

$\hat{f}_k$ estimated frequency of the k-th sinusoid

$\hat{\eta}_k$ error of frequency estimation of the k-th sinusoid

f_k frequency of the k-th sinusoid

$w(t)$ continuous-time analysis window

a_k instantaneous amplitude of the k-th sinusoid

ϕ_k instantaneous phase of the k-th sinusoid

$v(t)$ additive noise

α_k complex non-stationary sinusoidal parameters of the k-th sinusoid

ω_0 instantaneous frequency of the k-th sinusoid

$\langle \cdot, \cdot \rangle$ dot product

$\mathcal{H}[\cdot]$ Hilbert transform

$\psi[\cdot]$ Teager energy operator

$\phi_{\mathrm{u}}(t)$ unit phase function

$\phi'_{\mathrm{u}}(t)$ first derivative of the unit phase function

ϵ fundamental frequency change rate

a slope of fundamental frequency change

$\alpha_a(n)$ unit phase function for fundamental frequency slope a

$S_a(\cdot)$ complex harmonic spectrum for fundamental frequency slope a

$\phi_\alpha(t)$ time-warping function with chirp rate α

A_k amplitude of the k-th harmonic

$\varphi_k(0)$ initial phase of the k-th harmonic

$\hat{h}(n)$ estimated harmonic component

$r(f)$ pitch refinement score

α chirp rate

$\hat{\alpha}$ discrete-time chirp rate

$\tilde{x}(\tau)$ time-warped version of the signal $x(t)$

$\tilde{\rho}(\tau)$ time-warped version of the scaling function $\rho(t)$

$\psi_\alpha(t)$ time-warping function, inverse of $\phi_\alpha(t)$

$\rho_0(f)$ gathered log-spectrum at frequency f

$\rho(f)$ refined gathered log-spectrum

P_{s} signal power

P_{n} noise power

E_{h} energy of harmonic component

E_{n} energy of noise component

LIST OF APPENDICES

A SOUND SAMPLES

A list of the sound samples used in this thesis and their location on the included DVD in the `/samples` directory.

- *happychild.wav*
- *viola.wav*
- *salvation.wav*
- *salvationmod.wav*
- *soprano.wav*
- *sopranoshort.wav*
- *micf01sa2.wav*

B MATLAB SCRIPTS OF THE PRESENTED ALGORITHMS

```
function [ S ] = DHT( x, a )
%DHT Summary of this function goes here
%   Detailed explanation goes here
%   x - input signal
%   a - phase function

% length of the segment
N = length(x);

% input samples indices
n = 0:N-1;
% phase function
alpha = n.*(1-a/2+a.*n/(2*N));
% normalization - derivative of the phase function
alphad = 1-a/2+(a.*n)/N;

% preallocate the output buffer
S = zeros(N,1);
% compute the direct Harmonic transform
parfor k = 0:N-1
    for n = 0:N-1
        S(k+1) = S(k+1) + ...
            alphad(n+1).*x(n+1).*exp(((-1j*2*pi*k)/N)*alpha(n+1));
    end
end

end
```

```
function [ S ] = DHTF0( x, a, fr, fs, nh )
%DHTF0 Computes nh harmonics starting from fundamental frequency fr
%   Detailed explanation goes here
%   x - input signal
%   a - phase function
%   fr - fundamental frequency
%   fs - sampling frequency
%   nh - number of harmonics

% length of the segment
N = length(x);

