Expressive sampling synthesis. Learning extended source-filter models from instrument sound databases for expressive sample manipulations

Henrik Hahn

THÈSE DE DOCTORAT DE
l'UNIVERSITÉ PIERRE ET MARIE CURIE

École doctorale Informatique, Télécommunications et Électronique (EDITE)

Expressive Sampling Synthesis

Learning Extended Source–Filter Models from Instrument Sound Databases for Expressive Sample Manipulations

Presented by

Henrik Hahn

September 2015

Submitted in partial fulfillment of the requirements for the degree of

DOCTEUR de l'UNIVERSITÉ PIERRE ET MARIE CURIE

Supervisor	Axel Röbel	Analysis/Synthesis group, IRCAM, Paris, France
Reviewer	Xavier Serra	MTG, Universitat Pompeu Fabra, Barcelona, Spain
	Vesa Välimäki	Aalto University, Espoo, Finland
Examiner	Sylvain Marchand	Université de La Rochelle, France
	Bertrand David	Télécom ParisTech, Paris, France
	Jean-Luc Zarader	UPMC Paris VI, Paris, France

This page is intentionally left blank.

Abstract

This thesis addresses imitative digital sound synthesis of acoustically viable instruments with support of expressive, high-level control parameters. A general model is provided for quasi-harmonic instruments that reacts coherently with its acoustical equivalent when control parameters are varied.

The approach builds upon recording-based methods and uses signal transformation techniques to manipulate instrument sound signals in a manner that resembles the behavior of their acoustical equivalents using the fundamental control parameters intensity and pitch. The method preserves the inherent quality of discretized recordings of a sound of acoustic instruments and introduces a transformation method that retains the coherency with its timbral variations when control parameters are modified. It is thus meant to introduce parametric control for sampling sound synthesis.

The objective of this thesis is to introduce a new general model representing the timbre variations of quasi-harmonic music instruments regarding a parameter space determined by the control parameters pitch as well as global and instantaneous intensity. The model independently represents the deterministic and non-deterministic components of an instrument's signal and an extended source-filter model will be introduced for the former to represent the excitation and resonance characteristics of a music instrument by individual parametric filter functions. The latter component will be represented using a classic source-filter approach using filters with similar parameterization. All filter functions are represented using tensor-product B-splines to support for multivariate control variables.

An algorithm will be presented for the estimation of the model's parameters that allows for the joint estimation of the filter functions of either component in a multivariate surface-fitting approach using a data-driven optimization strategy. This procedure also includes smoothness constraints and solutions for missing or sparse data and requires suitable data sets of single note recordings of a particular musical instrument.

Another original contribution of the present thesis is an algorithm for the calibration of a note's intensity by means of an analysis of crescendo and decrescendo signals using the presented instrument model. The method enables the adjustment of the note intensity of an instrument sound coherent with the relative differences between varied values of its note intensity.

A subjective evaluation procedure is presented to assess the quality of the transformations obtained using a calibrated instrument model and independently varied control parameters pitch and note intensity. Several extends of sound signal manipulations will be presented therein.

For the support of inharmonic sounds as present in signals produced by

the piano, a new algorithm for the joint estimation of a signal's fundamental frequency and inharmonicity coefficient is presented to extend the range of possible instruments to be manageable by the system.

The synthesis system will be evaluated in various ways for sound signals of a trumpet, a clarinet, a violin and a piano.

This page is intentionally left blank.

Preface

This thesis represents the results of a part of the multidisciplinary research project Sample Orchestrator 2 conducted from november 2010 until may 2013 at the IRCAM in Paris under the direction of its scientific director Hugues Vinet. The project involved the collaboration with the industrial partner Univers Sons[1], a Paris-based music software company.

The research project involved contributions from several research teams across the IRCAM including the Analysis/Synthesis, the {Sound Music Movement} Interaction, Acoustics and Cognitive Spaces as well as the Music Representations team and additionally the Sound & Software department of Univers Sons. Their respective responsible leads at IRCAM have been Axel Röbel, Norbert Schnell, Markus Noisternig and Gerard Assayag as well as Alaine from Univers Sons (US).

The research program has been setup as a multidisciplinary project for the development of new techniques and algorithms to create a next generation sample-based sound synthesizer.

The following main research directions have been targeted within the project:

- the creation of innovative, realistic sounding virtual solo instruments that allow for expressively controlling its sound,
- the creation of realistic ensembles from individual instruments using a parametric convolution engine for sound spatialization and
- the development of new techniques for a on-the-fly arrangement synthesis of musical sequences for the augmentation of musical performances,

whereas the former research item eventually led to this thesis.

The goal for the creation of expressively controllable solo virtual instruments was to design new sound transformation methods, which render actual instrument characteristics of these instruments for phenomena such as transposition, intensity changes, note transitions and modulations with the target of enhancing the quality/cost ratio of current state-of-the-art methods for instrument sound synthesis.

The Sample Orchestrator 2 project has been financed by the french research agency *Agence nationale de la recherche* as part of the CONTINT (Digital Content and Interactions) program.

[1] www.uvi.net, last accessed 2015-08-07

This page is intentionally left blank.

Acknowledgements

First of all I'd like to thank my supervisor Axel Röbel for giving me the opportunity to do my PhD at the Ircam and with whom I had the privilege to share the office giving me the chance to debate my work almost on a daily basis. His comments and suggestions regarding my work have been invaluable throughout the whole project and the amount of things I was able to learn from him throughout these years is truly amazing.

I would further like to thank all members of my PhD committee: Xavier Serra, Vesa Välimäki, Sylvain Marchand, Bertrand David as well as Jean-Luc Zarader for their commitment and interest into my research.

I also like to thank the various members of the analysis/synthesis team whose company I enjoyed a lot while working at IRCAM and at various after work occasions: Stefan Huber, Nicolas Obin, Marco Liuni, Sean O'Leary, Pierre Lanchantin, Marcelo Caetano, Alessandro Saccoia, Frederic Cornu, Wei-Hiang Lao, Geoffroy Peeters, Maria Gkotzampougiouki, Alex Baker, Reuben Thomas but also members from other IRCAM research departments with whom I interacted a lot within the 3 years of being there: Norbert Schnell, Jean-Philippe Lambert and Markus Noisternig. Working with and next to these people all sharing the passion for music, technology and research was a deeply inspiring experience.

I further like to thank Sean O'Leary and Stefan Huber for providing shelter in Paris when needed by the time I had moved back to Berlin at the end of my work.

All the anonymous participants of our subjective online survey deserve words of thanks as they all took part voluntarily without compensation and participated in the test only motivated by their interest into current research developments..

This page is intentionally left blank.

Publications

[HR12] H. Hahn and A. Roebel. Extended Source-Filter Model of Quasi-Harmonic Instruments for Sound Synthesis, Transformation and Interpolation. In *Proceedings of the 9th Sound and Music Computing Conference (SMC)*, Copenhagen, Denmark, July 2012.

[HR13a] H. Hahn and A. Roebel. Joint f0 and inharmonicity estimation using second order optimization. In *Proceedings of the 10th Sound and Music Computing Conference (SMC)*, Stockholm, Sweden, August 2013.

[HR13b] H. Hahn and A. Roebel. Extended Source-Filter Model for Harmonic Instruments for Expressive Control of Sound Synthesis and Transformation. In *Proc. of the 16th Int. Conference on Digital Audio Effects (DAFx-13)*, Maynooth, Ireland, September 2013.

List of Figures

List of Tables

Contents

This page is intentionally left blank.

Chapter 1

Introduction

1.1 Background

The very beginnings of computer music and digital sound generation is often closely recognized with the development of the computer program series MUSIC I-V developed by Max Mathews at Bell Labs [Mat69] and hence he is perceived as being the first to realize digital sound synthesis [Ris07]. In his highly acclaimed Science article *The digital computer as a musical instrument* [Mat63], Max Mathews states that there *are no theoretical limitations to the performance of the computer as a source of musical sounds* and *any perceivable sound* can be produced using a digital computer. This insight follows the band and dynamic range limitations of the human hearing [Moo12] and its implications regarding the sampling theory introduced by Shannon, Nyquist, Whittaker and Kotelnikov [Lue99]. Though, as he also portrays in [Mat69], digital sound synthesis comes with two fundamental problems: First, the necessity of a very fast and efficient computer program and second, the need for a simple and powerful language in which to describe a complex sequence of sounds.

Smith [Smi91] as well as Serra [Ser97b] conclude, that the first problem has largely being solved due to the progression in computer technology with the advent of machines capable of solving highly complex computations in real-time, whereas the second remains open and may potentially never been solved in general. Both agree in that sound represented by digitized pressure waves must be described with a heavily reduced amount of information, which implies a great loss in generality. Fortunately, there is no need in describing all possible waveforms as most are not of musical interest and the focus should be put on a reduced set of synthesis and control mechanisms [Ser97b].

While considering the demands of artists and composers, Serra in [Ser97b] introduces 2 main objectives for digital sound synthesis and transformation methods: (1) The possibility to create any imaginable sound and (2) the ability to manipulate any pre-existing sound in any conceivable way. However, these objectives are very high-level and rather limited due to a humans restricted ability to imagine the unknown or unheard. It hence seems reasonable to narrow the scope to sounds that have a reference in the real-world. According to Serra, for digital sound synthesis in the context of music this translates to the imitation of natural music instruments that have evolved over centuries

and hence provide an interesting challenge for digital sound synthesis.

Since the appearance of the last iteration of the MUSIC program series by Max Mathews [Mat69], tools for the generation of musical sounds using signal processing techniques play an ever-growing role for composers, producers, sound artists and engineers. Much research has hence been devoted to the development of concepts and algorithms for digital sound synthesis and sound signal transformation since decades [Ris07] and dozens of algorithms have been developed, all having their own unique strengths and weaknesses. Many of these methods however share similar properties or paradigms and in his ICMC keynote speech in 1991 [Smi91], Julius Smith introduced a taxonomy for the categorization of the numerous sound synthesis techniques, which has been revised by Curtis Roads for publication in the *Cahier de l´IRCAM* one year later [Smi92b] and reissued online in late 2005 [Smi05].

In his article he introduces 4 categories each containing numerous synthesis algorithms sharing similar concepts or paradigms:

- Abstract Algorithm
- Physical Model
- Processed Recording
- Spectral Model

Abstract algorithms may refer to digitized versions of voltage controlled oscillators, amplifiers, filters and modifiers and can hence be regarded descendants of analog additive and subtractive synthesis using electric sound synthesizers [Roa96]. Frequency Modulation synthesis [Cho73] and Waveshaping Synthesis [LB78] are also included in this class as well as the original Plucked-String Algorithm by Karplus and Strong [KS83] [1] even though its name states otherwise [KVT98].

An extended version of the Plucked-String Algorithm [JS83] however created the relation to the physics of a plucked string and is hence already classified an actual Physical Model [KVT98]. The relationship between these two algorithms from two categories allows to consider Physical Models a descendant of abstract algorithms [Bil09].

As of today, the paradigm of physical modeling of musical instruments has brought up a variety of synthesis algorithms including Digital Waveguides [Smi87, Smi92a, Smi10a] generalizing the extended KS-algorithm, Modal Synthesis [Adr88, Adr91], Functional Transfer Method [TR03], Wave Digital Filters [Fet86, SdP99], Finite-Difference Schemes [HR71a, HR71b, Bil09] and Mass-Spring Networks [CLFC03].

These approaches typically allow an in-depth control of the sound creation process at the expense of either a limited sound quality due to approximations within the required analytical representations of musical instruments or high computational complexity caused by massive numeric calculations. A thorough and comprehensive introduction to physical modeling techniques available today can be found in [VPEK06].

The use of recordings of musical instruments is mainly associated with wavetable and sampling synthesis [Mas02, Smi05] as well as granular synthesis

[1] This algorithms is also known as Karplus–Strong (KS) Algorithm

[Roa04] and can hence be considered to have its analog origins in the *Musique concrète* [Roa96, Smi05] as well as in the *Tape Music* with their pioneering researchers and composers Pierre Schaeffer and Pierre Henry in the former as well as John Cage and his associates in the latter.

The composer and computer music researcher Jean-Claude Risset criticized *Musique concrète* for favoring an: *aesthetics of collage* [Ris85], which according to [Smi05] also applies to sampling synthesis. Even though, the use of instrument recordings for digital sound synthesis is probably the most often applied sound synthesis technique for the imitation of natural music instruments. This may be derived from the popularity of commercial products as the Vienna Symphonic Library [vie], Garritan Virtual Instruments [gar] or Native Instruments Kontakt Instrument Library [kon].

The process of recording a musical instrument for the purpose of digital sound synthesis refers to a sampling procedure in which the possible timbre space of a music instrument gets being quantized. The granularity of this sampled space therein determines the expressivity of the digital instrument as it delimits the amount of possible performance actions. A loss in expressivity is thus inevitable for sample-based instruments. Most industrial approaches aiming to enhance the expressivity of their digital instruments increase the amount of sampled data using a multi-sampling approach [Mas02], which eventually leads to vast instrument sample libraries exhibiting dozens to hundreds of gigabytes of recorded instrument sound data.

Techniques that follow Serra`s second musical objective [Ser97b] of obtaining the ability to manipulate a sound in any conceivable way could allow for the interpolation of these quantized sound spaces, by using signal transformation methods that deliver results that are coherent with the sound properties of the respective music instruments. Such methods however need to manipulate the time-varying spectral characteristics of the sounds and hence, the progression of the sampling methods towards spectral modeling techniques appears to be reasonable [Smi05].

Spectral Modeling techniques refer to methods which utilize representations of the time-varying spectral characteristics of sound signals for synthesis and transformations. These can be either purely parametric as in additive and VOSIM synthesis [KT78] or obtained through an analysis of a recorded sound. The former hence requires massive manual adjustments to obtain rich and interesting sounding timbres whereas the second suffers from the limited parametric control.

For yielding persuasive results in the imitation of sounds of natural music instruments only two categories of synthesis paradigms are considered prolific [Smi91]: Physical Models and Spectral Models, whereas the former may be considered as models of the sound source and Spectral Models on the other hand may be regarded receiver models [Smi91, Ser97b]. The present thesis is however only concerned with the spectral modeling paradigm using a sample- and hence recording-based approach and the other paradigms will be neglected. The required Spectral Models though will have to provide analysis support as they need to operate on real-world data obtained via digitization of natural instrument sounds specifying the approach even further.

1.2 State-of-the-art

According to [Ser97b], sampling synthesis can be considered to exhibit a high sound quality for the imitation of natural music instruments as the perfect reconstruction of the recorded timbral characteristics can be guaranteed by the todays computing capabilities using high enough sampling rates and bit resolutions. This however comes at the expense of a limited controllability, which mainly results from the sampling process of an instruments sound continuum leading to a quantized timbre space. Its granularity is thereby primarily conditioned by the data storage requirements for all the instrument recordings.

In [Smi05], Julius O. Smith states, that: "a recorded sound can be transformed into any other sound by a linear transformation" using some linear, time-varying filter. We may hence derive that there is no inherent loss of generality in the sample-based sound synthesis method and interpolation of the quantized representation of an instrument's timbre space is possible using standard filtering techniques.

The original Phase Vocoder [FG66] can be regarded the earliest attempt of a Spectral Model with analysis support [Smi05]. It has its origins in the Dudley Vocoder [Dud39b] and extends the Vocoder analysis/synthesis technique [Sch66, Rab68] by an explicit analysis of a signals amplitude and phase spectrum for an improved signal reconstruction. Though, it took until the development of reasonably efficient representations of a signals time-frequency distribution [Coh95] like the Short-Time-Fourier-Transform (STFT) [All77, AR77, RS78] which employs the highly efficient Fast Fourier Transform [CT65] to bring the Phase Vocoder to wider application [Por76, Por78, Por80]. Its early use in computer music and sound synthesis has been described by Moorer [Moo76] and two comprehensive introductions to the Phase Vocoder can be found in [Dol86, Ser97a]

Signal modifications using the Phase Vocoder often rely on the source-filter signal model also introduced by Dudley in 1939 [Dud39a]. The source-filter model of a signal assumes the signal to be produced by an excitation source signal with white spectral distribution and a coloring filter [RR05a, AKZV11]. While the source is either being represented by a pulse train or a white noise signal, the filter modulates the energy of the excitation signal with respect to certain frequency bands [Roe10b]. Thanks to the FFT and vastly raised computing power, the amount of frequency bands available for sound processing within the Phase Vocoder has seen a dramatic increase from 30 bands with 100Hz bandwidth in the original approach [FG66] to several thousand bands called bins in todays implementations. The band-wise filtering procedure has therefore now being replaced by a continuous filter function, the spectral envelope [Roe10b].

With the availability of the STFT as an efficient time-frequency representation, explicit sinusoidal signal representations originally developed for speech processing [AT83, QM86] and low- to mid-bitrate coding for signal transmission [AT82, MQ85] have been introduced for music signal analysis and synthesis with the PARSHL method [SS87]. This modeling approach estimates the individual sinusoidal components of a signal and has shortly thereafter been extended by a dedicated noise model [Ser89, SS90]. The method became known as Spectral Modeling Synthesis (SMS) [Ser97c, ABLS11] and Smith denotes it as *sampling synthesis done right* [Smi91]. It is now being used within successful

commercial applications like Melodyne by Celemony and Vocaloid by Yamaha Coorp., but also within free software as Loris [FHLO02] or Spear [Kli09].

An issue within the spectral modeling approach lies within the analysis of the signals that is required to transform a spectral representation into the sinusoidal domain. Most authors pursue signal analysis for purely harmonic signals since signal analysis of inharmonic signals requires additional algorithms to estimate the frequencies of the sinusoids. Especially for piano sound signals inharmonicity is a decisive property and needs to be considered appropriately [You52, FBS62, FR98]. Several authors proposed methods for the estimation of the inharmonicity coefficient and fundamental frequency of inharmonic sounds, though they all are not designed for sound synthesis but analysis only [GA99, RLV07, RV07, HWTL09, RDD11, RDD13].

Although more advanced models with analysis support for the representation of audio signals [Mal09, Stu09, Bar13] have emerged recently and seem to enable new possibilities for sound synthesis and signal transformations [KD11, CS11, KP12], the Sinusoids plus Noise signal model can be regarded a standard technique in the available repertoire for sound signal synthesis and transformation [Kli09, O'L09, Cae11, Glo12].

Proprietary approaches are the so called Authentic Expression Technology by Native Instruments [Mag10], which allows non-parametric interpolation of spectral envelopes to gradually interpolate the sound space of an instriument between two recordings and the Synful Synthesizer [Lin07] which combines fragments of recorded musical performances in a concatenative approach [Sch04] using neural networks to predict likely sound sequences. Morphing techniques [Cae11] have also been proposed to create more convincing intermediate sounds. However, all these methods do not enable realistic sound transformations as they do not account for actual instrument properties, but rely on spectral envelope interpolation [Mag10, Cae11] or highly specialized and manually annotated recording datasets [Lin07, Sch04].

Many of these methods perform high quality sound transformations with computational costs that are sufficiently low to allow for real time processing [LD99a, LD99b, Roe03b, Roe03a, ABLS11] to essentially interpolate the instruments sampled timbre space. An important issue with these methods is the fact that the signals with either modified pitch or intensity are not coercively acoustically coherent with the untransformed sounds of the same instrument with that specific combination of pitch and intensity as the transformed ones. This severely limits the use of state-of-the-art signal transformation methods for the sample-based sound synthesis method and other sound signal manipulations.

We assume that an improved signal transformation method that is capable of interpolating the quantized timbre space perceptually coherent with the according acoustic instrument needs to incorporate support for extended source-filter models as those which have been introduced for speech processing [FLL85] and since then widely being applied for voice synthesis and transformation [Chi95, DLRR13, CRYR14]. Extended source-filter models provide at least two independent filter functions and hence allow for colored excitations rather than single filter functions only as within the standard source-filter approach. Such models have recently also shown to give significant improvements in several music information retrieval tasks [Kla07, HKV09, KVH10, MEKR11, COVVC$^+$11, CDM14], though they are much more an emerging topic for musical sound synthesis [O'L09, Car09, MV14a].

In [O'L09], a generalized source-filter model with distinct filters for a signals excitation and resonance component has been introduced for signal modeling suitable for sound transformations and in [Car09] an instrument model with separate excitation and resonance filters has been applied to violin sound synthesis using a dedicated gestural parameterization. In a very recent approach an extended source-filter method has also been applied to subtractive synthesis [MV14a]. All these approaches furthermore incorporate the independent manipulation of the individual harmonic and residual components of an instrument's sound signal and hence use distinct component models and implement parallel processing chains.

However, no general method with support for expressive control parameters has yet been introduced to the repertoire of sound transformation techniques that retains the inherent sound quality of the recordings in terms of their perceptual coherence with their acoustic equivalents. Such a method requires knowledge about the characteristics of the according music instrument regarding the selected expressive control parameters.

