Tampere University of Technology
Publications 460

Anssi Klapuri

Signal Processing Methods for the Automatic Transcription of Music

Thesis for the degree of Doctor of Technology to be presented with
due permission for public examination and criticism in Auditorium S1,
at Tampere University of Technology, on the 17th of March 2004,
at 12 o clock noon.

Tampere 2004

ISBN 952-15-1147-8
ISSN 1459-2045

Anssi.Klapuri@tut.fi

http://www.cs.tut.fi/~klap/

Abstract

Signal processing methods for the automatic transcription of music are developed in this thesis. Music transcription is here understood as the process of analyzing a music signal so as to write down the parameters of the sounds that occur in it. The applied notation can be the traditional musical notation or any symbolic representation which gives sufficient information for performing the piece using the available musical instruments. Recovering the musical notation automatically for a given acoustic signal allows musicians to reproduce and modify the original performance. Another principal application is structured audio coding: a MIDI-like representation is extremely compact yet retains the identifiability and characteristics of a piece of music to an important degree.

The scope of this thesis is in the automatic transcription of the harmonic and melodic parts of real-world music signals. Detecting or labeling the sounds of percussive instruments (drums) is not attempted, although the presence of these is allowed in the target signals. Algorithms are proposed that address two distinct subproblems of music transcription. The main part of the thesis is dedicated to *multiple fundamental frequency (F0) estimation*, that is, estimation of the F0s of several concurrent musical sounds. The other subproblem addressed is *musical meter estimation*. This has to do with rhythmic aspects of music and refers to the estimation of the regular pattern of strong and weak beats in a piece of music.

For multiple-F0 estimation, two different algorithms are proposed. Both methods are based on an iterative approach, where the F0 of the most prominent sound is estimated, the sound is cancelled from the mixture, and the process is repeated for the residual. The first method is derived in a pragmatic manner and is based on the acoustic properties of musical sound mixtures. For the estimation stage, an algorithm is proposed which utilizes the frequency relationships of simultaneous spectral components, without assuming ideal harmonicity. For the cancelling stage, a new processing principle, *spectral smoothness*, is proposed as an efficient new mechanism for separating the detected sounds from the mixture signal.

The other method is derived from known properties of the human auditory system. More specifically, it is assumed that the peripheral parts of hearing can be modelled by a bank of bandpass filters, followed by half-wave rectification and compression of the subband signals. It is shown that this basic structure allows the combined use of time-domain periodicity and frequency-domain periodicity for F0 extraction. In the derived algorithm, the higher-order (unresolved) harmonic partials of a sound are processed collectively, without the need to detect or estimate individual partials. This has the consequence that the method works reasonably accurately for short analysis frames. Computational efficiency of the method is based on calculating a frequency-domain approximation of the *summary autocorrelation function*, a physiologically-motivated representation of sound.

Both of the proposed multiple-F0 estimation methods operate within a single time frame and arrive at approximately the same error rates. However, the auditorily-motivated method is superior in short analysis frames. On the other hand, the pragmatically-oriented method is “complete” in the sense that it includes mechanisms for suppressing additive noise (drums) and for estimating the number of concurrent sounds in the analyzed signal. In musical interval and chord identification tasks, both algorithms outperformed the average of ten trained musicians.

For musical meter estimation, a method is proposed which performs meter analysis jointly at three different time scales: at the temporally atomic *tatum* pulse level, at the *tactus* pulse level which corresponds to the tempo of a piece, and at the *musical measure* level. Acoustic signals from arbitrary musical genres are considered. For the initial time-frequency analysis, a new technique is proposed which measures the degree of musical accent as a function of time at four different frequency ranges. This is followed by a bank of comb filter resonators which perform feature extraction for estimating the periods and phases of the three pulses. The features are processed by a probabilistic model which represents primitive musical knowledge and performs joint estimation of the tatum, tactus, and measure pulses. The model takes into account the temporal dependencies between successive estimates and enables both causal and non-causal estimation. In simulations, the method worked robustly for different types of music and improved over two state-of-the-art reference methods. Also, the problem of detecting the beginnings of discrete sound events in acoustic signals, *onset detection*,is separately discussed.

Keywords—Acoustic signal analysis, music transcription, fundamental frequency estimation, musical meter estimation, sound onset detection.

Preface

This work has been carried out during 1998–2004 at the Institute of Signal Processing, Tampere University of Technology, Finland.

I wish to express my gratitude to Professor Jaakko Astola for making it possible for me start working on the transcription problem, for his help and advice during this work, and for his contribution in bringing expertise and motivated people to our lab from all around the world.

I am grateful to Jari Yli-Hietanen for his invaluable encouragement and support during the first couple of years of this work. Without him this thesis would probably not exist. I would like to thank all members, past and present, of the Audio Research Group for their part in making a motivating and enjoyable working community. Especially, I wish to thank Konsta Koppinen, Riitta Niemistö, Tuomas Virtanen, Antti Eronen, Vesa Peltonen, Jouni Paulus, Matti Ryynänen, Antti Rosti, Jarno Seppänen, and Timo Viitaniemi, whose friendship and good humour has made designing algorithms fun.

I wish to thank the staff of the Acoustic Laboratory of Helsinki University of Technology for their special help. Especially, I wish to thank Matti Karjalainen and Vesa Välimäki for setting an example to me both as researchers and as persons.

The financial support of the Tampere Graduate School in Information Science and Engineering (TISE), the Foundation of Emil Aaltonen, Tekniikan edistämissäätiö, and the Nokia Foundation is gratefully acknowledged.

I wish to thank my parents Leena and Tapani Klapuri for their encouragement on my path through the education system and my brother Harri for his advice in research work.

My warmest thanks go to my dear wife Mirva for her support, love, and understanding during the intensive stages of putting this work together.

I can never express enough gratitude to my Lord and Saviour, Jesus Christ, for being the foundation of my life in all situations. I believe that God has created us in his image and put into us a similar desire to create things – for example transcription systems in this context. However, looking at the nature, its *elegance* in the best sense that a mathematician uses the word, I have become more and more aware that Father is quite many orders of magnitude ahead in engineering, too.

God is faithfull, through whom you were called into fellowship
with his son, Jesus Christ our Lord. –1.COR. 1:9

Tampere, March 2004

Anssi Klapuri

Contents

List of publications

This thesis consists of the following publications and of some earlier unpublished results. The publications below are referred in the text as [P1], [P2], ..., [P6].

[P1] A. P. Klapuri, “Number theoretical means of resolving a mixture of several harmonic sounds,” In *Proc. European Signal Processing Conference*, Rhodos, Greece, 1998.

[P2] A. P. Klapuri, “Sound onset detection by applying psychoacoustic knowledge,” In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, Phoenix, Arizona, 1999.

[P3] A. P. Klapuri, “Multipitch estimation and sound separation by the spectral smoothness principle,” In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, Salt Lake City, Utah, 2001.

[P4] A. P. Klapuri and J. T. Astola, “Efficient calculation of a physiologically-motivated representation for sound,” In *Proc. 14th IEEE International Conference on Digital Signal Processing*, Santorini, Greece, 2002.

[P5] A. P. Klapuri, “Multiple fundamental frequency estimation based on harmonicity and spectral smoothness,” *IEEE Trans. Speech and Audio Proc.*, 11(6), 804–816, 2003.

[P6] A. P. Klapuri, A. J. Eronen, and J. T. Astola, “Automatic estimation of the meter of acoustic musical signals,” Tampere University of Technology, Institute of Signal Processing, Report 1–2004, Tampere, Finland, 2004.

Abbreviations

ACF	Autocorrelation function.
ASA	Auditory scene analysis.
CASA	Computational auditory scene analysis.
DFT	Discrete Fourier transform. Defined in (4.21) on page 38
EM	Expectation-maximization.
ERB	Equivalent rectangular bandwidth. Defined on page 33.
F0	Fundamental frequency. Defined on page 3.
FFT	Fast Fourier transform.
flex	Flatted-exponential (filter). Defined in (4.11) on page 35.
FWOC	Full-wave (odd) v^{th}-law compression. Defined on page 36.
HWR	Half-wave rectification. Defined on page 27.
IDFT	Inverse discrete Fourier transform.
MIDI	Musical Instrument Digital Interface. Explained on page 1.
MPEG	Moving picture experts group.
roex	Rounded-exponential (filter). Defined in (4.2) on page 33.
SACF	Summary autocorrelation function. Defined on page 28.
SNR	Signal-to-noise ratio.

1 Introduction

Transcription of music is here defined as the process of analyzing an acoustic musical signal so as to write down the parameters of the sounds that constitute the piece of music in question. Traditionally, written music uses *note symbols* to indicate the pitch, onset time, and duration of each sound to be played. The loudness and the applied musical instruments are not specified for individual notes but are determined for larger parts. An example of the traditional musical notation is shown in Fig. 1.

In a representational sense, music transcription can be seen as transforming an acoustic signal into a symbolic representation. However, written music is primarily a *performance instruction*, rather than a representation of music. It describes music in a language that a musician understands and can use to produce musical sound. From this point of view, music transcription can be viewed as discovering the "recipe", or, reverse-engineering the "source code" of a music signal. The applied notation does not necessarily need to be the traditional musical notation but any symbolic representation is adequate if it gives sufficient information for performing a piece using the available musical instruments. A guitar player, for example, often finds it more convenient to read *chord symbols* which characterize the note combinations to be played in a more general manner. In the case that an electronic synthesizer is used for resynthesis, a MIDI[1] file is an example of an appropriate representation.

A musical score does not only allow reproducing a piece of music but also making musically meaningful modifications to it. Changes to the symbols in a score cause meaningful changes to the music at a high abstraction level. For example, it becomes possible to change the arrangement (i.e., the way of playing and the musical style) and the instrumentation (i.e., to change, add, or remove instruments) of a piece. The relaxing effect of the sensomotoric exercise of performing and varying good music is quite a different thing than merely passively listening to a piece of music, as every amateur musician knows. To contribute to this kind of active attitude to music has been one of the driving motivations of this thesis.

Other applications of music transcription include

- *Structured audio coding*. A MIDI-like representation is extremely compact yet retains the identifiability and characteristics of a piece of music to an important degree. In structured audio coding, sound source parameters need to be encoded, too, but the bandwidth still stays around 2–3 kbit/s (see MPEG-4 document [ISO99]). An object-based representation is able to utilize the fact that music is redundant at many levels.
- *Searching musical information* based on e.g. the melody of a piece.
- *Music analysis*. Transcription tools facilitate the analysis of improvised music and the man-

Figure 1. An excerpt of a traditional musical notation (a score).

[1] Musical Instrument Digital Interface. A standard interface for exchanging performance data and parameters between electronic musical devices.

agement of ethnomusicological archives.

- *Music remixing* by changing the instrumentation, by applying effects to certain parts, or by selectively extracting certain instruments.
- *Interactive music systems* which generate an accompaniment to the singing or playing of a soloist, either off-line or in real-time [Rap01a, Row01].
- *Music-related equipment*, such as syncronization of light effects to a music signal.

A person without a musical education is usually not able to transcribe polyphonic music[1], in which several sounds are playing simultaneously. The richer is the polyphonic complexity of a musical composition, the more the transcription process requires musical ear training[2] and knowledge of the particular musical style and of the playing techniques of the instruments involved. However, skilled musicians are able to resolve even rich polyphonies with such an accuracy and flexibility that computational transcription systems fall clearly behind humans in performance.

Automatic transcription of polyphonic music has been the subject of increasing research interest during the last ten years. Before this, the topic was explored mainly by individual researchers. The transcription problem is in many ways analogous to that of automatic speech recognition, but has not received a comparable academic or commercial interest. Larger-scale research projects have been undertaken at Stanford University [Moo75,77, Cha82,86a,86b], University of Michigan [Pis79,86, Ste99], University of Tokyo [Kas93,95], Massachusetts Institute of Technology [Haw93, Mar96a, 96b], Tampere University of Technology [Kla98, Ero01, Vii03, Pau03a, Vir03, Ryy04], Cambridge University [Hai01, Dav03], and University of London [Bel03, Abd_]. Doctoral theses on the topic have been prepared at least by Moorer [Moo75], Piszczalski [Pis86], Maher [Mah89], Mellinger [Mel91], Hawley [Haw93], Godsmark [God98], Rossi [Ros98b], Sterian [Ste99], Bello [Bel03], and Hainsworth [Hai01, Hai_]. A more complete review and analysis of the previous work is presented in Chapter 5.

Despite the number of attemps to solve the problem, a practically applicable general-purpose transcription system does not exist at the present time. The most recent proposals, however, have achieved a certain degree of accuracy in transcribing limited-complexity polyphonic music [Kas95, Mar96b, Ste99, Tol00, Dav03, Bel03]. The typical limitations for the target signals are that the number of concurrent sounds is limited (or, fixed) and the interference of drums and percussive instruments is not allowed. Also, the relatively high error rate of the systems has reduced their practical applicability. Some degree of success for real-world music on CD recordings has been previously demonstrated by Goto [Got01]. His system aims at extracting the melody and the bass lines from complex music signals.

A few commercial transcription systems have been released [AKo01, Ara03, Hut97, Inn04, Mus01, Sev04] (see [Bui04] for a more comprehensive list). However, the accuracy of the programs has been very limited. Surprisingly, even the transcription of single-voice singing is not a solved problem, as indicated by the fact that the accuracy of the “voice-input” functionalities in score-writing programs is not comparable to humans (see [Cla02] for a comparative evaluation of available monophonic transcribers). *Tracking the pitch* of a monophonic musical pas-

[1] In this work, *polyphonic* refers to a signal where several sounds occur simultaneously. The word *monophonic* is used to refer to a signal where at most one note is sounding at a time. The terms *monaural signal* and *stereo signal* are used to refer to single-channel and two-channel audio signals, respectively.

[2] The aim of ear training in music is to develop the faculty of discriminating sounds, recognizing musical intervals, and playing music by ear.

sage is practically a solved problem but *quantization* of the continuous track of pitch estimates into note symbols with discrete pitch and timing has turned out to be a very difficult problem for some target signals, particularly for singing. Efficient use of musical knowledge is necessary in order to “guess” the score behind a performed pitch track [Vii03, Ryy04]. The general idea of an automatic music transcription system was patented in 2001 [Ale01].

1.1 Terminology

Some terms have to be defined before going any further. *Pitch* is a perceptual attribute of sounds, defined as the frequency of a sine wave that is matched to the target sound in a psychoacoustic experiment [Ste75]. If the matching cannot be accomplished consistently by human listeners, the sound does not have pitch [Har96]. *Fundamental frequency* is the corresponding physical term and is defined for periodic or nearly periodic sounds only. For these classes of sounds, fundamental frequency is defined as the inverse of the period. In ambiguous situations, the period corresponding to the perceived pitch is chosen.

A *melody* is a series of single notes arranged in a musically meaningful succession [Bro93b]. A *chord* is a combination of three or more simultaneous notes. A chord can be consonant or dissonant, depending on how harmonious are the pitch intervals between the component notes. *Harmony* refers to the part of musical art or science which deals with the formation and relations of chords [Bro93b]. *Harmonic analysis* deals with the structure of a piece of music with regard to the chords of which it consists.

The term *musical meter* has to do with rhythmic aspects of music. It refers to the regular pattern of strong and weak beats in a piece of music. Perceiving the meter can be characterized as a process of detecting moments of musical stress in an acoustic signal and filtering them so that underlying periodicities are discovered [Ler83, Cla99]. The perceived periodicities (*pulses*) at different time scales together constitute the meter. Meter estimation at a certain time scale is taking place for example when a person taps foot to music.

Timbre, or, *sound colour*, is a perceptual attribute which is closely related to the recognition of sound sources and answers the question “what something sounds like” [Han95]. Timbre is not explained by any simple acoustic property and the concept is therefore traditionally defined by exclusion: “timbre is the quality of a sound by which a listener can tell that two sounds of the same loudness and pitch are dissimilar” [ANS73]. The human timbre perception facility is very accurate and, consequently, sound synthesis is an important area of music technology [Roa96, Väl96, Tol98].

1.2 Decomposition of the music transcription problem

Automatic transcription of music comprises a wide area of research. It is useful to structurize the problem and to decomposing it into smaller and more tracktable subproblems. In this section, different strategies for doing this are proposed.

1.2.1 Modularity of music processing in the human brain

The human auditory system is the most reliable acoustic analysis tool in existence. It is therefore reasonable to learn from its structure and function as much as possible. *Modularity* of a certain kind has been observed in the human brain. In particular, certain parts of music cognition seem to be functionally and neuro-anatomically isolable from the rest of the auditory cog-

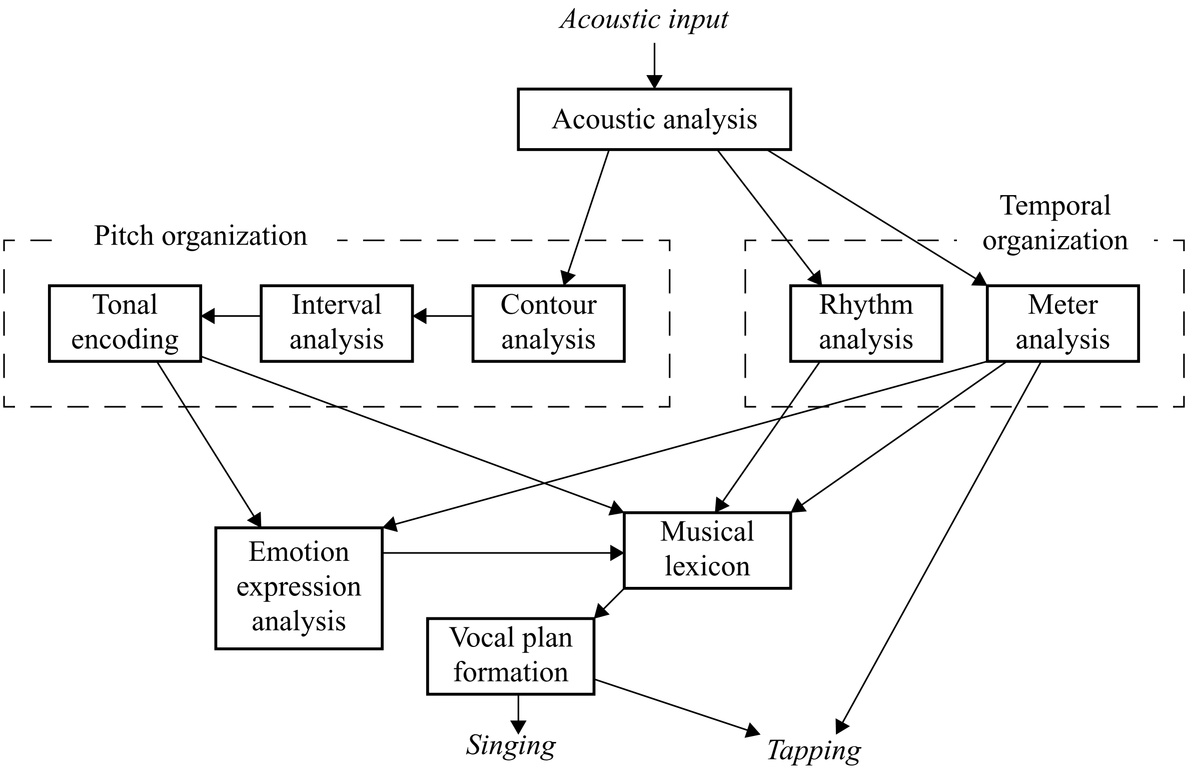

Figure 2. Functional modules of the music processing facility in the human brain as proposed by Peretz *et al.* (after [Per03]; only the parts related to music processing are reproduced here). The model has been derived from case studies of specific impairments of musical abilities in brain-damaged patients [Per01, 03]. See text for details.

nition [Per01,03, Zat02, Ter_]. There are two main sources of evidence: studies with brain-damaged patients and neurological imaging experiments in healthy subjects.

An accidental brain damage at the adult age may selectively affect musical abilities but not e.g. speech-related abilities, and vice versa. Moreover, studies of brain-damaged patients have revealed something about the internal structure of the music cognition system. Figure 2 shows the functional architecture that Peretz and colleagues have derived from case studies of specific music impairments in brain-damaged patients. The “breakdown pattern” of different patients was studied by representing them with specific music-cognition tasks, and the model in Fig. 2 was then inferred based on the assumption that a specific impairment may be due to a damaged processing component (box) or a broken flow of information (arrow) between components. The detailed line of argument underlying the model can be found in [Per01].

In Fig. 2, the *acoustic analysis* module is assumed to be common to all acoustic stimuli (not just music) and to perform segregation of sound mixtures into distinct sound sources. The subsequent two entities carry out *pitch organization* and *temporal organization*. These two are viewed as parallel and largely independent subsystems, as supported by studies of patients who suffer from difficulties to deal with pitch variations but not with temporal variations, or vice versa [Bel99, Per01]. In music performance or in perception, either of the two can be selectively lost [Per01]. The *musical lexicon* is characterized by Peretz *et al.* as containing representations of all the musical phrases a person has heard during his or her lifetime [Per03]. In some cases, a patient cannot recognize familiar music but can still process musical information otherwise adequately.

The main weakness of the studies with brain-damaged patients is that they are based on a relatively small number of cases. It is more common that an auditory disorder is global in the sense that it applies for all types of auditory events. The model in Fig. 2, for example, has been inferred based on approximately thirty patients only. This is particularly disturbing because the model in Fig. 2 corresponds “too well” to what one would predict based on the established tradition in music theory and music analysis [Ler83, Deu99].

Neuroimaging experiments in healthy subjects provide another important source of evidence concerning the modularity and localization of the cognitive functions. In particular, it is known that speech sounds and higher-level speech information are preferentially processed in the left auditory cortex, whereas musical sounds are preferentially processed in the right auditory cortex. Interestingly, however, when musical tasks involve specifically processing of temporal information (temporal synchrony or duration), the processing is associated with the *left* hemisphere [Zat02, Per01]. Also, Bella et al. suggest that in music, pitch organization takes place primarily in the right hemisphere and the temporal organization recruits more the left auditory cortex [Bel99]. As concluded both in [Zat02] and in [Ter_], the relative asymmetry between the two hemispheres is not bound to informational sound content but to the acoustic characteristics of the signals. Rapid *temporal information* is more common in speech, whereas accurate processing of *spectral and pitch information* is more important in music.

Zatorre *et al*. used functional imaging (positron emission tomography) to examine the response of human auditory cortex to spectral and temporal variation [Zat01]. In the experiment, the amount of temporal and spectral variation in the acoustic stimulus was parametrized. As a result, responses to the increase in temporal variation were weighted towards the left, while responses to the increase in melodic/spectral variation were weighted towards the right. In [Zat02], the authors review different types of evidence which support the conclusion that there is a relative specialization of the auditory cortices in the two hemispheres so that the left auditory cortex is specialized to a better temporal resolution and the right auditory cortex to a better spectral resolution. Tervaniemi *et al.* review additional evidence from imaging experiments in healthy adult subjects and come basically to the same conclusion [Ter_].

In computational transcription systems, rhythm and pitch have most often been analyzed separately and using different data representations [Kas95, Mar96b, Dav03, Got96,00]. Typically, a better time resolution is applied in rhythm analysis and a better frequency resolution in pitch analysis. Based on the above studies, this seems to be justified and not only a technical artefact. The overall structure of transcription systems is often determined by merely pragmatic considerations. For example, temporal segmentation is performed prior to pitch analysis in order to allow the sizing and positioning of analysis frames in pitch analysis, which is typically the computationally more demanding stage [Kla01a, Dav03].

1.2.2 Role of internal models

Large-vocabulary speech recognition systems are critically dependent on *language models*, which represent linguistic knowledge about speech signals [Rab93, Jel97, Jur00]. The models can be of very primitive nature, for example merely tabulating the occurrence probabilities of different three-word sequences (*N*-gram models), or more complex, implementing part-of-speech tagging of words and syntactic inference within sentences.

Musicological information is equally important for the automatic transcription of polyphonically rich musical material. The probabilities of different notes to occur concurrently or sequentially can be straightforwardly estimated, since large databases of written music exist in an electronic format [Kla03a, Cla04]. More complex rules governing music are readily available in the theory of music and composition and some of this information has already been quantified to computational models [Tem01].

Thus another way of structurizing the transcription problem is according to the *sources of knowledge* available. Pre-stored internal models constitute a source of information in addition to the incoming acoustic waveform. The uni-directional flow of information in Fig. 2 is not realistic in this sense but represents a *data-driven* view where all information flows bottom-up: information is observed in an acoustic waveform, combined to provide meaningful auditory cues, and passed to higher level processes for further interpretation. *Top-down processing* utilizes internal high-level models of the input signals and prior knowledge concerning the properties and dependencies of the sound events in it [Ell96]. In this approach, information also flows top-down: analysis if performed in order to justify or cause a change in the predictions of an internal model.

Some transcription systems have applied musicological models or sound source models in the analysis [Kas95, Mar96b, God99], and some systems would readily enable this by replacing certain prior distributions by musically informed ones [Got01, Dav03]. Temperley has proposed a very comprehensive rule-based system for modelling the cognition of basic musical structures, taking an important step towards quantifying the higher-level rules that govern musical structures [Tem01]. More detailed introduction to the previous work is presented in Chapter 5.

Utilizing diverse sources of knowledge in the analysis raises the issue of how to integrate the information meaningfully. In automatic speech recognition, probabilistic methods have been very successful in this respect [Rab93, Jel97, Jur00]. Statistical methods allow representing uncertain knowledge and learning from examples. Also, probabilistic models have turned out to be a very fundamental “common ground” for integrating knowledge from diverse sources. This will be discussed in Sec. 5.2.3.

1.2.3 Mid-level data representations

Another efficient way of structurizing the transcription problem is through so-called *mid-level representations*. Auditory perception may be viewed as a hierarchy of representations from an acoustic signal up to a conscious percept, such as a comprehended sentence of a language [Ell95,96]. In music transcription, a musical score can be viewed as a high-level representation. Intermediate abstraction level(s) are indispensable since the symbols of a score are not readily visible in the acoustic signal (transcription based on the acoustic signal directly has been done in [Dav03]). Another advantage of using a well-defined mid-level representation is that it structurizes the system, i.e., acts as an “interface” which separates the task of computing the mid-level representation from the higher-level inference that follows.

A fundamental mid-level representation in human hearing is the signal in the auditory nerve. Whereas we know rather little about the exact mechanisms of the brain, there is much wider consensus about the mechanisms of the physiological and more peripheral parts of hearing. Moreover, precise *auditory models* exist which are able to approximate the signal in the auditory nerve [Moo95a]. This is a great advantage, since an important part of the analysis takes place already at the peripheral stage.

The mid-level representations of different music transcription systems are reviewed in Chapter 5 and a summary is presented in Table 7 on page 71. Along with auditory models, a representation based on *sinusoid tracks* has been a very popular choice. This reprerentation is introduced in Sec. 5.2.1. An excellent review of the mid-level representations for audio content analysis can be found in [Ell95].

1.2.4 How do humans transcribe music?

One more approach to structurize the transcription problem is to study the *conscious transcription process* of human musicians and to inquire their transcription strategies. The aim of this is to determine the sequence of actions or processing steps that leads to the transcription result. Also, there are many concrete questions involved. Is a piece processed in one pass or listened through several times? What is the duration of an elementary audio chunk that is taken into consideration at a time? And so forth.

Hainsworth has conducted interviews with musicians in order to find out how they transcribe [Hai02, personal communication]. According to his report, the transcription proceeds sequentially towards increasing detail. First, the global structure of a piece is noted in some form. This includes an implicit detection of style, instruments present, and rhythmic context. Secondly, the most dominant melodic phrases and bass lines are transcribed. In the last phase, the inner parts are examined. These are often heard out only with the help from the context generated at the earlier stages and by applying the priorly gained musical knowledge of the individual. Chordal context was often cited to be used as an aid to transcribing the inner parts. This suggests that harmonic analysis is an early part of the process. About 50% of the respondees used musical instrument as an aid, mostly as a means of reproducing notes for comparison with the original (most others were able to do this in their heads via “mental rehearsal”).

In [Hai02], Hainsworth points out certain characteristics of the above-described method. First, the process is sequential rather than concurrent. Secondly, it relies on the human ability to attend to certain parts of a sonic spectrum while selectively ignoring others. Thirdly, information from the early stages is used to inform later ones. The possibility of feedback from the later stages to the lower levels should be considered [Hai02].

1.3 Scope and purpose of the thesis

This thesis is concerned with the automatic transcription of the harmonic and melodic parts of real-world music signals. Detecting or labeling the sounds of percussive (drum) instruments is not attempted but an interested reader is referred to [Pau03a,b, Gou01, Fiz02, Zil02]. However, the presence of drum instruments is allowed. Also, the number of concurrent sounds is not restricted. Automatic recognition of musical instruments is not addressed in this thesis but an interested reader is referred to [Mar99, Ero00,01, Bro01].

Algorithms are proposed that address two different subproblems of music transcription. The main part of this thesis is dedicated to what is considered to be the core of the music transcription problem: *multiple fundamental frequency (F0) estimation*. The term refers to the estimation of the fundamental frequencies of several concurrent musical sounds. This corresponds most closely to the “acoustic analysis” module in Fig. 2. Two different algorithms are proposed for multiple-F0 estimation. One is derived from the principles of human auditory perception and is described in Chapter 4. The other is oriented towards more pragmatic problem solving and is introduced in Chapter 6. The latter algorithm has been originally proposed in [P5].

Musical meter estimation is the other subproblem addressed in this work. This corresponds to the “meter analysis” module in Fig. 2. Contrary to the flow of information in Fig. 2, however, the meter estimation algorithm does not utilize the analysis results of the multiple-F0 algorithm. Instead, the meter estimator takes the raw acoustic signal as input and uses a filterbank emulation to perform time-frequency analysis. This is done for two reasons. First, the multiple-F0 estimation algorithm is computationally rather complex whereas meter estimation as such can be done much faster than in real-time. Secondly, meter estimation benefits of a relatively good time resolution (23ms Fourier transform frame is used in the filterbank emulation) whereas multiple-F0 estimator works adequately for 46ms frames or longer. The drawbacks of this basic decision are discussed in Sec. 2.3.

Musical meter estimation and multiple-F0 estimation are complementary to each other. The musical meter estimator generates a temporal framework which can be used to divide the input signal into musically meaningful temporal segments. Also, musical meter can be used to perform *time quantization*, since musical events can be assumed to begin and end at segment boundaries. The multiple-F0 estimator, in turn, indicates which notes are active at each time but is not able to decide the exact beginning or end times of individual note events. Imagine a time-frequency plane where time flows from left to right and different F0s are arranged in ascending order on the vertical axis. On top of this plane, the multiple-F0 estimator produces horizontal lines which indicate the probabilities of different notes to be active as a function of time. The meter estimator produces a framework of vertical “grid lines” which can be used to decide the onset and offset times of discrete note events.

Metrical information can also be utilized in adjusting the positions and lengths of the analysis frames applied in multiple-F0 estimation. This has the practical advantage that multiple-F0 estimation can be performed for a number of discrete segments only and does not need to be performed in a continuous manner for a larger number of overlapping time frames. Also, by positioning multiple-F0 analysis frames according to metrical boundaries minimizes the interference from sounds that do not occur concurrently, since event beginnings and ends are likely to coincide with the metrical boundaries. This strategy was used in producing the transcription demonstrations available at [Kla03b].

The focus of this thesis is in bottom-up signal analysis methods. Musicological models and top-down processing are not considered, except that the proposed meter estimation method utilizes some primitive musical knowledge in performing the analysis. The title of this work, “*signal processing methods for...*”, indicates that the emphasis is laid on the acoustic signal analysis part. The musicological models are more oriented towards statistical methods [Vii03, Ryy04], rule-based inference [Tem01], or artificial intelligence techniques [Mar96a].

1.3.1 Relation to auditory modeling

A lot of work has been carried out to model the human auditory system [Moo95a, Zwi99]. Unfortunately, important parts of the human hearing are located in the central nervous system and can be studied only indirectly. *Psychoacoustics* is the science that deals with the perception of sound. In a psychoacoustic experiment, the relationships between an acoustic stimulus and the resulting subjective sensation is studied by presenting specific tasks or questions to human listeners [Ros90, Kar99a].

The aim of this thesis is to develop practically applicable solutions to the music transcription problem and *not to propose models of the human auditory system*. The proposed methods are ultimately justified by their practical efficiency and not by their psychoacoustic plausibility or the ability to model the phenomena in human hearing. The role of auditory modeling in this work is to help towards the practical goal of solving the transcription problem. At the present time, the only reliable transcription system we have is the ears and the brain of a trained musician.

Psychoacoustically motivated methods have turned out to be among the most successful ones in audio content analysis. This is why the following chapters make an effort to examine the proposed methods in the light of psychoacoustics. It is often difficult to see what is an important processing principle in human hearing and what is merely an unimportant detail. Thus, departures from psychoacoustic principles are carefully discussed.

It is important to recognize that a musical notation is primarily concerned with the (mechanical) sound production and not with perception. As pointed out by Scheirer in [Sch96], it is not likely that note symbols would be the representational elements in music perception or that there would be an innate transcription facility in the brain. The very task of music transcription differs fundamentally from that of trying the predict the response that the music arises in a human listener. For the readers interested in the latter problem, the doctoral thesis of Scheirer is an excellent starting point [Sch00].

Ironically, the perceptual intentions of music directly oppose those of its transcription. Bregman pays attention to the fact that music often wants the listener to accept simultaneous sounds as a single coherent sound with its own striking properties. The human auditory system has a tendency to segregate a sound mixture to the physical sources, but orchestration is often called upon to oppose these tendencies [Bre90,p.457–460]. For example, synchronous onset times and harmonic pitch relations are used to knit together sounds so that they are able to represent higher-level forms that could not be expressed by the atomic sounds separately. Because the human perception handles such entities as a single object, music may recruit a large number of harmonically related sounds (that are hard to transcribe or separate) without adding too much complexity to a human listener.

1.4 Main results of the thesis

The original contributions of this thesis can be found in Publications [P1]–[P6] and in Chapter 4 which contains earlier unpublished results. The main results are briefly summarized below.

1.4.1 Multiple-F0 estimation system I

Publications [P1], [P3], and [P5] constitute an entity. Publication [P5] is partially based on the results derived in [P1] and [P3].

In [P1], a method was proposed to deal with *coinciding frequency components* in mixture signals. These are partials of a harmonic sound that coincide in frequency with the partials of other sounds and thus overlap in the spectrum. The main results were:

- An algorithm was derived that identifies the partials which are the *least* likely to coincide.
- A weighted order-statistical filter was proposed in order to filter out coinciding partials when a sound is being observed. The sample selection probabilities of different harmonic partials were set according to their estimated reliability.
- The method was applied to the transcription of polyphonic piano music.

In [P3], a processing principle was proposed for finding the F0s and separating the spectra of concurrent musical sounds. The principle, *spectral smoothness,* was based on the observation that the partials of a harmonic sound are usually close to each other in amplitude within one critical band. In other words, the spectral envelopes of real-world sounds tend to be smooth as a function of frequency. The contributions of Publication [P3] are the following.

- Theoretical and empirical evidence was presented to show the importance of the smoothness principle in resolving sound mixtures.
- Sound separation is possible (to a certain degree) without *a priori* knowledge of the sound sources involved.
- Based on the known properties of the peripheral hearing in humans [Med91], it was shown that the spectral smoothing takes a specific form in the human hearing.
- Three algorithms of varying complexity were described which implement the new principle.

In [P5], a method was proposed for estimating the F0s of concurrent musical sounds within a single time frame. The method is “complete” in the sense that it included mechanisms for suppressing additive noise (drums) and for estimating the number of concurrent sounds in the analyzed signal. The main results were:

- Multiple-F0 estimation can be performed reasonably accurately (compared with trained musicians) within a single time frame, without long-term temporal features.
- The taken iterative estimation and cancellation approach makes it possible to detect at least a couple of the most prominent F0s even in rich polyphonies.
- An algorithm was proposed which uses the frequency relationships of simultaneous spectral components to group them to sound sources. Ideal harmonicity was not assumed.
- A method was proposed for suppressing the noisy signal components due to drums.
- A method was proposed for estimating the number of concurrent sounds in input signals.

1.4.2 Multiple-F0 estimation system II

Publication [P4] and Chapter 4 of this thesis constitute an entity. Computational efficiency of the method proposed in Chapter 4 is in part based on the results in [P4].

Publication [P4] is concerned with a perceptually-motivated representation for sound, called the *summary autocorrelation function* (SACF). An algorithm was proposed which calculates an approximation of the SACF in the frequency domain. The main results were:

- Each individual spectral bin of the Fourier transform of the SACF can be computed in $O(K)$ time, i.e., in a time which is proportional to the analysis frame length K, given the complex Fourier transform of the wideband input signal.
- The number of distinct subbands in calculating the SACF does not need to be defined. The algorithm implements a model where one subband is centered on each discrete Fourier spectrum sample, thus approaching a continuous density of subbands (in Chapter 4, for example, 950 subbands are used). The bandwidths of the subbands need not be changed.

In Chapter 4 of this thesis, a novel multiple-F0 estimation method is proposed. The method is derived from the known properties of the human auditory system. More specifically, it is assumed that the peripheral parts of hearing can be modelled by (i) a bank of bandpass filters and (ii) half-wave rectification (HWR) and compression of the time-domain signals at the subbands. The main results are:

- A practically applicable multiple-F0 estimation method is derived. In particular, the method works reasonably accurately in short analysis frames.

- It is shown that half-wave rectification at subbands amounts to the combined use of time-domain periodicity and frequency-domain periodicity for F0 extraction.
- Higher-order (unresolved) partials of a harmonic sound can be processed collectively. Estimation or detection of individual higher-order partials is not robust and should be avoided.

1.4.3 Musical meter estimation and sound onset detection

Publication [P2] proposed a method for *onset detection*, i.e., for the detection of the beginnings of discrete sound events in acoustic signals. The main contributions were:

- A technique was described to cope with sounds that exhibit onset imperfections, i.e., the amplitude envelope of which does not rise monothonically.
- A psychoacoustic model of intensity coding was applied in order to find parameters which allow robust one-by-one detection of onsets for a wide range of input signals.

In [P6], a method for musical-meter analysis was proposed. The analysis was performed jointly at three different time scales: at the temporally atomic *tatum* pulse level, at the *tactus* pulse level which corresponds to the tempo of a piece, and at the *musical measure* level. The main contributions were:

- The proposed method works robustly for different types of music and improved over two state-of-the-art reference methods in simulations.
- A technique was proposed for measuring the degree of musical accent as a function of time. The technique was partially based on the ideas in [P2].
- The paper confirmed an earlier result of Scheirer [Sch98] that comb-filter resonators are suitable for metrical pulse analysis. Four different periodicity estimation methods were evaluated and, as a result, comb-filters were the best in terms of simplicity vs. performance.
- Probabilistic models were proposed to encode prior musical knowledge regarding well-formed musical meters. The models take into account the dependencies between the three pulse levels and implement temporal tying between successive meter estimates.

1.5 Outline of the thesis

This thesis is organized as follows. Chapter 2 considers the musical meter estimation problem. A review of the previous work in this area is presented. This is followed by a short introduction to Publication [P6] where a novel method for meter estimation is proposed. Technical details and simulation results are not described but can be found in [P6]. A short conclusion is given to discuss the achieved results and future work.

Chapter 3 introduces harmonic sounds and the different approaches that have been taken to the estimation of the fundamental frequency of isolated musical sounds. A model of the human pitch perception is introduced and its benefits from the point of view of F0 estimation are discussed.

Chapter 4 elaborates the pitch model introduced in Chapter 3 and, based on that, proposes a previously unpublished method for estimating the F0s of multiple concurrent musical sounds. Also, Chapter 4 presents background material which serves as an introduction to [P4].

Chapter 5 reviews previous approaches to multiple-F0 estimation. Because this is the core problem in music transcription, the chapter can also be seen as an introduction to the potential approaches to music transcription in general.

Chapter 6 serves as an introduction to the other, problem-solving oriented method for multiple-

F0 estimation. The method has been originally published in [P5] and is “complete” in the sense that it includes mechanisms for suppressing additive noise and for estimating the number of concurrent sounds in the input signal. These are needed in order to process real-world music signals. Introduction to Publications [P1] and [P3] is given in Sec. 6.4. An epilogue in Sec. 6.5 presents some criticism of the method.

Chapter 7 summarizes the main conclusions and discusses future work.

2 Musical meter estimation

This chapter reviews previous work on musical meter estimation and serves as an introduction to Publication [P6]. The concept *musical meter* was defined in Sec. 1.1. Meter analysis is an essential part of understanding music signals and an innate cognitive ability of humans even without musical education. Virtually anybody is able to clap hands to music and it is not unusual to see a two-year old child swaying in time with music. From the point of view of music transcription, meter estimation amounts to *temporal segmentation* of music according to certain criteria.

Musical meter is a hierarchical structure, consisting of pulse sensations at different levels (time scales). In this thesis, three metrical levels are considered. The most prominent level is the *tactus*, often referred to as the foot tapping rate or the beat. Following the terminology of [Ler83], we use the word *beat* to refer to the individual elements that make up a pulse. A musical meter can be illustrated as in Fig. 3, where the dots denote beats and each sequence of dots corresponds to a particular pulse level. By the *period* of a pulse we mean the time duration between successive beats and by *phase* the time when a beat occurs with respect to the beginning of the piece. The *tatum* pulse has its name stemming from "temporal atom" [Bil93]. The period of this pulse corresponds to the shortest durational values in music that are still more than incidentally encountered. The other durational values, with few exceptions, are integer multiples of the tatum period and onsets of musical events occur approximately at a tatum beat. The *musical measure* pulse is typically related to the harmonic change rate or to the length of a rhythmic pattern. Although sometimes ambiguous, these three metrical levels are relatively well-defined and span the metrical hierarchy at the aurally most important levels. *Tempo* of a piece is defined as the rate of the tactus pulse. In order that a meter would make sense musically, the pulse periods must be slowly-varying and, moreover, each beat at the larger levels must coincide with a beat at all the smaller levels.

The concept *phenomenal accent* is important for meter analysis. Phenomenal accents are events that give emphasis to a moment in music. Among these are the beginnings of all discrete sound events, especially the onsets of long pitch events, sudden changes in loudness or timbre, and harmonic changes. Lerdahl and Jackendoff define the role of phenomenal accents in meter perception compactly by saying that "the moments of musical stress in the raw signal serve as cues from which the listener attempts to extrapolate a regular pattern" [Ler83,p.17].

Automatic estimation of the meter alone has several applications. A temporal framework facilitates the cut-and-paste operations and editing of music signals. It enables synchronization with light effects, video, or electronic instruments, such as a drum machine. In a disc jockey application, metrical information can be used to mark the boundaries of a rhythmic loop or to

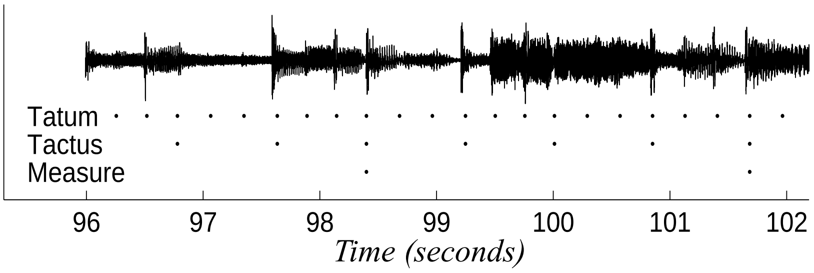

Figure 3. A musical signal with three metrical levels illustrated (reprinted from [P6]).

synchronize two or more percussive audio tracks. Meter estimation for symbolic (MIDI) data is required in time *quantization*, an indispensable subtask of score typesetting from keyboard input.

2.1 Previous work

The work on automatic meter analysis originated from algorithmic models which tried to explain how a human listener arrives at a particular metrical interpretation of a piece, given that the meter is not explicitly spelled out in music [Lee91]. The early models performed meter estimation for symbolic data, presented as an artificial impulse pattern or as a musical score [Ste77, Lon82, Lee85, Pov85]. In brief, all these models can be seen as being based on a *set of rules* that are used to define what makes a musical accent and to infer the most natural meter. The rule system proposed by Lerdahl and Jackendoff in [Ler83] is the most complete, but is described in verbal terms only. An extensive comparison of the early models has been given by Lee in [Lee91], and later augmented by Desain and Honing in [Des99].

Table 1 lists characteristic attributes of more recent meter analysis systems. The systems can be classified into two main categories according to the *type of input* they process. Some algorithms are designed for symbolic (MIDI) input whereas others process acoustic signals. The column, “evaluation material”, gives a more specific idea of the musical material that the systems have been tested on. Another defining characteristic of different systems is the *aim* of the meter analysis. Many algorithms do not analyze meter at all time scales but at the tactus level only. Some others produce useful side-information, such as quantization of the onset and offset times of musical events. The columns “approach”, “mid-level representation” and “computation” in Table 1 attempt to summarize the technique that is used to achieve the analysis result. More or less arbitrarily, three different approaches are discerned, one based on a set of rules, another employing a probabilistic model, and the third deriving the analysis methods mainly from the signal processing domain. Mid-level representations refer to the data representations that are used between the input and the final analysis result. The column “computation” summarizes the strategy that is applied to search the correct meter among all possible meters.

2.1.1 Methods designed primarily for symbolic input (MIDI)

Rosenthal has proposed a system which processes realistic piano performances in the form of MIDI files. His system attempted to emulate the human rhythm perception, including meter perception [Ros92]. Notable in his approach is that other auditory functions are taken into account, too. During a preprocessing stage, notes are grouped into melodic streams and chords, and this information is utilized later on. Rosenthal applied a set of rules to rank and prune competing meter hypotheses and conducted a beam search to track multiple hypotheses through time. The beam-search strategy was originally proposed for pulse tracking by Allen and Dannenberg in [All90].

Parncutt has proposed a detailed model of meter perception based on systematic listening tests [Par94]. His algorithm computes the salience (weigth) of different metrical pulses based on a quantitative model for phenomenal accents and for pulse salience.

Apart from the rule-based models, a straightforward signal-processing oriented approach was taken by Brown who performed metrical analysis of musical scores using the autocorrelation function [Bro93a]. The scores were represented as a time-domain signal (sampling rate

Table 1: Characteristics of some meter estimation systems

Reference	Input	Aim	Approach	Mid-level representation	Computation	Evaluation material
Rosenthal, 1992	MIDI	meter, time quantization	Rule-based, model auditory organization	At a preprocessing stage, notes are grouped into streams and chords	Multiple-hypothesis tracking (beam search)	92 piano performances
Brown, 1993	score	meter	DSP	Initialize a signal with zeros, then assign note-duration values at their onset times	Autocorrelation function (only periods were being estimated)	19 classical scores
Large, Kolen, 1994	MIDI	meter	DSP	Initialize a signal with zeros, then assign unity values at note onsets	Network of oscillators (period and phase locking)	A few example analyses; straightforward to reimplement
Parncutt, 1994	score	meter, accent modeling	Rule-based, based on listening tests	Phenomenal accent model for individual events (event parameters: length, loudness, timbre, pitch)	Match an isochronous pattern to accents	Artificial synthesized patterns
Temperley, Sleator, 1999	MIDI	meter, time quantization	Rule-based	Apply discrete time-base, assign each event to the closest 35ms time-frame	Viterbi; "cost functions" for event occurrence, event length, meter regularity	Example analyses; all music types; source code available
Dixon, 2001	MIDI, audio	tactus	Rule-based, heuristic	MIDI: parameters of MIDI-events. Audio: compute overall amplitude envelope, then extract onset *times*	First find periods using IOI histogram, then phases with multiple-agents (beam search)	222 MIDI files (expressive music); 10 audio files (sharp attacks); source code available
Raphael, 2001	MIDI, audio	tactus, time quantization	Probabilistic generative model	Only onset *times* are used	Viterbi; MAP estimation	Two example analyses; expressive performances
Cemgil, Kappen, 2003	MIDI	tactus, time quantization	Probabilistic generative model	Only onset *times* are used	Sequential Monte Carlo methods; balance score complexity vs. tempo continuity	216 polyphonic piano performances of 12 Beatles songs; clave pattern
Goto, Muraoka, 1995, 1997	audio	meter	DSP	Fourier spectra, onset components (time, reliability, frequency range)	Multiple tracking agents (beam search); IOI histogram for periodicity analysis; pre-stored drum patterns used in (1995)	85 pieces; pop music; 4/4 time signature
Scheirer, 1998	audio	tactus	DSP	Amplitude-envelope signals at six subbands	First find periods using a bank of comb filters, then phases based on filter states	60 pieces with "strong beat"; all music types; source code available
Laroche, 2001	audio	tactus, swing	Probabilistic	Compute overall "loudness" curve, then extract onset times and weights	Maximum-likelihood estimation; exhaustive search	Qualitative report; music with constant tempo and sharp attacks
Sethares, Staley, 2001	audio	meter	DSP	RMS-energies at 1/3-octave subbands	Periodicity transform	A few examples; music with constant tempo
Gouyon *et al.*, 2002	audio	tatum	DSP	Compute overall amplitude envelope, then extract onsets times and weights	First find periods (IOI histogram), then phases by matching isochronous pattern	57 drum sequences of 2–10 s. in duration; constant tempo
Klapuri *et al.*, 2003	audio	meter	DSP, probabilistic back-end	Degree of accentuation as a function of time at four frequency ranges	First find periods (bank of comb filters, Viterbi back-end), then phases using filter states and rhythmic pattern matching	474 audio signals; all music types

200Hz), where each individual note was represented as an impulse at the position of the note onset time and weighted by the duration of the note. Pitch information was not used. Large and Kolen associated meter perception with *resonance* and proposed an “entrainment” oscillator which adjusts its period and phase to an incoming pattern of impulses, located at the onsets of musical events [Lar94].

As a part of a larger project of modeling the cognition of basic musical structures, Temperley and Sleator proposed a meter estimation algorithm for arbitrary MIDI files [Tem99,01]. The algorithm was based on implementing the preference rules verbally described in [Ler83], and produced the whole metrical hierarchy as output. Dixon proposed a rule-based system to track the tactus pulse of expressive MIDI performances [Dix01]. Also, he introduced a simple onset detector to make the system applicable for audio signals. The methods works quite well for MIDI files of all types but has problems with audio files which do not contain sharp attacks. The source codes of both Temperley’s and Dixon’s systems are publicly available for testing.

Cemgil and Kappen developed a *probabilistic generative model* for the event times in expressive musical performances [Cem01, 03]. They used the model to infer a hidden continuous tempo variable and quantized ideal note onset times from observed noisy onset times in a MIDI file. Tempo tracking and time quantization were performed simultaneously so as to balance the smoothness of tempo deviations versus the complexity of the resulting quantized score. The model is very elegant but has the drawback that it processes only the onset *times* of events, ignoring duration, pitch, and loudness information. In many ways similar Bayesian model has been independently proposed by Raphael who has also demonstrated its use for acoustic input [Rap01a,b].

2.1.2 Methods designed for acoustic input

Goto and Muraoka were the first to present a meter-tracking system which works to a reasonable accuracy for audio signals [Got95,97a]. Only popular music with 4/4 time signature was considered. The system operates in real time and is based on an architecture where multiple agents track alternative meter hypotheses. Beat positions at the larger levels were inferred by detecting certain drum sounds [Got95] or chord changes [Got97]. Gouyon *et al*. proposed a system for estimating the tatum pulse in percussive audio tracks with constant tempo [Gou02]. The authors computed an inter-onset interval histogram and applied the two-way mismatch method of Maher [Mah94] to find the tatum (“temporal atom”) which best explained multiple harmonic peaks in the histogram. Laroche used a straightforward probabilistic model to estimate the tempo and swing[1] of audio signals [Lar01]. Input to the model was provided by an onset detector which was based on differentiating an estimated “overall loudness” curve.

Scheirer proposed a method for tracking the tactus pulse of music signals of any kinds, provided that they had a “strong beat” [Sch98]. Important in Scheirer’s approach was that he did not detect discrete onsets or sound events as a middle-step, but performed periodicity analysis directly on the half-wave rectified differentials of subband power envelopes. Periodicity at each subband was analyzed using a bank of comb-filter resonators. The source codes of Scheirer’s system are publicly available for testing. Since 1998, an important way to categorize acoustic-input meter estimators has been to determine whether the systems extract discrete events or

[1] *Swing* is a characteristic of musical rhythms most commonly found in jazz. Swing is defined in [Lar01] as a systematic slight delay of the second and fourth quarter-beats.

onset times as a middle-step or not. The meter estimator of Sethares and Staley is in many ways similar to Scheirer's method, with the difference that a periodicity transform was used for periodicity analysis instead of a bank of comb filters [Set01].

2.1.3 Summary

To summarize, most of the earlier work on meter estimation has concentrated on symbolic (MIDI) data and typically analyzed the tactus pulse only. Some of the systems ([Lar94], [Dix01], [Cem03], [Rap01b]) can be immediately extended to process audio signals by employing an onset detector which extracts the beginnings of discrete acoustic events from an audio signal. Indeed, the authors of [Dix01] and [Rap01b] have introduced an onset detector themselves. Elsewhere, onset detection methods have been proposed that are based on using an auditory model [Moe97], subband power envelopes [P2], support vector machines [Dav02], neural networks [Mar02], independent component analysis [Abd03], or complex-domain unpredictability [Dux03]. However, if a meter estimator has been originally developed for symbolic data, the extended system is usually not robust to diverse acoustic material (e.g. classical vs. rock music) and cannot fully utilize the acoustic cues that indicate phenomenal accents in music signals.

There are a few basic problems that a meter estimator needs to address to be successful. First, the degree of musical accentuation as a function of time has to be measured. In the case of audio input, this has much to do with the initial time-frequency analysis and is closely related to the problem of onset detection. Some systems measure accentuation in a continuous manner [Sch98, Set01], whereas others extract discrete events [Got95,97, Gou02, Lar01]. Secondly, the periods and phases of the underlying metrical pulses have to be estimated. The methods which detect discrete events as a middle step have often used inter-onset interval histograms for this purpose [Dix01, Got95,97, Gou02]. Thirdly, a system has to choose the metrical level which corresponds to the tactus or some other specially designated pulse level. This may take place implicitly, or by using a prior distribution for pulse periods [Par94], or by applying rhythmic pattern matching [Got95]. Tempo halving or doubling is a symptom of failing to do this.

2.2 Method proposed in Publication [P6]

The aim of the method proposed in [P6] is to estimate the meter of acoustic musical signals at three levels: at the tactus, tatum, and measure-pulse levels. The target signals are not restricted to any particular music type but all the main genres, including classical and jazz music, are represented in the validation database.

An overview of the method is shown in Fig. 4. For the time-frequency analysis part, a new technique is proposed which aims at measuring the degree of accentuation in music signals. The technique is robust to diverse acoustic material and can be seen as a synthesis and generalization of two earlier state-of-the-art methods [Got95] and [Sch98]. In brief, preliminary time-frequency analysis is conducted using a quite large number $b_0 > 20$ of subbands and by measuring the degree of spectral change at these channels. Then, adjacent bands are combined to arrive at a smaller number $3 \le c_0 \le 5$ of "registral accent signals" for which periodicity analysis is carried out. This approach has the advantage that the frequency resolution suffices to detect harmonic changes but periodicity analysis takes place at wider bands. Combining a certain number of adjacent bands prior to the periodicity analysis improves the analysis accuracy. Interestingly, neither combining all the channels before periodicity analysis, $c_0 = 1$, nor ana-

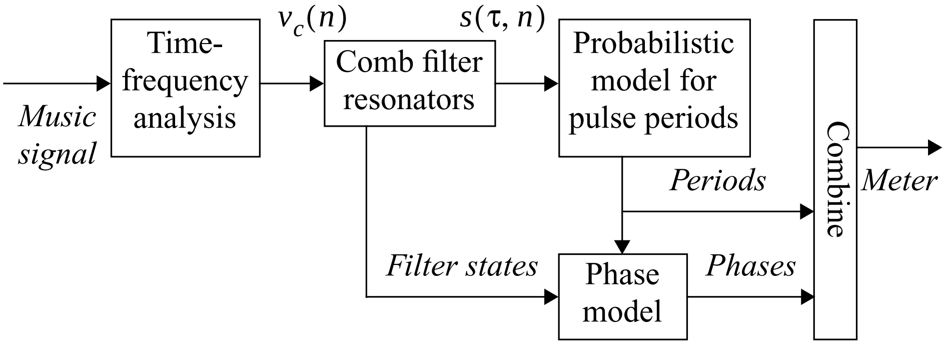

Figure 4. Overview of the meter estimation method. The two intermediate data representations are registral accent signals $v_c(n)$ at band c and metrical pulse strengths $s(\tau, n)$ for resonator period τ at time n. (Reprinted from [P6].)

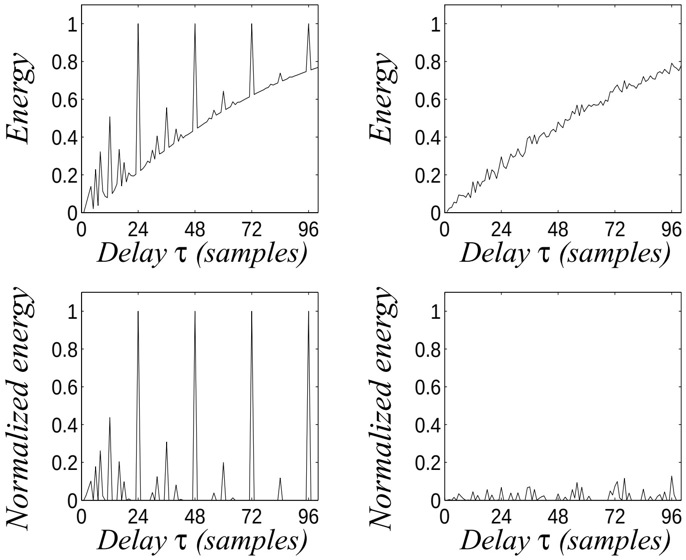

Figure 5. Output energies of comb filter resonators as a function of their feedback delay (period) τ. The energies are shown for an impulse train with a period-length 24 samples (left) and for a white noise signal (right). Upper panels show the raw output energies and the lower panels the energies after a specific normalization. (Reprinted from [P6].)

lyzing periodicity at all channels, $c_0 = b_0$, is an optimal choice but using a large number of bands in the preliminary time-frequency analysis (we used $b_0 = 36$) and three or four registral channels c_0 leads to the most reliable analysis.

Periodicity analysis of the registral accent signals is performed using a bank of comb filter resonators very similar to those used by Scheirer in [Sch98]. Figure 5 illustrates the energies of the comb filters as a function of their feedback delay, i.e., period, τ. The energies are shown for two types of artificial signals, an impulse train and a white-noise signal. It is important to notice that all resonators that are in *rational-number relations* to the period of the impulse train (24 samples) show response to it. This turned out to be important for meter analysis. In the case of an autocorrelation function, for example, only integer multiples of 24 come up and, in order to achieve the same meter estimation performance, an explicit postprocessing step ("enhancing") is necessary where the autocorrelation function is progressively decimated and summed with the original autocorrelation function.

Before we ended up using comb filters, four different period estimation algorithms were evaluated: the above-mentioned “enhanced” autocorrelation, enhanced *YIN* method of de Cheveigné and Kawahara [deC02], different types of comb-filter resonators [Sch98], and banks of phase-locking resonators [Lar94]. As an important observation, three out of the four period estimation methods performed equally well after a thorough optimization. This suggests that the key problems in meter estimation are in measuring phenomenal accentuation and in modeling higher-level musical knowledge, not in finding exactly the correct period estimator. A bank of comb filter resonators was chosen because it is the least complex among the three best-performing algorithms.

The comb filters serve as feature extractors for two probabilistic models. One model is used to estimate the period-lengths of metrical pulses at different levels. The other model is used to estimate the corresponding phases (see Fig. 4). The probabilistic models encode prior musical knowledge regarding well-formed musical meters. In brief, the models take into account the dependencies between different pulse levels (tatum, tactus, and measure) and, additionally, implement temporal tying between successive meter estimates. As shown in the evaluation section of [P6], this leads to a more reliable and temporally stable meter tracking.

2.3 Results and criticism

The method proposed in [P6] is quite successful in estimating the meter of different kinds of music signals and improved over two state-of-the-art reference methods in simulations. Similarly to human listeners, computational meter estimation was easiest at the tactus pulse level. For the measure pulse, period estimation can be done equally robustly but estimating the phase is less straightforward. This appears to be due to the basic decision that multiple-F0 analysis was not employed prior to the meter analysis. Since the measure pulse is typically related to the harmonic change rate, F0 information could potentially lead to significantly better meter estimation at the measure-pulse level. For the tatum pulse, in turn, phase estimation does not represent a problem but deciding the period is difficult both for humans and for the proposed method.

The critical elements of a meter estimation system appear to be the initial time-frequency analysis part which measures musical accentuation as a function of time and the (often implicit) internal model which represents primitive musical knowledge. The former is needed to provide robustness for diverse instrumentations in e.g. classical, rock, and electronic music. The latter is needed to achieve temporally stable meter tracking and to fill in parts where the meter is only faintly implied by the musical surface. A challenge in the latter part is to develop a model which is generic for various genres, for example for jazz and classical music. The model proposed in [P6] describes sufficiently low-level musical knowledge to generalize over different genres.

3 Approaches to single-F0 Estimation

There is a multitude of different methods for determining the fundamental frequency of monophonic acoustic signals, especially that of speech signals. Extensive reviews of the earliest methods can be found in [Rab76, Hes83] and those of the more recent methods in [Hes91, deC01, Gom03]. Comparative evaluations of different algorithms have been presented in [Rab76, Hes91, deC01]. Here, it does not make sense to list all the previous methods one-by-one. Instead, the aim of this chapter is to introduce the main principles upon which different methods are built and to present an understandable overview of the research area. Multiple-F0 estimators are not reviewed here but this will done separately in Chapter 5. Also, pre/post-processing mechanisms are not considered but an interested reader is referred to [Hes91, Tal95, Gom03].

Fundamental frequency is the measurable physical counterpart of *pitch*. In Sec. 1.1, pitch was defined as the frequency of a sine wave that is matched to the target sound by human listeners. Along with *loudness*, *duration*, and *timbre,* pitch is one of the four basic perceptual attributes used to characterize sound events. The importance of pitch for hearing in general is indicated by the fact that the auditory system tries to assign a pitch frequency to almost all kinds of acoustic signals. Not only sinusoids and periodic signals have a pitch, but even noise signals of various kinds can be consistently matched with a sinusoid of a certain frequency. For a steeply lowpass or highpass filtered noise signal, for example, a pitch is heard around the spectral edge. Amplitude modulating a random noise signal causes a pitch percept corresponding to the modulating frequency. Also, the sounds of bells, plates, and vibrating membranes have a pitch although their waveform is not clearly periodic and their spectra do not show a regular structure. A more complete review of this “zoo of pitch effects” can be found in [Hou95, Har96]. The auditory system seems to be strongly inclined towards using a single frequency value to summarize certain aspects of sound events. Computational models of pitch perception attempt to replicate this phenomenon [Med91a,b, Hou95].

In the case of F0 estimation algorithms, the scope has to be restricted to periodic or nearly periodic sounds, for which the concept fundamental frequency is defined. For many algorithms, the target signals are further limited to so-called *harmonic sounds.* These are discussed next.

3.1 Harmonic sounds

Harmonic sounds are here defined as sounds which have a spectral structure where the dominant frequency components are approximately regularly spaced. Figure 6 illustrates a harmonic sound in the time and frequency domains.

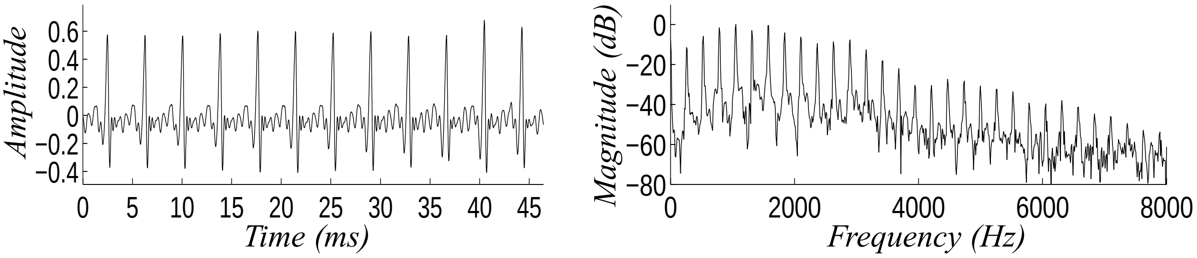

Figure 6. A harmonic sound illustrated in the time and frequency domains. The example represents a trumpet sound with fundamental frequency 260Hz and fundamental period 3.8ms. The Fourier spectrum shows peaks at integer multiples of the fundamental frequency.

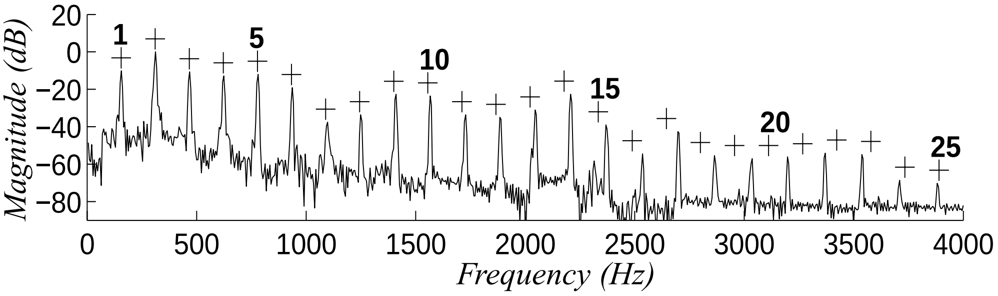

Figure 7. Spectrum of a vibrating piano string ($F = 156\,\text{Hz}$). Ideal harmonic locations are numbered and indicated with "+" marks above the spectrum. The inharmonicity phenomenon (i.e., non-ideal harmonicity) shifts the 24th harmonic partial to the position of the 25th ideal harmonic.

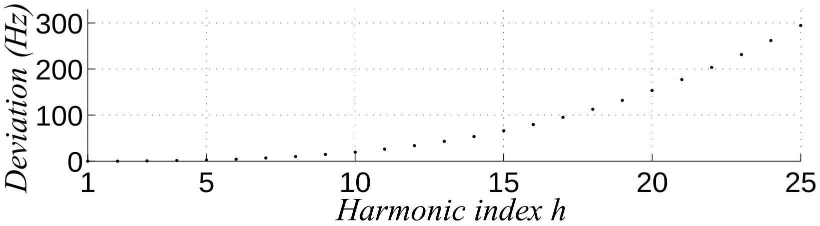

Figure 8. Deviation of the partial frequency f_h from the ideal (hF), when (3.1) with $F = 100\,\text{Hz}$ and moderate inharmonicity factor $\beta = 0.0004$ is used to calculate f_h.

For an ideal harmonic sound, the frequencies of the overtone partials (harmonics) are integer multiples of the F0. In the case of many real-world sound production mechanisms, however, the partial frequencies are not in *exact* integral ratios although the general structure of the spectrum is similar to that in Fig. 6. For stretched strings, for example, the frequencies of the partials obey the formula

$$f_h = hF\sqrt{1 + \beta(h^2 - 1)}, \tag{3.1}$$

where F is the fundamental frequency, h is harmonic index (partial number), and β is inharmonicity factor [Fle98, p.363]. Figure 7 shows the spectrum of a vibrating piano string with the ideal harmonic frequencies indicated above the spectrum. The *inharmonicity* phenomenon appears so that the higher-order partials have been shifted upwards in frequency. However, the structure of the spectrum is in general very similar to that in Fig. 6 and the sound belongs to the class of harmonic sounds. Here, the inharmonicity is due to the stiffness of real strings which contributes as a restoring force along with the string tension [Jär01]. As a consequence, the strings are *dispersive*, meaning that different frequencies propagate with different velocities in the string. Figure 8 illustrates the deviation of the frequency f_h from the ideal harmonic position, when a moderate inharmonicity value $\beta = 0.0004$ is substituted to (3.1).

Figure 9 shows an example of a sound which does not belong to the class of harmonic sounds although it is nearly periodic in time domain and has a clear pitch. In Western music, *mallet percussion instruments* are a case in point: these instruments produce pitched sounds which are not harmonic. The vibraphone sound in Fig. 9 represents this family of instruments.

The methods proposed in this thesis are mainly concerned with harmonic sounds (not assuming ideal harmonicity, however) and do not operate quite as reliably for nonharmonic sounds, such as that illustrated in Fig. 9. This limitation is not very severe in Western music, though.

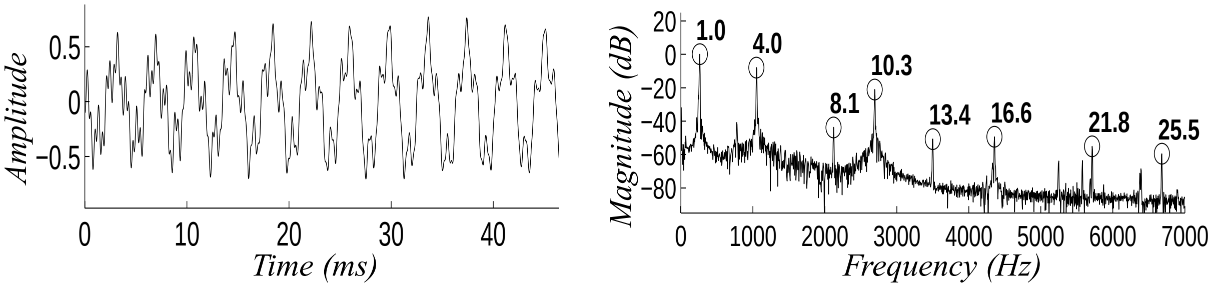

Figure 9. A vibraphone sound with fundamental frequency 260Hz illustrated in the time and frequency domains. In the right panel, frequencies of the most dominant spectral components are shown in relation to the F0.

Table 2: Western musical instruments which do or do not produce harmonic sounds.

Produced sounds	Instrument family	Instruments involved
Harmonic	String instruments	Piano, guitars, bowed strings (violin etc.)
	Reed instruments	Clarinets, saxophones, oboe, bassoon
	Brass instruments	Trumpet, trombone, tuba, english/french horn
	Flutes	Flute, bass flute, piccolo, organ
	Pipe organs	Flue pipes and reed pipes
	Human voice (singing)	Voiced phonemes
Not harmonic	Mallet percussions	Marimba, xylophone, vibraphone, glockenspiel
	Drums	Kettle drums, tom-toms, snare drums, cymbals

Table 2 lists Western musical instruments that do or do not produce harmonic sounds. The family of mallet percussion instruments is not very commonly used in contemporary music.

3.2 Taxonomy of F0 estimation methods

F0 estimation algorithms do not only differ in technical details but in regard to the very information that the calculations are based on. That is, there is no single obvious way of calculating the F0 of an acoustic signal which is not perfectly periodic and may be presented in background noise. Another problem is that often the model-level assumptions of the algorithms have not been explicitly stated, making it difficult to compare different algorithms and to combine their advantages. To this end, some categorization and model-level analysis of various methods is presented here.

In psychoacoustics, computational models of pitch perception have been traditionally classified as either place models or temporal models. An excellent introduction to these competing theories and their supporting evidence can be found in [Har96]. A convincing attempt to unify the two theories (a “unitary model”) has been presented by Meddis and colleagues in [Med91a,b, 97].

In the case of practical F0 estimation methods, a different categorization is more useful. Algorithms are here grouped to those that look for frequency partials at harmonic *spectral locations* and to those that observe *spectral intervals* (frequency intervals) between partials. The underlying idea of both of these categories can be understood by looking at Fig. 6 and is described in more detail in the next two subsections. Algorithms which measure periodicity of the time-domain signal belong to the first category. Algorithms which measure periodicity of the Fourier spectrum belong to the latter category. Algorithms which measure periodicity of the time-domain amplitude envelope represent a tradeoff between the two classes and are described in Sec. 3.5. Recent models of human pitch perception are of this kind: both spectral locations and spectral intervals are important in hearing. In the following, representative variants from each category are introduced and analyzed in order to describe their properties and advantages.

3.3 Spectral-location type F0 estimators

3.3.1 Time-domain periodicity analysis methods

Time-domain autocorrelation function (ACF) based algorithms are among the most frequently used F0 estimators (see e.g. [Bro91, Tal95]). As pointed out by Tolonen and Karjalainen in [Tol00], ACF-based F0 estimators have close model-level similarities with cepstrum-based F0 estimators [Nol67], and there is a continuum between them. This becomes evident when calculating ACF of a time-domain signal $x(n)$ via the discrete Fourier transform (DFT) and its inverse (IDFT) as

$$r(\tau) = \mathrm{IDFT}\{|\mathrm{DFT}[x(n)]|^2\}. \tag{3.2}$$

Definition of the cepstrum $c(\tau)$ of $x(n)$ is very analogous to (3.2) and is obtained by replacing the second power with a logarithm function. The difference between the ACF and cepstrum-based F0 estimators is quantitative. Raising the magnitude spectrum to the second power emphasizes spectral peaks in relation to noise but, on the other hand, further aggravates spectral peculiarities of the target sound. Applying the logarithm function causes the opposite for both. And indeed, ACF-based F0 estimators have been reported to be relatively noise immune but sensitive to formant structures in speech: especially the first and the strongest formant may mislead the algorithm [Rab76, Tal95]. On the contrary, cepstrum-based F0 estimators perform relatively poorly in noise, but well for exotic sounds [Rab76]. As a tradeoff, Tolonen *et al.* suggest using a “generalized autocorrelation function” where the second power is replaced with a real-valued exponent (0.67 in their case) [Tol00].

Both ACF and cepstrum-based F0 detectors are implicit realizations of a model which emphasizes frequency partials at *harmonic locations* of the magnitude spectrum. This can be seen by writing the ACF in terms of the Fourier spectrum $X(k)$ of a a real-valued input signal as

$$r(\tau) = \frac{1}{K}\sum_{k=0}^{K-1}\left[\cos\left(\frac{2\pi\tau k}{K}\right)|X(k)|^2\right], \tag{3.3}$$

where K is the length of the transform frame. The above formula is equivalent to (3.2). Figure 10 illustrates the calculations in the case when τ corresponds to the true period of the example sound. Squared magnitude-spectrum components are weighted according to their spectral locations and then summed. Thus we call ACF and cepstrum-based methods *spectral location* type F0 estimators.