```

```
% input samples indices
n = 0:N-1;
% phase function
alpha = n.*(1-a/2+a.*n/(2*N));
% normalization - derivative of the phase function
alphad = 1-a/2+(a.*n)/N;

% preallocate the output buffer
S = zeros(nh,1);
% compute the fundamental frequency aligned direct Harmonic ...
    transform
parfor k = 1:nh
    for n = 0:N-1
        S(k) = S(k) + alphad(n+1).*x(n+1) .* ...
            exp(((-1j*2*pi*k*fr)/fs)*alpha(n+1));
    end
end
end
```

```
function [ sfm ] = SFM( S, sfscale )
%SFM Computes Spectral flatness measure for double sided spectrum
% input - S         double-sided spectrum
%         sfscale   'lin' for SFM
%                   'log' for SFM_dB

% input spectrum length
N = length(S);

% take the absolute value of the spectrum and take only the ...
    positive values
absS = abs(S);
absS = absS(absS>0);

% compute the nominator of the SFM
num = exp((1/N)*sum(log(absS)));
% compute the denominator of the SFM
den = (1/N) * sum(absS);

% Select linear or logarithmic scale
switch sfscale
    case 'log'
        sfm = 20*log10(num/den);
    case 'lin'
        sfm = num/den;
    otherwise
```

```
        sfm = 20*log10(num/den);
end
```

```
function [ msfm ] = MSFM( S, sfscale )
%MSFM Computes Modified Spectral flatness measure for signle ...
    sided spectrum
% input - S         double-sided spectrum
%         sfscale   'lin' for MSFM
%                   'log' for MSFM_dB

% number of samples of the first half of the input spectrum
N = (length(S)/2)+1;
% take half of the input spectrum
S = S(1:N);

% take the absolute value of the spectrum and take only the ...
    positive values
absS = abs(S);
absS = absS(absS>0);

% compute the nominator of the SFM
nom = exp((1/N)*sum(log(absS)));
% compute the denominator of the SFM
den = (1/N) * sum(absS);

% Select linear or logarithmic scale
switch sfscale
    case 'log'
        msfm = 20*log10(nom/den);
    case 'lin'
        msfm = nom/den;
    otherwise
        msfm = 20*log10(nom/den);
end
```

```
function [ S ] = FHT( x, a, os, NFFT )
%FHT Computation of Fast Harmonic Transform
%   x - input signal
%   a - phase function
%   os - oversampling factor
%   NFFT - length of fft
% zero-phase zero-padding = M-1/2+1:N-(M-1)/2-1

% number of input samples multiplied by the oversampling factor
```

```
10 M = length(x)*os;
11
12 % fill in missing arguments
13 if nargin < 4
14     N = length(x)*os;
15 else
16     N = NFFT;
17 end
18
19 % resampling of the input signal
20 x = resample(x, os, 1)./os;
21 % indices of the input samples
22 n = 0:M-1;
23
24 % inverse phase function
25 alpha = M/2 - M/a + (M*(a^2/4 - a + (2*a*n)/M + 1).^(1/2))/a;
26 % inverse normalization function
27 alphad = 1./((a*(M/2 - M/a + (M*(a^2/4 - a + (2*a*n)/M + ...
       1).^(1/2))/a))/M - a/2 + 1);
28
29 % of the phase function is zero, do not interpolate
30 if a == 0
31     xa = x';
32 else
33 % normalize and interpolate the input segment at indices of the ...
       warped time axis given by the phase function
34     x = x.*alphad';
35     xa = interp1(n,x,alpha,'sinc','extrap');
36 end
37
38 % If we are using zero-padding, insert the zero samples in the ...
       middle of the segment
39 if N > M % zero-padding
40     if ~rem((M-1),2)
41         Mo = (M-1)/2;
42         xa = [xa(Mo+1:M) zeros(1,N-M) xa(1:Mo)];
43     else
44         Mo = ceil((M-1)/2);
45         xa = [xa(Mo+1:M) zeros(1,N-M) xa(1:Mo)];
46     end
47 else
48 % if we are not using zero-padding, do only fftshift
49     xa = fftshift(xa);
50 end
51
52 % perform FFT algorithm
```

```
53 S = fft(xa,NFFT);
54
55 end
```

```
1  function [ x ] = IFHT( S, a, os, NFFT )
2  %IFHT Computation of Inverse Harmonic Transform
3  %   Detailed explanation goes here
4  %   x — input signal
5  %   a — phase function
6  %   os — oversampling factor
7  %   NFFT — length of fft
8
9  % take length of the input segment
10 M = length(S);
11 NFFT = M;
12 % perform the inverse Fourier transform
13 xa = real(ifft(S));
14
15 % If we are using zero—padding, remove the zero samples from the ...
       middle of the segment
16 if NFFT > M
17     if ¬rem((M—1),2)
18         % zero—padding for even segment lengths
19         Mo = (M—1)/2;
20         xa = [xa(Mo+1:M) zeros(1,N—M) xa(1:Mo)];
21     else
22         % zero—padding for odd segment lengths
23         Mo = ceil((M—1)/2);
24         xa = [xa(Mo+1:M) zeros(1,N—M) xa(1:Mo)];
25     end
26 else
27 % if we are not using zero—padding, do only fftshift
28     xa = fftshift(xa);
29 end
30
31 % input samples indices
32 n = 0:M—1;
33
34 % phase function
35 alpha = M/2 — M/a + (M*(a^2/4 — a + (2*a*n)/M + 1).^(1/2))/a;
36 % normalization function
37 alphad = 1./((a — 2).^2/4 + (2*a.*n)/M).^(1/2);
38
39 % if the phase function is zero, do not perform interpolation
40 if a == 0
```

```
41      xa = xa';
42  else
43  % interpolate the input segment at the samples indices of the ...
        original signal and normalize
44      x = interp1(alpha,xa,n,'cubic','extrap');