1.3 Thesis' Scope

Within this thesis an imitative sound synthesis system will be introduced that is applicable to most quasi-harmonic instruments. The system bases upon single-note recordings that represent a quantized version of an instrument's possible timbre space with respect to its pitch and intensity dimension. A transformation method then allows to render sound signals with continuous values of the expressive control parameters, which are perceptually coherent with its acoustic equivalents.

Expressivity in these terms is hence incorporated as part of a general instrument model where sound signal parameters are represented as functions of two manually adjustable control parameters: pitch and global note intensity. The instantaneous intensity is used as a third control parameter to account for time-varying signal variations according to a signal's specific attack and release phase characteristics. The system shall support expressive control parameters by providing a model that is capable of representing smooth signal parameter transitions regarding continuous-valued control parameters.

The parametric instrument model will furthermore provide separate filter functions for the harmonic and residual signal components of an instrument's timbre space to allow processing them individually. The filter function for the harmonic and hence deterministic component uses an extended source-filter model to allow for non-white source excitation functions and separate resonance filter modeling. Thus, the general instrument model is assumed to allow modifications of a signal's pitch or global note intensity in a manner that resembles actual instrument characteristics and hence is supposed to enable synthesis results that are perceptually more convincing than with state-of-the-art methods.

Furthermore, a dedicated model of an instrument's global note intensity will be established to eventually calibrate the synthesis method such that the relative signal level differences for varying values of the global note intensity can be adjusted automatically. This shall be done likewise coherently to actual instrument properties. The intensity model is created using a comparative

analysis of available crescendo and decrescendo sound signals with instrument model estimates of varying values of signal intensity. The intensity model is then utilized to create calibrated instrument models such that level differences for varying global note intensities are incorporated into the instrument model additionally to the spectral characteristics. This allows to modify a signals spectral characteristics and level simultaneously using a single model.

A subjective evaluation procedure will be introduced to assess a variety of transformation results by a direct comparison with unmodified recordings to determine how perceptually close the synthesis results are regarding their respective original correlates.

A new method for the joint estimation of an instrument signal's inharmonicity coefficient and fundamental frequency is furthermore introduced to allow the presented approach for imitative digital sound synthesis to be applicable for piano sound signals as well. The algorithm is not a substantial part of the instrument modeling approach, though required for a high-quality transformation of the spectral representation of piano sound signals into the sinusoidal domain.

1.4 Thesis' Structure

This thesis is constituted of 3 parts:

The first part of the thesis covers an analysis of the state-of-the-art approaches and methods. The part briefly introduces the concept of sample-based sound synthesis since it focuses on a detailed analysis of spectral modeling paradigms as well as thorough descriptions of the sound transformation algorithms representing compulsory requirements for the subsequent part of the thesis. The part will also give an analysis of the available literature on competitive approaches and recent developments within the digital sound synthesis research community.

The second part of the thesis covers its core contribution to the research community and hence contains all necessary components to establish an instrument model that can be used to control a sample-based sound synthesizer with expressive parameters. Within this part of the thesis, a parametric model for the representation of the sounds of a quasi-harmonic instrument will hence be presented together with required parameter estimation and model selection strategies. A method for the application of the presented instrument model for sound transformation using a standard signal processor will be described as well as a method for level calibration of the instrument model. The part concludes with a new subjective evaluation method which gets presented thoroughly and applied to our synthesis method. Its results are given and summarized.

The last part of this thesis contains several conclusions and insights which the author of this thesis has obtained throughout the respective research project and from the prolific subjective evaluation procedure. The part eventually also covers propositions for future improvements and possible research directions for researcher who are deeply concerned about improving the expressiveness of sample-based sound synthesis methods.

In the appendix a new method for the joint estimation of the fundamental frequency and inharmonicity coefficient is presented which had been developed as part of the research conducted for the present thesis. This new method has

been required for the analysis and transformation of inharmonic instrument sound signals to obtain the signal representations required for the instrument model as well as for sound synthesis.

This page is intentionally left blank.

Part I

State-of-the-art

Chapter 2

Sampling-based Sound Synthesis

The sample-based sound synthesis method is sometimes also denoted Sampling Wavetable Synthesis or just Sampling Synthesis [Mas02]. Within this method recordings of entire musical notes of acoustic instruments are stored in some memory and the basic operation of such a synthesizers involves the playback of these digitized recordings according to a trigger event. A musical phrase or performance is hence constructed by multiple note triggers which then create sequences of notes which may be monophonic or polyphonic. In [Mas02] a thorough introduction to Sampling Wavetable Synthesis is given.

Also in [Mas02], two metrics for the assessment of sound synthesis in general and sampling synthesis in particular have been defined: expressivity and accuracy.

Expressivity is defined as the variation of the spectrum and the time evolution of a signal according to the some user input during a performance. This includes that two notes played in succession are never identical regardless of how strongly a musician attempts to create identical sound signals. The two signals may however be identified as correct realizations of the same note event with equal pitch and intensity, though their waveform details may be different and these differences may also even be perceivable.

Accuracy of a synthesis scheme refers to the fidelity of reproduction of the sound of a given musical instrument and may hence only be considered for imitative synthesis. A good accuracy therefore can be seen as a benchmark for the possible quality of a synthesis method and that if a synthesis method achieves an acceptable level of quality could then enable unheard and new sonic experiences using some sort of extrapolation.

As already discussed in sec. 1.1, accuracy of the sampling method can be regarded as very high in comparison to other sound synthesis paradigms as long as only simple playback of a note is required. In terms of expressivity however, the sampling technique lacks the possibility of in-depth control of the spectral content being played back and hence changing the spectral or temporal structure of recordings requires filtering techniques [Mas02] and resampling methods [SG84, FV13]. Several state-of-the-art signal representations that may be used for sound signal transformations will be presented in the next chapter.

The most trivial attempt in sampling synthesis for improving the expressivity while retaining the inherent quality of the recordings consists in the

addition of recordings in a multi-sample approach [Mas02]. Most sample libraries and according synthesizer algorithms have hence distinct recordings available for various pitches playable by the respective instrument as well as separate recordings for several levels of note intensity for every pitch. Such sample libraries therefore "sample" the timbre space of an instrument within two dimensions spanned by a pitch and a note intensity axis eventually yielding a discretized representation of an instrument's sound continuum. However, more control dimensions may be used for even more expressive control, though this typically comes with a loss of generality as further controls may not be available for all instruments. We may hence derive the pitch and the note intensity as the least common set of control variables available for a majority of quasi-harmonic instruments.

These two control parameters are also supported by most control interfaces required to trigger note events in real-time performances and the restriction to such a limited set of control variables may hence well-preserve the universality of a transformation method aiming to increase the overall expressivity of the sampling synthesis method.

Chapter 3

Signal Models for Sound Transformation

Introduction

This chapter deals with spectral modeling techniques with analysis support which are essentially based upon the source-filter model and the phase vocoder method. We assume these signal models and their extensions to be suitable for various spectral processing tasks and will therefore give a recapitulatory review of the state of the art of their according models and methods.

We start with shortly revisiting the STFT signal representation, which represents the basis for most spectral modeling approaches and introduce some important aspects regarding the Phase Vocoder and the sinusoidal modeling approach. Certain facets of sinusoidal modeling which are important for this work will be discussed. This includes a short review of 2 parameter estimation methods for sinusoidal parameter estimation and further requirements for this thesis.

3.1 The Short-Time Fourier Transform

When dealing with signals in a digital computer, we typically refer to them by their time-discrete representations $x(t)$ representing a sampled version of the signal at a given constant sampling intervall $1/T$. Furthermore, the respective amplitude values at each sampling position are discretized with a fixed resolution to eventually obtain a digital signal.

The time-varying spectral properties of a signal can be represented using the Short-Time Fourier-Transform [RS78, Coh95] (STFT) also denoted Short-Term [All77, AR77] or Time-Dependent Fourier Transform in the literature [OS10]. For discrete frequencies f the STFT may be written as follows:

$$X(f,n) = \sum_{m=0}^{L-1} w_x(m)\, x(nR+m)\, e^{-j(2\pi/N)fm} \tag{3.1}$$

whereas $w_x(m)$ refers to a symmetric window sequence [Har78, Nut81] with length L and the term within the sum represents the Discrete Fourier Transform (DFT) of the windowed portion of the signal. The signal portion is being taken

with respect to some hop size R and frame index n starting at 0. The time and frequency discrete representation $X(f, n)$ of the signal hence exhibits N equidistant frequency values called bins at each time frame n while assuming $R < L \leq N$ to ensure its reversibility.

In signal analysis, the windowing sequence typically refers to some raised cosine function, whereas several types are being used. Window functions differ by their spectral main lobe width and side lobe attenuation and window functions either exhibit wide main lobes with good side lobe attenuation or the opposite and may therefore be denoted high dynamic range windows in the former or high resolution windows in the latter. They should hence be chosen with respect to the signal to be analyzed.

An important property of the STFT representation of a time-discrete signal became known as the Uncertainty Principle of signal analysis [Coh95]. There is however nothing uncertain, but a well-known mathematical fact and fundamental statement regarding joint time-frequency analysis, which essentially states, that the time and frequency resolution of the analysis are mutually reciprocal. This means, that by raising the frequency resolution of the analysis one decreases the time resolution and vice versa.

Within the STFT, the frequency and time resolution is determined only by the analysis block length and in harmonic signal analysis the minimum analysis block length L for a meaningful STFT representation depends on the lowest fundamental frequency contained in the signal to be analyzed and the applied windowing sequence. The fundamental frequency is closely related to the perceived pitch [Moo12] of a sound and hence a signal attribute of significant importance for many signal analysis and sound synthesis tasks. A more thorough discussion on the fundamental frequency of quasi-harmonic signals and their estimation will be given in sec. 3.3.4.

The Hann window function for example requires about 4 periods of the lowest sinusoidal signal component representing its fundamental frequency to obtain a reasonably good spectral separation of the main lobes of the harmonic series of overtones, whereas a Blackman window function requires 5 to 6 fundamental periods for a similar separation of the spectral signal content [Har78].

Taking the DFT of a signal segment includes the assumption about its quasi-stationarity within this segment as its DFT yields complex values for each spectral frequency bin averaged over the whole analysis block length L. When assuming an analysis hop size $R = L$ this would result in a single complex value per spectral bin every 4 to 6 fundamental periods which can be a fairly poor temporal resolution for harmonic signal analysis. To increase the amount of temporal sampling positions for the spectral analysis R is typically set to some value between $1/4-$ or $1/8-$th of the analysis block length L. This method for increasing the amount of temporal sampling positions can be denoted temporal oversampling, though it is important to note, that this approach does not increase the actual temporal resolution of the STFT.

Another method to enhance the analysis result of the STFT is called spectral oversampling. Using a DFT length N greater than the analysis block length L yields a time-varying spectral representation with an amount of spectral sampling positions greater than the analysis block length. This increased amount of values however does not refer to an increased spectral resolution, but an interpolated version of the spectrum of the windowed sequence of the

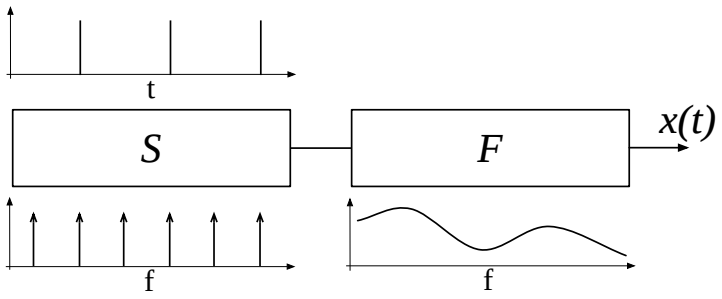


Figure 3.1: The classic source-filter model with an ideal pulse as source signal shown in time (top) and frequency (bottom) domain and a time-variant filter. The spectrum of the source signal exhibits a white distribution and hence all spectral coloring is due to the filter.

signal with a resolution according to the used analysis block length L. Spectral oversampling however can increase the accuracy of subsequent spectral analysis methods and N is often set to some value between 2 to 8 times L. Due to computational performance reasons when computing the DFT using an FFT algorithm, the amount of spectral sampling positions N is often constrained to powers of 2.

3.2 The Source-Filter Model

Within the source-filter model of speech production [Dud39a], a voiced speech signal is assumed to be produced by a source emitting an ideal pulse and a time-variant filter, which modifies the excitation signal by superimposing its spectral envelope as shown in fig. 3.1. The time-varying spectral representation of a signal $x(t)$ is hence assumed to be produced as in eq. (3.2).

$$X(f, n) = S(f) \cdot F(f, n) \tag{3.2}$$

where $S(f)$ refers to an excitation source signal and $F(f, n)$ represents the time-varying filter function.

Within the classic technique for source-filter modeling, the source is assumed to exhibit a white spectral distribution and source-filter processing then refers to the estimation of the time-varying spectral envelope of the filter $F(f, n)$ to enable various kinds of sound manipulations [AKZV11]. In voice processing, the spectral envelope is typically seen as the transfer function of the vocal tract whereas more generally, the spectral envelope of a quasi-stationary sound may be viewed as the acoustic representation of its timbre [MV13].

Estimation of the transfer function of the filter essentially means finding a smooth curve which approximates the spectrum of a signal frame with an envelope that passes through its prominent peaks. Within the source-filter framework, the peaks are assumed to represent the sinusoids created by the source with equal amplitude and hence their relative amplitude deviations refer to the filter sampled at the peaks frequency locations.

A large variety of envelope estimation procedures have been proposed in the literature. The Linear Predictive Coding technique (LPC) [Mak75] has been

introduced for the estimation of the human vocal tract from voice recordings but exhibits some quite undesirable strong peakiness. Other methods are the Discrete All-pole Model (DAP) [EJM91] as well as the regularized discrete cepstrum method [GR87, CLM95, CM96]. Both methods have been extended with a penalized likelihood criterion in [OCM97, COCM01] for an improved stability of the estimations.

The True Envelope method [RR05a] is an iterative method based upon the signal's real cepstrum whose properties have shown to be superior for arbitrary envelopes when compared with the LPC and DAP method [RVR07]. Warped frequency scales have also been studied [VRR08, MLV13] for perceptually more meaningful estimates of the spectral envelope. The probably latest addition to the family of spectral envelope estimation techniques is the True Discrete Cepstrum [MV14d] which yields improved results for high model orders.

Within the paradigm of source-filter based sound processing, independent manipulations of the source and filter are possible. This is particularly important for pitch transformations of speech signals where formant locations and hence the spectral envelope shall be preserved, but also allows for advanced and complex signal manipulations like sound morphing [Cae11] or cross-synthesis [Smi10b].

3.3 The Sines plus Noise Model

The sinusoidal model as used in the Sines plus Noise model has its origins in the source-filter model of speech production [Roe10b] and has hence first been developed for speech signals [MQ86]. There, an analysis/synthesis technique has been proposed for the creation of a sinusoidal model that is characterized by the amplitude, phase and frequency trajectories of a signals sine wave components. This sinusoidal representation is considered to consist of a glottal excitation that is represented by means of a sum of sine waves to which a time-varying vocal tract filter has been applied.

This model has been applied for music and instrument sound signals for the first time with the PARSHL method [SS87] and shortly thereafter extended by an explicit model of a signals residual, non-deterministic component in [Ser89, SS90]. The approach is denoted Spectral Modeling Synthesis and represents a sound signal $x(t)$ as a superposition of a deterministic, sinusoidal component $x_h(t)$ and a residual $x_r(t)$ signal. It allows to link an instrument's signal with its physical source as the sinusoidal components can be considered the filtered modes of a vibrating string or an air column, while the residual may refer to wind or bow noise and a potentially transient signal component.

We may thus write the superposition of a deterministic $x_h(t)$ and a non-deterministic component $x_r(t)$ as:

$$x(t) \;=\; x_h(t) \;+\; x_r(t) \tag{3.3}$$

whereas the deterministic component refers to the linear combination of K

time-varying sinusoids:

$$\begin{aligned} x_h(t) &= \sum_{k}^{K} a_{(k)}(t) \cdot \cos(\phi_{(k)}(t)) \\ &= \sum_{k}^{K} a_{(k)}(t) \cdot \cos(2\pi f_{(k)}(t)\, t) \end{aligned} \tag{3.4}$$

with its parameters:

$$\begin{array}{rcl} \phi_{(k)}(t) & : & \text{phase,} \\ a_{(k)}(t) & : & \text{instantaneous amplitude and} \\ f_{(k)}(t) & : & \text{instantaneous frequency} \\ & & \text{for a partial } k \text{ at sampling position } t. \end{array}$$

Estimating such a representation from an audio signal requires an elaborate analysis that decomposes the signal into such a linear combination of sinusoidal components and a residual $x_r(t)$ signal, whereas the non-deterministic component is typically obtained by inverse filtering of $x(t)$ with the estimate of $x_h(t)$.

Decomposing a signal into atomic components like time-varying sine waves is persistently driving parts of the signal processing research community since many decades and a tremendous amount of methods and algorithms have been proposed since then.

Methods that do not rely on a time-frequency representation like the STFT and hence do not suffer from the Uncertainty Principle briefly discussed in sec. 3.1 are MUSIC [Sch86], ESPRIT [RK89] or HRHATRAC [DBR06, BDR06] which are based on subspace exploration. Matching-[Mal93] or Basis Pursuit [CDS01] based methods are using dictionaries of atomic prototypes with their dictionary size being the main limiting factor. All these methods allow for a simultaneous increase of the time and spectral resolution and are therefore also called High-Resolution methods. However, this generally comes with either a highly increased computational cost or a reduced signal to residuum ratio.

Approaches that work on the basis of overlapping signal frames typically come with an inherent assumption about the quasi-stationarity of the signal to analyze and require the discussed compromise on temporal and spectral resolution, though they may however benefit from fast FFT implementations. As in the original proposition for PARSHL and Spectral Modeling Synthesis, the STFT $X(f, n)$ of a signal $x(t)$ can be used to estimate the instantaneous parameters of the sinusoids, but also (Weighted) Least-Squares (LS) methods [Lar89a, Lar89b, LSM93, Sty96, Oud98] have been proposed, which allow to use shorter time frames for the estimation than STFT based methods [Sty96], but otherwise do not allow for the estimation of all instantaneous parameters simultaneously.

The reassignment [AF95] and the signal derivatives method [DCM98] as well as their generalizations for non-stationary sinusoids [XS09, MB11] represent approaches that are also based on the STFT, but aim to increase the precision of the instantaneous estimates of the sinusoidal parameters beyond the limits drawn by the compromise of time and spectral resolution as well as by the assumption of intra-frame stationarity by introducing linear or logarithmic modulation parameters to the sinusoidal model from eq. (3.4).

Review articles which study various parameter estimation techniques for quasi-stationary sinusoids can be found in [KM02, HM03], whereas methods for the estimation of the instantaneous parameters of non-stationary, modulated sinusoids have been investigated recently in [MB10, HD12].

Within this thesis we particularly considered two approaches for parameter estimation of quasi-stationary sinusoids as in conjunction they allow for reasonable high quality results for our purpose of analyzing single-note recordings of musical instrument sounds which can be assumed to exhibit fairly slow modulations if at all. We hence employ the parabolic interpolation of the peaks of a zero-padded discrete Fourier transform as originally proposed for the PARSHL method [SS87] extended by a bias correction as proposed in [AS04]. A subsequent LS method [Sty96] is used to increase the time resolution of the partial estimates in critical regions for a subset of musical instruments.

3.3.1 Parameter Estimation Using Parabolic Interpolation

A time-continuous signal with a time-invariant sinusoidal component exhibits a continuous spectrum with a single line at the sinusoids frequency. However, in digital spectrum analysis, the windowing effect also known as spectral leakage [Har78] and the discrete nature of the analysis lead to a sampled version of the spectrum, which exhibits a main lobe and and an unlimited amount of sidelobes (the effect of spectral aliasing may be ignored here). The frequency sampling positions of the DFT hence do not necessarily lie at the exact frequency location of the sinusoid and hence the analysis appears blurred and the exact location of the sinusoid can only be measured approximately.

In [MQ86] sinusoidal components are detected by means of locating local maxima within a signals DFT starting with the peak with the largest magnitude and iteratively selecting maxima with the next lower magnitude. This procedure is being repeated until a fixed amount of spectral peaks has been selected. This method however is rather inaccurate due to the reasons discussed above and hence the PARSHL [SS87] method proposes an interpolation of the position of the spectral peak using a parabola fitted to a local maxima and its direct neighbors to estimate the true frequency location of the sinusoid. The second order polynomial has been chosen as it approximates the shape of the main lobe of the window function efficiently. The use of a second order polynomial gave this method its name: Quadratic Interpolated FFT (QIFFT) [Smi10b].

Nonetheless, the approximation of the shape of the main lobe using a parabola introduces a general bias which may result in audible artifacts when resynthesizing the estimates [AS04]. In [AS04] a bias correction term has hence been introduced, which efficiently corrects the frequency and amplitude estimates by cubic and parabolic bias correction functions respectively. These functions introduce heuristic knowledge about the shape of the main lobe as window-dependent functions and the authors have shown the accuracy improvement of the parameter estimates while maintaining the overall efficiency of the method.