Recently, a conceptually simple and very accurate F0 estimation method was proposed by de Cheveigné and Kawahara in [deC02]. Their algorithm is called “YIN” and is based on the ACF with certain modifications. The novelty of the method culminates to a specific normalization of the autocorrelation function which reduces the number of free parameters instead of increasing it. In [deC01], the method was thoroughly evaluated and compared with previous

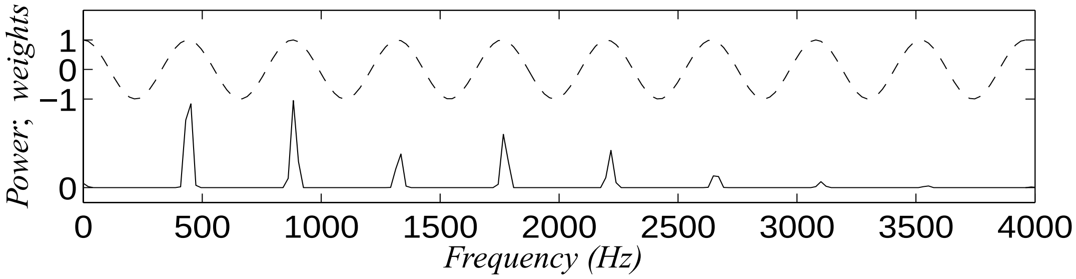

Figure 10. Solid line illustrates the power spectrum of a trumpet sound. Dashed line shows the weights $\cos(2\pi\tau k/K)$ of ACF calculation in (3.3) when τ corresponds to the fundamental period of the example sound. The two curves are displaced vertically for clarity.

methods using a large database of speech signals. The usability of the method for music transcription has been evaluated by us in [Vii03] and in Publication [P5].

Autocorrelation-based methods are closely related to methods which average the absolute difference between a signal and its delayed version. An interesting generalization over the time-delay based methods has been recently proposed by Terez who applied state-space embedding in searching the time delay with which the signal resembles itself [Ter02].

3.3.2 Harmonic pattern matching in frequency domain

Another way of weighting frequency components according to their spectral locations is to perform explicit pattern matching in the frequency domain. In [Bro92b], Brown proposed a method where an input sound is first analyzed by simulating a 1/24th-octave filterbank. This leads to a spectral representation where the frequency components are logarithmically spaced. In log-frequency domain, the partials of a harmonic sound have a spacing which is independent of the F0 of the sound. For example, the distance between the second and the third harmonic partial is $\ln(3/2)$ regardless of the F0. As a consequence, F0 estimation can be performed by cross-correlating the vector of filterbank energies with an ideal harmonic pattern where unity values are assigned to harmonic positions and zeros elsewhere. The maximum of the cross correlation function indicates the position of the fundamental frequency.

A maximum-likelihood spectral pattern matching F0 estimator was proposed by Doval and Rodet in [Dov91, 93]. The authors used a set of sinusoidal partials to represent the spectrum of an input sound and a F0 was then sought for which best explained the observed partials. This was done in a Bayesian manner so that Gaussian functions centered on each multiple of a hypothesized F0 were used to represent the likelihood of observing the partials given the F0 candidate. Supplementary non-harmonic and noisy spectral components were separately modeled.

Another completely different and interesting spectral pattern matching approach is the two-way mismatch method of Maher and Beauchamp [Mah94]. In their method, the F0 is chosen to minimize discrepancies between observed frequency components and harmonic frequencies generated by trial F0 values. The first mismatch measure is calculated as an average of the frequency differences between each observed partial and its nearest neighbour among the predicted harmonic frequencies. This is combined with a mismatch measure calculated by averaging the frequency differences between each predicted harmonic frequency and its nearest neighbour among the observed partials.

3.3.3 A shortcoming of spectral-location type F0 estimators

A major shortcoming of the spectral-location oriented F0 estimation methods is that they are not able to handle non-ideal harmonic sounds appropriately. As described in Sec. 3.1, the partials of real physical vibrators cannot be assumed to be found exactly at harmonic spectrum positions. The spectrum of a piano sound in Fig. 7 illustrates this situation. Inharmonicity is not a big concern in speech processing, but is immediately met when analyzing musical sounds at a wide frequency band. The methods described in the next section are advantageous in this respect.

3.4 Spectral-interval type F0 estimators

The spectrum autocorrelation method and its variants have been successfully used in several F0 estimators (see e.g. [Lah87, Kun96]). The idea is derived from the observation that a periodic but non-sinusoidal signal has a periodic magnitude spectrum, the period of which is the F0. In its simplest form, the autocorrelation function $\tilde{r}(m)$ over the positive frequencies of a K-length magnitude spectrum can be calculated as

$$\tilde{r}(m) = \frac{2}{K} \sum_{k=0}^{K/2-m-1} |X(k)||X(k+m)| . \tag{3.4}$$

The information on which the above equation bases F0 calculations is fundamentally different from that of a time-domain ACF or cepstrum-based F0 detector. Any two frequency components with a certain spectral interval support the corresponding F0. This means that the spectrum can be arbitrarily shifted without affecting the output value. Different kinds of preprocessing can be done for the spectrum before the periodicity analysis. For example, Kunieda *et al.* compress the magnitude spectrum with a logarithm function and then remove the spectral trend by highpass liftering[1] [Kun96]. Lahat *et al.* apply a bank of bandpass lifters prior to the periodicity analysis [Lah87].

Building F0 calculations upon the *intervals* between frequency partials works better for sounds that exhibit inharmonicities. Even though the intervals do not remain constant, they are more stable than the locations of the partials, which shift cumulatively, as can be seen in (3.1) and in Fig. 8.

Another interesting difference between spectral-location and spectral-interval based approaches is that the former methods are prone to errors in F0 halving and the latter to errors in F0 doubling. As an example, consider the time-domain ACF estimator (spectral-location type method). The time-domain signal is periodic at half the F0 rate (twice the fundamental period) and the harmonic partials of a sound match the locations of even harmonics of a two times lower sound. In the case of the frequency-domain ACF estimator (spectral-interval type method), the magnitude spectrum is periodic at double the F0 rate but shows no periodicity at half the F0 rate.

[1] Liftering refers to the process of convolving the sequence of magnitude-spectrum samples with the impulse response of the specified filter.

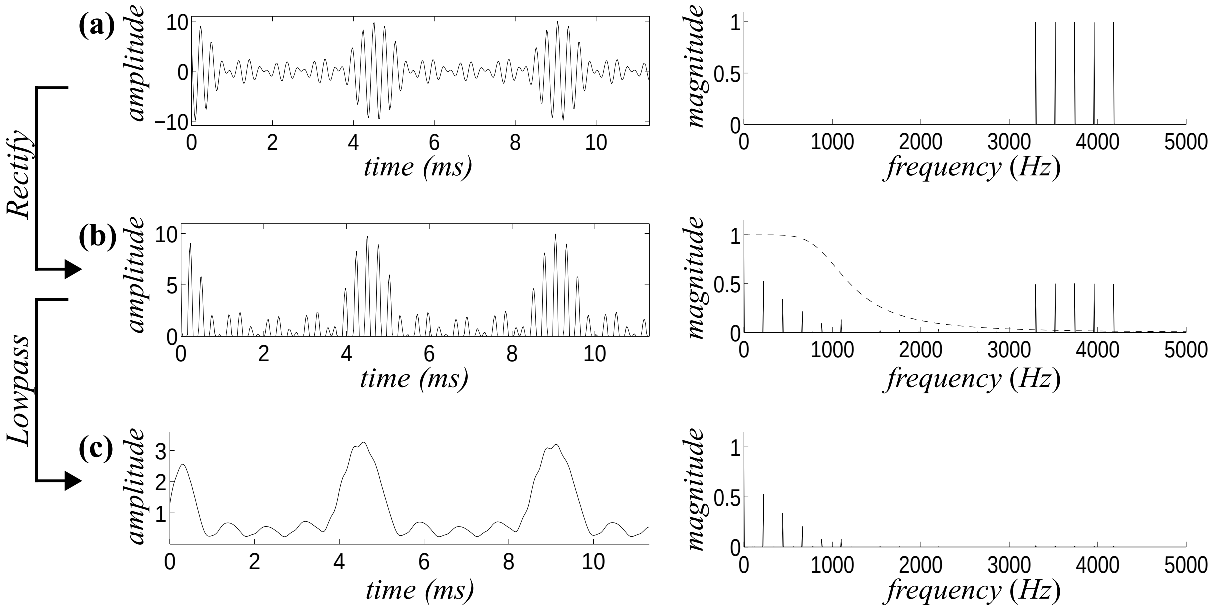

Figure 11. Reading top-down: (a) a signal containing the harmonics 15–19 of a sound with F0 220Hz, (b) the signal after half-wave rectification, and (c) the signal after rectification and lowpass filtering. The response of the lowpass filter is shown as a dashed line in (b).

3.5 "Unitary model" of pitch perception

3.5.1 Periodicity of the time-domain amplitude envelope

In the previous sections, algorithms were introduced which measure the periodicity of the time-domain signal or the periodicity of the Fourier spectrum. A third, fundamentally different approach is to measure the periodicity of the time-domain amplitude envelope. Several successful F0 estimators are based on this principle, especially those developed in an attempt to model the human auditory system [Med91a,91b,97, Hou95, Tol00]. The idea is derived from the observation that any signal with more than one frequency component exhibits periodic fluctuations, *beating*, in its time-domain amplitude envelope. That is, the partials alternatingly amplify and cancel each other. The rate of beating depends on the frequency difference between each two frequency components. In the case of a harmonic sound, the frequency interval corresponding to the F0 dominates, and the fundamental beat is visible in the amplitude envelope of the signal.

Figure 11 illustrates the beating phenomenon for a set of harmonic partials of a sound with a 220Hz fundamental frequency. The amplitude envelope of the signal is obtained by applying half-wave rectification and lowpass filtering on the signal in time domain. The half-wave rectification operation is defined as

$$\mathrm{HWR}(x) = \begin{cases} x & x \geq 0 \\ 0 & x < 0 \end{cases}. \tag{3.5}$$

As can be seen in Fig. 11, the rectification generates spectral components around zero frequency. These represent the spectrum of the amplitude envelope. Then, the amplitude envelope is obtained by lowpass filtering the rectified signal. The fundamental period of the signal (4.5ms) is clearly visible in the resulting signal.

At first it seems that analyzing the periodicity of the amplitude envelope leads to a spectral-interval type F0 estimator. However, the principle of rectification and lowpass filtering allows an elegant tradeoff between the spectral-location and spectral-interval type information. This is achieved by tuning the cutoff frequency of the lowpass filter in Fig. 11. If the cutoff frequency of the lowpass filter is risen so that the filter passes the spectral components of the original narrowband signal, a subsequent periodicity analysis (ACF computation, for example) utilizes both spectral-location and spectral-interval information. This is because the spectrum after rectification contains partials at frequencies which correspond to the spectral locations of partials in the input signal *and* to the spectral intervals between the partials. In human hearing, a tradeoff between the two types of information is achieved using this kind of representation. This is described in more detail in the following.

3.5.2 Unitary model of pitch perception

Meddis and colleagues have proposed a computational model of human pitch perception where the periodicity of the amplitude envelope is analyzed at the outputs of a bank of bandpass filters [Med91a,b, 97]. The roots of the model are in the early work of Licklider [Lic51]. Simulation experiments with the model have shown that this single model (hence the name “unitary”) is capable of reproducing a wide range of phenomena in human pitch perception, thus reconciling discrepancies between competing psychoacoustic theories (as mentioned in Sec. 3.2). For convenience, this model is referred to simply as “unitary model” in the following.

The unitary model consists of the following processing steps [Med97]:

1. An acoustic input signal is passed though a bank of bandpass filters which represent the frequency selectivity of the inner ear. The bands (or, channels) are approximately uniformly distributed on a logarithmic frequency scale. Typically, 40–128 filters with partly overlapping passbands are applied.
2. The signal at each channel is compressed, half-wave rectified, and lowpass filtered.
3. Periodicity estimation within channels is carried out by calculating short-time autocorrelation functions.
4. The ACF estimates are linearly summed across channels to obtain a *summary autocorrelation function* (SACF) defined as

$$s_t(\tau) = \sum_c r_{t,c}(\tau) \tag{3.6}$$

where $r_{t,c}(\tau)$ is the autocorrelation function at time *t* in subband *c*. The maximum value of $s_t(\tau)$ is typically used to determine the time delay (lag) which corresponds to the pitch period at time *t*.

The first two steps of the above algorithm correspond to the peripheral parts of hearing which produce a signal to the auditory nerve. There is a wide consensus concerning the general properties of these processing steps because the signal in the auditory nerve can be directly measured. The steps 3 and 4 are more controversial since they represent processing which takes place in the central nervous system and is not directly observable. Particularly the use of the ACF for periodicity estimation has been a subject to criticism since no nervous units have been found that could implement ACF as such (see e.g. [Har96, p.3498]) and some experimental evidence contradicts the ACF [Kae98, 01]. The authors themselves admit that the ACF is not necessarily the exact mechanism that the auditory system uses. In [Med91a, p. 2879], they write: “The model, therefore, remains neutral on the exact mechanism whereby temporal information is extracted from the activity of the auditory-nerve fibers - - -” (activity of the auditory-nerve fibers is modelled by the amplitude envelopes). The overall processing chain, however, has been very successful in reproducing phenomena in human hearing and the model therefore prevails until a better one is found. A viable substitute for ACF has not been found despite some attempts e.g. in [Bra99, 00].

Envelope periodicity is in practice always analyzed at subbands and the results are then combined in the end. This applies not only to the unitary pitch model but also to practical F0 extraction methods. It should be noted that analyzing the periodicity of the time-domain signal itself or the Fourier spectrum at subchannels (and then summing) does not make sense since this is equivalent to performing the periodicity analysis directly for the wideband signal.

3.5.3 Attractive properties of the unitary model

The unitary model has a number of properties which are particularly attractive from the point of view of F0 estimation. Furthermore, these properties stem already from the first two (widely accepted) processing steps which produce the signal traveling in the auditory nerve. Three properties are discussed in the following.

First, the model provides a psychoacoustically plausible way of weighting the spectral-location and spectral-interval type information in F0 estimation. The half-wave rectification operation retains the spectral components of the input signal around f_c (center frequency of the bandpass filter at channel c) but, additionally, generates the spectrum of the amplitude envelope around zero frequency (in addition, harmonic distortion is generated at integer multiples of f_c; this is discussed later). The spectrum around f_c represent spectral-location type information whereas the spectral components of the amplitude envelope correspond to intervals between the partials in the input signal as described in Sec. 3.5.1. The subsequent ACF computation performs pattern matching on this spectrum, as illustrated in Fig. 10.

The magnitude response of the lowpass filter (illustrated in Fig. 11) determines the balance between the two types of information. In the unitary model, the lowpass filter is common to all subbands and is typically designed to pass signal components below about 1kHz and to have a smooth transition band so that signal components above 1kHz are increasingly attenuated. As a consequence, the passband overlaps the passbands of the bandpass filters of the channels below 1kHz.

Secondly, envelope-periodicity models perform implicit *spectral smoothing*: the amplitude of the beating caused by each two frequency partials is determined by the smaller of the two amplitudes. This is illustrated in Fig. 12. When the spectrum of a harmonic sound is considered, each two neighboring harmonics contribute to the beating at the fundamental rate. However, since the magnitude of the beating is determined by the smaller of the two amplitudes, individual higher-amplitude harmonic partials are filtered out. The phenomenon is well-known in human hearing: if one partial of a harmonic sound raises clearly above the other partials, it is perceptually segregated and stands out as an independent sound [Bre90]. This feature turned out to be of vital importance in multiple-F0 estimation, as described in [P3] and Sec. 6.4.2.

A third property of the envelope periodicity models is that they are phase sensitive in aggregating the beating caused by each combination of two frequency partials. This is likely to be advantageous in multiple-F0 estimation since the frequency components arising from a same physical source are sometimes phase-locked to each other. In music, this applies to the brass

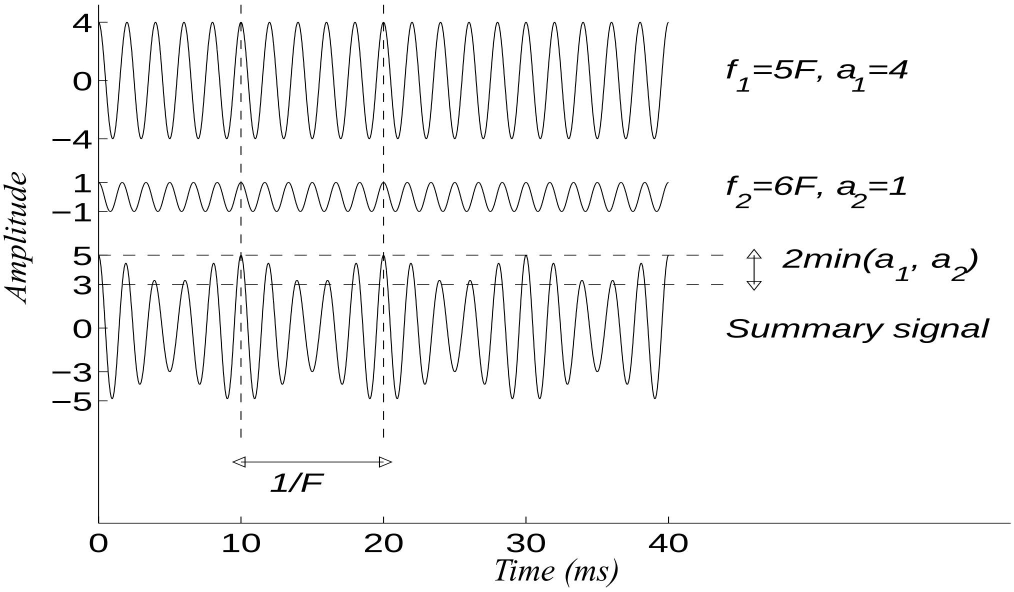

Figure 12. Illustration of the beating caused by two sinusoidal signals. The sinusoids represent the multiples 5 and 6 of a fundamental $F = 100$ Hz. The lowest curve shows the linear sum of the two. Note that the magnitude of the beating is according to the smaller-magnitude sinusoid. The beating frequency is the frequency difference of the two.

and the reed instruments, and to the human voice. However, the usefulness of this feature in computational multiple-F0 estimation has not been empirically validated.

The role of compression in Step 2 of the model amounts to spectral whitening (i.e., spectral flattening) and generates harmonic distortion components on odd multiples of f_c. This will be discussed in more detail in Sec. 4.1.3. Many authors omit the compression altogether and perform spectral whitening as a preprocessing step by inverse linear-prediction filtering [Tol00], or, by normalizing the powers of the outputs of the bandpass filterbank [Ell96].

4 Auditory-model based multiple-F0 estimator

The aim of this chapter is to investigate how the principles of the unitary pitch model (introduced in Sec. 3.5) can be utilized in practical multiple-F0 estimation. A novel method is proposed which estimates the F0s of several concurrent musical sounds in a single time frame. Computational efficiency of the method is based on the algorithm proposed in [P4]. Accuracy of the method is based on specific modifications to the unitary model which are originally proposed here.

In Sec. 4.1, a detailed frequency-domain analysis of the unitary pitch model is presented. This serves as background material for [P4] and paves the way for the algorithm presented in it. In Sec. 4.2, the auditory-model based multiple-F0 estimator is presented. The method is based on the analysis of the unitary model in Sec. 4.1.

Throughout this chapter it should be kept in mind that our aim is not to propose a model of the human auditory system (this issue was discussed in Sec. 1.3.1). The approximations and modifications to be proposed to the unitary model are ultimately justified by the fact that, as a result, a practically applicable multiple-F0 estimation algorithm is obtained.

4.1 Analysis of the unitary pitch model in frequency domain

The summary autocorrelation function (SACF) of the unitary pitch model has turned out to be a very useful *mid-level representation* (see Sec. 1.2.3) in several tasks, such as multiple-F0 estimation, sound separation, and computational auditory scene analysis [deC99, Tol00, Wan99, Ell95,96, Ros98a]. However, the computational complexity of calculating the SACF is rather high, due to the ACF calculations at 40–128 subbands (the number varies in different implementations). This has limited the usability of the model in practical applications.

In [P4], we have proposed a method for calculating an approximation of the SACF in the frequency domain. Concretely, what the algorithm achieves is to compute one spectral line of the Fourier transform of the SACF, $\mathrm{DFT}(s_t(\tau))$, in $O(K)$ time, given the complex Fourier spectrum of the input sound. Here K is the length of the transform frame and the order-of-growth notation $O(\bullet)$ is according to the common usage [Cor90]. The individual spectral lines allow computationally efficient multiple-F0 estimation as will be described in Sec. 4.2.

Another thing achieved by the algorithm in [P4] is that it removes the need to define the number of distinct subbands. The algorithm implements a model where one subband is centered on each discrete Fourier spectrum sample, thus approaching a continuous density of subbands. The bandwidths of the channels need not be changed. This is exactly the opposite to the approach of Tolonen *et al.* in [Tol00], where a computationally efficient version of the unitary pitch model was proposed by *reducing* the number of subbands to two. Tolonen's multiple-F0 estimation method is rather successful and has been evaluated in [P5].

The presentation in [P4] is very condensed due to the page limit. In the following, background material that facilitates the understanding of [P4] is presented. A starting point for the algorithm is to perform the four calculation phases of SACF (see Sec. 3.5) in the frequency domain. This inevitably leads to frame-based processing. However, this is not a serious problem, since the ACF calculations involved in the conventional SACF calculations are in practice always performed on a frame-by-frame basis to allow FFT-based ACF computations ([Ell96] represents an exception).

In the following, we look at the processing steps of the unitary pitch model in more detail. Particular emphasis is laid on the first two steps of the unitary model: (i) the bank of bandpass filters and (ii) compression, rectification, and lowpass filtering at the subbands. As already mentioned in the previous chapter, performing the mentioned nonlinear operations at specific subbands leads to a number of attractive properties from the point of view of multple-F0 estimation. Thus, the two steps are studied in depth. Time-domain compression, which was not discussed in [P4], is included here.

4.1.1 Auditory filters (Step 1 of the unitary model)

The *cochlea* is an organ in the inner ear which transforms sound pressure level variations into neural impulses in the auditory nerve. Frequency analysis is an essential part of this process. Frequency components of a complex sound can be perceived separately and are coded independently in the auditory nerve (in distinct nerve fibers), provided that their frequency separation is sufficiently large [Moo95b]. The auditory frequency analyzer is usually modeled as a bank of overlapping, linear, bandpass filters, called *auditory filters*. Combination of information across channels then takes place in the central nervous system.

The concept *critical band* is closely related to the auditory filters. The term was coined by Fletcher in 1940 [Fle40]. Fletcher's experiment is illustrated in Figure 13*a*. He measured the threshold of detecting a sinusoidal signal in the presence of a bandpass noise masker which was centered on the signal frequency. The power spectral density of the noise was held constant but the noise bandwidth was varied. Detection threshold increased as a function of the noise bandwidth, but only up to a certain point. Fletcher labeled this bandwidth a "critical band" and suggested that the frequency analysis of the inner ear could be modeled with a bank of bandpass filters (which became to be called *auditory filters*). Noise components contibute to masking in proportion to their power at the output of the filter which captures the target signal.

Even later, the auditory filters have been most often studied using the masking phenomenon. *Masking* refers to a situation where an audible sound becomes inaudible in the presence of another, louder sound. In particular, if the distance of two spectral components is less than the critical bandwidth, one easily masks the other. The situation can be thought of as if the components would go to the same auditory filter, to the same "channel" in auditory nerve. If the frequency separation is larger, the components are coded separately and are both audible.

By making certain assumptions about the auditory filters, it is possible to measure the bandwidth and the shape of their power response. The notched-noise method as originally suggested by Patterson in [Pat76] is illustrated in Fig. 13*b*. A wide-band noise signal with a spectral notch (a stopband) is used to mask a sinusoidal tone. The notch is centered on the tone, and the threshold of detecting the sinusoid as a function of the width of the notch is measured. Two main assumptions are made. First, that the auditory filter which captures the sinusoidal signal is centered on the signal frequency and is reasonably symmetric on a linear frequency scale, and secondly, that masking is due to the part of noise which leaks to the same auditory channel at the sides. Because the detection threshold level is known be directly proportional to the masking noise level [Pat86], the amount of noise leaking to the channel and thus the shape of the auditory filter can be deduced. By placing noise bands on both sides it can be ensured that the signal is not heard through the neighbouring auditory filters, i.e., by "off-frequency listening".

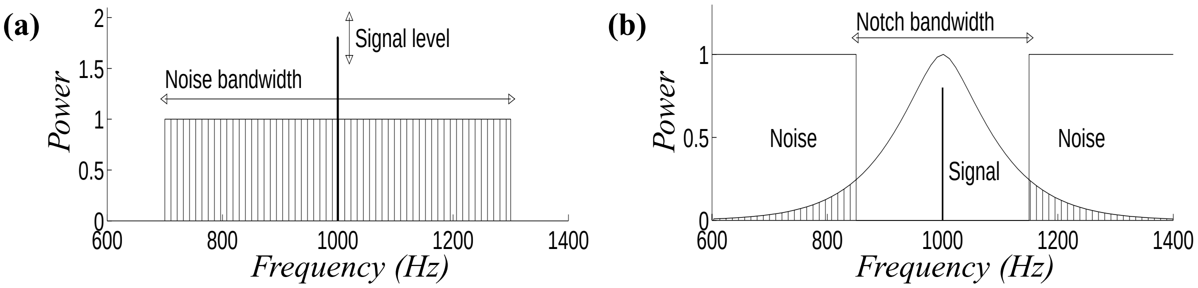

Figure 13. (a) Illustration of Fletcher's experiment. See text for details. (b) Idea of the notched-noise method which has been used to determine the shape of the auditory filter response (after [Pat76]). In both panels, the power spectral density of the noise is constant and signal level is varied.

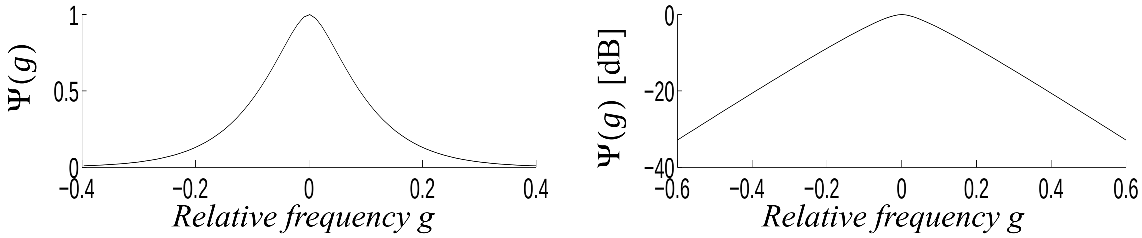

Figure 14. Squared magnitude response of the rounded-exponential (roex) filter.

The shape of the auditory filters has been approximated with simple mathematical expressions that have few free parameters. Patterson *et al.* have suggested three different formulas [Pat86]. It is convenient to express the formulas in terms of a new frequency variable

$$g = \frac{f - f_c}{f_c}, \tag{4.1}$$

where normalization with the center frequency f_c facilitates the comparison of filters with different center frequencies. As can be seen, g measures normalized deviation from the center frequency. The squared magnitude response of the *rounded exponential,* roex, filter as suggested in [Pat86] is given by

$$\Psi^{(roex)}(g) = (1 + p|g|)e^{-p|g|} \tag{4.2}$$

where *p* is a parameter which determines the bandwidth of the filter. The response is illustrated in Fig. 14. Other models have been proposed, too, but only at the expense of more parameters for the magnitude response function. In the unitary model, the exact shape of the auditory filter response is not critical, therefore it does not make sense to look at the more complex functions. The model in (4.2) is completely sufficient for our purposes and has spread to wide use, largely because the response can be implemented computationally efficiently using gammatone filters [Sla93].

The bandwidths of auditory filters can be conveniently expressed using the *equivalent rectangular bandwidth* (ERB) concept. The ERB of a filter is defined as the bandwidth of a perfect rectangular filter which has a unity response in its passband and integral over the squared magnitude response which is the same as for the specified filter. Integral over the squared roex response is obtained by substituting (4.1) to (4.2), integrating over the right half of the response, and multiplying the result by two:

$$2\int_{f_c}^{\infty}\Psi^{(roex)}(f)df = 2\int_{f_c}^{\infty}[(1+p(f-f_c)/f_c)e^{-p(f-f_c)/f_c}]df = \frac{4f_c}{p}. \qquad (4.3)$$

Glasberg and Moore measured the ERB values of the auditory filters over a wide range of center frequencies using the notched noise method [Gla90]. They found that the ERB values can be described as a linear function of the center frequency as

$$u(f_c) = 24.7[1+4.37f_c/1000], \qquad (4.4)$$

where the center frequency and the bandwidth are in hertz units. In the following, we use the shorthand notation u_c to refer to the ERB value of an auditory filter according to (4.4), and f_c to refer to the corresponding center frequency of the filter.

By setting the ERB values in (4.3) and (4.4) equal, the variable p of the roex filters can be calculated as

$$p = \frac{4f_c}{u_c}. \qquad (4.5)$$

As a result, the response of a roex filter at a given center frequency is fully determined.

The center frequencies of the filters in the auditory filterbank are typically assumed to be uniformly distributed on a *critical-band scale*. This frequency-related scale is derived by integrating the inverse of (4.4) which yields

$$e(f) = 21.4\log_{10}(4.37f/1000+1). \qquad (4.6)$$

In the above expression, f is frequency in hertz and $e(f)$ gives the critical-band scale. Intuitively, this means that the auditory filters are more densely distributed at the low frequencies where the ERB values are smaller. More exactly, the power responses of the filters sum to a flat response over the whole range of center frequencies. When f varies between 20Hz and 20kHz (the hearing range), $e(f)$ varies between 0.8 and 42. Intuitively, this means that approximately 41 critical bands (auditory filters) would fit to the range of hearing if the passbands of the auditory filters were non-overlapping and rectangular in shape.

4.1.2 Flatted exponential filters

The algorithm in [P4] has been derived for a certain family of filters called *flatted-exponential*, flex, filters. The shape of the response of these filters differs somewhat from that of rounded-exponential filters. From the point of view of the unitary pitch model, however, the exact shape of the response is not critical, but the important characteristics of the auditory filters are that they are approximately uniformly distributed on the critical band scale given by (4.6), that the bandwidths are according to (4.4), and that the number of filters is large enough to make the passbands of adjacent filters overlap significantly.

Three requirements suffice to fully define the response of the flatted-exponential filters. (i) The filters implement a flat (unity) response around the center frequency of the filter. (ii) The slope of attenuation further away from the center frequency is defined to be the same as the asymptotic attenuation of the rounded-exponential filters. The attenuation of the roex filters follows the slope $e^{-p|g|}$ further away from the center frequency, since the factor $(1+p|g|)$ in (4.2) becomes insignificant for large values of $|g|$. (iii) The ERB value of the filters is defined to be the same as that of the corresponding roex filter, i.e., according to (4.4).

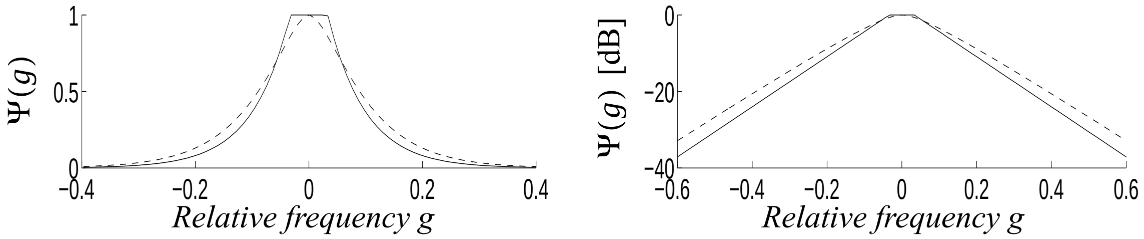

Figure 15. Solid line shows the squared magnitude response of the flatted-exponential (flex) filter. Dashed line shows the response of the rounded-exponential (roex) filter

The flatted-exponential filter response is illustrated in Fig. 15 and can be written as

$$\Psi^{(flex)}(g) = \begin{cases} 1 & |g| \le g_0 \\ \exp[-p(|g| - g_0)] & |g| > g_0 \end{cases} \tag{4.7}$$

where the frequency variable g is as defined in (4.1) and the variable $p = 4f_c/u_c$ stems from the requirement (ii) above.

The parameter g_0, i.e., the half-width of the flatted top can be determined by requiring that the ERB-bandwidths of flex and roex filters must be equal for a given parameter p:

$$\int_{-\infty}^{\infty} \Psi^{(flex)}(g)dg = \int_{-\infty}^{\infty} \Psi^{(roex)}(g)dg \tag{4.8}$$

Integral over the squared flex response is obtained by integrating piecewise over the right half and multiplying by two. The above equation becomes

$$2\int_0^{g_0} 1dg + 2\int_{g_0}^{\infty} [\exp[-p(g - g_0)]]dg = 2\int_0^{\infty} (1 + pg)e^{-pg}dg \tag{4.9}$$

and by integrating

$$2g_0 + \frac{2}{p} = \frac{4}{p} \tag{4.10}$$

from which we obtain $g_0 = 1/p$. The flex filter in (4.7) can now be written as

$$\Psi^{(flex)}(g) = \begin{cases} 1 & |g| \le 1/p \\ \exp(-p|g| + 1) & |g| > 1/p \end{cases} \tag{4.11}$$

In terms of frequency variable f this becomes (simply substitute p from (4.5) and g from (4.1)):

$$\Psi^{(flex)}(f_c, f) = \begin{cases} 1 & |f - f_c| \le u_c/4 \\ \exp[-4|f - f_c|/u_c + 1] & |f - f_c| > u_c/4 \end{cases} . \tag{4.12}$$

Please note that (4.12) gives the *squared* magnitude response of the flex filter. In the following, we use the shorthand notation $\Psi_{h_c}(f) \sim \Psi^{(flex)}(f_c, f)$ for the above power response and $H_c(k)$ to refer to the corresponding discrete magnitude response of a flex filter at center frequency f_c and with the ERB bandwidth u_c according to (4.4). That is, a bank of flatted-exponential bandpass filters will be applied in the unitary-model in the following.

4.1.3 Compression and half-wave rectification at subbands (Step 2 of the unitary model)

In the unitary pitch model as originally described in [Med91a,b], the output of each auditory filter was processed by a *hair-cell model*. In human hearing, when a sound enters the inner ear, it travels as a pressure wave within the cochlea. Hair cells are the elements which transform the resulting mechanical movement to neural impulses. The hair-cell model used in [Med91a,b] was a dynamic system described in terms of differential equations [Med86]. However, as later pointed out by the authors, the hair cells in the unitary pitch model can equally well be modeled as a cascade of compression, half-wave rectification, and lowpass filtering [Med97] (see also [Med91a], p. 2869). This approach is followed here. In some previous implementations, the compression has been omitted (see e.g. [Ell96, Tol00]). This can be done if the task of the compression is carried out by some other means of preprocessing.

In the following, compression and half-wave rectification are each discussed separately. These are thought of as a cascade, where full-wave v^{th}-law compression is performed first and this is followed by rectification.

Full-wave (odd) v^{th}-law **compression** (FWOC) is defined to have the transfer characteristic

$$\mathrm{FWOC}(x) = \begin{cases} x^v & x \geq 0 \\ -(-x)^v & x < 0 \end{cases}, \tag{4.13}$$

where $v > 0$. This specific form of compression is employed here because it matches sufficiently well the measured compression in hair cells (see [Med86]) and because it is analytically tractable and numerically safe.

What happens in the frequency domain when a narrowband signal is compressed as above? As a nonlinear operation, FWOC of course does not have a “frequency response”. However, consider a narrowband input $x(t) = A(t)\cos(\omega_c t + \phi(t))$, where $f_c = \omega_c / 2\pi$ is the center frequency of the spectral band and $A(t) \geq 0$. Here “narrowband” means that the bandwidth of $x(t)$ is small compared to f_c. This, in turn, implies that the variations of $A(t)$ and $\phi(t)$ are slow compared to those of $\cos(\omega_c t)$. For such a signal, the output $y(t)$ of a full-wave v^{th}-law compressor can be expressed as [Dav87]

$$y(t) = \sum_{\substack{m = 1 \\ (m \text{ odd})}}^{\infty} 2C(v, m)[A(t)]^v \cos[m\omega_c t + m\phi(t)], \tag{4.14}$$

where the coefficient $C(v, m)$ is

$$C(v, m) = \frac{\Gamma(v + 1)}{2^v \Gamma\left(1 - \frac{m - v}{2}\right)\Gamma\left(1 + \frac{m + v}{2}\right)}, \tag{4.15}$$

and $\Gamma(x)$ is the Gamma function. That is, the output of the v^{th}-law compressor consists of a signal at the frequency of the input narrowband signal and at the odd multiples of it. The envelope of each of these is modulated by the v^{th}-power of the input envelope and the m^{th} harmonic is phase-modulated by m times the input phase modulation.

The expression in (4.14) applies reasonably accurately for the subband signals at the outputs of the auditory filterbank. The subband signals are sufficiently narrowband to satisfy the assumptions for all except few lowest channels. That is, the subband signals can be imagined as having been modulated to their center frequencies. We denote by $x_c(n)$ the time-domain signal at the output of an auditory filter centered on frequency f_c. The discrete Fourier transform of this signal is denoted by

$$X_c(k) = \mathrm{DFT}[x_c(n)] \tag{4.16}$$

and the discrete Fourier transform of the v^{th}-law compressed signal is denoted by

$$X_c^{(comp)}(k) = \mathrm{DFT}[x_c(n)^v]. \tag{4.17}$$

At the passband of the auditory filter, the Fourier spectrum of the compressed signal can be approximated as

$$X_c^{(comp)}(k) \approx \gamma_c X_c(k)\text{, when } |kf_s/K - f_c| < u_c. \tag{4.18}$$

Here γ_c is a scalar, K is the transform length, and f_s is the sampling rate. The approximation gets less accurate when v gets very small ($v < 0.3$) or large ($v > 3$) values but, for moderate compression levels used here ($0.3 < v < 1$), the approximation is sufficiently accurate. The scaling factor γ_c can be calculated from (4.14) as

$$\gamma_c = 2C(v, 1)(\sigma_c\sqrt{2})^{(v-1)}, \tag{4.19}$$

where σ_c is the standard deviation of the subband signal and $\sigma_c\sqrt{2}$ is the amplitude of a virtual "carrier" sinusoidal signal which has the standard deviation σ_c. The approximation in (4.18)–(4.19) can be straightforwardly verified experimentally.

At the passband of the auditory filter, the scaling factor in (4.19) tends to normalize the standard deviation of the subband signal towards unity when $v < 1$. In this sense, the compression can be omitted if the sound spectrum is appropriately preprocessed (whitened) before the auditory filterbank. This has been done e.g. in [Tol00]. Ellis, in turn, took the approach of normalizing the outputs of the auditory filters by their energies [Ell96, p.77]. The distortion spectrum at odd multiples of f_c could be accurately modeled, too, but this will not be done in the following. At moderate compression levels, the magnitude of these is small compared to the passband of the auditory filter and, furthermore, the subsequent lowpass filter eliminates much of the distortion spectrum. Thus it is seen as not having an important role in the unitary pitch model.

In the following, the scaling of subbands according to (4.18) is seen purely and simply as a *psychoacoustically inspired way of performing spectral whitening*. The fact that the model was derived from time-domain v^{th}-law compression is not so important. The presented form of spectral whitening is convenient because it provides a single parameter with which to determine the degree of applied whitening. Also, interestingly, when v in (4.19) gets very small values, the spectrum due to scaling with γ_c at subbands does not approach a white spectrum, but rather, a pink spectrum. This is because the standard deviations σ_c are measured at critical bands which are approximately logarithmically spaced and have bandwidths which are linearly proportional to the center frequency. For these reasons, spectral whitening according to (4.18) is considered to be advantageous over e.g. inverse linear-prediction filtering.

The non-linear **half-wave rectification** (HWR) operation, as defined in (3.5), is a core part of the unitary model. In [Dav87] it has been shown that for an input zero-mean gaussian random process $x(t)$, an approximate expression for the power spectral density $\Psi_y(f)$ of the output $y(t) = \mathrm{HWR}(x(t))$ of the half-wave rectifier can be written as

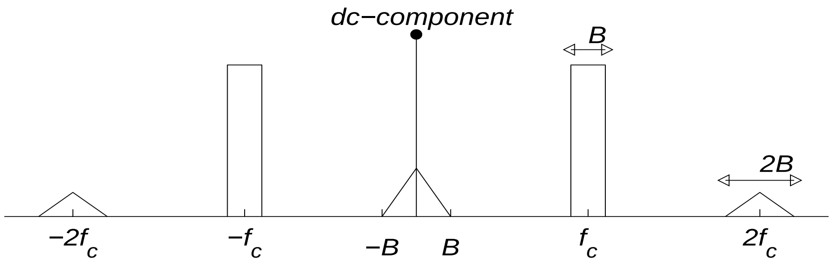

Figure 16. Power spectrum of a half-wave rectified narrowband signal which is centered on frequency f_c and has a bandwidth B (after [Dav87]).

$$\Psi_y(f) \approx \frac{\sigma_x^2}{2\pi}\delta(f) + \frac{1}{4}\Psi_x(f) + \frac{1}{4\pi\sigma_x^2}\int_{-\infty}^{\infty}\Psi_x(f')\Psi_x(f-f')df', \tag{4.20}$$

where $\Psi_x(f)$ is the power spectral density of the input and $\delta(f)$ is the unit impulse function. The approximation involved in the above expression is that higher-order convolution terms have been omitted due to their small powers. As shown in [Dav87], this leads to an error in the output variance which is less than 3%.

Figure 16 illustrates the magnitude spectrum of the output of the half-wave rectifier in the case that the input process has a narrow-band rectangular spectrum. (It should be noted that (4.20) does *not* assume a narrowband signal although these are of interest here.) For a narrowband signal $x_c(n)$ centered on f_c, the output spectrum consists of a dc-component, of the original power spectrum scaled down by four, and of a convolution of the original power spectrum by itself which produces spectral components centered on zero-frequency and twice the input center frequency. In principle, HWR generates spectral components also to bands centered on all integer multiples of f_c without upper limit, but the other bands are very small in power [Dav87].

In the unitary pitch model, phases play an important role and therefore the approximation derived for the power spectral densities in (4.20) is not sufficient. In terms of complex-valued discrete Fourier spectra, the approximation becomes the following[1]. Let

$$X(k) = \sum\nolimits_{n=1}^{K} x(n)e^{-i2\pi(n-1)k/K} \tag{4.21}$$

be the short-time Fourier transform of time-domain signal $x(n)$ before rectification and

$$W(k) = \sum\nolimits_{n=1}^{K} \mathrm{HWR}[x(n)]e^{-i2\pi(n-1)k/K} \tag{4.22}$$

be the short-time Fourier transform of the rectified time-domain signal. We use $\hat{W}(k)$ to denote an approximation of the spectrum $W(k)$. The approximation becomes

$$\hat{W}(k) = \frac{\sigma_x}{\sqrt{8\pi}}\delta(k) + \frac{1}{2}X(k) + \frac{1}{\sigma_x\sqrt{8\pi}}\sum_{l=-K/2+k}^{K/2-k} X(l)X(k-l), \tag{4.23}$$

where σ_x is the standard deviation of the time-domain signal $x(n)$ before rectification[2].

In the present context, the signals of interest are composed of a limited number of sinusoidal

[1] Deriving these approximations is rather straightforward, although not trivial, based on the standard analyses in [Dav87]. An interested reader is referred to Chapter 12 in the cited book.

components and do not satisfy the gaussianity assumption very well. Despite that, the approximation in (4.23) turned out to be quite accurate, as can be easily verified experimentally. Certain complications arise, however, if the time-domain signal $x(n)$ is windowed prior to the rectification. This is because time-domain windowing violates the stationarity assumption of the model. The problem can be easily circumvented by performing windowing as a convolution in the frequency domain *after* computing $\hat{W}(k)$. That is, $\hat{W}(k)$ is convolved with the Fourier spectrum of the window function. (In the time domain, this is achieved simply by applying windowing as usual when computing the term $(1/2)X(k)$ in (4.23) but using the square root of the window function to obtain $X'(k)$ which is used in the place of $X(k)$ for the convolution term in (4.23). These technical details are relatively unimportant from the point of view of the analysis here.)

Let $x_c(n)$ be the time-domain signal at the output of an auditory filter centered on frequency f_c. According to (4.23), the complex spectrum of $x_c(n)$ after rectification can be approximated by

$$\hat{W}_c(k) = \frac{\sigma_c}{\sqrt{8\pi}}\delta(k) + \frac{1}{2}X_c(k) + \frac{1}{\sigma_c\sqrt{8\pi}}V_c(k), \tag{4.24}$$

where we have denoted the convolution term

$$V_c(k) = \sum_{l=-K/2+k}^{K/2-k} X_c(l)X_c(k-l) \tag{4.25}$$

for convenience. An unbiased estimate of the standard deviation σ_c of $x_c(n)$ is obtained based on $V_c(k)$ as

$$\sigma_c = \sqrt{V_c(0)/(K-1)}, \tag{4.26}$$

since $V_c(0)$ gives the power of the signal $x_c(n)$ at channel *c*.

Spectral whitening according to (4.18)–(4.19) can be included by writing

$$\hat{W}_c(k) = \frac{\gamma_c\sigma_c}{\sqrt{8\pi}}\delta(k) + \frac{\gamma_c}{2}X_c(k) + \frac{\gamma_c^2}{\sigma_c\sqrt{8\pi}}V_c(k), \tag{4.27}$$

where the scaling factors γ_c at different channels can be calculated by substituting σ_c to (4.19). Weighting the last term by the square of γ_c stems from the definition of $V_c(k)$ in (4.25).

After compression and half-wave rectification, the subband signals are **lowpass filtered**, as mentioned in the beginning of this subsection. Traditionally, a fixed lowpass filter $H_{LP}(k)$ is used for all different channels and, in addition, the dc-component of the subband signals is removed. More exactly, the filter typically implements a bandpass response with a -3dB cutoff frequencies around 60Hz and 1.0kHz. As a consequence, the dc-term in (4.27) can be omitted and the spectrum at the output of channel *c* can be expressed as

2. At first, the convolution terms in (4.20) and (4.23) may appear as contradictory. However, it has to be remembered that these are merely approximations. The latter approximation is more precise among the two.

$$H_{LP}(k)\hat{W}_c(k) = H_{LP}(k)\left(\frac{\gamma_c}{2}X_c(k) + \frac{\gamma_c^2}{\sigma_c\sqrt{8\pi}}V_c(k)\right). \tag{4.28}$$

4.1.4 Periodicity estimation and across-channel summing (Steps 3 and 4 of the unitary model)

The autocorrelation function (ACF) is equal to the inverse Fourier transform of the power spectrum. Thus the Fourier transform $R_c(k)$ of the autocorrelation function $r_c(\tau)$ at subband c is obtained as

$$R_c(k) = \left|H_{LP}(k)\left(\frac{\gamma_c}{2}X_c(k) + \frac{\gamma_c^2}{\sigma_c\sqrt{8\pi}}V_c(k)\right)\right|^2. \tag{4.29}$$

In practice, the spectra $X_c(k)$ and $V_c(k)$ are non-overlapping at all except few lowest channels and (4.29) can be approximated by

$$\hat{R}_c(k) = \left|H_{LP}(k)\frac{\gamma_c}{2}X_c(k)\right|^2 + \left|H_{LP}(k)\frac{\gamma_c^2}{\sigma_c\sqrt{8\pi}}V_c(k)\right|^2. \tag{4.30}$$

This approximation is valid at channels above about 150Hz but, since we are not interested in reproducing all the nuances and artefacts of the unitary pitch model, we use it at all channels. Also, from the point of view of harmonic sounds, the lowest channels usually do not contain more than one significant frequency component, in which case the magnitude of $V_c(k)$ is very small for $k \neq 0$ (see (4.25)).

Summary autocorrelation function $s(\tau)$ is calculated by summing the within-channel ACF estimates as given by (3.6). Since Fourier transform and its inverse are linear operations, we can sum $R_c(k)$ already in the frequency domain to obtain the Fourier transform of $s(\tau)$,

$$S(k) = \sum\nolimits_c R_c(k) \tag{4.31}$$

and then perform a single inverse Fourier transform to obtain the summary autocorrelation function $s(\tau)$.

Using the approximation in (4.30), the two terms can be squared and summed separately. It follows that (4.31) can be approximated by

$$\hat{S}(k) = |H_{LP}(k)|^2\sum\nolimits_c\left|\frac{\gamma_c}{2}X_c(k)\right|^2 + |H_{LP}(k)|^2\sum\nolimits_c\left|\frac{\gamma_c^2}{\sigma_c\sqrt{8\pi}}V_c(k)\right|^2. \tag{4.32}$$

The bandpass filters are typically designed so that their power responses sum to unity, as mentioned around (4.6). In this case $\sum_c |X_c(k)|^2 = |X(k)|^2$ and, because γ_c is slowly-varying as a function of c, the above formula can be written as

$$\hat{S}(k) = |H_{LP}(k)|^2\left|\frac{\gamma_k}{2}X(k)\right|^2 + |H_{LP}(k)|^2\sum\nolimits_c\left|\frac{\gamma_c^2}{\sigma_c\sqrt{8\pi}}V_c(k)\right|^2, \tag{4.33}$$

where γ_k is the compression-modeling scaling factor corresponding to the frequency bin k. A good approximation of γ_k is obtained by computing the standard deviation of the subband signal at the band centered on the frequency bin k and by substituting this value to (4.19)[1]. The algorithm presented in [P4] computes $V_c(0)$ in $O(K)$ time for all channels c with center frequencies $f_c = (1, 2, \ldots, K/2) f_s/K$, thus the standard deviations needed to compute γ_k are in practice really available. The spectra $X_c(k)$ do not need to be calculated at all.

In the discussion to follow, it will be convenient to denote

$$\bar{X}(k) = \left|\frac{\gamma_k}{2} X(k)\right|^2, \tag{4.34}$$

$$\bar{V}(k) = \sum_c \left|\frac{\gamma_c^2}{\sigma_c \sqrt{8\pi}} V_c(k)\right|^2, \tag{4.35}$$

and to express $\hat{S}(k)$ in two parts as

$$\hat{S}(k) = |H_{LP}(k)|^2 (\bar{X}(k) + \bar{V}(k)). \tag{4.36}$$

Note that $\bar{X}(k)$ is the whitened power spectrum of the input wideband signal.

Finally, the **inverse Fourier transform** is used to obtain an approximation of the summary autocorrelation function $\hat{s}(\tau)$ as

$$\hat{s}(\tau) = \mathrm{IDFT}(\hat{S}(k)). \tag{4.37}$$

Because IDFT is a linear operation, also $\hat{s}(\tau)$ can be decomposed into two terms as

$$\hat{s}(\tau) = \hat{s}_1(\tau) + \hat{s}_2(\tau), \tag{4.38}$$

where

$$\hat{s}_1(\tau) = \mathrm{IDFT}(|H_{LP}(k)|^2 \bar{X}(k)), \tag{4.39}$$

$$\hat{s}_2(\tau) = \mathrm{IDFT}(|H_{LP}(k)|^2 \bar{V}(k)). \tag{4.40}$$

As described around Fig. 10 on page 25, the IDFT performs implicit pattern matching on the power spectrum. In (4.39), pattern matching is performed on harmonic locations of the (whitened) power spectrum $|\bar{X}(k)|^2$ of the input wide-band signal, similarly to the conventional ACF (see Fig. 10). In (4.40), pattern matching is performed on $\bar{V}(k)$ which is the across-channel sum of the power spectra of the amplitude envelopes of the subband signals. The envelope spectra contain frequency components which correspond to the frequency *intervals* in the original spectrum. This was described in Sec. 3.5.1 and is explicitly visible in the definition of $V_c(k)$ in (4.25).

Thus the two types of information, spectral-location information ($\hat{s}_1(\tau)$) and spectral-interval information ($\hat{s}_2(\tau)$) can be kept (approximately) separate until the very end, and can be accumulated separately across bands.

The summary autocorrelation function value $\hat{s}(\tau)$ represents the weight of a period candidate τ, or, the strength of the pitch sensation for pitch period candidate τ. The lag corresponding to the maximum of $\hat{s}(\tau)$ is typically considered as the pitch period.

In the above discussion, $V_c(k)$ has been referred to as the amplitude-envelope spectrum at subband *c*. In addition to that, however, $V_c(k)$ contains the distortion spectrum centered on frequency $2f_c$, and the IDFT operation in (4.40) collects frequency components from both bands. Figure 17 shows an example of the power spectrum $|V_c(k)|^2$ for an artificial subchan-

[1] In principle, γ_k should be computed as a weighted average of several γ_c in the vicinity of *k* but in practice the difference is negligible.

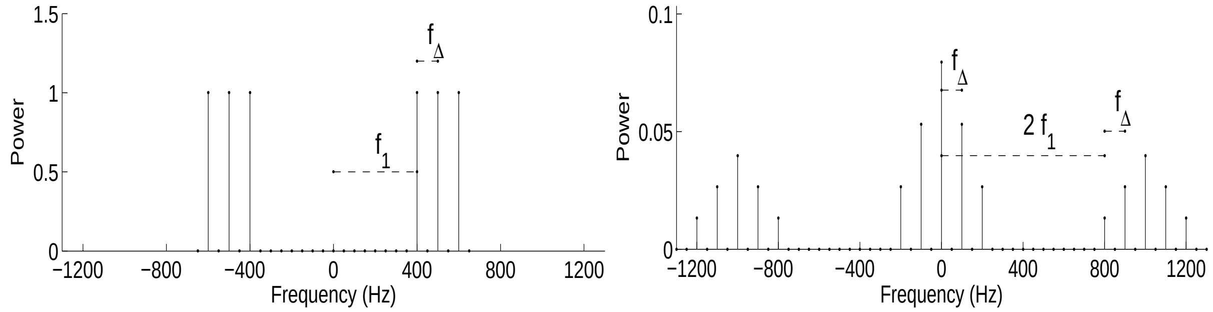

Figure 17. Left: Power spectrum $|X_c(k)|^2$ of an artificial subband signal with 100Hz fundamental frequency. Center frequency of the auditory filter is f_c=500Hz and, for simplicity, a rectangular response is used as the auditory filter. Right: Power spectrum $|V_c(n)|^2$ (defined in (4.25)) corresponding to the power spectrum on the left.

nel signal, the power spectrum of which is shown in the left panel.

There are reasons to believe that the distortion spectrum centered on $2f_c$ does not play any significant role in the unitary model and therefore does not deserve a more detailed analysis. First, the lowpass filter $H_{LP}(k)$ attenuates the distortion spectrum at channels for which $f_c > 500$ Hz. As will be described in Sec. 4.2, these higher channels are the most important for the envelope-related term $\hat{s}_2(\tau)$ because their bandwidth is wider and they typically contain more dense sets of frequency partials. Secondly, the channels below 500Hz contain only harmonic partials whose harmonic index *h* (see (3.1)) is small ($1 \le h \le 10$ even for low-pitched sounds). For these partials, significant inharmonicity is not observed, and therefore the spectrum centered on $2f_c$ consists of frequency components at ideal harmonic position as in Fig. 17. In this case, the IDFT pattern-matcher picks the generated spectral peaks both around zero frequency and around $2f_c$, and the role of the distortion spectrum on $2f_c$ is merely quantitative: to emphasize the information which is already represented by the band around zero frequency. The situation would be more complicated if significant inharmonicity would be observed at the lower channels because, in this case, the peaks around $2f_c$ would not occur at harmonic locations and IDFT would not “match”.

4.1.5 Algorithm proposed in [P4]

Publication [P4] is concerned with the term $\overline{V}(k)$ which consists of the sum of the envelope spectra $V_c(k)$ at different channels as seen in (4.35). Each spectrum $V_c(k)$, in turn, is calculated by convolving the subband spectrum $X_c(k)$ with itself according to (4.25). Thus, it is easy to see that calculating $\overline{V}(k)$ is computationally demanding for a large number of subbands.

The algorithm proposed in [P4] computes $V_c(k)$ for a fixed *k* and for all channels *c* with center frequencies $f_c = (1, 2, \ldots, K/2)f_s/K$ in a time which is linearly proportional to *K* (transfrom length), when given the wideband Fourier spectrum $X(k)$ as input. This means that each frequency bin in $\overline{V}(k)$ can be computed in *O*(*K*) time, given the spectrum $X(k)$. As mentioned in the beginning of this chapter, this has consequences which lead to a computationally efficient unitary-model based F0 estimator.

The basic idea of the algorithm described in [P4] is to compute an estimate of $V_{c+1}(k)$ or $V_{c-1}(k)$ based on $V_c(k)$ in constant time (*O*(1)), with few arithmetic operations. Thus we only need to initialize $V_c(k)$ for $c = 1$ or $c = K/2$ and then iteratively calculate $V_c(k)$ for

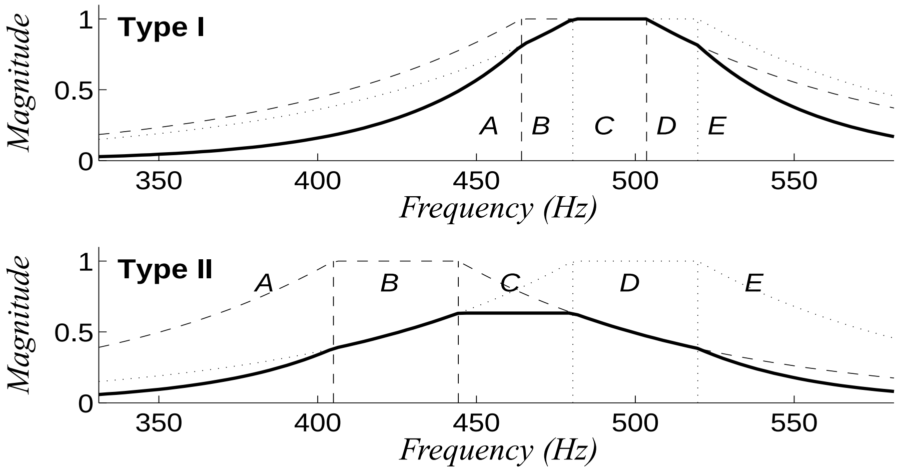

Figure 18. The thick curves in the two panels illustrate the two basic types of the convolved response $J_{c,k}(l)$ given by (4.44). The two responses $H_c(l)$ and $H_c(k-l)$ are shown with dotted and dashed lines, respectively.

all channels *c* by using the update rules proposed in [P4]. As a results, we obtain one spectral line of $\bar{V}(k)$. The update rules were given for the family of flatted-exponential, flex, filters but similar rules could be derived for any filter with a magnitude response which is piecewise representable with exponential functions $(\alpha_1 \exp(\alpha_2 f + \alpha_3))$ as is the case with the flex filter in (4.12).

As a starting point for deriving the update rules, $X_c(k)$ is decomposed as

$$X_c(k) = H_c(k)X(k) \tag{4.41}$$

where $H_c(k)$ is the response of the flex filter centered at f_c. Now (4.25) can be written as

$$V_c(k) = \sum_{l=-K/2+k}^{K/2-k} H_c(l)X(l)H_c(k-l)X(k-l) . \tag{4.42}$$

In the above equation, the convolution spectrum for a given *k*,

$$Z_k(l) = X(l)X(k-l) \tag{4.43}$$

is common to all *c*. This is filtered with a varying *convolved response*

$$J_{c,k}(l) = H_c(l)H_c(k-l) . \tag{4.44}$$

As illustrated in Figure 18, the convolved response can be of two basic types, depending on whether the flatted tops of the two flex responses overlap or not. However, both types consist of five parts (*A*–*E* in Fig. 18) which are of the form of an exponential function $\alpha_1 \exp(\alpha_2 f + \alpha_3)$. The update rules proposed in [P4] are based on this observation. Imagine that we have computed the value of $V_c(k)$ for a certain channel *c* in (4.42) and that the sum over each of the five parts of the convolved response are separately known. Take part *A* as an example. Sum over this part in the neighbouring channel *c*+1 is obtained by “shifting” the part *A* upwards in the spectrum and integrating over that part. The integral for *c*+1 is directly obtained by multiplying the sum in channel *c* with a constant (smaller than one) which causes the old values gradually leak out from the sum and by adding new spectral components when they come under the integral. This way the sum can be updated iteratively over the whole spectrum. More detailed formulas for computing estimates of $V_{c+1}(k)$ or $V_{c-1}(k)$ based on $V_c(k)$ are given in [P4]. Important in doing this is that the iterative calculations are performed so that numerical errors do not cumulate in the calculations.

The computations briefly introduced above lead to a *linear* distribution of subband center frequencies where the center frequency of channel *c* is $f_c = cf_s/K$. The desired channel distribution, however, is uniform on the critical-band scale defined in (4.6). This problem is easily circumvented. An arbitrary distribution of center frequencies can be simulated by weighting the outputs of the linearly-distributed channels accordingly. This is possible when the channel density is sufficiently high, as is the case here. A uniform distribution of channels on the critical-band scale can be simulated by weighting the outputs of the linearly-distributed channels by $1/u_c$ where the bandwidth u_c is computed according to (4.4).

The spectrum $V_c(k)$ at each channel *c* consists of two frequency bands, one centered on zero frequency and another centered on $2f_c$. These two have to be computed separately using the algorithm described in [P4]. However, only the calculations of the band centered on zero frequency were described in [P4] and the band centered on $2f_c$ was ignored. This was not very clearly stated in [P4] and is therefore mentioned in the errata of [P4] on page 111. The band centered on $2f_c$ could be computed exactly in the same way (simply changing $Z_k(l)$ in (4.43)), but since we are interested only in the spectrum of the amplitude envelope centered on zero frequency, this is not presented in more detail. For convenience in the following, we denote

$$V_c'(k) = H_{LP(f_c)}(k)V_c(k), \qquad (4.45)$$

where $H_{LP(f_c)}(k)$ is an ideal lowpass filter with cutoff frequency f_c and zero-valued phase response. That is, $V_c'(k)$ contains only the amplitude envelope spectrum centered on zero frequency and not the distortion spectrum centered on $2f_c$.

4.2 Auditory-model based multiple-F0 estimator

In this section, the above-presented analysis is applied in order to obtain a practically applicable multiple-F0 estimation tool for use in music signals. The method is based on using the whitened wide-band spectrum $\bar{X}(k)$ in (4.34) and the amplitude envelope spectra $V_c'(k)$ in (4.45). The main difference compared to the unitary model is that the ACF calculations in (4.39)–(4.40) (IDFT of the power spectrum) are replaced by a more suitable technique. In the present context, practical F0-estimation accuracy is what counts.

The human auditory system is very good at multiple-F0 estimation. The unitary pitch model, in turn, successfully simulates many of the qualitative properties of human pitch perception, suggesting that the model works in a way similar to its physiological counterpart. Nothing would be more natural than to pursue accurate multiple-F0 estimation by recruiting the unitary model. This has been attempted by us for monophonic signals in [Kla99a] and later for polyphonic mixtures (unpublished). Other authors have reported work to this direction in [Mar96b, deC99, Tol00, Wu02]. However, it has turned out that the unitary model *as such* is not a very accurate multiple-F0 estimator.

In the following, we analyze why the unitary model often fails in practical multiple-F0 estimation tasks. Then, a few modifications are proposed to remove these shortcomings. An overview of the modifications will be given in Sec. 4.2.2. Before that, the backgrounding signal model and the concept *resolvability* will be introduced in the next subsection. Sections 4.2.3–4.2.5 will discuss the proposed modifications in detail. Computational efficiency will be considered in Sec. 4.2.6. Extending a single-F0 estimator to the multiple-F0 case is described in

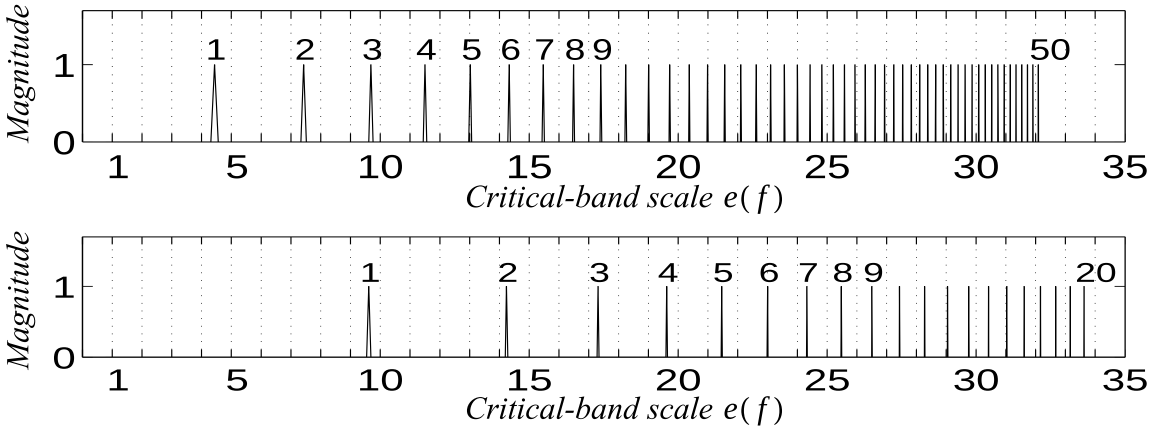

Figure 19. Overtone partials of a harmonic sound on a critical-band scale (see (4.6)). The upper panel shows the first 50 harmonics when F0 is 140Hz. In the lower panel, F0 is 415Hz. The dotted vertical lines indicate the boundaries of adjacent critical bands, i.e., the ERB bandwidths of the auditory filters at these positions.

Sec. 4.2.7.

4.2.1 Harmonic sounds: resolved vs. unresolved partials

In the rest of this chapter, we consider specifically harmonic sounds. The following local model is assumed for the time-domain signal of a harmonic sound

$$x(t) = \sum_{p=1}^{P} a_p \cos(2\pi f_p t + \varphi_p) \tag{4.46}$$

where a_p, f_p, and φ_p are the amplitude, frequency and phase of the p^{th} harmonic partial, respectively, and P is the number of partials. These parameters are assumed to be time-invariant within one analysis frame. Note that (4.46) does not assume perfect harmonicity or that the frequencies f_p would obey the specific expression in (3.1). The basic notion of a harmonic sound involves certain assumptions, however. These were described verbally in Sec. 3.1 and will be stated more exactly when needed in Sec. 4.2.4.

The concept *resolved* vs. *unresolved* harmonic components is very important here. Resolved harmonics refer to partials which are resolved into separate auditory channels: the output of those auditory filters is dominated by one harmonic partial. Unresolved harmonics, in turn, go to a same channel with their neighbouring partials and the frequencies of individual partials are not resolved [Hou90, Moo95b]. Figure 19 shows the overtone partials of a harmonic sound on a critical-band scale (the scale is given by (4.6)). As can be seen, the lowest about ten harmonics are resolved into separate auditory channels, whereas the higher harmonics are not. The situation does not change significantly when F0 is varied. The critical-band range $e(f) \in [1, 35]$ corresponds to frequencies between 25Hz and 10kHz.

In pitch perception, it seems that *spectral locations* of partials are more important for the resolved harmonics, whereas *spectral intervals* are more important for the higher, unresolved, harmonics. This stands to reason since the spectral-interval information appears as beating in the amplitude envelopes but this cannot occur at channels which comprise only one (resolved) partial. Secondly, the precise frequencies and amplitudes of resolved partials are available to the central auditory system, but this is not the case for unresolved partials which go to a same channel with their neighbouring harmonics.

Also simple psychoacoustic experiments support the view that spectral-location information is more important for the low-order (resolved) partials. For example, the pitch of a harmonic sound does not change significantly if its even-numbered harmonics are removed. However, if the odd-numbered harmonics are removed, the pitch doubles because the remaining harmonics correspond to those of a two-times higher sound. This suggests that the spectral locations affect the perception of pitch. However, this is not the case when only higher-order harmonics are involved: the spectral intervals dominate and it becomes impossible to say whether an even-numbered or an odd-numbered series of harmonics is higher in pitch. In [Kla99b], we carried out a small psychoacoustic experiment to determine the limit, up to which the human auditory system discerns the spectral locations of harmonic partials. A total of nine subjects were presented with a pair of harmonic sounds, one composed of a set of five successive odd-numbered harmonics, and another of a set of five successive even-numbered harmonics. The subjects were asked which one of the sounds was higher in pitch. In cases where the series of partials was among the lower harmonics, the subjects perceived a clear octave difference. In the higher end, however, the octave difference disappeared and the subjects selected odd/even series to be higher with 50% probability. I.e., the spectral-location dependency disappeared. In all cases, the F0 of the sound was 137Hz and the order of presentation was randomized.

In the unitary model, the terms $\bar{X}(k)$ and $\bar{V}(k)$ correspond to spectral-location and spectral-interval information, respectively. Based on the above discussion, we make the important interpretation that the term $\bar{X}(k)$ represents mainly the resolved harmonics and the term $\bar{V}(k)$ represents mainly the unresolved harmonics. This interpretation is maintained throughout the analysis in the previous section so that also the terms $X_c(k)$ and $V_c(k)$ in (4.24)–(4.33) are interpreted to be oriented towards resolved and unresolved partials, respectively. The most obvious argument for this is the fact that amplitude envelope beating at the F0 rate, or, significant components in $V_c(k)$, occur only when several partials go to a same auditory channel which is the case for the unresolved harmonics. Finally, $\hat{s}_1(\tau)$ in (4.39) is interpreted to represent mainly the resolved partials and $\hat{s}_2(\tau)$ in (4.40) mainly the unresolved partials.

4.2.2 Overview of the proposed modifications

The modifications to be proposed in the following are directed at the steps 3 and 4 of the unitary model. The ACF calculations are replaced by a technique called *harmonic selection* and more complex subband-weighting is applied when combining the results across bands. As described in Sec. 3.5, the steps 3–4 of the unitary model are a matter of dispute in psychoacoustics and, from the point of view of practical F0-estimation, the attractive properties of the unitary model stem from the steps 1 and 2.

Harmonic selection is here defined as the principle that only a set of selected spectral components are used when computing the “salience” of a F0 candidate, instead of using the overall spectrum. The word *salience* is here used to refer to the calculated weight, or, likelihood, of a F0 candidate. The word “likelihood” is not used because it carries a probabilistic connotation which is not given to the word salience. In multiple-F0 estimation, it is desirable to minimize the interference of other co-occurring sounds and, therefore, it is more robust to use only the partials of a hypothesized F0 candidate to compute its salience and to ignore the spectrum between the partials. In cases where several F0s are present, usage of the overall spectrum is problematic since the salience calculations get confused by the co-occurring sounds and inter-

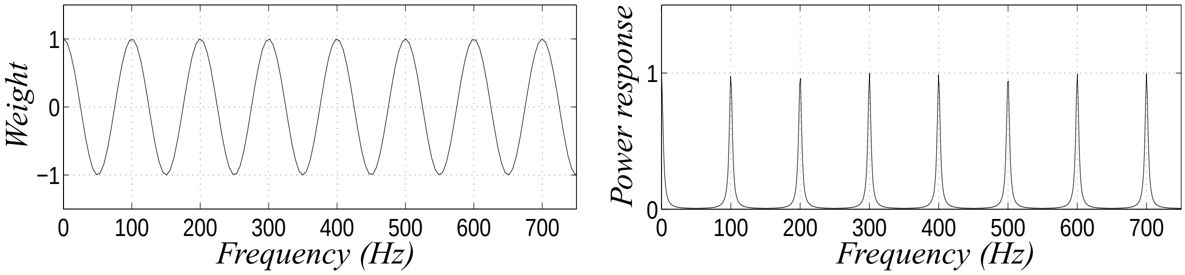

Figure 20. Left panel shows the weights $\cos(2\pi\tau k/K)$ of ACF calculation in (3.3) when the lag τ corresponds to 100Hz (i.e., lag is 10ms). Right panel shows the power response of a comb filter with a feedback delay corresponding to 100Hz and feedback gain 0.85.

relations of different sounds cause unpredictable combination-effects. The general principle of harmonic selection was originally proposed by Parsons in [Par76] and a comprehensive review of the previous harmonic selection and harmonic cancellation methods can be found in [deC93].

In the time domain, harmonic selection can be implemented by using a bank of comb filters and by measuring the energies at the outputs of the filters. Each comb filter is characterized by a certain feedback delay τ and a feedback gain $0 < \alpha < 1$. The output $y_\tau(n)$ of a comb filter with feedback delay τ can be written as

$$y_\tau(n) = (1-\alpha)x(n) + \alpha y_\tau(n-\tau). \tag{4.47}$$

Figure 20 illustrates the power response of a comb filter with τ corresponding to 100Hz and $\alpha = 0.85$. For comparison, the weights of the ACF calculation are shown for the same delay, analogous to Fig. 10 on page 25. Estimating the energy of the output $y_\tau(n)$ can be performed by squaring and summing in the time domain which is equivalent to summing over the power spectrum of the filtered signal.

We implement harmonic selection (i.e., a comb-filter like response) directly in the frequency domain. **For the resolved harmonics**, a frequency bin to represent the harmonic partial *p* of a fundamental frequency candidate $F = f_s/\tau$ is selected from the spectrum as

$$k_{p,\tau} = \underset{k \in K_{p,\tau}}{\arg\max}(|X(k)|), \tag{4.48}$$

where

$$K_{p,\tau} = [k_{p,\tau}^{(0)}, \ldots, k_{p,\tau}^{(1)}], \tag{4.49}$$

$$k_{p,\tau}^{(0)} = \lfloor pK/(\tau + \Delta\tau/2) \rfloor + 1, \tag{4.50}$$

$$k_{p,\tau}^{(1)} = \max\{\lfloor pK/(\tau - \Delta\tau/2) \rfloor, k_{p,\tau}^{(0)}\}. \tag{4.51}$$

Above, $K_{p,\tau}$ defines a range in the vicinity of the ideal harmonic frequency pK/τ where the maximum of $|X(k)|$ is assumed to indicate the partial *p*. The scalar $\Delta\tau$ represents spacing between successive period estimates τ. Here, we use a constant sampling of lag values, $\Delta\tau = 1$, analogous to the ACF. The set $K_{p,\tau}$ is defined so that, for a fixed partial index *p*, all spectral components belong to the range $K_{p,\tau}$ of at least one candidate τ, and the ranges of adjacent period candidates τ and $\tau + \Delta\tau$ cannot overlap by more than one frequency bin.