45      x = x./alphad;
46  end
47
48  % resample the signal to its orignal length
49  x = resample(x, 1, os).*os;
50
51  end
```

```
1   %% Script for computation of Harmonic transform using gathered ...
        log-Spectrum as the fundamental frequency estimating algorithm
2   clear all;
3
4   os = 2; % oversampling factor
5   M = 512; % segment length
6   NFFT = 1024; % analysis window length
7   FMIN = 40; % minimum analyzed fundamental frequency
8   FMAX = 450; % maximum analyzed fundamental frequency
9
10  [x, fs] = wavread('mic_F01_sa2_8k');
11
12  hops = ceil((5e-3)*fs);
13  hop = M-hops;
14  Z=segmentace(x,M,hop);
15  [M, N] = size(Z);
16  N = 1:N;
17
18  win = hamming(M);
19  nwin = win./sum(win); % normalize the window
20  ar = -0.3:0.01:0.3; % range of fundamental frequency change
21
22  f = os*fs/2*linspace(0,1,NFFT/2+1);
23  freqs = (logspace(log10(FMIN),log10(FMAX),1000)); % 1000 ...
        logarithmically spaced values between FMIN and FMAX
24
25  nh = 4; % number of hypothetical harmonics in the analyzed segment
26  bt = 1:nh;
27  Sharma = zeros(length(freqs),length(ar));
28  Sharmaout = zeros(length(freqs), length(N));
29  Sout = zeros(NFFT,N); % output spectrogram
30  Hout = Sout; %
```

```
Aout = zeros(1,N); %
f0out = Aout;
S = zeros(NFFT,length(ar));
for g = N
    Zx = Z(:,g).*win;
    for i = 1:length(ar)
        S(:,i) = FHT(Zx, ar(i), os, NFFT); % Fast Harmonic Transform
    end

    Sx = S(1:floor(NFFT/2+1),:); % take left side of the spectrum
    absS = abs(Sx); % its magnitude value
    parfor f0s = 1:length(freqs) % for all input fundamental ...
        frequencies
        Sharma(f0s,:) = (1/nh) * ...
            sum(log(1+10*interp1(f,absS,freqs(f0s).*bt,'linear'))); ...
            % gathered log-Spectrum
    end

[G, I] = max(Sharma);
[H, Y] = max(G); % find the maximum value of the glogS
Aout(g) = ar(Y); % selected fundamental frequency change
f0out(g) = freqs(I(Y)); %selected fundamental frequency
Sharmaout(:,g) = Sharma(:,Y); % glogS of spectrogram with the ...
    selected fundamental frequency change
Sout(:,g) = FHT(Zx, ar(Y), os, NFFT); % output spectrogram

end
```

```
%% Script for computation of Harmonic transform using ...
    analysis-by-synthesis as the fundamental frequency ...
    estimating algorithm
M = 512; % segment length
NFFT = 512; % analysis window length
FMIN = 80; % minimum analyzed fundamental frequency
FSTEP = 0.5;
FMAX = 450; % maximum analyzed fundamental frequency

[x, fs] = wavread('mic_M01_sa2_8k');

hops = ceil((5e-3)*fs); % hop size
ovl = M-hops; % overlap from hop size
hop = ovl;
Z=segmentace(x,M,ovl);
[M, N] = size(Z);
N = 1:N;
```

```
16 win = hann(M); % analysis window
17 win = win./sum(win); % window normalization
18
19 ar = -0.05:0.001:0.05; % fundamental frequency change range
20
21 f = fs/2*linspace(0,1,NFFT/2+1);
22 freqs = FMIN:FSTEP:FMAX; % input fundamental frequencies
23
24 nh = 6; % hypothetical harmonics in the analyzed segment
25 bt = 1:nh;
26 Sharma = zeros(length(freqs),length(ar));
27 Sout = zeros(NFFT,length(N));
28 Hout = Sout;
29 Aout = zeros(1,length(N));
30 f0out = Aout;
31 S = zeros(NFFT,length(ar));
32 St = zeros(nh,1);
33 SNRe = zeros(length(freqs), length(ar));
34 SNReout = zeros(length(freqs), length(N));
35
36 k = 0:M-1;
37 for i = 1:length(ar)
38     a(:,i) = k.*(1-ar(i)/2+ar(i).*k/(2*M)); % precomputation of ...
          the phase modifier for input fundamental frequency ...
          change range
39 end
40
41 for g = N
42     Zx = Z(:,g).*win;
43     sumZx = sum(Zx.^2);
44     for i = 1:length(ar)
45         alp = ar(i);
46         parfor f0s = 1:length(freqs)
47             St = DHTF0(Zx,alp,freqs(f0s),fs,nh); % complex ...
                  coefficients of the fundamental and its harmonics
48             A = sqrt(real(St).^2 + imag(St).^2); % amplitude of ...
                  the harmonics
49             phi0 = atan2(imag(St), real(St)); % phase of the ...
                  harmonics
50             phi0 = unwrap(phi0);
51
52             R = zeros(1,M);
53             ph = (2*pi*freqs(f0s)/fs).*a(:,i)'; % phase
54             for k = 1:length(St)
55                 R = R + 2*A(k) * cos(k.*ph + phi0(k)); % ...
                      reconstruction of the input signal from the ...
```

```
                    harmonic parameters
56              end
57              HNRe(f0s,i) = ...
                    10*log10(sumZx/(sum((Zx-win.*R').^2))); % HNR ...
                    computation
58          end
59      end
60
61  [maxhnrval, ind] = max(HNRe(:));
62  [freqsind, arind] = ind2sub(size(HNRe),ind);
63
64  Aout(g) = ar(arind); % output fundamental frequency change
65  f0out(g) = freqs(freqsind); % output fundamental frequency
66  Hr(g) = maxhnrval; % output HNR value
67  Hr0(g) = HNRe(freqsind,ar==0); % output HNR value for a signal ...
        reconstructed without account for frequency modulation
68  Sout(:,g) = FHT(Zx, ar(arind), 1, NFFT); % output spectrogram
69
70  end
```