3.3.2 Parameter Estimation Using Weighted Least–Squares

As briefly discussed in sec. 3.1, the analysis of harmonic signals in a DFT based approach requires analysis blocks with a length that is greater than or equal to 4 to 6 times the period of the lowest fundamental frequency contained in the signal depending on the window function being used. An estimation based on such long analysis blocks may yield a fairly poor approximation of the sinusoidal parameter trajectories when the parameters exhibit rapid fluctuations. In the previous section we have mentioned various methods that target for an improved analysis using different kinds of sinusoidal models that support for local non-stationarity. The class of Least Squares methods though allow for sinusoidal parameter estimation using the same quasi-stationary sinusoidal model for parameter estimation as the parabolic interpolation method, though they are not based on the DFT of a signal, but the signals time domain representation and allow for shorter analysis frames [Sty96]. However, they require to estimate the frequency and amplitude trajectories of the signals sinusoids in separate steps.

The Harmonic plus Noise model in [LSM93, Sty96] has originally been proposed for speech signals and assumes the signal segments of interest to exhibit sinusoids in a harmonic relationship. They further assume the amplitudes and frequencies of the sinusoidal components of the signal being nearly constant within the frame to be analyzed. Their analysis of the deterministic signal component of a harmonic signal segment then consists of a pitch estimation technique from which the frequencies of sinusoids will be generated and a subsequent estimation of the partials instantaneous amplitudes and phases is done using a Weighted Least-Squares (WLS) approach.

We have adapted the WLS method in [Sty96] such that it allows to improve the temporal resolution of a preceded analysis using parabolic interpolation with bias correction. As the harmonic instrument model to be introduced in sec. 6.1 only aims to represent partial amplitude trajectories, we target for an improved analysis of the sinusoid's amplitudes only and consider the estimates of its phase and frequency trajectories obtained by the interpolation method to be precise enough for our purpose.

We may hence reevaluate the real-valued amplitudes of the sinusoids of a signal by use of the weighted least-squares method aiming at minimizing the error criterion in eq. (3.5) with respect to $a_{(k)}(n')$ [Sty96].

$$\epsilon(n') = \sum_{m=0}^{L'-1} w_{\text{ls}}^2(m)\big|x(n'R'+m) - \hat{x}_h(n'R'+m)\big|^2 \tag{3.5}$$

Variable n' in eq. (3.5) denotes the new set of frame indices and L' and R' refer to a new block length and hopsize respectively, which can be made smaller than for the STFT analysis. The $x(n'R'+m)$ refers to an according segment of the original signal and $\hat{x}_h(n'R'+m)$ represents an estimated version of the harmonic signal component of that segment which has been defined in eq. (3.4). The weighting function $w_{\text{ls}}^2(m)$ is used to provide a better localization of the signal towards the center of the frame to be analyzed. The use of a non-rectangular weighting window is strongly recommended to avoid audible artifacts in the residual signal [LSM93].

The estimate of the harmonic signal component may now be rewritten in matrix notation for a single frame n':

$$\hat{\mathbf{x}}_h = \mathbf{B}\mathbf{a} \tag{3.6}$$

with $\mathbf{a} = [a_{(1)}(n'), \ldots, a_{(K)}(n')]^T$ being the vector of unknown partial amplitudes. The symbol ^{T} denotes the transpose operator.

The matrix $\mathbf{B}$ can be established using knowledge about the sinusoidal phase trajectories. Assuming a preceded analysis of the deterministic component of a harmonic signal using the STFT-based parabolic interpolation method, we are able to interpolate the frame-based phase estimates $\phi_{(k)}(n)$ of the K detected sinusoids at sample positions t by using the cubic phase interpolation method proposed in [MQ86] for maximum smoothness of the phase functions $\phi_{(k)}(t)$. These phase functions can be used directly to develop the phase trajectories for the sinusoids K within a frame n':

$$\mathbf{B} = \begin{bmatrix} \cos(\phi_{(1)}(n'R')) & \cdots & \cos(\phi_{(1)}(n'R' + L' - 1)) \\ \vdots & \ddots & \vdots \\ \cos(\phi_{(K)}(n'R')) & \cdots & \cos(\phi_{(K)}(n'R' + L' - 1)) \end{bmatrix} \in \mathbb{R}^{K \times L'} \tag{3.7}$$

The solution to the Weighted Least-Squares problem is then given by the normal equations

$$\left(\mathbf{B}^T\mathbf{W}^T\mathbf{W}\,\mathbf{B}\right)\mathbf{a} = \mathbf{B}^T\mathbf{W}^T\mathbf{W}\,\mathbf{x} \tag{3.8}$$

whereas $\mathbf{W}$ is a $L' \times L'$ diagonal matrix with the values of the weighting function $w_{\mathrm{ls}}(m)$ as diagonal elements. The vector $\mathbf{x}$ represents the frame of the original signal to be analyzed and solving the equation for $\mathbf{a}$ yields the desired partial amplitude values.

3.3.3 Partial Tracking

The parabolic interpolation method introduced in sec. 3.3.1 only yields instantaneous parameter values and does not reveal any information about the temporal connections of the estimated sinusoidal peaks within each frame. For the estimation method using WLS though, we already need to know, which spectral peaks in consecutive frames belong to the same sinusoid to interpolate the phase properly. An intermediate step is hence necessary to assign the instantaneous parameter estimates of consecutive analysis frames to sinusoidal trajectories. This procedure is typically called Partial Tracking.

In [MQ86], spectral peaks are matched recursively on a frame by frame basis according to their frequency difference and a general concept of a sinusoids birth and death is introduced. A more complex system has been proposed in [Ser89, Ser97c] in which an algorithm has been developed that is motivated by line detection methods in image processing. There, several additional constraints are introduced extending the simple frequency difference condition for peak matching by a minimum length for each trajectory, a sleeping or zombie time for a trajectory as well as a maximum allowed frequency deviation among others.

Other strategies have been developed using for example Hidden Markov Models [DGR93] or Linear Prediction techniques [LMR07] for creating trajectories from the unconnected, frame-wise measurement data, which show superior results for complex sounds with crossing partials or highly modulated sinusoids as well as polyphonic signals.

However, in this thesis we will only analyze signals that are monophonic, harmonic signals with constant pitch and therefore we use a partial tracking approach which resembles the one proposed in [Ser89, Ser97c], though we are using less constraints as we may make stronger assumptions about the content of our signals to analyze. For all signals, we assume them to exhibit a recording of a single note of a quasi-harmonic instrument only whereas its only slightly time-varying fundamental frequency $f_0(n)$ is known a priori and hence express our set of conditions for the partial continuation algorithm as follows:

1. Harmonicity is enforced. We only consider peaks for the analysis, which satisfy the harmonic relationship within a tolerance level of $\mu = 0.2$ around the hypothetically exact location of $f_{(k)}$ and we only consider the strongest spectral peak within each band around $f_{(k)}$. This shall ensure, that the resulting sinusoidal model represents actual harmonics of the signal only. The peaks are sorted in such a way, that the partial index k of the sinusoidal model equals the harmonic index which can be done using:

$$f_{(k)}(n) = \arg\max_{f} \left\{ X([k \cdot f_0(n) - \mu, \ldots, k \cdot f_0(n) + \mu], n) \right\} \tag{3.9}$$

2. Partials at all harmonic indexes are only allowed to be born once throughout the signal. Once they disappear in the spectrum, they will not get reborn if a peak reappears later in time.

3. However, a maximum sleep time for about 100ms will be allowed. These time gaps in a partial trajectory are being interpolated using the McAulay/ Quatieri method before resynthesis [MQ86].

4. Partial trajectories need to exhibit at minimum length of about 100ms, otherwise they are being deleted from the model.

In our case of a uniform and predictable set of sound signals, these 4 simple qualifiers yielded sufficient results for the partial tracking algorithm as long as the fundamental frequency has been tracked previously with high accuracy and that the analysis parameters for the STFT have been adjusted properly such that all relevant harmonic sinusoids of the signal could be resolved. The tracking of the fundamental frequency of monophonic harmonic as well as inharmonic sound signals will be discussed in the next section.

It is important to note, that the QIFFT as well as the WLS parameter estimation method only yield values at a certain frame rate n or n' respetively whereas the partial tracking procedure establishes links between the frame-wise partial data. To synthesize the partial trajectories from the frame-wise data at a certain samplerate the data needs to be interpolated. A well-known method for synthesizing partial data has been described in [MQ86] which uses linear interpolation for amplitudes and third order interpolation for their respective

phases. This method has shown to be sufficient for most cases [GMdM+03]. Though several approaches for approximate and faster interpolation using linear interpolation at reduced sampling rates have been proposed [HB96] as well.

3.3.4 The Fundamental Frequency

The fundamental frequency is an important feature for the harmonic analysis of monophonic signals. As briefly discussed in sec. 3.1, the analysis window for an STFT representation needs to account for the lowest fundamental frequency contained within a signal to yield a frequency resolution that is precise enough to resolve its harmonic partials. Hence, to estimate the instantaneous parameters of the sinusoids for the Sines plus Noise model as well as for tracking the partials with respect to their harmonic index, the estimation of the signals time-varying fundamental frequency is necessary.

Assuming a Sines plus Noise model specified by eq. (3.4) for a quasi-harmonic signal, the partial index k typically refers to the harmonic index of the overtone series of the signal, which is defined as the harmonic series shown in eq. (3.10). One may note that the Sines plus Noise model is not restricted to a certain order of its partials, though it is typically convenient to have an order of increasing frequency value and therefore the partial with index $k = 1$ equals the fundamental frequency f_0.

$$f_{(k)} = k \cdot f_0, \quad k = 1 \dots K \tag{3.10}$$

Various approaches and techniques for the estimation of the fundamental frequency from quasi-harmonic signals can be found in the literature. Early approaches have been based on the signals cepstrum [Nol67] or its spectrum [Nol70], though the time-domain based YIN algorithm [dCK02] is perhaps the most popular approach due to its computational efficiency and its robust estimations [KZ10]. More recently, fundamental frequency estimation methods that extend the YIN algorithm by either a probabilistic framework [MD14] or a normalization and windowing technique [MW05, McL08] have been proposed for further improvements.

In [YRR10] a method has been proposed which ranked very high in several succeeding MIREX evaluations [YR09, YR10, YR11] for tracking multiple fundamental frequencies in polyphonic sounds. The method however also supports the analysis of the fundamental frequency within monophonic signals and will hence be used for such within this thesis.

There are however quasi-harmonic instruments, whose sound signals do not exhibit a harmonic series as shown in eq. (3.10) and hence require a different model for the estimation of their fundamental frequency as shown in the following.

3.3.4.1 Fundamental Frequency Estimation of Inharmonic Signals

Inharmonicity means that the frequencies of the partials are not exact integer multiples of their fundamental frequency but located at increased frequencies. This due to the stiffness of the strings which effects the frequencies of the modes of vibration [You52] leading to a shift of the partial frequencies. This effect is perceptually significant for the piano but also applies to all other percussively excited string based instruments like guitars [JVK01]. The amount of increase

of the partial frequencies is reflected by the inharmonicity coefficient β within eq. (3.11) [FBS62], whereas this coefficient of inharmonicity is a characteristic of the stiff string [You52].

$$f_{(k)} = k \cdot f_0 \sqrt{1 + k^2 \beta}, \quad k = 1 \dots K \tag{3.11}$$

This inharmonicity coefficient does not only depend on the diameter and tension of the string, but also on its length, and hence its according value varies with the signals fundamental frequency. The fundamental frequency however, is then only a theoretical value, as there is no partial with that specific frequency present in an inharmonic signal as can be seen in eq. (3.11). Both, the inharmonicity coefficient β as well as the fundamental frequency f_0 can hence not easily be measured from an instruments signal.

Several approaches for the automatic estimation of the inharmonicity coefficient β and fundamental frequency f_0 of inharmonic signals have been proposed in the literature. In [GA99] inharmonic comb filters have been proposed, whereas the parameters for the filter have been found by an exploration of a vast range of possible values within three consecutive steps, while the parameter search grid is refined in each iteration. The algorithm finally interpolates the best parameter sets to obtain its f_0 and β coefficient.

A partial frequencies deviations method has been proposed in [RLV07, RV07] as well as a median-adjustive trajectories method in [HWTL09, Hod12]. Both methods are using an initial estimate for the fundamental frequency obtained by either using an estimation method for harmonic signals or by some user input. The algorithms then iteratively refine the estimates by either analyzing the trend of deviation from the harmonic series and modifying β accordingly in the former or by solving an analytic expression for the mutual relationship of two partials in the latter.

Another approach is based on non-negative matrix factorization (NMF) which allows to jointly estimate the fundamental frequency f_0 and the inharmonicity coefficient β for the whole pitch range of an instrument at once [RDD12, RDD11].

All these methods are characterized by their target application of pitch estimation for piano recordings and for such, they typically consider only the first 30 harmonic partials or even less and most use fixed amplitude thresholds. This seems to be appropriate for most pitch estimation and music transcription tasks, though to reliably estimate the frequency locations of all harmonic partials of piano sounds, a much higher accuracy for the estimation of the inharmonicity coefficient is required.

Piano recordings cover a wide dynamic range and a large spectral bandwidth. The use of fixed amplitude thresholds apparently represents a major limitation for an algorithm and so does the use of a fixed amount of partials. In our publication [HR13] we have studied the effect of small deviations in the estimation of the inharmonicity coefficient for the estimation of the frequencies of upper partials which remain undetected or get even mismatched if only up to 30 partials are considered for the estimation of the inharmonicity coefficient. We have hence proposed a new technique for the joint estimation of the inharmonicity coefficient and fundamental frequency in [HR13] and a description of the method will be given in the appendix A. Other recent approaches are presented in [RDD13] or [KCOL14].

3.3.5 Residual Modeling

Assuming a monophonic, quasi-harmonic signal $x_h(t)$ and a parameter estimation technique that perfectly finds its harmonic partials, we may conclude that the residual signal $x_r(t)$ obtained by inverse filtering of the input signal $x(t)$ with the synthesized harmonic signal will only contain noise. To allow for any further processing of a residual signal as well as for any meaningful manipulation we utilize the source-filter framework and hence its representation by its time-varying spectral envelope. Within this thesis, spectral envelope estimations techniques have been discussed briefly for processing of deterministic signals and envelope estimators have hence been focusing on smooth envelopes passing through the prominent spectral peaks refering to a signal's sinusoidal components.

For residual signals we may assume no deterministic components to be present and hence require a spectral envelope that describes its statistical properties rather than instantaneous sinusoidal parameters. In [RS07] cepstrum based methods are proposed for envelope estimation of residual signals that are assumed to be random since cepstral coefficients are well suited for further processing of such signals. The method of cepstral smoothing [Smi10b] may be used to obtain a smooth envelope of the spectrum of a noise signal using its real cepstrum [RS78, OS10]. This method is known to yield an envelope that follows the mean of the spectrum [RR05b] rather than its heavy fluctuations due to the lowpass filtering of the real cepstrum.

The signal's time-varying real cepstrum is obtained by taking the inverse Fourier Transform of the logarithm of the magnitude spectrum shown in eq. (3.12) with l being the index for the cepstral coefficient.

$$C_{(l)}(n) = \sum_{f=0}^{N} \log(|X_r(f,n)|)\; e^{j(2\pi/N)fl} \tag{3.12}$$

The cepstral smoothing method then refers to filtering the cepstral coefficients with a lowpass filter function $w_c(f)$ in the cepstral domain and applying the Fourier transformation as shown in eq. (3.13) followed by applying the exp function on its real component.

$$F(f,n) = \exp\Big(\mathrm{Re}\sum_{l=0}^{L}\big(w_c(l)\; C_{(l)}(n)\big)\; e^{-j(2\pi/N)lf}\Big) \tag{3.13}$$

The window function is defined as in eq. (3.14) using some value $l_c < L$ for the cutoff of the lowpass filter.

$$w_c(l) = \begin{cases} 1, & |l| < l_c \\ 0, & \text{else.} \end{cases} \tag{3.14}$$

Using such a window function essentially refers to a truncation of the cepstral coefficients and as this operation yields an even function, eq. (3.13) may also be expressed using only cosine terms [RS07]:

$$F(f,n) = \exp\Big(\sum_{l=0}^{l_c}\big(w_c(l)\; C_{(l)}(n)\big)\cos(\pi lf/N)\Big) \tag{3.15}$$

In [RS07] it has been shown, that the argument of the cosine term can be precomputed and eq. (3.15) then simplifies to a single matrix-vector multiplication to which the exp-function is applied.

3.4 The Phase Vocoder

The Phase Vocoder (PV), introduced in [FG66], extends the classic Dudley Vocoder [Dud39b] with an explicit processing of a signals phase information for an improved reconstruction accuracy when resynthesizing a signal from an analysis. The PV gained much popularity with the availability of the STFT [Por76] as a suitable signal transform which is using efficient FFT algorithms. The analysis result of an STFT yields a series of complex spectra which contain the signals phase and amplitude information at discrete time and frequency locations. The PV algorithm then uses such a time-frequency representation of a signal to apply modifications to its amplitude or phase information before synthesizing a signal from this modified representation. Introductions to the algorithm can be found in [Dol86, Ser97a].

Common applications for the PV are time-stretching and -compression or transposition of a signals pitch [Por81] and their usage in computer music applications has already been studied in [Moo76]. These transformations can be achieved by relocating the analysis frames of the STFT during synthesis or by shifting operations of the spectra along the frequency axis. Such modifications require according adjustments to the phase information of the short-time spectra to ensure the horizontal phase coherence. Though, to properly adjust the phases if sinusoids are contained within a signal, the effect of spectral leakage needs to be taken into account by adjusting the phases of the spectral bins that belong to the same sinusoid in an appropriate manner [Puc95, LD97, LD99a].

The separate treatment of the sinusoidal components of the short-term spectra makes the PV an implicit Sines plus Noise model [LR13]. However, to further preserve the shape of the waveform when transposing a signal in the frequency domain it is necessary to account for what is called vertical phase coherence, which means that the phase relations in-between the sinusoids need to be preserved and according methods have been proposed in [LD99b, Roe10a, Roe10c].

Spectral modifications that effect the magnitudes of the STFT which are essentially equivalent to time-domain filtering of a signal [All77] may also be applied. These transformations may be interpreted as the application of the source-filter model in the PV and an efficient method for preserving the spectral envelope when pitch shifting has been introduced with the True Envelope method in [RR05a].

However, most modifications to the STFT of a signal yield an inconsistent STFT in the sense, that no signal has this modified STFT. The introduction of a computationally efficient method for signal reconstruction from a modified STFT of a signal is given in [GL84]. This method represents a significant landmark in the development and applicability of the Phase Vocoder. A review of more recent methods for signal reconstruction from modified STFTs is given in [SD11].

Eventually, transient smearing due to the analysis window is a well-known artifact within the PV and several methods have been proposed for its reduction

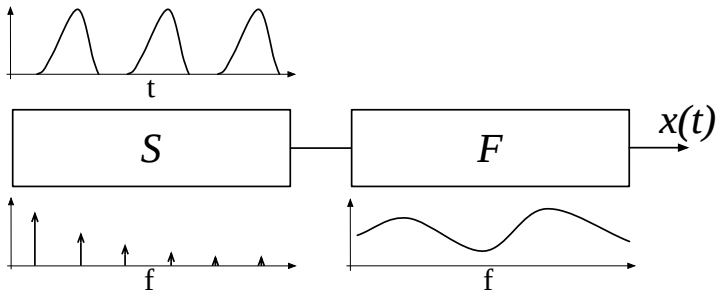


Figure 3.2: An extended source-filter model with a band-limited pulse as source signal shown in time (top) and frequency (bottom) domain and a time-variant filter. The spectrum of the band-limited pulse exhibits lowpass characteristic.

as well [DDS02, Roe03a, Roe03b].

A Phase Vocoder implementation that takes into account all mentioned aspects is a powerful tool for many kinds of signal manipulations. It is computationally efficient and introduces artifacts and signal distortions only on extreme transformations. We will therefore utilize the PV in various ways for our signal transformation strategies.

3.5 Extended Source-Filter Models

The source-filter model of speech production as introduced by Dudley [Dud39a] is a well-known paradigm for many signal transformation and synthesis methods and is therefore being used in a large variety of music signal processing applications [AKZV11]. In the original approach discussed in sec. 3.2, a speech signal is assumed to be created by an ideal pulse source signal with a white spectral distribution and a time-varying filter. This assumption however is very approximate and has hence been revised for various kinds of applications in voice and music signal processing respectively. Within this thesis, we denote models which enhance the classic notion of source-filter models by less idealized and more realistic representations of the excitation component or by introducing additional filter components as extended source-filter models.

3.5.1 For Voice Signals

Within voiced speech, the time-variable and non-linear glottal impedance is responsible for shaping the waveform of the source signal, which is characterized by a band-limited rather than an ideal pulse as within the classic source-filter model. The spectral distribution of such a signal therefore does not exhibit a white or uniform distribution, but shows lowpass characteristics, whereas its frequency slope is determined by the shape of the pulse [Fan79a, Fan79b].