The selected frequency bins $k_{p,\tau}$ are used to compute a *salience function* $\lambda_1(\tau)$ which replaces $\hat{s}_1(\tau)$ in (4.38). The function $\lambda_1(\tau)$ represents the contribution of resolved harmonics to the salience of a period candidate τ. The function is computed as

$$\lambda_1(\tau) = \sum_p [w(p,\tau)\sqrt{\bar{X}(k_{p,\tau})}], \tag{4.52}$$

where the two-parameter function $w(p,\tau)$ determines the weights of different harmonics in the sum and, as a side-effect, determines the upper limit up to which the partials are considered as “resolved” and thus included in the sum (i.e., the weight approaches zero for large *p*). The sum cannot be extended to arbitrarily large values of *p* because only the lower-order harmonics can be expected to reside at approximately harmonic positions of the spectrum. As will be seen in the next subsection, the “degree of resolvability” of a partial, $w(p,\tau)$, is a two-parameter function, depending on the frequency of the partial *p* and the fundamental frequency f_s/τ.

Applying the weights $w(p,\tau)$ in (4.52) represents another fundamental departure from the unitary model. It should be noted that the summing in (4.52) occurs *across* auditory channels. This is because resolved harmonics, by definition, are located at distinct channels. In hearing, the resolved harmonics are separately coded in the auditory nerve and it is the task of the central auditory system to recombine this information so that the partials of a harmonic sound are perceived as a coherent whole. Thus it is conceivable that some more complex weighting can take place for the resolved harmonics in pitch calculations.

For comparison, let us write out the salience function $\hat{s}_1(\tau)$ which is being replaced by $\lambda_1(\tau)$:

$$\hat{s}_1(\tau) = \frac{1}{K}\sum_{k=0}^{K-1}\left[\cos\left(\frac{2\pi\tau k}{K}\right)|H_{LP}(k)|^2\bar{X}(k)\right]. \tag{4.53}$$

Above, a unity weight is assigned to partials at harmonic positions of the spectrum and, provided that there are partials at these locations, their powers are summed with unity weights up to the cutoff frequency of the lowpass filter $H_{LP}(k)$. This is not very suitable as such[1].

There is a lot of evidence that pitch perception for harmonic sounds resembles a spectral pattern-matching process, especially for the lower-order partials which are resolved into separate auditory channels [Gol73, Wig73, Ter74,82a,b, Har96]. However, the process is not likely to take the form of lowpass filtering followed by ACF calculation but happens in a more complex manner in the central nervous system.

The salience function $\lambda_1(\tau)$ in (4.52) is based on the assumption that the frequencies and amplitudes of individual resolved partials are available to the central auditory system which processes those parameters whatever way it wishes. There is no particular necessity to stick to the ACF. It is important to note that the lowpass filter $H_{LP}(k)$ in (4.53) and the weight function $w(p,\tau)$ in (4.52) model two different things. The filter $H_{LP}(k)$ models the response of the hair-cells which transform mechanical vibration into neural impulses in the inner ear. The weights $w(p,\tau)$, in turn, model a pattern-matching process which is assumed to take place in

[1] Consider a low-pitched sound with F0 100Hz. In polyphonic music, the frequency range between 100Hz and 1.0kHz is heavily occupied by other sounds. Blindly picking components from all harmonic positions and weighting them equally is not robust. The implicit spectral smoothing mechanism described in Sec. 3.5.3 would partly prevent from “stealing” the partials of other sounds, but this mechanism is involved only in the amplitude-envelope related part $\bar{V}(k)$. For a high-pitched sound with F0 1.0kHz, in turn, the lowest and most important harmonic partials are each resolved to separate auditory channels. As such, they cannot generate beating frequencies to $V_c(k)$ and, in (4.53), they are rejected by the lowpass filter. In practice, F0s above about 600Hz are detected very poorly.

the central nervous system. In principle, the filter $H_{LP}(k)$ should be included in (4.52) (the hair-cell response cannot be bypassed), but since $w(p, \tau)$ acts as a lowpass filter, $H_{LP}(k)$ can be omitted.

Another, less important modification is that, in (4.52), magnitude spectrum is used instead of the power spectrum in (4.53). This modification is due to both simplicity and accuracy. The second power stems from the ACF computations and there is no particular reason to maintain it. On the contrary, as discussed below (3.2), it is usually advantageous to use a “generalized ACF” where the exponent is below two. Tolonen and Karjalainen suggest the value 0.67 in [Tol00]. This is a matter of fine-tuning and not important here, therefore the unity value is used for simplicity.

The other part, $\hat{s}_2(\tau)$ in (4.38), represents mainly the **unresolved harmonics** as discussed in the previous subsection. This is replaced by a salience function $\lambda_2(\tau)$ where harmonic selection is applied in a manner analogous to that described above. The salience function $\lambda_2(\tau)$ is defined as

$$\lambda_2(\tau) = \max_{k \in K_{1,\tau}} \left\{ \eta_0 H_{LP}(k) \sum_c \left| \frac{\gamma_c^2}{\sigma_c \sqrt{8\pi}} V_c'(k) \right| \right\}, \tag{4.54}$$

where η_0 is a real-valued constant and $K_{1,\tau}$ is obtained by substituting $p = 1$ in (4.49). First, note that magnitude spectra at different channels c are used, instead of the power spectra as in (4.35). This modification is due to the same reasons as for the $\lambda_1(\tau)$ part. Also, as defined in (4.45), $V_c'(k)$ contains only the amplitude-envelope spectrum centered on zero frequency and not the distortion spectrum centered on $2f_c$. Secondly, only *one* frequency bin is selected in the vicinity of the fundamental frequency K/τ. This can be computed very efficiently using the algorithm in [P4] and leads to a very efficient overall implementation as will be described in Sec. 4.2.6.

Individual frequency bins $V_c'(k_0)$ at each channel c suffice to represent the salience contribution of unresolved harmonics for the fundamental frequency candidate $F = (k_0/K)f_s$. A complete argument for this is given in Sec. 4.2.4 but an intuitive description is given here. Note that the single frequency bin $V_c'(k_0)$ retains all the desirable properties of the unitary pitch model described in Sec. 3.5. First, $V_c'(k_0)$ represents spectral-interval oriented information regarding the fundamental frequency $F = (k_0/K)f_s$. The magnitude of $V_c'(k_0)$ reveals the amount of amplitude-envelope beating at rate F at subband c. The beating, in turn, is due to all frequency components with interval F at channel c. If the fundamental frequency F is present, each two neighbouring harmonics contribute to the beating at the rate F, and the magnitude of $V_c'(k_0)$ is high. Secondly, due to the manner how the amplitude of the beating is formed (see Fig. 12 on page 30), $V_c'(k_0)$ retains the property of implicit spectral smoothing. Thirdly, $V_c'(k_0)$ is phase-dependent because it is computed using the complex Fourier spectrum, according to (4.25) and (4.45). If the harmonics of a sound are phase-locked to each other (linear phase, i.e., the phase difference between each pair of neighbouring harmonics is approximately constant), the magnitude of the beating at the fundamental rate is larger.

The overall salience of a period candidate τ is then defined (analogously to (4.38)) as:

$$\lambda(\tau) = \lambda_1(\tau) + \lambda_2(\tau). \tag{4.55}$$

The maximum of $\lambda(\tau)$ is used to indicate the fundamental period. The corresponding fundamental frequency is $F = f_s/\tau$. This represents one detected F0 in a mixture signal. An extension to multiple-F0 estimation will be described in Sec. 4.2.7.

4.2.3 Degree of resolvability $w(p, \tau)$

This subsection is concerned with the coefficient $w(p, \tau)$ in (4.52). The coefficient was interpreted as the *degree of resolvability* for harmonic *p* of fundamental period candidate τ. The salience function $\lambda_1(\tau)$ was calculated by selecting frequency components nearby harmonic spectral locations, by weighting their magnitudes by $w(p, \tau)$, and by summing these. Weighting different partials *p* by their resolvability is motivated by the fact that the spectral locations of partials are important only in the case of resolved partials, as discussed in Sec. 4.2.1.

The boundary between resolved and unresolved partials is not sharp but we have to speak about “degree of resolvability”. How could this be measured? For the overtone partials of a harmonic sound, resolvability depends on the F0 and on the harmonic index *p* of the partial. This can be seen by estimating the number of partials which go to a same auditory channel together with harmonic number *p*. The frequency interval between partials is determined quite accurately by the the fundamental frequency *F*, and the ERB bandwidth of the auditory channel around harmonic *p* can be estimated by substituting $f_c = pF$ to (4.4). The ratio of these can be used to estimate the number of partials $\Upsilon(p, F)$ going to a same auditory channel with harmonic *p*:

$$\Upsilon(p, F) = \frac{24.7[1 + 4.37pF/1000]}{F}. \qquad (4.56)$$

As can be seen, $\Upsilon(p, F)$ grows linearly as a function of the harmonic index *p*. (Note that the above simple ratio gets values below unity when the inter-partial interval is larger than the ERB value.)

We propose to model the degree of resolvability as proportional to the inverse of $\Upsilon(p, F)$:

$$w(p, F) = w_0 \frac{F}{24.7[1 + 4.37pF/1000]}, \qquad (4.57)$$

where the scalar w_0 is an unknown parameter that has to be experimentally found. Apart from saying that $w(p, F)$ represents the degree of resolvability, the function is not given any exact psychoacoustic interpretation. It has merely practical use. Note that the above ratio (excluding w_0) gets values above unity for the few lowest harmonics, meaning that some auditory channels around those partials do not contain partials at all. Although the neighbouring channels actually should not affect the resolvability, for simplicity, the measure in (4.57) is used in the following without limiting the maximum value of the above ratio to a unity value.

The function $w(p, F)$ controls the contribution of partial *p* to the salience function $\lambda_1(\tau)$. Figure 21 illustrates the values of $w(p, F)$ for the first 20 harmonics of a few fundamental frequency values. As can be seen, the contribution of partial *p* to the salience function $\lambda_1(\tau)$ decreases as a function of the harmonic index *p*. Here, more or less arbitrarily, we use $w_0 = 0.5$.

It is interesting to note that the contribution of an overtone partial *p* to the other salience function, $\lambda_2(\tau)$, *increases* as a function of *p*. This occurs although no weighting function is explicitly included in (4.54). As described around (4.54), a single frequency bin $V_c'(k_0)$ is selected to represent the salience contribution of unresolved harmonics to the fundamental frequency candidate $F = (k_0/K)f_s$ at channel *c*. According to the decomposition in (4.42), $V_c'(k_0)$

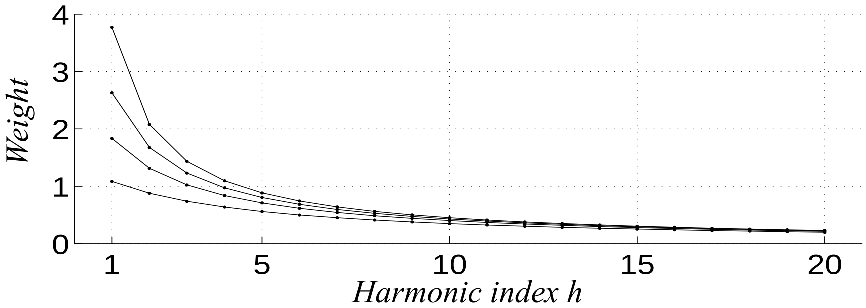

Figure 21. Values of $w(p, F)$ for the first 20 harmonic overtones of a few F0 values. The curves (bottom-up) correspond to F0 values 70Hz, 150Hz, 300Hz, and 1.0kHz, respectively. The applied value of the scalar w_0 is 0.5.

can be calculated by summing over *l* in $Z_{k_0}(l)J_{c,k_0}(l)$, where $Z_{k_0}(l)$ is the convolution spectrum and $J_{c,k_0}(l)$ is the convolved response at channel *c*. For a fixed fundamental frequency corresponding to k_0, the term $Z_{k_0}(l)$ remains the same at different channels. However, the term $J_{c,k_0}(l) = H_c(l)H_c(k_0 - l)$ varies along with the channel index *c*. As can be seen in Figure 18, if k_0 is small compared to the bandwidth of $H_c(l)$, the passbands of $H_c(l)$ and $H_c(k_0 - l)$ overlap a lot in the convolution and the overall power transmission of the convolved response $J_{c,k_0}(l)$ is high. Because the bandwidth increases linearly as a function of the center frequency (see (4.4)), the power transmission of $J_{c,k_0}(l)$ *increases with frequency.* This has the consequence that the contribution of the harmonic series of a sound to the salience function $\lambda_2(\tau)$ *increases as a function of the harmonic index p*. On the other hand, for a fixed subband *c*, the power response of $J_{c,k_0}(l)$ decreases with an increasing k_0 (increasing fundamental frequency).

Based on the above description, we can express the average spectral density $\xi(f_c, F)$ of the convolved response $J_{c,k}(l)$ at channel centered on f_c and for fundamental frequency *F* as

$$\xi(f_c, F) = \frac{\sum_l J_{c,k_F}(l)}{\sum_l J_{c,0}(l)} = \frac{\sum_l H_c(l)H_c(k_F - l)}{\sum_l H_c(l)H_c(-l)}, \tag{4.58}$$

where $k_F = (F / f_s)K$ is the frequency bin corresponding to F. Note that the denominator represents the ERB value (i.e., bandwidth) of the flex response at channel *c*. Normalization with the bandwidth is important in order to compensates for the effect which is merely due to the fact that the ERB values increase with frequency. That is, the average *spectral density* of the convolved response, not the overall power transmission, acts as the weight for individual harmonic partials that reside at that particular channel.

The value $\xi(pF, F)$ represents an implicit weight which affects the salience contribution of an overtone partial *p* to the function $\lambda_2(\tau)$. Figure 22 illustrates $\xi(f_c, F)$ for the first 20 harmonics of a few fundamental frequency values. The curves have been obtained by substituting $f_c = pF$ in (4.58). As can be seen, the salience contribution increases monothonically with harmonic index *p* and saturates to unity. As a consequence, the function $\lambda_2(\tau)$ indeed represents mainly the unresolved partials. No upper limit for the harmonic index needs to be set. In practice, however, significant harmonic components are not observed above 6–8kHz and it does not make sense to consider subbands with center frequencies above this.

It seems reasonable that the functions $w(p, F)$ and $\xi(pF, F)$ would sum approximately to

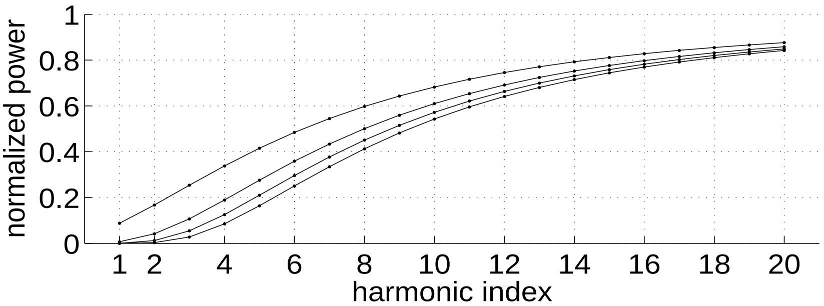

Figure 22. The function $\xi(pF, F)$ for the first 20 harmonic overtones of a few F0 values. The function reflects the contribution of the harmonics to the salience function $\lambda_2(\tau)$. The curves (top-down) correspond to F0 values 70Hz, 150Hz, 300Hz, and 1.0kHz, respectively.

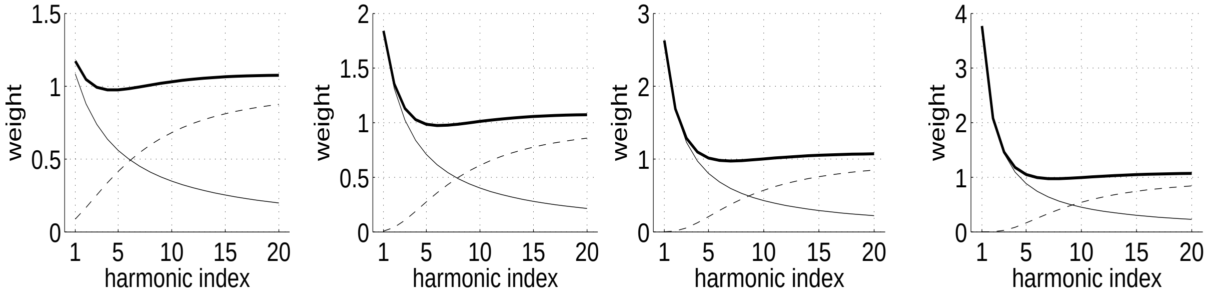

Figure 23. Thin solid line shows $w(p, F)$ (with $w_0 = 0.5$) as a function of the harmonic index *p*. Dashed line shows $\xi(pF, F)$. The sum of the two functions is indicated by thick solid line. F0 values are 70Hz, 150Hz, 300Hz, and 1.0kHz, from left to right, respectively.

unity over the harmonic series of a sound, so that the harmonics which contribute only little to the salience function $\lambda_1(\tau)$ would contribute more to the salience function $\lambda_2(\tau)$, implementing a smooth transition between the resolved and unresolved partials.

The thin solid line in Fig. 23 illustrates the degree of resolvability $w(p, F)$ (with $w_0 = 0.5$) as a function of the harmonic index *p*. Along with that, the dashed line shows $\xi(pF, F)$ as a function of *p*. The sum of the two is drawn with a thicker line. As can be seen, the two functions sum approximately to unity over the harmonic series of a sound. For the lowest few harmonics of high-pitched sounds, however, the sum clearly exceeds the unity value. This is because the modeled resolvability $w(p, F)$ in (4.57) was not limited to values below unity. However, the additional boost for the few lowest harmonics of high-pitched sounds turned out to be a good feature, since the high-pitched sounds typically have only few harmonic components altogether, and the boost compensates for this.

Table 3 shows the pseudocode of the algorithm which computes the salience function $\lambda_1(\tau)$. The algorithm has several nice properties. First, it is not computationally very complex, since it is safe to consider only the first 20 harmonics of each period candidate due to the fact that the weight $w(p, F)$ becomes quite small above this. When the partial index *p* is fixed, the number of elements in the sets $[k^{(0)}, \ldots, k^{(1)}]$ for different $\tau \in T$ altogether is less than $K + |T|$, where $|T|$ is the number of period candidates and *K* is the transform length. Thus, the overall complexity of the algorithm is of the order $O(20(K + |T|)) = O(K + |T|)$ when a fixed number of 20 partials is considered. Secondly, only the selected frequency components affect the salience of a F0 candidate, not the overall spectrum. This provides some robustness in

Table 3: Algorithm for computing the salience function $\lambda_1(\tau)$.

```
# Compute saliences $\lambda_1(\tau)$ of fundamental period candidates $\tau$
for $\tau \leftarrow$ from 2 to $\tau_{\max}$ with stepsize $\Delta\tau$ do
    $\lambda_1(\tau) \leftarrow 0$
    $F \leftarrow f_s/\tau$
    $p \leftarrow 1$
    while $p \le 20$ and $\lfloor p(K/\tau + \Delta\tau/2) \rfloor < K/2$ do
        # Harmonic selection: find maximum amplitude in the specified range
        $k^{(0)} = \lfloor pK/(\tau + \Delta\tau/2) \rfloor + 1$
        $k^{(1)} = \lfloor pK/(\tau - \Delta\tau/2) \rfloor$
        if $k^{(1)} < k^{(0)}$ then
            $k^{(1)} \leftarrow k^{(0)}$
        end if
        $k' \leftarrow \arg\max\{|X(k^{(0)})|, \ldots, |X(k^{(1)})|\}$
        # Cumulate partial amplitudes to $\lambda_1(\tau)$
        # Term $w(p, F)$ represents the degree of resolvability (see (4.57))
        # Term $\gamma_{k'}$ is due to compression modeling (see (4.19))
        $\lambda_1(\tau) \leftarrow \lambda_1(\tau) + w(p, F)\gamma_{k'}|X(k')|$
        $p \leftarrow p + 1$
    end while
end for
```

polyphony. Thirdly, only the lower-order partials are selected according to their ideal spectral locations and this is appropriate even for sounds that exhibit some inharmonicity. Finally, the algorithm does not require an unrealistic frequency resolution: the spectral band $[k^{(0)}, \ldots, k^{(1)}]$ is never extremely narrow in relation to its center frequency because only the lowest 20 harmonics are considered. This is not only a matter of psychoacoustic plausibility but leads to an algorithm which works accurately in relatively short analysis frames.

4.2.4 Assumptions underlying the definition of $\lambda_2(\tau)$

The aim of this subsection is to describe the assumptions that underlie the definition of $\lambda_2(\tau)$ in (4.54). In particular, the purpose is to explain why individual frequency bins $V_c'(k_0)$ at each channel *c* can be used to represent the contribution of unresolved partials to the salience of fundamental frequency $F = (k_0/K)f_s$. The assumptions culminate to the question *what exactly is assumed about the input signals, i.e., about harmonic sounds*. This class of sounds was defined merely verbally and through examples in Sec. 3.1. We show that by making certain assumptions about the input signals, the amplitudes a_p of the unresolved harmonics of fundamental frequency $F = (k_0/K)f_s$ at the subband *c* can be estimated using only the single frequency bin $V_c'(k_0)$.

Let us consider the model for the time-domain signal $x(t)$ of a harmonic sound in (4.46). The power spectral density of the signal can be written as

$$\Psi_x(f) = \sum_{p \in N} a_p^2 \delta(f - f_p), \tag{4.59}$$

where δ is the unit impulse function and the set *N* is defined to include both positive and negative indexes, $N = [-P, -1] \cup [1, P]$. Negative indexes are used to express negative frequencies, i.e., $f_{-p} = -f_p$ and $a_{-p} = a_p$ for $n \in N$.

When the signal $x(t)$ is passed through a critical-band auditory filter with a squared magnitude response $\Psi_{h_c}(f)$ as defined in (4.12), the output of the filter can be denoted by $\Psi_{x_c}(f) = \Psi_x(f)\Psi_{h_c}(f)$. Using the signal model in (4.59), this can be written as

$$\Psi_{x_c}(f) = \sum_{p \in N} a_p^2 \delta(f - f_p) \Psi_{h_c}(f_p). \tag{4.60}$$

When half-wave rectification is applied on the subband signal $x_c(t)$, the power spectral density of the resulting rectified signal $y_c(t)$ can be approximated using (4.20) as

$$\Psi_{y_c}(f) = \frac{\sigma_c^2}{2\pi}\delta(f) + \frac{1}{4}\sum_{p \in N} a_p^2 \delta(f - f_p)\Psi_{h_c}(f_p) \tag{4.61}$$

$$+ \frac{1}{4\pi\sigma_c^2}\sum_{i \in N}\sum_{j \in N}\delta(f - f_i + f_j) a_i^2 a_j^2 \Psi_{h_c}(f_i)\Psi_{h_c}(f_j)$$

where σ_c^2 is the variance of the subband signal $x_c(t)$.

We denote the convolution term[1] in the above expression by $\Psi_{\hat{v}_c}(f)$:

$$\Psi_{\hat{v}_c}(f) = \frac{1}{4\pi\sigma_c^2}\sum_{i \in N}\sum_{j \in N}\delta(f - f_i + f_j) a_i^2 a_j^2 \Psi_{h_c}(f_i)\Psi_{h_c}(f_j). \tag{4.62}$$

Examples of the power spectra $\Psi_{x_c}(f)$ and $\Psi_{\hat{v}_c}(f)$ were illustrated in Fig. 17 on page 42.

The first necessary assumption concerning the input harmonic sounds is that the frequency interval between adjacent harmonics, $f_\Delta = f_{p+1} - f_p$, remains approximately constant within one critical band. This assumption is reasonable for the kind of inharmonicity that we consider in this work (dispersive strings etc.). Even when sounds exhibit inharmonicity, the spectral intervals are slowly-varying as a function of frequency (see e.g. (3.1)) and can be assumed to be piecewise constant at sufficiently narrow bands. Using this assumption, the single spectral line $\Psi_{\hat{v}_c}(f_\Delta)$ at subband c can be calculated as

$$\Psi_{\hat{v}_c}(f_\Delta) = \frac{2}{4\pi\sigma_c^2}\sum_{p=1}^{P-1} a_p^2 a_{p+1}^2 \Psi_{h_c}(f_p)\Psi_{h_c}(f_{p+1}), \tag{4.63}$$

when $f_\Delta = f_{p+1} - f_p$ is constant at the subband c.

Secondly, we know that for the higher-order unresolved harmonics, the human auditory system does not distinguish the amplitudes of individual harmonic components. Instead, the rough spectral shape of several components is perceived (approximately one level measure per a critical band and a distinct sound source). If one harmonic raises clearly above the other partials, it is usually perceptually segregated and stands out as an independent sound. This feature of hearing is well modeled by the unitary pitch model as discussed in Sec. 3.5.

In the computational analysis of sounds, it is useful to make a “spectral smoothness” assumption similar to that in human hearing. More specifically, we assume that the partial amplitudes a_p within one critical band c centered on frequency f_c can be approximated by a single level

[1] Due to the differences in the two approximations (4.20) and (4.23), $\Psi_{\hat{v}_c}(f)$ is not the exact power spectrum of $V_c(k)$ as defined in (4.25). Instead, $\Psi_{\hat{v}_c}(f)$ is an approximation. For this reason, the notation $\Psi_{\hat{v}_c}(f)$ is used instead of $\Psi_{v_c}(f)$.

measure A_c so that

$$a_p \approx A_c, \text{ when } |f_p - f_c| < u_c, \tag{4.64}$$

where the ERB bandwidth u_c is given by (4.4).

Why should we assume spectral smoothness? This argument requires some elaboration. Here it suffices to consider musical instruments specifically. It is generally known that high-quality synthesis of harmonic sounds can be achieved by employing only one time-varying level measure per each critical band. From musical instrument construction point of view, in turn, if one harmonic raises above the other partials, it is perceptually segregated and no more perceived to belong to the complex. This is an unwanted effect and typically avoided in instrument design. The spectrum of musical sounds depends on their physical sound production mechanism. Many instruments can be seen to consist of two acoustically coupled parts, a vibrating source (e.g. a string or an air column) and a sympathetic resonator (such as the guitar body or the piano soundboard). It is theoretically possible to make the vibrating source vibrate in only one of its vibration modes (frequencies) by e.g. playing a sinusoid of certain frequency nearby a string [Ros90]. More commonly, however, the excitation signal to a vibrating system resembles a transient signal or an impulse train, resulting in a spectrum where no individual harmonic stands out. For a plucked string, the spectrum in the beginning of the vibration corresponds to the Fourier transform of the shape of the displaced string just before it is released [Fle98, p.41]. The vibration spectrum is then filtered by the frequency response of the coupled body resonator. It is musically undesirable that the symphatetic resonator would have very sharp resonance modes (formants) but usually the resonator is strongly damped, radiates acoustic energy efficiently, and has a smooth spectrum[1].

By making the spectral smoothness assumption, (4.63) can be written as

$$\Psi_{\hat{v}_c}(f_\Delta) = \frac{A_c^4}{2\pi\sigma_c^2} \sum_{p=1}^{P-1} \Psi_{h_c}(f_p)\Psi_{h_c}(f_{p+1}) \tag{4.65}$$

From which A_c can be solved as

$$A_c^4 = \frac{\Psi_{\hat{v}_c}(f_\Delta)2\pi\sigma_c^2}{\sum_{p=1}^{P-1} \Psi_{h_c}(f_p)\Psi_{h_c}(f_{p+1})}. \tag{4.66}$$

It should be noted that the frequencies f_p of individual unresolved partials are not known. However, at bands that contain unresolved components, the partial density is sufficiently high to “sample” the squared frequency response $\Psi_{h_c}(f)$ so that the exact spectral positions of the harmonics become insignificant. In other words, harmonics with f_Δ inter-partial distance are distributed all over the response $\Psi_{h_c}(f)$. When the “sampling interval” f_Δ gets smaller, the exact “sampling positions” do not matter, and (4.66) approaches the limit

[1] In regard to spectral smoothness, speech and singing signals are an important borderline case. The *Q*-values of the lowest formants of vowel sounds are between 8-15, and those of the higher formants between 15-30 [Dun61; Ste98,p.258]. For comparison, the *Q*-values of the auditory filters are around 8 for channels above 1kHz and vary between 3 and 8 below that. The *Q*-value of a bandpass filter is defined as the ratio of the center frequency to the -3dB bandwidth of the filter. Thus the higher the *Q*-value of a filter, the sharper the shape of its magnitude response.

$$A_c^4 \approx \frac{\Psi_{\hat{v}_c}(f_\Delta) 2\pi\sigma_c^2}{\frac{1}{f_\Delta}\int_f \Psi_{h_c}(f)\Psi_{h_c}(f + f_\Delta)df} . \tag{4.67}$$

Here, the frequencies of the individual partials need not be known.

In conclusion, the single frequency bin $\Psi_{\hat{v}_c}(f_\Delta)$ suffices to represent the level of the partials of fundamental frequency $F = (k_0/K)f_s$ at subband c. Of course, this result depends on the validity of the above assumptions and of the approximation for a half-wave rectified signal given by (4.20). Virtanen and Klapuri used the described assumptions and (4.20) to estimate the parameters of the unresolved harmonic components in a sound separation system (unpublished at the present time). In resynthesis, good perceptual quality was achieved for the higher-order unresolved partials using the described model and assumptions.

Provided that an input sound indeed has sinusoidal partials with constant intervals f_Δ at channel c, it is easy to see that the convolution term $\Psi_{\hat{v}_c}$ contains partials also at the multiples mf_Δ, where integer $m = 1, 2, 3, \ldots$. However, it is not necessary to extend the harmonic selection so that also these components would be picked from $\Psi_{\hat{v}_c}$ and summed. In fact, the amplitudes A_c given by (4.67) can be used to *predict* (approximate) the magnitudes of the spectral lines $\Psi_{\hat{v}_c}(mf_\Delta)$, for $m > 1$ as:

$$\begin{aligned} \Psi_{\hat{v}_c}(mf_\Delta) &= \frac{A_c^4}{2\pi\sigma_c^2} \sum_{p=1}^{P-m} \Psi_{h_c}(f_p)\Psi_{h_c}(f_{p+m}) \\ &\approx \frac{A_c^4}{2\pi\sigma_c^2}\left(\frac{1}{f_\Delta}\right)\int_f \Psi_{h_c}(f)\Psi_{h_c}(f + mf_\Delta)df \end{aligned} \tag{4.68}$$

Substituting A_c^4 from (4.67) this can be written as

$$\Psi_{\hat{v}_c}(mf_\Delta) \approx \Psi_{\hat{v}_c}(f_\Delta)\frac{\int_f \Psi_{h_c}(f)\Psi_{h_c}(f + mf_\Delta)df}{\int_f \Psi_{h_c}(f)\Psi_{h_c}(f + f_\Delta)df} . \tag{4.69}$$

Let us see what difference it would make to select and sum up all the harmonically related partials $\Psi_{\hat{v}_c}(mf_\Delta)$, where $m = 1, 2, 3, \ldots$. The sum would be

$$\sum_m \Psi_{\hat{v}_c}(mf_\Delta) \approx \Psi_{\hat{v}_c}(f_\Delta)\sum_m \frac{\int_f \Psi_{h_c}(f)\Psi_{h_c}(f + mf_\Delta)df}{\int_f \Psi_{h_c}(f)\Psi_{h_c}(f + f_\Delta)df} . \tag{4.70}$$

The sum on the right-hand side of the above equation merely amounts to a very complicated weighting of the individual spectrum line $\Psi_{\hat{v}_c}(f_\Delta)$. Such weighting turned out to be completely unnecessary.

The purpose of the above analysis was to motivate the use of only one frequency bin $V_c'(k_0)$ at each channel c to represent the contribution of unresolved partials to the salience of fundamental frequency candidate $F = (k_0/K)f_s$ at those channels. As already discussed in Sec. 4.2.2, the single frequency bin $V_c'(k_0)$ retains the desirable properties of the unitary pitch model.

For convenience, let us rewrite here the definition of $\lambda_2(\tau)$ from (4.54):

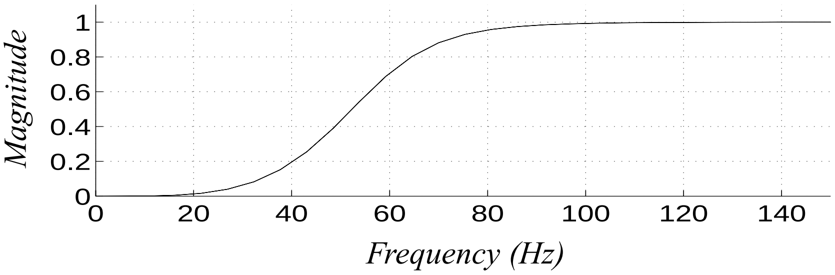

Figure 24. Magnitude response of the filter $H_{LP}(k)$ applied on $V_c'(k)$ in (4.71). Due to the narrowband nature of $V_c'(k)$, characteristics of the filter above 1kHz are not important.

$$\lambda_2(\tau) = \max_{k \in K_{1,\tau}} \left\{ \eta_0 H_{LP}(k) \sum_c \left| \frac{\gamma_c^2}{\sigma_c \sqrt{8\pi}} V_c'(k) \right| \right\}. \tag{4.71}$$

Note that in the above equation, the values $V_c'(k)$ are used *directly* and are not used to estimate the magnitudes of the individual partials $a_p \approx A_c$ according to (4.67). Using the magnitudes of the partials, a_p, would omit the implicit “weighting” of individual harmonics by $\xi(pF, F)$ as described in the previous subsection and in Figs. 22–23. This would not be appropriate since $\lambda_2(\tau)$, by definition, should represent mainly the *unresolved* harmonics. This, in turn, is because $V_c'(k)$ was interpreted to represent mainly the unresolved harmonics.

The filter $H_{LP}(k)$ in (4.71) cannot be omitted. As described in the end of Sec. 4.1.3, the rectified signal contains a significant dc-component and, in order to remove this, the filter $H_{LP}(k)$ typically implements a bandpass response with -3dB cutoffs around 60Hz and 1kHz. Since we are using $V_c'(k)$ instead of $V_c(k)$, only the highpass characteristics of the filter are important. (The term $V_c'(k)$ contains only the amplitude-envelope spectrum centered on zero frequency and not the distortion spectrum centered on $2f_c$.) Figure 24 illustrates the frequency response of the applied butterworth filter of order four per each transition band and with the -3dB cutoff at 60Hz. The scaling factor η_0 in (4.71) is a free parameter which will be determined in the next subsection.

4.2.5 Model parameters

The overall salience $\lambda(\tau)$ of different fundamental period candidates is given by (4.55). A nice property of the overall model is that it contains only few free parameters. However, an important parameter which remains to be determined is the relative weight of the two parts $\lambda_1(\tau)$ and $\lambda_2(\tau)$. Note that both parts include an unknown scaling factor: w_0 for $\lambda_1(\tau)$ in (4.57) and η_0 for $\lambda_2(\tau)$ in (4.54). The numerical ranges of the two parts have to be matched in order that both terms would have an appropriate effect on $\lambda(\tau)$. It suffices to balance the levels of $\lambda_1(\tau)$ and $\lambda_2(\tau)$ in relation to each other since the absolute numerical range of $\lambda(\tau)$ is not important. Thus, for simplicity, we fix $\eta_0 = 1$ and consider only w_0 as a free parameter.

Another important parameter is the value v of the v^{th}-law compression in (4.19). Thirdly, the widths of the subbands u_c were found to be an important tunable parameter in the model. We applied a scalar u_0 so that the used bandwidths were $u_0 u_c$, where u_c is according to (4.4). In other words, the factor scales the defined critical bandwidths and this has an effect everywhere where these bandwidths are used, without having to modify the other parts.

The parameters w_0, v, and u_0 were found experimentally in simulations. Musical instrument samples were randomized from an acoustic database and the accuracy of the F0 estimation method for this material was evaluated using different parameter values. The database comprised a total of 2536 individual recorded samples from 30 different musical instruments (the same database was used in [P5]). Random mixtures of sounds were generated by first allotting an instrument and then a random note from its whole playing range, restricting, however, the pitch over five octaves between 65Hz and 2100Hz. The desired number of simultaneous sounds was allotted and then mixed with equal mean-square levels. The acoustic input was fed to the F0 estimation method which estimated the F0 in one 93ms time frame, 100ms after the onset of the sounds. The sampling rate was 44.1kHz.

In polyphonic mixtures, we used the *predominant-F0 estimation* accuracy as the criterion for parameter selection. In predominant-F0 estimation, the F0 estimate is defined to be correct if it matches the correct F0 of any of the component sounds. In other words, only one F0 was being estimated. A correct F0 was defined to deviate less than 3% from the pitch of the reference musical note. Multiple-F0 estimation, i.e., estimation of the F0s of all the component sounds, will be considered in Sec. 4.2.7.

Note that in these experiments, $\lambda(\tau)$ was computed for all $\tau \in T$ where the set T comprised lag values between 50Hz and 5000Hz. This requires computing $V_c'(k)$ for all k and all c to obtain $\lambda_2(\tau)$ in (4.54). As the complexity of computing $V_c'(k)$ for *one* k and all c is proportional to the frame length K, the complexity of the overall algorithm becomes high $(O(K^2 + |T|)$, where $|T|$ is the cardinality of the set T). However, computational efficiency is not the main concern now. A solution which overcomes this and leads to an efficient algorithm will be described in the next subsection.

A three-dimensional search was conducted to find the values of the parameters w_0, v, u_0. For v, the values 0.10, 0.20, 0.33, 0.50, 0.66, and 1.0 were considered. For u_0, only the values 1.0, 1.5, and 2.0 were considered. The weighting factor w_0 was varied in a more continuous manner for each combination of v and u_0.

As a result of the experiments, the combination of v and u_0 which performed best was $v = 0.33$ and $u_0 = 1.5$. These values will be fixed in the following. The value combination $v = 0.5$ and $u_0 = 1.5$ was almost as good and could be used in DSP applications where the square root is more efficient to compute (see (4.19)).

Additionally, it was found to be advantageous to use zero-padding in time domain prior to the Fourier transform to obtain $X(k)$ in (4.21). This is because the resolution of $\lambda_2(\tau)$ is bound to the resolution of the Fourier spectrum $V_c(k)$, as can be seen from (4.54). A very good frequency resolution is needed to analyze low-pitched sounds with the required 3 % accuracy. Zeros were padded to the end of the 93ms analysis frame so as to twice its length prior to the Fourier transform. (Note that the resolution of $\lambda_1(\tau)$ is *not* tied to the Fourier spectrum resolution.)

Figure 25 shows the F0 estimation error rate of the proposed method as a function of the weight factor w_0 when the parameters $v = 0.33$ and $u_0 = 1.5$ were fixed. The panel on the left shows the F0 estimation error rate for isolated sounds (monophonic signals) and the panel on the right shows the predominant-F0 estimation performance for four-sound mixtures.

From the point of view of F0 estimation in general, it is interesting that the graph on the left (monophonic case) shows the importance of both spectral-location and spectral-interval information in F0 estimation. An appropriate balance between the two achieves a good accuracy.

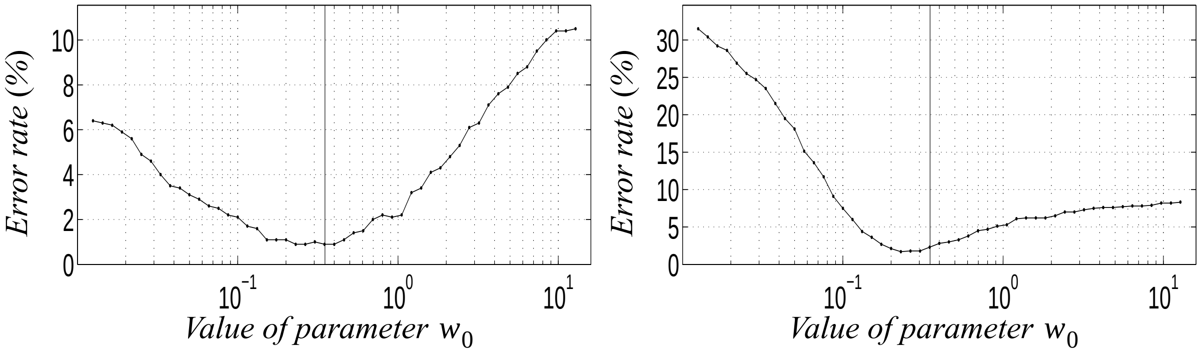

Figure 25. Predominant-F0 estimation performance as a function of the parameter w_0 for isolated sounds (left) and for four-note mixtures (right). The vertical line shows the selected parameter value.

Table 4: Predominant-F0 estimation error rates of the proposed method.

Analysis frame size	Number of concurrent sounds (polyphony)					
	1	2	3	4	5	6
46ms	3.0	2.4	3.4	5.7	9.3	13
93ms	1.1	1.0	1.6	2.3	2.6	4.2
190ms	0.9	0.4	1.0	2.0	2.8	2.8

On the other hand, both types of information alone are moderately successful, too. Value $w_0 = 0$ leads to error rates 10% (monophonic) and 44% (four sounds) and $w_0 = \infty$ leads to error rates 11 % (monophonic) and 8.5 % (four sounds).

In polyphonic mixtures, the spectral-location oriented term $\lambda_1(\tau)$ appears to become relatively more important. However, this is in part illusory and is due to the definition of the predominant-F0 estimation task. In higher polyfonies, there often happens to be at least one rather high-pitched sound. The $\lambda_1(\tau)$ part is successful in detecting these and therefore achieves a low error rate alone. When *all* the F0s of the component sounds have to be estimated, the other term $\lambda_2(\tau)$ becomes important again. This task will be considered in Sec. 4.2.7.

To summarize, the parameter w_0 is very important. Fortunately, however, the performance varies slowly as a function of w_0 and finding exactly the correct value is not critical. An unfortunate aspect is that the best value of w_0 depends somewhat on the frame size and zero-padding factor. In the illustrated cases, a 93ms frame with zero-padding was used and $w_0 \approx 0.35$ performs well. For a 190ms frame with no zero-padding, $w_0 \approx 0.18$ performs well. The effect is important enough so that we fixed different values of w_0 to be used with different analysis frame sizes.

Table 4 shows the predominant-F0 estimation rates of the proposed method in different polyphonies and for three different analysis frame sizes. One thousand random sound mixtures were allotted in each polyphony, and the average predominant-F0 estimation error rate was calculated for these. On the whole, the proposed method is very successful in predominant-F0 estimation, taken the diversity of the acoustic material considered.

4.2.6 Reducing the computational complexity

As mentioned in the previous subsection, the computational complexity of the proposed method as such is rather high. The complexity of computing $\lambda_1(\tau)$ alone is $O(K\log K + K + |T|)$ where the first term is due to the Fourier transform in (4.21), the latter part is due to the algorithm in Table 3, and $|T|$ is the number of period candidates to consider. The complexity of computing $\lambda_2(\tau)$ for all period candidates $\tau \in T$ becomes $O(K^2 + |T|)$ as mentioned in the previous subsection. According to the common use of the order-of-growth notation, the overall complexity becomes $O(K\log K + K + |T| + K^2 + |T|) = O(K^2 + |T|)$ [Cor90].

In this subsection, a solution is described which reduces the overall complexity to $O(K\log K + |T|)$. This is important because in multiple-F0 estimation, longer analysis frames are typically required than in single-F0 estimation. This is due to the relatively higher partial density in sound mixtures.

Computational efficiency is here achieved by proposing a *candidate generation* scheme, which is able to produce a constant small number of period candidates, a set $T_{(sub)}$, so that the candidate corresponding to the maximum of $\lambda(\tau)$ is preserved in the subset $T_{(sub)}$. It follows that $\lambda(\tau)$ in (4.55) needs to be evaluated only for $\tau \in T_{(sub)}$, in order to find the single best candidate among them. In F0-estimation in general, a small number of the most likely F0 candidates can usually be rather easily generated, but the difficult part is to choose the correct estimate among these.

The algorithm in Table 3 turned out to be suitable for the purpose of candidate generation. In other words, we first evaluate only the $\lambda_1(\tau)$ part in (4.55). This is not very demanding computationally. A constant number of 10–15 local maxima are then selected in $\lambda_1(\tau)$ to constitute a set $\hat{T}_{(sub)}$. Then, the candidate period values are refined by further evaluating $\lambda_1(\tau)$ in the vicinity of each $\hat{\tau} \in \hat{T}_{(sub)}$. This is done by calculating $\lambda_1(\tau)$ for $\tau' = K/k'$, where $k' \in K_{1,\hat{\tau}}$ and the set $K_{1,\hat{\tau}}$ is defined for $\hat{\tau}$ according to (4.49). The value $\hat{\tau}$ is then replaced by the value τ' which corresponds to the maximum of $\lambda_1(\tau)$ within the set $K_{1,\hat{\tau}}$. These refined period values are stored to the set $T_{(sub)}$. The $\lambda_2(\tau)$ part is then evaluated *only* for $\tau \in T_{(sub)}$ which comprises 10–15 period candidates. The importance of the refining step is that we can omit the maximization in (4.54) and evaluate $V_c'(k)$ *only* at the positions $k = K/\tau$ for each $\tau \in T_{(sub)}$. Thus $V_c'(k)$ has to be evaluated only for 10–15 different values of *k*. Selecting a number of 10–15 local maxima in $\lambda_1(\tau)$ to constitute the set $\hat{T}_{(sub)}$ (and then $T_{(sub)}$) was found to preserve the overall maximum of $\lambda(\tau)$ in the set. This was observed in all polyphonies for which the method was evaluated (1–6 simultaneous sounds).

The candidate generation step has a theoretically interesting consequence, however. The overall method is no more able to find the F0s of sounds for which *all* the resolved harmonics are missing. In practice this happens if all the harmonics from one to about fifteen are missing. It is known that even in such a case, humans still hear a faint (usually ambiguous) pitch percept [Hou90]. However, this limitation of the method has only theoretical relevance and no importance of whatsoever in practical multiple-F0 estimation tasks. The usual cases like missing the first harmonic component are handled without problems.

Note that the set of values $\tau \in T$ considered when computing $\lambda_1(\tau)$ in Table 3 is in no way restricted either. The sampling does not need to be uniform on any known frequency scale. For simplicity, we have assumed that the sampling of lag values is the same as that of ACF in the given sampling rate, i.e., that $\Delta\tau = 1$ is constant in Table 3. Figure 26 shows the resolutions of three basic F0-sampling schemes as a function of the musical note in the vicinity of which

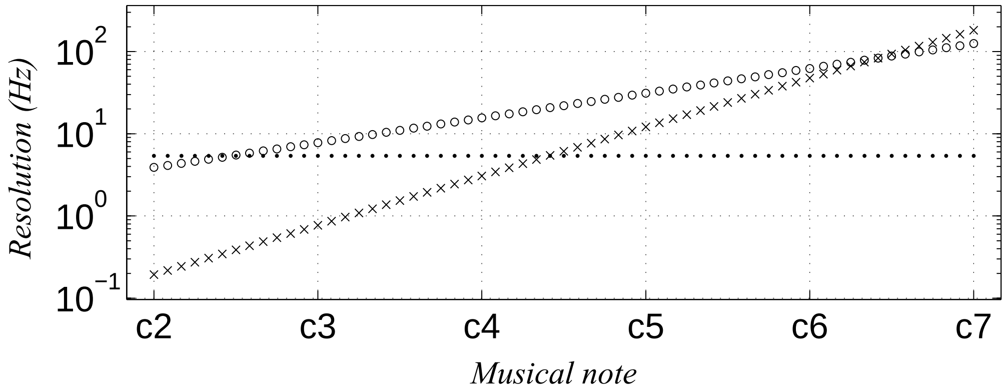

Figure 26. Resolution of three different F0 sampling scales. The circles (o) indicate the inter-note-intervals (Hz) of musical notes on the well-tempered musical scale. The crosses (x) illustrate the F0 difference (Hz) of each note in comparison with a “detuned” note which has one sample longer period. This is the ACF resolution which depends only on the sampling rate f_s (here 22.05 kHz). The dots (•) show the resolution of the Fourier spectrum which is constant f_s/K and is here shown for transform length $K = 4096$.

the resolution is measured. In the figure, the musical notes c2 and c7 correspond to fundamental frequencies 65Hz and 2.1kHz, respectively. As can be seen, neither the Fourier spectrum nor the ACF resolution is equal to the resolution of the logarithmic musical scale which is closest to that of human hearing. However, the resolution of the ACF is in general more natural than that of the Fourier spectrum.

Some accuracy improvement was achieved by using a non-uniform sampling of τ when computing $\lambda_1(\tau)$. However, the difference to the simple ACF-type sampling was negligible. Also, it seems that even on a non-uniform scale, the set T to consider in Table 3 has to be rather large (a couple of hundreds of different values). In the following, the uniform ACF-type sampling with $\Delta\tau = 1$ will be used.

4.2.7 Multiple-F0 estimation by iterative estimation and cancellation

In music transcription, it is typically not sufficient to find one correct F0 in polyphonic mixtures. Instead, the goal is to find the F0s of all the component sounds, or, at least the F0s of the aurally most prominent sounds.

A simple way of extending the described F0 estimation method to multiple-F0 estimation would be to pick *several* local maxima in the function $\lambda(\tau)$ in (4.55), instead of using only the single global maximum. This straightforward approach is only moderately successful, however. For an analysis frame size of 93ms, this technique leads to 40% error rate in four-sound polyphonies, meaning that on the average, 2.4 fundamental frequencies out of four were correctly estimated.

One of the basic problems in multiple-F0 estimation is that even when a predominant-F0 estimator detects a correct F0, the next-highest weight is often assigned to half or twice of this correct F0 value. Thus, the effect of any detected F0 must be cancelled from its harmonics and subharmonics before deciding the next most likely F0.

In the following, we describe an iterative multiple-F0 estimation method which consists of two main steps. First, the described F0-estimation method is used to find one F0 in a mixture sig-

Table 5: Pseudocode of the algorithm which computes a residual spectrum $X_R(k)$ where the effect of the resolved harmonics of all detected sounds is cancelled from the mixture spectrum $X(k)$. The estimated period of a detected sound is denoted by $\hat{\tau}_i$.

```
if This is the first iteration then
    # Initialize the spectrum of all detected sounds (resolved partials only)
    $X_D(k) \leftarrow 0$ for $k = 0, 1, 2, \ldots, K/2$
end if
# Residual spectrum (zero-phase)
$X_R(k) \leftarrow \max(|X(k)| - X_D(k), 0)$ for $k = 0, 1, 2, \ldots, K/2$.
$\hat{F}_i \leftarrow f_s/\hat{\tau}_i$
$p \leftarrow 1$
while $p \leq 20$ and $\lfloor p(K/\tau + \Delta\tau/2) \rfloor < K/2$ do
    # Harmonic selection as in Table 3
    $k^{(0)} \leftarrow \lfloor pK/(\hat{\tau}_i + \Delta\tau/2) \rfloor + 1$
    $k^{(1)} \leftarrow \lfloor pK/(\hat{\tau}_i - \Delta\tau/2) \rfloor$
    if $k^{(1)} < k^{(0)}$ then
        $k^{(1)} \leftarrow k^{(0)}$
    end if
    $k' \leftarrow \arg\max\{X_R(k^{(0)}), \ldots, X_R(k^{(1)})\}$
    Use quadratic interpolation [Ser97] of $X_R(k)$ in the vicinity of $k'$ to estimate the
        frequency $f_p$ and amplitude $a_p$ of a time-invariant sinusoidal partial assumed to
        underlie the local maximum
    Estimate the Fourier spectrum in the vicinity of the assumed sinusoidal component,
        weight it with $w(p, \hat{F}_i)$ (see (4.57)), take absolute values, and add result to $X_D(k)$
    $p \leftarrow p + 1$
end while
```

nal. This is followed by the cancellation of the detected sound, and the estimation is then iteratively repeated for the residual. Depending on the number of F0s to extract, I, the estimation step has to be repeated I times and the cancellation step $I - 1$ times. The estimation part has already been described, therefore only the cancellation procedure remains to be presented.

We use $\hat{\tau}_i$ to denote the period of a sound detected at iteration *i*. The effect of $\hat{\tau}_i$ is cancelled from the mixture signal by performing cancellation separately for the two parts, $\lambda_1(\tau)$ and $\lambda_2(\tau)$, in (4.55). The algorithm which implements the cancellation for the $\lambda_1(\tau)$ part is given in Table 5. The algorithm performs harmonic selection in the same way as in Table 3 but considers only one period value $\tau = \hat{\tau}_i$. Also, instead of cumulating amplitudes of the harmonics to $\lambda_1(\tau)$, the parameters of the sinusoidal partials are estimated and further used to estimate the magnitude spectrum in the vicinity of the partials. The magnitude spectra of the partials of all detected sounds are cumulated to $X_D(k)$ which represents the resolved harmonics of all detected sounds. A residual magnitude spectrum $X_R(k)$ is obtained by subtracting $X_D(k)$ from the initial magnitude spectrum $|X(k)|$ and by constraining resulting negative values to zero.

The mechanism how the cancellation affects the *estimation* part is that, at iterations $i > 1$, the residual spectrum $X_R(k)$ is used instead of the mixture spectrum $X(k)$ when calculating

$\lambda_1(\tau)$ in Table 3. At the first iteration, $X_R(k) \equiv |X(k)|$ and either one can be used.

A certain characteristic of the algorithm in Table 5 is important and deserves special mentioning. Before adding the partials of a detected sound to $X_D(k)$, they are weighted by the modeled resolvability $w(p, F)$, in the same way as in the estimation part in Table 3. This has the consequence that the higher-order partials are not entirely removed from the mixture spectrum when the residual $X_R(k)$ is calculated. This principle is very important in order not to corrupt the sounds that remain in the residual spectrum and have to be detected at the coming iterations. For example, consider a low-pitched sound with F0 70Hz. If all frequency components at the positions of the harmonics of this sound would be removed from the mixture spectrum, the residual would become severely corrupted. Also, it is very unlikely that the parameters of the harmonics could be reliably estimated up to the 20th partial. The described weighting limits the effect of cancellation to the most important harmonics of each sound.

In principle the weights $w(p, F)$ in Tables 3 and 5 are completely independent and would not need to be a “matched pair” in any way. However, the *interpretation* of the weights is the same in both cases and therefore it is logical to use the same values. In Table 5, the interpretation is that we can subtract only resolved harmonics because the frequencies of individual unresolved partials are not known.

For the $\lambda_2(\tau)$ part, the cancellation is somewhat more complicated. However, the backgrounding analysis was already presented in Sec. 4.2.4 and the results can now be applied here. According to (4.54), $\lambda_2(\tau)$ is computed using the convolutions spectra $V_c'(k)$ at different channels *c*. Cancelling the effect of a detected period $\hat{\tau}_i$ can thus be achieved by cancelling the effect of $\hat{\tau}_i$ from $V_c'(k)$ at different channels *c*.

The cancellation procedure presented in the following is based on the assumption that the detected sound consists of partials with constant inter-harmonic frequency intervals $\hat{k}_i = K/\hat{\tau}_i$ at each channel. In this case, the sound causes peaks to $|V_c'(k)|$ at positions $m\hat{k}_i$, where integer $m \geq 0$. This can be easily seen by looking at the definition of $V_c(k)$ in (4.25)[1]. By assuming that the peak at the position $\hat{k}_i$ is due to the detected sound only, the effect of the detected sound at multiples $m\hat{k}_i$, $m = 2, 3, \ldots$ can be *predicted* (approximated) based on the value $|V_c(\hat{k}_i)|$. The values $|V_c'(\hat{k}_i)|$ at different channels have already been computed to obtain $\lambda_2(\hat{\tau}_i)$ in (4.54).

An expression for predicting $|V_c(m\hat{k}_i)|^2$ based on $|V_c(\hat{k}_i)|^2$ is given by (4.69). However, in order to avoid evaluating the integrals in (4.69), the prediction formula is here formulated in terms of magnitude spectra. This becomes

$$|V_c(m\hat{k}_i)| \approx |V_c(\hat{k}_i)| \frac{\sum_l H_c(l) H_c(m\hat{k}_i - l)}{\sum_l H_c(l) H_c(\hat{k}_i - l)} = |V_c(\hat{k}_i)| \frac{\sum_l J_{c,\, m\hat{k}_i}(l)}{\sum_l J_{c,\, \hat{k}_i}(l)}. \tag{4.72}$$

Derivation of the above formula is very similar to that of (4.69) and is therefore not presented. In addition to the assumptions of piecewise constant inter-partial intervals and spectral smoothness, linear phases have to be assumed. Whereas linear phases cannot be assumed for all sound sources, in practice the resulting error is sufficiently small. At first, the two approximations in (4.69) and (4.72) appear as contradictory. However, the difference is due to the fact that (4.72)

[1] As defined in (4.45), the difference between $V_c(k)$ and $V_c'(k)$ is that the latter includes only the amplitude envelope spectrum on zero frequency, and not the distortion spectrum on $2f_c$.

has been derived using (4.23) as a starting point instead of (4.20). The difference stems from the asymmetry of (4.20) and (4.23) as mentioned in the footnote on page 38. Both approximations, (4.69) and (4.72), are usable.

Using (4.72), the effect of a detected sound (with period $\hat{\tau}_i$) on the convolution spectra $|V_c'(k)|$ at different channels c can be estimated. The values $V_c'(\hat{k}_i)$ at different channels are known and, as discussed above, the effect is assumed to be limited to the vicinity of positions $m\hat{k}_i$ in $|V_c'(k)|$, where integer $m \geq 0$. The effect of the detected sound can then be taken into account when the next most likely period is decided. This can be efficiently implemented as follows:

1. After having detected period $\hat{\tau}_i$ at iteration i, the following data structures are produced:

- Compute vector $\lambda_{2,c}^{(i)}$ for the detected period $\hat{\tau}_i$ and for all channels c as

$$\lambda_{2,c}^{(i)} = \left| \eta_0 H_{LP}(\langle \hat{k}_i \rangle) \frac{\gamma_c^2}{\sigma_c \sqrt{8\pi}} V_c'(\langle \hat{k}_i \rangle) \right| , \tag{4.73}$$

  where $\hat{k}_i = K/\hat{\tau}_i$ and $\langle \bullet \rangle$ denotes rounding towards nearest integer. Note that this is analogous to recalculating the part $\lambda_2(\tau)$ for the detected period $\tau = \hat{\tau}_i$ (the maximization in (4.54) can be omitted when using the candidate generation procedure, as described in the previous subsection). However, instead of summing the bandwise terms as in (4.54), they are stored to the vector $\lambda_{2,c}^{(i)}$ which represents the level of the unresolved harmonics of the detected sound at different channels c. The algorithm in [P4] can be used without modifications to compute (4.73).

- Parallel to (4.73), compute vector $\tilde{J}_c^{(i)}$ for the detected period $\hat{\tau}_i$ and for all channels c as

$$\tilde{J}_{c,\hat{k}_i} = \sum_{l = -K/2 + \hat{k}_i}^{K/2 - \hat{k}_i} H_c(l) H_c(\hat{k}_i - l) = \sum_{l = -K/2 + \hat{k}_i}^{K/2 - \hat{k}_i} J_{c,\hat{k}_i}(l) . \tag{4.74}$$

  By comparison with the decomposition of $V_c(k)$ in (4.42), it can be seen that (4.74) is equivalent to computing $V_c(\hat{k}_i)$ so that a unity value is substituted in place of $X(k)$. In practice, the values $\tilde{J}_{c,\hat{k}_i}$ are efficiently obtained as a side-product when computing $\lambda_{2,c}^{(i)}$.

- Initialize $L_i(k) \leftarrow 0$ for $k = 0, 1, 2, \ldots, K/2$. Then translate the spectrum of the time-domain window function to frequencies $m\hat{k}_i$, where $m = 1, 2, \ldots$. Take absolute values, scale the spectra so that their maxima correspond to unity, and add the spectra to $L_i(k)$.

2. The data structures computed at step 1 are used to cancel the effect of the detected sound at the coming iterations. At the first iteration, $\lambda_2(\tau)$ is calculated simply according to (4.54). At iterations $i = 2, 3, \ldots$, the following formula replaces (4.54):

$$\lambda_2(\tau) = \max_{k \in K_{1,\tau}} \left\{ \eta_0 H_{LP}(k) \sum_c \left| \frac{\gamma_c^2}{\sigma_c \sqrt{8\pi}} V_c'(k) \right| - \sum_{j=1}^{i-1} \sum_c \left(L_j(k) \lambda_{2,c}^{(j)} \frac{\tilde{J}_{c,\hat{k}_i}}{\tilde{J}_{c,\hat{k}_j}} \right) \right\} . \tag{4.75}$$

   In the above formula, the first part equals (4.54). The second part subtracts the amount which is predicted to be due to the already-detected sounds. As an example, let us consider the second iteration, $i = 2$. If k in the above formula is an integer multiple of $K/\hat{\tau}_1$, where $\hat{\tau}_1$ is the period detected at the first iteration, then the value of $L_1(k)$ is unity. In this case, the quantity $\lambda_{2,c}^{(1)}(\tilde{J}_{c,\hat{k}_2}/\tilde{J}_{c,\hat{k}_1})$ represents the amount that is predicted to be due to the detected sound. If $\tau = \hat{\tau}_1$, the value of $\tilde{J}_{c,\hat{k}_2}/\tilde{J}_{c,\hat{k}_1}$ is unity and the latter part of (4.75) subtracts the same amount that is cumulated in the first part and $\lambda_2(\tau)$ becomes zero.

Table 6: Multiple-F0 estimation error rates (%) of the proposed iterative method. The performance of the F0 estimator proposed in [P5] is shown as a reference.

Method	Analysis frame size	Number of concurrent sounds (polyphony)					
		1	2	3	4	5	6
Proposed method	46ms	3.1	8.7	15	22	29	36
	93ms	**1.3**	**5.0**	**8.2**	**11**	**16**	**19**
	190ms	1.1	3.6	6.2	9.3	13	16
Reference method proposed in Publication [P5]	46ms	17	26	36	46	52	57
	93ms	**4.2**	**8.7**	**16**	**22**	**29**	**34**
	190ms	1.8	3.9	6.3	9.9	14	18

The presented cancellation procedures are not computationally demanding. Also, it should be noted that the part $\lambda_2(\tau)$ is calculated only for a small subset of candidate values $\tau \in T_{(sub)}$. In practical implementations, the data structures $L_i(k)$ can be computed on-the-fly in (4.75) to reduce memory usage.

4.2.8 Multiple-F0 estimation results

Simulations were run to validate the proposed auditory-model based multiple-F0 estimation method. The acoustic database was the same as that used in [P5]. It consists of samples from four different sources: the McGill University Master Samples collection [Opo87], University of Iowa website [Iow04], IRCAM Studio Online [IRC04], and independent recordings for acoustic guitar. There were altogether 30 different musical instruments, comprising brass and reed instruments, strings, flutes, the piano, and the guitar. The total number of samples was 2536 and these were randomly mixed to generate test cases. The instruments marimba and the vibraphone were excluded from the data set since these do not represent harmonic sounds (see Sec. 3.1) and the proposed method admittedly does not work very reliably for these instruments. Sampling rate was 44.1kHz.

Semirandom sound mixtures were generated by first allotting an instrument and then a random note from its whole playing range, restricting, however, the pitch over five octaves between 65Hz and 2100Hz. The desired number of simultaneous sounds was allotted and then mixed with equal mean-square levels. The acoustic signal was fed to the proposed multiple-F0 method which estimated the F0s in a single time frame, 100ms after the onset of the sounds. The number of concurrent sounds, i.e., the number of F0s to extract, was given along with the acoustic mixture signal. The parameters of the system were fixed, except the parameter w_0 which was varied according to the size of the analysis frame, as described in Sec. 4.2.5.

A correct F0 estimate was defined to deviate less than half a semitone (±3 %) from the true value, making it “round” to a correct note on a Western musical scale. Errors smaller than this are not significant from the point of view of music transcription. The error rate was computed as the number of erroneous F0 estimates divided by the number of F0s presented. One thousand mixture signals were generated per each polyphony and the error rates were averaged over these.

Table 6 shows the multiple-F0 estimation results of the proposed method. The set of values $\tau \in T$ applied when computing $\lambda_1(\tau)$ in Table 3 was uniform, i.e., $\Delta\tau = 1$ was constant (see

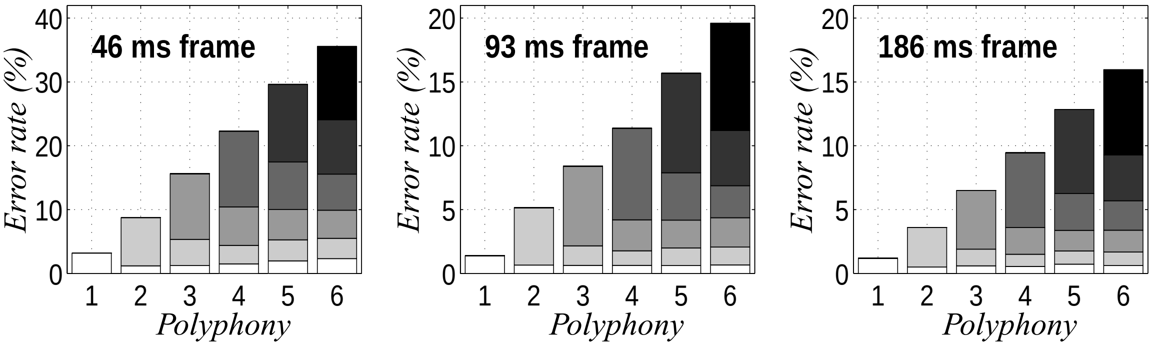

Figure 27. Multiple-F0 estimation error rates using the proposed algorithm. Bars represent the overall error rates, and the different shades of gray the error cumulation in iteration.

Table 3). In computing $\lambda_2(\tau)$, a candidate generation procedure with $|T_{(sub)}| = 15$ was applied in all cases (see Sec. 4.2.6). Values beyond 15 did not bring additional performance improvement. In computing the Fourier spectrum in (4.21), the analysis window was always zero-padded to the length 8192 samples (190ms).

As a general impression, the proposed auditory-model based method is very accurate and is able to handle the diversity of the acoustical material involved. For example, the error rate for four-voice polyphonies and 93ms analysis window was 11 %, meaning that 3.56 sounds out of four were correctly estimated on the average. In increasing polyphony, the error rate grows gradually but the method does not break up at any point.

A particularly attractive feature of the proposed method is that it works accurately in relatively short analysis frames. For comparison, the last three rows of Table 6 show the results for the algorithm proposed by us in [P5][1]. It is fair to say that the latter algorithm breaks down when the analysis frame becomes shorter than 93ms. Even for the 93ms frame size, the method proposed here is significantly better. The superiority of the auditory-model based approach in short analysis frames is due to the fact that the method does not attempt to resolve individual higher-order partials, but these are represented collectively by the amplitude envelope spectrum $V_c'(k)$ in (4.54).

Another attractive feature of the method proposed here is that it comprises only few free parameters. The three parameters that had to be tuned were w_0 which determines the balance of the two parts in (4.55), the degree of compression, v, and the scaling factor for the subband widths u_0. Among these, only w_0 is completely "free" in the sense that it cannot be deduced from known psychoacoustic quantities.

Figure 27 illustrates error cumulation in the iterative estimation and cancellation process. The bars represent the overall error rates as a function of the number of concurrent sounds. The different shades of gray in each bar indicate the error cumulation in the iteration, errors which occurred in the first iteration at the bottom, and errors of the last iteration at the top. As can be seen, the error rate approximately doubles when the analysis frame is shortened from 93ms to 46ms. The difference between 186ms and 93ms frames is not very big.

Figure 28 shows the error rate of the proposed method as a function of the F0s of the target sounds which were presented to the system. The number of concurrent sounds (polyphony)

[1] The accuracy of the reference method has been compared with that of trained musicians in [P5].

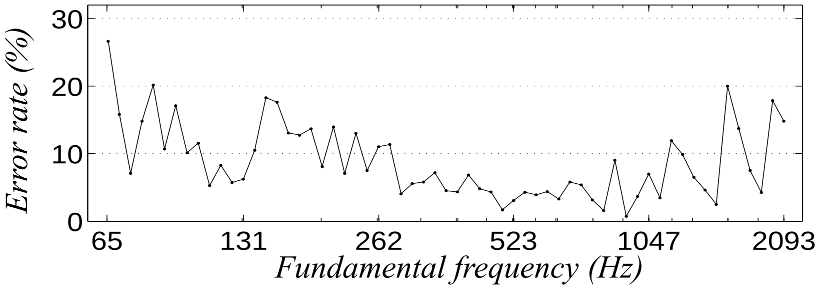

Figure 28. Multiple-F0 estimation error rate of the proposed method as a function of the F0s of the target sounds. The number of concurrent sounds was four in this experiment.

was four in this experiment. The error rate for F0 value 523Hz, for example, was computed by counting the number of sounds with F0 523Hz that were not detected by the method, and dividing this by the number of sounds with F0 523Hz that were presented to the system. As can be seen, very low-pitched or high-pitched sounds were more often incorrectly estimated. Musical sounds at the both ends are typically more difficult to handle, due to their irregularity.

The presented method remains incomplete in the sense that mechanisms for suppressing additive noise or for estimating the number of concurrent sounds were not presented. These will be considered in Chapter 6. Also, the method proposed in this chapter comprises many directions of potential further improvement that have not yet been explored. For example, the expectation values of $\lambda(\tau)$ for different τ were not balanced in any way. It is possible that the method has a preference towards high or low F0s, although this has not been verified. Also, many details of the method are not necessarily the optimal ones but, rather, examples of well-working solutions. It is a matter of future research to optimize these details.

5 Previous Approaches to Multiple-F0 Estimation

In Chapter 3, different approaches to single-F0 estimation were reviewed. The aim of *multiple*-F0 estimation is to find the F0s of all the component sounds in a mixture signal. An instance of such an algorithm was proposed in Chapter 4. The complexity of the multiple-F0 estimation problem is significantly higher than that of single-F0 estimation. Some intuition of this can be developed by comparing the spectrum of a harmonic sound with that of a mixture of four harmonic sounds in Fig. 29.

Multiple-F0 estimation is closely related to sound separation. An algorithm that is able to estimate the F0 of a sound in the presence of other sounds is, in effect, also organizing the respective spectral components to their sound sources [Bre90, p.240]. Regardless of whether this organization takes place prior to the F0 estimation or vice versa, the two are closely related. Also, multiple-F0 estimation raises a number of problems that need not be addressed in single-F0 estimation. To name a few examples, concurrent sounds in certain F0 relationships may cause a non-existing sound to be detected, such as the root of a chord in its absence. Also, the partials of concurrent sounds often coincide in frequency, in which case the parameters of the partials can no more be directly estimated from the spectrum. In practical music transcription tasks, the number of concurrent sounds has to be estimated.

The diversity of approaches taken towards multiple-F0 estimation is even wider than that in single-F0 estimation. The aim of this chapter is to review the previous work in this area.

5.1 Historical background and related work

Music signals can be viewed as the “home ground” for multiple-F0 estimation, in the same way as speech signals are the principal target signals for single-F0 estimation. The first multiple-F0 algorithms were designed for the purpose of transcribing polyphonic music. These attempts date back to 1970s, when Moorer built a system for transcribing duets, i.e., two-voice compositions [Moo75,77]. The work was continued by Chafe and his collegues [Cha82,86a,b]. At the same time, the problem was independently studied by Piszczalski [Pis86]. Further advances were made by Maher [Mah89,90,94] and de Cheveigné [deC93]. However, the early systems suffered from severe limitations in regard to the pitch range and relationships of simultaneous sounds, and the polyphony was restricted to two concurrent sounds. Attempts towards higher polyphony were made by limiting to one carefully modeled instrument [Haw93, Ros98b], or by allowing more errors to occur in the output [Kat89, Nun94].

More recent music transcription systems have recruited psychoacoustically motivated analysis

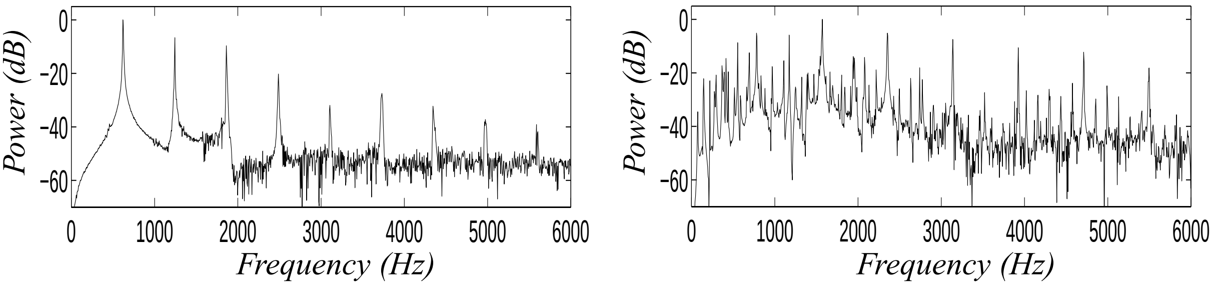

Figure 29. An example of the spectrum of a harmonic sound (left) and that of a mixture of four harmonic sounds (right).

principles [Kas95, Ste99, God99], models of the human auditory periphery [Mar96b, deC99, God99, Tol00], Bayesian inference techniques [Kas95, Got01, Dav03], processing architectures from the artificial intelligence domain [Mar96a,b, God99, Got01], and sparse coding methods [Vir03, Abd_]. Each of these areas will be examined in more detail in Sec. 5.2.

As mentioned above, multiple-F0 estimation and sound source separation are closely linked. The human auditory system is very effective in separating and recognizing individual sound sources in mixture signals. This cognitive function is called called *auditory scene analysis* (ASA). Computational modeling of ASA has been a subject of increasing interest since 1990 when Bregman put together his influential work describing the principles and mechanisms of the psychology of ASA in humans [Bre90]. Doctoral theses on the various aspects of computational ASA (CASA) were prepared by Mellinger [Mel91], Cooke [Coo91,94], Brown [Bro92a,94], and Ellis [Ell96]. More recent overviews of this field can be found in [Ros98a] and [Coo01a]. From the point of view of multiple-F0 estimation, the research in CASA has not produced as many practical methods as e.g. the models of the more peripheral parts of hearing, such as the unitary pitch model described in Chapter 3 [Med91a,b]. This is partly due to the fact that CASA in general is concerned with all types of sound sources and in practice often related to noise-like or speech sounds.