Curriculum Vitæ

Michal Trzos

- Tel: (+420) 776 690 786
- Email: michal.trzos@gmail.com

Personal information

- Born on 17th June, 1984
- Czech nationality

Education

- 2009–2015 Ph.D. in Teleinformatics, Brno University of Technology, Brno.
- 2008–2010 M.Sc. in Company Management and Economics , Brno University of Technology, Brno
- 2007–2009 M.Sc. in Communications and Informatics, Brno University of Technology, Brno
- 2004–2007 B.Sc. in Teleinformatics, Brno University of Technology, Brno

Employment

- 2013–present Senior Software Developer, Institute of Computer Science, Masaryk University, Brno, Czech Republic
- 2011–2012 Software developer, Phonexia, Brno, Czech Republic
- 2010–2011 Audio Programmer, Tonic Games, Copenhagen, Denmark
- 2007–2010 Software Developer, Audiffex, Boskovice, Czech Republic
- 2006–2008 ICT Specialist, Masaryk University, Brno, Czech Republic

Participation in Projects

- 2297/G1/2012 – Inovation of the Electroacoustics course. Holder: Ing. Trzos. 2012
- 3160/G1/2011 – Introduction of general-purpose computing on graphics processing units to Multimedia and graphics processors course. Holder: Ing. Trzos. 2011
- FEKT-S-11-17 – Research of Sophisticated Methods for Digital Audio and Image Signal Processing. Holder: Prof. Z. Smékal. 2011
- FEKT-S-10-16 – Research on Electronic Communication Systems. Holder: Prof. K. Vrba. 2010
- FR-TI1/495 – Manifold System for Multimedia Digital Signal Processing. Holder: Ing. Schimmel. 2009–2012

Invited Talks

- Estimation of harmonic parameters of linearly frequency modulated signals using analysis by synthesis approach. 4th SPLab workshop, November 20 to November 21, 2014
- Representation of frequency modulated audio signals using Fast harmonic transform. 3rd SPLab workshop, October 30 to November 1, 2012

Results

- Publications: 13
  - In proceedings of international conferences: 4
  - In other journals: 6
  - In other conferences: 3
- Software/Products: 2
- Citations (without self-citations): 1

Awards

- EEICT 2010 student conference and competition - 2nd prize