This notion of the excitation led to the introduction of a revised concept of source-filter modeling of human speech with discrete models for the source and filter and analytic models for each to accommodate for their separate contributions to the spectral characteristics of the signal [Fan81]. A schematic

representation of such an extended source-filter model is given fig. 3.2. In contrast to the classic source-filter processing, using an extended approach requires the creation of models for the source as well as the filter and parameter estimation for both components from a signal. Models for the filter have already been discussed in sec. 3.2 and such models are also used within an extended source-filter paradigm.

Various models have been introduced in the literature for the excitation source of voiced human speech and a review of several approaches is given in [Deg10], whereas the Liljencrants-Fant model (LF) [FLL85] and its revised version [FLL95] represent probably the most popular ones [Chi95, dP08, Deg10, May12]. The original LF model is a time-domain, multi-parameter model, whereas its revised version uses a single parameter only to characterize the band-limited pulse of the human speech excitation signal in time-domain.

The usage of analytic models for a signals source and filter allow for sound signal manipulations, which are coherent with the revised concept of human speech production and are therefore assumed to produce synthesis results, which are perceptually more convincing than those using the classic source-filter paradigm. For voice signals, variations to the source allow for high-quality transformations of the speaker or singers age and gender [FODR15] or modifications into tense or breathy voice [DRR11b, DLRR13]. Likewise popular applications are singing voice synthesis [RHRD12], voice conversion [Chi95, dP08, May12] as well as prosodic modifications [May12].

To allow for signal manipulations using an extended source-filter model in an analysis/synthesis framework, the parameters of the model need to be estimated from a signal first. Depending on the selected analytic model descriptions, various approaches and techniques have been proposed for parameter estimation in the literature. Methods based on either inverse filtering [WM07] or minimum/maximum phase decomposition [DBD11] estimate the components in consecutive steps. Other methods use the analytic descriptions of the source and filter to formulate a convex optimization problem to jointly estimate the models parameters from a signal [FM06, DRR10, DRR11a, HRD12, CRYR14, DAAY14].

3.5.2 For Instrument Signals

For instrument sound signals however, no general, parametric model of the excitation source is available due to their diverse range of mechanical designs with substantially different sound generating setups [FR98]. Moreover, even for a particular musical instrument it requires complex numerical solutions to obtain an analytic description of an instruments source signal [VPEK06, Bil09].

To the authors knowledge, the first appearance of an extended source-model for the purpose of representing instrumental sound signals has been published by A. Klapuri [Kla07]. There, a source-filter-decay model has been introduced to represent the sound of a quasi-harmonic musical instrument using three individual filter functions: excitation, resonance and time-dependent loss.

In this approach, the author develops a generic and compact model to describe instrument sounds for the purpose of instrument recognition, coding or sound synthesis. The model does neither represent an exact physical interpretation of a musical instrument as with most physical modeling methods or with voice signal processing nor does it represent only spectral energy distributions

as within classic source-filter methods. The author hence argues for a structured approach that combines high frequency resolution and pitch invariance using functions of harmonic index and frequency to account for specific aspects of instrumental sounds jointly.

The distinction between these two functions is supported by the fact that instrument sounds exhibit some features that may be best described by harmonic index while others are best described by frequency [MEKR11]. This principle is for example supported by the well known property of clarinet sounds, which exhibit strong odd and weak even partials due to its flaring bell construction, promoting the use of harmonic index as an independent parameter. In contrast, the existence of pronounced resonances and formants within sounds of the violin family due to the physical metrics of their corpora promotes the use of additional frequency parameters.

Using the terminology for an extended source-filter model, functions of harmonic index are hence best interpreted as excitation source, whereas functions of frequency refer better to the filter. The source may therefore be interpreted as some vibrating structure like a string or an air column and the filter would refer to the resonating structure of the rest of a musical instrument. In [Kla07], the author furthermore introduces a loss filter, that models some frequency-varying decay for percussively excited instrument sounds.

The three components of that extended source-filter model in [Kla07] may therefore be interpreted as parametric filters, whereas the source is parameterized using harmonic index and the two remaining functions depend on frequency only. All components are further represented by linear models of basis functions using decibel values to obtain a summation formula for the complete model and the model parameters that need to be estimated are represented by the weights of the basis functions.

To obtain the parameters of the model that represents the sound of a whole instrument, the author estimates sinusoidal models from instrument recordings that cover its complete pitch range. The sinusoidal models are setup using harmonic indexes for its partial index up to a maximum of 32 harmonics and all magnitudes are converted to decibel values, neglecting their phases. The parameter estimation is then carried out using a weighted least-squares approach considering all pitches of an instrument and the remaining error will include variations due to dynamic or due to plucking point differences among other errors [Kla07].

A similar approach has been taken by the author of this thesis in [HRBW10] in a pilot study, which preceded the research work of the present thesis. There, an extended source-filter has been established using a source and a filter component only, though the source component incorporated a dependency on the amplitude envelope additionally to the harmonic index to account for temporal energy variations due to an instrument sound signal's attack and release segments. This addition essentially extends the notion of a loss filter by an explicit representation of an instrument's temporal sound characteristics. Both model components have further been represented by B-spline basis functions [dB01] and the basis function's weight parameters have been estimated in a least-squares approach for a single music instrument using a database of recordings covering its complete pitch range.

Summarizing the paradigm of extended source-filter modeling for music sound signal applications, is important to highlight that such extended source-

filter models need to learn their parameters from a whole collection of sounds of a particular instrument. Application of an extended source-filter model for music sound signal applications is hence restricted to available, pre-trained instrument models. Parameter learning though, requires a preceding analysis of a dataset of instrument recordings that cover a desired range of instrument timbres and this analysis needs to yield a signal representation, which is suitable for learning the model's parameters. As with all learning approaches, a remaining modeling error will persist and that error will contain all intrinsic variations of the sounds that are not captured by the model.

In the authors opinion, the two studies [Kla07, HRBW10] justified the applicability of extended source-filter models for music instrument modeling and hence represent the main motivation behind the selected paradigm of the present work.

3.5.2.1 In Music Information Retrieval Applications

The extended source-filter model presented in [Kla07] has extensively been used in various music information retrieval tasks. For the analysis of polyphonic audio in a non-negative matrix factorization as well as deconvolution framework [VK06], instrument recognition in polyphonic audio [HKV09], sound source separation using an expectation-maximization algorithm [KVH10] and for music transcription using again non-negative matrix factorization [KDK13].

Learned instrument models based on such a model within a non-negative matrix factorization framework for source separation are also used successfully in [COVVC$^+$11, COCVCRS13, RSDVC$^+$15] and promising results have also been achieved by the authors of [DDR11, HDR11] in similar setups for obtaining mid-level representations in the former and audio atom decompositions in the latter.

A study comparing some of the approaches above that are using extended source-filter models in non-negative matrix factorization frameworks has recently been published in [CDM14].

3.5.2.2 For Sound Synthesis Applications

To the authors knowledge only two approaches have been published so far which are using extended source-filter models for the purposes of sound synthesis of acoustic music instruments.

In the first approach, the two student researchers E. Maestre and A. P. Carillo together with their respective supervisors and mentors X. Serra and J. Bonada among various others have developed an extended source-filter model using the violin as a case study, which was hence be called the "violin project". They estimated the model parameters in two separate steps starting with an explicit measurement of the violin's resonance characteristics [PCBPV11] and subsequently learning the remaining excitation component using a neural network [CBM$^+$12]. The neural network had been presented Mel-scale based, subsampled spectral data for the signals harmonic and residual component both as functions of particular violin controls. These controls have been the bows transversal position, the bows velocity, the bows acceleration, the bows force and several other specific input parameters which had been measured during dedicated recording sessions with specialized 3D motion-tracking equipment.

These complex model control parameters have been further represented using statistical modeling methods to enable rendering of the particular gestural controls using high-level abstractions [MBB+10]. Sound synthesis using the violin model may eventually be achieved using either original recordings of a violin or by feeding artificially generated sinusoidal and noise signals into the estimated spectral envelopes.

This joint research project eventually led to their respective doctoral thesis' [Mae09, Car09].

In a more recent method, R. Mignot and V. Välimäki have taken a universal approach for an extended source-filter model in a way that their method is applicable to all quasi-harmonic instruments using a unified synthesis scheme [MV14a]. They denote their method ESUS which refers to "Extended Subtractive Synthesis". The ESUS method uses an artificially generated saw tooth signal for synthesis of the harmonic component and a dedicated noise generator for the signals residual. All spectral variations from the generated signals need hence to be applied using the filters of the extended source-filter model. Within the ESUS method this is also done separately for the signals components using independent filters. Therefore, the method identifies three independent filter functions for each component using an instruments sound database that contains all possible pitches of the respective instrument. One overall instrument filter that covers sound features that are equal for all pitches, a tone filter that represents the pitch-dependent variations of the spectral envelopes and a modulation filter to emulate time-dependent variations.

The filters are established using a new spectral envelope estimation technique [MV14d] which they have shown to yield superior approximations than previous methods [MLV13, MV13] and their parameters are estimated using a single representative frame of each instrument recording. The method is eventually made-up for highly efficient sound synthesis and hence they introduced a method for low-order ARMA approximations of the estimated filters to apply sound synthesis in the time-domain only [MV14b, MV14c].

Both discussed approaches eventually prove the applicability of the extended source-filter model paradigm for sound synthesis of acoustic music instrument since either method has shown to yield promising results.

This page is intentionally left blank.

Part II

Expressive Sampling Synthesis

Chapter 4

Arbitrary–Order Multivariate Regression Splines

Introduction

For our purpose of expressively transforming instruments recordings, we are seeking for a suitable model, which either describes the partial amplitude or cepstral coefficient data measurements with respect to some control variables. Such a model needs hence to be able to interpolate amplitude or cepstral values measured at strongly sampled control signals to allow for a synthesis with continuously valued control parameters.

Assuming that the sound signal data has been generated by a deterministic source -the music instrument -and corrupted by additive noise, a model can be constructed in terms of a curve-fitting approach, that approximately fits the data using a smooth function of the control variables. Such a function may be defined in terms of a finite number of unknown parameters that will be estimated from the data [Bis06].

Since the true form of the data generating process is typically unknown, a few assumptions about this process will have to be made in advance.

One reasonable assumption may be the non-linearity of the data with respect to the independent variables. This may easily be justified by the non-linear spectro-temporal characteristics of music instruments playing with varying intensity or pitch. Furthermore, we also aim for support of several independent control parameters within one single model and hence demand a model with multivariate characteristics.

For a possible parameter estimation technique we may shortly consider the regression function to be estimated as the likelihood function of a statistical model and for such, the Maximum Likelihood Estimation (MLE) method is a well-known technique. Under the assumption of a Gaussian distribution for the additive noise, the minimization of the mean of squares of the model with respect to the data arises as a consequence of MLE [Bis06]. Using the method of Least-Mean-Squares (LMS) for parameter estimation hence allows for a non-probabilistic estimation technique, while ensuring statistical inference. The assumption of Gaussian distribution for the additive noise is a standard approach in most data analysis techniques as long as no other knowledge about its distribution is available.

Therefore, we require a parametric, multivariate, non-linear regression model, whose parameters can be estimated in a LMS approach using measured data.

With respect to these considerations, we employ piecewise polynomials in B-form, better known as B-splines [dB01] to create a continuous, multivariate regression model for partial amplitude values that have been measured at distinct, discrete control values. In the literature, B-splines may also be denoted basis-splines [dB01].

B-splines are widely being used in the domain of computer aided design, numerical data analysis and surface fitting [Chu91, Hoe03, Sed14], but have also been used for modeling partial amplitude data trajectories [Roe06].

4.1 B-Splines

B-spline functions belong to the class of linear basis function models. Such models can be considered linear combinations of fixed non-linear functions of some input variables, whereas these non-linear functions are denoted basis functions. A B-spline is hence a piecewise polynomial function and is defined by its B-spline order o and knot sequence s. Every piecewise polynomial of the function has compact support and their linear combination creates the spline. Eq. (4.1) expresses a B-spline in mathematical terms, whereby the spline $g(u)$ is constructed by the sum of basis functions b_p, weighted by some associated factor w_p. Note that within a complete formulation of the B-spline, its order o and knot sequence s would have to appear next its parameter index p, but those have been left our for readability and to avoid confusion.

$$g(u) = \sum_{p=1}^{P} b_p(u) \cdot w_p \tag{4.1}$$

A single basis function is defined to have only small compact support within $s_p \dots s_{p+o}$ in the sense that

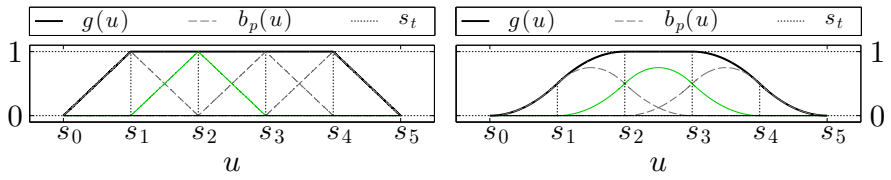
$$b_p(u) = 0 \quad \text{for} \quad u \notin [s_p \dots s_{p+o}] \tag{4.2}$$

whereas the basis functions $b_p(u)$ are normalized in such a way, that their linear combination sums up to 1 as shown in:

$$\sum_{p=\tau}^{\upsilon} b_p(u) = 1 \quad \forall \quad u \in [s_{\tau+o-1}, \dots, s_{\upsilon-o+1}] \tag{4.3}$$

In fig. 4.1 two B-splines are shown using a B-spline order $o = 2$ in the left 4.1(a) and an order $o = 3$ in the right 4.1(b). In both figures a single basis-function is accentuated using green color to illustrate their shape and compact support spanning an amount of segments s equal to its respective B-spline order o.

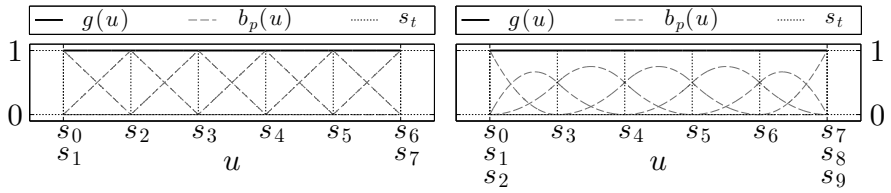
The two subfigures in 4.1 furthermore demonstrates the compact support of every polynomial of the piecewise function as well as the specific property that each polynomial has an order equal to $o - 1$. This follows from the fact, that the B-spline order o refers to the amount of coefficients required to resemble the polynomial rather than its highest degree.



(a) Model using B-splines order $o = 2$ (b) Model using B-splines order $o = 3$

Figure 4.1: Two B-splines with different orders illustrating their basis functions b_p with respect to their knot sequence s and its according linear combination g using $w_p = 1, \forall\, p$

A fundamental property which allows for modeling of its complete domain u using the piecewise approach is denoted knot-multiplicity and is depicted in fig. 4.2, whereas a knot multiplicity of 2 and 3 at the sequence boundaries are shown. Multiplicity of knots hence allows for full domain modeling, whereas its value is dependent on the B-spline order o.



(a) 2nd order B-splines with multiplicity 2 (b) 3rd order B-splines with multiplicity 3

Figure 4.2: Two B-splines with different B-spline orders and knot multiplicity at the domains boundaries which equal their respective orders.

4.2 Parameter Estimation

The use of a linear combination of piecewise polynomials within a B-spline allows for using a linear model to represent non-linear data properties since the models only free parameters are given by their respective weights w_p. In a curve-fitting application, these weights need to be estimated from some given data with respect to a certain criterion and for such, two approaches will be discussed.

Doing curve-fitting using a linear basis function model refers to finding the set of parameters $\mathbf{w} = [w_1, \ldots, w_P]^T$ for which the linear combination of their products with its basis functions fits best to a given set of input data. As is common with most data analysis methods, the amount of input data needs to exceed by far the amount of free parameters to obtain a good estimate for the mean of the data, while assuming a Gaussian distribution for the additive noise. If one considers a set of N univariate input variables $\tilde{\mathbf{u}} = [u_1, \ldots, u_N]^T$ with their corresponding values $\mathbf{y} = [y_1, \ldots, y_N]^T$ with $N \gg P$, the transformation matrix $\mathbf{B}$ of the linear system of equations becomes non–square as can be seen

in eq. (4.4) and a solution using a simple matrix inversion technique can not be expressed due to this overdetermination.

The tilde symbol for the vector of univariate input variables is used, because in sec. 4.3 we will introduce a vector $\mathbf{u}$ to refer to multivariate input variables.

$$\mathbf{B} = \begin{bmatrix} b_1(u_1) & \cdots & b_P(u_1) \\ \vdots & \ddots & \vdots \\ b_1(u_N) & \cdots & b_P(u_N) \end{bmatrix} \in \mathbb{R}^{N \times P} \tag{4.4}$$

Though, the technique for minimization of the least-mean-square error allows for finding an approximate solution given the input data and can be phrased as in eq. (4.5) using matrix notation.

$$E(\mathbf{w}; \mathbf{y}|\tilde{\mathbf{u}}) = \frac{1}{2}||\mathbf{y} - \mathbf{B}\mathbf{w}||_2^2 \tag{4.5}$$

The factor $\frac{1}{2}$ is only used for convenience when processing the gradient and will hence appear always throughout this thesis when gradients need to be derived for parameter estimation.

Eventually, finding the weight vector $\mathbf{w}$ which minimizes E with respect to the given input data $(\mathbf{y}|\tilde{\mathbf{u}})$ refers to solving the least-means-square criterion and since $\mathbf{B}\mathbf{w}$ is linear, a global optimum for the minimization can be assured.

However, the estimation of the free parameters of a piecewise polynomial in B-form using the least-means-squares error criterion can be done in various different ways, though we only consider two.

4.2.1 Direct Method

The direct-form solution for linear systems of equations with non-square transformation matrixes can be derived by using the gradient of eq. (4.5) with respect to the weight vector $\mathbf{w}$ which is:

$$\frac{\partial E(\mathbf{w}; \mathbf{y}|\tilde{\mathbf{u}})}{\partial \mathbf{w}} = -\big(\mathbf{y} - \mathbf{B}\mathbf{w}\big)\mathbf{B}^T \tag{4.6}$$

Since we assured a global minimum for the cost function, setting the gradient to zero as in eq. (4.7) and solving for $\mathbf{w}$ leads to the normal equations for the least squares problem shown in eq. (4.8).

$$0 = \mathbf{B}^T\mathbf{y} - \mathbf{B}^T\mathbf{B}\mathbf{w} \tag{4.7}$$

$$\mathbf{w} = \mathbf{B}^\dagger\mathbf{y} \tag{4.8}$$

whereas $\mathbf{B}^\dagger$ represents the Moore-Penrose matrix, also known as pseudo inverse of $\mathbf{B}$ which is defined to be:

$$\mathbf{B}^\dagger = (\mathbf{B}^{\mathrm{T}}\mathbf{B})^{-1}\mathbf{B}^{\mathrm{T}} \tag{4.9}$$

The pseudo-inverse can be regarded a generalization of the notion of the matrix inverse to non-square matrices [Bis06].

4.2.2 Iterative Method

With increasing amounts of input data and model complexity represented by an increased amount of free parameters, not only the size of the system of linear equations may become unfeasible large for calculating the pseudo-inverse, but it also turns out, that the transformation matrix $\mathbf{B}$ becomes increasingly sparse with increasing P.

This property can be easily seen from eq. (4.2), as for a single datum u_i only an amount of B-splines equal to the B-spline order o will exhibit a value that is not zero. The density of $\mathbf{B}$ can hence be expressed as the ratio between the B-spline order o and the overall amount of free parameters P:

$$\text{density}\big(\mathbf{B}\big) = \frac{o}{P} \tag{4.10}$$

For solving large and sparse systems of linear equations, the method of Conjugate Gradients (CG) can be considered the most prominent one, since it does not require to store the whole matrix at once and its algorithmic complexity is linear with the amount of non-zero entries in the matrix $\mathbf{B}$, both in terms of memory and calculations [She94]. The method has originally been developed by Hestenes and Stiefel [HS52] for linear systems with symmetric matrixes, but can also be applied to non-symmetric as well as non-linear systems.

The CG method is typically applied in an iterative manner, which means that, starting with some possibly random initial weight vector $\mathbf{w}_0$, the values of the parameter vector get being updated in each iteration step until the conditions of an abort criterium are fulfilled. A highly general update rule for many iterative methods can be expressed as in eq. (4.11), where t denotes the iteration index and $\Delta\mathbf{w}_t$ the search direction.

$$\mathbf{w}_{t+1} \leftarrow \mathbf{w}_t + \Delta\mathbf{w}_t \tag{4.11}$$

Often applied abort criteria are either a minimum change of the parameter vector in-between consecutive iterations assuming convergence of the algorithm or until a maximum amount of iterations is reached, though a combination of both also seems reasonable. When using a quadratic error function in conjunction with the CG method, convergence is guaranteed when the number of iterations equals the total number of free parameters [HS52, She94].