Separation of speech from interfering speech or other sounds is an important special area of sound separation. Early work on this problem has been done by Parsons [Par76] and Weintraub [Wei86]. They concentrated on utilizing the pitch information to carry out the task. More recently, Wang and his colleagues have focused on using the pitch information for speech segregation [Wan99, Hu02]. A multipitch tracking algorithm for noisy speech was presented in [Wu02]. However, consonants are the main carriers of information in speech signals and speech is voiced only part of the time. Multiple-F0 estimation alone would be more appropriate for separating singing from interfering singing, although e.g. Parsons reports that “normal speech separated by the process in its present form not only is intelligible, but also gives the illusion of preserving most of the recovered talker’s consonants”. Despite this valid observation, other techniques prevail in noise-robust speech recognition. The “missing data” approach is another separation-oriented approach to noise-robust speech recognition and is not limited to the voiced parts only [Bar01, Coo01b]. This line of research attempts to identify the spectro-temporal regions that represent the target speech and is more closely connected to CASA in general. Okuno *et al.* and Nakatani *et al.* applied two microphones to utilize directional information along with harmonicity in separating two simultaneous speakers [Oku99, Nak99].

5.2 Approaches to multiple-F0 estimation

It is difficult to categorize multiple-F0 estimation methods according to any single taxonomy. This is because the methods are complex and typically combine several processing principles. As a consequence, there is no single dimension which could function as an appropriate basis for categorization. However, some main viewpoints to the problem can be discerned and it is the aim of this subsection to introduce these.

Table 7 lists the characteristic attributes of ten different multiple-F0 estimation methods. The given list of methods is not intended to be complete. Instead, an attempt was made to select two representative and good examples from each of the main viewpoints to the problem. For a comprehensive historical overview of F0 estimation methods in music, see [Hai01]. In Table 7, *mid-level representations* refer to the data representations that are used between the acoustic

Table 7: Characteristics of some multiple-F0 estimation methods

Reference	**Main viewpoint**	**Mid-level representations**	**Knowledge applied**	**Computation**	**Evaluation material**
Kashino *et al.*, 1995	knowledge integration	sinusoidal tracks	• *partial grouping*: harmonicity, onset timing • *musical*: chord-note relations, statistics of chord transitions • *sound sources*: tone memory, timbre model space, automatic tone modeling	Bayesian probability network; conceptually based on blackboard model	• statistical evaluation • random mixtures of acoustic samples from five instruments • max polyphony 3
Sterian, 1999	perceptual grouping of partials	sinusoidal tracks	• *partial grouping*: harmonicity, onset and offset timing, low partial support, partial gap, partial density	evaluation of grouping likelihoods, multiple-hypothesis tracking	• short example cases, one per each polyphony 1–4 • sample-based synthesis with a fixed instrumentation
de Cheveigné *et al.*, 1999	auditory modeling	unitary pitch model: SACF, subchannel signals	• harmonicity, • implicit knowledge of human pitch perception	iterative estimation and cancellation	• statistical evaluation • synthetic perfectly periodic sounds • pitch range less than an octave • max polyphony 3
Tolonen *et al.*, 2000	auditory modeling	simplified unitary pitch model: SACF	• harmonicity, • implicit knowledge of human pitch perception	efficient version of the unitary pitch model, no iteration but "enhancing" SACF	• example cases of musical chords and mixed speech signals, noisy and clean • statistical evaluation shown in [P5]
Martin, 1996	knowledge integration	Log-lag correlogram, SACF	• *partial grouping*: harmonicity, • *musical*: rules governing tonal music, • implicit knowledge of human pitch perception, • "garbage-collection" heuristics	Blackboard architecture	• example transcription cases • piano performances of four-voice Bach chorales in 18th century counterpoint style
Godsmark *et al.*, 1999	knowledge integration	auditory model: synchrony strands with features	• *grouping cues for strands*: harmonicity, onset and offset timing, time-frequency proximity, common movement, • *musical*: detect and utilize recurrent melodic phrases, metrical predictions, • *stream formation*: pitch and timbre proximity	Blackboard architecture	• short example cases • acoustic data from sample-based MIDI synthesizer • max polyphony 6
Goto, 2001	signal model	estimated spectral power distribution in one analysis frame	• *signal model*: multiple notes, note partials have a Gaussian spectrum centered at harmonic positions, • *tone models* which are estimated, • *prior distributions* for parameter values, • *musical*: frequency ranges for melody and bass, temporal continuity of bass/melody	Maximum *a posteriori* (MAP) estimation using expectation maximization (EM) algorithm	• detection of the melody and bass lines (predominant-F0) on real-world CD recordings
Davy *et al.*, 2003	signal model	time-domain signal	• *signal model*: multiple notes, notes may have inharmonic partials (represented as time-localized sinusoids), non-white residual noise, • *prior distributions* for parameter values	Bayesian model, variable dimension Markov chain Monte Carlo (MCMC) sampling of posterior likelihoods	• example cases • results shown for polyphonies 1–2
Virtanen, 2003	sparse coding	magnitude spectrogram, sources have constant spectrum & time-varying gain	• simple source mixing model, • cost function which minimizes the *reconstruction error* while preserving the *sparseness* of sources and *temporal continuity* of their gains	algorithm which combines projected gradient descent and a multiplicative step	• demo signals for real-world CDs • drum transcription
Abdallah *et al.*, unpublished	sparse coding	magnitude spectrogram, sources have constant spectrum & time-varying gain	• simple source mixing model, • *reconstruction error* minimization, favouring *sparseness* of sources	modified quasi-Newton optimizer, gradient-ascent inference of sources, maximum-likelihood learning of gains	• MIDI synthesis of keyboard music • max polyphony 3

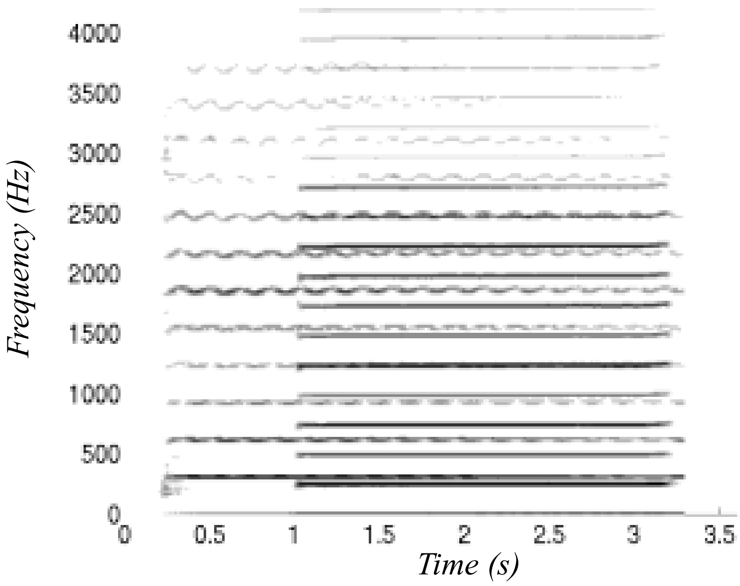

Figure 30. Time-frequency representation of a mixture of two harmonic sounds: A cello sound (F0 310Hz) setting on at 250ms and a saxophone sound (F0 250Hz) setting on at 1.0s.

input and the final analysis result. Often a front-end of some kind is used to transform the input data to a more accessible form before the more complex reasoning takes place. The column titled *knowledge applied* lists the types of knowledge that are utilized in performing the analysis. In music signals, very diverse sources of knowledge are available, relating to physical sound production, to music theory, and to the human auditory perception, for example. The column *computation* summarizes how the actual computations are carried out, given the data representations and the knowledge to use. The *evaluation* column gives an idea of the target material of each system.

In the following, the methods are described in more detail. Subheadings are provided to improve readability but it should be remembered that the cited papers really cannot be put under a single label.

5.2.1 Perceptual grouping of frequency partials

CASA is usually viewed as a two-stage process where an incoming signal is first decomposed into its elementary time-frequency components and these are then organized to their respective sound sources. Provided that this is successful, a conventional F0 estimation method could be used to measure the F0 of each of the separated component sounds, or, in practice, the F0 estimation often takes place as a part of the organization process already.

The organization part is the most complicated one among the above-mentioned processing stages. An important step forward in this area was to discover a set of perceptual cues which promote the grouping of time-frequency components to a same sound source in human listeners. The “cues” are measurable acoustic features of the elementary time-frequency components. In [Bre90], Bregman points out the following cues: proximity in time-frequency, harmonic frequency relationships, synchronous changes in the parameters of the components, and spatial proximity (i.e., the same direction of arrival).

Figure 30 shows the spectrogram of a mixture of two harmonic sounds, a cello sound and a saxophone sound. Many of the above-mentioned cues are visible in the figure. The partials of the cello sound start 750ms before the saxophone sound and exhibit synchronous frequency modulation. Also, the partials within each sound are in harmonic relationships although, due to the inharmonicity phenomenon described in Sec. 3.1, perfect harmonicity cannot be assumed. These features are effectively used by the auditory system in order to "hear out" each of the two sounds. An example of coinciding partials can be seen at the frequency band around 1200Hz.

Temporally continuous sinusoidal components, *sinusoidal tracks*, have often been used as the elementary components for which the mentioned features are measured [Kas93,95, Ste99, Vir00]. Reliable extraction of these components in real-world music signals is not as easy as it may seem. Pioneering work in this area has been done by McAulay and Quatieri [McA86] and Serra [Ser89,97], and the work has been continued by several authors [Dep93,97 Rod97, Goo97, Ver97,00, Lev98, Vir01]. More recently, also auditorily-motivated representations have been used as the mid-level representation [God99].

Kashino *et al*. brought Bregman's ideas to *music scene analysis* and also proposed several other new ideas for music transcription [Kas93, 95]. The front-end of their system used a "pinching plane method" to extract sinusoidal tracks from the input signal. These were clustered into note hypotheses by applying a subset of the above-mentioned perceptual cues. Harmonicity rules and onset timing rules were implemented. Other types of knowledge were integrated to the system, too. *Timbre models* were used to identify the source of each note and pre-stored *tone memories* were used to resolve coinciding frequency components. Chordal analysis was performed based on the probabilities of the notes to occur under a given chord. Chord transition probabilities were encoded into trigram models (Markov chains). For computations, a Bayesian probability network was used to integrate the knowledge and to do simultaneous bottom-up analysis, temporal tying, and top-down processing (chords predict notes and notes predict components). Evaluation material comprised five different instruments and polyphonies of up to three simultaneous sounds. The work still stands among the most elegant and complete transcription systems. Later, Kashino *et al.* have addressed the problem of source identification and source stream formation when the F0 information is given *a priori* [Kas99].

The PhD work of Sterian was more tightly focused on implementing the perceptual grouping principles for the purpose of music transcription [Ste99]. Sinusoidal partials were used as the mid-level representation. These were extracted by picking peaks in successive time frames using modal distribution and then by applying Kalman filtering to estimate temporally continuous sinusoidal tracks. Sterian represented the perceptual grouping rules as a set of likelihood functions, each of which evaluated the likelihood of the observed partials given a hypothesized grouping. Distinct likelihood functions were defined to take into account onset and offset timing, harmonicity, low partial support, partial gap, and partial density (see [Ste99] for the definitions of the latter concepts). The product of all the likelihood functions was used as a criterion for optimal grouping. While an exhaustive search over all possible groupings is not possible, a multiple-hypothesis tracking strategy was used to find a suboptimal solution. For each new partial, new competing hypotheses were formed and the most promising hypotheses were tracked over time. Evaluation results were given for a small test set with 1–4 concurrent sounds.

Godsmark and Brown used an auditory model as a front-end to extract "synchrony strands" for which the grouping cues were extracted [God99]. The latter system is introduced in more detail in Sec. 5.2.3. Nakatani and Okuno have used the spatial proximity cue along with the other cues to separate the voiced sections of several simultaneous speakers [Nak99, Oku99]. A deterministic way of encoding the partial grouping principles has been proposed by Virtanen and Klapuri in [Vir00].

5.2.2 Auditory-model based approach

The unitary pitch model of Meddis *et al.* (see Chapter 3) has had a strong influence on F0 estimation research in general [Med91a,b, 97]. While Bregman's theory is primarily concerned with the psychology of auditory perception, the unitary model addresses the more peripheral (largely physiological) parts of hearing. Although multipitch estimation in sound mixtures was not addressed, research to this direction was inspired, too. The method proposed in Chapter 4 belongs to this category.

de Cheveigné and Kawahara extended the unitary model to the multiple-F0 case. They proposed a system where pitch estimation was followed by the cancellation of the detected sound, and the estimation was then repeated for the residual signal [deC99]. The iterative approach to multiple-F0 estimation was originally proposed by de Cheveigné in [deC93], where also a comprehensive review of the previous harmonic selection and harmonic cancellation models was given. In [deC99], the cancellation was performed either by channel selection as in the concurrent vowel identification model of Meddis *et al.* [Med92], or, by performing within-channel cancellation filtering. In addition, a computationally exhaustive joint estimator was proposed where the F0s of two concurrent sounds were simultaneously estimated. Although the evaluation results were presented for a rather artificial and perfectly periodic data set, the proposed iterative approach was indeed a successful one.

Tolonen and Karjalainen developed a computationally efficient version of the unitary pitch model and applied it to the multiple-F0 estimation of musical sounds [Tol00]. In pitch computations, only two frequency bands were used instead of the 40–120 bands in the original model, yet the main characteristics of the model were preserved. Practical robustness was addressed by flattening the spectrum of an incoming sound by inverse warped-linear-prediction filtering and by using the generalized ACF method (see Sec. 3.3.1) for periodicity estimation. Extension to multiple-F0 estimation was achieved by cancelling subharmonics in the summary autocorrelation function which is produced by the model. From the resulting *enhanced summary autocorrelation function*, all F0s were picked without iterative estimation and cancellation. The method is relatively accurate and it has been described to sufficient detail to be exactly implementable based on [Tol00]. Statistical evaluation of the method can be found in [P5]. Also, Karjalainen and Tolonen have proposed iterative approaches to multiple-F0 estimation and sound separation using the described simplified auditory model [Kar99b].

5.2.3 Emphasis on knowledge integration: Blackboard architectures

As already mentioned, content analysis of audio signals is a many-faceted process and involves the use of both acoustic data and prestored internal models [Bre90, Ell96]. Meaningful integration of the various processing principles has turned out to be very difficult. A list of requirements for a flexible and extendable system architecture comprises at least the following:

- Analysis algorithms and data types of very different kinds can be integrated to the system.
- After being encapsulated to the architecture, the individual algorithms collaborate and compete without explicit reference to, or knowledge of, each other.
- The architecture should make it relatively easy to add and remove processing modules.

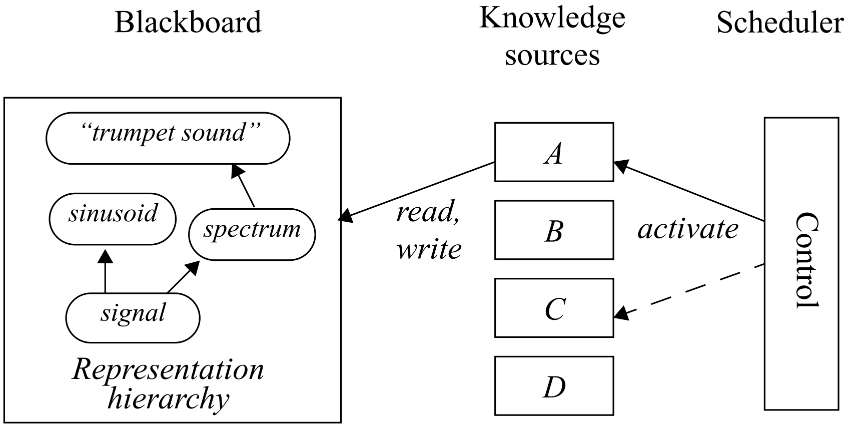

Figure 31. Overview of the blackboard architecture (reprinted from [Kla01b]).

- System should be able to handle uncertain data, and let several alternative explanations evolve side-by-side.

Blackboard systems, originally developed in the field of artificial intelligence, meet several of the above-mentioned architectural needs [Nii86, Rus95]. Figure 31 illustrates the three main components of a blackboard architecture. The name blackboard refers to the metaphor of a group of experts working around a physical blackboard to solve a problem. The *blackboard* is hierarchy of data representations at different abstraction levels. Representations for audio content analysis have been proposed e.g. in [Kla95, Ell95,96]. The state of the analysis is completely encoded in the hypotheses on the blackboard. This data is common to a number of autonomous *knowledge sources*, i.e., processing algorithms which manipulate the data when requested. The *control* component decides when each knowledge source is activated. The design of this part largely determines the successfulness of a blackboard system in integrating the functional entities (knowledge sources) [Car92].

In [Mar96a,b], Martin proposed a system for transcribing piano performances of four-voice Bach chorales. In his system, an auditory model (log-lag correlogram of Ellis [Ell96]) was used as a front-end to a blackboard model which employed knowledge about physical sound production, rules governing tonal music, and "garbage collection" heuristics. Support for F0s was raised on a frame-by-frame basis and then combined with the longer-term power-envelope information to create note hypotheses. Musical rules favoured F0s in certain intervallic relations. The knowledge sources consisted of precondition–action pairs (cf. if–then structure). At each time step, the control component evaluated the preconditions of the knowledge sources in a priority order and executed the first whose precondition was satisfied.

A more recent model of Godsmark and Brown was particularly designed to facilitate the integration of the auditory organization principles described in Sec. 5.2.1, and competition between these [God99]. The system is quite complex and could equally well be introduced under the Section 5.2.1 or 5.2.2. The applied auditory front-end produced "synchrony strands" which represented dominant time-frequency components at different bands [Coo91]. These were grouped to sound events by extracting features from each strand and by applying Bregman's organization principles. Sound events were further grouped to their respective sources (event "streams") by computing pitch and timbre proximities between successive sounds. Musical meter information was used to predict when events will occur and melodic pattern induction to predict events in recurrent patterns. The model was evaluated by showing that it could segregate melodic lines from polyphonic music and to resolve interleaved melodies. Transcription accuracy as such was not the main goal.

It is important to note that the blackboard architecture as such is merely a conceptual model, primarily concerned with the implementation rather than the actual algorithms. It does not provide a *computational model* for knowledge integration. For this, probababilistic models, especially dynamic Bayesian networks [Mur_], are powerful modeling tools. Statistical methods are a solid “common ground” for integrating diverse types of knowledge (acoustic data, internal models, musicological models etc.) and have served excellently for example in speech recognition. Kashino *et al.* used the Blackboard architecture as the backgrounding conceptual model, but applied Bayesian networks to carry out the quantitative integration task. We have proposed a blackboard architecture and a few related inference techniques in [Kla01b].

5.2.4 Signal-model based probabilistic inference

It is possible to state the whole multiple-F0 estimation problem in terms of a signal model, the parameters of which should be estimated. Consider e.g. the model [Dav03]:

$$y(t) = \left\{ \sum_{n=1}^{N} \sum_{m=1}^{M_n} a_{n,m} \cos[m\omega_n t] + b_{n,m} \sin[m\omega_n t] \right\} + e(t) \tag{5.1}$$

where N is the number of simultaneous sounds, M_n is the number of partials in sound n, ω_n is the fundamental frequency of sound n, and $a_{n,m}$, $b_{n,m}$ together encode the amplitude and phase of individual partials. The term e_t is a residual noise component.

In principle, *all* the parameters on the right-hand side of the above equation should be estimated based on the observation $y(t)$ and possible prior knowledge about the parameter distributions. As pointed out by Davy *et al.* in [Dav03], the problem is Bayesian in the sense that there is a lot of prior knowledge concerning music signals.

Davy and Godsill elaborated the above signal model to accommodate time-varying amplitudes, non-ideal harmonicity, and non-white residual noise [Dav03]. A likelihood function for observing $y(t)$ given model parameters was defined. Prior distributions for the parameters were carefully selected. An input signal was first segmented into excerpts where no note transitions occur. Then the parameters of the signal model were estimated in the *time domain*, separately for each segment. The main challenge of this approach is in the actual computations. For any sufficiently realistic signal model, the parameter space is huge and the posterior distribution is highly multimodal and strongly peaked. Davy and Godsill used variable-dimension Markov chain Monte Carlo sampling of the posterior, reporting that much of the innovative work was spent on finding heuristics for the fast exploration of the parameter space [Dav03]. Although computatinally inefficient, the system was reported to work robustly for polyphonies up to three simultaneous sounds.

Goto proposed a method which models the *short-time spectrum* of a music signal as a weighted mixture of tone models [Got01]. Each tone model consists of a fixed number of harmonic components which are modeled as Gaussian distributions centered at integer multiples of the F0 in the spectrum. Goto derived a computationally feasible expectation-maximization (EM) algorithm which iteratively updates the tone models and their weights, leading to a maximum *a posteriori* estimate. Temporal continuity was considered by tracking framewise F0 weights in a multiple-agent architecture. Goto used the algorithm successfully to track the melody and the bass lines on CD recordings in real-time. The algorithm utilized prior knowledge of the typical frequency ranges for the melody and bass lines and favoured temporal continuity of the two trajectories.

Although the overall system of Goto is relatively complex, the core EM algorithm can be easily implement based on [Got01]. The algorithm estimates the weights of all F0s, but typically only one (predominant) F0 was found in our simulations, exactly as claimed by Goto. Goto's signal model resembles that of Doval and Rodet for monophonic F0 estimation [Dov91].

5.2.5 Data-adaptive techniques

In data-adaptive systems, there is no parametric model or other knowledge of the sources. Instead, the source signals are estimated from the data. Typically, it is not even assumed that the sources (which here refer to indivitual notes) have harmonic spectra! For real-world signals, the performance of e.g. independent component analysis alone is poor. However, by placing certain restrictions for the sources, the data-adaptive techniques become applicable in realistic cases. Such restrictions are e.g. independence of the sources and *sparseness* which means that the sources are assumed to be inactive most of the time.

Virtanen added *temporal continuity* constraint to the sparse coding paradigm [Vir03]. He used the signal model

$$\Psi^{(t)}(f) = \sum_{n=1}^{N} a_{t,n}\Psi_n(f) + \Psi_e^{(t)}(f), \tag{5.2}$$

where $\Psi^{(t)}(f)$ is the power spectrogram of the input, t is time, f is frequency, $\Psi_n(f)$ is the static power spectrum of source n, and $a_{t,n}$ are time-varying gains of the sources. The term $\Psi_e^{(t)}(f)$ represents the error spectrogram. Virtanen propsed an iterative optimization algorithm which estimates non-negative $a_{t,n}$ and $\Psi_n(f)$ based on the minimization of a cost function which takes into account reconstruction error, sparseness, and temporal continuity. The algorithm was used to separate pitched and drum instruments in real-world music signals [Vir03, Vir04].

Also Abdallah and Plumbley have applied sparse coding for the analysis of music signals [Abd_]. Input data was represented as magnitude spectrograms, and sources as magnitude spectra, leading to a source mixing model which is essentially the same as in (5.2). The authors proposed an algorithm where sources were obtained using gradient-ascent inference and the time-varying gains with maximum-likelihood learning. Their results were promising, although shown only for one example case of synthesized Bach piece (2-3 simultaneous sounds). In this case, the system learned 55 spectra, 49 of which were note spectra. The authors made a very strong conclusion that "There is enough structure in music (or at least certain kinds of music) for a sparse coder learn about and detect notes in an unsuperwised way, even when the music is polyphonic. There is no need to bring any prior musical knowledge to the problem, such as the fact that musical notes have approximately harmonic spectra." [Abd_]

5.2.6 Other approaches

An important line of research has been pursued by Okuno, Nakatani, and colleaques who have demonstrated effective use of the direction-of-arrival information in segregating simultaneous speakers [Nak99, Oku99]. The system in [Nak99] was designed to segregate continuous streams of harmonic sounds, such as the voiced sections of two or three simultaneous speakers. Multiple agents were deployed to trace harmonic sounds in stereo signals. Then, the detected sounds were cancelled from the input signal and the residual was used to update the parameters of each sound and to create new agents when new sounds were detected.

The periodicity transform method of Sethares and Staley is an example of a mathematical approach to multiple-F0 estimation [Set99]. The algorithm finds a set of nonorthonormal basis elements based on data, instead of using a fixed basis as in the Fourier transform, for example. The authors proposed a residue-driven sound separation algorithm, where one periodic component at a time was estimated and cancelled from the mixture signal, and this process was then repeated for the residual. The overall approach bears a close resemblance to the iterative method of de Cheveigné in [deC93].

Marolt has used neural networks for the different subproblems of music transcription [Mar01a, 02]. In [Mar01a], the author proposes a system which is a combination of an auditory model, adaptive oscillators, and neural networks. The unitary pitch model is first used to process the input signal [Med91a,b]. Adaptive oscillators similar to those in [Lar94] were used to track partials in the output of each channel. In order to track harmonically related partials, the oscillators were interconnected to *oscillator nets*, one per each candidate musical note. A neural network was then trained for each individual note in order to recognize whether the corresponding note occurs at a given time or not. Good results were obtained for an evaluation set of three real and three synthesized piano performances. A basic problem encountered was the slow synchronization of the adaptive oscillators which caused problems especially with low notes.

A potentially very successful approach in some applications is to focus on modeling a specific musical instrument. This has been done e.g. in [Haw93, Ros98b, Kla98] where only piano music was considered.

6 Problem-Oriented Approach to Multiple-F0 Estimation

This chapter serves as an introduction to Publications [P1], [P3], and [P5]. Among these, [P5] proposes a “complete” multiple-F0 estimation system in the sense that it includes mechanisms for suppressing additive noise and for estimating the number of concurrent sounds in an input signal. These problems were not addressed in Chapter 4, although they both have to be solved in order to perform automatic transcription of real-world music signals. In publications [P1] and [P3], in turn, two different principles are proposed to deal with *coinciding frequency partials*. These are harmonic components which coincide in frequency with the partials of other sounds and thus overlap in the spectrum. This problem is of particular importance in music, as will be described in Sec. 6.4. The solution proposed in [P1] is introduced in Sec. 6.4.3 and the solution proposed in [P3] is introduced in Sec. 6.4.2. Only the latter one is used in the overall system described in [P5].

The methods in this chapter represent a very pragmatic approach to multiple-F0 estimation. The problem is decomposed into smaller subproblems and solutions to these are sought one-by-one. Also, the various sources of error are analyzed and techniques of dealing with these are sought for. In practice, these have turned out to be effective ways of improving a transcription system.

Many of the principles to be presented in this chapter were already applied in Chapter 4. However, it should be noted that the system to be described here has been developed earlier. Whereas the method in Chapter 4 is in many respects more elegant than that in [P5], an advantage of the latter is that it constitutes an “explicit” reference implementation of many basic mechanisms that are needed for successful multiple-F0 estimation. Such an implementation is quite instructive in understanding the acoustic and musical constraints of the problem, not only the auditory point of view. Also, the system presented in [P5] is quite flexible with regard to testing different system parameters and configurations.

6.1 Basic problems of F0 estimation in music signals

Fundamental frequency estimation in music signals is in many ways more challenging than that in speech signals. In music, the pitch range is wide and the sounds produced by different musical instruments vary a lot in their spectral content. The inharmonicity phenomenon has to be taken into account. Robustness in the presence of the interference of drums and percussive instruments has to be addressed. Typically several harmonic sounds are playing concurrently and harmonic components of different sounds coincide in frequency.

On the other hand, the dynamic (time-varying) properties of speech signals are more complex than those of an average music signal. The F0 values in music are temporally more stable than in speech. It is likely to be more difficult to track the F0s of four simultaneous speakers than to perform music transcription of four-voice vocal music.

The basic problems of multiple-F0 estimation can be classified in four categories:

- *Grouping problem*. Given a mixture of harmonic sounds (possibly contaminated with noise), how should the spectral components be organized to their sound sources? This issue was discussed in length in Sec. 5.2.1. Due to the inharmonicity phenomenon (see Sec. 3.1), the components of a harmonic sound cannot be simply assumed to reside at ideal harmonic positions in the spectrum.

- *Computing the saliences (or, weights, see Sec. 4.2.2) of different F0 candidates* given the partials of a sound. We use the term *harmonic summation model* to refer to the function which calculates the salience of a hypothesized F0 candidate given the parameters (amplitudes, frequencies, phases) of its harmonic partials. This is complicated by the wide pitch range and the variation in the musical instrument timbres.
- *Noise robustness*. In the case that drums or percussive instruments are present, the signal-to-noise ratio can be around zero dB from time to time.
- *Coinciding frequency partials*. In Western polyphonic music, it is rather a rule than an exception that the partials of a harmonic sound overlap with the partials of other, concurrent, sounds. Thus, it does not suffice to merely group partials to sound sources but even individual partials need to be shared between sources.

Proposed solutions to these problems are introduced in the following sections. Section 6.2 looks at the noise robustness problem. Sections 6.3.1 and 6.3.2 address the partial grouping problem and sections 6.3.3 and 6.3.4 address the F0 salience computation. Section 6.4 describes methods that can be used to resolve coinciding partials. Estimating the number of concurrent sounds is not discussed but a method for this purpose can be found in [P5].

Technical details are not described here but can be found in the original publications. As an exception, the predominant-F0 estimation is described to a more detail because the publications where this was originally done ([Kla99a]) is not included in this thesis.

6.2 Noise suppression

The definition of “noise” is subjective and depends on the application. From the viewpoint of performing multiple-F0 estimation in music signals, everything except the partials of harmonic sounds is considered as additive noise that should be suppressed. In practice, the non-harmonic parts are mainly due to drums and percussive instruments.

Noise suppression has been extensively studied in the domain of speech processing. Speech enhancement is typically based on the assumption that the background noise characteristics are slowly-varying compared to the target speech signal. This enables a two-stage approach where, first, the noise spectrum is estimated over a longer period of time and, secondly, the spectrum of the noisy speech signal is weighted so as to suppress the noise component in the mixture signal [Var98, Sta00]. The more or less standard methods of optimal Wiener filtering and spectral subtraction are widely used. More recent advances can be found e.g. in [Mar01b, Gus02, Wol03].

In music, the sounds of drums and percussive instruments are transient-like and short in duration, making it difficult to estimate the noise spectrum over a longer period of time[1]. Due to this non-stationarity, the noise-suppression algorithm proposed in [P5] estimates and removes noise independently in each analysis frame. The preprocessing method proposed in [P5] has two goals. It suppresses additive noise and flattens the spectral shape of the target harmonic sounds. The signal model assumed by the method is

$$\Psi_x(f) = \Psi_h(f)\Psi_s(f) + \Psi_n(f), \tag{6.1}$$

where $\Psi_x(f)$ is the power spectral density of an observed acoustic signal and $\Psi_s(f)$ is the

[1] This is difficult but not impossible in theory. The same drum sounds typically occur repeatedly in music.

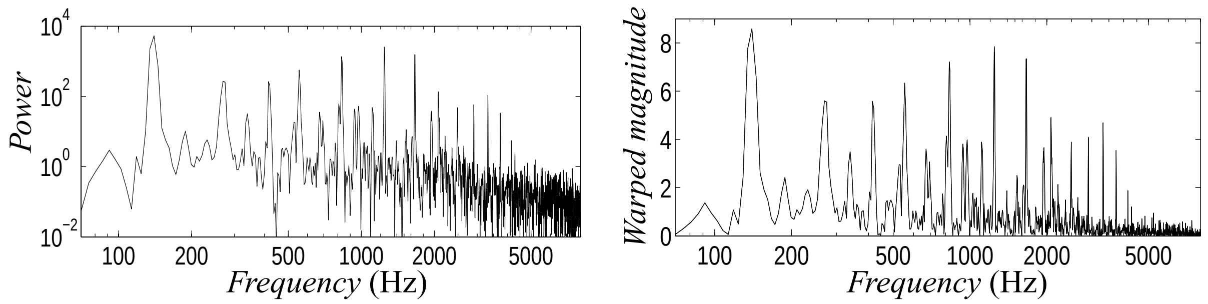

Figure 32. An example signal which contains two harmonic sounds and a snare drum sound with SNR -3dB. Left panel shows the scaled power spectrum of the signal, $(1/g)\Psi_x(f)$. Right panel shows the warped-magnitude spectrum $\log(1 + (1/g)\Psi_x(f))$.

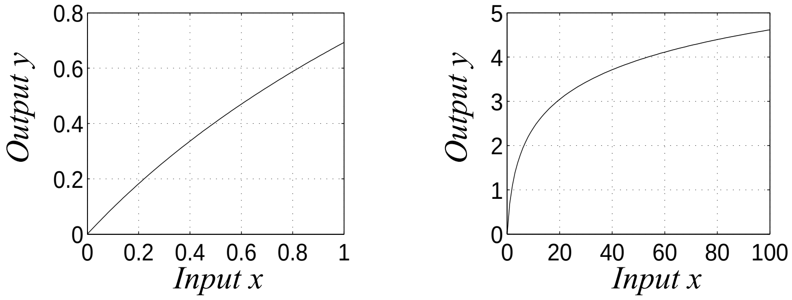

Figure 33. Illustration of the magnitude-warping transform $y = \ln(1 + x)$, when x gets values between zero and one (left), or between zero and 100 (right).

power spectrum of a vibrating system whose fundamental frequency should be measured (for example a guitar string). The factor $\Psi_h(f)$ represents the frequency response of the body of the musical instrument and other convolutive noise which filters the signal of the vibrating source. Elimination of $\Psi_h(f)$ is often referred to as spectral whitening. The term $\Psi_n(f)$ represents the power spectrum of additive noise. The additive model assumes that the signal and noise are uncorrelated.

Elimination of both $\Psi_h(f)$ and $\Psi_n(f)$ is achieved in Publication [P5] by applying *magnitude warping* which equalizes $\Psi_h(f)$ and still allows the additive noise to be linearly subtracted from the result. The power spectrum $\Psi_x(f)$ is magnitude-warped as

$$\Psi_y(f) = \ln\left\{1 + \frac{1}{g}\Psi_x(f)\right\}, \tag{6.2}$$

where the scaling factor g is adaptively calculated in each analysis frame so as to scale the level of the additive noise floor $\Psi_n(f)$ numerically close to unity. The amplitudes of the important frequency partials of the vibrating system $\Psi_h(f)\Psi_s(f)$, in turn, are assumed to be noticeably above the additive noise floor. Figure 32 illustrates the scaled power spectrum of two harmonic sounds and a snare drum sound before and after the magnitude warping. As can be seen in the left panel, the noise floor is around a unity value (10^0) after scaling, whereas the spectral peaks are significantly above this (around 10^2). It follows that when the function $\ln(1 + x)$ is applied, additive noise goes through a linear-like magnitude-warping transform, whereas the spectral peaks go through a logarithmic-like transform. The warped spectrum is shown in the right panel. Figure 33 illustrates the magnitude-warping transform when the input is scaled in different ways. As can be seen, the magnitude warping is close to a linear function for small

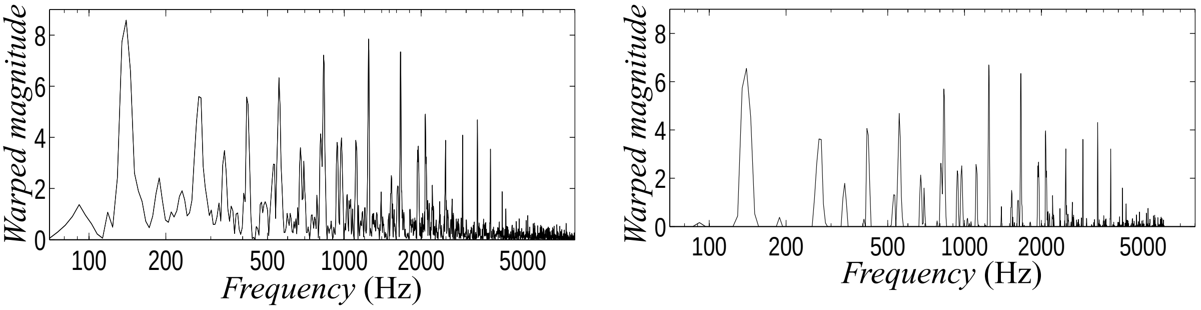

Figure 34. A magnitude-warped spectrum before and after subtracting the estimated noise spectrum. The example signal is the same as in Fig. 32.

input values. This has the nice consequence that additive noise remains additive after the warping[1].

Additive noise is suppressed by applying a specific spectral subtraction on $\Psi_y(f)$. A moving average $\hat{\Psi}_n(f)$ over $\Psi_y(f)$ is calculated on a logarithmic frequency scale and then linearly subtracted from $\Psi_y(f)$. More exactly, local averages were calculated at $2/3$-octave bands while constraining the minimum bandwidth to 100Hz at the lowest bands. The same bandwidths are used in the subsequent F0 calculations and are motivated by the frequency resolution of the human auditory system and by practical experiments with generated mixtures of musical sounds and noise.

The response $\Psi_h(f)$ is strongly compressed by the logarithmic-like transform, since subsequent processing takes place in the warped magnitude scale. Additionally, the estimated additive noise component $\hat{\Psi}_n(f)$ captures a significant amount of the convolutive noise, too, because this becomes additive in the logarithmic-like transform.

The estimated spectral average $\hat{\Psi}_n(f)$ is linearly subtracted from $\Psi_y(f)$ and resulting negative values are constrained to zero. The resulting preprocessed spectrum $\Psi_z(f)$ is used by the subsequent multiple-F0 estimator. In [P5], the preprocessed spectrum $\Psi_z(f)$ is denoted by $Z(k)$ in the discrete domain. For the sake of consistence, we use the same notation in the following.

Figure 34 shows the magnitude-warped spectrum before and after the spectral subtraction. Thus, the right-hand panel of Fig. 34 is an example of the kind of noise-suppressed input spectrum $Z(k)$ that functions as an input to the subsequent multiple-F0 computations.

6.3 Predominant-F0 estimation

Figure 35 shows the overview of the multiple-F0 estimation method proposed in [P5]. A core part of the method is the predominant-F0 estimation module which computes the saliences of different F0 candidates in the presence of other harmonic sounds and noise. The term *predominant-F0 estimation*, as defined in Sec. 4.2.5, refers to the task of finding the F0 of one (any) of the harmonic sounds in a mixture signal. Initially, we did not rule out the possibility that the algorithm would reveal all the component F0s simultaneously. However, as it turned out, the algorithm often assigns the second-highest salience to a candidate which corresponds to half or

[1] The magnitude-warping in (6.2) is closely related to μ-law compression, $y = \ln[1 + \mu x]/\ln(1 + \mu)$, where the input *x* is assumed to get values between zero and one and the value of μ can be used to decide between a linear-like and a logarithmic-like compression.

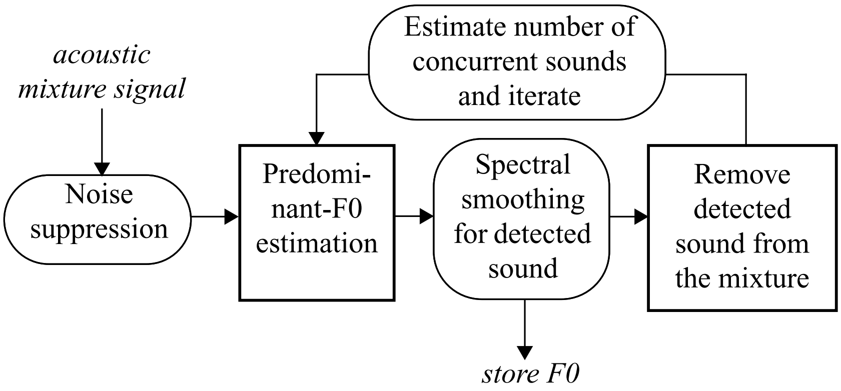
acoustic mixture signal
Estimate number of concurrent sounds and iterate
Noise suppression
Predominant-F0 estimation
Spectral smoothing for detected sound
Remove detected sound from the mixture
store F0

Figure 35. Block diagram of the multiple-F0 estimator described in [P5] (reprinted from [P5]).

twice the firstly detected correct F0 (the same was observed for the auditory-model based method as described in Sec. 4.2.7). The best solution we found for this symptom was to cancel each detected sound from the mixture spectrum and to repeat the estimation step for the residual. This leads to the iterative estimation and cancellation structure shown in Fig. 35.

The predominant-F0 algorithm was given in [P5] without explaining much how it was derived. The aim of this subsection is to describe the line of thought and the involved decisions which led to that particular algorithm.

All the processing in the proposed method takes place for the preprocessed spectrum $Z(k)$ (see Sec. 6.2 above) in one time frame of the input signal. Longer-term temporal processing is not considered and phase information is ignored.

6.3.1 Bandwise F0 estimation

The proposed predominant-F0 estimator calculates the saliences of different F0 candidates independently at different frequency bands and then combines the results to determine the global saliences. The primary motivation for attempting bandwise F0 estimation was to achieve robustness in the presence of interfering sounds. When estimation is first performed at separate frequency bands, interference (noise) in one band does not “leak” to the estimates at the other bands. This provides flexibility when the bandwise results are combined later on and enables the detection of F0s which, due to noise, are observable only at a limited frequency range.

Another issue addressed by the bandwise processing is the partial grouping problem. In a single time frame, long-term temporal features are not available but the partial have to be grouped based on their frequencies only. According to (3.1), the higher harmonics may deviate from their expected spectral positions and in this case even the intervals between them are not constant. However, we can assume the spectral intervals to be piecewise constant at sufficiently narrow bands. Thus we utilize spectral intervals between partials to group them at distinct frequency bands, and then combine the results across bands in a manner that takes inharmonicity into account. A third reason to resort to bandwise processing is that it is an important principle in the human auditory perception [Med91a, Bre90,p.247, Hou95,Moo97a].

Figure 36 illustrates the magnitude responses of the 18 frequency bands at which the bandwise F0 estimation takes place. For the sake of algorithmic flexibility, all calculations are performed in the frequency domain. This has the advantage that bandwise operation can be achieved via a

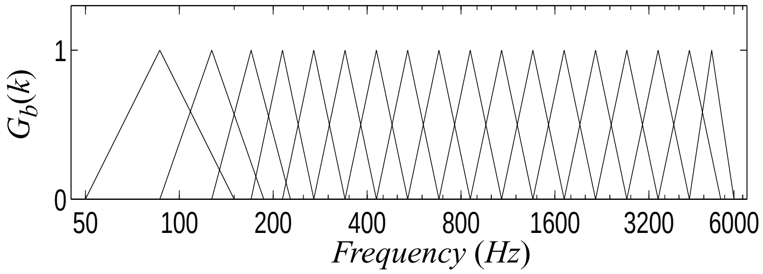

Figure 36. Magnitude responses of the 18 frequency bands at which the bandwise F0 estimation takes place. Each band comprises a 2/3-octave region of the spectrum, constraining, however, the minimum bandwidth to 100Hz (reprinted from [P5]).

single fast Fourier transform, after which local regions of the spectrum are separately processed.

6.3.2 Harmonic selection

Another defining characteristic of the proposed algorithm is that the salience of a hypothesized F0 candidate is calculated using only the frequency components which are considered to belong to the corresponding sound. This principle, *harmonic selection*, was discussed in Sec. 4.2.2 and has been originally proposed by Parsons in [Par76]. Here, the mechanism that is used to select the partials differs considerably from that in Sec. 4.2.2 but the basic idea is the same. Using only the selected partials instead of the overall spectrum in F0 computations provides some robustness in sound mixtures.

Bandwise saliences of different F0 candidates are computed as follows. Let $L_b(n)$ be a vector of F0 saliences at band *b*. Here *n* corresponds to the fundamental frequency $F_n = f_s n / K$, where K is the frame size and f_s is the sampling rate. Let $Z_b(k)$ be the preprocessed spectrum (as described in Sec. 6.2) which is additionally filtered with the response of the bandpass filter at band *b* (the responses are shown in Fig. 36). The frequency components (or, bins) at band *b* are denoted by $k \in [k_b, k_b + K_b - 1]$, where k_b is the lowest bin at band *b* and K_b is the number of bins at the band. The bandwise saliences $L_b(n)$ are calculated by finding a series of every n^{th} frequency components at band *b* that maximizes

$$L_b(n) = \max_{m \in \boldsymbol{M}} \Phi\{Z_b(k_b + m), Z_b(k_b + m + n), \ldots, Z_b(k_b + m + n(J(m, n) - 1))\}, \quad (6.3)$$

where

$$J(m, n) = \lceil (K_b - m) / n \rceil \quad (6.4)$$

is the number of the equidistant partials at the band and the function Φ (i.e., the harmonic summation model, see Sec. 6.1) remains to be specified later. The set $\boldsymbol{M} = \{0, 1, \ldots, n - 1\}$ contains the different offsets of the series of partials. The offset *m* is varied to find the maximum of (6.3), which is then stored in $L_b(n)$. Different offsets have to be tested because the series of higher harmonic partials may have shifted due to inharmonicity.

In (6.3), the problem of finding the partials that belong to the candidate F0 is solved by defining that the group of *equidistant partials which maximizes the function* Φ constitutes the sought group of partials. The salience of the F0 candidate *n* at band *b* is then defined as the value of the function Φ for those partials.

Figure 37 illustrates the calculations for a single harmonic sound at the band $b = 12$ between

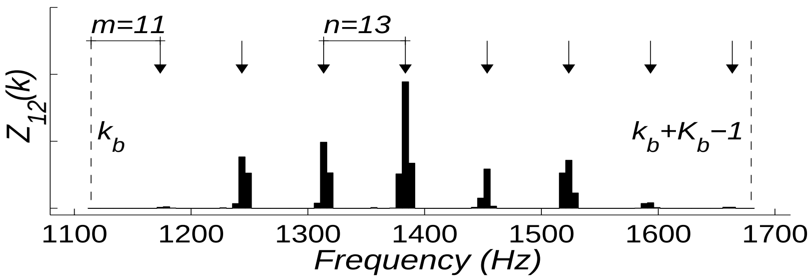

Figure 37. Harmonic selection according to (6.3) (reprinted from [P5]).

1100Hz and 1700Hz. The arrows indicate the series of frequency components which maximizes $L_{12}(n)$ for the true F0 (corresponding to $n = 13$ in this case).

The values of the offset *m* are further restricted to physically realistic inharmonicities, a subset of $\boldsymbol{M}$. The exact limit is not critical, therefore (3.1) with a constant inharmonicity factor $\beta = 0.01$ can be used to determine the maximum allowable offset from the ideal harmonic positions. In the extreme case where there is only one harmonic partial at the band (i.e. $J(m, n) = 1$), inharmonicity is not allowed at all but the partial is selected from the ideal harmonic position in the spectrum. As a consequence, the "spectral-interval oriented" harmonic selection in (6.3) reduces to a special case where spectral-location information is used for harmonic selection.

6.3.3 Determining the harmonic summation model

The remaining problem is to determine the function Φ in (6.3) which computes the salience of a F0 candidate *n* based on the magnitudes of the selected harmonic components. We use $F_n = f_s n / K$ to denote the fundamental frequency corresponding to *n*. Also, we use $\boldsymbol{O}_b^{(n)}$ to denote the set of the equidistant frequency bins at band *b* that maximizes (6.3) for candidate *n*.

The function Φ was found via a two-stage process. As a starting point, we used a function which estimates the *perceived loudness* of the set of the partials $\boldsymbol{O}_b^{(n)}$ at band *b*. In the second step, the function was parametrized and machine-learning techiques were used to find such parameters that, "most of the time", the function gives the highest salience to the correct F0.

Given that the set $\boldsymbol{O}_b^{(n)}$ for candidate *n* at band *b* contains at least one partial[1], the loudness of the partials can be estimated as

$$\hat{L}_b(n) = \sum_{k \in \boldsymbol{O}_b^{(n)}} \left[\Psi_{x_b}(f_k)^{0.23} \left(F_n \frac{d}{df} e(f_k) \right) \right], \tag{6.5}$$

where $\Psi_{x_b}(f)$ is the power spectral density of the input signal at band *b* and $f_k = f_s k / K$ is the frequency of partial *k* (note that the above expression is used here as the background model; the continuous power spectrum $\Psi_{x_b}(f_k)$ will be replaced by the noise-suppressed discrete spectrum in the following). The exponent 0.23 performs power spectrum compression according to a recent model of loudness perception [Moo97b] and the overall sum approximates the integral over the excitation pattern caused by partials $\boldsymbol{O}_b^{(n)}$ on a critical-band scale. The coefficient

[1] Inharmonicity is not allowed for the lowest partials as described in the previous subsection (Sec. 6.3.2). Consequently, there are frequency bands which do not contain any partials of a certain F0 candidate *n*.

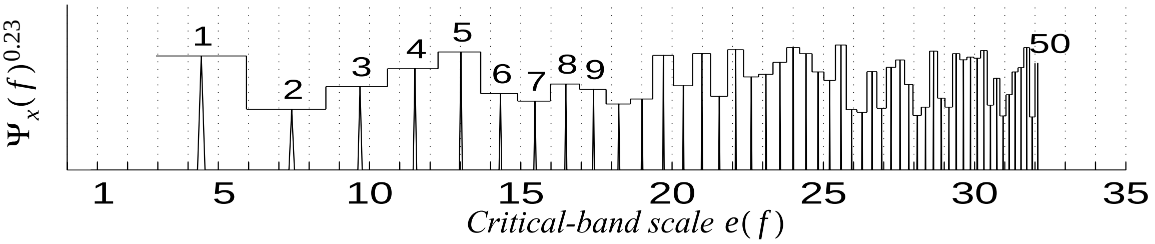

Figure 38. The first 50 overtone partials of a harmonic sound (F0 140Hz) on a critical-band scale. The dotted vertical lines indicate the boundaries of adjacent critical bands. Expression (6.5) measures the area under the stepwise curve separately for each band *b*.

$$F_n \frac{d}{df} e(f_k) \tag{6.6}$$

is the inter-partial interval around partial *k* on a critical-band scale. The mapping $e(f)$ from the linear frequency scale to the critical-band scale is given by (4.6). Figure 38 illustrates the spectral area that is summed in (6.5), one critical band at a time. (Note that this model of loudness is simplified and does not limit the effect of the lowest partials to one critical band around the partials.)

By derivating, it turns out that (6.6) is equal to $\alpha F_n / u(f_k)$, where $u(f_k)$ is the ERB value of the auditory filter centered on f_k and α is a constant scalar which can be omitted. Note that this is equivalent to the concept of resolvability as defined in (4.57) on page 50. In order to simplify (6.5), we replace $u(f_k)$ by the bandwidths $f_s K_b / K$ of the F0 estimator (see Fig. 36). The coefficient in (6.6) then, can be replaced by n / K_b and (6.5) can be written in a simplified form as

$$\tilde{L}_b(n) = \frac{n}{K_b} \sum_{k \in \boldsymbol{O}_b^{(n)}} \Psi_{x_b}(f_k)^{0.23} . \tag{6.7}$$

In a subsequent step, the above formula was parametrized and machine learning techniques were used to find the optimal values of the parameters. In the parametrized formula, $1 / J(m, n)$ is used as the variable in place of n / K_b. These two are closely related, as can be seen in (6.4). Also, $\Psi_{x_b}(f_k)^{0.23}$ is replaced by the discrete preprocessed spectrum $Z_b(n)$. The preprocessing step involves compression and allows us to omit the exponent 0.23. The parametrization of (6.7) is given as

$$L_b(n) = \left(a_0 + a_1 \frac{1}{J(m, n)}\right) \sum_{k \in \boldsymbol{O}_b^{(n)}} Z_b(n) , \tag{6.8}$$

where the parameters a_0 and a_1 are to be learned[1].

A well-posed machine-learning problem requires the definition of the *task* to do, the *performance measure* to use, and the *experience* from which to learn [Mit97]. Here, the task assigned to the algorithm was to estimate the F0s of isolated musical sounds. The performance measure used was the percentage of cases where the maximum of $L_b(n)$ at different bands corresponded to the true F0 of the sound in question. The learning algorithm had access to the cor-

[1] It should be noted that the formula in (6.8) can no more be considered as estimating the loudness of partials $\boldsymbol{O}_b^{(n)}$. This is due to the described departures from the original model in (6.5). The loudness model was used as a starting point but precise modeling of loudness as such is not of interest here.

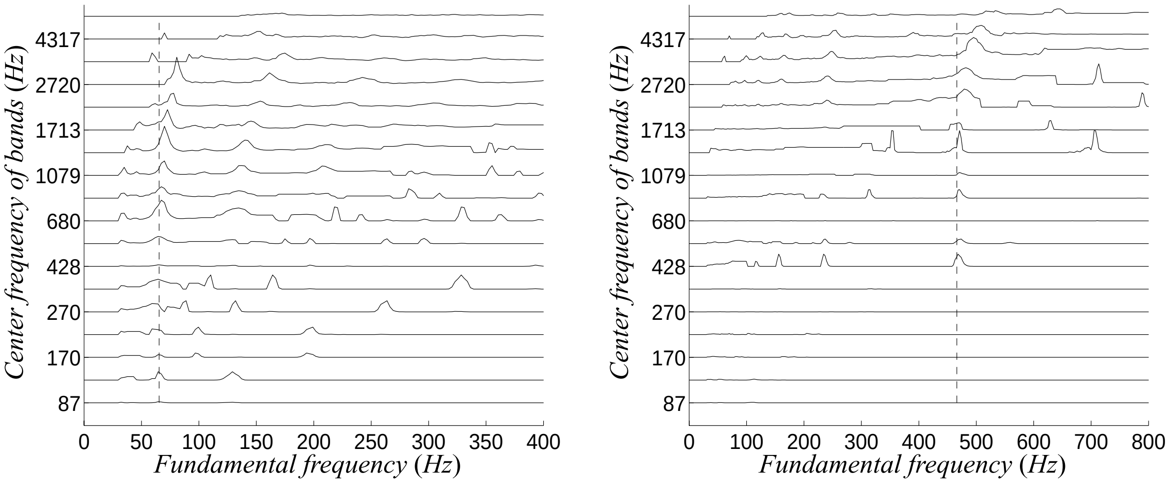

Figure 39. Bandwise-calculated F0 saliences $L_b(n)$ for two piano tones. The vectors are displaced vertically for clarity. The true F0s of the two sounds are 65Hz and 470Hz in the left and right panels, respectively, and are indicated with dashed vertical lines (reprinted from [P5]).

rect F0s of the sounds and attempted to improve the measure of success by changing the values of the coefficients a_0 and a_1. As a results, the values $a_0 = 0.25$ and $a_1 = 0.75$ were learned. A second-order polynomial of $1/J(m, n)$ was trained, too, but it did not perform significantly better than the first-order model which was therefore taken into use.

Thus, the function in (6.3) which computes the salience $L_b(n)$ of a F0 candidate *n* at a frequency band *b* becomes

$$L_b(n) = \max_{m \in \boldsymbol{M}} \left\{ c(m, n) \sum_{j=1}^{J(m, n)} Z_b(k_b + m + n(j-1)) \right\}, \tag{6.9}$$

where

$$c(m, n) = 0.25 + 0.75/J(m, n) \tag{6.10}$$

and $J(m, n)$ is as defined in (6.4).

6.3.4 Cross-band integration and estimation of the inharmonicity factor

Figure 39 shows the calculated salience vectors $L_b(n)$ at different bands for two isolated piano tones. The vectors are arranged in increasing band center frequency order. As expected, the maximum salience is usually assigned to the true F0, provided that there is a harmonic partial at that band. The inharmonicity phenomenon appears in the the two panels: the fundamental frequencies show a rising trend as a function of band center frequency.

The bandwise F0 saliences are combined to yield a global F0 estimate. A straightforward summation across the salience vectors does not accumulate them appropriately since the F0 estimates at different bands may not match for inharmonic sounds, as can be seen in Fig. 39. To overcome this, the inharmonicity factor is estimated and taken into account. Two different inharmonicity models were implemented, the one given by (3.1) and another mentioned in [Fle98, p.363]. In simulations, the performance difference between the two was negligible. The model in (3.1) was adopted.

Global saliences *L*(*n*) are obtained by summing squared bandwise saliences $L_b(n)$ that are

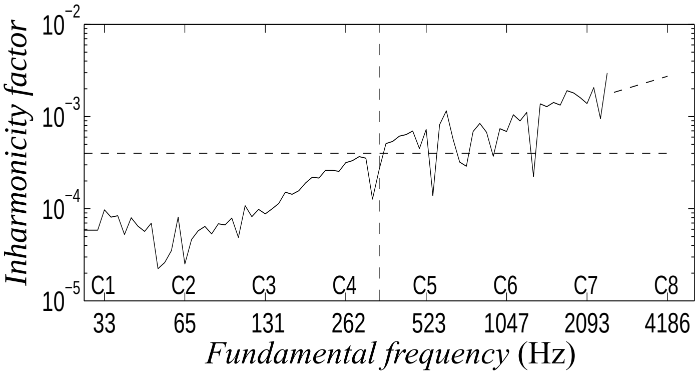

Figure 40. Measured inharmonicity factors β for piano strings. The value β=0.0004 given by Fletcher and Rossing [Fle98] for the middle register of the piano is indicated with a dashed horizontal line.

selected from different bands according to a curve determined by (3.1). A search over possible values of the inharmonicity factor $\beta(n)$ is conducted for each n, and the highest $L(n)$ and the corresponding $\beta(n)$ are stored in the output. Squaring the bandwise F0 saliences prior to summing was found to provide robustness in very noisy cases where the pitch may be detectable at a limited frequency range only. Weighting of different bands according to their estimated signal-to-noise ratios was not attempted.

The described method yields the inharmonicity factors β (see (3.1)) of the detected sounds as a side-product. Figure 40 illustrates the measured inharmonicity factors β for the different notes of an upright piano. The measured data agrees with that given by Fletcher and Rossing for the piano in [Fle98, pp.363,390].

6.4 Coinciding frequency partials

The described predominant-F0 estimator operates reliably even in cases where several concurrent harmonic sounds are present. However, the method is not able to estimate all the component F0s simultaneously but the iterative estimation and cancellation procedure shown in Fig. 35 has to be applied.

The partials of a detected predominant-F0 cannot be completely removed from the mixture spectrum. This kind of "partial grouping" is not appropriate for coinciding partials[1]. The spectral components that are due to several coinciding partials need to be shared between the corresponding sounds. If the partials of a detected sound are completely removed, the coinciding partials of other sounds are deleted in the subtraction procedure. After several iterations, a sound remaining in the residual spectrum may become too corrupted to be correctly analyzed in the iterations that follow.

In addition to the above-described issue, coinciding partials of other sounds bring noise to the F0 salience calculations in (6.9). Although this problem is less severe, it sometimes causes errors in the predominant-F0 estimation.

The aim of this section is to introduce two different techniques to deal with coinciding partials. The method proposed in [P3] is able to resolve coinciding partials to a certain degree and is

[1] In practice, coinciding partials do not need to have *exactly* the same frequencies. The partials can be considered to coincide if their frequency difference is smaller than the width of the mainlobe of the spectrum of the time-domain analysis window.

introduced in Sec. 6.4.2. This mechanism is applied in the final system in Publication [P5]. The method introduced in Sec. 6.4.3 has been originally proposed in [P1] and is oriented towards avoiding the use of coinciding partials when a sound is being observed.

6.4.1 Diagnosis of the problem

When two sinusoidal partials with amplitudes a_1 and a_2 and phase difference θ_Δ coincide in frequency, the amplitude of the resulting sinusoid can be calculated as

$$a_s = \left|a_1 + a_2 e^{i\theta_\Delta}\right| . \tag{6.11}$$

If the two amplitudes are roughly equivalent, the partials may either amplify or cancel each other, depending on their phases. However, if one of the amplitudes is significantly larger than the other, as is usually the case, a_s is close to the maximum of the two.

The condition that a harmonic partial h of a sound S coincides a harmonic j of another sound R can be written as $hF_S = jF_R$, where F_S and F_R are the fundamental frequencies of the two sounds. Here ideal harmonicity is assumed, which is valid for many important classes of sound sources and for the lower-order harmonics $(h < 10)$ of all the instruments considered in this work. When the common factors of integers h and j are reduced, we obtain

$$F_R = \frac{p}{q} F_S , \tag{6.12}$$

where $(p, q) \geq 1$ are integer numbers. This implies that partials of two sounds can coincide only if the fundamental frequencies of the two sounds are in rational number relationships. Furthermore, when the fundamental frequencies of two sounds are in the above relationship, then every p^{th} harmonic pk of the sound S coincides every q^{th} harmonic qk of the sound R, where $k = 1, 2, \ldots$. This is evident since hF_S equals jF_R for each pair $h = pk$ and $j = qk$, when (6.12) holds. If $p = 1$, the sound R overlaps all the partials of sound S at their common frequency bands.

An important principle governing Western music is paying attention to the pitch relations, *intervals*, of simultaneously played notes. Simple harmonic relations satistying Eq. (6.12) are favoured over dissonant ones. The smaller the values of p and q are, the closer is the harmonic relation of the two sounds and the more perfectly they play together. For instance, fundamental frequencies in relationships 4 : 5: 6 constitute a basic *major* chord and fundamental frequencies in relationships $(1/6) : (1/5) : (1/4)$ constitute a basic *minor* chord. Because harmonic relations are so common in music, these “worst cases” must be handled well in general. Also, this partly explains why multiple-F0 estimation is particularly difficult in music.

Western music arranges notes to a quantized logarithmic scale[1], where the fundamental frequency of a note n is $F_n = 440 \times 2^{n/12}$ Hz, and $-48 \leq n \leq 39$ in the standard piano keyboard, for example. Although the scale is logarithmic, it can surprisingly well produce the different harmonic F0 relationships that can be derived by substituting small integers to (6.12) [Kla98]. Table 8 shows some basic musical intervals. As can be seen, the realizable F0 relationships deviate a little from their harmonic ideals, but the amount of error is so small that it is not aurally disturbing to an average human listener. Moreover, for a feasible frequency analysis resolution, the coinciding of the partials appears as perfect[2].

[1] The scale is often called *twelve tone equal-tempered scale*.

Table 8: Some basic musical intervals.

Interval name	Size (semitones)	Ideal F0 relationship	Deviation from ideal relationship
octave	12	2:1	0.0%
perfect fifth	7	3:2	–0.11%
perfect fourth	5	4:3	+0.11%
major third	4	5:4	+0.79%
minor third	3	6:5	–0.91%
major second	2	9:8	–0.23%
minor second	1	16:15	–0.68%

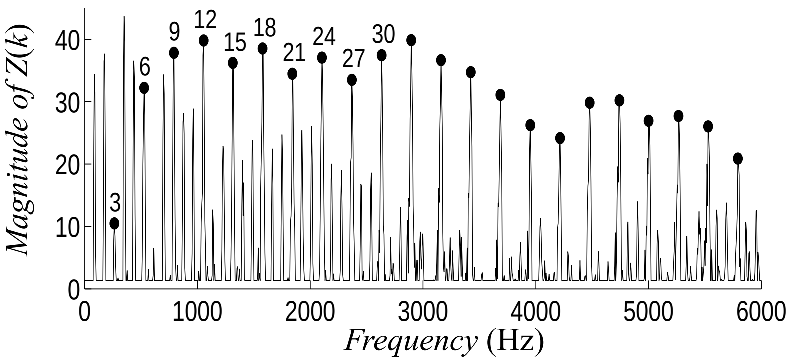

Figure 41. Preprocessed spectrum $Z(k)$ containing two sounds with F0s in the relation 1:3 (reprinted from [P5]).

6.4.2 Resolving coinciding partials by the spectral smoothness principle

The amplitudes and phases of coinciding frequency partials can no more be deduced from their sum. However, by making certain assumptions concerning the involved musical sounds, it is possible to resolve the component partials to a certain degree. A method for this purpose has been proposed in [P3] and is now introduced.

Consider the preprocessed spectrum of two concurrent harmonic sounds in Fig. 41. The F0s of the two sounds are in $1 : 3$ relationships and, as a consequence, the partials of the higher-pitched sound coincide with every third harmonic of the lower-pitched sound. As predicted by (6.11), the coinciding partials randomly cancel or amplify each other at the low frequencies, whereas at the higher frequencies the summary amplitudes approach the maximum of the two, i.e., the spectral envelope of the higher sound.

In cases such as that in Fig. 41, the lower sound is usually detected first because it captures much of the power of the higher-pitched sound, too. In general, sounds which are able to “steal” much of the energy of the other sounds are often detected first. Removing the partials of these sounds completely would effectively corrupt the other sounds.

[2] The described principles generalize beyond Western music. Sethares has presented an extensive analysis of how music utilizes F0 relationships so as to cause partials of concurrent sounds coincide and to make the sounds “blend” better [Set98]. Also, he draws an interesting connection between the applied musical instruments and the musical scales in different cultures.

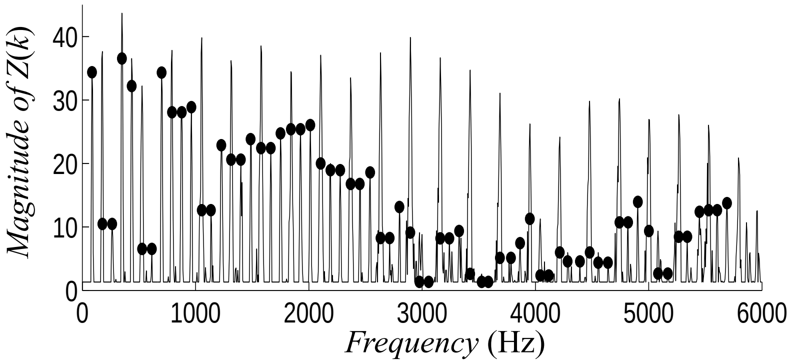

Figure 42. The dots in the spectrum illustrate the spectral envelope of the lower-pitched sound as estimated by (6.13). The example signal is the same as in Fig. 41.

An algorithm for revealing the underlying magnitudes of coinciding partials is found by imitating the mechanisms of the human auditory system. As described in Sec. 3.5.3, the unitary pitch model performs implicit *spectral smoothing*, especially for the unresolved harmonic partials. Each two adjacent harmonic partials cause amplitude beating, i.e., alternatingly amplify and cancel each other at the fundamental frequency rate. However, the magnitude of the beating is determined by the smaller of the two amplitudes. When the amplitude envelope of a harmonic sound is considered, this has the consequence that single higher-amplitude harmonic partials are filtered out.

The described smoothing mechanism can be isolated into a separate algorithm which amounts to nonlinear filtering of the spectral envelope of a detected sound (i.e., the magnitudes of the harmonic partials of the sound). In [P3], three different smoothing algorithms have been described and evaluated. The simplest of these replaces the amplitude a_h of a harmonic partial h with the minimum of the amplitudes of the harmonic h and its neighbour $h + 1$:

$$a_h \leftarrow \min(a_h, a_{h+1}) . \tag{6.13}$$

Interestingly, even this simple operation is rather efficient in estimating the spectra of harmonic sounds in polyphonic musical signals. Considering Fig. 41 again, (6.13) would do a good job in estimating the spectral envelope of the lower-pitched sound. Figure 42 illustrates the amplitudes of the lower-pitched sound as estimated by (6.13) for this example. More sophisticated and accurate algorithms exist, though, and these are described in [P3].

In physical terms, the described approach can be understood as relying on the assumption that the spectral envelopes of musical sounds are smooth, i.e., relatively slowly-varying as a function of frequency. This assumption was discussed on page 55.

The estimated magnitudes are used in subtracting the detected sound from the mixture spectrum. Compared to the case where the partials are completely removed, this is much safer. Also, the estimated magnitudes can be used in (6.9) to recalculate a refined salience of the F0 candidate in question and to subdue the noise caused by the coinciding partials of other sounds. The spectral smoothness principle is used for both purposes in [P5].

6.4.3 Identifying the harmonics that are the least likely to coincide

Another approach to deal with coinciding partials has been proposed in [P1]. The method is based on identifying the harmonic partials that are the least likely to have been corrupted by the coinciding partials of other sounds. Those partials are then given more weight when making observations concerning a hypothesized harmonic sound in a mixture signal.

The starting point of the method in [P1] is the assumption that if a harmonic sound *S* with fundamental frequency F_S occurs in music, it is quite probable that other harmonic sounds R_i with fundamental frequencies $F_{R_i} = (p/q)F_S$ (here integers $(p, q) \geq 1$) occur simultaneously. This is due to the principles of Western music as discussed in Sec. 6.4.1. Moreover, it is reasonable to assume that all values of *p* and *q* binding the fundamental frequencies F_{R_i} are equally probable, except that small values are in general more probable than large values. As a consequence, the partials of an interfering sound R_i are equally probable to overlap any subset of every p^{th} partial of *S*. The set of every p^{th} partial of *S* is here denoted by

$$E_p = \{pk\}, k = 1, 2, \dots . \tag{6.14}$$

Let us denote by π_0 the probability that an interfering sound R_i overlaps some subset E_p of the sound *S*. The probability π_0 is assumed to be the same for all the sets E_p as mentioned above. However, the likelihood that an *individual* harmonic partial does not coincide with the partials of the other sounds is not equal to all different harmonics *h*. Instead, the likelihood is proportional to the probability that none of the sets E_p that the partial *h* belongs to is overlapped. This can be calculated as $(1 - \pi_0)^{D(h)}$, where $D(h)$ is the number of subsets E_p that the harmonic *h* belongs to. It is easy to prove that $D(h)$ is the number of integers that divide *h*, $D(1) = 1$. An integer *a* is defined to divide another integer *b*, if and only if $b = da$ for some integer *d*. Here, perfect harmonicity is assumed.

The method in [P1] utilizes the above analysis in order to make reliable observations of a harmonic sound in polyphonic musical signals. The most fundamental observation, of course, is whether a hypothesized sound exists in the mixture signal or not. This is reduced to the question whether the individual harmonics of a sound appear in the spectrum or not. However, there are two types of *outlier* partials, i.e., harmonics that are not valid to represent a hypothesized sound. Some harmonics may be due to the coinciding partials of other sounds only, whereas some harmonics may be missing from the target sound even when it is present. In [P1], a weighted order-statistical filter [Kuo94, Ast97] is proposed which is able to filter out the outlier values. Moreover, the sample selection probabilities of the filter are set according to the relative trustworthiness of different harmonics. The h^{th} *sample selection probability* is the probability that the sample *h* in a set is selected to the output of the filter [Kuo94].

A complete description of the method can be found in [P1]. The method has been extensively used in a transcription system for piano music as described in [Kla98]. The method as such is not applied in the multiple-F0 estimation method in [P5]. However, a spectral smoothing algorithm which utilizes the statistical dependencies of the subsets E_p achieves a slight improvement in [P5].

6.5 Criticism

The multiple-F0 estimation method described in this chapter has a certain major weakness. That is, the iterative cancellation of the detected sounds is performed by separating the spectra of the sounds in the frequency domain. Estimation and separation of the individual higher-order harmonic partials cannot be done reliably. The smoothing mechanism partly saves the day because it prevents from completely removing the higher-order partials and thus from destroying the corresponding frequency range of the mixture spectrum.

The successfulness of the frequency-domain separation depends critically on the resolution of the spectrum. As a consequence, the described method requires a relatively long analysis frame to perform well. The method proposed in Chapter 4 is advantageous in this respect because it uses a different mechanism to estimate and cancel the higher-order (unresolved) partials. As a result, the method in Chapter 4 achieves a better accuracy, particularly in shorter time frames, as can be seen in Table 6 on page 65.

The method presented here has some advantages compared to that in Chapter 4, as well. These were mentioned in the beginning of this chapter.

7 Conclusions and future work

7.1 Conclusions

7.1.1 Multiple-F0 estimation

Main part of this thesis deals with *multiple-F0 estimation* which was considered to be the core of the music transcription problem. Two different methods were proposed for this purpose. The first was derived from the unitary pitch model in Chapter 4 and the other, as originally published in [P5], is oriented towards more pragmatic problem-solving. The obtained results indicate that multiple-F0 estimation can be performed reasonably accurately at the level of a single time frame. For a variety of musical sounds, *a priori* knowledge of the sound sources is not necessary, although this might further improve the performance.

The method described in [P5] represents a "complete" multiple-F0 estimation system in the sense that it includes mechanisms for suppressing additive noise and for estimating the number of concurrent sounds in an input signal. Also, it provides an explicit reference implementation of many basic mechanisms that are needed in multiple-F0 estimation. However, the method described in Chapter 4 is more accurate and, in particular, operates more reliably in short analysis frames. A comparative evaluation of the two methods was presented in Table 6 on page 65. The performance advantage of the method described in Chapter 4 is due to the principle that higher-order (unresolved) harmonic partials are processed collectively; estimation and separation of individual higher-order partials is not attempted. The author of this work was quite surprised at the efficiency of the combined use of spectral-location and spectral-interval information, basically directly according to the model for half-wave rectification in (4.27).

Both of the proposed multiple-F0 estimation methods are based on an iterative estimation-and-cancellation approach. The method described in [P5] *separates* the spectra of detected sounds from the mixture, whereas the method described in Chapter 4 does not separate but, rather, *cancels the effect* of detected sounds from the mixture signal. The very reason to resort to the iterative approach was that we could not find any other technique that would have led to a comparable degree of accuracy. A particularly attractive property of the iterative approach is that at least a couple of the most prominent F0s can be detected even in rich polyphonies. The probability of error increases rapidly in the course of iteration but, as described in [P5], this appears to be at least partly due to the inherent characteristics of the problem itself: some sounds in a mixture signal are more difficult to detect and remain in the residual till the last iterations. The sounds which are aurally the most prominent are usually detected first.

It seems unlikely that the frame-level multiple-F0 estimation accuracy could get substantially better using bottom-up signal analysis techniques. The two methods converge close to the same error rate (although the auditory-model based method is superior in short analysis frames) and the performance of both methods is comparable to that of human listeners, as described in [P5]. However, substantial performance improvements can still be expected by utilizing longer-term acoustic features and by constructing internal musicological models or sound sources models. These will be discussed in Sec. 7.2 below.

7.1.2 Musical meter estimation

The proposed musical meter estimator is fairly successful in estimating the meter of different types of music signals. This conclusion was drawn by comparing the obtained results with those of the two reference systems in Publication [P6] and, informally, by auralizing meter estimatimation results as a generated drum track along with the original piece.

As mentioned in Sec. 1.3, pitch information is *not* utilized in meter estimation. This basic decision contradicts with what we know about the human cognition of music but, nevertheless, was made for two main reasons. First, the proposed meter analysis is computationally very efficient compared to the multiple-F0 analysis. Secondly, the meter estimator benefits of a relatively good time-resolution which cannot be adequately provided by the multiple-F0 estimators. The disadvantage of ignoring the pitch information is that meter analysis at the measure-pulse level is not very reliable. Since the measure-pulse correlates with harmonic changes, pitch information would most probably improve the accuracy. Despite this criticism, the simulation results indicate that the meter of a large part of musical material can be analyzed without resorting to multiple-F0 analysis. The most important elements of a successful meter estimator turned out to be measuring the degree of musical accentuation as a function of time and modeling primitive musical knowledge which governs musically meaningful meter abstractions.