Within the CG method, the search direction in the parameter space is calculated, such that it is mutually conjugate to all the previous directions and a line-search technique estimates its optimal step size. The line-search component of the CG method uses second order information without explicitly calculating all second derivatives of the error function, which makes the CG method belong to the class of second-order optimization methods. It can be expressed as in eq. (4.12) [DHS00], whereas β_t needs to be determined such that $\Delta\mathbf{w}_t$ becomes conjugate to all previous directions.

$$\Delta\mathbf{w}_t = -\frac{\partial E(\mathbf{w}_t; \mathbf{y}|\tilde{\mathbf{u}})}{\partial \mathbf{w}} + \beta_t \Delta\mathbf{w}_{t-1} \tag{4.12}$$

It can be seen from eq. (4.12), that the descent direction at iteration t takes the negative gradient plus a component along the previous direction, which makes it analogous to calculating a “smart“ momentum [DHS00]. This momentum is proportional to β_t for which several formulae have been proposed

in the literature. Among others, the Hestenes–Stiefel [HS52], Fletcher–Reeves [FR64] as well as the Polak–Ribiere [PR69] formulae belong to the most notable ones.

There are however various other modifications to the basic algorithm for an improved estimation of the search direction. The so called Scaled Conjugate Gradient (SCG) [Mø93] substitutes the time-consuming line-search algorithm by a scaling of the learning parameter, which solely depends on the success of the previous learning iteration and makes the algorithm fully-automated and independent of manual parameter adjustments. It eventually utilizes the Hestenes–Stiefel equation for calculating the β_t coefficient.

For these reasons we have decided to use the SCG method to estimate the free parameters of all B-splines used throughout this thesis.

For clarity reasons and later use, we rewrite the cost function to be optimized from (4.5) by the SCG learning strategy in non-matrix notation:

$$E(\mathbf{w};\mathbf{y}|\tilde{\mathbf{u}}) = \frac{1}{2}\sum_{i=1}^{N}\big(y_i \;-\; g(u_i)\big)^2 \tag{4.13}$$

$$= \frac{1}{2}\sum_{i=1}^{N}\Big(y_i \;-\; \sum_{p}^{P} b_p(u_i)\cdot w_p\Big)^2 \tag{4.14}$$

Using this notation, the gradient for a single weight w_p used by the SCG method can hence be expressed as in eq. (4.15), which is mathematically equivalent to eq. (4.6).

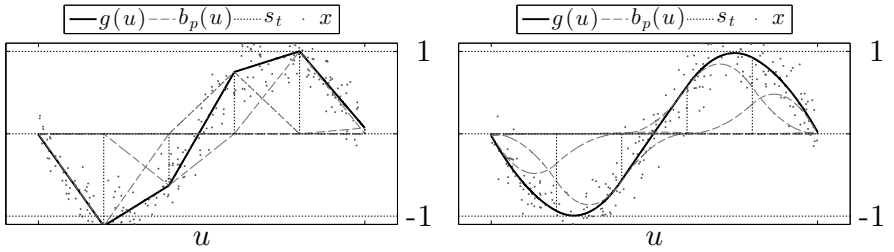
$$\frac{\partial}{\partial w_p}E(\mathbf{w};\mathbf{y}|\tilde{\mathbf{u}}) = -\sum_{i=1}^{N}\Big(y_i - \sum_{p'=1}^{P} b_{p'}(u_i)\cdot w_{p'}\Big)b_p(u_i) \tag{4.15}$$

When using the SCG method for estimating the parameters $\mathbf{w}$ of a B-spline function, we discovered, that using zero-valued initial weights w_p for the iterative learning process is advantageous for ensuring the convergence of the algorithm. Using a random initialization even if bounded reasonably occasionally led to numerical instabilities while performing the parameter estimation.

4.2.3 A Simple Example

For the purpose of demonstrating a curve-fitting application using B-spline functions, fig. 4.3 shows two examples, whose parameters have been estimated using the SCG technique, while utilizing the gradient given in eq. (4.15). In both examples, input data values u_i are drawn from a uniform distribution bounded to $[-\pi \dots \pi]$, whereas its target values y_i are generated using the sine function and additive noise with variance $\sigma = .2$. It can easily be observed that the B-spline function $g(u)$ using an order $o = 2$ allows for a piecewise linear fit, whereas the B-spline function with an order $o = 3$ constitutes a piecewise squared fit.

The major difficulty in this curve-fitting approach lies in determining the B-spline order o as well as in the selection of an appropriate amount of knots and their respective placement within the domain prior to the estimation of the free parameters. Various approaches have been discussed in the literature,



(a) Model using 2nd order B-splines (b) Model using 3rd order B-splines

Figure 4.3: Two B-spline models

which either require a thorough statistical analysis of the input data or training of several configurations for the piecewise function [dB01].

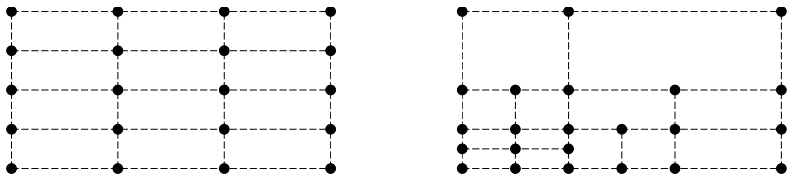
For large systems for which training of various models with different specifications may impractical and statistical analysis non-trivial, it seems reasonable to consider the incorporation of a priori knowledge about the domain distribution of the data as well as assumptions about the properties of the data distribution to be fitted. Such pre-training analysis reflections will done in place later in this thesis.

4.3 Multivariate Variables

The joint representation of data with several independent variables using B-splines refers to the extension of the 1-dimensional curve-fitting problem to the application of fitting surfaces to multi-dimensional data. Such surfaces may also be called hyperplanes due to their possibly high-dimensional nature and various approaches have been proposed in the literature to represent such planes using B-splines.

De Boor introduced multivariate splines as a generalization of univariate splines in many independent variables [dB76], which have been studied and improved by several other authors [Dah80, Hoe82, DM83]. Multivariate splines are very flexible and hence highly common in industrial surface design, though their implementation is very difficult and to the authors knowledge, there exists no free or open-source implementation. Alternatively, simple tensor-product B-splines (TPB) are a natural and attractive choice for surface-fitting in two or more dimensions [dB01], as they can easily be created by an expansion of the univariate case and all univariate identities and algorithms generalize easily [Hoe03].

Their most important difference can be seen in that the simple tensor-product approach requires rectilinear knot distributions, while the multivariate spline method supports non-uniform rectangular partitioning. Figure 4.4 shows examples for knot distribution patterns of both approaches to emphasize the potential drawback of the TPB method. Assuming a strongly localized feature of the data in the lower left corner of the figures and a much smoother approximate in the upper right, the multivariate spline model is likely to obtain a better fit in terms of its cost value and with less parameters than the TPB.



(a) Rectilinear knot grid (b) Nonuniform knot grid

Figure 4.4: Possible knot distribution patterns in 2D space. Left figure shows uniform partitioning as with tensor-product B-splines and figure to the right shows a potential non-uniform grid as possible when using multivariate B-splines. In the figures, each intersection refers to a B-spline parameter.

Some noteworthy extensions for an improved local refinement of the rectilinear mesh of the standard TPB are T-Splines [Sed14] and Hierarchical [FB88, Kra94, Kra97] or Truncated Hierarchical B-splines [GJS12, KGJ12]. These are however either protected by patents as with T-Splines [Sed07], miss some reference implementation as for Hierarchical B-splines or have just been introduced too recently for serious consideration within this research work.

However, in situations where it may not harm to have preferred directions in the approximated surface parallel to its axes[dB01], using the fast and easy to implement simple tensor-product solution is not severely limiting the approach for surface fitting. Therefore, as our measurement data comes with discrete control parameters and hence presumably strong directional features along its axes, we utilize the classic tensor-product approach for modeling the multivariate data.

Therefore, we may now express the B-splines domain variable as the vector $\mathbf{u} = [u_1, \ldots, u_D]^T$ with D dimensions, whereas each vector element refers to a single axis of the multivariate data.

We can hence formulate the tensor-product B-spline for an arbitrary amount of dimensions D with mutually independent B-spline configurations as in eq. (4.16) again omitting the B-spline order o and knot sequence s for readability.

$$g(\mathbf{u}) = \sum_{p_1=1}^{P_1} \cdots \sum_{p_D=1}^{P_D} \left(\prod_{d=1}^{D} b_{d,p_d}(u_d) \right) w_{p_1,\ldots,p_D} \tag{4.16}$$

Accessing the mutually independent, univariate B-spline configurations along each dimension is achieved using d as a primary index for b, while the latter index p_d refers to a certain B-spline polynomial along dimension d. The weight parameter in eq. (4.16) now has also changed appropriately to a coefficient tensor with an amount of dimensions d equal to that of the independent variable $\mathbf{u}$.

Parameter estimation using gradient based approaches however, requires a vector rather than a tensor for the weights as well as for its spline configuration. For a multivariate B-spline model the weight tensor $w_{p_1,\ldots,p_D}$ needs hence to be reshaped to a vector to be usable by the SCG method. For this we redefine the weight vector as follows:

$$\mathbf{w} := [w_{1,\dots,1}, \dots, w_{P_1,\dots,1} \quad , \dots, \quad w_{1,\dots,P_D}, \dots, w_{P_1,\dots,P_D}]^T \tag{4.17}$$

while the tensor-product of the univariate B-spline functions $\prod_{d=1}^{D} b_{d,p_d}(u_d)$ gets defined according to:

$$\begin{aligned}\mathbf{b} := [&b_{1,1}(u_1) \cdot \dots \cdot b_{D,1}(u_D), \dots, b_{1,P_1}(u_1) \cdot \dots \cdot b_{D,1}(u_D) \quad , \dots, \\ &b_{1,1}(u_1) \cdot \dots \cdot b_{D,P_D}(u_D), \dots, b_{1,P_1}(u_1) \cdot \dots \cdot b_{D,P_D}(u_D)]^T\end{aligned} \tag{4.18}$$

We reused the variable identifier $\mathbf{w}$ for the vector of the B-spline weight parameters from the univariate case, because all the introduced equations remain valid also for multivariate variables after vectorization. Moreover, the definitions above for $\mathbf{w}$ and $\mathbf{b}$ will reduce to the standard B-spline formulations introduced in sec. 4.2 in the case of $D = 1$. This allows to vastly simplify eq. (4.16) by using b_p to refer to the $p-$th tensor-product within $\mathbf{b}$ and using w_p to refer to its respective tensor weight. The simplified equation for the multivariate B-spline model using tensor-products can hence be written as in eq. (4.19), whereas it only differs from the univariate formulation in eq. (4.1) by the usage of a vector of independent variables instead of a one-dimensional variable.

$$g(\mathbf{u}) = \sum_{p=1}^{P} b_p(\mathbf{u}) \cdot w_p \tag{4.19}$$

One may however note, that different orderings of the variables in eq. (4.17) and eq. (4.18) may be chosen for the vectorization, though the weight tensor and the tensor-product of B-spline function values need to be transformed in an analogous manner to ensure their consistency.

It shall also be noted, that the lengths of the vectors $\mathbf{b}$ and $\mathbf{w}$ are equal to the product of the lengths of all univariate parameter vectors used for the tensor-product as shown in (4.20) hence yielding the new P.

$$\dim(\mathbf{b}) = \dim(\mathbf{w}) = \prod_{d=1}^{D} P_d = P \tag{4.20}$$

Therefore, special caution is advised when adding new parameters or even another dimension, as the actual parameter space increases exponentially with each parameter of its univariate components.

It may also be noted, that the sparsity of the multivariate system using tensor-products also increases substantially with its amount of dimensions, because every single data point will only yield non-zero B-spline values for each univariate spline used to create the tensor-product equal to the respective B-spline order. The amount of non-zero values hence increases linearly with the amount of univariate B-splines, while the amount of the parameters of the TPB increases exponentially. We may express the density of the vectorized tensor-product B-spline $\mathbf{b}$ using o_d to refer to the B-spline order used along dimension d.

$$\text{density}(\mathbf{b}) = \frac{\sum_{d=1}^{D} o_d}{\prod_{d=1}^{D} P_d} \tag{4.21}$$

For estimating surfaces from multivariate data we may now consider the matrix $\mathbf{U} = [\mathbf{u}_1, \ldots, \mathbf{u}_N]$ being the set of multivariate input variables .Their corresponding values $\mathbf{y}$ are defined as in sec. 4.2 and we redefine the cost function E for multivariate data similarly to the univariate case as follows:

$$E(\mathbf{w}; \mathbf{y}|\mathbf{U}) = \frac{1}{2} \sum_{i=1}^{N} \big(y_i \quad - \quad g(\mathbf{u}_i)\big)^2 \tag{4.22}$$

and also its gradient shown in eq. (4.23) differs from the first derivative of the univariate model in eq. (4.15) only in the use of the multivariate variable.

$$\frac{\partial}{\partial w_p} E(\mathbf{w}; \mathbf{y}|\mathbf{U}) = -\sum_{i=1}^{N} \Big(y_i - \sum_{p'=1}^{P} b_{p'}(\mathbf{u}_i) \cdot w_{p'}\Big) b_p(\mathbf{u}_i) \tag{4.23}$$

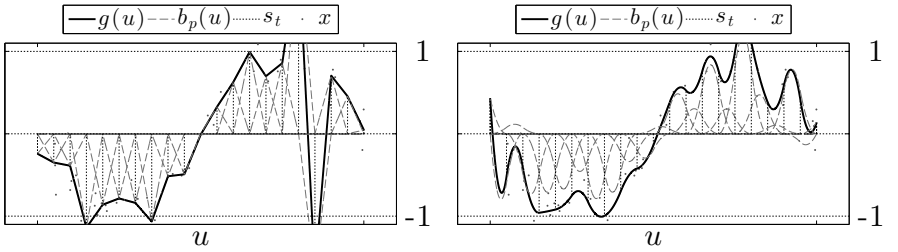
The equations (4.22) and (4.23) hence allow to perform surface fitting of multivariate data in arbitrary dimensions using tensor-products of univariate B-splines while considering the proposed vectorization of the respective weight tensor and B-spline tensor-product shown in def. (4.17) and (4.18).

4.4 Regularization

One well–known problem in parameter learning, which arises in most machine learning tasks and also when estimating parameters for B-splines is called overfitting. Generally spoken, overfitting occurs if the models complexity is relatively high compared to the available amount of observational data. In such a case, the model will instead of successfully generalizing the data, overly adapt to its noisy fluctuations. As we will later see, overfitting represents a significant issue, when representing the data using B-splines for the purpose of instrument modeling as done in this thesis and hence will be discussed thoroughly here.

In terms of B-spline based regression modeling, overfitting occurs when the the number of knots is relatively large, such that the estimated curve shows more variation than can be justified by the data. For the purpose of demonstrating the effect of overfitting occurring in data modeling using B-spline functions, fig. 4.5 shows two examples similar to fig. 4.3, but with an increased amount of knots and decreased amount of data used for the parameter estimation. The data has been generated using the same procedure as above and it can be observed from the figure, that the sine function is not well approximated by the B-spline function, instead it overly follows the additive noise and hence generalizes badly.

In the examples in fig. 4.5, the data is still drawn from a uniform distribution in the domain u whose limits equal the domain boundaries of the B-spline function. As we will see later, this assumption will not hold true in our application for partial data modeling, where data might be drawn from an interval whose limits represent only a subspace of the B-splines domain. Similarly, the distribution of the data might exhibit gaps or empty regions. Both cases will eventually lead to little or no support for some of the free parameters of the B-spline function and a singular or ill–conditioned system of normal equations will result. The fitted curve will then show heavy fluctuations or oscillations.



(a) Model using 2nd order B-splines (b) Model using 3rd order B-splines

Figure 4.5: Two B-spline models

In the literature, two approaches for smoothing the estimated curve fitted by the B-spline function are proposed, whereas both techniques have been proposed for univariate variables only. The methods can be regarded regularization methods since as such they introduce a penalty term to the least-means-square optimization shown in eq. (4.24) which extends the cost function E by an additional term $\mathcal{R}$ to adjust for the smoothness of the fit. It hence requires the models free parameters to penalize appropriately with respect to these parameters. Throughout this thesis, we will denote functions that optimize jointly for data and additional constraints as objective functions $\mathcal{O}$.

The scaling parameter λ represents a hyper parameter, which balances between smoothness and data fit and as such will not be estimated automatically from the data, but needs to be adjusted prior to the parameter optimization.

$$\mathcal{O}(\mathbf{w};\mathbf{y}|\mathbf{U}) = E(\mathbf{w};\mathbf{y}|\mathbf{U}) + \lambda\mathcal{R}(\mathbf{w}) \tag{4.24}$$

Both methods target for smoothing the interpolation of the data, rather than allowing for extrapolation of the curve into domain ranges without any data. However, we will analyze these methods briefly to introduce a new approach that incorporates ideas from both and proposes a solution to our specific problem including support for multivariate variables.

The first method is known as Smoothing Spline [Rei67, dB01] and introduces the regularization term (4.25) to the parameter estimation. It takes the square of the second derivative of the B-spline function, integrated over all data points N to penalize for curvature.

$$\int_{u_i} \Big(\sum_{p}^{P} b_p''(u_i) w_p \Big)^2 du \tag{4.25}$$

Applying this regularization scheme to any iterative method requires the calculation of second derivative B-spline terms for all data points u_i at each iteration. Therefore, we can easily estimate that the smoothing spline adds computational cost linear with the amount of data N and similar to the cost of the unconstrained optimization.

The alternative method is denoted penalized B-splines also known as P–splines [EM96] and utilizes the term shown in (4.26). It is using higher-order,

finite differences of the coefficients of adjacent B-splines for measuring smoothness.

$$\sum_{p=z+1}^{P} (\triangle^z w_p)^2 \tag{4.26}$$

The parameter z in eq. (4.26) refers to the order of the difference allowing to constrain for slope, curvature or even higher orders. In [EM96], the author states, that the difference penalty is a good discrete approximation to the integrated square of the z–th derivative. The finite differences $\triangle^z w_p$ for $z = 1$ and $z = 2$ can be derived from the formula for derivatives of B-splines [dB01]:

$$\triangle^1 w_p = w_p - w_{p-1} \tag{4.27}$$

$$\triangle^2 w_p = w_p - 2w_{p-1} + w_{p-2} \tag{4.28}$$

As can be seen from eq. (4.26) and (4.27), (4.28), this approach only requires the B-spline coefficients itself for constraining the optimization, instead of taking all observations N into account. Hence, while still assuming $N \gg P$, we may conclude that this method works much more efficiently than smoothing splines.

There is however a significant drawback, as the finite differences shown in eq. (4.27) and (4.28) have been derived for equidistant knot sequences [EM96]. In case of non-uniformly distributed knot sequences, the derivation looses much of its beauty.

We will hence introduce a new scheme, which incorporates ideas from both approaches and further extends them to multivariate variables. The term shall be established generically in terms of the derivative order z to support for slope or curvature penalties and arbitrary knot sequences. An explicit B-spline formulation will be used as done for the smoothing spline, though we employ a multivariate mesh grid to sample the B-spline surface at J fixed positions which are defined in a similar manner to the multivariate input variables $\mathbf{V} = [\mathbf{v}_1, \ldots, \mathbf{v}_J]$, whereas each column refers to a single multivariate sampling position $\mathbf{v}_j = [v_1, \ldots, v_D]^T$ to obtain a discrete approximation of the surface. These positions can be also regarded virtual data points and they are established using equidistant sampling positions along all univariate B-spline domains, whereas $P < J \ll N$ to ensure a similar computational efficiency as with P-splines, but also a good approximation of the spline surface. A few sampling points in-between two adjacent knots have shown to deliver a fairly good approximation of the curve.

Eq. (4.29) shows the regularization as we define it. The term is established with respect to a selected dimension d along which the $z-$derivative is being taken from the surface. All other axes remain unaltered to allow for slope or curvature penalties independent for each axis. The B-spline formulation uses the simplified tensor-product expression introduced in sec. 4.3 and hence the definition of the regularization is generic in terms of the amount of surface dimensions and with respect to the desired penalty along a selected dimension.

$$\mathcal{R}^{(z,d)}(\mathbf{w}) = \frac{1}{2} \sum_{j=1}^{J} \eta^{(z,d)}(\mathbf{v}_j) \Big| \sum_{p=1}^{P} \left(b_p^{(z,d)}(\mathbf{v}_j) w_p \right) \Big|^2 \tag{4.29}$$

We further introduce a multivariate scaling polynomial $\eta^{(z,d)}(\mathbf{v}_j)$ to locally amplify the regularization term. The term therefore allows for an adjustment of the strength of smoothness depending on the independent variable of the B-spline function and therefore needs to be set separately for each selected penalty hence its parameterization with respect to z and d. The local control of the impact of the regularization can be a useful tool, if the data is not distributed uniformly within its domain or if the variance of the additive noise varies. In both cases, overfitting may occur locally, if the models complexity is constant or can not be adapted as needed within its domain and hence a locally amplified regularization can be reasonable. Note that the coefficients of the polynomial need to be set manually and hence a priori knowledge about the data distribution and/or noise properties is required. Using a first order polynomial with its constant term being set to zero is therefore a good initial strategy to keep the amounts of non-automatically adjusted parameters low.

Using the regularization term proposed in eq. (4.29) now makes it mandatory to adjust the λ parameter in such way, that it becomes easy to balance the data fit and smoothness of the function. Recalling the notation of virtual data points for the sampling positions of the regularization, we propose to use the ratio of the $l1-$norm of the squared B-spline function values of the data points and the $l1-$norm of the squared B-spline function values of the virtual data points which has been derived along d according to z as shown in eq. (4.30). We thus obtain a value which represents the ratio of the impact of the data and of the regularization mesh grid for the optimization procedure.

$$\lambda^{(z,d)} = \lambda_0^{(z,d)} \frac{\left|\left| \sum_i^N \left(\sum_p^P b_p^{(z,d)}(\mathbf{u}_i) \right)^2 \right|\right|_1}{\left|\left| \sum_j^J \left(\sum_p^P b_p^{(z,d)}(\mathbf{v}_j) \right)^2 \right|\right|_1} \tag{4.30}$$

The scaling parameter $\lambda_0^{(z,d)}$ now allows to balance between data fit and smoothness of the surface mutually independent for each axes. Setting $\lambda_0^{(z,d)} = 0$ refers to no smoothing, whereas $\lambda_0^{(z,d)} = 1$ refers to smoothness being equally important as the data. One may note, that the scaling parameter in eq. (4.30) only depends on the data and the selected mesh grid and hence needs to be processed only once before the actual parameter estimation unless both stay unchanged. The regularization term however, requires the current B-spline parameter vector and therefore needs to be calculated at every iteration of the parameter estimation procedure.

We eventually introduce a new objective function in eq. (4.31) in extension to the previously introduced eq. (4.24), which allows for multiple regularizations at the same time by using a simple linear combination of regularization terms that are parameterized by z and d.

$$\mathcal{O}(\mathbf{w}; \mathbf{y}|\mathbf{U}) = E(\mathbf{w}; \mathbf{y}|\mathbf{U}) + \sum_{z,d} \lambda^{(z,d)} \mathcal{R}^{(z,d)}(\mathbf{w}) \tag{4.31}$$

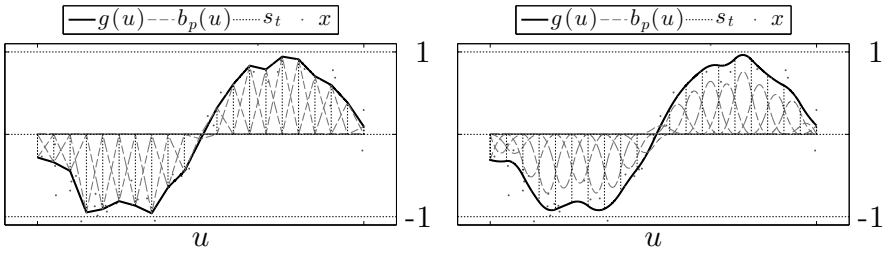
Using for example $(z, d) = (1, 2)$ and $(z, d) = (2, 2)$ simultaneously for the parameter estimation allows to constrain slope and curvature along dimension 2 of the surface at the same time. Such a setting would allow extrapolations that will fade smoothly to constant values and hence prevents possible overshooting or oscillations, when free parameters of the B-spline functions are undetermined given the data. Though, in order to optimize a B-spline surface with respect

to some data and the just introduced regularization, its gradient needs still to be determined. Therefore, the first derivative of the objective function in eq. (4.31) is derived as follows:

$$\frac{\partial}{\partial w_p}\mathcal{O}(\mathbf{w};\mathbf{y}|\mathbf{U}) = \frac{\partial}{\partial w_p}E(\mathbf{w};\mathbf{y}|\mathbf{U}) + \sum_z \lambda^{(z,d)}\frac{\partial}{\partial w_p}\mathcal{R}^{(z,d)}(\mathbf{w}) \tag{4.32}$$

The first term at the right hand side has already been solved in eq. (4.23) and only the first derivative for the regularization term needs to be determined:

$$\frac{\partial}{\partial w_p}\mathcal{R}^{(z,d)}(\mathbf{w}) = \sum_{j=1}^{J}\eta^{(z,d)}(\mathbf{v}_j)b_p^{(z,d)}(\mathbf{v}_j)\Big(\sum_{p'=1}^{P} b_{p'}^{(z,d)}(\mathbf{v}_j)w_{p'}\Big) \tag{4.33}$$



(a) Model using 2nd order B-splines (b) Model using 3rd order B-splines

Figure 4.6: Two adapted complex B-spline models using only few data with applied regularization.

Using the objective function in eq. (4.31) and its first derivative in eq. (4.32) can then be used to apply constrained optimization as described. Fig. 4.6 shows two adapted B-spline functions using the same B-spline configuration and data as in fig. 4.5, but with additional regularization. Both models, the second as well as the third order model have been constrained using jointly a first and second order regularization with $\lambda_0 = .3$.

4.5 Preconditioning

Assuming a large system $\mathbf{y} = \mathbf{B}\mathbf{w}$ with thousands of free parameters, an iterative method like SCG may still tend to be slow to converge, as convergence for a quadratic function is guaranteed only for an amount of iterations equal to the amount of free parameters. This is especially the case if the condition number of the system is relatively high. One known method to improve the efficiency in terms of the amount of iterations required for convergence is called Preconditioning [BBC+94, She94] and it is said to be the determining ingredient for the success of an iterative method, as in many real–world problems, the reliability of an iterative method for a linear system much more depends on the quality of the preconditioner than that of the particular parameter estimation technique [Saa03].

Preconditioning refers to a technique for improving the condition number of a matrix [She94] using any form of implicit or explicit transformation of the original linear system into one, which has the same solution, though is likely to require fewer steps to converge than with the original system using an iterative method [Saa03]. It may be interpreted as an attempt to stretch the quadratic form of the linear system, hence error function, to make it appear more spherical. A perfect preconditioner would allow an iterative method to converge within one iteration [She94]. However, preconditioning can be done in various different ways [BBC+94, Saa03] and its computational complexity may vary a lot, hence for yielding an effective improvement over the use of the untransformed system, its operational cost needs to be considered wisely.

An explicit form of preconditioning can be considered any scaling of the linear system [Saa03] with the mentioned aim for making it easier to solve. We therefore apply a scaling to the Hessian matrix of the system, such that its diagonal elements are all equal to 1. Though the actual value is fairly arbitrary, scaling the diagonal entries of the Hessian matrix to make them all identical, refers to the transformation of the surface spanned by the quadratic form from an ellipsoid with possibly highly different properties along each axis to a rather perfect sphere. Such a transformed error surface is then assumed to be a lot easier to solve using any CG method [She94, Saa03].

To apply such a scaling, we substitute the B-spline weighting parameter w_p of the original system by 2 new parameters as in eq. (4.34), whereas the system may already be a simplified tensor-product B-spline surface. The variable c_p refers to the preconditioner coefficient, which shall scale the error surface as desired. The parameter $\widetilde{w}_p$ then represents the B-spline weighting parameter of the transformed system.

$$w_p = \widetilde{w}_p \cdot c_p \tag{4.34}$$

In case of the simplified tensor-product B-spline surface, c_p as well as $\widetilde{w}_p$ essentially refer to the $p-$th value of their respective vectorized tensors shown in eq. (7.19) and (7.20).

$$\widetilde{\mathbf{w}} = [\widetilde{w}_{1,\dots,1}, \dots, \widetilde{w}_{P_1,\dots,1} \quad , \dots, \quad \widetilde{w}_{1,\dots,P_D}, \dots, \widetilde{w}_{P_1,\dots,P_D}]^T \tag{4.35}$$

$$\mathbf{c} = [c_{1,\dots,1}, \dots, c_{P_1,\dots,1} \quad , \dots, \quad c_{1,\dots,P_D}, \dots, c_{P_1,\dots,P_D} \ ]^T \tag{4.36}$$

The parameter vector of the original system $\mathbf{w}$ now represents the element–wise product of $\widetilde{\mathbf{w}}$ and $\mathbf{c}$.

Now, prior to the iterative estimation of the weighting parameters $\widetilde{w}_p$ of the transformed system, the preconditioner coefficients c_p and new derivatives for the transformed system need to be determined. We therefore write the gradient of the objective functions with respect to the weighting parameter of the transformed system as follows:

$$\frac{\partial}{\partial \widetilde{w}_p} \mathcal{O}(\mathbf{w}; \mathbf{y}|\mathbf{U}) = \frac{\partial}{\partial \widetilde{w}_p} E(\mathbf{w}; \mathbf{y}|\mathbf{U}) + \sum_{z,d} \lambda^{(z,d)} \frac{\partial}{\partial \widetilde{w}_p} \mathcal{R}^{(z,d)}(\mathbf{w}) \tag{4.37}$$

For the sake of completeness, the cost function E with the substituted weight parameter is:

$$E(\mathbf{w};\mathbf{y}|\mathbf{U}) = \frac{1}{2}\sum_{i=1}^{N}\Big(y_i - \sum_{p=1}^{P} b_p(\mathbf{u}_i)\,\widetilde{w}_p\,c_p\Big)^2 \tag{4.38}$$

and its first derivative as required for the parameter estimation procedure of the transformed system hence yields:

$$\frac{\partial}{\partial\widetilde{w}_p}E(\mathbf{w};\mathbf{y}|\mathbf{U}) = -\sum_{i=1}^{N}\Big(y_i - \sum_{p'=1}^{P} b_{p'}(\mathbf{u}_i)\,\widetilde{w}_{p'}\,c_{p'}\Big)b_p(\mathbf{u}_i)\,c_p \tag{4.39}$$

The regularization is of course also part of the linear system and hence subject for preconditioning as well and the weight parameters get substituted there as well. The penalty term therefore becomes:

$$\mathcal{R}^{(z,d)}(\mathbf{w}) = \frac{1}{2}\sum_{j=1}^{J}\eta^{(z,d)}(\mathbf{v}_j)\Big|\sum_{p=1}^{P}\big(b_p^{(z,d)}(\mathbf{v}_j)\widetilde{w}_p\,c_p\big)\Big|^2 \tag{4.40}$$

and its first derivative can hence be written as follows:

$$\frac{\partial}{\partial\widetilde{w}_p}\mathcal{R}^{(z,d)}(\mathbf{w}) = \sum_{j=1}^{J}\eta^{(z,d)}(\mathbf{v}_j)\Big|\sum_{p'=1}^{P}\big(b_{p'}^{(z,d)}(\mathbf{v}_j)\widetilde{w}_{p'}\,c_{p'}\big)\Big|b_p^{(z,d)}(\mathbf{v}_j)\,c_p \tag{4.41}$$

To eventually perform parameter estimation of the constrained, preconditioned and multivariate system, the coefficients $\mathbf{c}$ still need to be determined. As we aim for scaling the diagonal entries of the Hessian matrix of the overall system to reshape the error surface to become more sphere–like, the second derivative of the objective function (4.42) will be set equal to 1 and solved for all entries of $\mathbf{c}$.

$$\frac{\partial^2}{\partial\widetilde{w}_p^2}\mathcal{O}(\mathbf{w};\mathbf{y}|\mathbf{U}) = \frac{\partial^2}{\partial\widetilde{w}_p^2}E(\mathbf{w};\mathbf{y}|\mathbf{U}) + \sum_{z,d}\lambda^{(z,d)}\frac{\partial^2}{\partial\widetilde{w}_p^2}\mathcal{R}^{(z,d)}(\mathbf{w}) = 1 \tag{4.42}$$

The second derivative of the error function E with respect to the weighting parameters of the transformed system $\widetilde{w}_p$ yields:

$$\frac{\partial^2}{\partial\widetilde{w}_p^2}E(\mathbf{w};\mathbf{y}|\mathbf{U}) = \sum_{i}^{N}\big(b_p(\mathbf{u}_i)\,c_p\big)^2 \tag{4.43}$$

whereas the second derivative of the regularization term gives:

$$\frac{\partial^2}{\partial\widetilde{w}_p^2}\mathcal{R}^{(z,d)}(\mathbf{w}) = \sum_{j=1}^{J}\eta^{(z,d)}(\mathbf{v}_j)\Big|\big(b_p^{(z,d)}(\mathbf{v}_j)\,c_p\big)\Big|^2 \tag{4.44}$$

To determine the coefficients for preconditioning c_p, the two derivations of the center term within eq. (4.42) are substituted with eq. (4.43) and (4.44) respectively and eq. (4.42) is subsequently solved for all c_p, yielding:

$$c_p = \Big(\sum_i^N \big(b_p(\mathbf{u}_i)\big)^2 + \sum_{z,d} \lambda^{(z,d)} \sum_{j=1}^{J} \eta^{(z,d)}(\mathbf{v}_j) b_p^{(z,d)}(\mathbf{v}_j) \Big)^{-\frac{1}{2}} \tag{4.45}$$

As can be seen from eq. (4.45), the preconditioner coefficient c_p only depends on the values of the B-splines of the input data as well as the B-spline values of the virtual data points used for regularization and will therefore be constant for all iterations of the parameter estimation procedure. Its calculation hence needs only to be done once with a computational cost similar to a single iteration of the algorithm. It is therefore very likely, that the use of the preconditioner will facilitate a significant performance improvement over the use of the original system.

4.6 Conclusion

In this chapter we have introduced a parametric, multivariate regression model using B-splines with a parameter estimation technique based on least-squares-optimization in a conjugate gradient framework. We have further proposed a new regularization scheme to support variable order smoothing of the multivariate regression function leading to an objective function balancing data fit and desired smoothness of the fit using hyper-parameters. Eventually, we have developed a preconditioning method for an accelerated convergence of the conjugate gradient parameter estimation algorithm.

Therefore, this multivariate regression model allows for non-linear modeling of possibly high-dimensional data with scalable model complexity while preserving the advantageous properties of a linear model.

We may conclude that the model supports for partial amplitude and cepstral coefficient modeling with respect to several independent control variables simultaneously, while allowing for instrument specific configurations to adjust for their presumably varying data distribution properties.

Chapter 5

Signal Representation

Introduction

To eventually enable high-level control for sound transformations of recorded instrument data, a link needs to be established to connect low-level signal properties with control parameters, that are perceptually relevant and appropriate in terms of musical expression. This linkage will be established using a general model for musical instruments that is capable of representing sound features which are related to a set of selected controls. In the current chapter, we hence introduce the signal models and control parameters that are required to establish such general instrument models.

However, instrument sound signals whose sound features shall be captured by the model and therefore available for synthesis need to be collected first. We may call this part of our proposed technique for expressive sampling synthesis the analysis stage, as the selected recordings of a musical instrument will be transformed into a representation, which is suitable for learning a model and certain control parameters will be either created or estimated from these signals to represent the instrument sound's features as functions of some meaningful controls.

Within the work of this thesis, we have used sample libraries of musical instrument recordings, which had been created for the purpose of digital sound synthesizers. All libraries consisted of monophonic recordings of quasi-harmonic music instruments. For all instruments, separate sound files had been available for each pitch playable by that particular instrument. Each sound file hence yields the recording of a single-note without vibrato, tremolo, glissando, crescendo or alike. Furthermore, each pitch had been available at at least three different intensity levels representing playing styles from pianissimo up to fortissimo with one or more intermediate steps, though all sound files have been normalized with some headroom, leading to a general loss of information about the physical signal intensity when recording as no reference levels were included with the datasets.

Eventually, all sound files of the sound set libraries have been annotated manually by their creators with their respective pitch and intensity information.

Within this chapter, we will describe the applied signal model and how it gets transformed into a specific signal representation that is suitable for modeling an instruments timbre regarding a certain set of control parameter

also introduced in the present chapter.

5.1 Sound Signal Representation

As discussed in ch. 3, various spectral models with analysis support and according techniques have been proposed in the literature for the representation and transformation of recorded instrument sounds. Within this thesis, we utilize the explicit Sines plus Residual signal model introduced with the Spectral Modeling Synthesis technique to estimate the harmonic and residual components from every single recording contained within a selected dataset. For the purpose of describing the signal's spectral properties, we represent the Sines plus Residual model from eq. (3.3) using frequency domain variables without loss of generality:

$$X(f,n) \;=\; X_h(f,n) \;+\; X_r(f,n) \tag{5.1}$$

The estimation is preceded by a fundamental frequency estimation using the monophonic version of algorithm [YRR10] or our new proposed fundamental frequency estimator described in chapter A depending on the type of instrument. As every single recording of the available datasets is accompanied by its respective pitch information, the fundamental frequency estimation algorithm is adjusted to search within a very limited range of possible values and a pitch-adaptive analysis window length is used for optimal frequency resolution. The precise determination of the fundamental frequency allows a much more accurate estimation of a signal's higher harmonics in the subsequent analysis, as already small errors of the fundamental frequency may lead to mismatches of partials at higher harmonic indexes.

The fundamental frequency trajectory $f_0(n)$ is hence used as an input parameter for the estimation of the harmonic sinusoids.

5.1.1 Harmonic Sound Signal Representation

Each sinusoid of the deterministic signal component $X_h(f,n)$ can also be described by its respective frequency domain variable $X_{(k)}(f,n)$, which with respect to eq. (3.4) may be expressed as follows:

$$X_h(f,n) \;=\; \sum_{k}^{K} X_{(k)}(f,n) \tag{5.2}$$

However, to obtain the partials trajectories, we estimate the instantaneous parameters of the harmonic sinusoids using the QIFFT procedure described in sec. 3.3.1 and the according partial tracking method, which is utilizing the previously estimated function of the fundamental frequency. The amount of analyzed partials K per frame n is limited only by the signal's samplerate and some lower threshold. Eventually, this yields the trajectories of the instantaneous parameters of the deterministic component $a_{(k)}(n)$, which will furthermore be transformed to decibel values:

$$A_{(k)}(n) = 20 \cdot \log_{10}(a_{(k)}(n)) \tag{5.3}$$

The harmonic component will also be resynthesized using the method described in sec. 3.3.3 yielding the time-domain signal $x_h(t)$.

5.1.1.1 Special Case: Impulsively Excited Sounds

In the case of impulsively excited signals however, an additional procedure for an improved estimation of the partial trajectories is performed using the WLS approach introduced in sec. 3.3.2.

In the example of piano sounds, we observed for low pitches, that the harmonic analysis using pitch-dependent analysis windows yields only a very little amount of frames of data during the signal's attack phase. This is due to the percussive amplitude envelope with a steep attack slope and the very low fundamental frequencies, which require very long analysis windows. This leads to quite poor signal approximations and may eventually yield synthesis results with audible artifacts or recognizable perceptual differences. It may furthermore become difficult to estimate the signal's attack properties for our instrument model, if only little data is available.

We hence employ the WLS method succeeding the QIFFT method for a signal segment, which comprises all frames from the onset until a few frames behind the signals maximum amplitude. A more elaborate description of how the signal's temporal amplitude envelope is estimated and how a frame is determined as being the starting point of the signal's release segment, which determines the end of the enhanced analysis is given sec. 5.2.3.1.

As the WLS method allows to reduce the length of the analysis window to a minimum of 2 cycles of the fundamental period, it yields a much more precise estimate of the temporal evolution of the partial's amplitude trajectories. We only assume the amplitude of the partial trajectories within the signal's attack phase to exhibit rapid changes and hence keep the interpolated phase trajectories from the QIFFT method and do not update their respective instantaneous frequencies.

The release segment of sounds with a percussive amplitude envelope is however much smoother and therefore the WLS method is not applied to it.

5.1.2 Residual Sound Signal Representation

The residual component $x_r(t)$ of a sound signal $x(t)$ is obtained by subtracting the synthesized version of the estimated deterministic component $x_h(t)$ from the original recording as proposed in [Ser89] and shown in eq. (5.4), yielding a signal which could be described as filtered white noise plus transients.

$$x_r(t) \;=\; x(t) \;-\; x_h(t) \tag{5.4}$$

To obtain a representation, which is suitable for modeling the residual characteristics of a musical instrument, we utilize an envelope model that approximates the time-varying spectral distribution of the residual by a smooth time-dependent function. This has already been proposed in [Ser89] and in [RS07], a liftered version of the signal real cepstrum has been proposed for this purpose. In sec. 3.3.5 we gave a description of how such an envelope can be obtained, which essentially refers to the application of the source-filter model for noise signals with an appropriate envelope model.

The time-varying characteristics of the residual signal will hence be represented by a likewise time-varying envelope model using the bivariate cepstral coefficients function $C_{(l)}(n)$ using eq. (3.12) with a fixed amount of coefficients as well as fixed STFT parameters for all sounds. The analysis window length is set to 23ms (1024 samples at a samplerate of 44.1kHz) and the amount of cepstral coefficients L is set to 16.

For sound synthesis and transformation, the envelope model will be used to apply transformations onto the residual signal obtained using eq. (5.4) and hence the residual signal needs to be kept. This is due to the fact, that the assumption of the residual being filtered white noise is too loose. In [CKD+13], the authors have shown, that synthesis using filtered white noise yields audible differences and hence keeping only the envelope for synthesis is not sufficient.