7.2 Future work

7.2.1 Musicological models

The scope of this thesis was restricted to bottom-up signal analysis methods. As mentioned in Sec. 1.2.2, however, the use of *musicological information* is almost equally important in the automatic transcription of real-world musical material. Although the accuracy of the proposed multiple-F0 methods is comparable to that of trained humans in musical-chord identification tasks, the accuracy of the methods is still inferior in the transcription of *continuous* musical pieces. This is largely due to the fact that the proposed methods do not include any internal “language model” for music but, instead, consider each individual analysis frame separately, apart from its context. Temporal continuity of musical sounds or melodic phrases is not taken into account at all. In brief, the program performs multiple-F0 estimation but it does not understand anything about music. Demonstrations of the transcription of continuous musical pieces using the described musically-agnostic system are available at [Kla03b].

There are straightforward and efficient ways of representing musicological knowledge. As an example, consider the following experiment. We represented combinations of co-occurring notes as 12-bit numbers that we call *chord unigrams*. There are 4096 such unigrams. Each bit signifies the presence/absence of one of the 12 *pitch classes*[1]. A total of 359 MIDI songs were collected and cut into segments where note onsets and offsets do not occur. The harmonic content of each segment was then represented with the corresponding unigram. The probability of occurrence for each unigram was computed within the pieces and averaged over all pieces. The results were interesting: among the 30 most probable unigrams were the 12 single notes (pitch classes), seven different *major triad* chords, five *minor triads,* and three *minor-seventh* chords.

The described kind of “brute force” statistical approach has several advantages. First, the estimated prior probabilities of different F0 combinations can be used to rate the likelihoods of several competing F0 hypotheses in a transcription system. Secondly, the described estimation

[1] The pitch class “*c*”, for example, represents all *c* notes in different octaves since these play the same harmonic role. The principle that the notes *c*3 (130Hz), *c*4 (260Hz), *c*5 (520Hz) etc. are considered equivalent is called *octave equivalence*. The twelve pitch classes are: *c, c#, d, d#, e, f, f#, g, g#, a, a#, h.*

procedure involves *no* heuristic parameters or rules. Thirdly, no musical expertise was employed, yet the system knows about major and minor triad chords, the building blocks of Western harmony. New music types can be addressed simply by re-estimating the unigram probabilities using different training material.

The described experiment with chord unigrams is merely an example of the probabilistic approach to musicological modeling. Similar formulations can be proposed for the temporal continuity of melodies and for harmonic progression, for example. Readily-collected statistics have been published e.g. in [Kru90, pp. 67, 181, 195]. More complex rules governing Western music can be found in music-theory workbooks. Temperley has proposed a very comprehensive rule-based system which models the cognition of basic musical structures [Tem01]. He used the system for the automatic musicological analysis of MIDI files. From the point of view of music transcription, a remaining challenge is to transform these rule-based models into *probabilistic* models which are able to evaluate the likelihoods of several candidate analyses already during the transcription process.

7.2.2 Utilizing longer-term temporal features in multiple-F0 estimation

In Sec. 5.2.1, several *perceptual cues* were listed which promote the grouping of time-frequency components to a same sound source in human listeners. The cues, when present, facilitate the auditory organization (analysis) of sound mixtures. The proposed multiple-F0 estimators utilize two cues extensively: harmonic frequency relationships of partials and spectral smoothness of their amplitude values. In contrast to this, a certain important feature was not utilized at all: *synchronous changes* of time-frequency components. For example, the components belonging to a same sound typically set on simultaneously, they may exhibit synchronous frequency-modulation (vibrato) or amplitude-modulation, or the components may have a “common fate”, such as synchronously ascending frequencies. All these cues are commonly present in real-world music signals.

There is a straightforward way of utilizing the longer-term temporal features in multiple-F0 estimation. The proposed multiple-F0 estimation methods perform the analysis in a single time frame. Moreover, the analysis frames can be relatively short (down to 46ms) in the case of the auditory-model based method. Taking hanning-windowing into account, it is meaningful to compute the F0-salience vectors $\lambda(\tau)$ every 23 milliseconds. Consider calculating the *difference* between two temporally successive F0-salience vectors, $\lambda^{(t+1)}(\tau) - \lambda^{(t)}(\tau)$. In a quite typical musical situation, several long-duration notes are playing in the background and, on top of this static harmonic background, a sequence of shorter notes (a melody) is played. When a new sound sets on, it appears as a peak in the *differential F0-salience*, whereas the long-duration notes do not pop up because they are present in both the past and the future frames. In other words, the differential includes only the freshly onsetting sound and, from the point of view of the differential F0-salience, the *polyphony is virtually one* in this case.

The above-described principle can be used to model all the “synchronous changes” cues. A clear peak in the differential F0-salience vector occurs exactly when all the partials of a certain F0 change synchronously. For example, when several sounds are playing and only one of the sounds exhibits vibrato, the sound with vibrato comes up in the differential F0-salience. This is because the peaks corresponding to the vibrato-sound in successive F0-salience vectors are in different positions τ. In a practical implementation, instead of merely picking maxima in the F0-salience vectors themselves, it might be reasonable to inspect the differential of F0-salience, too, or, to pick peaks in the weighted sum of the two. From the point of view of psychoacoustics, we know that the auditory-nerve response is very strong at the onset of a sound but then steadily falls to a lower “adaptation” level when the sound continues playing [Med86]. The use of differential F0-salience would not depart much from this basic principle and is able to model the perceptual effect of the “synchronous changes” cues in auditory organization.

7.3 When will music transcription be a “solved problem”?

An important fact about music transcription is that it is *difficult*. The problem is at best comparable to automatic speech recognition which has been studied for fifty years and is only now becoming practically applicable. In music transcription, the development will probably be faster as the computational power is already available and we can borrow theoretical methods and approaches from speech recognition. However, the problem is really not in finding fast computers but in discovering the mechanisms and principles that humans use when listening to music. Modelling perception is difficult because the world in which we live is complex, because the things that humans create are complex (music being just one example), and because the human brain is complex.

Anyone who claims to have a quick solution to the polyphonic transcription problem, or a single mechanism that solves the problem once and for all – is mistaken. The human brain combines a large number of processing principles and heuristics. We will be searching for them for years, perhaps even decades, before arriving at, say, 95% of a skilled musician’s accuracy and flexibility.

There is a certain factor which may crucially change the above prediction regarding the *time* needed to release an accurate general-purpose[1] music transcriber. This has to do with the generative nature of music versus speech. The development of speech recognition systems is constantly confronted with the problem that the amount of targeted and carefully annotated training data is limited. Synthetic speech is not valid for training a speech recognizer. In music transcription, the very problem stems from *combinatorics*: the sounds of different instruments occur in varying combinations and make up musical pieces. The dynamic variability and complexity of a *single* sound event is not as high as that of speech sounds[2]. For these reasons, we argue that synthetic music *is* valid for training a music transcriber. In principle, astronomical amounts of training data can be generated since acoustic measurements for isolated musical sounds are available, combinations of these can be generated by mixing, and effects can be added. An exact reference annotation is immediately available.

The availability of training data helps us to automatize away the most frustrating part of algorithm development: parameter optimization. However, it does not free us from designing the methods themselves. It is quite unlikely that the transcription problem could be solved simply by training a huge neural network, for example. The “space” of possible algorithms and models may be even larger than we can think of. The interesting part is to explore this space in a meaningful and efficient way until we have found the necessary ingredients of a successful transcription system. People from different disciplines are needed in this pursuit, including signal processing, acoustics, computer science, music, linguistics, and experimental psychology.

[1] The concept *music* is not well-defined: in principle anything can be called music. The scope of a “general-purpose” transcriber is here limited to signal types that are commonly regarded as music.

[2] Even for singing, transcribing the melody is significantly easier than recognizing the lyrics.

Bibliography

[Abd03] S. A. Abdallah and M. D Plumbley, “Probability as metadata: Event detection in music using ICA as a conditional density model,” In *Proc. 4th International Symposium on Independent Component Analysis and Blind Signal Separation*, Nara, Japan, April 2003.

[Abd_] S. A. Abdallah and M. D. Plumbley, “Sparse coding of music signals,” Submitted to *Neural Computation*.

[AKo01] AKoff Sound Labs, *AKoff Music Composer* (transcription software), 2001.
URL: http://www.akoff.com/

[Ale01] J. S. Alexander, K. A. Daniel, and T. G. Katsianos, “Apparatus for detecting the fundamental frequencies present in polyphonic music,” United States Patent Application Publication, Pub. No. US 2001/0045153 A1, Nov. 29, 2001.

[All90] P. E. Allen and R. B. Dannenberg, “Tracking Musical Beats in Real Time,” In *Proc. International Computer Music Conference*, San Francisco, 1990.

[Ara03] Arakisoftware, *AmazingMIDI* (transcription software), 2003.
URL: http://www.pluto.dti.ne.jp/~araki/amazingmidi/

[ANS73] American National Standards Institute, ANSI. *Psychoacoustical terminology.* American National Standards Institute, New York, 1973.

[Ast97] J. T. Astola and P. Kuosmanen, *Fundamentals of Nonlinear Digital Filtering*. CRC Press, 1997.

[Bar01] J. Barker, M. Cooke, and P. Green, “Robust ASR Based On Clean Speech Models: An Evaluation of Missing Data Techniques For Connected Digit Recognition in Noise,” in *Proc. Eurospeech*, Aalborg, Denmark, 2001.

[Bel99] S. D. Bella and I. Peretz, “Music agnosias: Selective impairments of music recognition after brain damage,” *Journal of New Music Research*, 28(3), 209–216, 1999.

[Bel03] J. P. Bello, “Towards the Automated Analysis of Simple Polyphonic Music: A Knowledge-Based Approach,” Ph.D. thesis, Univ. of London, 2003.

[Bil93] J. Bilmes, “Timing is of the Essence: Perceptual and Computational Techniques for Representing, Learning, and Reproducing Expressive Timing in Percussive Rhythm,” M.Sc. thesis, Massachusetts Institute of Technology, 1993.

[Bra99] M. Braun, “Auditory midbrain laminar structure appears adapted to f_0 extraction: further evidence and implications of the double critical bandwidth,” *Hearing Res.*, 129, 71–82, 1999.

[Bra00] M. Braun, “Inferior colliculus as candidate for pitch extraction: multiple support from statistics of bilateral spontaneous otoacoustic emissions,” *Hearing Res.* 145, 130–140, 2000.

[Bre90] A. S. Bregman, *Auditory Scene Analysis*. The MIT Press, Cambridge, Massachusetts, 1990.

[Bro92a] G. J. Brown. “Computational auditory scene analysis: A representational approach,” Ph.D. thesis, Dept. of Comp. Sci., Univ. of Sheffield, 1992.

[Bro94] G. J. Brown and M. Cooke, “Perceptual grouping of musical sounds: A computational model,” *J. of New Music Research* 23, 107–132, 1994.

[Bro91] J. C. Brown and B. Zhang, “Musical frequency tracking using the methods of conventional and "narrowed" autocorrelation," *J. Acoust. Soc. Am.*, 89 (5), 2346–2354, 1991.

[Bro92b] J. C. Brown, "Musical fundamental frequency tracking using a pattern recognition method," *J. Acoust. Soc. Am*., 92 (3), 1394–1402, 1992.

[Bro93a] J. C. Brown, "Determination of the meter of musical scores by autocorrelation," *J. Acoust. Soc. Am.*, 94 (4), 1953-1957, 1993.

[Bro01] J. C. Brown, "Feature dependence in the automatic identification of musical woodwind instruments," *J. Acoust. Soc. Am.*, 109 (3), 1064-1072, 2001.

[Bro93b] L. Brown (Ed.), *The New Shorter Oxford English Dictionary.* Oxford University Press, New York, 1993.

[Bui04] BuildOrBuy Group Network, *List of Wav to Midi conversion and audio transcription software*, 2004. URL: http://www.buildorbuy.org/wavtomidi.html

[Car92] N. Carver and V. Lesser, "The evolution of blackboard control architectures," Massachusetts Amherst University Technical Report 92-71, Oct. 1992.

[Cem01] A. T. Cemgil, B. Kappen, P. Desain, and H. Honing, "On tempo tracking: Tempogram representation and Kalman filtering," *Journal of New Music Res.*, 28(4), 259–273, 2001.

[Cem03] A. T. Cemgil and B. Kappen, "Monte Carlo Methods for Tempo Tracking and Rhythm Quantization," *Journal of Artificial Intelligence Research* 18, 45-81, 2003.

[Cha82] C. Chafe, B. Mont-Reynaud, and L. Rush, "Toward an Intelligent Editor of Digital Audio: Recognition of Musical Constructs". *Computer Music Journal*, 6 (1), Spring 1982.

[Cha86a] C. Chafe and D. Jaffe, "Source separation and note identification in polyphonic music," in *Proc. IEEE International Conf. on Acoust., Speech, and Signal Processing*, Tokyo, 1986.

[Cha86b] C. Chafe and D. Jaffe, "Techniques for Note Identification in Polyphonic Music," in *Proc. IEEE International Conference on Acoustics Speech and Signal Processing*, Tokyo, 1986.

[Cla02] L. P. Clarisse, J. P. Martens, M. Lesaffre, B. De Baets, H. De Mayer, and M. Leman, "An Auditory Model Based Transcriber of Singing Sequences," In *Proc. International Conference on Music Information Retrieval*, Paris, France, Oct. 2002.

[Cla99] E. F. Clarke, "Rhythm and Timing in Music," In *The Psychology of Music*, D. Deutsch, (Ed.), Academic Press, 1999.

[Cla04] Classical Archives LLC, *Classical Archives* (an archive of classical music), 2004.
URL: http://www.classicalarchives.com/

[Coo91] M. Cooke. "Modelling Auditory Processing and Organisation," Ph.D thesis, Dept. of Comp. Sci., Univ. of Sheffield, 1991.

[Coo94] M. Cooke and G. J. Brown, "Separating Simultaneous Sound Sources: Issues, Challenges and Models," in *Fundamentals of speech synthesis and speech recognition*, E. Keller, (Ed.), J. Wiley, pp. 295-312, 1994.

[Coo01a] M. Cooke and D. P. W. Ellis "The auditory organization of speech and other sources in listeners and computational models," *Speech Communication*, 35(3–4), 141–177, 2001.

[Coo01b] M. Cooke, P. Green, L. Josifovski, and A. Vizinho, "Robust automatic speech recognition with missing and unreliable acoustic data," *Speech Communication*, 34 (3), 267–285, 2001.

[Cor90] T. H. Cormen, C. E. Leiserson, and R. L. Rivest, *Introduction to Algorithms*. MIT Press, Cambridge, Massachusetts, 1990.

[Dav02] M. Davy and S. Godsill, “Detection of Abrupt Spectral Changes using Support Vector Machines. An application to audio signal segmentation,” In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, Orlando, Florida, May 2002.

[Dav03] M. Davy and S. J. Godsill, “Bayesian harmonic models for musical signal analysis, ” In *Bayesian Statistics VII*, J.M. Bernardo, J.O. Berger, A.P. Dawid, and A.F.M. Smith (Eds.), Oxford University Press, 2003.

[Dav87] W. B. Davenport and W. L. Root, *An Introduction to the Theory of Random Signals and Noise*, IEEE Press, 1987.

[deC93] A. de Cheveigné, “Separation of concurrent harmonic sounds: fundamental frequency estimation and a time-domain cancellation model of auditory processing,” *J. Acoust. Soc. Am.*, 93 (6), 3271–3290, 1993.

[deC99] A. de Cheveigné and H. Kawahara, “Multiple period estimation and pitch perception model,” *Speech Communication* 27, 175–185, 1999.

[deC01] A. de Cheveigné and H. Kawahara, “Comparative evaluation of F0 estimation algorithms,” in *Proc. Eurospeech*, Copenhagen, Denmark, 2001.

[deC02] A. de Cheveigné and H. Kawahara, “YIN, a fundamental frequency estimator for speech and music,” *J. Acoust. Soc. Am.*, 111 (4), April 2002.

[Dep93] Ph. Depalle, G. García, and X. Rodet, “Tracking of partials for additive sound synthesis using hidden Markov models,” In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Minneapolis, Minnesota, 1993.

[Dep97] Ph. Depalle and T. Hélie, “Extraction of spectral peak parameters using a short-time Fourier transform modeling and no sidelobe windows,” In *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, New York, 1997.

[Des99] P. Desain and H. Honing, “Computational models of beat induction: the rule-based approach,” *Journal of New Music Research*, 28 (1), 29-42, 1999.

[Deu99] D. Deutsch (Ed.), *The Psychology of Music*. Academic Press, San Diego, 1999.

[Dix01] S. Dixon, “Automatic Extraction of Tempo and Beat from Expressive Performances,” *J. New Music Research* 30 (1), 39-58, 2001.

[Dov91] B. Doval and X. Rodet, “Estimation of fundamental frequency of musical sound signals,” in *Proc. IEEE International Conf. on Acoust., Speech, and Signal Processing*, 1991.

[Dov93] B. Doval and X. Rodet. “Fundamental frequency estimation and tracking using maximum likelihood harmonic matching and HMM’s,” In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, 1993.

[Dun61] H. K. Dunn, “Methods of Measuring Vowel Formant Bandwidths,” *J. Acoust. Soc. Am.*, 33 (12), 1737–1746, 1961.

[Dux03] C. Duxbury, J. P. Bello, M. Davies, and M. Sandler, “Complex domain onset detection for musical signals,” In *Proc. 6th Int. Conference on Digital Audio Effects*, London, UK, Sep. 2003.

[Ell95] D. P. W. Ellis and D. F Rosenthal, “Mid-level representations for Computational Auditory Scene Analysis,” In *Proc. Workshop on Computational Auditory Scene Analysis*, Intl. Joint Conf. on Artif. Intell., Montreal, Aug. 1995.

[Ell96] D. P. W. Ellis, “Prediction-driven computational auditory scene analysis,” Ph.D. thesis, MIT Media Laboratory, Cambridge, Massachusetts, 1996.

[Ero00] A. Eronen and A. P. Klapuri, “Musical instrument recognition using cepstral coefficients and temporal features,” In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Istanbul, Turkey, June 2000.

[Ero01] A. Eronen, “Comparison of features for musical instrument recognition,” In *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, New York, Oct. 2001.

[Fiz02] D. FitzGerald, E. Coyle, and B. Lawlor, “Sub-band Independent Subspace Analysis for Drum Transcription”, in *Proc. 5th Int. Conference on Digital Audio Effects (DAFX-02)*, Hamburg, Germany, 2002.

[Fle40] H. Fletcher, “Auditory patterns,” *Reviews of Modern Physics*, 12, 47–65, 1940.

[Fle98] N. F. Fletcher and T. D. Rossing, *The physics of musical instruments.* 2nd ed. Springer–Verlag New York, 1998.

[Gla90] B. R. Glasberg and B. C. J. Moore, “Derivation of auditory filter shapes from notched-noise data,” *Hearing Research* 47, 103–138, 1990.

[God98] D. Godsmark, “A computational model of the perceptual organization of polyphonic music,” Ph.D. thesis, Dep. of Comp. Sc., Univ. of Sheffield, 1998.

[God99] D. Godsmark and G. J. Brown, “A blackboard architecture for computational auditory scene analysis,” *Speech Communication* 27, 351–366, 1999.

[Gom03] E. Gómez, A. Klapuri, and B. Meudic, “Melody Description and Extraction in the Context of Music Content Processing,” *Journal of New Music Research*, 32 (1), 2003.

[Gol73] J. L. Goldstein, “An optimum processor theory for the central formation of the pitch of complex tones,” *J. Acoust. Soc. Am.*, 54 (6), 1973.

[Goo97] M. M. Goodwin, “Adaptive Signal Models: Theory, Algorithms, and Audio Applications,” Ph.D. thesis, Univ. of California, Berkeley, 1997.

[Got95] M. Goto and Y. Muraoka, “Music understanding at the beat level — real-time beat tracking for audio signals,” In *Working Notes of the IJCAI-95 Workshop on Computational Auditory Scene Analysis*, 1995, pp. 68-75.

[Got96] M. Goto and Y. Muraoka, “Beat Tracking based on Multiple-agent Architecture — A Real-time Beat Tracking System for Audio Signals,” *In Proc. Second International Conference on Multiagent Systems*, 1996, pp.103–110.

[Got97] M. Goto and Y. Muraoka, “Real-time Rhythm Tracking for Drumless Audio Signals — Chord Change Detection for Musical Decisions,” In *Proc. IJCAI-97 Workshop on Computational Auditory Scene Analysis*, 1997, pp. 135-144.

[Got00] M. Goto, “A robust predominant-F0 estimation method for real-time detection of melody and bass lines in CD recordings,” in *Proc. IEEE International Conf. on Acoust., Speech, and Signal Processing*, Istanbul, Turkey, June 2000.

[Got01] M. Goto, “A predominant-F0 estimation method for real-world musical audio signals: MAP estimation for incorporating prior knowledge about F0s and tone models,” in *Proc. Workshop on Consistent and reliable acoustic cues for sound analysis,* Aalborg, Denmark, Sep. 2001.

[Gou01] F. Gouyon and P. Herrera, “Exploration of techniques for automatic labeling of audio drum tracks’ instruments”, in *Proc. MOSART: Workshop on Current Directions in Compu-*

ter Music, Barcelona, Spain, 2001.

[Gou02] F. Gouyon, P. Herrera, and P. Cano, “Pulse-dependent analyses of percussive music,” In *Proc. AES 22nd International Conference on Virtual, Synthetic and Entertainment Audio*, Espoo, Finland, June 2002.

[Gus02] S. Gustafsson, R. Martin, P. Jax, and P. Vary, “A Psychoacoustic Approach to Combined Acoustic Echo Cancellation and Noise Reduction,” *IEEE Trans. Speech and Audio Proc.* 10(5), July 2002.

[Hai01] S. W. Hainsworth, “Analysis of Musical Audio for Polyphonic Transcription,” 1st year report, Dept. of Eng., Univ. of Cambridge, 2001.

[Hai02] S. W. Hainsworth and M. D. Macleod, “A Physiologically Motivated Approach to Music Transcription,” Poster presented at the *Digital Music Research Network Launch Day*, Dec. 17th, 2002.

[Hai_] S. W. Hainsworth, “Techniques for the Automated Analysis of Musical Audio,” Ph.D thesis, Cambridge Univ., *to appear.*

[Han95] S. Handel, “Timbre perception and auditory object identification,” In *Hearing — handbook of perception and cognition*, B. C. J. Moore (Ed.), Academic Press, San Diego, California, 1995.

[Har96] W. M. Hartmann, “Pitch, periodicity, and auditory organization,” *J. Acoust. Soc. Am.* 100 (6), 3491–3502, 1996.

[Haw93] M. Hawley, “Structure out of sound,” Ph.D. thesis, MIT Media Laboratory, Cambridge, Massachusetts, 1993.

[Hes83] W. J. Hess, *Pitch Determination of Speech Signals*. Springer–Verlag, Berlin Heidelberg, 1983.

[Hes91] W. J. Hess, “Pitch and voicing determination,” in *Advances in speech signal processing*, Sadaoki Furui, M. Mohan Sondhi (Ed.), Marcel Dekker, Inc., New York, 1991.

[Hou90] A. J. M. Houtsma and J. Smurzynski, “Pitch identification and discrimination for complex tones with many harmonics,” *J. Acoust. Soc. Am.*, 87 (1), 304–310, Jan. 1990.

[Hou95] A. J. M. Houtsma, “Pitch Perception,” in *Hearing — handbook of perception and cognition*, B. C. J. Moore (Ed.), Academic Press, San Diego, California, 1995.

[Hu02] G. Hu and D. L. Wang, “Monaural Speech Segregation Based on Pitch Tracking and Amplitude Modulation,” Technical Report OSU-CISRC-3/02-TR6, Dept. of Comp. and Inf. Sci., Ohio State University, March 2002.

[Hut97] P. Hutchinson, *NoteChaser* (transcription software), 1997.
URL: http://www.tnsoptima.com/soundidea/notechaser.html

[Inn04] Innovative Music Systems, *IntelliScore* (transcription software), 2004.
URL: http://www.intelliscore.net/

[Iow04] University of Iowa, *The University of Iowa Musical Instrument Samples*, Iowa City, Iowa, 2004. URL: http://theremin.music.uiowa.edu/MIS.html

[IRC04] IRCAM, *IRCAM Studio Online*, Paris, France, 2004. URL: http://soleil.ircam.fr/

[ISO99] ISO/IEC FDIS 14496-3 sec5, “Information Technology — Coding of Audiovisual Objects; Part 3: Audio; Subpart 5: Structured Audio”. (International Organization for Standardization, 1999). URL: http://web.media.mit.edu/~eds/mpeg4/SA-FDIS.pdf,

[Jel97] F. Jelinek, *Statistical Methods for Speech Recognition*. MIT Press, Cambridge, Massachusetts, 1997.

[Jur00] D. Jurafsky and J. H. Martin, *Speech and Language Processing*. Prentice Hall, New Jersey, 2000.

[Jär01] H. Järveläinen, V. Välimäki, and M. Karjalainen, “Audibility of the timbral effects of inharmonicity in stringed instrument tones,” *Acoustics Research Letters Online* 2(3), July, 2001.

[Kae98] C. Kaernbach and L. Demany, “Psychophysical evidence against the autocorrelation theory of auditory temporal processing,” *J. Acoust. Soc. Am.*, 104 (4), 2298–2306, Oct. 1998.

[Kae01] C. Kaernbach and C. Bering, “Exploring the temporal mechanism involved in the pitch of unresolved harmonics,” *J. Acoust. Soc. Am.*, 110 (2), 1039–1048, Aug. 2001.

[Kar99a] M. Karjalainen, *Kommunikaatioakustiikka*. Espoo, Finland: Teknillinen korkeakoulu, Akustiikan ja äänenkäsittelytekniikan laboratorio, 1999. Report 51, in Finnish (Communication Acoustics).

[Kar99b] M. Karjalainen and T. Tolonen, “Separation of speech signals using iterative multipitch analysis and prediction,” in *Proc. 6th European Conf. Speech Communication and Technology (EUROSPEECH’99)*, Vol. 5, 2187–2190, Budapest, Hungary, Sep. 5-9, 1999.

[Kas93] K. Kashino and H. Tanaka, “A Sound Source Separation System with the Ability of Automatic Tone Modeling,” in *Proc. International Computer Music Conference*, Tokyo, 1993, pp.248-255.

[Kas95] K. Kashino, K. Nakadai, T. Kinoshita, and H. Tanaka, “Organization of hierarchical perceptual sounds: music scene analysis with autonomous processing modules and a quantitative information integration mechanism,” in *Proc. International Joint Conf. on Artificial Intelligence*, Montréal, 1995.

[Kas99] K. Kashino and H. Murase, “A sound source identification system for ensemble music based on template adaptation and music stream extraction,” *Speech Communication* 27, 337–349, 1999.

[Kat89] H. Katayose and S. Inokuchi, “The Kansei music system,” *Computer Music Journal*, 13 (4), 72–77, Winter 1989.

[Kla98] A. P. Klapuri, “Automatic transcription of music,” M.Sc. thesis, Tampere Univ. of Technology, 1998.

[Kla99a] A. P. Klapuri, “Wide-band pitch estimation for natural sound sources with inharmonicities,” 106th Audio Eng. Soc. Convention preprint No. 4906, Munich, Germany, 1999.

[Kla99b] A. P. Klapuri, “Pitch Estimation Using Multiple Independent Time-Frequency Windows,” In *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, New York, Oct., 1999.

[Kla01a] A. P. Klapuri, “Automatic transcription of musical recordings,” in *Proc. Consistent and Reliable Acoustic Cues Workshop*, D. P. W. Ellis, M. Cooke (Chairs), Aalborg, Denmark, Sep. 2001.

[Kla01b] A. P. Klapuri. “Means of integrating audio content analysis algorithms,” 110th Audio Engineering Society Convention preprint, Amsterdam, Netherlands, 2001.

[Kla03a] A. P. Klapuri, “Automatic transcription of music,” *In Proc. Stockholm Music Acoustics Conference*, Stockholm, Sweden, Aug. 2003.

[Kla03b] A. P. Klapuri, *Automatic transcription of music demonstrations*, 2003.

URL: http://www.cs.tut.fi/~klap/iiro/.

[Kla95] Klassner, F., Lesser, V., and Nawab, H., “The IPUS Blackboard Architecture as a Framework for Computational Auditory Scene Analysis,” In *Proc. of the Computational Auditory Scene Analysis Workshop*, International Joint Conference on Artificial Intelligence, Montreal, Quebec, 1995.

[Kru90] C. L. Krumhansl, *Cognitive Foundations of Musical Pitch*. Oxford University Press, New York, 1990.

[Kun96] N. Kunieda, T. Shimamura, and J. Suzuki, “Robust method of measurement of fundamental frequency by ACLOS — autocorrelation of log spectrum,” in *Proc. IEEE International Conf. on Acoust., Speech, and Signal Processing*, 1996.

[Kuo94] P. Kuosmanen, “Statistical Analysis and Optimization of Stack Filters,” Ph.D. thesis, Acta Polytechnica Scandinavia, Electrical Engineering Series, 1994.

[Lah87] M. Lahat, R. J. Niederjohn, and D. A. Krubsack, “A spectral autocorrelation method for measurement of the fundamental frequency of noise-corrupted speech,” *IEEE Trans. Acoust., Speech, and Signal Processing*, ASSP-35(6), 741–750, 1987.

[Lar94] E. W. Large and J. F. Kolen, “Resonance and the perception of musical meter”. *Connection science*, 6 (1), 177-208, 1994.

[Lar01] J. Laroche, “Estimating tempo, swing and beat locations in audio recordings,” In *Proc. Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, New York, Oct. 2001.

[Lee85] C. S. Lee, “The rhythmic interpretation of simple musical sequences: towards a perceptual model,” In *Musical Structure and Cognition*, I. Cross, P. Howell, and R. West Eds., Academic Press, London, 1985.

[Lee91] C. S. Lee, “The perception of metrical structure: experimental evidence and a model,” In *Representing musical structure*, P. Howell, R. West, and I. Cross, Eds., Academic Press, London, 1991.

[Ler83] F. Lerdahl and R. Jackendoff, *A Generative Theory of Tonal Music*. MIT Press, Cambridge, MA, 1983.

[Lev98] S. N. Levine, “Audio Representation for Data Compression and Compressed Domain Processing,” Ph.D thesis, Univ. of Stanford, 1998.

[Lic51] J. C. R. Licklider, “A duplex theory of pitch perception,” *Experientia* 7, 128–133, 1951.

[Lon82] H. C. Longuet-Higgins and C. S. Lee, “The perception of musical rhythms,” *Perception* 11, 115-128, 1982.

[Mah89] R. C. Maher, “An Approach for the Separation of Voices in Composite Music Signals,” Ph.D. thesis, Univ. of Illinois, Urbana, 1989.

[Mah90] R. C. Maher, “Evaluation of a method for separating digitized duet signals,” *J. Audio Eng. Soc.*, 38 (12), 956–979, 1990.

[Mah94] R. C. Maher and J. W. Beauchamp, “Fundamental frequency estimation of musical signals using a two-way mismatch procedure,” *J. Acoust. Soc. Am.* 95 (4), 2254–2263, 1994.

[Mar01a] M. Marolt, “SONIC: Transcription of Polyphonic Piano Music with Neural Networks,” In *Proc. Workshop on Current Research Directions in Computer Music*, Barcelona, Nov. 2001.

[Mar01b] R. Martin, “Noise Power Spectral Density Estimation Based on Optimal Smoothing and Minimum Statistics,” *IEEE Trans. Speech and Audio Proc.* 9(5), July 2001.

[Mar02] M. Marolt, A. Kavcic, and M. Privosnik, “Neural Networks for Note Onset Detection in Piano Music,” In *Proc. International Computer Music Conference*, Göteborg, Sweden, Sep. 2002.

[Mar96a] K. D. Martin, “A Blackboard System for Automatic Transcription of Simple Polyphonic Music,” MIT Media Laboratory Perceptual Computing Section Technical Report No. 385, 1996.

[Mar96b] K. D. Martin, “Automatic transcription of simple polyphonic music: robust front end processing,” MIT Media Laboratory Perceptual Computing Section Technical Report No. 399, 1996.

[Mar99] K. D. Martin, “Sound-Source Recognition: A Theory and Computational Model,” Ph.D. thesis, MIT Media Laboratory, Cambridge, Massachusetts, 1999.

[McA86] R. J. McAulay and T. F. Quatieri, “ Speech Analysis/Synthesis Based on a Sinusoidal Representation,” In *Proc. IEEE Trans. Acoust., Speech and Signal Processing,* 34 (4), 744–754, 1986.

[Med86] R. Meddis, “Simulation of mechanical to neural transduction in the auditory receptor,” *J. Acoust. Soc. Am.*, 79 (3), 702–711, 1986.

[Med91a] R. Meddis and M. J. Hewitt, “Virtual pitch and phase sensitivity of a computer model of the auditory periphery. I: Pitch identification,” *J. Acoust. Soc. Am.* 89 (6), 2866–2882, 1991.

[Med91b] R. Meddis and M. J. Hewitt, “Virtual pitch and phase sensitivity of a computer model of the auditory periphery. II: Phase sensitivity,” *J. Acoust. Soc. Am.* 89 (6), 2883–2894, 1991.

[Med92] R. Meddis and M. J. Hewitt, “Modeling the identification of concurrent vowels with different fundamental frequencies,” *J. Acoust. Soc. Am.* 91 (1), 233–245, 1992.

[Med97] R. Meddis and L. O’Mard, “A unitary model of pitch perception,” *J. Acoust. Soc. Am.* 102 (3), 1811–1820, 1997.

[Mel91] D. K. Mellinger, “Event formation and separation in musical sound,” PhD thesis, Center for Computer Research in Music and Acoustics, Stanford University, 1991.

[Mit97] T. M. Mitchell, *Machine Learning*. McGraw–Hill Series in Computer Science, 1997.

[Moe97] D. Moelants and C. Rampazzo, “A Computer System for the Automatic Detection of Perceptual Onsets in a Musical Signal,” In *KANSEI - The Technology of Emotion*, A. Camurri, Ed., Genova: AIMI/DIST, 1997, pp. 141-146.

[Moo75] J. A. Moorer, “On the Segmentation and Analysis of Continuous Musical Sound by Digital Computer,” Ph.D. thesis, Dept. of Music, Stanford University, 1975.

[Moo77] J. A. Moorer, “On the transcription of musical sound by computer,” *Computer Music Journal*, pp. 32–38, Nov. 1977.

[Moo95a] B. C. J. Moore (Ed.), *Hearing — Handbook of Perception and Cognition (2nd Edition).* Academic Press, San Diego, California, 1995.

[Moo95b] B. C. J. Moore, “Frequency Analysis and Masking,” In *Hearing — Handbook of Perception and Cognition*, B. C. J. Moore (Ed.), Academic Press, San Diego, California, 1995.

[Moo97a] B. C. J. Moore, *Introduction to the Psychology of Hearing*. Academic Press, London, 1997.

[Moo97b] B. C. J. Moore, B. R. Glasberg, and T. Baer. “A Model for the Prediction of Thresholds, Loudness, and Partial Loudness,” *J. Audio Eng. Soc.*, 45 (4), April 1997.

[Mur_] K. Murphy, “Dynamic Bayesian Networks,” In *Probabilistic Graphical Models*, M. Jordan (Ed.), *to appear*. URL: http://www.ai.mit.edu/~murphyk/Papers/dbnchapter.pdf

[Mus01] Music Recognition Team, *WIDI Recognition System 2.7* (transcription software), 2001. URL: http://www.hitsquad.com/smm/programs/WIDI/

[Nak99] T. Nakatani and H. G. Okuno, “Harmonic sound stream segregation using localization and its application to speech stream segregation,” *Speech Communication* 27, 209–222, 1999.

[Nii86] H. P. Nii. “The Blackboard Model of Problem Solving and the Evolution of Blackboard Architectures,” *The AI Magazine*, 7 (2), 38-53, 1986.

[Nol67] A. M. Noll, “Cepstrum Pitch Detection,” *J. Acoust. Soc. Am.*, 41 (2), 293–309, 1967.

[Nun94] D. J. E. Nunn, A. Purvis, and P. D. Manning, “Source separation and transcription of polyphonic music,” In *Proc. International Colloquium on New Music Research*, Ghent, Belgium, 1994.

[Oku99] H. G. Okuno, T. Nakatani, and T. Kawabata, “Listening to two simultaneous speeches,” *Speech Communication* 27, 299–310, 1999.

[Opo87] F. Opolko and J. Wapnick, *McGill University Master Samples* (compact disk). McGill University, Montreal, Quebec, Canada, 1987.

[Par76] T. W. Parsons, “Separation of speech from interfering speech by means of harmonic selection,” *J. Acoust. Soc. Am.*, 60, 911–918, 1976.

[Par94] R. Parncutt, “A Perceptual Model of Pulse Salience and Metrical Accent in Musical Rhythms,” *Music Perception*, 11 (4), 409-464, Summer 1994.

[Pat76] R. D. Patterson, “Auditory filter shapes derived with noise stimuli,” *J. Acoust. Soc. Am.*, 59 (3), 640–654, March 1976.

[Pat86] R. D. Patterson and B. C. J. Moore, “Auditory filters and excitation patterns as representations of frequency resolution,” In *Frequency Selectivity in Hearing*, B. C. J. Moore (Ed.), pp. 123–177. Academic Press, London, 1986.

[Pau03a] J. K. Paulus and A. P. Klapuri, “Conventional and periodic N-grams in the transcription of drum sequences,” In *Proc. IEEE International Conferences on Multimedia and Expo*, Baltimore, USA, 2003.

[Pau03b] J. K. Paulus and A. P. Klapuri, “Model-based Event Labeling in the Transcription of Percussive Audio Signals,” In *Proc. 6th International Conference on Digital Audio Effects*, London, UK, Sep. 2003.

[Per01] I. Peretz, “Music perception and recognition,” In *The Handbook of Cognitive Neuropsychology*, B. Rapp (Ed.), Hove: Psychology Press, 2001, pp. 519–540.

[Per03] I. Peretz and M. Coltheart, “Modularity of music processing,” *Nature Neuroscience*, 6(7), July 2003.

[Pis79] M. Piszczalski and B. A. Galler, “Predicting musical pitch from component frequency ratios,” *J. Acoust. Soc. Am.*, 66 (3), 710–720, 1979.

[Pis86] M. Piszczalski, “A computational model of music transcription,” Ph.D. thesis, Univ. of

Michigan, Ann Arbor, 1986.

[Pov85] D. J. Povel and P. Essens, “Perception of temporal patterns,” *Music Perception* 2(4), 411-440, 1985.

[Rab76] L. R. Rabiner, M. J. Cheng, A. E. Rosenberg, and C. A. McGonegal, “A comparative performance study of several pitch detection algorithms,” *IEEE Trans. Acoust., Speech, and Signal Processing*, ASSP-24(5), 399–418, 1976.

[Rab93] L. R. Rabiner and B.–H. Juang, *Fundamentals of Speech Recognition*. Prentice Hall, Englewood Cliffs, New Jersey, 1993.

[Rap01a] C. Raphael, “Modeling the Interaction Between Soloist and Accompaniment,” In *Proc. 14th Meeting of the FWO Research Society on Foundations of Music Research*, Ghent, Belgium, Oct. 2001.

[Rap01b] C. Raphael, “Automated Rhythm Transcription,” In *Proc. International Symposium on Music Information Retrieval*, Indiana, Oct. 2001, pp. 99-107.

[Roa96] C. Roads, *Computer Music Tutorial*. The MIT Press, Cambridge, Massachusetts, 1996.

[Rod97] X. Rodet, “Musical Sound Signal Analysis/Synthesis: Sinusoidal+Residual and Elementary Waveform Models,” In *Proc. IEEE Time-Frequency and Time-Scale Workshop*, Univ. of Warwick, Coventry, UK, 1997.

[Ros92] D. F. Rosenthal, “Machine rhythm: Computer emulation of human rhythm perception,” Ph.D. thesis, Massachusetts Institute of Technology, 1992.

[Ros98a] D. F. Rosenthal and H. G. Okuno (Editors), *Computational auditory scene analysis*. Lawrence Erlbaum Associates, Mahwah, New Jersey, 1998.

[Ros98b] L. Rossi, “Identification de sons polyphoniques de piano,” Ph.D. thesis, L’Université de Corse, Corsica, France, 1998.

[Ros90] T. D. Rossing, *The Science of Sound*. Addison–Wesley Publishing Company, Reading, Massachusetts, 1990.

[Row01] R. Rowe, *Machine Musicianship*. MIT Press, Cambridge, Massachusetts, 2001.

[Rus95] S. J. Russell and P. Norvig. *Artificial intelligence — a modern approach*. Prentice-Hall Inc., 1995.

[Ryy04] M. Ryynänen, “Probabilistic Modelling of Note Events in the Transcription of Monophonic Melodies,” M.Sc. thesis, Tampere University of Technology, March, 2004.

[Sch96] D. Scheirer, “Bregman s chimerae: Music perception as auditory scene analysis,” In *Proc. 1996 International Conference on Music Perception and Cognition*, Montreal: Society for Music Perception and Cognition, 1996, pp. 317–322.

[Sch98] E. D. Scheirer, “Tempo and beat analysis of acoustic musical signals,” *J. Acoust. Soc. Am.* 103 (1), 588-601, 1998.

[Sch00] E. D. Scheirer, “Music-Listening Systems,” Ph.D. thesis, Massachusetts Institute of Technology, June, 2000.

[Ser89] X. Serra, “A System for Sound Analysis/Transformation/Synthesis based on a Deterministic plus Stochastic Decomposition,” Ph.D. thesis, Stanford University, 1989.

[Ser97] X. Serra, “Musical Sound Modeling with Sinusoids plus Noise,” In *Musical Signal Processing*, C. Roads, S. Pope, A. Picialli, G. De Poli (Eds.), Swets & Zeitlinger Publishers, 1997.

[Set98] W. A. Sethares, *Tuning, Timbre, Spectrum, Scale*. Springer–Verlag, London, 1998.

[Set99] W. A. Sethares and T. W. Staley, "Periodicity transforms," *IEEE Trans. Signal Processing*, 47 (11), 2953–2964, 1999.

[Set01] W. A. Sethares and T. W. Staley, "Meter and Periodicity in Musical Performance," *Journal of New Music Research*, 22(5), 2001.

[Sev04] Seventh String Software, *Transcribe!* (transcription software), 2004.
URL: http://www.seventhstring.demon.co.uk/

[Sla93] M. Slaney, "An Efficient Implementation of the Patterson–Holdsworth Auditory Filter Bank," Apple Computer Technical Report #35, Perception Group, Advanced Technology Group, Apple Computer, 1993.

[Sta00] V. Stahl, A. Fischer, and R. Bippus, "Quantile based noise estimation for spectral subtraction and Wiener filtering," In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, Istanbul, Turkey, 2000.

[Ste77] M. J. Steedman, "The perception of musical rhythm and metre," *Perception* (6), 555–569, 1977.

[Ste99] A. D. Sterian, "Model-based segmentation of time-frequency images for musical transcription," Ph.D thesis, University of Michigan, 1999.

[Ste75] S. S. Stevens, *Psychophysics*. John Wiley & Sons, New York, 1975.

[Ste98] K. N. Stevens, *Acoustic Phonetics*. The MIT Press, Cambridge, Massachusetts, 1998.

[Tal95] D. Talkin, "A robust algorithm for ptch tracking," In *Speech Coding and Synthesis*, W. B. Kleijn and K. K. Paliwal (Eds.), Elseview Science B. V., 1995.

[Tem99] D. Temperley and D. Sleator, "Modeling Meter and Harmony: A Preference-Rule Approach," *Computer Music Journal*, 23(1), 10–27, Spring 1999.

[Tem01] D. Temperley, *Cognition of Basic Musical Structures*. MIT Press, Cambridge, MA, 2001.

[Ter74] E. Terhardt, "Pitch, consonance, and harmony," *J. Acoust. Soc. Am.*, 55(5), 1061–1069, May 1974.

[Ter82a] E. Terhardt, G. Stoll, and M. Seewann, "Pitch of complex signals according to virtual-pitch theory: Tests, examples, and predictions," *J. Acoust. Soc. Am.*, 71(3), 671–678, March 1982.

[Ter82b] E. Terhardt, G. Stoll, and M. Seewann, "Algorithm for extraction of pitch and pitch salience from complex tonal signals," *J. Acoust. Soc. Am.*, 71(3), 679–688, March 1982.

[Ter02] D. E. Terez, "Robust pitch determination using nonlinear state-space embedding," In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, Orlando, Florida, May 2002.

[Ter_] M. Tervaniemi, and K. Hugdahl, "Lateralization of auditory-cortex functions," *Brain Research Reviews*, to appear.

[Tol98] T. Tolonen, V. Välimäki, and M. Karjalainen, *Evaluation of Modern Sound Synthesis Methods*. Espoo, Finland: Helsinki University of Technology, Laboratory of Acoust. and Audio Sig. Proc., 1998. (Report 48).

[Tol00] T. Tolonen and M. Karjalainen, "A computationally efficient multipitch analysis model," *IEEE Trans. Speech Audio Processing*, 8(6), 708-716, Nov. 2000.

[Var98] P. Vary, U. Heute, and W. Hess, *Digitale Sprachsignalverarbeitung*. B. G. Teubner, Stuttgart, 1998.

[Ver97] T. Verma, S. Levine, and T. Meng, “Transient Modeling Synthesis: a flexible analysis/ synthesis tool for transient signals,” In *Proc. International Computer Music Conference*, Thessaloniki, Greece, Sep. 1997.

[Ver00] T. Verma, “A perceptually based audio signal model with application to scalable audio compression,” Ph.D. thesis, Stanford University, 2000.

[Vii03] T. Viitaniemi, A. Klapuri, and A. Eronen, “A probabilistic model for the transcription of single-voice melodies,” In *Proc. Finnish Signal Processing Symposium*, Tampere, Finland, May 2003.

[Vir00] T. Virtanen and A. P. Klapuri, “Separation of Harmonic Sound Sources Using Sinusoidal Modeling,” in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, Istanbul, Turkey, 2000.

[Vir01] T. Virtanen, “Audio Signal Modeling with Sinusoids Plus Noise,” M.Sc. thesis, Tampere University of Technology, 2001.

[Vir03] T. Virtanen, “Sound source separation using sparse coding with temporal continuity objective,” International Computer Music Conference, Singapore, 2003.

[Vir04] T. Virtanen, *Music Content Analysis Demonstrations*, 2004.
URL: http://www.cs.tut.fi/~tuomasv/demopage.html

[Väl96] V. Välimäki and T. Takala, “Virtual musical instruments – natural sound using physical models,” *Organized Sound*, 1(2), 75-86, Aug. 1996.

[Wan99] D. L. Wang and G. J. Brown, “Separation of Speech from Interfering sounds Based on Oscillatory Correlation,” *IEEE Trans. on Neural Networks*, 10(3), May 1999.

[Wei86] M. Weintraub, “A Computational Model for Separating Two Simultaneous Talkers,” in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, Tokyo, 1986.

[Wig73] F. L. Wightman, “The pattern-transformation model of pitch,” *J. Acoust. Soc. Am.*, 54(2), 407–416, 1973.

[Wol03] P. J. Wolfe and S. J. Godsill, “Efficient Alternatives to the Ephraim and Malah Suppression Rule for Audio Signal Enhancement,” *EURASIP J. on Applied Signal Proc.*, No. 10, Sep. 2003.

[Wu02] M. Wu, D. L. Wang, and G. J. Brown, “A multi-pitch tracking algorithm for noisy speech,” in *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, Orlando, Florida, May 2002.

[Zat01] R. J. Zatorre and P. Belin, “Spectral and Temporal Processing in Human Auditory Cortex,” *Cerebral Cortex* 11, 946–953, Oct. 2001.

[Zat02] R. J. Zatorre, P. Belin, and V. B. Penhune, “Structure and function of auditory cortex: music and speech,” *TRENDS in Cognitive Sciences*, 6(1), 37–46, Jan. 2002.

[Zil02] A. Zils, F. Pachet, O. Delerue, and F. Gouyon, “Automatic Extraction of Drum Tracks from Polyphonic Music Signals”, in *Proc. 2nd Int. Conference on Web Delivering of Music (WedelMusic2002)*, Darmstadt, Germany, 2002.

[Zwi99] E. Zwicker and H. Fastl, *Psychoacoustics — Facts and Models*. Springer Series in Information Sciences, Springer–Verlag, Berlin, Heidelberg, 1999.

Appendices

Author’s contribution to the publications

Publications [P1]–[P3] and [P5] were done by the author alone.

The algorithm in [P4] was derived and implemented by the author. In the beginning of the work, Prof. Astola helped in finding a good way of modelling half-wave rectification in the unitary model by pointing out the treatment of nonlinear analog devices in [Dav87].

The meter estimation method in [P6] was designed and implemented by the author. M.Sc. Antti Eronen contributed significantly to the mathematical formulation and description of the probabilistic model. Also Prof. Astola helped in the final formulation of the model.

Errata

In addition to some misprints, the following errors have been found in the publications.

Publication [P3]

- In [P3], in the end of Sec. 2 there is a statement:

  > “The predominant pitch estimation algorithm is capable of finding one of the correct pitches with 99% certainty even in six-voice polyphonies.”

  This should be

  > “...with more than 90% certainty even in six-voice polyphonies”.

  In the erroneous statement, the error rate was calculated as

  > <number of errors> / (<number of test cases> • <polyphony>).

  However, when calculating *predominant*-pitch estimation error rates this should have been simply

  > <number of errors> / <number of test cases>.

  The error is very annoying but does not change the conclusions of the paper in any way. The error rates in the evaluation section (in Table I) are correct.

Publication [P4]

- Equation (4) should be

  $$W_c(k) = \frac{1}{\sigma_x\sqrt{8\pi}}V_c(k) + \frac{1}{2}X_c(k),$$

  not

  $$W_c(k) = \frac{V_c(k)}{4\sigma_x} + X_c(k).$$

  This does not change the forthcoming considerations where $V_c(k)$ and $X_c(k)$ are each discussed separately. The notation σ_x refers to the standard deviation of the signal at subband *c*. This is misleading. A better notation would have been σ_c, of course.
- Equation (5) should be

  $$V_c(\delta) = \sum_{k = -K/2 + \delta}^{K/2 - \delta} [H_c(k)X(k)H_c(k-\delta)X^*(k-\delta)] \qquad ,$$

not

$$V_c(\delta) = \sum_{k = -K/2 + \delta}^{K/2 - \delta} [H_c(k)X(k)H_c(k + \delta)X^*(k + \delta)] \quad .$$

This does not change the forthcoming considerations since, as described in Sec. 2.5 of [P4], the squared magnitudes of $V_c(\delta)$ are used. The erroneous formula gives $V_c(-\delta)$ which is the complex conjugate of $V_c(\delta)$.

- Below Eq. (8) should be $\beta_2 = (4.37/1000)(f_s/K)$, not $\beta_2 = 4.37/1000$.
- Equation (13): Here, the distortion spectrum centered on $2f_c$ has been dropped. This should have been explicitly stated (dropping the distortion spectrum was mentioned earlier in the paper) and we should have denoted $V_c'(\delta)$ instead of $V_c(\delta)$. The notation $V_c'(\delta)$ is used in this thesis when referring to the spectrum $V_c(\delta)$ where the distortion spectrum has been dropped and only the spectrum centered on zero frequency is retained.

Publication [P5]

- Under equation (9) should be

  $\boldsymbol{M} = \{0, 1, \ldots, n-1\}$

  not

  $\boldsymbol{M} = \{0, 1, \ldots, k-1\}$.

- Table I (pseudocode of the core algorithm): The seventh row of the algorithm should be

  $\delta \leftarrow l_b[\sqrt{1 + 0.01[(l_b/n)^2 - 1]} - 1]$,

  not

  $\delta \leftarrow \frac{l_b f_s}{K}[\sqrt{1 + 0.01[(l_b/n)^2 - 1]} - 1]$.

  That is, δ should be expressed in frequency-bin units, not in Hertz units.

- Table I (pseudocode of the core algorithm): The second-last row of the algorithm should be

  if $k_1 > k_b + K_b - 1$ **then** $k_1 \leftarrow k_b + K_b - 1$,

  not

  if $k_1 > k_b + K_b$ **then** $k_1 \leftarrow k_b + K_b$.

- In the end of Equations (13) and (14), $\hat{N}_{(pow)}(1)$ should read $\hat{N}_{(pow)}(l)$.

Publication 1

A. P. Klapuri, “Number theoretical means of resolving a mixture of several harmonic sounds,” In *Proc. European Signal Processing Conference*, Rhodos, Greece, 1998.

NUMBER THEORETICAL MEANS OF RESOLVING A MIXTURE OF SEVERAL HARMONIC SOUNDS

Anssi Klapuri
Signal Processing Laboratory, Tampere University of Technology,
P.O.Box 553, FIN-33101 Tampere, FINLAND
Tel: +358 3 3652124; fax: +358 3 3653857
e-mail: klap@cs.tut.fi

ABSTRACT

In this paper, a number theoretical method is developed for the purpose of analyzing the spectre of a mixture of harmonic sounds. The method is based on the properties of prime numbers and on non-linear filtering. It is shown that a number theoretical approach is of vital importance in order to detect and observe harmonic sounds in musical polyphonies. The method is verified by applying it to the automatic transcription of piano music.

1 INTRODUCTION

Multiple fundamental frequency tracking is an almost unexplored area of research, although in the moniphonic case several algorithms have been proposed that are robust, commercially applicable and operate in real time. Some published efforts towards multipitch tracking have been made in the field of automatic transcription of music [1,2]. Until these days, however, the performance of the transcription systems has been very limited in polyphonic signals.

We will discuss the spectral properties of a mixture of harmonic sounds and demonstrate why single pitch tracking algorithms are not appropriate as such for use in polyphonic signals. Then we attempt to establish a number theoretical method to detect and observe harmonic sounds in polyphonic signals. This does not only concern multiple fundamental frequency tracking, but observing any of the features of harmonic sounds in polyphonic signals.

2 *FEATURE* OF A SOUND

A harmonic sound consists of a series of frequency partials, harmonics. They appear as peaks in the frequency spectrum at constant frequency intervals $f0$, with the lowest partial at frequency $f0$, which is therefore called the fundamental frequency of the sound.

We denote harmonic sounds with uppercase letters S and R. These are used consistently in such roles that sound S is being observed in the interference (presence) of a sound R, or R_i , if there are several interfering sounds. We denote the harmonic partials of a sound by h_j, where $j \geq 1$. Braces are used to denote *sets*, thus $\{h_j\}$ being a set of harmonics.

Further, we denote by $g(x)$ a *feature* of x, where x can be a sound S, or its single harmonic partial h_j. We will separate the different features by subscript characters, for example $g_F(x)$, $g_L(x)$ and $g_T(x)$ refeffing to the frequency, loudness, and onset time of x, respectively. Because the very substance of a harmonic sound is its series of equidistant sinusoid partials, any observation of a harmonic sound must rely on its harmonic partials, no matter if it is made in time or in frequency domain.

3 BASIC PROBLEM IN RESOLVING A MIXTURE OF HARMONIC SOUNDS

There are several good methods for measuring the frequency and amplitude contours and phases of the sinusoid partials in a signal [3,4,5]. Separating a *mixture* of harmonic sounds is problematic for two specific reasons.

1. It is most difficult to organize sinusoid partials to their due fundamental frequencies, because most often the harmonic series of different sounds extend to common frequency bands.
2. The amplitude envelopes and phases of two sinusoids can no more be deduced from their sum, if they *overlap*, i.e. share the same frequency.

Proposition 1. If any harmonic h_j^S of a sound S is overlapped by any harmonic h_i^R of an interfering sound R, then the fundamental frequency of the sound R must be $f0_R = \frac{m}{n} \cdot f0_S$, where m and n are positive integer numbers.

Proof. The condition of a harmonic h_j^S of a sound S to be overlapped by a harmonic h_i^R of an interfering sound R can be expressed as

$$i \cdot f0_R = j \cdot f0_S . \tag{1}$$

When the common factors of j and i are reduced, this can be expressed as

$$f0_R = \frac{m}{n} \cdot f0_S , \tag{2}$$

where $(m, n) \geq 1$ and can be calculated from the integers i and j. ❑

Proposition 2. If the fundamental frequencies of two harmonic sounds S and R are $f0_S$ and $f0_R = \frac{m}{n} \cdot f0_S$, respectively, then every n^{th} harmonic h_{nk} of the sound R overlaps every m^{th} harmonic h_{mk} of the sound S, where integer $k \geq 1$.

Proof. Substituting (2) to (1) we can rewrite the condition of a harmonic h_j of a sound S to be overlapped by a harmonic h_i of an interfering sound R as

$$(i \cdot f0_R = j \cdot f0_S) \Leftrightarrow \left(i \cdot \frac{m}{n} \cdot f0_S = j \cdot f0_S \right) \Leftrightarrow (i \cdot m = j \cdot n) ,$$

which is true for each pair i=nk and j=mk, where $k \geq 1$. ❑

It is easy to see that if m=1 in equation 2, R overlaps all the harmonics of S at their common frequency bands. In this case, detecting and observing S is difficult and even theoretically ambiguous. This case will be separately discussed.

4 CERTAIN PRINCIPLES IN WESTERN MUSIC

An important principle governing music is paying attention to the frequency relations, intervals, of simultaneously played notes. Two notes are in a harmonic relation to each other if their fundamental frequencies satisfy

$$f0_2 = \frac{m}{n} \cdot f0_1 , \qquad (3)$$

where *m* and *n* are small integers. The smaller the values of *m* and *n* are, the closer is the harmonic relation of the two sounds and the more perfectly they play together.

Western music arranges notes to a quantized logarithmic scale, where the fundamental frequency of a note *k* is $f0_k = 440 \cdot 2^{k/12}$ Hz, and $-48 \le k \le 39$ in a standard piano keyboard, for example. Although the scale is logarithmic, it can surprisingly well produce the different harmonic intervals that can be derived from Equation (3) by substituting small integers to *m* and *n*. The realizable musical intervals deviate a little from their ideals, but the amount of error is so little that it practically does not disturb the human ear. Moreover, for a feasible frequency analysis resolution, the overlapping of the harmonics of the two sounds is the same as if the harmonic relation were perfect.

For instance, the fundamental frequencies of the notes in a basic *major* chord are in 4 : 5 : 6 relations to each other. Based on the proposition 2, 47%, 33% and 60% of the harmonic partials of the notes are overlapped by the other two notes in the chord. In this case, 60% of the partials of the third note would be found from the signal even in its absence. This demonstrates why the algorithms that have been designed for the detection and observation of a single harmonic sound cannot be straightforwardly applied to resolving polyhonic musical contents. Instead, we need to rethink the very kernel, how to collect the information of a sound from its harmonics.

5 PRIME NUMBER HARMONICS

Prime number harmonics {h_1, h_2, h_3, h_5, h_7,...} of a sound share a desired common property that is derived from the very definition of the prime numbers: they are divisible only by one and themselves. This has an important consequence, which will give a steadfast starting point in organizing frequency partials to their due fundamental frequencies.

Proposition 3. Any harmonic sound *R* can overlap only one prime number harmonic of a sound *S*, provided that the fundamental frequency of *R* is not $f0_R = \frac{1}{n} \cdot f0_S$, where integer $n \ge 1$. If *R* overlaps two prime number harmonics of *S*, it overlaps all the harmonics of *S*, and its fundamental frequency is in the mentioned relation to *S*.

Proof. This can be proved by assuming that two prime number harmonics of *S* are overlapped by the harmonics of *R* and showing that in this case $f0_R = \frac{1}{n} \cdot f_S$, where $n \ge 1$, and the sound *R* overlaps all the harmonics of the sound *S*.

Let $f0_S$ and $f0_R$ be the fundamental frequencies of the sounds *S* and *R*, respectively. We denote an arbitrary prime number by p_i. The condition of two prime number harmonics of *S* being overlapped by any harmonics h_j of *R* can be expressed as

$$\begin{cases} i_1 \cdot f0_R = p_1 \cdot f0_S \\ i_2 \cdot f0_R = p_2 \cdot f0_S \end{cases} , \qquad (4)$$

where p_2 can be solved as

$$p_2 = \frac{p_1 \cdot i_2}{i_1} .$$

In order for p_2 to be a prime number and not equal to p_1 , i_1 must satisfy

$$i_1 = n \cdot p_1 , \qquad (5)$$

where *n* is an integer and implies

$$i_2 = n \cdot p_2 .$$

Substituting (5) to (4) we get

$$f0_R = \frac{p_1 \cdot f0_S}{i_1} = \frac{p_1 \cdot f0_S}{n \cdot p_1} = \frac{f0_S}{n} , \qquad (6)$$

where $n \ge 1$. ❑

If Equation 6 holds, all the harmonics of *S* are overlapped by every n^{th} harmonic of *R*, based on proposition 2.

6 DEALING WITH *OUTLIER* VALUES

Let us denote the set of prime harmonics by {h_p | *p* is prime}, and the set of the features of the prime harmonics by {$g(h_p)$ | *p* is prime}, where the type of the feature is not yet fixed. Based on proposition 3, prime number harmonics of a sound *S* can be considered as independent pieces of evidence for the existence of the sound *S*, or for any of its features that can be deduced from its harmonics.

In the set of representative features {$g(h_p)$ | *p* is prime} there are two kinds of *outliers*, i.e., irrelevant values in respect of the true feature *g*(*S*) of the sound. Some prime harmonics have been disturbed by interfering sounds, while others may be totally lacking from *S*. Those values that are not outliers vary somewhat in value, but outliers are single, clearly deviated values, and invalid to represent the true feature of *S*. However, a majority of the representatives should be reliable, it being unprobable that a majority of the prime number harmonics would be either missing or each corrupted by an independent interfering sound.

This is the motivation for the design of a filter which would pick the estimated feature $\hat{g}(S)$ from the set of independent representatives {$g(h_p)$ | *p* is prime} and drop out the irrelevant values. The class of median and order statistic filters is prompted by the fact that they are particularly effective in dealing with the kind of data that was characterized above. These filters depend on *sorting* the set of representatives. Under or overestimated outlier values map to the both ends of the sorted set, and in between, the reliable samples are sorted from the smallest up to the largest value. Thus a trivial way to estimate a feature of a sound would be

$$\hat{g}(S) = median\{g(h_p) \mid p \text{ is prime}\}. \qquad (7)$$

Weighted order statistic (WOS) filters are defined in [7]. They allow convenient tailoring of the filter's sample selection probabilities. The j^{th} *sample selection probability* is the probability that the sample h_j in a set {h_j} is selected to be the output of the filter [8]. We denote the sample selection probabilities of a filter by $P_s(j)$.

7 GENERALIZATION OF THE RESULT

A still remaining shortcoming of the proposed procedure is that it utilizes only the prime number harmonics. This degrades the usability of the algorithm and makes it sensitive to the tonal content of a sound. We proceed towards a model where this defect is removed but the advantages of the set of prime number harmonics are preserved.

We denote by v a WOS filter that picks the estimated feature of a sound from the set of features of its harmonics. This can be written as

$$\hat{g}(S) = v\{g(h_j)\} . \qquad (8)$$

Further, we denote by $E_m = \{h_{mj}\}$, $j \geq 1$, a set which contains every m^{th} harmonic of a sound, starting from harmonic m. In Proposition 2 we proved that if an interfering sound R overlaps a harmonic of an observing sound, it overlaps every m^{th} harmonic of it, i.e., exactly the subset E_m.

The requirements of the filter v can now be exactly expressed as follows. Given a number N of interfering sounds, they should together contribute only up to a limited probability λ that a corrupted harmonic is chosen to the output of v. At the same time, the filter should utilize all the harmonics of the observed sound as equally as possible to make it applicable and robust to different kinds of sounds.

These requirements can be achieved by finding sample selection probabilities $P_s(j)$ for the filter v so that the selection probabilities of the N largest subsets E_m together sum up to the given limit probability λ. N largest sets that are not subsets of each other are the prime sets $\{E_m \mid m$=2,3,5,7...$\}$. E_1 is excluded since the case of all harmonics being overlapped will be discussed separately. If N is set to 1 this can be expressed as finding $P_s(j)$ in a minimizing problem

$$min\left\{ \begin{array}{c} max \\ m \geq 2 \end{array} \left\{ \sum_{i=1}^{J} P_s(m \cdot i) \right\} \right\} , \qquad (9)$$

where J denotes the total number of detectable harmonics of the observed sound.

We assume all fundamental frequencies of interfering sounds R_i to be equally probable. Based on the assumption, all m and n values binding $f0_R$ in equation 2 are equally probable, from where it follows that R is equally probable to choose to overlap any subset E_m. However, the relative trustworthiness is not the same for all the single harmonics h_j, but equals the probability that none of the sets E_m that h_j belongs to is overlapped. This is calculated as $\tau^{D(j)}$, where τ represents the overall probability of an interfering sound to overlap some subset E_m and $D(j)$ is the number of subsets E_m that harmonic h_j belongs to. It can be easily proved that $D(j)$ is the number of integers that divide j, $D(1)$=1. An integer a is defined to *divide* another integer b, if and only if $b = da$ holds for some integer d [9].

Selection probabilities $P_s(j)$ of the harmonics should be according to their probability of being trustworthy. We can therefore write $P_s(j)$ in the form

$$P_s(j) = \tau^{D(j)} , \qquad (10)$$

where $j \geq 1$, and $D(j)$ is as defined above.

We can now rewrite the requirements of the feature exraction filter v as

$$\sum_{j \in I} \tau^{D(j)} = \lambda \cdot \sum_{j=1}^{J} \tau^{D(j)} , \qquad (11)$$

where set I is defined to contain the numbers j of the harmonics h_j that belong to some of the N largest subsets $\{E_m \mid m$=2,3,5,7...$\}$. If N=1, set I simply contains even numbers up to J. Thus the left side sums the selection probabilities of the harmonics in the N largest subsets. The right side summation goes over the selection probabilities of all the harmonics and should equal unity.

From equation 11, τ can be solved. If the problem is solvable for given N, λ and J, there is only one root that is real and between 0 and 1. This root is the earlier discussed value of τ. Selection probabilities $P_s(j)$ can then be calculated by substituting τ to Equation 10, and scaling the overall sum of $P_s(j)$ to unity.

We arrive at selection probabilities $P_s(j)$, where N interfering sounds may together contribute only up to λ probability that an overlapped harmonic exists in the output. Another very important property of this algorithm is that we can flexibly make a tradeoff between the two requirements of the filter: the less we put emphasis on the robustness of the filter v in the presence of interfering sounds, the more equally the filter utilizes all the harmonics of the observed sound, and vice versa. Figure 1 illustrates the selection probabilities for N=2, λ=0.45 and J=20.

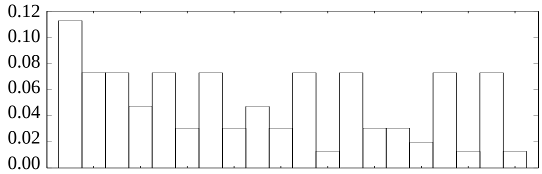

Figure 1. $P_s(j)$ for 20 harmonics, when N=2 and λ=0.45.

Thus we reduced the observation of a feature $g(S)$ of a harmonic sound S in the presence of other interfering harmonic sounds R_i to measuring the features $g(h_j)$ of the harmonics of the sound S and applying a weighted order statistic filter v to yield an estimate for $g(S)$. A design procedure to find a WOS filter which *implements* the calculated selection probabilites is presented in [10].

8 FEATURE SUBTRACTION PRINCIPLE

Our algorithm and discussion on the observation of the features of a harmonic sound in the presence of other harmonic sounds was based on an assumption that the observed sound S is not *totally* overlapped by an interfering sound R, whose fundamental frequency is $f0_R = \frac{1}{n} \cdot f0_S$.

The basic idea of our solution to this problem is to *compensate* the effect of the interfering sound R, the properties of which can be robustly extracted in the presence of S using the procedure presented before, because the interfering sound is not totally overlapped by S. Thus it will be enough to develop an algorithm to subtract, remove, or compensate, the revealed properties of the lower sound and then proceed to determine the properties of the sound *S*, which is laid bare from under the interfering sound. The subtraction process depends on the feature under inspection, and cannot be presented in a general form.

9 ALGORITHM EVALUATION

Our algorithm was evaluated and verified by applying it in a computer program whose purpose is to transcribe polyphonic piano music. The program is first allowed to study piano notes one by one, a sufficient amount to represent all the different *tone colours* that can be produced by that instrument. After this we require the program to transcribe rich polyphonic musical signals played with the same instrument, i.e., to determine the fundamental frequencies and the loudnesses of the sounds in the signals.

In all test cases, the transcription was done without knowledge of the polyphony of the transcribed signals, and with a fixed constant set of parameters. The range of fundamental frequencies was restricted to extend from 65 Hz to 2100 Hz, where five octaves and 61 piano keys fit in between. An acoustic upright piano was used in simulations.

Transcription proceeds by first detecting all potential note candidates in the spectrum, and then resolving their loudnesses one by one, using the new method. Naturally, there are much more potential note candidates than truly played notes. We call *true notes* the notes that were truly played in the recorded signals, and *false notes* those that appear as note candidates, although they were not actually played.

The goodness of the algorithm is justified by its ability to indicate the truly existing sound in the signal, i.e. the loudness of the true notes should raise clearly above the loudness of the false ones. The loudnesses of the candidates in each time segment are scaled between the values 0 to 100.

Certain note combinations were separately played and fed to the transcriber to review its ability to resolve rich musical polyphonies. Results are presented in Table 1. In the first type of tests, consonant and dissonant chords were played, two in each of the five octaves. In the second test, groups of adjacent notes in the piano keyboard were played in each octave. Third, groups of seven random notes were allotted in the allowed range of pitces and played. In each of these tests, average loudnesses among the true and false notes were calculated and recorded. Also the worst cases, where the loudness of the false notes gets closest to the true notes, was recorded. The polyphony, number of notes in each test, is also indicated.

Table 1: Relative loudnesses of the true and false notes.

Test type	Polyp.	Averages		Worst case	
		min true	max false	min true	max false
chords	3-4	64	17	78	43
groups	4-5	48	7	30	7
random	7	21	11	10	11

Chosen classical compositions were played and excerpts from them were posed to our transcription system to test its practical transcription efficiency. Here we used 25% relative loudness as a threshold to segregate between true and false notes. Weaker candidates were discarded as false notes. Results are presented in Table 2. The last piece was played by a computer on an electric piano. The effect of all notes having roughly equal playing loudness and the absence of cross resonance and noise can be noticed in the results.

Table 2: Transcription results using 25% loudness limit.

Composition	Notes in total	Typical polyphony	Missing notes	Erroneous extra notes
Für Elise, I	86	1	-	3
Für Elise, II	190	half:2 half:4	9	6
Inventio 8	205	2	15	4
Rondo alla Turca	142	3 (up to 5)	1	-

Finding the exact fundamental frequencies of the sounds in the analyzed signals proved successful in all cases. They were not assumed to be quantized to the closest legal notes.

10 CONCLUSION

The problem of resolving rich musical polyphonies was the motivation for developing the new methods. Simulations illustrate that the current system works within certain error limits up to seven notes polyphony. Especially, although increase in polyphony brings the levels of the weakest true note and the strongest false note closer to each other, the system does not totally break down even in rich polyphonies. We conclude that a number theoretical analysis of a sound mixture is the key to a robust detection and observion of harmonic sounds in the interference of each other.

REFERENCES

[1] Kashino, Nakadai, Kinoshita, Tanaka. “Application of Bayesian probability network to music scene analysis”. Proceedings of the Int. Joint Conference on Artificial Intelligence, CASA workshop, 1995

[2] Martin. “A Blackboard System for Automatic Transcription of Simple Polyphonic Music”. MIT Media Laboratory Perceptual Computing Section Technical Report No. 399, 1996.

[3] McAulay, Quatieri. “Speech analysis/synthesis based on a sinusoidal representation”. *IEEE Trans. ASSP,* 34(4), pp. 744-754, 1986.

[4] Depalle, García, Rodet. “Tracking of Partials for Additive Sound Synthesis Using Hidden Markov Models”. *IEEE Trans. on ASSP*, 1993.

[5] Serra. “Musical Sound Modeling With Sinusoids Plus Noise”. Roads, Pope, Poli (eds.). “Musical Signal Processing”. Swets & Zeitlinger Publishers, 1997.

[6] Astola, Kuosmanen. “Fundamentals of Nonlinear Digital Filtering”. CRC Press LLC, 1997.

[7] Kuosmanen. “Statistical Analysis and Optimization of Stack Filters”. Tech.D. thesis., Acta Polytechnica Scandinavia, Electrical Engineering Series, 1994.

[8] Koblitz. “A Course in Number Theory and Cryptography”. Springer, Berlin, 1987.

[9] Klapuri. “Automatic Transcription of Music”. MSc thesis, Tampere University of Technology, 1998.

Publication 2

A. P. Klapuri, “Sound onset detection by applying psychoacoustic knowledge,” In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, Phoenix, Arizona, 1999.

SOUND ONSET DETECTION BY APPLYING PSYCHOACOUSTIC KNOWLEDGE

Anssi Klapuri

Signal Processing Laboratory, Tampere University of Technology
P.O.Box 553, FIN-33101 Tampere, FINLAND
klap@cs.tut.fi

ABSTRACT

A system was designed, which is able to detect the perceptual onsets of sounds in acoustic signals. The system is general in regard to the sounds involved and was found to be robust for different kinds of signals. This was achieved without assuming regularities in the positions of the onsets. In this paper, a method is first proposed that can determine the beginnings of sounds that exhibit onset imperfections, i.e., the amplitude envelope of which does not rise monotonically. Then the mentioned system is described, which utilizes band-wise processing and a psychoacoustic model of intensity coding to combine the results from the separate frequency bands. The performance of the system was validated by applying it to the detection of onsets in musical signals that ranged from rock to classical and big band recordings.

1. INTRODUCTION

Onset detection plays an important role in the computational segmentation and analysis of acoustic signals. It greatly facilitates cut-and-paste operations and editing of audio recordings. The onset information may also be used in audio/video synchronization and timing, or passed for further analysis and recognition for example in an acoustic supervision system.

We use the term *onset detection* to refer to the detection of the beginnings of discrete events in acoustic signals. A percept of an onset is caused by a noticeable change in the intensity, pitch or timbre of the sound [1]. A fundamental problem in the design of an onset detection system is distinguishing genuine onsets from gradual changes and modulations that take place during the ringing of a sound. This is also the reason why robust one-by-one detection of onsets has proved to be very hard to attain without significantly limiting the set of application signals.