5.2 Control Signal Representation

The control parameters refer to the set of expressive control variables required to enable sound signal transformations using the instrument model. Expressivity within the sound synthesis approach presented in this thesis shall be achievable by varying the expressive control parameters using standard digital control devices with support for the MIDI protocol and shall not require additional models of expressive musical gesture parameters. Even though this severely limits the possible amount of expressive control parameters, it guarantees its applicability of the model for most western acoustic instruments without the need for further parameter modeling methods.

Therefore, the possible compromises between universality of the approach and amount of expressive control as discussed in ch. 2 is bounded by the available set of parameters within the MIDI protocol and control parameters that can be estimated robustly from the audio signals [Wan01] or taken from supplied meta information. We are hence using a simple and generic model of musical gesture [CW00] and some derived variables to obtain a set of control variables, which allow parametric sample-based sound synthesis while retaining universality for most quasi-harmonic instruments.

The perhaps most decisive properties for the perception of acoustic musical sounds are its pitch and loudness [Moo12] which is also why most digital sound synthesizers implement these parameters for real-time control.

Using the pitch as a control parameter is reasonable as it is known from instrument physics that many acoustic instruments exhibit strong spectral variations for varying pitch values which do not just result from frequency shifts of the spectral content [FR98].

The loudness is typically denoted note intensity and is expressed in musical terms ranging from pianissimo (*pp*) to fortissimo (*ff*) and variations to this control variable also reasonably effect the resulting sound of acoustic instruments [FR98]. There are however variations to the signal intensity which may refer to the signals attack or release phase which is not represented by the categorial variable of note intensity, though, especially the signal's attack portion is known to be a decisive feature of acoustic instruments [Hel70]. We therefore also require a control parameter for a signal's temporal energy fluctuations and as we explain below use the instantaneous intensity as control variable as well.

Therefore, for the purpose of generality and to account for what we assume to be the most prominent sound signal parameters we use the three control parameters: pitch, global note intensity and instantaneous intensity.

5.2.1 The Musical Pitch as a Control Parameter

Within our method we make use of the musical pitch obtained from the manual annotations of the instrument recordings encoded using the MIDI protocol. There, a possible range of 127 discrete pitches can be represented using discrete semitone steps implying a logarithmic fundamental frequency scale due to the western equal temperament system. We will refer to a signal's pitch by using the time-dependent variable $P(n)$ as in (5.5), though the pitch is assumed to be constant for all sound files but shall allow for pitch variations during synthesis.

$$P(n) \in \{1, \ldots, 127\} \quad \wedge \quad P(n) = \text{const} \ \ \forall n \tag{5.5}$$

An additional index parameter to refer to a specific sound files is omitted here for clarity and readability, as all variables in the current chapter refer to a single sound file only if not stated otherwise. In later chapters we will be more explicit if necessary.

5.2.2 The Global Note Intensity as a Control Parameter

Information about a signals musical intensity is typically lost in the digital domain as soon as the signals get normalized to obtain a maximum of dynamic range for storing the sound data using any quantized number format. Therefore, we make use of the musical intensity delivered within the annotated meta information that accompanies the sound data sets. The signal's note intensity will be denoted its global intensity throughout this thesis using the variable I_g and likewise to the signals pitch be encoded using the MIDI protocol. Since the global intensity is typically referred to by using musical terms like mezzo-forte (mf) which refers to an intensity in the center between the two already introduced extreme values pp and ff. Representing these categories as MIDI velocity values requires their association with respective values on the MIDI scale. We assign the pp note intensity to the MIDI velocity value 1 and the ff category to a respective maximum value of 127. Intermediate values are obtained using linear interpolation and hence mf becomes 64 on the MIDI scale. Other non-linear mappings could have been considered as well [Dan06], though without proof of consistency with actual instrument characteristics.

$$I_g(n) \in \{1, \ldots, 127\} \quad \wedge \quad I_g(n) = \text{const} \ \ \forall n \tag{5.6}$$

5.2.3 The Instantaneous Energy as a Control Parameter

In the literature a large variety of methods for the estimation of a sound signal's temporal evolution have been proposed including segmentation strategies to identify a sound signal's characteristic temporal segments.

The perhaps most popular method for envelope estimation is represented by the instantaneous intensity [CR11], which can be calculated using the frame-wise root-mean-square (RMS) of the signal itself or within any invertible spectral domain. The resulting envelope may though yield strong fluctuations de-

pending on the used analysis frame size. Other methods use the low-pass filtered and half-wave rectified version of the signal [BDA+05] or the energy decay curve for temporal envelope estimation [Smi10a], while also linear predictive coding techniques have been proposed [AE03]. An amplitude-centroid trajectory method has been proposed [Haj96, Haj98] as well as an approach using the True-Envelope estimation method [CBR10, CR11]. Methods focusing on the segmentation of the temporal envelope using any of the above methods can be found in [Haj96, Haj98, Pee04, Jen99].

However, most methods are either computationally very demanding [CBR10, CR11], not suitable for a subsequent temporal segmentation [Smi10a] or designed for a specific kind or class of musical instruments [Jen99]. We have therefore decided to utilize the instantaneous energy of the deterministic and residual signal component respectively as control variable and utilize a simple though robust envelope segmentation scheme to identify three regions of interest in a naïve approach.

We will first refer to the normalized and to decibel values transformed short-time energy by harmonic local intensity $I_{l,h}$ and residual local intensity $I_{l,r}$ respectively, assuming the signal components have previously been separated successfully. The local intensities for both components are processed frame-wise using the same analysis window length L and hopsize R from the analysis described in sec. 5.1 as follows:

$$E_\gamma(n) = \Big| \sum_{m=0}^{L-1} \big(x_\gamma(nR+m)\big)^2 \Big|^{\frac{1}{2}} \tag{5.7}$$

$$I_{l,\gamma}(n) = 20 \cdot \log_{10}\Big(\frac{\mathrm{E}(n)}{\max(\mathrm{E}(n))}\Big) \tag{5.8}$$

Using the same analysis window length L and hopsize R from the sinusoidal and residual analysis serves two purposes: It first guarantees a fairly smooth temporal envelope, as the window length has been set to cover at least 5 fundamental periods, which will yield an envelope that does not oscillate with the signals fundamental frequency value but rather follows its mean energy over several period and second, all control data will be time-aligned with the audio data streams. Eventually, the variable $\gamma \in \{h, r\}$ in eq. (5.7) and (5.8) is used to refer to the respective signal components.

5.2.3.1 Segmentation Scheme for Discrete Modeling

Additionally to subtle changes within a signals sustain segment, the envelope represented by the normalized level function $I_{l,\gamma}(n)$ of a signal will exhibit increasing and decreasing slopes for a signals attack and release phase respectively. Both segments are equally characterized by values below the functions maximum which is set to 0dB due to the normalization, though music instrument signals can be assumed to exhibit significantly different signal characteristics for these segments. Especially the signals attack portion is known to be decisive for instrument sound recognition [CLA+63] and hence promotes to treat these signal segments separately with respect to their own distinctive characteristics. We will therefore account for a signals temporal variations by a segmentation of its respective temporal envelope $I_{l,\gamma}(n)$, which simultaneously

allows to learn discrete models for separate instrument sound features, while equally enabling smooth sound synthesis using these discrete models.

First, the temporal envelope function $I_{l,\gamma}(n)$ for the harmonic as well as residual signal component of a continuously driven instrument like woodwinds, brass or bowed strings is assumed to exhibit a raising part, referring to the signals attack-phase, a fairly constant phase, which represents its sustained segment and a decreasing slope, typically denoted as its release. This scheme can be denoted attack-sustain-decay (ASR) model and is shown to the left of fig 5.1. Music instruments which are impulsively or percussively excited exhibit a different kind of temporal envelopes, which differs from the ASR model by a missing sustain segment, hence denoted attack-release (AR). A schematic of such an envelope is given to the right of 5.1.

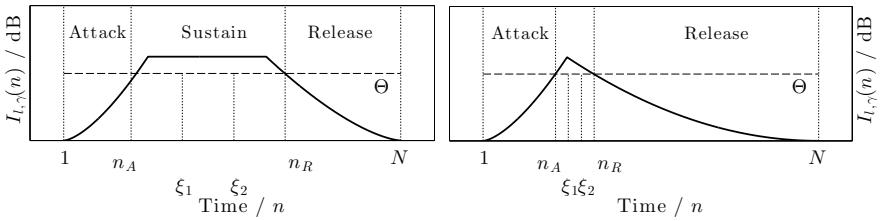


Figure 5.1: Temporal segmentation scheme for sustained (left) and impulsive (right) instrument signals. Signal regions Attack, Sustain and Release are indicated (top).

To enable distinct modeling of a signals attack and release properties, we employ a simple, though robust threshold based segmentation scheme also depicted in fig. 5.1, yielding estimated positions for the end of the attack n_A and the begin of the release n_R segment of either a continuously or impulsively driven instrument signal using a threshold Θ. The threshold for continuously driven instruments is calculated as in eq. (5.9) using the median of the temporal envelope in the center of the recording to obtain a rough level estimate of the signals sustain segment.

$$\Theta = \mathrm{median}(I_{l,\gamma}(n')) - 6, \quad n' = [1 + \frac{N}{3}, \ldots, N - \frac{N}{3}] \tag{5.9}$$

The threshold for impulsively excited signals is simply set to -6dB since the envelope's maximum is known to be at 0dB due to the normalization. The first and last intersections of the envelope with the constant threshold Θ are eventually interpreted as the end of the attack n_A and the begin of the release n_R location respectively. It may be noted, that the estimated position of the begin of the release phase of impulsively excited signals n_R is used to determine the range for improved sinusoidal analysis using the WLS technique, which involves an update of $I_{l,h}(n)$ using the smaller analysis window lengths for this signal segment.

These rough estimates are further being used in eq. (5.10) and (5.11) to calculate the boundaries $\xi_{\gamma,1}$ and $\xi_{\gamma,2}$.

$$\xi_{\gamma,1} = \frac{1}{3}\left(2\, n_{\mathrm{A}} + \quad n_{\mathrm{R}} \right) \tag{5.10}$$

$$\xi_{\gamma,2} = \frac{1}{3}\left(\quad n_{\mathrm{A}} + 2\, n_{\mathrm{R}} \right) \tag{5.11}$$

Two indicator functions $\tau_{\gamma,s}$ (5.12) and (5.13) are then created, representing overlapping temporal segments, referring either to a signals attack-to-sustain for $s = 1$ or its sustain-to-release region if $s = 2$ for each signal component γ.

$$\tau_{\gamma,1}(n) = \begin{cases} 1 & , \ 1 \leq n \leq \xi_{\gamma,2} \\ 0 & , \ \text{else} \end{cases} \tag{5.12}$$

$$\tau_{\gamma,2}(n) = \begin{cases} 1 & , \ \xi_{\gamma,1} \leq n \leq N \\ 0 & , \ \text{else} \end{cases} \tag{5.13}$$

These indicator functions will be used to learn two discrete, but unified models for a single instrument to represent its signal characteristics with respect to the temporal differences of its respective sound properties.

5.2.3.2 Fusion Scheme for Continuous Sound Synthesis

However, for sound synthesis the discrete models for the signals attack-to-sustain and the signals sustain-to-release need to be connected smoothly and we will therefore create two fusion functions which will perform a linear interpolation in-between the two segments with respect to the locations $\xi_{\gamma,1}$ and $\xi_{\gamma,2}$ at which we have divided the signal.

$$\varphi_{\gamma,1}(n) = \begin{cases} 1 & , \ 1 \leq n < \xi_{\gamma,1} \\ (n - \xi_{\gamma,2})(\xi_{\gamma,1} - \xi_{\gamma,2})^{-1} & , \ \xi_{\gamma,1} \leq n \leq \xi_{\gamma,2} \\ 0 & , \ \xi_{\gamma,2} < n \leq N \end{cases} \tag{5.14}$$

The second fusion function for the sustain-to-release segment then simply becomes:

$$\varphi_{\gamma,2}(n) = 1 - \varphi_{\gamma,1}(n) \tag{5.15}$$

These two fusion functions will later simplify the synthesis method as they control the linear cross-fade between the estimated attack-to-sustain characteristics and the sustain-to-release.

5.2.4 The Idealized Partial Frequencies as Control Parameter

Within our data sets of instrument recordings used within the work of this thesis, we assume them to exhibit single-notes only without any significant modulation of its fundamental frequency. Their partial frequency values can hence be assumed to be rather precisely at their ideal locations with respect to either eq. (3.10) or eq. (3.11) depending on the kind of musical instrument. We hence assume that, without loss of generality and accuracy, we may

approximate the time-dependent partials frequency trajectories by their non-varying ideal frequency values for the purpose of model parameter estimation as well as amplitude prediction. This approximation only limits the approach of parameter estimation to instrument sounds with constant pitch only, but not necessarily model predictions for sound synthesis, as we may easily switch to time-varying values ones we created the model.

It is important to note that the partial frequency approximation is only applied for the control parameters. Sound synthesis of the partials in general always requires their actual frequency trajectories to retain the high-quality of the sounds. To distinguish the approximated partial frequency values from their values which have been estimated from the real signals we introduce the vector $\hat{\boldsymbol{f}}$ as defined in eq. (5.16) to refer to their idealized values with respect to their likewise ideal fundamental frequency obtained from the sounds MIDI pitch value.

$$\hat{\boldsymbol{f}} = [\hat{f}_{(1)}, \ldots, \hat{f}_{(K)}]^T \in \mathbb{R}^{K \times 1} \tag{5.16}$$

Eq. (5.16) uses either the harmonic series from eq. (3.10) or the series for inharmonic sounds from eq. (3.11) using an inharmonicity coefficient, which had been obtained using the proposed new method in chapter A.

5.2.5 The Combined Sets of Control Parameters

To eventually unify notations and simplify some of the math used within the next chapters, we establish some a vector/matrix based notation system for the introduced control variables. The time-dependent control parameters $P(n)$, $I_g(n)$ as well as $I_{l,(}n)$ become collected into a single matrix as follows:

$$\boldsymbol{\Theta}_\gamma = \begin{bmatrix} P(1) & , \ldots , & P(N) \\ I_g(1) & , \ldots , & I_g(N) \\ I_{l,\gamma}(1) & , \ldots , & I_{l,\gamma}(N) \end{bmatrix} \in \mathbb{R}^{3 \times N} \tag{5.17}$$

When later referring to a vector containing pitch, global intensity and local intensity information of a single frame n, we will make use of $\boldsymbol{\Theta}_\gamma(n)$ which is defined as:

$$\boldsymbol{\Theta}_\gamma(n) = [P(n), I_g(n), I_{l,\gamma}(n)]^T \tag{5.18}$$

The two temporal segmentation functions for the attack-to-sustain and the sustain-to-release segments respectively get also be combined into a single matrix:

$$\boldsymbol{\tau}_\gamma = \begin{bmatrix} \tau_{\gamma,1}(1) & \cdots & \tau_{\gamma,1}(N) \\ \tau_{\gamma,2}(1) & \cdots & \tau_{\gamma,2}(N) \end{bmatrix} \in \mathbb{R}^{2 \times N} \tag{5.19}$$

The same is being done for two fusion functions:

$$\boldsymbol{\varphi}_\gamma = \begin{bmatrix} \varphi_{\gamma,1}(1) & \cdots & \varphi_{\gamma,1}(N) \\ \varphi_{\gamma,2}(1) & \cdots & \varphi_{\gamma,2}(N) \end{bmatrix} \in \mathbb{R}^{2 \times N} \tag{5.20}$$

This allows to establish our final control signal representations $\boldsymbol{\Psi}_\gamma$ for the harmonic and residual signal components $\gamma = h$ and $\gamma = r$ respectively as:

$$\boldsymbol{\Psi}_h = \{\boldsymbol{\Theta}_h \ , \ \boldsymbol{\tau}_h \ , \ \boldsymbol{\varphi}_h \ , \ \hat{\boldsymbol{f}} \ \} \tag{5.21}$$

$$\boldsymbol{\Psi}_r = \{\boldsymbol{\Theta}_r \ , \ \boldsymbol{\tau}_r \ , \ \boldsymbol{\varphi}_r \quad \} \tag{5.22}$$

The control variables for the harmonic and residual component hence only differ by the additional partial frequency vector $\hat{\boldsymbol{f}}$ for the latter.

When referring to control variables in a general way, we make use of $\boldsymbol{\Psi}$, representing the union of the two sets in eq. (5.21) and eq. (5.22) respectively, whereas when using $\boldsymbol{\Psi}_\gamma(n)$ we will refer to all function values of $\boldsymbol{\Theta}_\gamma$, $\boldsymbol{\tau}_\gamma$ and $\boldsymbol{\varphi}_\gamma$ at the specified analysis frame n.

5.3 Conclusion

This yields the time-domain signals $x_h(t)$ and $x_h(t)$ required for our proposed expressive synthesis as well as the signal representations used by our instrument modeling approach. These are the partial amplitude trajectories $A_{(k)}(n)$ and the time–varying cepstral coefficients $C_{(l)}(n)$ and a well-defined set of control variables required for the creation of the models as well as for performing parametric sample-based sound synthesis.

Chapter 6

The Instrument Model

Introduction

For an intuitive control over expressive features of a musical instrument sound, we initially establish individual source-filter models for the harmonic as well as the residual signal component shown in eq. (6.1) and schematically in fig. 6. The sources $\bar{X}_\gamma(f,n)$ are assumed to be whitened versions of their respective spectral signal components $X_\gamma(f,n)$, whereas the filters $F_\gamma\big(f,\boldsymbol{\Psi}_h(n)\big)$ shall be parametric with respect to their according control variables $\boldsymbol{\Psi}$. The required procedure to obtain the whitened source signals will be introduced in chapter. 10.

$$\begin{aligned} X(f,n) \quad &= \quad \bar{X}_h(f,n)\cdot F_h\big(f,\boldsymbol{\Psi}_h(n)\big) \\ &+ \quad \bar{X}_r(f,n)\cdot F_r\big(f,\boldsymbol{\Psi}_r(n)\big) \end{aligned} \tag{6.1}$$

This approach allows to independently control the harmonic and residual contributions by individual filters with their particular parameterizations, though to obtain sound transformations which are coherent with the instrument's sound characteristics, the filter functions then need to represent the features of their respective components with respect to the their particular set of control parameters $\boldsymbol{\Psi}_\gamma$.

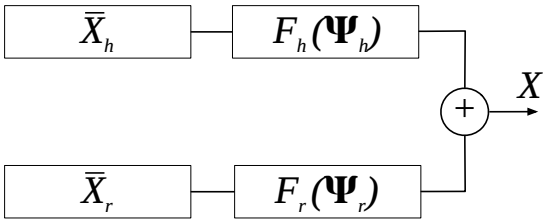


Figure 6.1: The proposed Extended Source Filter model for the harmonic component and an Envelope model based on a classic source filter. Frequency and time variables have been excluded for clarity.

The two respective filter models will be developed in the next two sections, whereas their internal representation will be presented in the next chapter in

a generic manner. Model parameter estimation as well as model selection will hence be presented thereafter.

6.1 Harmonic Model

As has been thoroughly described in sec. 3.5.2, the harmonic source of a musical instrument exhibits certain features which are best represented by their harmonic index while others should be represented by their frequency value. Therefore, for a model that shall describe those harmonic properties of a musical instrument we use two discrete components which are either parameterized by partial index k or by partial frequency $f_{(k)}$.

We may thus express the function for single partial amplitude $\hat{a}_{(k)}$ of a quasi-harmonic instrument as the product of a function of harmonic index $S_{(k)}$ and of a function of partial frequency $R(f_{(k)})$:

$$\hat{a}_{(k)} = S_{(k)} \cdot R(\hat{f}_{(k)}) \tag{6.2}$$

From the assumptions introduced in sec. 3.5.2, the function of partial index $S_{(k)}$ may refer to the modes of vibration of a particular instrument hence its excitation source, while the latter $R(\hat{f}_{(k)})$ represents the contribution of the whole rest of the instrument, which may be roughly approximated by the resonating structure of its corpus.

It is however advantageous to represent the partial amplitude function in the decibel domain, as the model may then be represented using a linear basis which will vastly simplify the optimization procedure later. Furthermore, optimization in log-domain will also be perceptually more reasonable [Kla07]. Eq. (6.2) can hence be rewritten in decibel domain as in eq. (6.3) with $\hat{A}_{(k)}$ representing the modeled partial amplitude in decibel.

$$\hat{A}_{(k)} = S_{(k)} + R(\hat{f}_{(k)}) \tag{6.3}$$

We do not introduce new variables for the dB domain variables of the two filter components as their linear counterparts from eq. (6.2) will not reappear in this thesis.