A lot of research related to onset detection has been carried out in recent years. However, only few systems have set out to solve the problem of one-by-one onset detection [1][2][3]. Instead, most systems aim at higher-level information, such as the perceived *beat* of a musical signal [4][5][6], in which case long-term autocorrelations and regularities can be used to remove single errors and to tune the sensitivity of the low-level detection process.

In this paper, we first propose a mathematical method to cope with sounds that exhibit onset imperfections, i.e., the amplitude envelope of which rises through a complex track and easily produces erroneous extra onsets or an incorrect time value. Then we propose the application of psychoacoustic models of intensity coding, which enable us to determine system parameters that apply to a wide variety of input signals. This allows processing them without a priori knowledge of signal contents or separate tuning of parameters.

The realized system was validated by applying it to the detection of onsets in musical signals. This was done mainly for two reasons. First, musical signals introduce a rich variety of sounds with a wide range of pitches, timbres and loudnesses. Different combinations of onsetting and backgrounding sounds are readily available. Second, verifying the contents of a musical signal is somewhat easier than in the case of environmental sounds. Also the concept of a perceivable onset is better defined. It should be noted, however, that the algorithm is not limited to musical signals, because the regularities and rhythmic properties of musical signals are not utilized in the detection process. The system performs reliably for input signals that ranged from rock music to classical and big band recordings, both with and without drums.

2. SYSTEM OVERVIEW

The earliest onset detection systems typically tried to process the amplitude envelope of a signal as a whole (see e.g. [7]). Since this was not very effective, later proposals have evolved towards band-wise processing. Scheirer was the first to clearly point out the fact that an onset detection algorithm should follow the human auditory system by treating frequency bands separately and then combining results in the end [4]. An earlier system of Bilmes's was on the way to the same direction, but his system only used a high-frequency and a low-frequency band, which was not that effective [2].

Scheirer describes a psychoacoustic demonstration on beat perception, which shows that certain kinds of signal simplifications can be performed without affecting the perceived rhythmic content of a musical signal [4]. When the signal is divided into at least four frequency bands and the corresponding bands of a noise signal are controlled by the amplitude envelopes of the musical signal, the noise signal will have a rhythmic percept which is significantly the same as that of the original signal. On the other hand, this does not hold if only one band is used, in which case the original signal is no more recognizable from its simplified form.

The overview of our onset detection system is presented in Figure 1. It utilizes the band-wise processing principle as motivated above. First, the overall loudness of the signal is normalized to 70 dB level using the model of loudness as proposed by Moore et al. [8]. Then a filterbank divides the signal into 21 non-overlapping bands. At each band, we detect *onset components* and determine their time and intensity. In final phase, the onset components

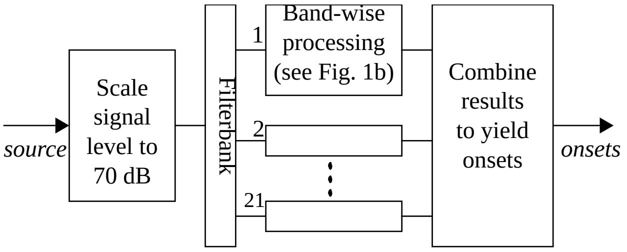
source
Scale signal level to 70 dB
Filterbank
1
Band-wise processing (see Fig. 1b)
2
21
Combine results to yield onsets
onsets

Figure 1a. System overview.

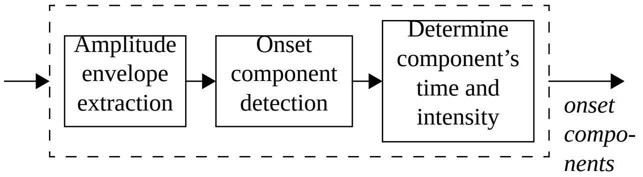
Amplitude envelope extraction
Onset component detection
Determine component's time and intensity
onset compo-nents

Figure 1b. Processing at each frequency band.

are combined to yield onsets.

Since we use psychoacoustic models both in onset component detection, in its time and intensity determination, and in combining the results, it is important to use a filterbank which can provide input to the models. Therefore, we choose a bank of nearly critical-band filters which covers the frequencies from 44 Hz to 18 kHz. The lowest three among the required 21 filters are one-octave band-pass filters. The remaining eighteen are third-octave band-pass filters. All subsequent calculations can be done one band at a time. This reduces the memory requirements of the algorithm in the case of long input signals, assumed that parallel processing is not desired.

The output of each filter is full-wave rectified and then decimated by factor 180 to ease the following computations. Amplitude envelopes are calculated by convolving the band-limited signals with a 100 ms half-Hanning (raised cosine) window. This window performs much the same energy integration as the human auditory system, preserving sudden changes, but masking rapid modulation [9][4].

3. CALCULATION OF ONSET COMPONENTS

3.1 Onset Component Detection

Several algorithms for picking potential onset candidates from an amplitude envelope function have been presented in the literature [5][6][2][4]. Despite the number of variants, practically all of them are based on the calculation of a first order difference function of the signal amplitude envelopes and taking the maximum rising slope as an onset or an onset component.

In our simulations, it turned out that the first order difference function reflects well the loudness of an onsetting sound, but its maximum values fail to precisely mark the time of an onset. This is due to two reasons. First, especially low sounds may take some time to come to the point where their amplitude is maximally rising, and thus that point is crucially late from the physical onset of a sound and leads to an incorrect cross-band association with the higher frequencies. Second, the onset track of a sound is most often not monotonically increasing, and thus we would have several local maxima in the first order difference function near the physical onset (see plots with a dashed line in Figure 2).

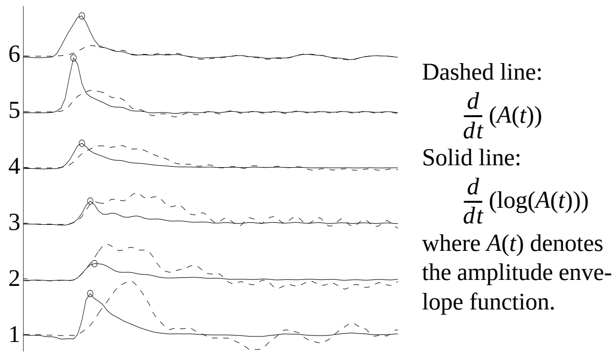

Figure 2. Onset of a piano sound. First order *absolute* (dashed) and *relative* (solid) difference functions of the amplitude envelopes of six different frequency bands.

We took an approach that effectively handles both of these problems. We begin by calculating a first order difference function

$$D(t) = \frac{d}{dt}(A(t)),$$

where $A(t)$ denotes the amplitude envelope function. $D(t)$ is set to zero where signal is below minimum audible field. Then we divide the first order difference function by the amplitude envelope function to get a first order *relative difference function W*, i.e., the amount of change in relation to the signal level. This is the same as differentiating the logarithm of the amplitude envelope.

$$W(t) = \frac{d}{dt}(\log(A(t)))$$

We use the relative difference function $W(t)$ both to detect onset components and to determine their time. This is psychoacoustically relevant, since perceived increase in signal amplitude is in relation to its level, the same amount of increase being more prominent in a quiet signal. According to Moore, the smallest detectable change in intensity is approximately proportional to the intensity of the signal [10]. That is, $\Delta I / I$, the Weber fraction, is a constant. This relationship holds for intensities from about 20 dB to about 100 dB above the absolute threshold. The function $\Delta I(t) / I(t)$ is equivalent to $W(t)$, since the frequency f in $I(t) = A(t) \cdot f$ is reduced in the division. Thus we detect onset components by a simple peak picking operation, which looks for peaks above a global threshold T_{det} in the relative difference function $W(t)$.

The relative difference function effectively solves the abovementioned problems by detecting the onset times of low sounds earlier and, more importantly, by handling complicated onset tracks, since oscillations in the onset track of a sound do not matter in relative terms after its amplitude has started rising. To clarify this, we plotted the absolute and relative difference functions of the onset of a piano sound in Figure 2. Both of the benefits discussed can be seen clearly.

3.2 Intensity of an Onset Component

Simultaneously occurring sounds combine by a linear summation. In determining the intensity of an already detected onset component, we can assume the level of backgrounding sounds to be momentarily steady and take the increase in level to be due to the onsetting sound(s). Thus the asked intensity can be picked from the first order difference function $D(t)$, multiplied by the band center frequency f_B. The intensity is needed later when onset components are combined to yield onsets of the overall signal.

An appropriate point in time to pick the intensity from $D(t)$ is not as early as where the onset was determined to occur. Instead, we scan forward up to the point where amplitude envelope starts decreasing and determine the intensity at the point of maximum slope, i.e., at the maximum value of $D(t)$ between the onset and the point where amplitude stops increasing.

After intensities has been determined for all onset components at the band, we check them through and drop out components that are closer than 50 ms to a more intense component. Remaining ones are accepted.

4. COMBINING THE RESULTS FROM THE BANDS

In the final phase we combine onset components from separate bands to yield onsets of the overall signal. For this purpose, we implemented the model of loudness as proposed by Moore, Glasberg and Baer [8]. Input to our implementation is a vector of sound intensities at third-octave bands between 44 Hz and 18 kHz, from which the program calculates the loudness of the signal in phons. To optimize the computational efficiency of the procedure, we slightly simplified the model by making the shape of the excitation pattern, i.e., the intensity spread between adjacent critical bands independent from sound pressure level. This accelerated the computations remarkably, but did not make a significant difference to the estimated loudness values for the sound intensity levels we are using.

The onsets of the overall signal are calculated as follows. First the onset components from different bands are all sorted in time order, and are regarded as sound onset candidates hereafter. Then each onset candidate is assigned a loudness value, which is calculated by collecting onset components in a 50 ms time window around the candidate and feeding their intensities to the corresponding frequency bands of the loudness model of Moore et al. Since most candidates have only a couple of contributing onset components at different bands, we must use minimum level, or background noise level for the other bands in the input of the model. Repeating this procedure to each onset candidate yields a vector of candidate loudnesses as a function of their times, as illustrated in Figure 3 for a popular music signal.

Onset loudnesses that were estimated using the abovementined procedure corresponded very well to the perceived loudnesses of the onsets in verificative listening tests. It turned out that a robust detection of onsets in very diverse kinds of signals can now be achieved by a simple peak picking operation, which looks for onset candidates above a global threshold value T_{final}. We drop out onset candidates whose loudness falls below the threshold. Then we also drop out candidates that are too close (50 ms) to a louder candidate. Among equally loud but too close candidates, the middle one (median) is chosen and the others are abandoned. The remaining onset candidates are accepted as true ones. A good value for T_{final} was found to be 25 dB for signals, whose average loudnesses had been normalized to 70 dB level.

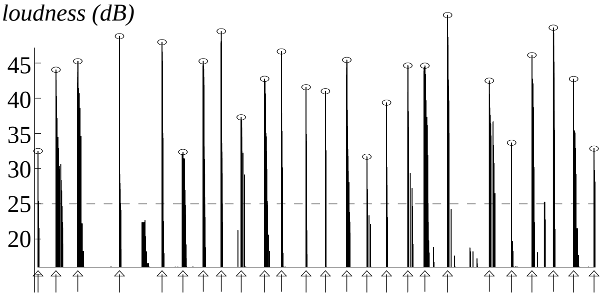

Figure 3. The loudness of onsets as a function of their time. The genuine onsets can now be quite easily discerned.

5. VALIDATION EXPERIMENTS

The presented procedure was verified by testing its performance in detecting onsets in musical signals. The signals were selected to comprise a large variation of musical instruments and a wide dynamic and pitch range. Signals both with and without drums were included. Another goal was to include representative excerpts from different musical genres, ranging from jazz and rock to classical and big band music.

Approximately ten second excerpts were sampled from each performance. These periods were carefully inspected and their onset times were marked. The excerpts were then feeded to the onset detection system and its results were compared to the manual transcription. All simulation cases were computed using the very same set of parameter values and thresholds, without separate tailoring for each simulation case. The algorithm itself was as explained above. Higher-level rhythmic properties and regularities of musical signals were not utilized in the detection.

It is interesting to note that the limitations of our detection system resemble those of human perception. We define a *pseudo-onset* to be a sound beginning, which undisputably exists in a signal, but cannot be detected by a human listener if the signal is not presented in short segments and several times. Since objective listening test could not be arranged, we regard undetected pseudo-onset as errors, too. It turned out that the detection of some pseudo-onsets could not be achieved without giving rise to several erroneous extra onsets that are due to gradual changes and modulations during the ringing of sounds.

Onset detection results for ten different musical signals are summarized in Table 1. The total number of onsets, number of undetected onsets and the number of erroneous extra onsets are given. A measure of correctness in the rightmost column is calculated as

$$correct = \frac{total - undetected - extra}{total} \cdot 100\,\%.$$

A more detailed discussion of each case follows.

Chopin's classical piano etude (op. 25, no. 4) was a trivial case. Still three onsets fell below threshold because the notes were low pitched, played softly and masked by other notes. *Al Di Meola*'s ‘Orient Blue’ represents a much more difficult case. The piece is polyphonic and employs the whole dynamic and pitch range of the acoustic guitar. Shortest inter-note intervals are only a fifteenth of a second. Good results were achieved partly because of

Table 1: Summary of onset detection results.

signal	worth notice in contents	onsets in total	unde-tected	extra	correct (%)
Chopin	acoustic piano	59	3	–	95
AldiMeola	acoustic guitar	62	5	1	92
Police	singing, el.guitar, drums	49	4	1	90
U2	el. guitar rif, distorted	19	1	2	84
Grusin	piano, percussion, drums	51	3	–	94
MDavis	brasses, double-bass	34	2	1	91
Miller	big band	46	5	1	87
Bach	chamber ensemble	51	3	1	92
Vivaldi	symphony orchestra	33	7	10	48
Beethoven	symphony orchestra	30	–	28	7

the absense of noise and other instruments.

Police's 'It's Alright for You' is from rock music genre, dominated in loudness by singing, electric guitars and drums. Onset detection is a success and resembles the results that were derived with other rock-pieces. At some moments singing produced double-onsets for phonem combinations like "-ps-", where both *p* and *s* produce an onset. All of these occurred inside the 50 ms time window, however, and were therefore fused. *U2* is an electric guitar rif, taken from the band's performance of 'Last Night on Earth'. The excerpt is played with distorted sound, without accompanying instruments. This case illustrates that even ambiguos situations, i.e., rough sounds, can be handled. *Grusin*'s 'Punta del Soul' is classified to fusion jazz, but the selected excerpt resembles mostly popular music. Various percussions included were detected without trouble.

Miles Davis's 'So What' introduces a selection of jazz band instruments: a trumpet, tenor and alto saxophones, piano, plucked double-bass and gentle drums. Both brass instrument onsets and soft pluckings of the double bass were consistently detected. *Glen Miller*'s 'In the Mood' is dominated by big band's brass instruments of the performing orchestra. All undetections occurred in a clarinet melody, which was partly masked by louder instruments.

Bach's Brandenburg Concerto was sampled from the performance of Munich Chamber Ensemble, which comprises strings, woodwinds and brass instruments. It is worth notice that onsets were detected even at moments where strings alone were carrying the rhythm and played tying consecutive notes to each other.

As a sharp contrast to the robust detections, all symphony orchestra performances turned out to be resolved very poorly. *Vivaldi*'s 'The Four Seasons' and *Beethoven*'s Symphony No. 5 are given as examples in Table 1. The clear discrepancy with human perception is not due to the type of instruments involved, since they were detected well in smaller ensembles. Instead, two causes are supposed. Firstly, individual physical sound sources can no more be followed in a symphony orchestra, but resulting onsets derive from several sources and are smoothed. Secondly, it was revealed by a certain hammond organ solo that a strong amplitude modulation at the middle frequencies confuses the system. It seems that the human auditory system has a special ability to ignore even a very loud amplitude modulation if it is inconsistent, and to concentrate on frequencies where structure is found.

6. CONCLUSIONS

We first discussed problems that arise in the one-by-one detection of sound onsets. Then a system was described, which builds upon the use of relative difference function and application of the psychoacoustic models of intensity coding. This was done in the framework of the band-wise processing idea. Experimental results show that the presented system exhibits a significant generality in regard to the sounds and signal types involved. This was achieved without higher-level logic or a grouping of the onsets. The system introduces only two thresholds that need to be experimentally found, i.e., that are not deduced from psychoacoustic metrics. These thresholds are common to all input signals.

One of the shortcomings of our method lies in its inability to deal with a strong amplitude modulation which is met in classical ensembles and in certain instrumental sounds. In general, the proposed system was well able to discern between genuine onsets and gradual changes and modulations in the sounds themselves. In the case of musical signals, an additional higher-level analysis would still significantly improve the accuracy of the system.

7. REFERENCES

[1] Moelants D., Rampazzo C. "A Computer System for the Automatic Detection of Perceptual Onsets in a Musical Signal". In Camurri, Antonio (Ed.). "*KANSEI, The Technology of Emotion*", pp. 140–146. Genova, 1997.

[2] Bilmes J. "Timing is of the Essence: Perceptual and Computational Techniques for Representing, Learning, and Reproducing Expressive Timing in Percussive Rhythm". MSc thesis, Massachusetts Instute of Technology, 1993.

[3] Schloss A. "On the Automatic Transcription of Percussive Music — From Acoustic Signal to High-Level Analysis". Ph.D. thesis, Stanford University, 1985. Report STAN-M-27.

[4] Scheirer E. "Tempo and Beat Analysis of Acoustic Musical Signals". Machine Listening Group, MIT Media Laboratory, 1996.

[5] Goto M., Muraoka Y. "Beat Tracking based on Multiple-agent Architecture - A Real-time Beat Tracking System for Audio Signals". *Proceedings of The Second International Conference on Multiagent Systems*, pp.103–110, 1996.

[6] Goto M., Muraoka Y. "A Real-time Beat Tracking System for Audio Signals". *Proceedings of the 1995 International Computer Music Conference*, pp.171–174, September 1995.

[7] Chafe C., Jaffe D., Kashima K., Mont-Reunaud B., Smith J. "Source Separation and Note Identification in Polyphonic Music". Stanford University, Department of Music, Report STAN-M-29. 1985

[8] Moore B., Glasberg B., Baer T. "A Model for the Prediction of Thresholds, Loudness, and Partial Loudness". *J. Audio Eng. Soc.*, Vol. 45, No. 4, pp. 224–240. April 1997

[9] Todd, McAulay. "The Auditory Primal Sketch: a Multiscale Model of Rhythmic Grouping". *Journal of New Music Research*, 23, pp. 25–70, 1992.

[10] Moore B. (ed). "Hearing". Handbook of Perception and Cognition, 2nd Edition. Academic Press, 1995.

Publication 3

A. P. Klapuri, “Multipitch estimation and sound separation by the spectral smoothness principle,” In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, Salt Lake City, Utah, 2001.

MULTIPITCH ESTIMATION AND SOUND SEPARATION BY THE SPECTRAL SMOOTHNESS PRINCIPLE

Anssi P. Klapuri

Tampere University of Technology, P.O.Box 553, FIN-33101 Tampere, Finland
klap@cs.tut.fi

ABSTRACT

A processing principle is proposed for finding the pitches and separating the spectra of concurrent musical sounds. The principle, spectral smoothness, is used in the human auditory system which separates sounds partly by assuming that the spectral envelopes of real sounds are continuous. Both theoretical and experimental evidence is presented for the vital importance of spectral smoothness in resolving sound mixtures. Three algorithms of varying complexity are described which successfully implement the new principle. In validation experiments, random pitch and sound source combinations were analyzed in a single time frame. Number of simultaneous sounds ranged from one to six, database comprising sung vowels and 26 musical instruments. Usage of a specific yet straightforward smoothing operation corrected approximately half of the pitch errors that occurred in a system which was otherwise identical but did not use the smoothness principle. In random four-voice mixtures, pitch error rate reduced from 18% to 8.1%.

1. INTRODUCTION

Pitch perception plays an important part in human hearing and understanding of sounds. In an acoustic environment, human listeners are able to perceive the pitches of several simultaneous sounds and make efficient use of the pitch to “hear out” a sound in a mixture [1]. Computational modeling of this function, multipitch estimation, has been relatively little explored in comparison to the availability of algorithms for single pitch estimation in monophonic speech signals [2].

Until these days, computational multipitch estimation (MPE) has fallen clearly behind humans in accuracy and flexibility. First attempts were made in the field of automatic transcription of music, but were severy limited in regard to the number of simultaneous sounds, pitch range, or variety of sound sources involved [3]. In recent years, further progress has taken place. Martin proposed a system that utilized musical knowledge in transcribing four voice piano compositions [4]. Kashino *et al*. describe a model which was able to handle several different instruments [5]. Goto’s system was designed to extract melody and bass lines from real-world musical recordings [6]. Psychoacoustic knowledge has been succesfully utilized e.g. in the models of Brown and Cooke [7], Godsmark *et al*. [8], and de Cheveigne and Kawahara [9]. Also purely mathematical approaches have been proposed [10].

Multipitch estimation and auditory scene analysis are intimately linked. If the pitch of a sound can be determined without getting confused by other co-occurring sounds, the pitch information can be used to organize simultaneous spectral components to their sources of production. Or, vice versa, if the spectral components of a source can be separated from the mixture, MPE reduces to single pitch estimation. This is why most recent MPE systems explicitly refer to and make use of the human auditory scene analysis principles. In human hearing, the perceptual organization of spectral components has been found to depend on certain acoustic cues. Two components may be associated to a same source by their closeness in time or frequency, harmonic concordance, synchronous changes in the frequency or amplitude of the components, or spatial proximity in the case of multisensor input [1].

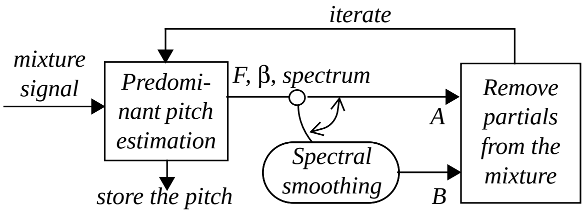

Fig. 1. Experimental framework: system which can be switched between two modes. (*A*) Straightforward iterative approach. (*B*) Spectral-smoothness based model.

The purpose of this paper is to propose a new efficient mechanism in computational MPE and auditory organization. *Spectral smoothness* refers to the expectation that the spectral envelopes of real sound sources tend to be continuous. Bregman points out this principle in human hearing by mentioning that spectral smoothness promotes integration of frequency partials to a same source and a single higher intensity partial is more likely to be perceived as an independent sound [1, p.232]. However, smoothness has not traditionally been included among the auditory organization cues. This paper presents evidence for the importance of spectral smoothness both in human and computational MPE. Also, three different algorithms are described that implement this principle.

Validation experiments were performed using an experimental model, where the spectral smoothness was either utilized in different ways, or was completely ignored. Acoustic database comprised sung vowels and the whole pitch range of 26 musical instruments. MPE was performed in a single time frame for random pitch and sound source combinations, number of simultaneous sounds ranging from one to six. Including the spectral smoothness principle in calculations made significant improvement in simulations. For example, the pitch error rate in random four-voice mixtures dropped from 18 % to 8.1 %, and in musical four-voice mixtures from 25 % to 12 %. As a result, MPE could be performed quite accurately at a wide pitch range and without *a priori* knowledge of the sound sources involved.

2. EXPERIMENTAL FRAMEWORK

Figure 1 shows the overview of the system which acts as an experimental framework in this paper. The system can be switched between two modes. The straightforward iterative MPE model, denoted by branch *A*, has been described earlier in [11]. It consists of two main parts that are applied in an iterative succession. The first part, predominant pitch estimation, finds the pitch of the most prominent sound in the interference of other harmonic and noisy sounds. As an output, it gives the fundamental frequency *F*, inharmonicity factor β, and the precise frequencies and amplitudes of the harmonic partials of the sound. In the second part, the spectrum of the detected sound is linearly subtracted from the mixture. These are then repeated for the residual signal.

A spectral-smoothness based model is obtained by locating an additional module between the estimation and subtraction stages. This is denoted by branch *B* in Fig. 1. The aim of the spectral smoothing algorithm is to use the pitch information to produce a more appropriate estimate for the spectrum of a separated sound before it is subtracted from the mixture. The need for such a module is strongly motivated by two observations. The predominant pitch estimation algorithm is capable of finding one of the correct pitches with 99 % certainty even in six-voice polyphonies [11]. However, the probability of error increases rapidly in the course of iteration. This indicates that the initial estimate of a sound spectrum as given by predominant pitch algorithm is not accurate enough to remove it correctly from the mixture.

3. DIAGNOSIS OF THE STRAIGHTFORWARD ITERATIVE SYSTEM

Simulations were run to analyze the behaviour of the straightforward iterative estimation and separation approach, i.e., the branch *A* in Figure 1. Random mixtures of *N* sounds were generated by first allotting an instrument and then a random note from its whole playing range, however, restricting the pitch over five octaves between 65 Hz and 2100 Hz. The desired number of sounds was allotted, and them mixed with equal mean square levels. The iterative process was then evoked and requested to extract *N* pitches from the acoustic mixture signal. As a general impression, the presented iterative approach works rather reliably.

However, an important observation is immediately made when the distribution of the remaining errors is analyzed. Figure 2 shows the errors as a function of the musical intervals that occur in the erroneously transcribed sound mixtures. It appears that the error rate is strongly correlated with certain pitch relations. More exactly, the straightforward estimation and subtraction approach is likely to fail in cases where the fundamental frequencies of simultaneous sounds are in simple rational number relations, also called *harmonic* relations. These are indicated over the corresponding bars in Fig. 2.

3.1 Coinciding sinusoidal partials

It turned out that coinciding frequency partials from different sounds make the algorithm fail. If sounds are in a harmonic relation to each other, a lot of partials coincide, i.e., share the same frequency. When the firstly detected sound is removed, the coinciding harmonics of remaining sounds are also removed in the subtraction procedure. In some cases, and particularly after several iterations, a remaining sound gets too corrupted to be correctly analyzed in the coming iterations.

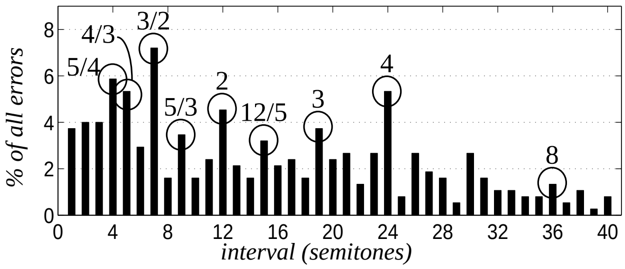

Fig. 2. Distribution of the pitch estimation errors as a function of the musical intervals that occur in the erroneously transcribed sound mixtures.

When two sinusoidal partials with amplitudes a_1 and a_2 and phase difference θ_Δ coincide in frequency, the amplitude of the resulting sinusoid can be calculated as

$$a_s = \left| a_1 + a_2 e^{i\theta_\Delta} \right| . \qquad (1)$$

If the two amplitudes are roughly equivalent, the partials may either amplify or cancel each other, depending on θ_Δ. However, if one of the amplitudes is significantly larger than the other, as is usually the case, a_s approaches the maximum of the two.

3.2 Fundamental frequency relations

The condition that a harmonic partial *h* of a sound *S* coincides a harmonic *j* of another sound *R* can be written as $hF_S = jF_R$, where F_S and F_R are the fundamental frequencies, and the two sides represent the frequencies of the partials. When the common factors of integers *h* and *j* are reduced, this yields

$$F_R = \frac{m}{n} F_S , \qquad (2)$$

where $(m, n) \geq 1$ are integer numbers. This implies that partials of two sounds can coincide only if the fundamental frequencies of the two sounds are in rational number relations. Furthermore, when the fundamental frequencies of two sounds are in the above relation, then every m^{th} harmonic *mk* of the sound *S* coincides every n^{th} harmonic *nk* of the sound *R* at their common frequency bands, where integer $k \geq 1$. This is evident since hF_S equals jF_R for each pair *h*=*mk* and *j*=*nk*, when Eq. (2) holds.

An important principle governing music is paying attention to the pitch relations, intervals, of simultaneously played notes. Simple harmonic relations satistying Eq. (2) are favoured over dissonant ones. Although western music arranges notes to a quantized logarithmic scale, it can surprisingly well produce the different harmonic intervals that can be derived by substituting small integers to Eq. (2) [3]. Because harmonic relations are so common in music, these “worst cases” must be handled well in general. Also, this explains why MPE is particularly difficult in music.

4. SOLUTION AND ITS ARGUMENTATION

The difficulties caused by harmonic pitch relations can be classified into two categories. First, the partials of an other sound may be erroneously removed along with the one that is being actually separated. This causes undetections. Second, two or more fundamental frequencies in certain relations may make a non-existent “ghost” sound appear, for example the root pitch of a chord. This causes insertion errors, i.e., extraneous pitch detections.

There is a solution to these problems that is both intuitive,

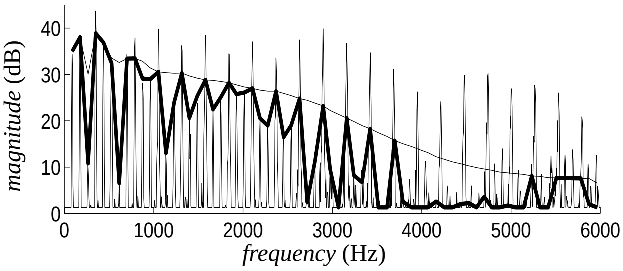

Fig. 3. Illustration of the spectral smoothness principle. Logarithmic magnitude spectrum containing two sounds, lower of which has been detected first. The spectrum has been high-pass liftered to remove spectral envelope.

efficient, and psychoacoustically valid: the spectra of the detected sounds must be smoothed before subtracting them from the mixture. Consider the logarithmic magnitude spectrum of a two-sound mixture in Fig. 3. The harmonic partials of the higher-pitched sound coincide every third harmonic of the lower-pitched sound, which has been detected first. As predicted by Eq. (1), the coinciding partials of the detected sound tend to have higher magnitudes than the other ones. However, when the sound spectrum is smoothed (thin slowly decreasing horizontal curve in Fig 3), these partials rise above the smooth spectrum, and thus remain in the residual after subtraction. In this way, the other sound is not removed with the detected one. When properly applied, the same mechanism can be used to treat ghost sounds, too.

4.1 Psychoacoustic knowledge applied

The design of the smoothing operation is not as simple as it seems to be at the first glance. As a matter of fact, simply smoothing the amplitude envelope (thin horizontal curve in Fig 3) before subtraction from the mixture does *not* work in the sense that it would reduce the pitch error rate in simulations.

Spectral smoothing in the human auditory system does not take the form of lowpass filtering. Instead, a nonlinear mechanism cuts off single higher amplitude partials. In following, a brief description of the human auditory processing is made in order to reveal the exact mechanism of an appropriate smoothing process.

Meddis and Hewitt have proposed a computer model of human auditory periphery which aims at reproducing a widest range of phenomena in human pitch perception [12]. The algorithm consists of four main steps. First, the input signal is passed through a bank of bandpass filters. At each band, the signal is halfwave rectified and lowpass filtered to extract the amplitude envelope of the bandpassed signal. Periodicity in the resulting signal is detected by calculating autocorrelation function estimates within channels. In the final phase, the estimates are linearly summed across channels to get a summary autocorrelation function, the maximum value of which points out the global pitch.

Amplitude envelope calculation within channels performs implicit spectral smoothing. When a harmonic sound is considered, each two neighbouring harmonic partials cause amplitude *beating*, i.e., alternatingly amplify and cancel each other at the fundamental frequency rate. However, the magnitude of the beating caused by each two sinusoidal partials is determined by the smaller of their amplitudes. When the spectrum of a harmonic sound is considered, this “minimum amplitude” property filters out single higher amplitude harmonic partials.

4.2 Three smoothing algorithms

A computer implementation of the implicit smoothing in the human auditory system can be isolated to a separate module. The algorithm simply goes through the harmonic partials of a sound and replaces the amplitude a_h of partial h with the minimum of the amplitudes of the partial and its neighbour

$$a_h \leftarrow min(a_h, a_{h+1}) . \qquad (3)$$

Interestingly, performing this simple operation in the spectral smoothing module of Fig. 1 corrects about 30 % of the errors of the straightforward iterative model. For example, the error rate in random four-voice mixtures reduces from 18 % to 12 %.

A still more efficient algorithm can be designed by focusing on the role of the smoothing algorithm. It is: to cut off single clearly higher amplitude partials. Equation (3) surely does that, but bases the estimate on two values only. The robustness of the method can be improved by imitating the calculations of the human auditory system at bandlimited frequency channels.

The second algorithm first calculates moving average over the amplitudes of the harmonic partials. An octave wide Hamming window is centered at each harmonic partial h, and a weighted mean m_h of the amplitudes of the partials in the window is calculated. This is the smooth spectrum illustrated by a thin horizontal curve in Fig. 3. The original amplitude value a_h is then replaced with the minimum of the original and the averaged amplitude

$$a_h \leftarrow min(a_h, m_h) . \qquad (4)$$

These values are illustrated by a thick horizontal curve in Fig. 3. This straightforward algorithm is already almost as good as could be designed. For example, for random four-voice mixtures, the average pitch error rate dropped from 18 % to 8.9 %.

A final slight improvement to the method can be made by utilizing the statistical dependency of every m^{th} harmonic partials, as explained in Sec. 3.2. The third algorithm applies a multistage filter which consists of the following steps. First, the numbers {..., h–1, h, h+1, h+2...} of the harmonic partials around harmonic h are collected from an octave wide window. Next, the surrounding partials are classified into groups, where all the harmonics that share a common divisor are put to a same group. Third, estimates for harmonic h are calculated inside groups in the same manner as in the second algorithm. In the last step, the estimates of different groups are averaged, weighting each group according to its mean distance from harmonic h.

The other problem category, that of ghost sounds, was solved by noticing that that the likelihood of a predominant pitch should be re-estimated after the new smooth spectrum is calculated. An example case clarifies why an erroneous sound may arise as a joint effect of the others and how the problem can be solved. If two harmonic sounds are played with fundamental frequencies 2F and 3F, the spectra of these sounds match every second and every third harmonics of a non-existent sound with fundamental frequency F, which is erroneously credited for all the observed partials, and thus appears as a ghost sound. However, if the harmonic amplitudes of the ghost sound are smoothed and its likelihood is re-estimated, the irregularity of the spectrum decreases the level of the smooth spectrum, and the likelihood remains low.

Table 1: Pitch error rates using different smoothing algorithms.

Applied smoothing algorithm	Random mixtures, four voices	Musical mixtures, four voices
—None—	18 %	25 %
SMOOTH	18 %	24 %
MIN (1st)	12 %	17 %
SMOOTH+MIN (2nd)	8.9 %	13 %
STAT+MIN (3rd)	8.1 %	12 %

5. SIMULATIONS RESULTS

A lot of simulations was run to verify the importance of the proposed spectral smoothness principle and to compare the described three algorithms. Table 1 gives the pitch error rates using different spectral smoothing algorithms. Algorithms are listed top–down in the order they were introduced in this paper. The first row gives the results using the straightforward iterative estimation and separation approach, with no smoothing. Label SMOOTH refers to simple smoothing of the amplitude envelope, which is of no help, as mentioned in Sec. 4.1. MIN refers to the minimum-among-neighbours algorithm implemented by Eq. (3). SMOOTH+MIN is the second algorithm, given by Eq. (4). STAT+MIN is the third algorithm utilizing statistical dependencies between the partials.

Random mixtures were generated in the way described in Sec. 3. In musical mixtures, different pitch relations were favoured according to a statistical profile discovered by Krumhansl in classical western music [13, p.68]. In all simulations, pitch estimation took place in a single 190 ms time frame 100 ms after the onsets of the sounds. A correct pitch was defined to deviate less than half a semitone (±3 %) from the correct value.

As the most important observation, spectral smoothing makes remarkable improvement to MPE accuracy. The third algorithm is slightly but consistently the best, but also by far the most complicated among the three. The second algorithm, while being very simple to implement, already achieves almost same performance.

Figure 4 shows multipitch estimation results in different polyphonies using the STAT+MIN algorithm. The bars represent the overall error rates as a function of the polyphony, where e.g. error rate for random four-voice polyphonies is 8.1 % on average. The different shades of grey in each bar indicate the error cumulation in the iteration, errors occurred in the first iteration at the bottom.

The system works reliably and exhibits graceful degradation in increasing polyphony, with no abrupt breakdown at any point. As predicted by the analysis in Sec. 3.2, musical mixtures were generally more difficult to resolve. However, the difference is not very big, indicating that the spectral smoothing works well.

6. CONCLUSIONS

Spectral smoothness principle was proposed as an efficient new mechanism in MPE and sound separation. Introduction of this idea corrected approximately half of the errors occurring in an otherwise identical system which did not use the smoothness principle. As a result, MPE could be performed quite accurately at a wide pitch range and without *a priori* knowledge of the sound sources involved. The underlying assumption that the spectral envelopes of natural sounds are rather continuous seems to hold, since the smoothing operation can be done without noticeable loss of information from the MPE viewpoint.

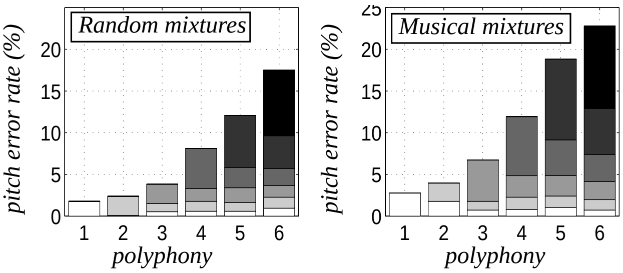

Fig. 4. Pitch error rates for multipitch estimation in different polyphonies. Bars represent the overall error rates, and the different shades of gray the error cumulation in iteration.

7. REFERENCES

[1] Bregman, A. S.(1990). "Auditory Scene Analysis," MIT Press.

[2] Rabiner, L. R., Cheng, M. J., Rosenberg, A. E., and McGonegal C. A. (1976). "A Comparative Performance Study of Several Pitch Detection Algorithms," IEEE Trans. Acoust., Speech, and Signal Processing, Vol.ASSP-24, No.5, 399–418.

[3] Klapuri, A. P. (1998). "Automatic Transcription of Music," MSc thesis, Tampere University of Technology, 1998.

[4] Martin, K. D. (1996). "Automatic Transcription of Simple Polyphonic Music: Robust Front End Processing", Massachusetts Institute of Technology Media Laboratory Perceptual Computing Section Technical Report No. 399.

[5] Kashino, K., Nakadai, K., Kinoshita, T., and Tanaka, H. (1995). "Organization of Hierarchical Perceptual Sounds: Music Scene Analysis with Autonomous Processing Modules and a Quantitative Information Integration Mechanism," Proc. International Joint Conf. on Artificial Intelligence, Montréal.

[6] Goto, M. (2000). "A robust predominant-F0 estimation method for real-time detection of melody and bass lines in CD recordings," Proc. IEEE International Conf. on Acoust., Speech, and Signal Processing, Istanbul, Turkey.

[7] Brown, G. J., and Cooke, M. P. (1994). "Perceptual grouping of musical sounds: A computational model," J. of New Music Research 23, 107–132.

[8] Godsmark, D., and Brown, G. J. (1999). "A blackboard architecture for computational auditory scene analysis," Speech Communication 27, 351–366.

[9] de Cheveigné, A., and Kawahara, H. (1999). "Multiple period estimation and pitch perception model," Speech Communication 27, 175–185.

[10] Sethares, W. A., and Staley, T. W. (1999). "Periodicity Transforms," IEEE Trans. Signal Processing, Vol. 47, No. 11.

[11] Klapuri, A. P., Virtanen T. O., and Holm, J.–M. (2000). "Robust multipitch estimation for the analysis and manipulation of polyphonic musical signals". In Proc. COST-G6 Conference on Digital Audio Effects, Verona, Italy.

[12] Meddis, R., and Hewitt, M. J. (1991). "Virtual pitch and phase sensitivity of a computer model of the auditory periphery. I: Pitch identification," J. Acoust. Soc. Am. 89 (6).

[13] Krumhansl, C. L. (1990). "Cognitive Foundations of Musical Pitch," Oxford University Press, New York.

Publication 4

A. P. Klapuri and J. T. Astola, “Efficient calculation of a physiologically-motivated representation for sound,” In *Proc. 14th IEEE International Conference on Digital Signal Processing*, Santorini, Greece, 2002.

EFFICIENT CALCULATION OF A PHYSIOLOGICALLY-MOTIVATED REPRESENTATION FOR SOUND

Anssi P. Klapuri and Jaakko T. Astola

Tampere University of Technology, P.O.Box 553, FIN-33101 Tampere, Finland
{klap,jta}@cs.tut.fi

Abstract: An algorithm is proposed which calculates a computationally efficient approximation of a certain physiologically-motivated representation for sound, called the summary autocorrelation function. This representation has been found very useful in several tasks, such as sound separation, multiple period estimation, and computational auditory scene analysis. However, it has been computationally too complex for most practical applications. The relatively fast algorithm described here proposes only an approximation of the summary autocorrelation function, but the achieved precision is likely to be good enough for most applications.

1. INTRODUCTION

The human auditory system is amazingly efficient in analyzing complex acoustic environments. It enables us to perceive and recognize simultaneously occurring sounds almost as easily as if the sounds would have been presented separately.

In performing analysis of acoustic signals, it is not only the algorithms that are important, but also the data representations. Analysis can be viewed as a hierarchy of representations from the acoustic signal up to a conscious percept [1]. While the latter usually cannot be directly deduced from the acoustic input, intermediate (mid-level) representations between these two are indispensable. Whereas we know rather little about the exact mechanisms of the brain, there is much wider consensus about the mechanisms of the physiological and more peripheral part of hearing. Moreover, precise *auditory models* exist which are able to calculate certain fundamental mid-level representations of hearing, such as the signal in the auditory nerve [2].

Correlogram has been widely accepted as being among the most generic and psychoacoustically valid mid-level representations. It models the physiology of hearing plus some psychoacoustic mechanisms, and is calculated as follows [3,1]:

1. Input signal is passed though a bank of bandpass filters which represent the frequency selectivity of the inner ear.
2. Signal at each frequency channel is half-wave rectified and lowpass filtered.
3. Periodicity estimation within channels is done by calculating short-time autocorrelation functions (ACF).
4. Periodicity estimates are aggregated across channels to obtain *summary autocorrelation function* (SACF) defined

$$s_t(\tau) = \sum_c r_{t,c}(\tau) \qquad (1)$$

Where $r_{t,c}(\tau)$ is the autocorrelation function in time frame *t* at frequency channel *c*.

The above calculations produce a three-dimensional *volume* with dimensions (i) time, (ii) frequency, and (iii) ACF lag. While the correlogram has proved very generic and efficient mid-level representation for audio analysis, it is easy to see that it is computationally very complex and data intensive, since the number of frequency channels in different models varies between 40 and 80. Data intensiveness is easily solved, since most analyses can be performed using only two marginal functions: rough spectral envelope, it is, $r_{t,c}(0)$, and the summary autocorrelation function. Rough spectral envelope forms the basis for sound source recognition and speech recognition, and several efficient methods exist to calculate it. However, SACF remains a computational nightmare, although it has been found to be very valuable in several tasks, such as sound separation, multiple period estimation, and computational auditory scene analysis [5,6,1]. In particular, SACF-based models have been shown to reproduce a wide range of characteristics of the human pitch perception [4].

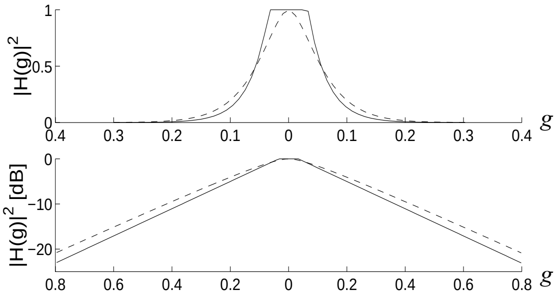

Figure 1. Roex (dashed line) and Flex (solid line) filter responses as a function of *g*, $g = |f - f_c| / f_c$.

In this paper, we propose a computatinally efficient method to calculate an approximation of SACF in the frequency domain. Whereas only an approximation is achieved, it should be noted that the SACF is not used for sound synthesis but to indicate perceptually relevant data, such as the highest SACF peak indicating the pitch period. For such analysis applications, the precision of the approximation is by far sufficient.

1.1. Auditory filterbank

The auditory frequency analyzer is usually modeled as a bank of overlapping, linear, bandpass filters. The equivalent rectangular bandwidths (ERB) of the auditory filters have been measured through listening tests, and can be calculated as [7]:

$$ERB(f_c) = 24.7[1 + 4.37 f_c / 1000] . \qquad (2)$$

where the center frequency and bandwidth are in Herz units.

Important characteristics of the auditory filterbank are that the auditory filters are approximately uniformly distributed on a logarithmic frequency scale, the bandwidths are according to Eq. (2), and that the number of filters is large enough to make the passbands of adjacent filters overlap much.

The exact shape of the response of individual filters can be modeled with Rounded-exponential, Roex(*p*), filters [7]:

$$|H(g)|^2 = (1 + pg)e^{-pg} \qquad (3)$$

where $g = |f - f_c| / f_c$ is the relative distance from the center frequency, and *p* is a parameter determining the bandwidth of the filter. Roex response is illustrated with dashed line in Fig. 1.

2. FAST APPROXIMATION

A starting point for performing the above-described correlogram calculations efficiently is to analyze the four calculation phases in the frequency domain. This inevitably leads to frame-based processing. However, this is not a serious problem, since the autocorrelation function calculations involved in the conventional SACF calculations are in practice always performed on a frame-by-frame basis to allow FFT-based ACF computations. We use $X(k)$ to denote the complex-valued discrete Fourier transform of a K-length frame of the acoustic input signal $x(n)$. The discrete frequency variable $k \in [-K/2, K/2]$.

The calculations are first presented straigthforwardly in the frequency domain in a manner that is not faster than the conventional calculations, and then the fast way is shown.

Phase (1): *band-pass filtering*. Filtering $x(n)$ with a linear bandpass filter $h_c(n)$ is equivalent to multiplying $X(k)$ with the frequency response of the filter, $X_c(k) = H_c(k)X(k)$.

Phase (2): *halfwave rectification (HWR) and lowpass filtering*. The non-linear HWR operation is an essentially important part of the correlogram model. For a narrowband signal $x_c(n)$ centered at f_c, HWR generates spectral components to bands centered on zero frequency, on f_c, and on integer multiples of f_c without upper limit (see the standard analyses in [8]). It can be shown that the desirable properties of SACF are due to the frequency bands centered on zero frequency and on f_c. The higher frequency components, here called the *harmonic distortion spectrum*, are unnecessary and cause inevitable aliasing in discrete signals. Therefore, we use the following techique to calculate the spectrum $W_c(k)$ of the rectified signal at channel c, so that aliasing and the distortion spectrum are rejected. As shown in [8], the spectral region generated by HWR to the band around zero frequency is a scaled version of the spectrum generated by squaring the signal, *if* the harmonic distortion spectrum is ignored, which causes an error smaller than 3 % around center frequency. The spectrum around f_c in the output of HWR, in turn, is that of the input narrowband signal $X_c(k)$. Thus we use a model for the spectrum of the rectified signal at channel c:

$$W_c(k) = V_c(k)/(4\sigma_x) + X_c(k). \tag{4}$$

where $V_c(k)$ is the spectrum of a squared time domain signal, lowpass filtered to pass only the band around zero frequency (up to f_c), and σ_x is the standard deviation of the signal at channel c. It is easy to verify this approximation of the HWR.

Squaring in time domain is equivalent to convolution in frequency domain, thus we write

$$V_c(\delta) = \sum_{k=-K/2+\delta}^{K/2-\delta} [H_c(k)X(k)H_c(k+\delta)X^*(k+\delta)] \tag{5}$$

for $\delta < f_c$, and $V_c(\delta) = 0$ otherwise. Vector $X^*(k)$ is the complex conjugate of $X(k)$.

Phase (3): *within-channel periodicity extraction.* Autocorrelation function calculation in time domain is equivalent to calculating the square of the absolute value of the Fourier transform of the signal. Thus the Fourier transform $R_c(k)$ of the autocorrelation function $r_c(\tau)$ of the rectified signal at frequency channel c is obtained as

$$R_c(k) = |V_c(k)/(4\sigma_x) + X_c(k)|^2. \tag{6}$$

Phase (4): *across-channel aggregation of periodicity estimates.* Summary autocorrelation function $s(\tau)$ is calculated in time domain according to Eq. (1). Since Fourier transform and its inverse are linear operations, we can sum $R_c(k)$ already in the frequency domain to obtain the Fourier transform of $s(\tau)$,

$$S(k) = \sum\nolimits_c R_c(k) \tag{7}$$

and then perform a single inverse Fourier transform to obtain the summary autocorrelation function $s(\tau)$.

2.1. Observation which leads to fast implementation

The presented frequency domain calculations as such are not essentially faster than the conventional ones. They include one computationally very intensive operation: spectral convolution to obtain the spectrum of rectified signal, $V_c(k)$.

Core idea behind the fast approximation is the observation that very efficient iterative update rules exist to calculate $V_c(k)$ from $V_{c-1}(k)$. Thus we only need to initialize $V_c(k)$ for c=1, and then iteratively calculate $V_c(k)$ for all channels c using the update rules to be described below. Spectrum of SACF, $S(k)$, can then be calculated straightforwardly from Eq. (7) for this particular value of k, and computations then proceed to next k.

Such update rules exist for a certain family of bandpass filters, flatted-exponential filters, defined below.

2.2. Bank of flatted-exponential bandpass filters

The center frequencies and ERB-bandwidths of the bank of filters $H_c(k)$ to be considered in the rest of this paper are as follows. Channel index c goes from 1 to $K/2$, where K is the size of the time frame. The center frequency of the filter at channel c is $f_s \times c/K$, thus there is one filter corresponding to each positive frequency sample k of $X(k)$. ERB-bandwidths for channel c are obtained from Eq. (2) and are in frequency sample units

$$w_c = \beta_1 + \beta_2 c, \tag{8}$$

where $\beta_1 = 24.7 f_s/K$, β_2=4.37/1000, and w_c is real-valued.

Because the desired distribution of frequency bands is not linear, the different channels in the sum of Eq. (7) can be weighted to correspond to an arbitrary distribution of channels, e.g. weights $1/c$ corresponding to a logarithimic distribution.

We define Flatted-exponential, Flex(p), filter to have unity response around the center frequency, followed by an exponentially decaying magnitude response further away from the center frequency. The response is illustrated in Fig. 1. The slope of attenuation is the same as the in Roex(p) filter:

$$H_c(g) = \begin{cases} 1 & g \le g_0 \\ \exp[-p(g-g_0)/2] & g \ge g_0 \end{cases} \tag{9}$$

where g can now be written as $g = |k-c|/c$.

The parameter g_0, i.e., the half-width of the flatted top can be solved by requiring that the ERB-bandwidths of Flex(p) and Roex(p) filters must be equal for a given parameter p. Writing the integrals over the squares of the two responses to be equal, we can solve $g_0 = 1/p$, where $p_c = 4c/w_c$ is a function of c, and Eq. (9) can now be written as

$$H_c(k) = \begin{cases} 1 & |k-c| \le w_c/4 \\ \exp[-2|k-c|/w_c + 1/2] & |k-c| \ge w_c/4 \end{cases} \tag{10}$$

2.3. Convolved response

The spectrum $V_c(\delta)$ at channel c can be calculated using convolution, as shown in Eq. (5). We denote the terms in Eq. (5)

$$Z_\delta(k) = X(k)X^*(k+\delta), \tag{11}$$

and the *convolved response* at channel c

$$J_{c,\delta}(k) = H_c(k)H_c(k+\delta). \tag{12}$$

Substituting Eqs. (11) and (12) to Eq. (5) and observing that the spectrum is conjugate symmetric for real signals, i.e., $X(-k) = X^*(k)$, we can limit the sum to positive frequencies and write Eq. (5) as

$$V_c(\delta) = 2\sum\nolimits_{k=0}^{K/2-\delta} [J_{c,\delta}(k)Z_\delta(k)]. \tag{13}$$

The term $Z_\delta(k)$ is common to all frequency channels. However, the term $J_{c,\delta}(k)$ varies for each channel. Conjugate symmetry applies to $V_c(\delta)$, too, thus we only need to calculate for $\delta \ge 0$. Based on Eq. (13), we can also assume $k \ge 0$.

The convolved response $J_{c,\delta}(k)$ can be of two different types, depending if the flatted tops of the two Flex responses overlap or not, as shown in Fig. 2. Type is I if $\delta \le 2\lfloor w_c/4 \rfloor$, and type II otherwise. Both are of the same general form, consisting of five parts, denoted *A*, *B*, *C*, *D*, and *E* in Fig. 2. The

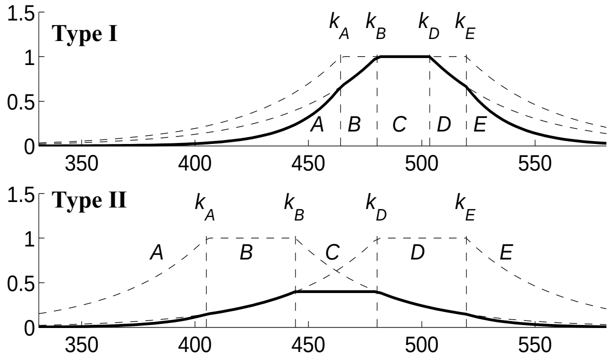

Figure 2. The two different types of the convolved responses.

convolved response is thus defined piecewise as

$$J_{c,\delta}(k) = \begin{cases} J^A_{c,\delta}(k) & k \in [0, k_A] \\ J^B_{c,\delta}(k) & k \in [k_A + 1, k_B] \\ J^C_{c,\delta}(k) & k \in [k_B + 1, k_D - 1] \\ J^D_{c,\delta}(k) & k \in [k_D, k_E - 1] \\ J^E_{c,\delta}(k) & k \in [k_E, K/2] \end{cases} \tag{14}$$

The formula for each of the five parts of the convolved response $J_{c,\delta}(k)$ can be calculated by substituting the Flex(*p*) response from Eq. (10) to Eq. (12). The resulting expressions are the same for the two types, except for the part *C*, for which we denote *C*1 for type I and *C*2 for type II. The formulae are

$$J^{A,E}_{c,\delta}(k) = \exp(-2|2k + \delta - 2c|/w_c + 1) \tag{15}$$

$$J^B_{c,\delta}(k) = \exp[-2(c - k)/w_c + 1/2] \tag{16}$$

$$J^{C1}_{c,\delta}(k) = 1 \tag{17}$$

$$J^{C2}_{c,\delta}(k) = \exp(-2\delta/w_c + 1) \tag{18}$$

$$J^D_{c,\delta}(k) = \exp[-2(k + \delta - c)/w_c + 1/2] \tag{19}$$

And the discrete boundaries $k_A \le k_B \le k_C \le k_D$:

$$k_A = c - \delta - \lceil w_c/4 \rceil \tag{20}$$

$$k_{B1} = c - \lceil w_c/4 \rceil \tag{21}$$

$$k_{B2} = c - \delta + \lfloor w_c/4 \rfloor \tag{22}$$

$$k_{D1} = c - \delta + \lceil w_c/4 \rceil \tag{23}$$

$$k_{D2} = c - \lfloor w_c/4 \rfloor \tag{24}$$

$$k_E = c + \lceil w_c/4 \rceil \tag{25}$$

where the formulas for the limits k_B and k_D are different for the two response types, as can be seen.

The overall sum $V_c(\delta)$ in Eq. (13) can then be written as

$$V_c(\delta) = V^A_c(\delta) + V^B_c(\delta) + V^C_c(\delta) + V^D_c(\delta) + V^E_c(\delta) \tag{26}$$

where $V^B_c(\delta)$, for example, can be calculated as

$$V^B_c(\delta) = \sum\nolimits_{k = k_A + 1}^{k_B} [J^B_{c,\delta}(k) Z_\delta(k)] \,. \tag{27}$$

2.4. Update formulas to calculate $V_c(\delta)$ efficiently

Each part (*A,B,C,D,E*) of $V_c(\delta)$ can be calculated iteratively from $V_{c-1}(\delta)$, or from $V_{c+1}(\delta)$, *in constant time* (*O*(1)) with few simple operations. The update rules are very similar for different parts, thus it suffices to thoroughly explain one part, let it be part *B,* which is illustrated in Fig. 3.

For part *B*, the algorithm to calculate $V^B_c(\delta)$, according to Eq. (27), for *c*=1,...,*K*/2 and for fixed δ is:

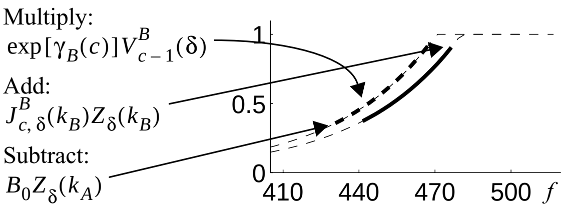

Figure 3. Update rules to calculate $V^B_c(\delta)$ from $V^B_{c-1}(\delta)$.

Initializations. Calculate $Z_\delta(k)$ according to Eq. (11), set $c \leftarrow 1$, $V^B_1(\delta) \leftarrow 0$, determine the type of the response, and calculate the corresponding boundary values k_A and k_B.
Iterative calculation of $V^B_c(\delta)$ for *c*=2,...,*K*/2:
Step 1. Store the current boundaries $k^o_A \leftarrow k_A$, $k^o_B \leftarrow k_B$.
Step 2. Set $c \leftarrow c + 1$ and calculate the corresponding new response type and boundaries.
Step 3. Set $V^B_c(\delta) \leftarrow \exp[\gamma_B(c)]V^B_{c-1}(\delta)$, where $\gamma_B(c) < 0$ is a real number to be determined below.
Step 4. If $k_B > k^o_B$

- Add new value $V^B_c(\delta) \leftarrow V^B_c(\delta) + J^B_{c,\delta}(k_B)Z_\delta(k_B)$
- Remember *c* which captured k_B: $\eta(k_B) \leftarrow c$

Step 5. If $k_A > k^o_A$

- Recall $c^o = \eta(k_A)$
- Subtract old value $V^B_c(\delta) \leftarrow V^B_c(\delta) - B_0 Z_\delta(k_A)$, where $B_0 = J^B_{c^o,\delta}(k_A)\exp[\gamma_B{}^\dagger(c) - \gamma_B{}^\dagger(c^o)]$ and $\gamma_B{}^\dagger(c)$ is cumulative sum over $\gamma_B(c)$.

Step 6. If $c < K/2$, return to Step 1.

Three update rules are embodied in the above algorithm, as illustrated in Fig. 3. The first rule is that always when *c* increases, the overall sum is multiplied with factor exp[γ_B(*c*)]<1, making old values exponentially leak out from the sum as they get further away from the center frequency. Secondly, if the boundary k_B changes, a new value is included to the sum $V^B_c(\delta)$ according to Eq. (27). In the third rule, a sample has to be removed from the sum in the case the boundary k_A changes. To do that, we have to know exactly the factors with which the sample has been multiplied during it belonging to the sum $V^B_c(\delta)$. This is calculated in the value B_0. The factor $J^B_{c^o,\delta}(k_A)$ was used when the sample was first included to the sum. The cumulative effect of repeated multiplying with values exp[γ_B(*c*)],...,exp[γ_B(*c*^{o})] can be efficiently solved using a cumulative sum $\gamma_B{}^\dagger(c)$, where $\gamma_B{}^\dagger(1) = \gamma_B(1)$, $\gamma_B{}^\dagger(2) = \gamma_B(1) + \gamma_B(2)$, and so on.

The values of γ_B(*c*) remain to be solved. It should be emphasized, that these constants are the same for all δ and in all time frames, and thus need to be initialized only once. Writing

$$J^B_{c,\delta}(k) = \exp[\gamma_B(c)]J^B_{c-1,\delta}(k) \tag{28}$$

reveals that a value that would lead to *exact* update rule does not exist, because w_c and thus the slope of attenuation changes as a function of *c*. However, an approximation is derived by starting with a value which realizes the slope of attenuation for the current w_c. Reading from Eq. (16):

$$\gamma_B'(c) = -2/w_c \,. \tag{29}$$

As a next step, we force the attenuation caused by successive multiplications to reach −3 dB level exactly at a same distance from *c* as in the ideal response $J_{c,\delta}(k)$. It is, we force the −3 dB bandwidth of the convolved response to be according to the ideal. The ideal −3 dB point can be found by writing $J^B_{c,\delta}(k_0) = 10^{-3/10}, k_0 < c$, from which k_0 is easily solved. Then we find a center frequency c_0 for which the boundary k_B is k_0, assuming response type I. Value c_0 is got from Eq. (21).

The *realized* attenuation at point k_0 through successive multiplications with $\gamma_B'(c)$ is now denoted $\exp(A_B)$. Value of A_B

can be calculated by integrating $\gamma_B'(c)$ from c_0 to c, yielding

$$A_B = -(2/\beta_2)\ln(w_c/w_{c_0})\,. \qquad (30)$$

Finally, the formula for $\gamma_B(c)$ which works out the desired −3 dB bandwidth is obtained by using a correction term for $\gamma_B'(c)$, where desired attenuation is divided with the realized:

$$\gamma_B(c) = [\log(10^{-3/10})/A_B](-2/w_c)\,. \qquad (31)$$

For part *A*, all is very similar. Only two update rules are applied, since no values drop from the sum. At each iteration

$$V_c^A(\delta) = \exp[\gamma_A(c)]V_{c-1}^A(\delta), \qquad (32)$$

where $\gamma_A(c)$ is simply $\gamma_A(c) = \gamma_B(c) + \gamma_B(c)$, since the slope of attenuation for part *A* is twice steeper than for part *B*. New values $J_{c,\delta}^A(k_A)Z_\delta(k_A)$ are added whenever k_A changes.

For part *D*, the calculations are analogous to part *B*, except that for numerical reasons it is essentially important to start from *c*=*K*/2, and iterately calculate $V_c^B(\delta)$ for decreasing values of *c*. The three update rules are analogous to part *B* and the values $\gamma_D(c)$ are determined through the same procedure.

For part *C*, exact update rules can be written for both types of the convolved response. However, since the level of the flatted top for type II decreases together with bandwidth w_c, it is numerically advantageous to perform the calculations for decreasing values of *c*, starting from *c*=*K*/2. The response type may change on the way.

Initializations. Set $c \leftarrow K/2$, determine type type of the response, calculate boundaries k_B and k_D, and initialize $J_{K/2,\delta}^C(k)$ according to Eqs. (17) and (18).

Iterative updating: If the response type is I, we only need to take in and drop out values when boundaries k_B or k_D change. However, if the type is II and the previous type was II, we set

$$V_c^C(\delta) \leftarrow \exp[\gamma_C(c)]V_{c+1}^C(\delta), \qquad (33)$$

where $\gamma_C(c)$ can be solved exactly by writing

$$J_{c,\delta}^{C2}(k) = \exp[\gamma_C(c)]J_{c+1,\delta}^{C2}(k) \qquad (34)$$

from which $\gamma_C(c) = (-2\delta\beta_2)/[w_c(w_c + \beta_2)]$.

If the response type is II and the previous type was I, we set

$$V_c^C(\delta) \leftarrow J_{c,\delta}^{C2}(k)V_{c+1}^C(\delta)\,. \qquad (35)$$

Adding new values and subtracting dropping values is relatively easy since the summing area is flat.

The overall response $V_c(\delta)$ is obtained by summing the different parts, as shown in Eq. (26).

2.5. Including the original spectrum

According to Eq. (4), the spectrum of the signal at channel *c*, $X_c(k)$ has to be added before calculating the ACF spectrum in Eq. (6). This is easier than it first seems. The spectra of $V_c(k)$ and $X_c(k)$ are non-overlapping, thus the two terms in Eq. (6) can be squared independently and then summed. In fact, the both terms can be squared and summed as late as in Eq. (7). It follows that in Eq. (7) we sum together $|X_c(k)|^2$ from all different bands. Since the channel density is very high, and channel distribution is usually designed so that the bands together sum to unity, we can use $|X(k)|^2$ in Eq. (7), and the spectra $X_c(k)$ do not need to be calculated at all.

3. PRECISION OF THE APPROXIMATION

Figure 4 illustrates the spectral densities of the ideal convolved response $J_{c,\delta}(k)$ (thick curve), the iteratively calculated response (thin curve close to the ideal), and the error between these two (thick broken curve). Due to the iterative calculations, bandwidth tends to be smaller than desired on the left side of the center frequency and vice versa. Error for part *C* is zero, and separate plateaus of error can be seen for the other four parts.

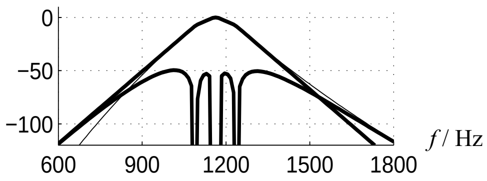

Figure 4. Spectral densities of the ideal convolved response (thick), realized response (thin) and the error (lower curve).

Signal-to-noise ratios between the ideal response $\overline{J_{c,\delta}}(k)$ and the approximation were calculated as

$$SNR = 10\log_{10}\left[\sum_{k=0}^{2c}\left|\overline{J_{c,\delta}}(k)\right|^2 \Big/ \sum_{k=0}^{2c}\left|E_{c,\delta}(k)\right|^2\right] \qquad (36)$$

where $E_{c,\delta}(k)$ is the difference between the ideal and the realized response. SNR values were found to be essentially independent of the center frequency and of the frame length, but to depend on δ, being 49 dB, 47 dB, 37 dB and 30 dB for δ values 0, $0.5w_c$, w_c, and $2w_c$, respectively. Actually the level of the error stays persistenty around −50 dB, but as the level of the ideal response gets smaller for type II according to Eq. (18), the level of the error in relation to the ideal gets higher.

Based on the above simulation results as such, it can be shown that the precision of the spectrum of the SACF is of the same order. This is because each sample $Z_\delta(k)$ in Eq. (13) coincides with different parts of the convolved responses of the frequency channels surrounding it. Thus each sample gets weighted with all parts of $J_{c,\delta}(k)$, resulting in same precision.

4. COMPUTATIONAL COMPLEXITY

The complexity of the conventional way of calculating the correlogram is $O\{N \times [KM + K\log(K)]\}$, where *N* is the number of frequency channels, $M \approx 15$ is the sum of the orders of the bandpass and lowpass filters applied, and *K*log(*K*) stands for the ACF calculations via FFT.

In the presented method, FFT and its inverse are $O[K \times \log(K)]$ complex operations. In bandpass filtering and rectifying the signal at all bands, we calculate the $O(K)$ complex iterative process for each value of δ. The complexity thus becomes $O\{K \times [\delta_{\max} + \log(K)]\}$. A practical and safe value for $\delta_{\max}$ is frequency sample corresponding to 1 kHz. For any reasonable selections for $\delta_{\max}$, the complexity $\delta_{\max} + \log(K)$ is significantly smaller than $N \times [M + \log(K)]$.

5. REFERENCES

[1] Ellis, D. P. W. “Prediction-driven computational auditory scene analysis,” PhD thesis, MIT, 1996.

[2] Moore (Ed.). “Hearing. Handbook of Perception and Cognition (2nd edition),” Academic Press Inc., 1995.

[3] Duda, R. O., Lyon, R. F., Slaney, M. “Correlograms and the separation of sounds,” Proc. IEEE Asilomar Conf. on Signals., Sys. & Computers, 1990.

[4] Meddis R., Hewitt M. J. “Virtual pitch and phase sensitivity of a computer model of the auditory periphery. I:Pitch identification”. *J. Acoust. Soc. Am.* 89 (6), June 1991.

[5] Wang, D. L., Brown, G.J. “Separation of speech from interfering sounds based on oscillatory correlation,” *IEEE Trans. Neural Networks*, Vol. 10, No. 3, May 1999.

[6] Klapuri, A. “Multipitch estimation and sound separation by the spectral smoothness principle,” Proc. IEEE International Conf. on Acoust., Speech and Signal Processing, 2001.

[7] Patterson, R. D., Moore B. C. J. “Auditory filters and excitation patterns as representations of frequency resolution,” In B. C. J. Moore (Ed.), *Frequency Selectivity in Hearing*, Academic Press, London, 1986.

[8] Davenport, W. B., Root, W. L. “An Introduction to the Theory of Random Signals and Noise,” IEEE Press, 1987.

Publication 5

A. P. Klapuri, “Multiple fundamental frequency estimation based on harmonicity and spectral smoothness,” *IEEE Trans. Speech and Audio Proc.*, 11(6), 804–816, 2003.

Multiple Fundamental Frequency Estimation Based on Harmonicity and Spectral Smoothness

Anssi P. Klapuri

***Abstract*—A new method for estimating the fundamental frequencies of concurrent musical sounds is described. The method is based on an iterative approach, where the fundamental frequency of the most prominent sound is estimated, the sound is subtracted from the mixture, and the process is repeated for the residual signal. For the estimation stage, an algorithm is proposed which utilizes the frequency relationships of simultaneous spectral components, without assuming ideal harmonicity. For the subtraction stage, the spectral smoothness principle is proposed as an efficient new mechanism in estimating the spectral envelopes of detected sounds. With these techniques, multiple fundamental frequency estimation can be performed quite accurately in a single time frame, without the use of long-term temporal features. The experimental data comprised recorded samples of 30 musical instruments from four different sources. Multiple fundamental frequency estimation was performed for random sound source and pitch combinations. Error rates for mixtures ranging from one to six simultaneous sounds were 1.8%, 3.9%, 6.3%, 9.9%, 14%, and 18%, respectively. In musical interval and chord identification tasks, the algorithm outperformed the average of ten trained musicians. The method works robustly in noise, and is able to handle sounds that exhibit inharmonicities. The inharmonicity factor and spectral envelope of each sound is estimated along with the fundamental frequency.**

***Index Terms*—Acoustic signal analysis, fundamental frequency estimation, music, music transcription, pitch perception.**

I. Introduction

PITCH perception plays an important part in human hearing and understanding of sounds. In an acoustic environment, human listeners are able to perceive the pitches of several simultaneous sounds and make efficient use of the pitch to acoustically separate a sound in a mixture [1]. Computational methods for multiple fundamental frequency (F0) estimation have received less attention, though many algorithms are available for estimating the F0 in single-voice speech signals [2]–[4]. It is generally admitted that these algorithms are not appropriate as such for the multiple-F0 case.

A sound has a certain pitch if it can be reliably matched by adjusting the frequency of a sine wave of arbitrary amplitude [5]. Pitch is a perceptual attribute of sounds. The corresponding physical term F0 is defined for periodic or nearly periodic sounds only. For these classes of sounds, F0 is closely related to pitch and is defined as the inverse of the period, i.e., the time shift for which the time-domain signal shows high correlation with itself. In cases where the fundamental period is ambigous, a candidate closest to the subjective pitch period is regarded as the correct F0.

Musical signals are natural candidates for the problem of multiple-F0 estimation, in the same way as speech signals are natural candidates for single-F0 estimation. Automatic transcription of music aims at extracting the pitches, onset times, and durations of the notes that constitute the piece. The first multiple-F0 algorithms were designed for the purpose of transcribing polyphonic music in which several sounds are playing simultaneously. These attempts date back to 1970s, when Moorer built a system for transcribing duets, i.e., two-voice compositions [6]. The work was continued by Chafe and his collegues [7]. Further advances were made by Maher [8]. However, the early systems suffered from severe limitations in regard to the pitch ranges and relationships of simultaneous sounds, and the polyphony was restricted to two concurrent sounds. Relaxation of these constraints was attempted by allowing some more errors to occur in the transcription [9], or by limitation to one carefully modeled instrument [10], [11].

More recent transcription systems have recruited psychoacoustically motivated analysis principles, used sophisticated processing architectures, and extended the application area to computational auditory scene analysis in general [12]. Kashino *et al.* integrated signal analysis with temporal and musical predictions by applying a Bayesian probability network [13]. Martin utilized musical rules in transcribing four-voice piano compositions [14]. Front-end processing in his system was performed using a log-lag correlogram model of the human auditory periphery, as described in [15]. Goto was the first to introduce a system which works reasonably accurately for real-world complex musical signals by finding the melody and bass lines in them [16].

Multiple-F0 estimation is closely related to auditory scene analysis: any algorithm that can find the F0 of a sound and not get confused by other co-occurring sounds is, in effect, doing auditory scene analysis [1, p. 240]. Because the human auditory system is very accurate in performing this task, imitation of its processing principles has become common and psychoacoustically inspired systems in general have been relatively successful. Brown and Cooke have built computational models of the human auditory processes and also addressed the auditory grouping and streaming of musical sounds according to common acoustic properties [17]. Godsmark and Brown proposed a blackboard architecture to integrate evidence from different auditory organization principles and demonstrated

Manuscript received November 29, 2001; revised April 16, 2003. This work was supported by the TISE Graduate School, Tampere University of Technology, the Foundation of Emil Aaltonen, Tekniikan edistämissäätiö, and the Nokia Foundation. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Sean A. Ramprashad.

The author is with Institute of Signal Processing, Tampere University of Technology, FIN-33720 Tampere, Finland (e-mail: klap@cs.tut.fi).

Digital Object Identifier 10.1109/TSA.2003.815516

that the model could segregate melodic lines from polyphonic music [18].

The unitary model of pitch perception proposed by Meddis and Hewitt has had a strong influence on F0 estimation research [19], [20]. Tolonen and Karjalainen have suggested a simplified version of the unitary pitch model and applied it to the multiple-F0 estimation of musical sounds [21]. In [22], de Cheveigné and Kawahara integrated the model with a concurrent vowel identification model of Meddis and Hewitt [23] and developed an approach where F0 estimation is followed by the cancellation of the detected sound and iterative estimation for the residual signal. A more straightforward version of this iterative approach was earlier proposed by de Cheveigné in [24].

The periodicity transform method proposed by Sethares and Staley in [25] bears a close resemblance to that of de Cheveigne in [24], although the former is purely mathematically formulated. A more dynamic approach to residue-driven processing has been taken by Nakatani and Okuno [26]. Their system was designed to segregate continuous streams of harmonic sounds, such as the voiced sections of two or three simultaneous speakers. Multiple agents were deployed to trace harmonic sounds in stereophonic input signals, the sounds were subtracted from the input signal, and the residual was used to update the parameters of each sound and to create new agents when new sounds were detected.

There are two basic problems that a multiple-F0 estimator has to solve in addition to those that are confronted with in single-F0 estimation. First, the calculated likelihoods (or weights) of different F0 candidates must not be too much affected by the presence of other, co-occurring sounds. To achieve this, multiple-F0 algorithms typically decompose incoming signals into smaller elements which are then selectively used to calculate the weight for each candidate. For example, some methods trace sinusoidal components and then group them into sound sources according to their individual attributes, such as harmonic relationships or synchronous changes in the components [7], [13], [16], [26], [27]. Other algorithms apply comb filtering in the time domain to select only the harmonically related components [22], [24], [25]. Several recent systems have employed auditory models which break an incoming sound into subchannel signals and perform periodicity analysis withing channels [18], [20], [22].

In the second place, even when a correct F0 has been detected, the next-highest weights are often assigned to half or twice of this correct F0 value. Thus, the effect of any detected F0 must be cancelled from harmonics and subharmonics before deciding the next most likely F0. Some algorithms perform this by manipulating the calculated F0 weights directly [21]. Other methods estimate the spectrum of each detected sound and then subtract it from the mixture in an iterative fashion [24], [25], or process as a joint estimation and cancellation pursuit [24], [26]. The latter scheme is similar to the analysis-by-synthesis techniques in parametric coding, where for example sinusoidal components are detected, modeled, and subtracted from the input in order to minimize the residual signal [28].

The aim of this paper is to propose a multiple-F0 analysis method that operates at the level of a single time frame and is applicable for sound sources of diverse kinds. Automatic transcription of music is seen as an important application area, implying a wide pitch range, varying tone colors, and a particular need for robustness in the presence of other harmonic and noisy sounds.

An overview of the proposed system is illustrated in Fig. 1. The method operates iteratively by estimating and removing the most prominent F0 from the mixture signal. The term *predominant-F0 estimation* refers to a crucial stage where the F0 of the most prominent sound is estimated in the presence of other harmonic and noisy sounds. To achieve this, the harmonic frequency relationships of simultaneous spectral components are used to group them to sound sources. An algorithm is proposed which is able to handle inharmonic sounds. These are sounds for which the frequencies of the overtone partials (harmonics) are not in exact integer ratios. In a subsequent stage, the spectrum of the detected sound is estimated and subtracted from the mixture. This stage utilizes the spectral smoothness principle, which refers to the expectation that the spectral envelopes of real sounds tend to be slowly varying as a function of frequency. In other words, the amplitude of a harmonic partial is usually close to the amplitudes of the nearby partials of the same sound. The estimation and subtraction steps are then repeated for the residual signal. A review and discussion of the earlier iterative approaches to multiple-F0 estimation can be found in [22], [24]. Psychoacoustic evidence in favor of the iterative approach can be found in [1, p. 240, 244], [5].

The motivation for this work is in practical engineering applications, although psychoacoustics is seen as an essential base of the analysis principles. The proposed algorithm is able to resolve at least a couple of the most prominent F0s, even in rich polyphonies. Reliable estimation can be carried out in cases where the signal has been corrupted by high levels of additive noise or where wide frequency bands are missing. Non-ideal sounds that exhibit inharmonicities can be handled. The applications thus facilitated comprise transcription tools for musicians, transmission and storage of music in a compact form, and new ways of searching musical information.

The paper is organized as follows. Section II will describe the different elements of the algorithm presented in Fig. 1. These include preprocessing, the harmonicity principle used, the smoothing of detected sounds, and estimation of the number of concurrent sounds. Section III will describe experimental results and will compare these with the performance of two reference methods and human listeners. Finally, Section IV will summarize the main conclusions and will discuss future work.

II. Proposed Multiple-F0 Estimation Method

This section will look at all the necessary elements required for the multiple-F0 estimation task and as illustrated in Fig. 1. To begin, Section II-A will describe the preprocessing stage which is necessary to achieve robustness in additive noise and to handle sounds with uneven spectral shapes. Next, the main principle behind using harmonic relationships is discussed in Section II-B. Section II-C will describe the smoothing algorithm which is needed to subtract each detected sound from the mixture so that the remaining sounds are not corrupted. The last subsection will propose a mechanism to control the stopping of the iterative estimation and cancellation process.

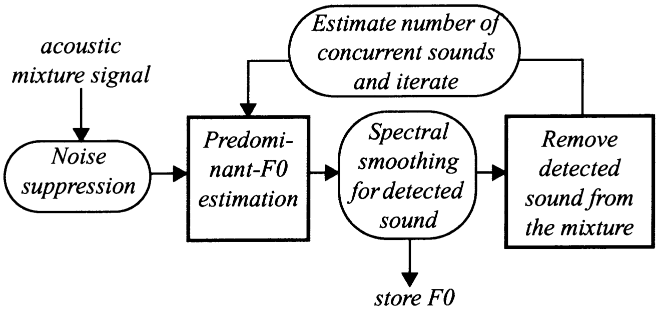
acoustic mixture signal
Noise suppression
Predominant-F0 estimation
Estimate number of concurrent sounds and iterate
Spectral smoothing for detected sound
Remove detected sound from the mixture
store F0

Fig. 1. Overview of the proposed multiple-F0 estimation method.

A. Preprocessing

All calculations in the proposed system take place in the frequency domain. A discrete Fourier transform is calculated for a Hamming-windowed frame of an acoustic input signal, sampled at 44.1 kHz rate and quantized to 16-bit precision. Frame lengths of 93 ms and 190 ms were used in simulations. These may seem long from the speech processing point of view, but are actually not very long for musical chord identification tasks. In such tasks, the pitch range is wide, mixtures of low sounds produce very dense sets of frequency partials, and F0 precision of 3% is required to distinguish adjacent notes (see Appendix).

Preprocessing the spectrum before the actual multiple-F0 analysis is an important factor in the performance of the system. It provides robustness in additive noise and ensures that sounds with varying spectral shapes can be handled. The signal model assumed by the proposed system is

$$X(k) = H(k)S(k) + N(k) \tag{1}$$

where $X(k)$ is the discrete power spectrum of an incoming acoustic signal and $S(k)$ is the power spectrum of a vibrating system whose fundamental frequency should be measured. The factor $H(k)$ represents the frequency response of the operating environment and the body of a musical instrument which filters the signal of the vibrating source. Elimination of $H(k)$ is often referred to as pre-whitening. The term $N(k)$ represents the power spectrum of additive noise. In music signals, the additive interference is mainly due to the transient-like sounds of drums and percussive instruments.

In principle, additive noise can be suppressed by performing spectral subtraction in the power spectral domain. The effect of $H(k)$, in turn, can be suppressed by highpass liftering[1] the log-magnitude spectrum. Confirming the reports of earlier authors, however, two noise-reduction systems in a cascade does not produce appropriate results [30]. Rather, successful noise suppression is achieved by applying magnitude warping which equalizes $H(k)$ while allowing the additive noise to be linearly subtracted from the result. The power spectrum $X(k)$ is magnitude-warped as

$$Y(k) = \ln\left\{1 + \frac{1}{g}X(k)\right\} \tag{2}$$

where

$$g = \left[\frac{1}{k_1 - k_0 + 1}\sum\nolimits_{l=k_0}^{k_1} X(l)^{\frac{1}{3}}\right]^3. \tag{3}$$

The frequency indices k_0 and k_1 correspond to frequencies 50 Hz and 6.0 kHz, respectively, and are determined by the frequency range utilized by the multiple-F0 estimator. The exact formula for calculating g is not as critical as the general idea represented by (2). The use of (2) and (3) is based on two reasonable assumptions. First, the amplitudes of the important frequency partials in $H(k)S(k)$ are above the additive noise $N(k)$. Secondly, it is assumed that a majority of the frequency components between k_0 and k_1 correspond to the additive noise floor, not to the spectral peaks of $H(k)S(k)$. In this case, $(1/g)$ scales the input spectrum so that the level of additive noise $N(k)$ stays close to unity and the spectral peaks of the vibrating system $H(k)S(k)$ are noticeably above unity. It follows that in (2), additive noise goes through a linear-like magnitude-warping transform, whereas spectral peaks go through a logarithmic-like transform.

The response $H(k)$ is efficiently flattened by the logarithmic-like transform, since subsequent processing takes place in the warped magnitude scale. Additive noise is suppressed by applying a specific spectral subtraction on $Y(k)$ [34]. A moving average $\hat{N}(k)$ over $Y(k)$ is calculated on a logarithmic frequency scale and then linearly subtracted from $Y(k)$. More exactly, local averages were calculated at 2/3-octave bands while constraining the minimum bandwidth to 100 Hz at the lowest bands. The same bandwidths are used in the subsequent F0 calculations and are motivated by the frequency resolution of the human auditory system and by practical experiments with generated mixtures of musical sounds and noise. The use of the logarithmic frequency scale was clearly advantageous over a linear scale since it balances the amount of spectral fine structure that is used with different F0s.

The estimated spectral average $\hat{N}(k)$ is linearly subtracted from $Y(k)$ and resulting negative values are constrained to zero

$$Z(k) = \max\left\{0,\ Y(k) - \hat{N}(k)\right\}. \tag{4}$$

The preprocessed spectrum $Z(k)$ is passed to the multiple-F0 estimator.

B. Harmonicity Principle

In this section, the "Predominant-F0 estimation" part of the algorithm is described. A process is proposed which organizes mixture spectra by utilizing the harmonic relationships between frequency components, without assuming ideal harmonicity.

Several fundamentally different approaches to F0 estimation have been proposed. One category of algorithms measures periodicity in the time-domain signal. These methods are typically based on calculating the time-domain autocorrelation function or the cepstrum representation [32], [33]. As shown in [34], this is theoretically equivalent to matching a pattern of frequency partials at *harmonic positions* of the sound spectrum. An explicit way of building upon this idea is to perform harmonic pattern matching in the frequency domain [35], [36]. Another category of algorithms measures periodicity in the frequency-domain, observing F0 from the *intervals* between the frequency partials of a sound. The spectrum autocorrelation method and its variants have been successfully used in several F0 estimators [37], [38]. An interesting difference

[1] The term "liftering" is defined [29].

between the time-domain and frequency-domain periodicity analysis methods is that the former methods are prone to errors in F0 halving and the latter to errors in F0 doubling. This is because the time-domain signal is periodic at half the F0 rate (twice the fundamental time delay) and the spectrum is periodic at double the F0 rate. A third, psychoacoustically motivated group of algorithms measures the *periodicity of the amplitude envelope* of a time-domain signal within several frequency channels [20], [21], [39].

A major shortcoming of many of the earlier proposed methods is that they do not handle inharmonic sounds appropriately. In the case of real nonideal physical vibrators, the harmonic partials are often not in exact integral ratios. For example for stretched strings the frequency f_h of an overtone partial h obeys

$$f_h = hF\sqrt{1+(h^2-1)\beta} \tag{5}$$

where F is the fundamental frequency and β is the inharmonicity factor [40]. Equation (5) means that the partials cannot be assumed to be found at harmonic spectrum positions, but are gradually shifted upwards in the spectrum. This is not of great concern in speech processing, but is important when analyzing musical sounds at a wide frequency band [41]. In the rest of this paper, capital letter F is used to denote fundamental frequency, and the lower case letter f to denote simply frequency.

The proposed predominant-F0 estimation method works by calculating independent F0 estimates at separate frequency bands and then combining the results to yield a global estimate. This helps to solve several difficulties, one of which is inharmonicity. According to (5), the higher harmonics may deviate from their expected spectral positions, and even the intervals between them are not constant. However, we can assume the spectral intervals to be piecewise constant at narrow-enough frequency bands. Thus, we utilize spectral intervals in a two step process which 1) calculates the weights of different F0s at separate frequency bands and 2) combines the results in a manner that takes inharmonicity into account. Another advantage of bandwise processing is that it provides robustness and flexibility in the case of badly corrupted signals where only a fragment of the whole frequency range can be used [41]. The two steps are now described.

1) Bandwise F0 Estimation: The preprocessed spectrum $Z(k)$ is analyzed at 18 bands that distribute approximately logarithmically between 50 Hz and 6 kHz, as illustrated in Fig. 2. Each band b comprises a 2/3-octave region of the spectrum, constraining, however, the minimum bandwidth to 100 Hz. Band b is subject to weighting with a triangular frequency response $G_b(k)$, shown in Fig. 2. The overlap between adjacent bands is 50%, making the overall response sum to unity at all except the lowest bands. Response at band b is denoted by

$$Z_b(k) = G_b(k)Z(k). \tag{6}$$

Non-zero frequency components of $Z_b(k)$ are defined for frequency indices, $k \in [k_b, k_b + K_b - 1]$ where k_b is the lowest frequency component at band b and K_b is the number of components at the band.

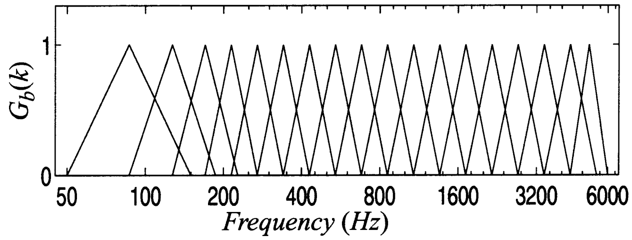

Fig. 2. Magnitude responses of the 18 frequency bands, at which the bandwise F0 estimation takes place.

In each band, the algorithm calculates a weight vector $L_b(n)$ across frequency indices. Note, index n corresponds to the fundamental frequency $F = (n/K)f_s$ where K is the number of samples in the time-domain analysis frame and f_s is the sampling rate. The resolution of the weight vector is the same as that of the preprocessed spectrum $Z(k)$. The bandwise weights $L_b(n)$ are calculated by finding a series of each n^{th} frequency components at band b that maximizes the sum

$$L_b(n) = \max_{m \in \boldsymbol{M}} \left\{ c(m,n) \sum_{j=0}^{J(m,n)-1} Z_b(k_b + m + nj) \right\} \tag{7}$$

where

$$J(m,n) = \left\lceil \frac{(K_B - m)}{n} \right\rceil \tag{8}$$

$$c(m,n) = \left[\frac{0.75}{J(m,n)} \right] + 0.25. \tag{9}$$

Here, $\boldsymbol{M} = \{0, 1, \ldots, k-1\}$ is the offset of the series of partials in the sum, $J(m,n)$ is the number of partials in the sum, and $c(m,n)$ is a normalization factor. A normalization factor is needed because J varies for different values of m and n. The form $c(m,n)$ was determined by training with isolated musical instrument samples in varying noise conditions. The offset m is varied to find the maximum of (7), which is then stored in $L_b(n)$. Different offsets have to be tested because the series of higher harmonic partials may have shifted due to inharmonicity.

The upper panel in Fig. 3 illustrates the calculations for a single harmonic sound at the band $b = 12$ between 1100 Hz and 1700 Hz. The arrows indicate the series of frequency components which maximizes $L_{12}(n)$ for the true F0.

The values of the offset m are restricted to physically realistic inharmonicities, a subset of $\boldsymbol{M}$. The exact limit is not critical, therefore (5) with a constant $\beta = 0.01$ inharmonicity factor can be used to determine the maximum allowable offset from the ideal harmonic positions. The harmonic index h in (5) can be approximated by $h \approx (k_b + K_b - 1)/n$. It follows that the fundamental partial $h = 1$ must be exactly in the harmonic spectral position, whereas the whole set $\boldsymbol{M}$ has to be considered for the highest partials. In other words, the algorithm combines the use of spectral positions for the lowest harmonic partials and the use of spectral intervals for the higher partials. For a frequency band which is assumed to contain only the first harmonic partial of a sound with fundamental frequency corresponding to index n, inharmonicity is not allowed. Here J is set to 1, and (7) reduces to the special case

$$L_b(n) = Z_b(n). \tag{10}$$

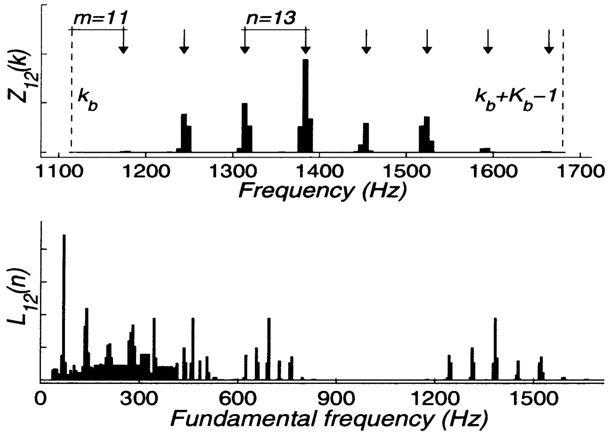

Fig. 3. Calculation of the bandwise F0 weight vectors according to (7).

TABLE I

ALGORITHM FOR CALCULATING THE WEIGHTS $L_b(n)$ FOR DIFFERENT F0s AT BAND b. SEE TEXT FOR THE DEFINITION OF SYMBOLS

```
# Implementation of the model in Eq. (7)
$n_0 \leftarrow \mathrm{floor}[(F_{min}/f_s)K]$
$n_1 \leftarrow K_b - 1$
$l_b \leftarrow k_b + K_b - 1$
for $n \leftarrow$ from $n_0$ to $n_1$ do
    $m_0 = \mathrm{round}[\mathrm{ceil}(k_b/n)n] - k_b$
    $\delta \leftarrow \frac{l_b f_s}{K}[\sqrt{1 + 0.01[(l_b/n)^2 - 1]} - 1]$
    $m_1 \leftarrow m_0 + \delta$
    if $m_1 > m_0+n-1$ then
        $m_0 \leftarrow 0$
        $m_1 \leftarrow n - 1$
    $L_b(n) \leftarrow 0$
    for $m \leftarrow$ from $m_0$ to $m_1$ do
        $J \leftarrow \mathrm{floor}[(K_b - m - 1)/n] + 1$
        $L_{now} \leftarrow (0.75/J + 0.25) \times$
                $\sum_{j=0}^{J-1} Z_b(k_b + m + nj)$
        if $L_{now} > L_b(n)$ then
            $L_b(n) \leftarrow L_{now}$
    end
end
# Range of n that have exactly one harmonic partial
# at frequency band b (inharmonicity not allowed)
$h \leftarrow 1$
$k_0 \leftarrow \mathrm{floor}[(k_b + K_b)/(h + 1)]$
if $k_0 < k_b$ then $k_0 \leftarrow k_b$
$k_1 \leftarrow k_b + K_b - 1$
while $k_0 \le k_1$ do
    for $k \leftarrow$ from $k_0$ to $k_1$ do
        $n \leftarrow \mathrm{round}(k/h)$
        if $L_b(n) < Z_b(k)$ then
            $L_b(n) \leftarrow Z_b(k)$
    end
    $h \leftarrow h + 1$
    # harmonic $h+1$ is above the band
    $k_0 \leftarrow \mathrm{ceil}[(k_b + K_b)h/(h + 1)]$
    if $k_0 < k_b$ then $k_0 \leftarrow k_b$
    # harmonic $h-1$ is below the band
    $k_1 \leftarrow \mathrm{floor}[(k_b - 1)h/(h - 1)]$
    if $k_1 > k_b + K_b$ then $k_1 \leftarrow k_b + K_b$
end
```

It follows that in this case the weights $L_b(n)$ are equal to $Z_b(n)$ between the frequency limits of the band. The algorithm is detailed in Table I.

The lower panel in Fig. 3 shows the entire weight vector $L_{12}(n)$ calculated at band $b = 12$ for the same signal as in the upper panel. As can be seen, the preprocessed spectrum $Z_{12}(n)$ appears as such at the corresponding band of $L_{12}(n)$. A twice narrower copy of $Z_{12}(n)$ is found an octave below, since the F0s in that range have exactly one harmonic partial at the band (the second partial). Yet lower F0 candidates have a series of higher overtones at the band and inharmonicity is allowed. This is the case for the true F0 (70 Hz) which has been assigned the highest weight.

An important property of the presented calculations is that only the selected frequency samples contribute to the weight $L_b(n)$, not the overall spectrum. The other co-occurring sounds affect the weight only to the extent that their partials overlap those of the sound being estimated (a solution for overlapping partials is given in Section II-C). Harmonic selection provides robustness in sound mixtures as long as we do not rely on the detection of single partials, as is the case here. Harmonic selection was originally proposed by Parsons in [27] and is used in most multiple-F0 algorithms, as described in Section I.

2) Integration of Weights Across Subbands: Fig. 4 shows the calculated $L_b(n)$ weight vectors at different bands for two isolated piano tones where the weight vectors are arranged in increasing band center frequency order. As expected, the maximum weight is usually assigned to the true F0, provided that there is a harmonic partial at that band. The inharmonicity phenomenon appears in Figs. 4(a) and 4(b) as a rising trend in the fundamental frequency.

The bandwise F0 weights are combined to yield a global F0 estimate. A straightforward summation across the weight vectors does not accumulate them appropriately since the F0 estimates at different bands may not match for inharmonic sounds, as can be seen from Fig. 4. To overcome this, the inharmonicity factor is estimated and taken into account. Two different inharmonicity models were implemented, the one given in (5) and another mentioned in [40, p. 363]. In simulations, the performance difference between the two was negligible. The model in (5) was adopted.

Global weights $L(n)$ are obtained by summing squared bandwise weights $L_b(n)$ that are selected from different bands according to a curve determined by (5). A search over possible values of $\beta(n)$ is conducted for each n, and the highest $L(n)$ and the corresponding $\beta(n)$ are stored in the output. Squaring the bandwise F0 weights prior to summing was found to provide robustness in the presence of strong interference where the pitch may be perceptible only at a limited frequency range.

The global F0 weights $L(n)$ and inharmonicity factors $\beta(n)$ do not need to be calculated for all fundamental frequency indices n. Instead, only a set of fundamental frequency indices $\{n_1, n_2, \ldots, n_Q\}$ is collected from the bandwise weight vectors $L_b(n)$. This is possible, and advantageous since if a sound is perceptible at all, it generally has a high weight in at least one of the bands. Selecting a couple of maxima from each band preserves the correct fundamental frequency among the candidates.

The maximum global weight $L(n)$ can be used as such to determine the true F0. However, an even more robust selec-

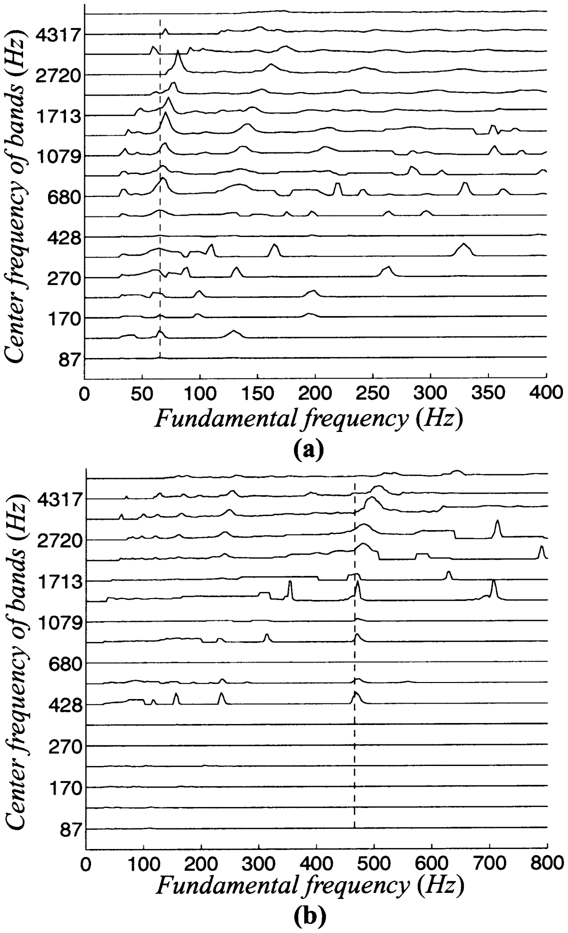

Fig. 4. Bandwise-calculated F0 weights $L_b(n)$ for two piano tones, Figure (a) with F0 65 Hz and Figure (b) with F0 470 Hz. The vectors are displaced vertically for clarity. The true pitches of the tones are indicated with dashed vertical lines.

tion among the candidates can be made by further inspecting the spectral smoothness of the F0s that have the highest global weights. This is the reason why a smoothing module is used in Fig. 1 before storing the F0. This module will be described in detail in Section III. For the sake of discussion in Section II-C one can assume that the maximum global score $L(n)$ determines the predominant F0.

C. Spectral Smoothness Principle

1) Iterative Estimation and Separation: The presented method is capable of making robust predominant-F0 detections in polyphonic signals. Moreover, the inharmonicity factor and precise frequencies of each harmonic partial of the detected sound are produced. A natural strategy for extending the presented algorithm to multiple-F0 estimation is to remove the partials of the detected sound from the mixture and to apply the predominant-F0 algorithm iteratively to the residual spectrum.

Detected sounds are separated in the frequency domain. Each sinusoidal partial of a sound is removed from the mixture spectrum in two stages. First, good estimates of the frequency and amplitude of the partials must be obtained. It is assumed that these parameters remain constant in the analysis frame. Second, using the found parameters, the spectrum in the vicinity of the partials is estimated and linearly subtracted from the mixture spectrum.

Initial estimates for the frequency and amplitude of each sinusoidal partial of a sound are produced by the predominant-F0 detection algorithm. Efficient techniques for estimating more precise values have been proposed e.g. in [42]. A method widely adopted is to apply Hamming windowing and zero padding in the time domain, to calculate Fourier spectrum, and to use quadratic interpolation of the spectrum around the partial. The second problem, estimating the spectrum in the vicinity of the partial is equivalent to translating the magnitude spectrum of the original analysis window at the frequency of the sinusoidal partial. For Hamming window without zero padding, it was found to be sufficient to perform the subtraction for five adjacent frequency bins.

2) The Problem of Coinciding Frequency Partials: One issue that is addressed in the algorithm is the problem of coinciding frequency partials. To illustrate this problem, simulations were run using the iterative procedure on randomly generated F0 mixtures. Fig. 5 shows the errors as a function of the musical intervals that occur in the erroneously transcribed sound mixtures (see Appendix). In most cases, the iterative approach works rather reliably. However, an important observation can be made when the distribution of the errors in Fig. 5 is analyzed. The error rate is strongly correlated with certain F0 relations. The conclusion to be noted is that a straightforward estimation and subtraction approach is likely to fail in cases where the fundamental frequencies of simultaneous sounds have simple rational number relations, also called *harmonic* relations. These are indicated over the corresponding bars in Fig. 5.

Coinciding frequency partials from different sounds can cause the algorithm to fail since many of the partials coincide in frequency. When the sound detected first is removed, the coinciding harmonics of remaining sounds are corrupted in the subtraction procedure. After several iterations, a remaining sound can become too corrupted to be correctly analyzed in the iterations that follow.

When two sinusoidal partials with amplitudes a_1 and a_2 and phase difference θ_Δ coincide in frequency, the amplitude of the resulting sinusoid can be calculated as

$$a_s = |a_1 + a_2 e^{i\theta_\Delta}|. \tag{11}$$

If the two amplitudes are roughly equivalent, the partials may either amplify or cancel each other, depending on θ_Δ. However, if one of the amplitudes is significantly greater than the other, as is usually the case, a_s approaches the maximum of the two.

Assuming ideal harmonicity, it is straightforward to prove that the harmonic partials of two sounds coincide if and only if the fundamental frequencies of the two sounds are in rational number relations. Moreover, if the harmonic indices of the coinciding partials are p and q, then every p^{th} partial of the first sound coincides with every q^{th} partial of the other sound. An important principle in Western music is to pay attention to the pitch relationships of simultaneously played notes. Simple harmonic relationships are favored over dissonant ones in order to make the sounds blend better. Because harmonic relationships

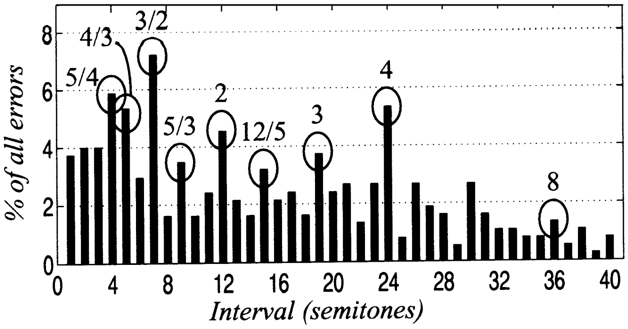

Fig. 5. Distribution of the F0 estimation errors as a function of the musical intervals that occur in the erroneously transcribed sound mixtures.

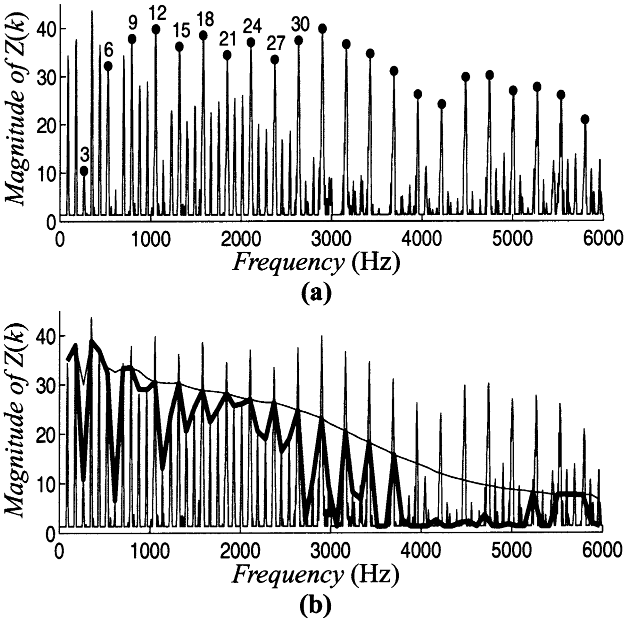

Fig. 6. Illustration of the spectral smoothness principle. (a) Preprocessed spectrum $Z(k)$ containing two sounds with F0s in the relation 1:3. (b) Two different smoothing operations have been used to estimate the spectral envelope of the lower-pitched sound. The results are indicated with thin and thick horizontal curves.

are so common in music, these "worst cases" must be handled well in general.

To solve this problem, the spectra of the detected sounds must be smoothed before subtracting them from the mixture. Consider the preprocessed spectrum $Z(k)$ of a two-sound mixture in Fig. 6(a). In the figure, the harmonic partials of the higher-pitched sound coincide with every third harmonic of the lower-pitched sound $(F_{higher} = 3F_{lower})$. As predicted by (11), the coinciding partials randomly cancel or amplify each other at the low frequencies, whereas at the higher frequencies the summary amplitudes approach the maximum of the two, i.e., the spectral envelope of the higher sound.

When the spectrum of the lower-pitched sound is smoothed (the thin slowly decreasing horizontal curve in Fig. 6(b)), the coinciding partials at the higher frequencies rise above the smooth spectrum and thus remain in the residual after subtraction. In particular, this solves a very common case where the dense harmonic series of a lower-pitched sound matches the few partials of a higher-pitched sound. Detecting the higher-pitched sound first is less common and in that case, only a minority of the harmonics of the lower-pitched sound are deleted.

It should be noted that simply smoothing the amplitude envelope (the thin curve in Fig. 6(b)) of a sound before subtracting it from the mixture does not result in lower error rates. A successful smoothing algorithm was found by applying psychoacoustic knowledge. The full motivation for this approach has been presented in [43] and is beyond the scope of this paper.

The algorithm first calculates a moving average over the amplitudes of the harmonic partials of a sound. An octave-wide triangular weighting window is centered at each harmonic partial h, and the weighted mean d_h of the amplitudes of the partials in the window is calculated. This is the smooth spectrum illustrated by a thin horizontal curve in Fig. 6(b). The original amplitude value a_h is then replaced with the minimum of the original (a_h) and d_h:

$$a_h \leftarrow min(a_h, d_h). \tag{12}$$

These values are illustrated by a thick curve in Fig. 6(b). Performing this straightforward smoothing operation before subtracting the sound from the mixture reduces the error rates significantly.

A further improvement to the smoothing method can be made by utilizing the statistical dependency of every p^{th} harmonic partial, as was previously explained following (11) in this section. The algorithm applies a multistage filter with the following steps [43]. First, the indices $\{\ldots, h-1, h, h+1, h+2, \ldots\}$ of the harmonic partials around harmonic h are collected from an octave-wide window. Next, the surrounding partials are classified into groups, where all the harmonics that share a common divisor are put in the same group, starting from the smallest prime factors. Third, weighted mean around harmonic h is calculated inside groups in the manner described above. In the last step, the estimates of different groups are averaged, weighting each group according to its mean distance from harmonic h.

3) Recalculation of F0 Weights After Smoothing: The described principle of smoothing provides an efficient solution to another common class of errors. In this class of errors two or more fundamental frequencies in specific relationships may cause the detection of a nonexistent sound, such as the root of a musical chord in its absence. For instance, when two harmonic sounds with fundamental frequencies $2F$ and $3F$ are played, the spectra of these sounds match every second and every third harmonic partial of a nonexisting sound with fundamental frequency F. This frequency F may be erroneously estimated in the predominant-F0 calculations given the observed partials.

The problem can be solved by applying smoothing and an ordered search when selecting among the candidate indices n_i calculated by the predominant-F0 algorithm (see the end of Section II-B). First, the candidate n_1 with the highest global weight $L(n_1)$ is taken and its spectrum is smoothed. Then the weight of this candidate is recalculated using the smoothed harmonic amplitudes. In the above-described case of a nonexistent sound, the irregularity of the spectrum decreases the level of the smooth spectrum significantly, and the weight remains low. If the recalculated weight drops below the second-highest weight, the next candidate n_2 is processed, and this is continued. The highest recalculated global weight determines the F0. The computational load of applying smoothing and recalculation to select among the candidates is negligible, since the recalculation procedure has to consider only one F0 and one value of m in (7).

D. Estimating the Number of Concurrent Sounds

A mechanism is needed which controls the stopping of the iterative F0 estimation and sound separation process. This leads to the estimation of the number of concurrent sounds, i.e. the polyphony. The difficulty of the task is comparable to that of finding the F0 values themselves. Huron has studied musicians' ability to identify the number of concurrently sounding voices in polyphonic textures [44]. According to his report by four-voice polyphonies the test subjects underestimated the number of voices in more than half of the cases.

A statistical-experimental approach was taken to solve the problem. Random mixtures of one to six concurrent harmonic sounds were generated by allotting sounds from McGill University Master Samples collection [45]. The mixtures were then contaminated with pink noise or random drum sounds from Roland R-8 mk II drum machine. Signal-to-noise ratio was varied between 23 dB and -2 dB.

The behavior of the iterative multiple-F0 estimation system was investigated using these artificial mixtures with known polyphonies. Based on this investigation it was decided to split the estimation task into two stages. The first stage detects if there are any harmonic sounds at all in the input, and the second estimates the number of concurrent sounds, if the first test has indicated that some are present. It was found that the best single feature to indicate the presence of harmonic sounds was the global weight $L_{max}^{(1)}$ of the winning F0 candidate at the first iteration. The best compound feature consists of $L_{max}^{(1)}$ and terms related to the signal-to-noise ratio (SNR) of the input signal:

$$v_0 = 4\ln[L_{max}] + \ln\left[\sum_{l=k_0}^{k_1} X(l)\right] - \ln\left[\sum_{l=k_0}^{k_1} \hat{N}_{(\mathrm{pow})}(1)\right]. \tag{13}$$

Here $X(k)$ is the discrete power spectrum of the input signal and $\hat{N}_{(\mathrm{pow})}(k)$ is the power spectrum of the estimated noise, obtained by applying inverse transform of (2) on $\hat{N}(k)$. Frequency indices k_0 and k_1 are the same as in (3). A signal is determined to contain harmonic sounds when v_0 is greater than a fixed threshold.

If an analysis frame has been determined to contain harmonic sounds, another model is used to estimate the number of sounds. The maximum global weight $L_{max}^{(i)}$ at iteration i was again the best single feature for controlling the iteration stopping. However, the weight values are affected by the SNR $L_{max}^{(i)}$ getting smaller in noise. The bias can be explicitly corrected, resulting in the measure

$$v_i = 1.8\ln\left(L_{max}^{(i)}\right) - \ln\left[\sum_{l=k_0}^{k_1} X(l)\right] + \ln\left[\sum_{l=k_0}^{k_1} \hat{N}_{(\mathrm{pow})}(1)\right]. \tag{14}$$

As long as the value of v_i stays above a fixed threshold, the sound detected at iteration i is accepted as a valid F0 estimate and the iteration is continued. In (13) and (14), the SNR-related terms have different roles and thus different signs.

III. Results

A. Experimental Setup

Simulations were run to validate the proposed methods. The acoustic database consisted of samples from four different sources. The McGill University Master Samples collection [45] and independent recordings for acoustic guitar were available already during the development phase of the system. In order to verify that the results generalize outside these data sets, the samples from the University of Iowa website [46] and IRCAM Studio Online [47] were added to the final evaluation set. There were altogether 30 different musical instruments, comprising brass and reed instruments, strings, flutes, the piano, and the guitar. These introduce several different sound production mechanisms and a variety of spectra. On the average, there were 1.8 pieces of each of the 30 instruments and 2.5 different playing styles per instrument. The total number of samples was 2536. These were randomly mixed to generate test cases. The instruments marimba and the vibraphone were excluded from the data set since their spectrum is quite different from the others and extremely inharmonic. The system admittedly cannot handle these sounds reliably.

Semirandom sound mixtures were generated according to two different schemes. *Random mixtures* were generated by first allotting an instrument and then a random note from its whole playing range, restricting, however, the pitch over five octaves between 65 Hz and 2100 Hz. The desired number of simultaneous sounds were allotted and then mixed with equal mean-square levels. *Musical mixtures* were generated in a similar manner, but favoring different pitch relationships according to a statistical profile discovered by Krumhansl in classical Western music [48, p. 68]. In brief, octave relationships are the most frequent, followed by consonant musical intervals, and the smallest probability of occurrence is given to dissonant intervals. In general, musical mixtures are more difficult to resolve (see Section II-C2).

Acoustic input was fed to the multiple-F0 algorithm that estimated F0s in a single time frame. Unless otherwise stated, the number of F0s to extract, i.e., the polyphony, was given along with the mixture signal. It was found to be more informative to first evaluate the multiple-F0 estimator without the polyphony estimator, because these two are separable tasks and because the reference methods do not implement polyphony estimation. The configuration and parameters of the system were fixed unless otherwise stated. A correct F0 estimate was defined to deviate less than half a semitone (±3%) from the true value, making it "round" to a correct note on a Western musical scale. Errors smaller than this are not significant from the point of view of music transcription.

B. Reference Methods

To put the results in perspective, two reference methods were used as a baseline in simulations. The first method, *YIN*, is a state-of-the-art *monophonic* F0 estimator for speech and music signals [49]. Naturally, the method can be used as a baseline in single-F0 analysis only. The algorithm has been designed to be reliable for individual analysis frames and has been thoroughly tested and compared with other methods in [49]. The original implementation by the authors was employed and parameters were left intact except the "absolute threshold" which was fine-tuned to value 0.15 to improve the performance.

The other reference method, referred to as *TK*, is a multiple-F0 estimator proposed by Tolonen and Karjalainen in [21]. The implementation was carefully prepared based on the reference, and the original code by the authors was used in the warped linear prediction part of the algorithm. Thorough testing was carried out to verify the implementation. Original parameters given in [21] were applied. As reported by the authors, the method cannot handle "spectral pitch," i.e., F0s above 1 kHz. It was further found out here that the method is best at detecting F0s in the three-octave range between 65 Hz and 520 Hz. Thus, in the simulations that follow, the mixtures given to the *TK* method were restricted to contain F0s below either 520 Hz or 1 kHz. The bound is specified for each case in the simulation results to follow.

C. Experimental Results

In the first experiment, different F0 estimators are compared. For this experiment, a predominant-F0 estimate (firstly detected F0) was defined to be correct if it matches the correct F0 of *any* of the component sounds. That is, only a single match among all possible F0s is required in this error measure. The error rate was calculated as the amount of predominant-F0 errors divided by the number of random sound mixtures (1000), not by the number of reference notes (e.g. 6000 in the six-note mixtures). F0 estimation was performed in a single 190 ms time frame 100 ms after the onset of the sounds. Fig. 7 shows the error rates for the predominant-F0 estimation in different polyphonies. Results are given for the proposed system and for the two reference systems.

For the proposed system, the error rates are generally below 10%, getting close only in six-note polyphonies. Surprisingly, increasing the number of concurrent sounds from one to two appears to help lower the error rate of detecting at least one F0 correctly. However, this is due to the fact that the acoustic database contains a small percentage of irregular sounds for which the simple model in (7) does not work. Among these are e.g. high flute tones and high plucked string tones. Two-sound mixtures are more likely to contain at least one clear sound with no anomalities, which then appears as the predominant F0.

The *YIN* method achieves 4.1% error rate for isolated notes. Since the method is not intended for multiple-F0 estimation, it is not fair to make comparison for polyphonic signals. Like other single-F0 estimators, the algorithm converges to 70% error rate already in three-note mixtures. The *TK* method is not quite as reliable for single-pitch signals, but works robustly in polyphony. If the method is given F0s only below 520 Hz, the predominant-F0 detection accuracy comes close to the proposed system in higher polyphonies. This is partly due to the relatively higher random guess rate.

In the second experiment, the performance of multiple-F0 estimation is explored in more detail. For multiple-F0 estimation, a more appropriate error measure is a *note error rate (NER)* metric. The NER is defined as the sum of the F0s in error divided by the number of F0s in the reference transcription. The errors are of three types:

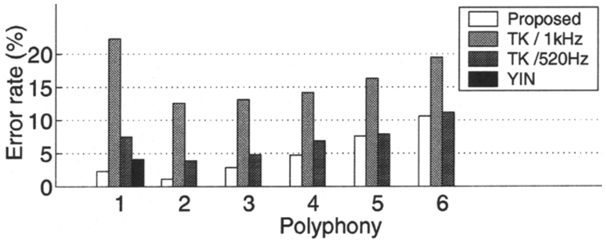

Fig. 7. Error rates for detecting any of the potential F0s in a sound as a function of the predominant-F0 estimation algorithm and the polyphony.

a) Substitution errors. These are defined as errors in which a given F0 is detected but the estimated value differs more than ±3% from the reference.

b) Deletion errors have occurred if the number of detected F0s is smaller than the number of F0s in the reference.

c) Insertion errors have occurred if the number of detected F0s exceeds that in the reference.

Substitution and deletion errors together can be counted from the number of F0s in the reference that are not correctly estimated. Insertion errors can be counted from the number of excessive estimates.

Results for multiple-F0 estimation in different polyphonies are shown in Fig. 8. Here the number of concurrent sounds to extract was given for each mixture signal, i.e., the polyphony was known. Thus insertion and deletion errors do not occur. Random and musical sound mixtures were generated according to the described schemes, and the estimator was then requested to find a given number of F0s in a single 190 ms time frame 100 ms after the onset of the sounds.

In Fig. 8, the bars represent the overall NER's as a function of the polyphony. As can be seen, the NER for random four-sound polyphonies is 9.9% on the average. The different shades of gray in each bar indicate the error cumulation in the iteration, errors which occurred in the first iteration at the bottom, and errors of the last iteration at the top. As a general impression, the system works reliably and exhibits graceful degradation in increasing polyphony. Results for musical mixtures are slightly worse than for random mixtures (see Section II-C2), but the difference is not great. This indicates that the spectral smoothing principle works well in resolving harmonically related pitch combinations.

Analysis of the error cumulation reveals that the errors which occurred in the last iteration account for approximately half of the errors in all polyphonies, and the probability of error increases rapidly in the course of iteration. Besides indicating that the subtraction process does not work perfectly, the conducted listening tests suggest that this is a feature of the problem itself, rather than only a symptom of the algorithms used. In most mixtures, there is a sound or two that are very difficult to perceive because their spectrum is virtually hidden under the other sounds.

For the reference method *TK*, note error rates for mixtures ranging from one to six sounds were 22%, 31%, 39%, 45%,

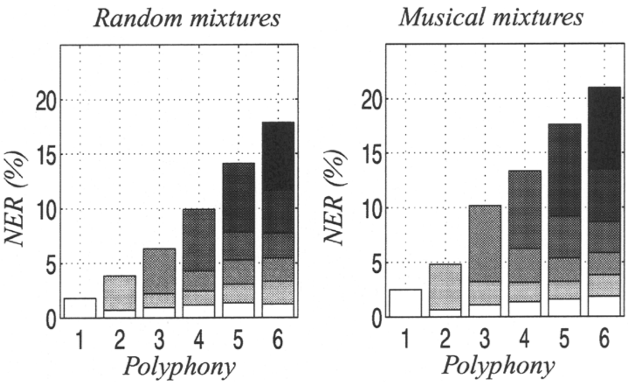

Fig. 8. Note error rates for multiple-F0 estimation using the proposed algorithm when the polyphony was known. Bars represent the overall error rates, and the different shades of gray the error cumulation in iteration.

49%, and 53%, respectively, when F0s were resticted to range 65 Hz–1 kHz. For the three-octave range between 65 Hz and 520 Hz, the corresponding error rates were 7.5%, 17%, 26%, 34%, 38%, and 43%. Given the complexity of the problem, even these error rates are rather low.

Table II gives the error rates for different system configurations. Different processing elements were disabled one-by-one in order to evaluate their importance. In each case, the system was kept otherwise fixed. In the first test, the mechanisms that accommodate inharmonicity were disabled. One mechanism is in bandwise F0-weight calculations, and in this case the offset m in (7) was constrained to a value which corresponds to an ideal harmonicity. Another mechanism is in the integration phase. Here the inharmonicity factor was constrained to zero, leading to a straightforward summing across squared weight vectors. The resulting performance degradation is mostly due to the bandwise calculations.

In the second test, the spectral smoothing algorithm was switched between the one presented in Section II-C2 and a version which leaves the harmonic series intact. The smoothing operation made a significant improvement to multiple-F0 estimation accuracy in all polyphonies, except for the single-note case where it did not have a noticeable effect on the performance.

In all the results presented above, the polyphony of the signals was known. Fig. 9 shows the statistical error rate of the overall multiple-F0 estimation system when the polyphony is estimated in the analysis frame, as described in Section II-D. Results are shown for two different polyphony estimation thresholds (i.e., thresholds for v_i in (14) which were 0.65 and 1.1 for the left and right panels, respectively). Depending on the application, either overestimating or underestimating the number of concurrent sounds may be more harmful. In a music transcription system, for example, extraneous notes in the output are very disturbing. However, if the frame-level F0 estimates are further processed at a higher level, it is usually advantageous to produce too many rather than too few note candidates.

In general, the proposed polyphony estimation method operates robustly. However, when the estimation threshold is tuned to avoid extraneous detections in monophonic signals, the polyphony is underestimated in higher polyphonies. On the other hand, when underestimations are avoided, many of the extraneous F0s appear in monophonic signals. This is likely to be characteristic of the problem itself (see Huron's report [44] mentioned in Section II-D). One or two sounds in rich polyphonies are usually very difficult to distinguish.

TABLE II
ERROR RATES FOR DIFFERENT SYSTEM CONFIGURATIONS WHEN THE POLYPHONY OF THE SIGNALS WAS KNOWN

System configuration	Polyphony	
	1	4
Complete system	1.8 %	9.9 %
Inharmonicity not allowed	6.2 %	17 %
No smoothing	2.2 %	20 %

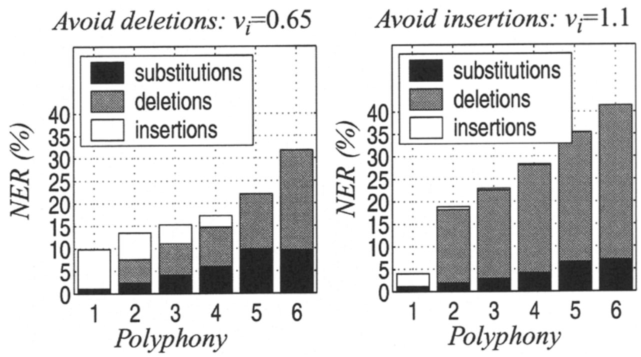

Fig. 9. Error rates for the two different polyphony estimations strategies.

Table III shows the influence of shortening the analysis frame. The significant difference between 190 ms and 93 ms frame sizes is partly caused by the fact that the applied technique was sometimes not able to resolve the F0 with the required ±3% accuracy. Also, irregularities in the sounds themselves, such as vibrato, are more difficult to handle in short frames. However, when the time frame was shortened from 190 ms to 93 ms, the error rate of the reference method *TK* increased only by approximately 5% for both 1000 Hz and 520 Hz F0 limits and in all polyphonies. Thus, the error rates of *TK* were essentially the same as those presented around Fig. 8. While the performance is still clearly worse than that of the proposed method (polyphony was known), an obvious drawback of the proposed method is that its accuracy depends on the length of the analysis frame. A basic reason for this is that the linear frequency resolution of spectral methods does not suffice at the low end, whereas the frequency resolution of autocorrelation-based methods is proportional to the inverse of frequency, being closer to the logarithmic frequency resolution of musical scales and human hearing. Despite these differences, reliable multiple-F0 estimation in general seems to require longer time frames than single-F0 estimation.

Fig. 10 shows the NER's in different types and levels of additive noise when the polyphony was known. Pink noise was generated in the band between 50 Hz and 10 kHz. Percussion instrument interference was generated by randomizing drum samples from a Roland R-8 mk II drum machine. The test set comprised 33 bass drum, 41 snare, 17 hi-hat, and 10 cymbal

TABLE III
ERROR RATES FOR DIFFERENT ANALYSIS FRAME LENGTHS

Polyphony estimation	Frame size	Actual polyphony					
		1	2	3	4	5	6
Polyphony known	190 ms	1.8	3.9	6.3	9.9	14	18
	93 ms	4.2	8.7	16	22	29	34
Estimate, avoid deletions	190 ms	11	14	16	18	22	32
	93 ms	14	19	23	30	38	46

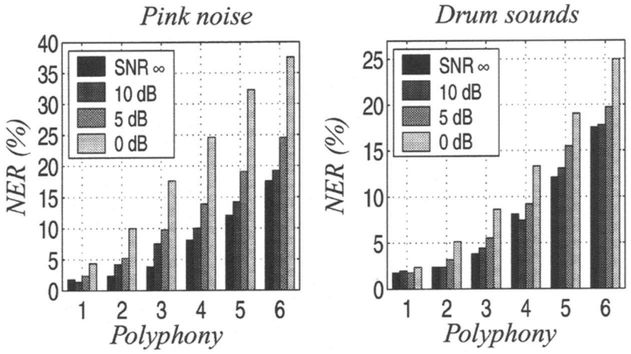

Fig. 10. Error rates in additive pink noise (left panel) and with interfering percussive sounds (right panel). For both noise types, error rates for a clean signal and for noisy signals with SNR's 10 dB, 5 dB, and 0 dB are given. Polyphony was known.

sounds. The signal-to-noise ratio was adjusted within the analysis frame, and the ratio was defined between the noise and the *sum* of the harmonic sounds. Thus, the SNR from the point of view of individual sounds is much worse in higher polyphonies. A 190 ms frame was applied.

D. Comparison With Human Performance

Listening tests were conducted to measure the human pitch identification ability, particularly the ability of trained musicians to transcribe polyphonic sound mixtures. Detailed analysis of the results is beyond the scope of this article. Only a summary of the main findings can be reviewed here.

Test stimuli consisted of computer-generated mixtures of simultaneously onsetting sounds that were reproduced using sampled Steinway grand piano sounds from the McGill University Master Samples collection [45]. The number of co-occurring sounds varied from two to five. The interval between the highest and the lowest pitch in each individual mixture was never wider than 16 semitones in order to make the task feasible for the subjects that did not have "absolute pitch", i.e., the rare ability of being able to name the pitch of a sound without a reference tone. Mixtures were generated from three pitch ranges (i.e., registers): low (33 Hz–130 Hz), middle (130 Hz–520 Hz), and high (520 Hz–2100 Hz). In total, the test comprised 200 stimuli.

The task was to write down the musical intervals, i.e., pitch relations, of the presented sound mixtures. Absolute pitch values were not asked for and the number of sounds in each mixture was given. Thus, the test resembles the musical interval and chord identification tests that are a part of the basic musical training in Western countries.

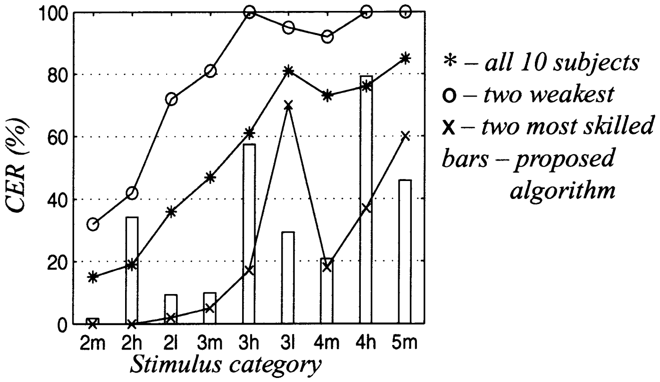

Fig. 11. Chord error rates of the human listeners (curves) and of the proposed algorithm (bars) for different stimulus categories. The lowest curve represents the two most skilled subjects, the middle curve the average of all subjects, and the highest curve the two clearly weakest subjects. The labels of the simulus categories consist of a number which signifies the polyphony, and of a letter which tells the pitch range used.

A total of ten subjects participated in the test. All of them were trained musicians in the sense of having taken several years of ear training[2] in music. Seven subjects were students of musicology at university level. Two were more advanced musicians, possessing absolute pitch and exceptional pitch identification abilities. One subject was an amateur musician of similar musical ability as the seven students.

Fig. 11 shows the results of the listening test. Chord error rates (CER) are plotted for different stimulus categories. CER is the percentage of sound mixtures where one or more pitch identification errors occurred. The labels of the categories consist of a number which signifies the polyphony, and of a letter which tells the pitch range used. Letter "m" refers to the middle, "h" to the high, and "l" to the low register. Performance curves are averaged over three different groups. The lowest curve represents the two most skilled subjects, the middle curve the average of all subjects, and the highest curve the two clearly weakest subjects.

The CER's cannot be directly compared to the NER's given in Fig. 8. The CER metric is more demanding, accepting only sound mixtures where all pitches are correctly identified. It had to be adopted to unambiguously process the musicians' answers, which were given as pitch intervals.

For the sake of comparison, the stimuli and performance criteria used in the listening test were used to evaluate the proposed computational model. Five hundred instances were generated from each category included in Fig. 11, using the same software that randomized samples for the listening test. These were fed to the described multiple-F0 system. The CER metric was used as a performance measure.

The results are illustrated with bars in Fig. 11. As a general impression, only the two most skilled subjects perform better than the computational model. However, performance differences in high and low registers are quite revealing. The devised algorithm is able to resolve combinations of low sounds that are beyond the ability of human listeners. This seems to be due to the good frequency resolution applied. On the other hand,

[2] The aim of ear training in music is to develop the faculty of discriminating sounds, recognizing musical intervals, and playing music by ear, i.e., without the aid of written music.

human listeners perform relatively well in the high register. This is likely to be due to an efficient use of the temporal features, onset asynchrony and different decay rates, of high piano tones. These were not available in the single time frame given to the multiple-F0 algorithm.

IV. Conclusions

The paper shows that multiple-F0 estimation can be performed reasonably well using only spectral cues, harmonicity and spectral smoothness, without the need for additional long-term temporal features. For a variety of musical sounds, a prior knowledge of the type of sound sources involved is not necessary, although adaptation of internal source (e.g. instrument) models would presumably further enhance the performance.

The primary problem in multiple-F0 estimation appears to be in associating partials correctly with their individual sources of production. The harmonicity principle must be applied in a manner that is flexible enough to accommodate a realistic amount of inharmonicity in sound production, and yet constraining enough to prevent erroneous groupings. Contrasted with the complexity needed in handling inharmonicity, the harmonic summation model used to calculating F0 weights from the amplitudes of the grouped partials is very simple, as embodied in (8) and (9).

A spectral smoothing approach was proposed as an efficient new mechanism in multiple-F0 estimation and spectral organization. The introduction of this principle corrected approximately half of the errors occurring in a system which was otherwise identical but did not use the smoothness principle.

An attractive property of the iterative estimation and separation approach is that at least a couple of the most prominent F0s can be detected even in very rich polyphonies. The probability of error increases rapidly in the course of the iteration, but on the basis of the listening tests it was suggested that this is at least in part due to the inherent characteristics of the problem itself. The last iteration, i.e., estimation of the F0 of the sound detected last, accounts for approximately half of the errors in all polyphonies.

The main drawback of the presented method is that it requires a relatively long analysis frame in order to operate reliably for low-pitched sounds. This is largely due to the fact that the processing takes place in the frequency domain where sufficiently fine frequency resolution is required for harmonic series of low-pitched sounds.

The described method has been applied to the automatic transcription of continuous music on CD recordings. Some demonstration signals are provided at [50]. Contrary to the musical chord identification task, however, the accuracy is not comparable to that of trained musicians. There are several possibilities that can be explored as areas of future development. Integration across multiple time frames can be used to improve performance. While independent multiple-F0 estimation in each time frame is important for feature extraction, it does not account for the real experience represented in a human listener. Analogous to the case of speech recognition in which models of words and language are used to improve performance, use of higher-level features in music are also expected to improve music estimation and transcription tasks.

TABLE IV
Some Basic Musical Intervals

Interval name	Size (semitones)	F0 relation
octave	12	2:1
perfect fifth	7	3:2
perfect fourth	5	4:3
major third	4	5:4
minor third	3	6:5
major second	2	9:8

Appendix

Western music typically uses a *well-tempered* musical scale. That is, the notes are arranged on a logarithmic scale where the fundamental frequency F_k of a note k is $F_k = 440 \times 2^{(k/12)}$ Hz.

The notes on a standard piano keyboard range from $k = -48$ up to $k = 39$. The term *semitone* refers to the interval between two adjacent notes and is used to measure other musical intervals. The F0 relation of two notes that are one semitone apart is $F_{k+1}/F_k = 2^{(1/12)} \approx 1.06$.

Although the well-tempered scale is logarithmic, it can surprisingly accurately generate F0s that are in rational number relations. Table IV lists some basic musical intervals and the corresponding ideal rational number relations. Intervals which approximate simple rational number relationships are called *harmonic*, or, *consonant* intervals, as opposed to *dissonant* intervals.

References

[1] S. Bregman, *Auditory Scene Analysis: The Perceptual Organization of Sound*. Cambridge, MA: MIT Press, 1990.

[2] L. R. Rabiner, M. J. Cheng, A. E. Rosenberg, and C. A. McGonegal, "A comparative performance study of several pitch detection algorithms," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-24, no. 5, pp. 399–418, 1976.

[3] W. J. Hess, "Pitch and voicing determination," *Advances in Speech Signal Processing*, 1991.

[4] A. de Cheveigné and H. Kawahara, "Comparative evaluation of F0 estimation algorithms," in *Proc. Eurospeech*, Copenhagen, Denmark, 2001, pp. 2451–2454.

[5] W. M. Hartmann, "Pitch, periodicity, and auditory organization," *J. Acoust. Soc. Amer.*, vol. 100, no. 6, pp. 3491–3502, 1996.

[6] J. A. Moorer, "On the transcription of musical sound by computer," *Comput. Music J.*, pp. 32–38, Nov. 1977.

[7] C. Chafe and D. Jaffe, "Source separation and note identification in polyphonic music," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Tokyo, 1986, pp. 1289–1292.

[8] R. C. Maher, "Evaluation of a method for separating digitized duet signals," *J. Audio Eng. Soc.*, vol. 38, no. 12, pp. 956–979, 1990.

[9] H. Katayose and S. Inokuchi, "The Kansei music system," *Comput. Music J.*, vol. 13, no. 4, pp. 72–77, 1989.

[10] M. Hawley, "Structure Out of Sound," Ph.D. dissertation, MIT Media Laboratory, Cambridge, MA, 1993.

[11] L. Rossi, "Identification de sons polyphoniques de piano," Ph.D. thesis, L'Universite de Corse, Corsica, France, 1998.

[12] D. F. Rosenthal and H. G. Okuno, Eds., *Computational Auditory Scene Analysis*. Mahwah, NJ: Lawrence Erlbaum, 1998.

[13] K. Kashino, K. Nakadai, T. Kinoshita, and H. Tanaka, "Organization of hierarchical perceptual sounds: music scene analysis with autonomous processing modules and a quantitative information integration mechanism," in *Proc. Int. Joint Conf. Artificial Intelligence*, Montréal, QC, Canada, 1995.

[14] K. D. Martin, "Automatic Transcription of Simple Polyphonic Music: Robust Front End Processing," Massachusetts Institute of Technology Media Laboratory Perceptual Computing Section, Tech. Rep. 399, 1996.

[15] D. P. W. Ellis, "Prediction-Driven Computational Auditory Scene Analysis," Ph.D. thesis, MIT Media Laboratory, Cambridge, Massachusetts, 1996.

[16] M. Goto, "A robust predominant-F0 estimation method for real-time detection of melody and bass lines in CD recordings," in *Proc. IEEE International Conf. on Acoust., Speech, and Signal Processing*, Istanbul, Turkey, June 2000.

[17] G. J. Brown and M. P. Cooke, "Perceptual grouping of musical sounds: a computational model," *J. New Music Res.*, vol. 23, pp. 107–132, 1994.

[18] D. Godsmark and G. J. Brown, "A blackboard architecture for computational auditory scene analysis," *Speech Commun.*, vol. 27, pp. 351–366, 1999.

[19] R. Meddis and M. J. Hewitt, "Virtual pitch and phase sensitivity of a computer model of the auditory periphery I: pitch identification," *J. Acoust. Soc. Amer.*, vol. 89, no. 6, pp. 2866–2882, 1991.

[20] R. Meddis and L. O'Mard, "A unitary model of pitch perception," *J. Acoust. Soc. Amer.*, vol. 102, no. 3, pp. 1811–1820, 1997.

[21] T. Tolonen and M. Karjalainen, "A computationally efficient multipitch analysis model," *IEEE Trans. Speech Audio Processing*, vol. 8, pp. 708–716, Nov. 2000.

[22] A. de Cheveigné and H. Kawahara, "Multiple period estimation and pitch perception model," *Speech Commun.*, vol. 27, pp. 175–185, 1999.

[23] R. Meddis and M. J. Hewitt, "Modeling the identification of concurrent vowels with different fundamental frequencies," *J. Acoust. Soc. Amer.*, vol. 91, no. 1, pp. 233–245, 1992.

[24] A. de Cheveigné, "Separation of concurrent harmonic sounds: fundamental frequency estimation and a time-domain cancellation model of auditory processing," *J. Acoust. Soc. Amer.*, vol. 93, no. 6, pp. 3271–3290, 1993.

[25] W. A. Sethares and T. W. Staley, "Periodicity transforms," *IEEE Trans. Signal Processing*, vol. 47, no. 11, pp. 2953–2964, 1999.

[26] T. Nakatani and H. G. Okuno, "Harmonic sound stream segregation using localization and its application to speech stream segregation," *Speech Commun.*, vol. 27, pp. 209–222, 1999.

[27] T. W. Parsons, "Separation of speech from interfering speech by means of harmonic selection," *J. Acoust. Soc. Amer.*, vol. 60, pp. 911–918, 1976.

[28] T. Verma, "A Perceptually Based Audio Signal Model With Application to Scalable Audio Compression," Ph.D. dissertation, Stanford University, 2000.

[29] L. R. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[30] H. Hermansky, N. Morgan, and H.-G. Hirsch, "Recognition of speech in additive and convolutive noise based on RASTA spectral processing," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Processing*, Minneapolis, Minnesota, 1993.

[31] A. P. Klapuri, "Automatic transcription of musical recordings," in *Proc. Consistent and Reliable Acoustic Cues Workshop*, Aalborg, Denmark, Sep. 2001.

[32] D. Talkin, "A robust algorithm for ptch tracking," in *Speech Coding and Synthesis*, W. B. Kleijn and K. K. Paliwal, Eds: Elseview Science B.V., 1995.

[33] J. C. Brown and B. Zhang, "Musical frequency tracking using the methods of conventional and "narrowed" autocorrelation," *J. Acoust. Soc. Amer.*, vol. 89, no. 5, pp. 2346–2354, 1991.

[34] A. P. Klapuri, "Qualitative and quantitative aspects in the design of periodicity estimation algorithms," in *Proc. European Signal Processing Conference*, Tampere, Finland, Sept. 2000.

[35] B. Doval and X. Rodet, "Estimation of fundamental frequency of musical sound signals," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing*, 1991, pp. 3657–3660.

[36] J. C. Brown, "Musical fundamental frequency tracking using a pattern recognition method," *J. Acoust. Soc. Amer.*, vol. 92, no. 3, pp. 1394–1402, 1992.

[37] M. Lahat, R. J. Niederjohn, and D. A. Krubsack, "A spectral autocorrelation method for measurement of the fundamental frequency of noise-corrupted speech," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-35, no. 6, pp. 741–750, 1987.

[38] N. Kunieda, T. Shimamura, and J. Suzuki, "Robust method of measurement of fundamental frequency by ACLOS—autocorrelation of log spectrum," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing*, 1996, pp. 232–235.

[39] A. J. M. Houtsma, "Pitch perception," in *Hearing—Handbook of Perception and Cognition*, B. J. C. Moore, Ed. San Diego, CA: Academic, 1995.

[40] N. F. Fletcher and T. D. Rossing, *The Physics of Musical Instruments*, 2nd ed. New York: Springer-Verlag, 1998.

[41] A. P. Klapuri, "Wide-band pitch estimation for natural sound sources with in harmonicities," in *Proc. 106th Audio Eng. Soc. Convention*, Munich, Germany, 1999.

[42] X. Rodet, "Musical sound signal analysis/synthesis: sinusoidal + residual and elementary waveform models," in *Proc. IEEE Time-Frequency and Time-Scale Workshop*, Coventry, U.K., Aug. 1997.

[43] A. P. Klapuri, "Multipitch estimation and sound separation by the spectral smoothness principle," in *Proc. IEEE Int. Conf. on Acoust., Speech, and Signal Processing*, 2001, pp. 3381–3384.

[44] D. Huron, "Voice denumerability in polyphonic music of homogeneous timbres," *Music Perception*, vol. 6, no. 4, pp. 361–382, Summer 1989.

[45] F. Opolko and J. Wapnick, *McGill University Master Samples* . Montreal, QC, Canada: McGill University, 1987.

[46] The University of Iowa Musical Instrument Samples*http://theremin.music.uiowa.edu/* [Online]

[47] IRCAM Studio Online*http://soleil.ircam.fr/* [Online]

[48] C. L. Krumhansl, *Cognitive Foundations of Musical Pitch*. New York: Oxford Univ. Press, 1990.

[49] A. de Cheveigné and H. Kawahara, "YIN, a fundamental frequency estimator for speech and music," *J. Acoust. Soc. Amer.*, vol. 111, no. 4, April 2002.

[50] Automatic Transcription of Music Demonstrations, A. P. Klapuri.*http://www.cs.tut.fil/~klap/iiro/* [Online]

Anssi P. Klapuri was born in Kälviä, Finland, in 1973. He received the M.Sc. degree in information technology from the Tampere University of Technology (TUT), Tampere, Finland, in June 1998. He is currently pursuing a postgraduate degree.

He has been with the TUT Institute of Signal Processing since 1996. His research interests include automatic transcription of music, audio content analysis, and signal processing.

Publication 6

A. P. Klapuri, A. J. Eronen, and J. T. Astola, “Automatic estimation of the meter of acoustic musical signals,” Tampere University of Technology, Institute of Signal Processing, Report 1–2004, Tampere, Finland, 2004.

A revised version of this report is to appear in *IEEE Trans. Speech and Audio Processing* as:
A.P. Klapuri, A. J. Eronen, and J. T. Astola, “Analysis of the meter of acoustic musical signals”

Automatic estimation of the meter of acoustic musical signals

Anssi P. Klapuri, Antti J. Eronen, Jaakko T. Astola
Tampere University of Technology, Institute of Signal Processing,
Korkeakoulunkatu 1, FIN-33100 Tampere, Finland
Tel:+358-50-364 8208, Fax: +358-3-365 4954, e-mail: {klap,eronen,jta}@cs.tut.fi

Abstract—A method is decribed which estimates the basic pattern of beats in a piece of music, the musical meter. Analysis is performed jointly at three different time scales: at the temporally atomic *tatum* pulse level, at the *tactus* pulse level which corresponds to the tempo of a piece, and at the musical *measure* level. Acoustic signals from arbitrary musical genres are considered. For the initial time-frequency analysis, a new technique is proposed which measures the degree of musical accent as a function of time at four different frequency ranges. This is followed by a bank of comb filter resonators which perform feature extraction for estimating the periods and phases of the three pulses. The features are processed by a probabilistic model which represents primitive musical knowledge and uses the low-level observations to perform joint estimation of the tatum, tactus, and measure pulses. The model takes into account the temporal dependencies between successive estimates and enables both causal and noncausal estimation. The method is validated using a manually annotated database of 474 musical signals from various genres. The method works robustly for different types of music and improves over two state-of-the-art reference methods in simulations.

Keywords—Acoustic signal analysis, music, musical meter estimation, music transcription.

I. Introduction

Meter analysis is an essential part of understanding music signals and an innate cognitive ability of humans even without musical education. Perceiving the meter can be characterized as a process of detecting moments of musical stress (accents) in an acoustic signal and filtering them so that underlying periodicities are discovered [1], [2]. For example, tapping foot to music indicates that a listener has abstracted metrical information about music and, based on that, is able to predict when the next beat will occur.

Musical meter is a hierarchical structure, consisting of pulse sensations at different levels (time scales). Here, three metrical levels are considered. The most prominent level is the *tactus*, often referred to as the foot tapping rate or the beat. Following the terminology of [1], we use the word *beat* to refer to the individual elements that make up a pulse. A musical meter can be illustrated as in Fig. 1, where the dots denote beats and each sequence of dots corresponds to a particular pulse level. By the *period* of a pulse we mean the time duration between successive beats and by *phase* the time when a beat occurs with respect to the beginning of the piece. The *tatum* pulse has its name stemming from "temporal atom" [3]. The period of this pulse corresponds to the shortest durational values in music that are still more than incidentally encountered. The other durational values, with few exceptions, are integer multiples of the tatum period and onsets of musical events occur approximately at a tatum beat. The *musical measure* pulse is typically related to the harmonic change rate or to the length of a rhythmic pattern. Although sometimes ambiguous, these three metrical levels are relatively well-defined and span the metrical hierarchy at the aurally most important levels. *Tempo* of a piece is defined as the rate of the tactus pulse. In order that a meter would make sense musically, the pulse periods must be slowly-varying and, moreover, each beat at the larger levels must coincide with a beat at all the smaller levels.

The concept *phenomenal accent* is important for meter analysis. Phenomenal accents are events that give emphasis to a moment in music. Among these are the beginnings of all discrete sound events, especially the onsets of long pitch events, sudden changes in loudness or timbre, and harmonic changes. Lerdahl and Jackendoff define the role of phenomenal accents in meter perception compactly by saying that *the moments of musical stress in the raw signal serve as cues from which the listener attempts to extrapolate a regular pattern* [1,p.17].

Automatic estimation of the meter has several applications. A temporal framework facilitates the cut-and-paste operations and editing of music signals. It enables synchronization with light effects, video, or electronic instruments, such as a drum machine. In a disc jockey application, metrical information can be used to mark the boundaries of a rhythmic loop or to synchronize two or more percussive audio tracks. Meter estimation for symbolic (MIDI[1]) data is required in time *quantization*, an indispensable subtask of score typesetting from keyboard input.

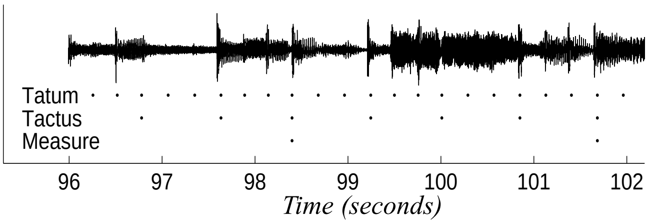

FIG. 1. A musical signal with three metrical levels illustrated.

[1] Musical Instrument Digital Interface. A standard interface for exchanging performance data and parameters between electronic musical devices.

A. Previous work

The work on automatic meter analysis originated from algorithmic models which tried to explain how a human listener arrives at a particular metrical interpretation of a piece, given that the meter is not explicitly spelled out in music [4]. The early models performed meter estimation for symbolic data, presented as an artificial impulse pattern or as a musical score [5], [6], [7], [8]. In brief, all these models can be seen as being based on a set of rules that are used to define musical accents and to infer the most natural meter. The rule system proposed by Lerdahl and Jackendoff in [1] is the most complete, but was described in verbal terms only. An extensive comparison of the early models has been given by Lee in [4], and later augmented by Desain and Honing in [9].

Parncutt has proposed a detailed and quantitative perceptual model based on systematic listening tests [10]. Brown performed metrical analysis of musical scores using an autocorrelation function where the notes were weighted according to their durations [11]. Large and Kolen associated meter perception with *resonance* and proposed an "entrainment" oscillator which adjusts its period and phase to an incoming pattern of impulses, located at the onsets of musical events [12].

Rosenthal aimed at emulating the human rhythm perception for realistic piano performances, presented as MIDI files [13]. Notable in his system was that other auditory organization functions were taken into account, too, by grouping notes into streams and chords. Rosenthal applied a set of rules to rank and prune competing meter hypotheses and conducted a beam search to track multiple hypotheses through time. The beam search strategy was originally proposed for pulse tracking by Allen and Dannenberg in [14].

Temperley has proposed a meter estimation algorithm for arbitrary MIDI files, based on implementing the preference rules verbally described in [1]. Dixon proposed a rule-based system to track the tactus pulse of expressive MIDI performances and introduced a simple onset detector to make the system applicable for audio signals [16]. The source codes of both Temperley's and Dixon's systems are publicly available.

Cemgil and Kappen have developed a probabilistic generative model for the timing deviations in expressive musical performances [24]. They used the model to infer a hidden continuous tempo variable and quantized ideal note onset times from observed noisy onset times in a MIDI file. Tempo tracking and time quantization were performed simultaneously so as to balance the smoothness of tempo deviations versus the complexity of the resulting quantized score. A similar probabilistic Bayesian model has been independently proposed by Raphael in [25].

Goto and Muraoka were the first to present a meter tracking system which works to a reasonable accuracy for audio signals [17], [18]. Only popular music was considered. The system operates in real time and is based on an architecture where multiple agents track alternative meter hypotheses. Beat positions at the larger levels were inferred by detecting certain drum sounds [17] or chord changes [18]. Gouyon et al. proposed a system for detecting the tatum pulse in percussive audio tracks with constant tempo [20]. Laroche used a straightforward probabilistic model to estimate the tempo and swing[1] of audio signals [26].

Scheirer proposed a method for tracking the tactus pulse of music signals of any kind, provided that they had a "strong beat" [22]. Important in Scheirer's approach was that he did not detect discrete onsets or sound events as a middle-step, but performed periodicity analysis directly on the half-wave rectified differentials of subband power envelopes. The source codes of Scheirer's system are publicly available. The meter estimator of Sethares and Staley resembles Scheirer's method, with the difference that a periodicity transform was used for periodicity analysis instead of a bank of comb filters [23].

In summary, most of the earlier work on meter estimation has concentrated on symbolic (MIDI) data and typically analyzed the tactus pulse only. Some of the systems ([12], [16], [24], [25]) can be immediately extended to process audio signals by employing an onset detector which extracts the beginnings of discrete acoustic events from an audio signal. Indeed, the authors of [16] and [25] have introduced an onset detector themselves. Elsewhere, onset detection methods have been proposed that are based on using subband energies [27], an auditory model [28], support vector machines [29], neural networks [30], independent component analysis [31], or complex-domain unpredictability [32]. However, if a meter estimator has been originally developed for symbolic data, the extended system is usually not robust to diverse acoustic material (e.g. classical vs. rock music) and cannot fully utilize the acoustic cues that indicate phenomenal accents in music signals.

There are a few basic problems that a meter estimator has to address to be successful. First, the degree of musical accentuation as a function of time has to be measured. In the case of audio input, this has much to do with the initial time-frequency analysis and is closely related to the problem of onset detection. Some systems measure accentuation in a continuous manner ([22], [23]), whereas others extract discrete events ([17], [20], [26]). Secondly, the periods and phases of the underlying metrical pulses have to be estimated. The methods which detect discrete events as a middle step have often used inter-onset interval (IOI) histograms for this purpose [16], [17], [18], [20]. Thirdly, a system has to choose the metrical level which corresponds to the tactus or some other specially designated pulse level. This may take place implicitly, or using a prior distribution for pulse periods [10] or rhythmic pattern matching [17]. Tempo halving or doubling is a symptom of failing to do this.

B. Proposed method

The aim of this paper is to develop a method for estimating the meter of acoustic musical signals at the tactus, tatum, and measure pulse levels. The target signals are not limited to any particular music type but all the main genres, including classical music, are represented in the validation database. The particular motivation for the present work is to utilize metrical information in further signal analysis and classification, more

[1] *Swing* is a characteristic of musical rhythms most commonly found in jazz. Swing is defined in [26] as a systematic slight delay of the second and fourth quarter-beats.

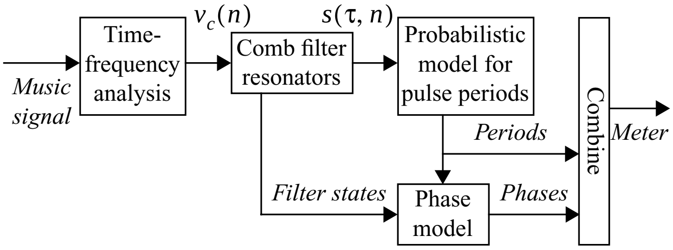

FIG. 2. Overview of the meter estimation method. The two intermediate data representations are registral accent signals $v_c(n)$ and metrical pulse strengths $s(\tau, n)$.

exactly, in music transcription [33], [34].

An overview of the method is shown in Fig 2. For the time-frequency analysis part, a new technique is proposed which aims at measuring the degree of accentuation in acoustic signals. The technique is robust to diverse acoustic material and can be seen as a synthesis and generalization of two earlier state-of-the-art methods [17] and [22]. Feature extraction for the pulse period and phase analysis is performed using comb filter resonators very similar to those used by Scheirer in [22]. This is followed by a probabilistic model where the period-lengths of the tactus, tatum, and measure pulses are jointly estimated and temporal continuity of the estimates is modelled. At each time instant, the periods of the pulses are estimated first and act as inputs to the phase model. The probabilistic models encode prior musical knowledge and lead to a more reliable and temporally stable meter tracking. Both causal and non-causal algorithms are presented.

This paper is organized as follows. Section II will describe the different elements of the system presented in Figure 2. Section III will present experimental results and compare the proposed method with two reference systems. Finally, Section IV will summarize the main conclusions and discuss future work.

II. Meter Analysis Model

This section will describe the different parts of the meter estimation method illustrated in Figure 2. Subsection A will describe the time-frequency analysis part which produces a measure of musical accent as a function of time. In Subsection B, the comb filter resonators will be introduced. Finally, Subsections C and D will describe the probabilistic models which are used to estimate the periods and phases of the three pulse levels.

A. Calculation of registral accent signals

All the phenomenal accent types mentioned in Introduction can be observed in the time-frequency representation of a signal. Although an analysis using a model of the human auditory system would be theoretically better, we did not manage to obtain a performance advantage using a model similar to [28] and [35]. Also, the computational complexity of such models makes them rather impractical.

In a time-frequency plane representation, some data reduction must take place to discard information which is irrelevant for meter analysis. A big step forward in this respect was taken by Scheirer who demonstrated that the perceived rhythmic content of many music types remains the same if only the subband power envelopes are preserved and then used to modulate a white noise signal [22]. A number of approximately five subbands was reported to suffice. Scheirer proposed a method where periodicity analysis was carried out at subbands and the results were then combined across bands.

Although Scheirer's method was indeed very successful, a problem with it is that it applies primarily to music with a "strong beat". Harmonic changes in e.g. classical or vocal music go easily unnoticed using only few subband envelopes. To detect harmonic changes or note beginnings in *legato*[1] passages, approximately 40 logarithmically-distributed subbands would be needed[2]. However, this leads to a dilemma: the resolution is sufficient to distinguish harmonic changes but measuring periodicity at each narrow subband separately is no more appropriate. The power envelopes of individual narrow bands are not guaranteed to reveal the correct metrical periods, or even to show periodicity at all, because individual events may occupy different frequency bands.

To overcome the above problem, consider another state-of-the-art system, that of Goto and Muraoka [17]. They detect narrowband frequency components and sum their power differentials across prefedefined frequency ranges *before* onset detection and periodicity analysis takes place. This has the advantage that harmonic changes are detected, yet periodicity analysis takes place at wider bands.

There is a continuum between the above two approaches. The tradeoff is: how many adjacent subbands are combined before the periodicity analysis and how many at the later stage when the bandwise periodicity analysis results are combined. In the following, we propose a method which can be seen as a synthesis of the approaches of Scheirer and Goto *et al.*

Acoustic input signals are sampled at 44.1 kHz rate and 16-bit resolution and then normalized to have zero mean and unity variance. Discrete Fourier transforms are calculated in successive 23 ms time frames which are Hanning-windowed and overlap 50 %. In each frame, 36 triangular-response bandpass filters are simulated. The filters are uniformly distributed on the equivalent-rectangular bandwidth critical-band scale between 50 Hz and 20 kHz [36,p.176]. The power at each band is calculated and stored to $x_b(k)$, where k is the frame index and band index $b = 1, 2, \ldots, b_0$, with $b_0 = 36$. The exact number of subbands is not critical.

There are many potential ways of measuring the degree of change in the power envelopes at critical bands. For humans, the smallest detectable change in intensity, ΔI, is approximately proportional to the intensity I of the signal, the same amount of increase being more prominent in a quiet signal. That is, $\Delta I / I$, the Weber fraction, is perceptually approximately constant [37,p.134]. This relationship holds for intensities from about 20 dB to about 100 dB above the absolute threshold. Thus it is reasonable to normalize the differential of

[1] A smooth and connected style of playing in which no perceptible gaps are left between notes.

[2] In this case, the center frequencies are approximately one *whole tone* apart, which is the distance between e.g. the notes *c* and *d*.

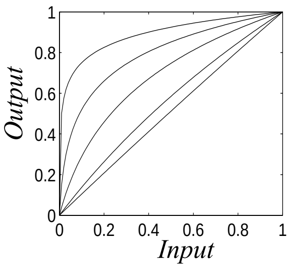

FIG. 3. The μ-law compression with μ getting values 0.1, 1, 10, 10^2, and 10^4, starting from the lowest curve.

power with power. This leads to $[(d/dt)x_b(k)]/x_b(k)$, which is equal to $(d/dt)\ln[x_b(k)]$. This measures spectral change and can be seen as an approximation of the differential of *loudness*, since the perception of loudness for steady sounds is rougly proportional to the sum of the log-powers at critical bands.

The logarithm and differentiation operations are both represented in a more flexible form. A numerically robust way of calculating the logarithm is the μ-law compression

$$y_b(k) = \frac{\ln[1 + \mu x_b(k)]}{\ln(1 + \mu)}, \qquad (1)$$

which performs a nonlinear mapping of $x_b(k)$ values between zero and one to values of $y_b(k)$ between zero and one. The constant μ can be used to compromize between a close-to-linear ($\mu < 0.1$) and a close-to-logarithmic ($\mu > 1000$) transformation, as illustrated in Fig 3. The value $\mu = 100$ was employed, but any value in the range [10, 10^6] would be valid.

To achieve a better time resolution, the compressed power envelopes $y_b(k)$ are interpolated by factor two by adding zeros between the samples. This leads to the sampling rate $f_r = 172$ Hz. A sixth-order Butterworth lowpass filter with $f_{LP} = 10$ Hz cutoff frequency is then applied to smooth the compressed and interpolated power envelopes. The resulting smoothed signal is denoted by $z_b(n)$.

Differentiation of $z_b(n)$ is performed as follows. First, a half-wave rectified differential of $z_b(n)$ is calculated as

$$z_b'(n) = \mathrm{HWR}\{z_b(n) - z_b(n-1)\} \qquad (2)$$

where HWR maps negative values to zero and is essential to make the differentiation useful. Then a weighted average of $z_b(n)$ and its differential $z_b'(n)$ is formed as

$$u_b(n) = (1-\lambda)z_b(n) + \lambda(f_r/f_{LP})z_b'(n) \qquad (3)$$

where $\lambda = 0.8$ and the factor f_r/f_{LP} roughly compensates for the fact that the differential of a lowpass-filtered signal is small in amplitude. Using the value $\lambda = 0.8$ instead of 1.0 has a slight but consistent positive impact on the performance of the overall system.

Figure 4 illustrates the described dynamic compression and weighted differentiation steps for an artificial subband-power signal $x_b(k)$. Although the present work is motivated purely from the practical application point of view, it is interesting to note that the graphs in Fig. 4 bear considerable resemblance to the response of Meddis's auditory-nerve model to acoustic stimulation [38].

Finally, each m_0 adjacent bands are linearly summed to get $c_0 = \lceil b_0/m_0 \rceil$ signals which we call "registral accent signals"

$$v_c(n) = \sum_{b=(c-1)m_0+1}^{cm_0} u_b(n) \qquad , c = 1, \ldots, c_0. \qquad (4)$$

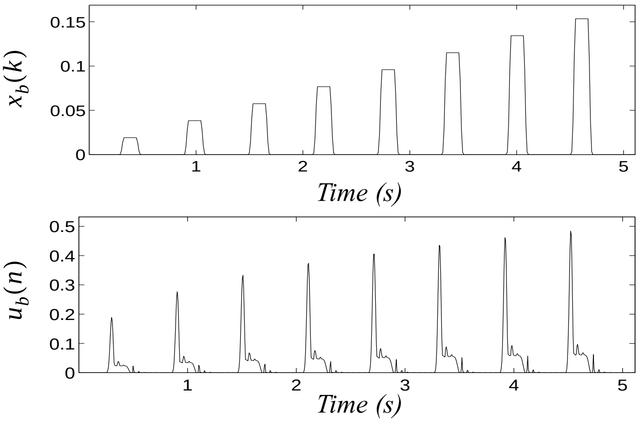

FIG. 4. Illustration of the dynamic compression and weighted differentiation steps for an artificial signal. Upper panel shows $x_b(k)$ and the lower panel shows $u_b(n)$.

The registral accent signals $v_c(n)$ serve as a middle-level representation for musical meter estimation. They represent the degree of musical accent as a function of time at frequency channels *c*. We use $b_0 = 36$ and $m_0 = 9$, leading to $c_0 = 4$.

It should be noted that combining each m_0 adjacent bands at this stage is not primarily an issue of computational complexity, but improves the analysis accuracy. A prototypical meter estimation system was used to thoroughly investigate the effect of different values of m_0. Surprisingly, it turned out that neither of the extreme values $m_0 = b_0$ or $m_0 = 1$ is optimal, but using a large number of initial channels, $b_0 > 20$, and three or four registral channels c_0 leads to the most reliable meter estimation. Other system parameters were re-estimated in each case to ensure that this was not merely a symptom of parameter couplings.

The presented form of calculating the registral accent signals is very flexible when varying μ, λ, b_0, and m_0. A representation similar to that used by Scheirer in [22] is obtained by setting μ=0.1, λ=1*,* b_0=6, m_0=1. A representation roughly similar to that used by Goto in [17] is obtained by setting μ=0.1, λ=1*,* b_0=36, m_0=6. In the following, fixed values μ=100, λ=0.8*,* b_0=36, m_0=9 are used.

B. Bank of comb filter resonators

Periodicity of the registral accent signals $v_c(n)$ is analyzed to estimate the *salience* (weight) of different metrical pulse period candidates. This resembles the idea of "registral IOI"[1] computation for MIDI data in [15]. Four different period estimation algorithms were evaluated. "Enhanced" autocorrelation, enhanced *YIN* method of de Cheveigné and Kawahara [39], different types of comb-filter resonators [22], and banks

[1] In [15], registral IOI is defined as the time-interval between events which are within a certain range of pitch. Registral IOI was considered as a factor of musical accentuation.

of phase-locking resonators [12]. Here *enhancing* refers to a postprocessing step which is not needed in the final method and thus not explained.

As an important observation, three of the four period estimation methods performed equally well after a thorough optimization. This suggests that the key problems in meter estimation are in measuring phenomenal accentuation and in modeling higher-level musical knowledge, not in finding exactly the correct period estimator. The period estimation method presented in the following was selected because it is the least complex among the three best-performing algorithms and because it has been earlier used in [22].

Using a bank of comb-filter resonators with a constant half-time has been originally proposed for tactus tracking by Scheirer [22]. Comb filters have an exponentially-decaying impulse response where the *half-time* refers to the delay during which the response decays to a half of its initial value. Output of a comb filter with delay τ is given for input $v_c(n)$ as

$$r_c(\tau, n) = \alpha_\tau r_c(\tau, n-\tau) + (1-\alpha_\tau)v_c(n) \tag{5}$$

where the feedback gain $\alpha_\tau = 0.5^{\tau/T_0}$ is calculated based on a selected half-time T_0. We used $T_0 = 3f_r$, i.e., a halftime of three seconds which is short enough to react to tempo changes but long enough to reliably estimate pulse-periods of up to four seconds in length. Scheirer used halftimes of 1.5–2.0 seconds but did not attempt to track the measure pulse.

The comb filters implement a frequency response where frequencies kf_r/τ, $k = 0, \ldots, \lfloor \tau/2 \rfloor$ have a unity response and the maximum attenuation between the peaks is $[(1-\alpha_\tau)/(1+\alpha_\tau)]^2$. Overall power $\gamma(\alpha_\tau)$ of a comb filter with feedback gain α_τ can be calculated by integrating over the squared impulse response, which yields

$$\gamma(\alpha_\tau) = (1-\alpha_\tau)^2/(1-\alpha_\tau^2). \tag{6}$$

A bank of such resonators was applied, with τ getting values from 1 to τ_{max}, where $\tau_{max} = 688$ corresponds to four seconds. Computational complexity of one resonator is $O(1)$ per each input sample, and the overall resonator filterbank requires of the order $c_0 f_r \tau_{max}$ operations per second, which is not computationally too demanding for a real-time application.

Instantaneous energies $\hat{r}_c(\tau, n)$ of each comb filter in channel *c* at time *n* are calculated as

$$\hat{r}_c(\tau, n) = \frac{1}{\tau}\sum_{i=n-\tau+1}^{n}[r_c(\tau, i)]^2. \tag{7}$$

These are then normalized to obtain

$$s_c(\tau, n) = \frac{1}{1-\gamma(\alpha_\tau)}\left[\frac{\hat{r}_c(\tau, n)}{\hat{v}_c(n)} - \gamma(\alpha_\tau)\right], \tag{8}$$

where $\hat{v}_c(n)$ is the energy of the registral accent signal $v_c(n)$, calculated by squaring $v_c(n)$ and by applying a leaky integrator, i.e., a resonator which has τ=1 and the same half-time as the other resonators. Normalization with $\gamma(\alpha_\tau)$ is applied to compensate for the differences in the overall power responses for different α_τ. The proposed normalization is advantageous because it preserves a unity response at the peak frequencies and at the same time removes the τ-dependent trend for a white-noise input.

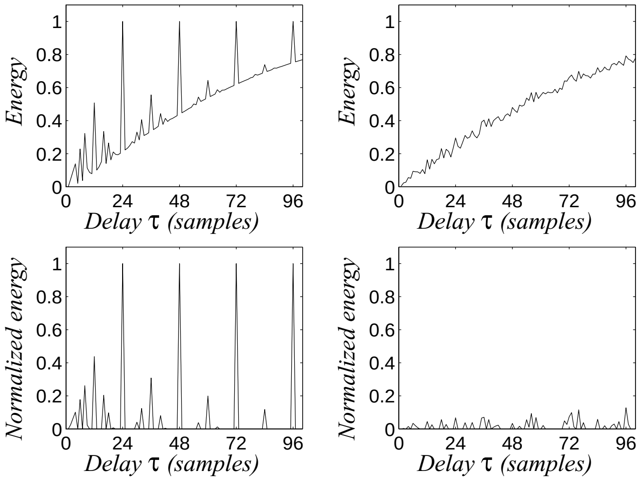

FIG. 5. Resonator energies for an impulse train with a period-length of 24 samples (left) and for white noise (right). Upper panels show the energies $\hat{r}_c(\tau, n)$ and the lower panels normalized energies $s_c(\tau, n)$.

Figure 5 shows the resonator energies $\hat{r}_c(\tau, n)/\hat{v}_c(n)$ and the normalized energies $s_c(\tau, n)$ for two types of artificial input $v_c(n)$, an impulse train and a white-noise signal. It is important to notice that all resonators that are in rational-number relations to the period of the impulse train (24 samples) show response to it. In the case of an autocorrelation function, for example, only integer multiples of 24 come up and an explicit postprocessing step ("enhancing") was necessary to generate responses to the subharmonic lags and to achieve the same meter estimation performance. This step is not needed for the comb filter resonators.

Finally, a function $s(\tau, n)$ which represents the overall saliences of different metrical pulses at time *n* is obtained as

$$s(\tau, n) = \sum_{c=1}^{c_0} s_c(\tau, n). \tag{9}$$

This function acts as the *observation* for the probabilistic model that estimates the pulse periods.

For tatum period estimation, the discrete power spectrum $S(f, n)$ of $s(\tau, n)$ is calculated as

$$S(f, n) = f\left|\frac{1}{\tau_{max}}\sum_{\tau=1}^{\tau_{max}}[s(\tau, n)\zeta(\tau)e^{-i2\pi f(\tau-1)/\tau_{max}}]\right|^2 \tag{10}$$

where the emphasis with *f* removes spectral trend and the window function $\zeta(\tau)$ is half-Hanning

$$\zeta(\tau) = 0.5\{1-\cos[\pi(\tau_{max}+\tau-1)/\tau_{max}]\}. \tag{11}$$

Frequencies above 20 Hz can be discarded from $S(f, n)$. The rationale behind calculating the discrete Fourier transform (DFT) in (10) is that, by definition, other pulse periods are integer multiples of the tatum period. Thus the overall function $s(\tau, n)$ contains information about the tatum and this is conveniently gathered for each tatum frequency candidate *f* using the DFT as in (10). Gouyon et al. used an IOI histogram and Maher's two-way mismatch procedure for the same purpose [20], [21]. Their idea was to find a tatum period which best explains the harmonically-related peaks in the histogram.

It should be noted that the observation $s(\tau, n)$ and its spectrum $S(f, n)$ are zero-phase, meaning that the *phases* of the pulses at different metrical levels have to be estimated using some other source of information. As will be discussed in Subsection D, the phases are estimated based on the states of the comb filters, after the periods have been solved first.

C. Probabilistic model for pulse periods

Period-lengths of the metrical pulses can be estimated independently of their phases and it is reasonable to compute the phase only for the few winning periods. Thus the proposed method finds periods first and then the phases (see Fig. 2). Although estimating the phases is not trivial, the search problem is largely completed when the period-lengths have been found.

Musical meter cannot be assumed to be static over the duration of a piece. It has to be estimated causally at successive time instants and there must be some temporal tying between the successive estimates. Also, the dependencies between different metrical pulse levels have to be taken into account. This requires prior musical knowledge which is encoded in the probabilistic model to be presented.

For period estimation, a hidden Markov model that describes the simultaneous evolution of four processes is constructed. The observable variable is the vector of instantaneous energies of the resonators, $s(\tau, n)$, denoted $\boldsymbol{s}_n$ in the following. The unobservable processes are the tatum, tactus, and measure periods. The corresponding hidden variables are the tatum period τ_n^A, tactus period τ_n^B, and measure period τ_n^C. As a mnemonic for this notation, recall that the tatum is the temporally atomic (*A*) pulse level, tactus pulse is often called “beat” (*B*), and musical measure pulse is related to the harmonic (i.e., chord) change rate (*C*). For convenience, we use $\boldsymbol{q}_n = [j, k, l]$ to denote a “meter state”, equivalent to $\tau_n^A = j$, $\tau_n^B = k$, and $\tau_n^C = l$. The hidden state process is a time-homogenous first-order Markov which has an initial state distribution $P(\boldsymbol{q}_1)$ and transition probabilities $P(\boldsymbol{q}_n|\boldsymbol{q}_{n-1})$. The observable variable is conditioned only on the current state, i.e., we have the state-conditional observation densities $p(\boldsymbol{s}_n|\boldsymbol{q}_n)$.

The joint probability density of a state sequence $Q = (\boldsymbol{q}_1\boldsymbol{q}_2\ldots\boldsymbol{q}_N)$ and observation sequence $O = (\boldsymbol{s}_1\boldsymbol{s}_2\ldots\boldsymbol{s}_N)$ can be written as

$$p(Q, O) = P(\boldsymbol{q}_1)p(\boldsymbol{s}_1|\boldsymbol{q}_1)\prod_{n=2}^{N} P(\boldsymbol{q}_n|\boldsymbol{q}_{n-1})p(\boldsymbol{s}_n|\boldsymbol{q}_n), \quad (12)$$

where the term $P(\boldsymbol{q}_n|\boldsymbol{q}_{n-1})$ can be decomposed as

$$\begin{aligned} &P(\boldsymbol{q}_n|\boldsymbol{q}_{n-1}) \\ &\quad = P(\tau_n^B|\boldsymbol{q}_{n-1})P(\tau_n^A|\tau_n^B, \boldsymbol{q}_{n-1})P(\tau_n^C|\tau_n^B, \tau_n^A, \boldsymbol{q}_{n-1}) \end{aligned} \quad . \quad (13)$$

It is reasonable to assume that

$$P(\tau_n^C|\tau_n^B, \tau_n^A, \boldsymbol{q}_{n-1}) = P(\tau_n^C|\tau_n^B, \boldsymbol{q}_{n-1}), \quad (14)$$

i.e., given the tactus period, the tatum period does not give additional information regarding the measure period. We further assume that given $\tau_{n-1}^{(i)}$, the other two hidden variables at time $n-1$ give no additional information regarding $\tau_n^{(i)}$. Here $i \in \{A, B, C\}$. It follows that (13) can be written as

$$\begin{aligned} &P(\boldsymbol{q}_n|\boldsymbol{q}_{n-1}) \\ &\quad = P(\tau_n^B|\tau_{n-1}^B)P(\tau_n^A|\tau_n^B, \tau_{n-1}^A)P(\tau_n^C|\tau_n^B, \tau_{n-1}^C) \end{aligned} \quad . \quad (15)$$

Using the same assumptions, $P(\boldsymbol{q}_1)$ is decomposed and simplified as

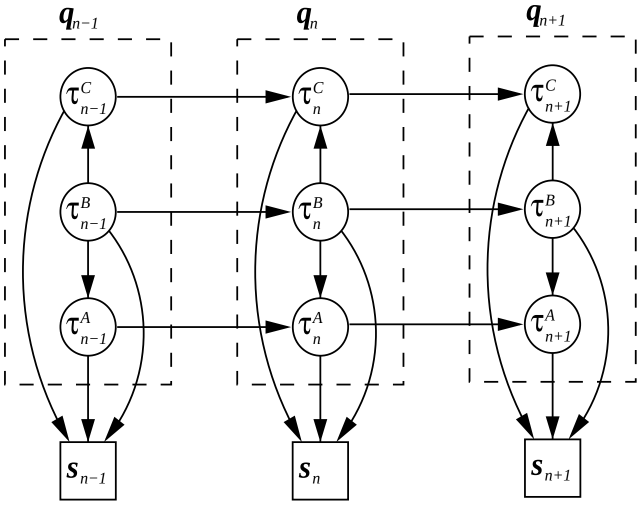

FIG. 6. Hidden markov model for the temporal evolution of the tatum, beat, and measure pulse periods.

$$P(\boldsymbol{q}_1) = P(\tau_1^B)P(\tau_1^A|\tau_1^B)P(\tau_1^C|\tau_1^B). \quad (16)$$

The described modeling assumptions lead to a structure which is represented as a directed acyclic graph in Figure 6. The arrays in the graph represent conditional dependencies between the variables. The circles denote hidden variables and the observed variable is marked with boxes. The tactus pulse has a central role in meter perception and it is not by chance that the other two variables are drawn to depend on it. The assumption in (14) is not valid if the variables are permuted.

1. Estimation of the state-conditional observation likelihoods

The remaining problem is to find reasonable estimates for the model parameters, i.e., for the probabilities that appear in (12)-(16). In the following, we ignore the time indeces for a while for simplicity. The state-conditional observation likelihoods $p(\boldsymbol{s}|\boldsymbol{q})$ are estimated from a database of musical recordings where the musical meter has been hand-labeled. However, the data is very limited in size compared to the number of parameters to be estimated. Estimation of the state densities for each different $\boldsymbol{q} = [j, k, l]$ is impossible since each of the three discrete hidden variables can take on several hundreds of different values. By making a series of assumptions we arrive at the following approximation for $p(\boldsymbol{s}|\boldsymbol{q})$:

$$p(\boldsymbol{s}|\boldsymbol{q} = [j, k, l]) \propto s(k)s(l)S(1/j). \quad (17)$$

Appendix A presents the derivation of (17) and the underlying assumptions in detail. An intuitive rationale of (17) is that a truly existing tactus or measure pulse appears as a peak in $s(\tau)$ at the lag that corresponds to the pulse period. Analogously, the tatum period appears as a peak in $S(f)$ at the frequency that corresponds to the inverse of the period. The product of these three values correlates approximately linearly with the likelihood of the observation given the meter.

2. Estimation of the transition and initial probabilities

In (15), the term $P(\tau_n^A|\tau_n^B, \tau_{n-1}^A)$ can be decomposed as

$$P(\tau_n^A|\tau_n^B, \tau_{n-1}^A) = P(\tau_n^A|\tau_{n-1}^A)\frac{P(\tau_n^A, \tau_n^B|\tau_{n-1}^A)}{P(\tau_n^A|\tau_{n-1}^A)P(\tau_n^B|\tau_{n-1}^A)}, \quad (18)$$

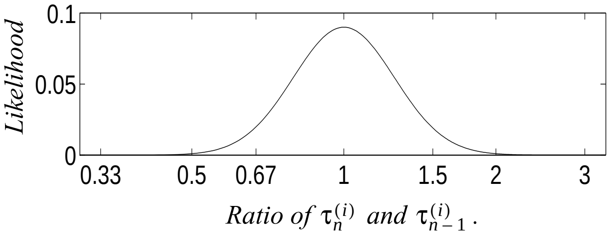

FIG. 7. The likelihood function $f(\tau_n^{(i)}/\tau_{n-1}^{(i)})$ which describes the tendency that the periods are slowly-varying.

where the first term represents transition probabilities between successive period estimates and the second term represents the relation dependencies of simultaneous periods (τ_n^A and τ_n^B), independent of their actual frequencies of occurrence (in practice τ_n^B tends to be integer multiple of τ_n^A). Similarly, we write

$$P(\tau_n^C|\tau_n^B,\tau_{n-1}^C) = P(\tau_n^C|\tau_{n-1}^C)\frac{P(\tau_n^C,\tau_n^B|\tau_{n-1}^C)}{P(\tau_n^C|\tau_{n-1}^C)P(\tau_n^B|\tau_{n-1}^C)}. \quad (19)$$

The transition probabilities $P(\tau_n^{(i)}|\tau_{n-1}^{(i)})$ between successive period estimates are obtained as follows. Again, the number of possible transitions is too large for any reasonable estimates to be obtained by counting occurrences. The transition probability is modeled as a product of the prior probability for a certain period, $P(\tau_1^{(i)})$, and a term $f(\tau_n^{(i)}/\tau_{n-1}^{(i)})$ which describes the tendency that the periods are slowly-varying:

$$P(\tau_n^{(i)}|\tau_{n-1}^{(i)}) = P(\tau_n^{(i)})\frac{P(\tau_n^{(i)},\tau_{n-1}^{(i)})}{P(\tau_n^{(i)})P(\tau_{n-1}^{(i)})} \approx P(\tau_1^{(i)})f\left(\frac{\tau_n^{(i)}}{\tau_{n-1}^{(i)}}\right),(20)$$

where $i \in \{A, B, C\}$. The function f,

$$f\left(\frac{\tau_n^{(i)}}{\tau_{n-1}^{(i)}}\right) = \frac{1}{\sigma_1\sqrt{2\pi}}\exp\left\{-\frac{1}{2\sigma_1^2}\left[\ln\left(\frac{\tau_n^{(i)}}{\tau_{n-1}^{(i)}}\right)\right]^2\right\}, \qquad (21)$$

implements a normal distribution as a function of the logarithm of the ratio of successive period values. It follows that the likelihood of large changes in period is higher for long periods, and that period doubling and halving are equally probable. The parameter $\sigma_1 = 0.2$ was found by monitoring the performance of the system in simulations. The distribution (21) is illustrated in Fig. 7.

Prior probabilities for tactus period lengths, $P(\tau^B)$, have been measured from actual data by several authors [10], [40]. As suggested by Parncutt in [10], we apply the two-parameter lognormal distribution to model the prior densities:

$$p(\tau^{(i)}) = \frac{1}{\tau^{(i)}\sigma^{(i)}\sqrt{2\pi}}\exp\left\{-\frac{1}{2(\sigma^{(i)})^2}\left[\ln\left(\frac{\tau^{(i)}}{m^{(i)}}\right)\right]^2\right\}, \qquad (22)$$

where $m^{(i)}$ and $\sigma^{(i)}$ are the scale and shape parameters, respectively. For the tactus period, the values $m^B = 0.55$ and $\sigma^B = 0.28$ were estimated by counting the occurrences of different period lengths in our hand-labeled database (see Sec. III) and by fitting the log-normal distribution to the histogram data. Figure 8 shows the period-length histograms and the corresponding lognormal distributions for the tactus, measure, and tatum periods. The scale and shape parameters for the tatum and measure periods are $m^A = 0.18$, $\sigma^A = 0.39$, $m^C = 2.1$, and $\sigma^C = 0.26$, respectively. These were estimated from the hand-labeled data in the same way.

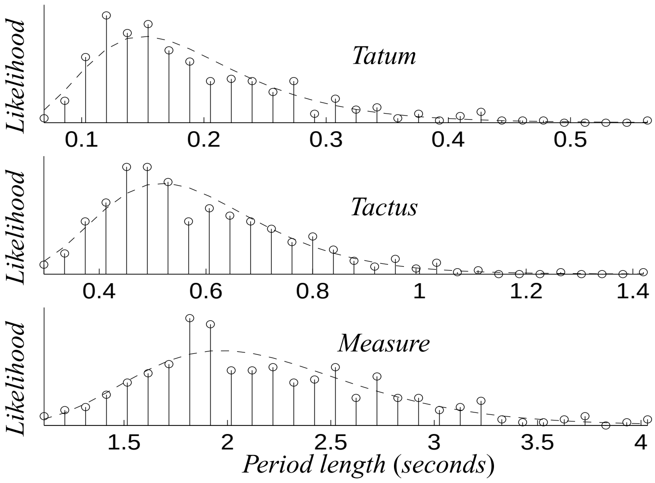

FIG. 8. Period-length histograms and the corresponding lognormal distributions for tatum, tactus, and measure pulses.

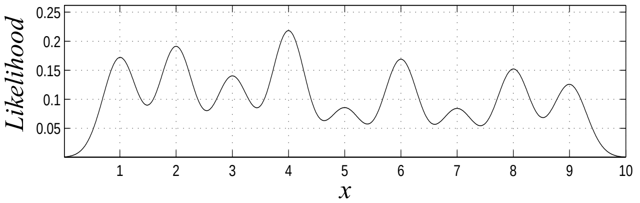

FIG. 9. Distribution $g(x)$ which models the relation dependencies of simultaneous periods (see (25)).

The relation dependencies of simultaneous periods are modeled as follows. We model the latter terms in (18)–(19) as

$$\frac{P(\tau_n^A,\tau_n^B|\tau_{n-1}^A)}{P(\tau_n^A|\tau_{n-1}^A)P(\tau_n^B|\tau_{n-1}^A)} \approx g\left(\frac{\tau^B}{\tau^A}\right), \qquad (23)$$

$$\frac{P(\tau_n^C,\tau_n^B|\tau_{n-1}^C)}{P(\tau_n^C|\tau_{n-1}^C)P(\tau_n^B|\tau_{n-1}^C)} \approx g\left(\frac{\tau^C}{\tau^B}\right), \qquad (24)$$

where $g(x)$ is a Gaussian mixture density of the form

$$g(x) = \sum_{l=1}^{9} w_l N(x;l,\sigma_2), \qquad (25)$$

where w_l are the component weights and sum to unity, l are the component means, and $\sigma_2 = 0.3$ is the common variance. The function models the relation dependencies of simultaneous periods, independent of their actual frequencies of occurrence. The exact weight values are not critical, but are designed to realize a tendency towards binary or ternary integer relationships between concurrent pulses. For example, it is quite probable that one tactus period consists of two, four, or six tatum periods, but multiples five and seven are much less likely in music and thus have lower weights. The distribution $g(x)$ is shown in Fig. 9. The weights were obtained by first assigning them values according to a musical intuition. Then the dynamic range of the weights was found by raising them to a common power which was varied between 0.1 and 10. The value which performed best in small-scale simulations was selected. Finally, small adjustments to the values were made.

3. *Finding the optimal sequence of period estimates*

Now we must obtain an estimate for the unobserved state variables given the observed front-end data and the model parameters. We do this by finding the most likely sequence of state variables $Q = (\boldsymbol{q}_1\boldsymbol{q}_2\ldots\boldsymbol{q}_N)$ given the observed front-end data $O = (\boldsymbol{s}_1\boldsymbol{s}_2\ldots\boldsymbol{s}_N)$. This can be straighforwardly computed using the Viterbi algorithm widely applied in speech recognition [41]. Thus, we seek the sequence of period estimates,

$$\hat{Q} = \underset{Q}{\operatorname{argmax}}[p(Q, O)] \tag{26}$$

where $p(Q, O)$ denotes the joint probability density of the hidden and observed variables, as defined in (12).

For the Viterbi-decoding, we need to define the quantity

$$\delta_n(j, k, l) = \max_{\boldsymbol{q}_1\cdots\boldsymbol{q}_{n-1}} P(\boldsymbol{q}_1\ldots\boldsymbol{q}_{n-1}, \boldsymbol{q}_n = [j, k, l], \boldsymbol{s}_1\ldots\boldsymbol{s}_n)\text{ ,}(27)$$

which is the best score along a single path at time n, which takes into account the first n observations and ends in state $\boldsymbol{q}_n = [j, k, l]$. By induction we then compute

$$\delta_{n+1}(j, k, l) = p(\boldsymbol{s}_{n+1}|\boldsymbol{q}_{n+1} = [j, k, l]) \tag{28}$$

$$\cdot \max_{j', k', l'}[\delta_n(j', k', l')P(\boldsymbol{q}_{n+1} = [j, k, l]|\boldsymbol{q}_n = [j', k', l'])]$$

In a causal model, the meter estimate $\boldsymbol{q}_n$ at time n is determined according to the end-state of the best partial path at that point in time. A noncausal estimate after seeing a complete sequence of observations can be computed using backward decoding, and

$$\max_{j, k, l}[\delta_n(j, k, l)] \approx \max_{Q}[p(Q, O)]\text{ .} \tag{29}$$

The inequality is due to "pruning" some of the path candidates by evaluating only a subset of best path candidates at each time instant, and thus the resulting path is not necessarily the global optimum. However, in practice the difference is small. Evaluating all the possible path candidates would be computationally very demanding. Therefore, we apply a suboptimal beam-search strategy, and evaluate only a predefined number of the most promising path candidates at each time instant. The selection of the most promising candidates is made using a greedy selection strategy. Once in a second, we select independently K best candidates for the tatum, tactus, and measure periods. The number of candidates $K = 5$ was found to be safe and was used in simulations. The selection is made by maximizing $p(\tau_n^{(i)})P(\boldsymbol{s}_n|\tau_n^{(i)})$ for $i = \{A, B, C\}$. After selecting the best candidates for each, we need only to compute the observation likelihoods for $K^3 = 125$ meter candidates, i.e., for the different combinations of the tatum, tactus, and measure periods. This is done according to Eq (17) and the results are stored into a data vector. The transition probabilities are computed using Eq. (15) and stored into a 125-by-125 matrix. These data structures are then used in the Viterbi algorithm.

D. *Phase estimation*

The phases of the three pulses are estimated at successive time instants, after the periods have been decided at these points. We use $\hat{\tau}_n^{(i)}$, $i \in \{A, B, C\}$ to refer to the estimated periods of the tatum, tactus, and measure pulses at time n, respectively. The corresponding phases of the three pulses, $\varphi_n^{(i)}$, are expressed as "temporal anchors", i.e., time values when the nearest beat unit occurs with respect to the beginning of a piece. Periods and phases, $\tau_n^{(i)}$ and $\varphi_n^{(i)}$, completely define the meter at time n.

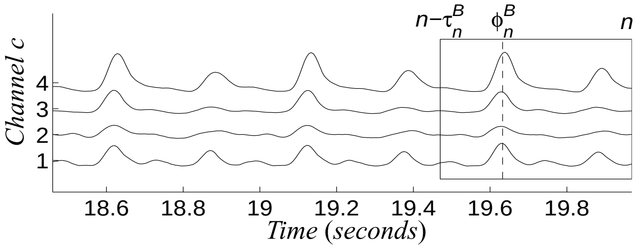

FIG. 10. The rectangle indicates the observation matrix R_n^B for tactus phase estimation at time n (here period τ_n^B is 0.51 s.). Dashed line shows the correct phase in this case.

In principle, the phase of the measure pulse, φ_n^C, determines the phase of all the three levels. This is because in a well-formed meter, each measure-level beat must coincide with a beat at all the lower metrical levels. However, determining the phase of the measure pulse is difficult and turned out to require pattern recognition techniques, whereas tactus phase estimation is more straightforward and robust. We therefore propose a model where the tactus and measure phases are estimated separately using two parallel models. For the tatum pulse, phase estimation is not needed but the tactus phase can be used.

Scheirer has proposed using the state vectors of comb filters to determine the phase of the tactus pulse [22]. This is equivalent to using the latest τ outputs of a resonator with delay τ. We have resonators at several channels c and, consequently, an output matrix $r_c(\tau, j)$ where $c = 1, 2, \ldots, c_0$ is the channel index and the phase index j takes on values between $n - \tau + 1$ and n when estimation is taking place at time n. For convenience, we use $R_n^{(i)}$ to denote the output matrix $r_c(\hat{\tau}_n^{(i)}, j)$ of a found pulse period $\hat{\tau}_n^{(i)}$ and the notation $(R_n^{(i)})_{c,j}$ to refer to the individual elements of $R_n^{(i)}$. The matrix $R_n^{(i)}$ acts as the observation for phase estimation at time n.

Fig. 10 shows an example of the observation matrix R_n^B when tactus phase estimation is taking place 20 seconds after the beginning of a piece. The four signals at different channels are the outputs of the comb filter which corresponds to the estimated tactus period $\hat{\tau}_n^B = 0.51$ seconds. The output matrix R_n^B contains the latest 0.51 seconds of the output signals, as indicated with the rectangle. The correct phase value φ_n^B has been marked with a dashed line. As discussed in Sec. II.B, comb filters implement a "harmonic" frequency response and therefore the outputs show clear periodicity with period $\hat{\tau}_n^B$.

Two separate hidden Markov models are evaluated in parallel, one for the tactus phase and another for the measure phase. No joint estimation is attempted. The two models are very similar and differ only in how the state-conditional observation densities are defined. In both models, the observable variable is the output matrix $R_n^{(i)}$ of the resonator $\hat{\tau}_n^{(i)}$ which corresponds to the found pulse period. The hidden variable is the phase of the pulse, $\varphi_n^{(i)}$, taking on values between $n - \hat{\tau}_n^{(i)} + 1$ and n. The hidden state process is a time-homogenous first-order Markov which has an initial state distribution $P(\varphi_1)$ and transition probabilities $P(\varphi_n|\varphi_{n-1})$. The observable variable is conditional only on the current state, thus we have the state-conditional observation densities $p(R_n^{(i)} | \varphi_n^{(i)})$.

Again, the remaining problem is to find reasonable estimates for the model parameters. State-conditional observation likelihoods $p(R_n^B | \varphi_n^B)$ for the tactus pulse are approximated as

$$p(R_n^B | \varphi_n^B = j) \propto \sum_{c=1}^{c_0} (c_0 - c + 2)(R_n^B)_{c,j}. \tag{30}$$

It is, the likelihood is proportional to a weighted sum of the resonator outputs across the channels. The exact weights of the different channels are not critical. Across-band summing is intuitively meaningful and earlier used in [22] and [35]. Emphasizing the low frequencies is motivated by the "stable bass" rule, as stated by Lerdahl and Jackendoff in [1], and improved the robustness of phase estimation in simulations.

For the purpose of estimating the measure pulse phase, a formula for the state-conditional observation likelihoods analogous to that in (30) is derived, but so that different channels are weighted and delayed in a more complex manner. It turned out that *pattern matching* of some form is necessary to analyze music at this time scale and to estimate the measure phase φ_n^C based on the output matrix R_n^C. It is, no simple formula such as (30) exists. In the case that the system would have access to the pitch content of an incoming piece, the points of harmonic change might serve as cues for estimating the measure phase in a more straightforward manner. However, this remains to be proved. Estimation of the higher-level metrical pulses in audio data has been earlier attempted by Goto and Muraoka, who resorted to pattern matching [17] or to straightforward chord change detection [18]. The method presented in the following is the most reliable that we found.

First, a vector $h_n(l)$ is constructed as

$$h_n(l) = \sum_{c=1}^{c_0} \sum_{k=0}^{3} \eta_{c,k} (R_n^C)_{c,\, j(k,l,n)}, \tag{31}$$

where

$$l = 0, 1, \ldots, \hat{\tau}_n^C - 1, \tag{32}$$

$$j(k, l, n) = n - \hat{\tau}_n^C + 1 + \left(\left(l + \frac{k \hat{\tau}_n^C}{4} \right) \bmod \hat{\tau}_n^C \right), \tag{33}$$

and (x mod y) denotes modulus after division. Scalars $\eta_{c,k}$ are weights for the resonator outputs at channels c and with delays k. The weights $\eta_{c,k}$ encode a typical pattern of energy fluctuations withing one measure period and are estimated so that the maximum of $h_n(l)$ indicates the measure phase. Two universally applicable patterns $\eta_{c,k}^{(1)}$ and $\eta_{c,k}^{(2)}$ were found, leading to the corresponding vectors $h_n^{(1)}(l)$ and $h_n^{(2)}(l)$. The values of these matrices are given in Appendix B. Both patterns can be characterized as a pendulous motion between a low-frequency event and a high-intensity event. The first pattern can be summarized as "low, loud, –, loud", and the second as "low, – , loud, –". The two patterns are combined into a single vector to perform phase estimation according to whichever pattern matches better to the data

$$h_n^{(1,2)}(l) = \max\{ h_n^{(1)}(l), h_n^{(2)}(l) \}. \tag{34}$$

The state-conditional observation likelihoods are then defined as

$$p(R_n^C | \varphi_n^C = j) \propto h_n^{(1,2)}(j - (n - \hat{\tau}_n^C + 1)). \tag{35}$$

Other pattern matching approaches were evaluated, too. In particular, we attempted to sample R_n^C at the times of the tactus beats and to train statistical classifiers to choose the beat which corresponds to the measure beat (see [42] for further elaboration on this idea). However, the methods were basically equivalent to that described above, yet less straightforward to implement and performed slightly worse.

Transition probabilities $P(\varphi_n^{(i)} | \varphi_{n-1}^{(i)})$ between successive phase estimates are modeled as follows. Given two phase estimates (i.e., beat occurrence times), the conditional probability which ties the successive estimates is assumed to be normally distributed as a function of a *prediction error e* which measures the deviation of $\varphi_n^{(i)}$ from a predicted next beat occurence time given the previous beat time $\varphi_{n-1}^{(i)}$ and the period $\hat{\tau}_n^{(i)}$:

$$P(\varphi_n^{(i)} | \varphi_{n-1}^{(i)}) = \frac{1}{\sigma_3 \sqrt{2\pi}} \exp\left\{ -\frac{e^2}{2\sigma_3^2} \right\} \tag{36}$$

where

$$e = \frac{1}{\hat{\tau}_n^{(i)}} \left\{ \left[\left(|\varphi_n^{(i)} - \varphi_{n-1}^{(i)}| + \frac{\hat{\tau}_n^{(i)}}{2} \right) \bmod \hat{\tau}_n^{(i)} \right] - \frac{\hat{\tau}_n^{(i)}}{2} \right\}, \tag{37}$$

and $\sigma_3 = 0.1$ is common for $i \in \{B, C\}$. In (37), it should be noted that none or several periods may elapse between $\varphi_{n-1}^{(i)}$ and $\varphi_n^{(i)}$. The initial state distribution $P(\varphi_1)$ is assumed to be uniformly distributed, i.e., $P(\varphi_1^{(i)} = j) = 1 / \hat{\tau}_n^{(i)}$ for all j.

Using (30), (35), and (36), causal and noncausal computation of phase is performed using the Viterbi algorithm as described in Sec II.C. Fifteen phase candidates for both the winning tactus and the winning measure period are generated once in a second. The candidates are selected in a greedy manner by picking local maxima in $p(R_n^{(i)} | \varphi_n^{(i)} = j)$. The corresponding probability values are stored into a vector and transition probabilities between successive estimates are computed using (36).

E. Sound onset detection and extrametrical events

Detecting the beginnings of discrete acoustic events one-by-one has many uses. It is often of interest whether an event occurs at a metrical beat or not, and what is the exact timing of an event with respect to its ideal metrical position. Also, in some musical pieces there are extrametrical events, such as *triplets*, where an entity of e.g. four tatum periods is exceptionally divided into three parts, or *grace notes* which are pitch events that occur a bit before a metrically stable event.

In this paper, we used an onset detector as a front-end to one of the reference systems (designed for symbolic MIDI input) to enable it to process acoustic signals. Rather robust onset detection is achieved by using an *overall accent signal* $v(n)$ which is computed by setting $m_0 = b_0$ in (4). Local maxima in $v(n)$ represent onset candidates and the value of $v(n)$ at these points reflects the likelihood that a discrete event occurred. A simple peak-picking algorithm with a fixed threshold level can then be used to distinguish genuine onsets from the changes and modulations that take place during the ringing of a sound.

Table 1: Statistics of the evaluation database.

Genre	# Pieces with annotated metrical pulses		
	tatum	tactus	measure
Classical	69	84	0
Electronic / dance	47	66	62
Hip hop / rap	22	37	36
Jazz / blues	70	94	71
Rock / pop	114	124	101
Soul / RnB / funk	42	54	46
Unclassified	12	15	4
Total	**376**	**474**	**320**

III. Results

This section will look at the performance of the proposed method in simulations. The results will be compared with those of two reference systems. The distribution of errors will be analyzed and the importance of different processing elements will be validated.

A. Experimental setup

Table 1 shows the statistics of the database that was used to evaluate the accuracy of the proposed meter estimation method and the two reference methods. Musical pieces were collected from CD recordings, downsampled to a single channel, and stored to a hard disc using 44.1 kHz sampling rate and 16 bit resolution.The database was created for the purpose of musical signal classification in general, and the balance between genres is according to an informal estimate of what people listen to.

The metrical pulses were manually annotated for approximately one-minute long excerpts which were selected to represent each piece. Tactus and measure pulse annotations were made by a musician who tapped along with the pieces. The tapping signal was recorded and the tapped beat times were then detected semiautomatically. The tactus pulse could be annotated for 474 of a total of 505 pieces. The measure pulse could be reliably marked by listening for 320 pieces. In particular, annotation of the measure pulse was not attempted for classical music without the musical scores. Tatum pulse was annotated by the first author by listening to the pieces together with the annotated tactus pulse and by determining the integer ratio between the tactus and the tatum period lengths. The integer ratio was then used to interpolate the tatum beats between the tapped tactus beats.

Evaluating a meter estimation system is not trivial. The issue has been addressed in depth by Goto and Muraoka in [19]. As suggested in [19], we use the longest *continuous* correctly estimated segment as a basis for measuring the performance. This means that one inaccuracy in the middle of a piece leads to 50 % performance. The longest continuous sequence of correct pulse estimates in each piece is sought and compared to the length of the segment which was given to be analyzed. The ratio of these two lengths determines the performance rate for one piece and these are then averaged over all pieces. However, prior to the meter analysis, all the algorithms under consideration were given a four-second "build-up period" in order to make it theoretically possible to estimate the correct period immediately from the beginning of the evaluation segment. Also, it was taken care that any of the input material did not involve tempo discontinuities. More specifically, the interval between two tapped reference beat times (pulse period) does not change more than 40% at a time, between two successive beats. Other tempo fluctuations were naturally allowed.

A correct period estimate is defined to deviate less than 17.5 % from the annotated reference and a correct phase to deviate from an annotated beat time less than 0.175 times the annotated period length. This precision requirement has been suggested in [19] and was found appropriate here since inaccuracies in the manually tapped beat times allow meaningful comparison of only up to that precision. However, for the measure pulse, the period and phase requirements were tightened to 10 % and 0.1, because the measure period-lengths are large and thus more accurate evaluation is possible – and also necessary as will be seen. For the tatum pulse, tactus phase is used and thus the phase is correct always when the tactus phase is correct, and only the period has to be considered separately.

Performance rates are given for three different criteria [19]:

- "Correct": A pulse estimate at time *n* is accepted if both its period and phase are correct.
- "Accept d/h": A pulse estimate is accepted if its phase is correct and the period matches either 0.5, 1.0, or 2.0 times the annotated reference. It is, period doubling or halving is accepted but the factor must not change within the continuous sequence. Correct meter estimation takes place, but a wrong metrical level is chosen to be e.g. the tactus pulse.
- "Period correct": A pulse estimate is accepted if its period is correct. Phase is ignored. For tactus, this is interpreted as the *tempo estimation* performance.

Which is the single best number to characterize the performance of a pulse estimator? This was investigated by auralizing meter estimation results. It was observed that temporal continuity in producing correct estimates is indeed aurally important. Secondly, phase errors are very disturbing. Third, period doubling or halving is not very disturbing. Tapping *consistently* twice too fast or slow does not matter much. Moreover, selecting the correct metrical level is in some cases ambiguous even for a human listener, especially in the case of the tatum pulse. In summary, it appears that the "accept d/h" criterion gives a single best number to characterize the performance of a system.

B. Reference systems

To put the results in perspective, two reference methods are used as a baseline in simulations. This is essential because the principle of using a continuous sequence of correct estimates for evaluation gives a somewhat pessimistic picture of the absolute performance.

The methods of Scheirer [22] and Dixon [16] are very different, but both systems represent the state-of-the-art in tactus pulse estimation and their source codes are publicly available. Here, the used implementations and parameter values were those of the original authors. However, for Scheirer's method, some parameter tuning was made which slightly improved the results. Dixon developed his system primarily for MIDI-input,

Table 2: Tactus estimation performance (%) of different methods.

Method	Continuity required			Individual estimates		
	correct	accept d/h	period correct	correct	accept d/h	period correct
Causal	57	**68**	74	63	78	76
Noncausal	59	**73**	74	64	80	75
Scheirer [22]	27	**31**	30	48	69	57
Dixon [16]	7	**26**	10	15	53	25
O + Dixon	12	**39**	15	22	63	30

Table 3: Meter estimation performance for the proposed method.

Method	Pulse	Continuity required			Individual estimates		
		correct	accept d/h	period correct	correct	accept d/h	period correct
Causal	tatum	44	**57**	62	51	**72**	65
	tactus	57	**68**	74	63	78	76
	measure	42	**48**	**78**	43	51	81
Non-causal	tatum	45	**63**	62	52	**74**	65
	tactus	59	**73**	74	64	80	75
	measure	46	**54**	**79**	47	55	81

and provided only a simple front-end for analyzing acoustic signals. Therefore, a third system denoted "O + Dixon" was developed where an independent onset detector (described in Sec. II.E), was used prior to Dixon's tactus analysis. Systematic phase errors were compensated for in both methods.

C. *Experimental results*

In Table 2 the tactus tracking performance of the proposed causal and noncausal algorithms is compared with those of the two reference methods. As the first observation, it was noticed that the reference methods did not maintain the temporal continuity of acceptable estimates. For this reason, the performance rates are also given as percentages of individual acceptable estimates (right half of Table 2). Dixon's method has difficulties in choosing the correct metrical level for tactus, but performs well according to the "accept d/h" criterion when equipped with the new onset detector. The proposed method outperforms the previous systems in both accuracy and temporal stability.

Table 3 shows the meter estimation performance for the proposed causal and noncausal algorithms. As for human listeners, meter estimation seems to be easiest at the tactus pulse level. For the measure pulse, period estimation can be done robustly but estimating the phase is difficult. A reason for this is that in a large part of the material, two rhythmic patterns elapse within one measure period, and the system has difficulties in choosing which one is the first. In the case that π-phase errors (each beat is displaced by a half-period) would be accepted, the performance rate would be essentially the same as for the tactus. However, π-phase errors *are* disturbing and should not be accepted.

For the tatum pulse, in turn, deciding the period is difficult. This is because the temporally atomic pulse rate typically comes up only occasionally, making temporally stable analysis hard to attain. The method often has to halve its period hypothesis when the first rapid event sequence occurs. This appears in the performance rates so that the method is not able to produce a consistent tatum period over time but alternates between e.g. the reference and double the reference. This degrades the temporally continuous rate, although the "accept d/h" rate is very good for individual estimates. The produced errors are not very disturbing when listening to the results.

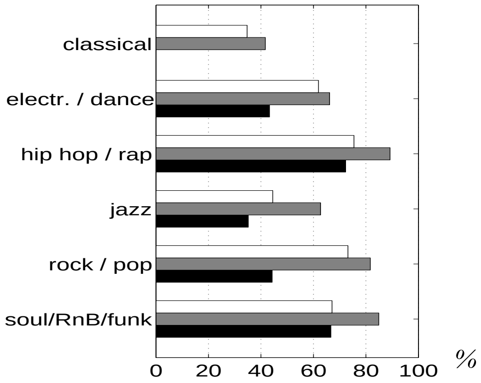

FIG. 11. Performance within different musical genres. The "accept d/h" (continuity required) percentages are shown for the tatum (white), tactus (gray), and measure (black) pulses.

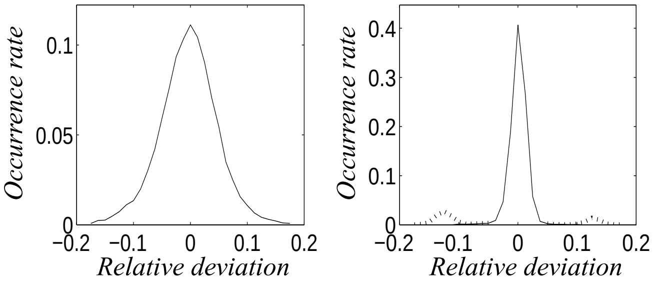

FIG. 12. Relative occurrence frequencies of different phase deviations from the reference phase. The deviation is measured in relation to the period length for the tactus (left) and measure pulse (right).

Figure 11 shows the "accept d/h" (continuity required) performance rates for different musical genres. For classical music, the proposed method is only moderately successful, although e.g. the tactus rate still outperforms the performance of the reference methods for the whole material. However, this may suggest that pitch analysis would be needed to analyze the meter in classical music. In jazz music, the complexity of musical rhythms is higher on the average and the task thus harder.

In Figure 12, the temporal precision of the proposed method is illustrated. We measured the time deviation of accepted phase estimates from the annotated beat times. The deviation is expressed in relation to the annotated period-length. The histogram shows the distribution of deviation values. It should be noted that the reference tapping is not absolutely accurate, but the histogram reflects inaccuracies in both. For the measure pulse, the histogram is quite exactly a 3.8-times narrower copy of that for the tactus. Thus the absolute time deviations are roughly the same, suggesting that they are mostly due to the reference tapping. The dashed line in the right-hand side histo-

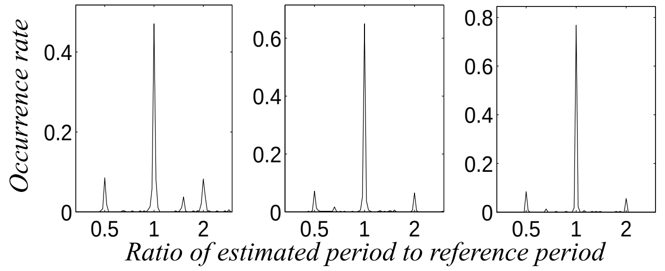

FIG. 13. Histogram of period-estimation errors for tatum, tactus, and measure pulses, from left to right, respectively.

Table 4: *Meter estimation* performance (%) for different system configurations

Method	Require continuity, accept d/h			Individual estimates, accept d/h		
	tatum	tactus	measure	tatum	tactus	measure
0. Baseline	**62**	**73**	**54**	74	80	55
1. No joint estim.	**58**	**68**	**49**	71	75	50
2. No temporal proc.	**45**	**54**	**31**	72	77	50
3. Neither of the two	**41**	**50**	**25**	70	72	44

gram shows the histogram if deviations of up to 0.175 times the period would be accepted for the measure pulse. It is, rhythmically confused phase estimates would be accepted. Therefore, a higher precision (0.1) was required for the measure phase.

Figure 13 shows histograms where the ratio of the estimated period and the annotated period was measured. As can be seen, almost all period estimation errors are either half or double the correct period. In the case that the phase is ignored, 83 % (tatum), 88 % (tactus), and 91 % (measure) of the period estimates are either correct, half, or double the reference period. The fact that the measure pulse could not be annotated for classical music explains why the measure period estimation rate is higher than that of the tactus.

D. *Importance of different parts*

Table 4 gives the performance rates for different system configurations. Different elements of the proposed model were disabled in order to evaluate their importance. In each case, the system was kept otherwise fixed. The baseline method is the noncausal system, as in Table 3.

In the first test, the dependencies between the different pulse levels were broken by using a non-informative (flat) distribution for $g(x)$ in (25). This slightly degrades the performance in all cases. In the second test, the dependencies between temporally successive estimates were broken by using a non-informative distribution for the transition probabilities between successive period and phase estimates, $P(\tau_n^{(i)}|\tau^{(i)}{}_{n-1})$ and $P(\varphi_n^{(i)}|\varphi_{n-1}^{(i)})$, respectively. This degrades the temporal stability of the estimates considerably and hence collapses the performance rates which use the longest continuous correct segment for evaluation. In the third case, the both types of dependencies were broken. The system still performs moderately, indicating that the initial time-frequency analysis method and the comb-filter resonators provide a rather high level of robustness as such.

IV. Conclusions

A method has been described which can successfully estimate the meter of acoustic musical signals. Musical genres of very diverse types can be processed with a common system configuration and parameter values. For most musical material, relatively low-level acoustic information can be used, without the need to model the higher-level auditory functions such as sound source separation or multipitch analysis.

Similarly to human listeners, computational meter estimation is easiest at the tactus pulse level. For the measure pulse, period estimation can be done equally robustly but estimating the phase is less straightforward. Either pattern recognition techniques or pitch analysis seems to be needed to analyze music at this time scale. For the tatum pulse, in turn, phase estimation is not difficult at all, but deciding the period is very difficult for both humans and a computational algorithm. This is because the temporally atomic pulse rate typically comes up only occasionally. Thus causal processing is difficult and it is often necessary to halve the tatum hypothesis when the first rapid event sequence occurs.

The critical elements of a meter estimation system appear to be the initial time-frequency analysis part which measures musical accentuation as a function of time and the (often implicit) internal model which represents primitive musical knowledge. The former is needed to provide robustness for diverse instrumentations in e.g. classical, rock, or electronic music. The latter is needed to achieve temporally stable meter tracking and to fill in parts where the meter is only faintly implied by the musical surface. A challenge in the latter part is to develop a model which is generic for various genres, for example for jazz and classical music. The proposed model describes sufficiently low-level musical knowledge to generalize over different genres.

The presented method enables both causal and noncausal processing within the same model. The backward decoding strategy in the Viterbi algorithm acts as a satisfying counterpart of a phenomenon called *revision* in human perception. Here revision refers to the manner in which the interpretation of previous material is affected by what happens afterwards. Backward decoding at successive time instants is not computationally demanding and gives a retrospective estimate over the whole history up to that point.

Appendix A: Derivation of Observation Densities

This appendix presents the derivation and underlying assumptions in the estimation of the state-conditional observation likelihoods $p(\boldsymbol{s}|\boldsymbol{q})$. We first assume the realizations of τ^A independent of the realizations of τ^B and τ^C:

$$\begin{aligned} &P(\boldsymbol{s}|\tau^A = j, \tau^B = k, \tau^C = l) \\ &\quad \propto P(\boldsymbol{s}|\tau^B = k, \tau^C = l)P(\boldsymbol{s}|\tau^A = j) \end{aligned} \qquad (38)$$

This violates the dependencies of our model but significantly simplifies the computations and makes it possible to obtain reasonable estimates. Furthermore, tatum information is most clearly visible in the spectrum of the resonator outputs, thus we use

$$P(\boldsymbol{s}|\tau^A = j) = P(\boldsymbol{S}|\tau^A = j), \qquad (39)$$

where $\boldsymbol{S}$ is the spectrum of $\boldsymbol{s}$, according to (10). We further assume the components of $\boldsymbol{s}$ and $\boldsymbol{S}$ to be conditionally independent of each other given the state, and write

$$P(\boldsymbol{s}|\tau^B = k, \tau^C = l)P(\boldsymbol{S}|\tau^A = j) \tag{40}$$

$$= \prod_{k'=1}^{\tau_{max}} P(s(k')|\tau^B = k, \tau^C = l) \prod_{j'=1}^{\tau_{max}} P(S(1/j')|\tau^A = j)$$

It is reasonably safe to make two more simplifying assumptions. First, we assume that the height of $\boldsymbol{s}$ and $\boldsymbol{S}$ at the lags corresponding to a period actually present in the signal depend only on the particular period, not on other periods. Second, the value at a lag where there is no period present in the signal is independent of the true periods τ^B, τ^C, and τ^A, and is dominated by the fact that no period corresponds to that particular lag. Hence, (40) can be written as

$$P(\boldsymbol{s}|\boldsymbol{q} = [j, k, l]) \tag{41}$$

$$= P(s(k)|\tau^B = k)P(s(l)|\tau^C = l) \prod_{k' \neq k, l} P(s(k')|\tau^B, \tau^C \neq k')$$

$$\cdot P(S(1/j)|\tau^A = j) \prod_{j' \neq j} P(S(1/j')|\tau^A \neq j')$$

where $P(s(\tau)|\tau^B = \tau)$ denotes the probability of value $s(\tau)$ given that τ is a tactus pulse period and $P(s(\tau)|\tau^B \neq \tau)$ denotes the probability of value $s(\tau)$ given that τ is not a tactus pulse period. These conditional probability distributions (tactus, measure, and tatum each have two distributions) were approximated by discretizing the value range of $s(\tau) \in [0, 1]$ and by calculating a histogram of $s(\tau)$ values in the cases that τ is or is not an annotated metrical pulse period.

Then, by defining

$$\beta(\boldsymbol{s}) = \prod_{k'=1}^{\tau_{max}} P(s(k')|\tau^B, \tau^C \neq k') \prod_{j'=1}^{\tau_{max}} P\left(S\left(\frac{1}{j'}\right)\middle|\tau^A \neq j'\right) \tag{42}$$

Equation (41) can be written as

$$P(\boldsymbol{s}|\boldsymbol{q} = [j, k, l]) = \beta(\boldsymbol{s}) \tag{43}$$

$$\cdot \frac{P(s(k)|\tau^B = k)}{P(s(k)|\tau^B, \tau^C \neq k)} \frac{P(s(l)|\tau^C = l)}{P(s(l)|\tau^B, \tau^C \neq l)} \frac{P(S(1/j)|\tau^A = j)}{P(S(1/j)|\tau^A \neq j)}$$

where the scalar $\beta(\boldsymbol{s})$ is a function of $\boldsymbol{s}$ but does not depend on $\boldsymbol{q}$.

By using the two approximated histograms for tactus, measure, and tatum, each of the three terms of the form $P(s(\tau)|\tau^{(i)} = \tau)/P(s(\tau)|\tau^{(i)} \neq \tau)$ in (43) can be represented as a single discrete histogram. These were modeled as first order polynomials. The first two terms depend linearly on the value $s(\tau)$ and the last term depends linearly on the value $S(1/\tau)$. Thus we can write

$$p(\boldsymbol{s}|\boldsymbol{q} = [j, k, l]) \propto s(k)s(l)S(1/j)\,. \tag{44}$$

The histograms could be more accurately modeled with third-order polynomials, but this did not bring performance advantage over the simple linear model in (44).

Appendix B

Numerical values of the matrices used in Sec. II.D:

$$\eta_{c,k}^{(1)} = \begin{bmatrix} 12 & 1.0 & 0 & 5.7 \\ 0 & 2.0 & 0 & 2.0 \\ 0 & 3.0 & 0 & 3.0 \\ 0 & 4.0 & 0 & 4.0 \end{bmatrix}, \eta_{c,k}^{(2)} = \begin{bmatrix} 10 & 0 & 1.4 & 1.3 \\ 0 & 0 & 2.8 & 0.8 \\ 0 & 0 & 4.3 & 1.2 \\ 0 & 0 & 5.8 & 1.5 \end{bmatrix}, \tag{45}$$

where channel c determines the row and delay k the column. The first row correspond to the lowest-frequency channel.

V. References

[1] F. Lerdahl, R. Jackendoff, *A Generative Theory of Tonal Music*. MIT Press, Cambridge, Massachusetts, 1983.

[2] E. F. Clarke, "Rhythm and Timing in Music," In *The Psychology of Music*, D. Deutsch, Ed., Academic Press, 1999.

[3] J. Bilmes, "Timing is of the Essence: Perceptual and Computational Techniques for Representing, Learning, and Reproducing Expressive Timing in Percussive Rhythm," M.Sc. thesis, Massachusetts Institute of Technology, 1993.

[4] C. S. Lee, "The perception of metrical structure: experimental evidence and a model," In *Representing musical structure*, P. Howell, R. West, and I. Cross, Eds., Academic Press, London, 1991.

[5] M. J. Steedman, "The perception of musical rhythm and metre," *Perception* (6), pp. 555-569, 1977.

[6] H. C. Longuet-Higgins, C. S. Lee, "The perception of musical rhythms," *Perception* (11), pp. 115-128, 1982.

[7] C. S. Lee, "The rhythmic interpretation of simple musical sequences: towards a perceptual model," In *Musical Structure and Cognition*, I. Cross, P. Howell, and R. West Eds., Academic Press, London, 1985.

[8] D. J. Povel, P. Essens, "Perception of temporal patterns," *Music Perception* 2 (4), pp. 411-440, 1985.

[9] P. Desain, H. Honing, "Computational Models of Beat Induction: The Rule-Based Approach," *Journal of New Music Research*, Vol. 28, No. 1, pp. 29-42, 1999.

[10] R. Parncutt, "A Perceptual Model of Pulse Salience and Metrical Accent in Musical Rhythms," *Music Perception*, Vol. 11, No. 4, pp. 409-464, Summer 1994.

[11] J. C. Brown, "Determination of the meter of musical scores by autocorrelation," *J. Acoust. Soc. Am.* 94, pp. 1953-1957, 1993.

[12] E. W. Large, J. F. Kolen, "Resonance and the perception of musical meter". *Connection science*, 6 (1), pp. 177-208, 1994.

[13] D. F. Rosenthal, "Machine rhythm: Computer emulation of human rhythm perception," Ph.D. thesis, Massachusetts Institute of Technology, 1992.

[14] P. E. Allen, R. B. Dannenberg, "Tracking Musical Beats in Real Time," In *Proc. International Computer Music Conference*, San Francisco, 1990.

[15] D. Temperley, *Cognition of Basic Musical Structures*. MIT Press, Cambridge, Massachusetts, 2001.

[16] S. Dixon, "Automatic Extraction of Tempo and Beat from Expressive Performances," *J. New Music Research* 30 (1), pp. 39-58, 2001.

[17] M. Goto, Y. Muraoka, "Music understanding at the beat

level — real-time beat tracking for audio signals," In *Working Notes of the IJCAI-95 Workshop on Computational Auditory Scene Analysis*, 1995, pp. 68-75.

[18] M. Goto, Y. Muraoka, "Real-time Rhythm Tracking for Drumless Audio Signals — Chord Change Detection for Musical Decisions," In *Proc. IJCAI-97 Workshop on Computational Auditory Scene Analysis*, 1997, pp. 135-144.

[19] M. Goto, Y. Muraoka, "Issues in Evaluating Beat Tracking Systems," *IJCAI-97 Workshop on Issues in AI and Music*, 1997, pp. 9-16.

[20] F. Gouyon, P. Herrera, P., Cano, "Pulse-dependent analyses of percussive music," In *Proc. AES 22nd International Conference on Virtual, Synthetic and Entertainment Audio*, Espoo, Finland, June 2002, pp. 396-401.

[21] R. C. Maher, J. W. Beauchamp, "Fundamental frequency estimation of musical signals using a two-way mismatch procedure," *J. Acoust. Soc. Am.* 95 (4), pp. 2254-2263, 1994.

[22] E. D. Scheirer, "Tempo and beat analysis of acoustic musical signals," *J. Acoust. Soc. Am.* 103 (1), pp. 588-601, 1998.

[23] W. A. Sethares, T. W. Staley, "Meter and Periodicity in Musical Performance," *Journal of New Music Research* Vol. 22, No. 5, 2001.

[24] A. T. Cemgil, B. Kappen, "Monte Carlo Methods for Tempo Tracking and Rhythm Quantization," *Journal of Artificial Intelligence Research* 18, pp. 45-81, 2003.

[25] C. Raphael, "Automated Rhythm Transcription," In *Proc. International Symposium on Music Information Retrieval*, Indiana, Oct. 2001, pp. 99-107.

[26] J. Laroche, "Estimating tempo, swing and beat locations in audio recordings," In *Proc. Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, New York, Oct. 2001, pp. 135-138.

[27] A. P. Klapuri, "Sound onset detection by applying psychoacoustic knowledge," In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, Phoenix, Arizona, 1999, pp. 3089-3092.

[28] D. Moelants, C. Rampazzo, "A Computer System for the Automatic Detection of Perceptual Onsets in a Musical Signal," In *KANSEI - The Technology of Emotion*, A. Camurri, Ed., Genova: AIMI/DIST, 1997, pp. 141-146.

[29] M. Davy, S. Godsill, "Detection of Abrupt Spectral Changes using Support Vector Machines. An application to audio signal segmentation," In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, Orlando, Florida, May 2002, pp. 1313-1316.

[30] M. Marolt, A. Kavcic, M. Privosnik, "Neural Networks for Note Onset Detection in Piano Music," In *Proc. International Computer Music Conference*, Göteborg, Sweden, Sep. 2002.

[31] S. A. Abdallah, M. D Plumbley, "Probability as metadata: Event detection in music using ICA as a conditional density model," In *Proc. 4th International Symposium on Independent Component Analysis and Blind Signal Separation,* Nara, Japan, April 2003.

[32] C. Duxbury, J. P. Bello, M. Davies, M. Sandler, "Complex domain onset detection for musical signals," In *Proc. 6th Int. Conf. on Digital Audio Effects,* London, UK, Sep. 2003.

[33] A. P. Klapuri, "Multiple Fundamental Frequency Estimation Based on Harmonicity and Spectral Smoothness," *IEEE Trans. Speech and Audio Processing,* 11(6), pp. 804–816, Nov. 2003.

[34] J. K. Paulus, A. P. Klapuri, "Conventional and periodic N-grams in the transcription of drum sequences," In *Proc. IEEE International Conference on Multimedia and Expo*, Baltimore, Maryland, July 2003.

[35] R. Meddis and M. J. Hewitt, "Virtual pitch and phase sensitivity of a computer model of the auditory periphery. I: Pitch identification," *J. Acoust. Soc. Am.* 89 (6), pp. 2866–2882, 1991.

[36] B. R. Glasberg, B. C. J. Moore, "Derivation of Auditory Filter Shapes from Notched-Noise Data," *Hear. Res.*, Vol. 47, pp. 103-138, 1990.

[37] B. J. C. Moore, Ed., *Hearing. Handbook of Perception and Cognition* (2nd edition). Academic Press Inc., 1995.

[38] R. Meddis, "Simulation of mechanical to neural transduction in the auditory receptor," *J. Acoust. Soc. Am.* 79 (3), pp. 702-711, March 1986.

[39] A. de Cheveigné and H. Kawahara, "YIN, a fundamental frequency estimator for speech and music," *J. Acoust. Soc. Am.* 111 (4), pp. 1917-1930, April 2002.

[40] L. van Noorden, D. Moelants, "Resonance in the Perception of Musical Pulse," *Journal of New Music Research*, Vol. 28, No. 1, pp. 43-66, 1999.

[41] L. R. Rabiner, B.–H. Juang, *Fundamentals of Speech Recognition*. Prentice-Hall, New Jersey, 1993.

[42] Jarno Seppänen, "Computational models of musical meter recognition," M.Sc. thesis, Tampere University of Technology, Tampere, Finland, 2001.