While still assuming, that $R(\hat{f}_{(k)})$ may refer to the resonance characteristic of an instrument, we may derive its independence of most expressive interactions. Hence, for parametric control using the control variables $\mathbf{\Theta}_h$ from eq. (5.17), we may conclude, that only the excitation source is subject of performative control and hence $S_{(k)}$ needs to be established with respect to $\mathbf{\Theta}_h$, which essentially makes it a tri-variate function, depending on the pitch $P(n)$ as well as global and local intensity $I_g(n)$ and $I_{l,h}(n)$, respectively.

Furthermore, since $\mathbf{\Theta}_h$ includes the local intensity $I_{l,h}$ as function of time and as the signals attack and release segments are equally characterized by intensity values below 0dB, we furthermore establish 2 mutually independent partial amplitude functions for each harmonic index to represent their temporal properties separately.

Hence, we formulate the parametric, extended source filter model for a single partial amplitude with respect to the control variables $\mathbf{\Psi}_h$:

$$\hat{A}_{(k,s)}(\mathbf{\Psi}_h) = S_{(k,s)}(\mathbf{\Theta}_h) + R(\hat{f}_{(k)}) \tag{6.4}$$

The temporal segmentation $\boldsymbol{\tau}_h$ and fusion functions $\boldsymbol{\varphi}_h$ also contained within $\boldsymbol{\Psi}_h$ are not yet used for the representation of the partial amplitudes, but the former will be required for parameter estimation whereas the latter is needed for sound synthesis.

6.2 Residual Model

The residual, non–deterministic component of quasi–harmonic instrument sounds may be caused through fundamentally different physical mechanisms. Hence, the noise signal may for example contain blowing or bowing noise in case of wind or string instruments, respectively. Though, the residual signal may also contain the impulsive sound of the striking hammer within piano sounds or the sound of the plectrum which is exciting the string within guitar sounds.

Hence, to universally support a wide variety of musical instruments, we assume all expressive manipulations to the residual sound component as being best performed by transformations of its spectral envelope, which can efficiently be represented by a limited amount of cepstral coefficients $C_{(l)}(n)$ as explained in more in details in sec. 3.3.5 and 5.1.1.

We therefore establish $\hat{C}_{(l)}(n)$ as presented in eq. (6.5) by mutually independent functions H for each single cepstral coefficient, whereas they are due to the control signal $\boldsymbol{\Theta}_r$ and similar to the harmonic model, separately represented for the signal's attack and release phase indicated using variable s.

$$\hat{C}_{(l,s)}(\boldsymbol{\Psi}_r) = H_{(l,s)}(\boldsymbol{\Theta}_r) \tag{6.5}$$

The model $\hat{C}_{(l,s)}(\boldsymbol{\Psi}_r)$ for all l and both segments s may be regarded a classic source-filter model as it represents a single spectral envelope only and is established in an analogous manner to the excitation component of the harmonic model.

6.3 Conclusion

The instrument model as presented in this chapter using discrete models for the deterministic and residual signal components is depicted in fig. 6.3, whereas an extended source-source filter model is used for the signal representation of the harmonic component and a classic source-filter approach is employed for the signal's residual.

The source and resonance components of the harmonic model as well as the envelope model for the residual component will be subject of modeling the sound characteristics instrument according to its inherent possibilities and limitations. All three components need to be represented using an internal description capable of representing the intrinsic, multi-variate features of their respective signal representation $A_{(k)}(n)$ and $C_{(l)}(n)$ as functions of their respective controls $\boldsymbol{\Theta}_h(n)$ and $f_{(k)}$ as well as $\boldsymbol{\Theta}_r(n)$ respectively.

A general model for representing multi-variate trajectories has been presented in ch. 4 and its application to the specific case of the instrument model is presented in the next chapter.

The two signal representations for the components however do neither yield spectral filter envelopes or coefficients for time domain-based filters and hence,

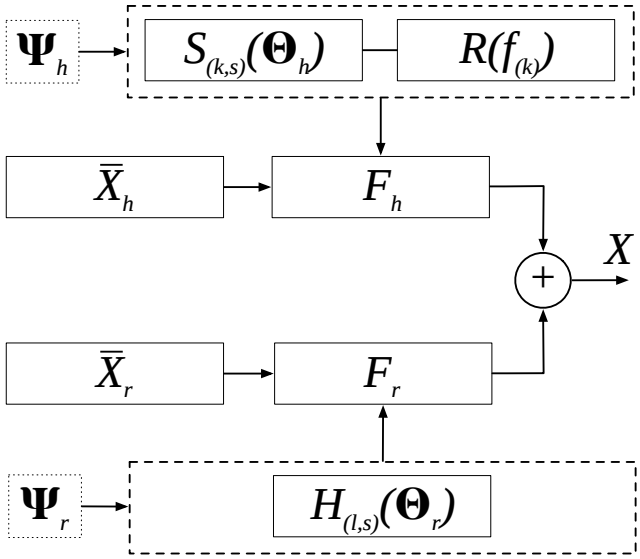


Figure 6.2: The proposed extended-source-filter model for the harmonic component and an envelope model based on a classic source filter approach.

their estimations according to some control parameters need to be transformed into representation applicable to actual signal transformations. Two distinct methods for the two signal model components will hence be presented in ch. 10.

Chapter 7

Model Parameter Estimation

Introduction

In ch. 6 we have introduced a parametric extended source-filter representation for the partial amplitude trajectories of the deterministic part of an instrument sound signal as well as a classic source-filter model for its residual using a cepstral representation of its spectral envelope. All components of these models have been established as functions of some control variables introduced in ch. 5.

The separate components of the source-filter models however require an internal representation for which we propose to use B-splines which have been thoroughly introduced in chapter 4. Therefore, in the current chapter a detailed description will be given, showing the application of such B-spline models for the representation of trajectories of sound signal data and how the model's parameters can be estimated in the least-mean-squares sense.

In this chapter we will hence present the parameter estimation techniques for the harmonic and residual component separately and will give all required equations to perform estimation using regularization and preconditioning in their respective contexts.

7.1 Harmonic Model

The parametric model for the partial amplitude data shown in eq. (6.4) uses a source excitation function $S_{(k,s)}(\mathbf{\Theta}_h)$ and a resonance component $R(\hat{f}_{(k)})$ to represent the sound signal contributions derived from the assumptions of features best described by either harmonic index or frequency.

Both components will be established using B-splines yielding a linear combination of such, though with different parameterizations:

$$S_{(k,s)}(\mathbf{\Theta}_h) \;=\; \sum_{p=1}^{P} b_p(\mathbf{\Theta}_h) \cdot w_p \tag{7.1}$$

$$R(f) \;=\; \sum_{p=1}^{P} b_p(f) \cdot w_p \tag{7.2}$$

The excitation function $S_{(k,s)}(\mathbf{\Theta}_h)$ for all partials k and both temporal segments s will be established using identical multivariate B-splines as indicated in eq. (7.1), though all having a unique weight vector. This weight vector for an individual partial k and temporal segment s will be denoted $\mathbf{w}_{(k,s)}$. The approach follows from the assumption that the excitation of the amplitudes of the partials are mutually independent. The excitation component of each partial of a quasi-harmonic instrument is hence depicted separately for each partial index, though they share the same control parameters to enable their conjoint control.

The resonance component as shown in eq. (7.2) is also represented using a B-spline basis, though with an univariate B-spline as a function of continuous frequency values. This component shall cover the frequency dependent features jointly for all partials within the harmonic model and we will refer to its dedicated parameter vector by using $\mathbf{w}_{(R)}$.

The use of B-splines for modeling the excitation source as well as the resonance component of all partials of a quasi-harmonic instrument allows for a continuous-valued representation of the partial amplitudes with respect to the control parameters.

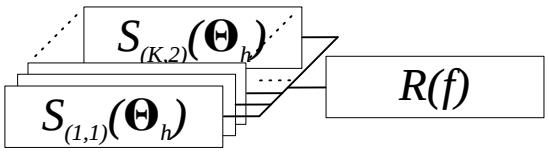


Figure 7.1: The internal representation of the extended source filter model for the harmonic component using individual source functions for all harmonics with identical parameters and a shared resonance component.

Fig. 7.1 shows a schematic overview of the proposed internal representation for the harmonic component of a quasi-harmonic instrument, showing the mutually independent excitation source functions for all partial indexes K and both temporal segments $s = \{1, 2\}$ as well as the conjoint resonance filter function $R(f)$.

7.1.1 Parameter Estimation

For a reasonable separation of the contributions of the disjoint sound features inherent within quasi-harmonic instruments represented by the individual models for a sound signals excitation and resonance, an overall system of linear equations needs to be established to allow for their joint estimation. Such a system of linear equations hence requires to include the weight parameters of all B-splines models used by the harmonic model and we therefore establish the coefficient vector for the harmonic model as:

$$\mathfrak{w} := [\mathbf{w}_{(1,1)}, \ldots, \mathbf{w}_{(K,1)}, \mathbf{w}_{(1,2)}, \ldots, \mathbf{w}_{(K,2)}, \mathbf{w}_{(R)}]^T \tag{7.3}$$

The overall system's weight vector $\mathfrak{w}$ hence consist of the concatenation of the vectorized weight parameters of the source' tensor-product B-splines $\mathbf{w}_{(k,s)}$ for all K partials and both temporal segments $s = \{1, 2\}$ as well as the vector

of weight parameters $\mathbf{w}_{(R)}$ of the univariate B-spline model used to represent the resonance component.

Estimation of the values of the weight vector shall then be done using the partial amplitude and control parameter data obtained from a database of instrument recordings. However, the transformation matrix required for the linear system of equation is not established explicitly as we can take advantage of some matrix properties that allow for a much more efficient parameter estimation using the iterative optimization strategy thoroughly discussed in sec. 4.2.2. We will therefore analyze the linear system of the harmonic model shown in eq. (7.4) represented by the overdetermined system $\mathbf{A} = \mathbf{B}\mathfrak{w}$, where $\mathbf{A}$ holds all partial amplitudes at all time frames from all analyzed recordings and $\mathbf{B}$ refers to the transformation matrix of the system holding the B-spline values of the excitation and resonance components regarding their respective control parameter values without their weights.

The two advantageous properties of the transformation matrix are its redundancy and its sparseness which can both be observed from the detailed excerpt of the matrix $\mathbf{B}$ of the system in eq. (7.4). There, we are using $\mathbf{0}$ to indicate a null vector, whose amount of dimensions is equal to the amount of parameters $\mathbf{w}_{(k,s)}$ of a single tensor-product B-spline.

$$
\overbrace{\begin{bmatrix} A_{(1)}(n) & A_{(2)}(n) & \cdots & A_{(K)}(n) & \dots \end{bmatrix}}^{\mathbf{A}}
$$

$$
=
$$

$$
\underbrace{\begin{bmatrix} \mathbf{w}_{(1,1)} \\ \mathbf{w}_{(2,1)} \\ \vdots \\ \mathbf{w}_{(K,1)} \\ \vdots \\ \mathbf{w}_{(K,2)} \\ \mathbf{w}_{(R)} \end{bmatrix}}_{\mathfrak{w}}
\times
\underbrace{\begin{bmatrix} \mathbf{b}_{(S)}(\mathbf{\Theta}_h(n)) & \mathbf{0} & \cdots & \mathbf{0} & \dots \\ \mathbf{0} & \mathbf{b}_{(S)}(\mathbf{\Theta}_h(n)) & \cdots & \mathbf{0} & \dots \\ \vdots & \vdots & \ddots & \vdots & \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{b}_{(S)}(\mathbf{\Theta}_h(n)) & \dots \\ \vdots & \vdots & & \vdots & \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \dots \\ \mathbf{b}_{(R)}(\hat{f}_{(1)}) & \mathbf{b}_{(R)}(\hat{f}_{(2)}) & \cdots & \mathbf{b}_{(R)}(\hat{f}_{(K)}) & \dots \end{bmatrix}}_{\mathbf{B}} \tag{7.4}
$$

In eq. (7.4), three measured partial amplitudes $A_{(k)}(n)$ for a specific time frame n of one instrument sound are shown on top using partial indices $k = 1$, 2 and K. All partial amplitudes are represented within the system of equations by the sum of its excitation and resonance component. It is easy to observe, that the excitation component of all three partials are having unique weight vectors $\mathbf{w}_{(k,1)}$, though identical basis function values $\mathbf{b}_{(S)}(\mathbf{\Theta}_h(n))$ and due to their different frequency locations $\hat{f}_{(k)}$ they exhibit non-equal basis function values $\mathbf{b}_{(R)}(\hat{f}_{(k)})$ for their resonance component. Please note, that we are using

$_{(S)}$ and $_{(R)}$ to refer to the source' B-spline and resonance B-spline functions respectively.

We may hence derive the following properties from the transformation matrix $\mathbf{B}$:

- **Sparsity**: From eq. (4.21) we have already seen that $\mathbf{b}(\mathbf{\Theta}_h(n))$ is highly sparse. The matrix $\mathbf{B}$ in eq. (7.4) however is by far less dense as each column contains $2 \cdot K - 1$ nullvectors and only the two subvectors $\mathbf{b}(\mathbf{\Theta}_h(n))$ and $\mathbf{b}(\hat{f}_{(k)})$ hold some non-zero values.

- **Redundancy**: All partial amplitude values $A_{(k)}(n)$ at one time frame n share an identical control parameter vector $\mathbf{\Theta}_h(n)$. The overall system of linear equations therefore contains a lot of redundant information as each vectorized tensor-product B-spline is contained up to $2 \cdot K$ times within the matrix. The factor 2 refers to the overlapping part of the temporal segmentation scheme. For the non-overlapping time frames, only $1 \cdot K$ times the TPB is contained.

With respect to the analyzed properties of the transformation matrix of the linear system of the harmonic model, we do neither establish the matrix explicitly nor using any sparse matrix representation to estimate the model's free parameters $\mathfrak{w}$ using its direct-form solution. We rather establish an iterative method that takes advantage of the redundancy and sparseness property and allows to estimate the parameters using the SCG method introduced in sec. 4.2.2.

A cost function shall hence be defined as in eq. (7.5), using $\mathfrak{w}$ to refer to the model's free parameters, $\mathbf{A}$ to denote all partial amplitudes at all frames of all sounds of a database of recordings with their respective control parameter sets $\mathbf{\Psi}_h$. The sum on the right hand side of the equation then adds the cost values for all single recordings i of the database to enable offline learning for the SCG method.

$$E_h(\mathfrak{w}; \mathbf{A}|\mathbf{\Psi}_h) = \frac{1}{2} \sum_i E_h^i(\mathfrak{w}; \mathbf{A}^i|\mathbf{\Psi}_h^i) \tag{7.5}$$

$$E_h^i(\mathfrak{w}; \mathbf{A}^i|\mathbf{\Psi}_h^i) = \frac{1}{\nu_h} \sum_{s,n} \tau_{h,s}^i(n) \sum_{k=1}^{K} \left| A_{(k)}^i(n) - \hat{A}_{(k,s)}(\mathbf{\Psi}_h(n)^i) \right|^2 \tag{7.6}$$

The error function for a single datum of the database represented by its partial amplitudes $\mathbf{A}^i$ and control parameters $\mathbf{\Psi}_h^i$ is eventually expressed in a least-mean-square manner as shown in eq. (7.6) using its distinct temporal segmentation function $\tau_{h,s}^i(n)$.

The variable ν_h is used for normalization and is defined as in eq. (7.7), whereas N refers to the amount of frames of the recording, K denotes the amount of partials and the factor 2 is used due to the two temporal individual segments.

$$\nu_h = 2NK \tag{7.7}$$

The normalization is crucial for numerical reasons. As the amount of partial data for a whole database of recordings may become tremendously large, round-off errors may occur even with double precision number format due to the extensive accumulation of small numerical errors. However, even though the normalization is introduced in eq. (7.6) using a single factor to be applied to the overall summation, this is not recommended for any real-world implementation. Normalization needs to be applied always next to its respective sum and the notation used here, though being mathematically correct is only used for convenience.

The temporal segmentation function $\tau^i_{h,s}(n)$ essentially applies a switch between the attack-sustain and sustain-release components of the harmonic model and therefore, the cost function needs to sum over the two temporal segments and all time frames before taking the square error of all measured partial data $A_{(k)}$ and the model's accordingly estimated value $\hat{A}_{(k,s)}$ for the respective temporal segment s and with respect to the control parameters $\mathbf{\Psi}_h$.

The least-mean-square error criterion as defined in eq. (7.6) can be implemented much more efficiently than the original linear system in terms of memory usage. The frame-wise control parameters do not need to be stored multiple times and all nullvectors of the transformation matrix of the linear system can be omitted.

The required gradients for the optimization strategy need then to be determined separately for the weights of the source excitation functions $\mathbf{w}_{(k,s)}$ and the weights of the resonance component $\mathbf{w}_{(R)}$. We may therefore express the gradient of the excitation component for a vectorized tensor-product of a single partial k and temporal segment s as follows:

$$\frac{\partial E^i_h}{\partial \mathbf{w}_{(k,s)}} = -\frac{1}{\nu_h} \sum_n \tau_{h,s}(n) \Big| A_{(k)}(n) - \hat{A}_{(k,s)}(\mathbf{\Psi}_h(n)) \Big| \mathbf{b}_{(S)}(\mathbf{\Theta}_h(n)) \tag{7.8}$$

and its derivative with respect to the weight parameters of the resonance component then becomes:

$$\frac{\partial E^i_h}{\partial \mathbf{w}_{(R)}} = -\frac{1}{\nu_h} \sum_{s,n} \tau_{h,s}(n) \sum_{k=1}^{K} \Big| A_{(k)}(n) - \hat{A}_{(k,s)}(\mathbf{\Psi}_h(n)) \Big| \mathbf{b}_{(R)}(\hat{f}_{(k)}) \tag{7.9}$$

The parameters of the cost function E^i_h for a single sound of the database $(\mathfrak{w}; \mathbf{A}^i | \mathbf{\Psi}^i_h)$ on the left hand side have been omitted for improved readability. Furthermore, the sound sample index i has also been omitted on the right hand side, though it shall be noted that $A_{(k)}(n)$ as well as $\hat{f}_{(k)}$ are specific variables of their respective sound sample i.

Both derivatives of the cost function (7.6) refer to a whole vector of the free parameters of the model which is also reflected by the use of the vector of B-spline values $\mathbf{b}_{(S)}$ and $\mathbf{b}_{(R)}$ respectively. Performing optimization using the conjugate gradient method SCG however, requires to take the cost and gradient values of all model parameters and of all training examples i of the dataset into account at once. To perform an iteration of the optimization procedure using

the update rule from eq. (4.11), the gradients of all free model parameters need hence to be determined to create the complete gradient vector.

7.1.2 Regularization

As thoroughly introduced in sec. 4.4, overfitting is an important issue with B-spline based data representations. We have hence introduced a framework for regularization to control the smoothness of the fit and we will employ several smoothness penalties. The according values for the derivative orders, selected dimensions for regularizing of the multi-variate source function as well as initial regularization parameters will be given in ch. 8.

There is however another obstacle when learning the free parameters of the harmonic model which can be subsumed as the model's inherent ambiguities.

The 2 filter components S and R are established as discrete and independent functions using a per partial source function and a resonance function for all partials jointly. This however allows the source functions to exhibit an additional arbitrary constant or offset, as long as the resonance function also yields an additional offset with opposite value. The optimization procedure will hence never converge as there is an infinite number of optimal solutions due to the arbitrary value of the constant for both filter components, which distorts the minimum peak of the cost function to an unbounded valley with minimum value.

This fact may also be derived by assuming that such a constant can be an additional gain factor, which gets added within one filter and subtracted within the other.

We therefore propose to keep the average position of the resonance filter fixed around some level value to solve the problem of the two filter functions drifting in opposite directions without affecting the model error during optimization. Thus, we will force the sum of the resonance function to be 0 using a regularization term similarly to the one introduced for general B-spline models in sec. 4.4. Note, that we could also regularize the source function for the same reason using a similar approach.

Therefore, the first regularization of the harmonic model applies to the resonance filter function only:

$$\mathcal{R}_{\mathbf{I}}(\mathfrak{w}) = \mathcal{R}_{\mathbf{I}}(\mathbf{w}_{(R)}) \tag{7.10}$$

For constraining the resonance function represented by its weights $\mathbf{w}_{(R)}$ to be 0 in average, we thus employ the regularization term shown in eq. (7.11) for a single B-spline parameter vector $\mathbf{w}$ to penalize the squared sum of the resonance filter function. The function is evaluated at equidistant positions v_j with J sampling positions along its domain and its according weight parameter $\lambda_{\mathbf{I}}$.

$$\mathcal{R}_{\mathbf{I}}(\mathbf{w}) = \frac{1}{2}\lambda_{\mathbf{I}}\Big|\sum_{j}^{J}\sum_{p=1}^{P} b_p(v_j)w_p\Big|^2 \tag{7.11}$$

The regularization term in eq. (7.11) is introduced in a similar manner to the regularization terms for slope or curvature penalty in sec. 4.4, though since we are using the 0-derivative and the squared sum of the virtual data points

v_j we are penalizing for the functions offset or DC value, rather than its sum of squared values.

In contrast to the initial location of the the regularization parameter $\lambda_{\mathbf{I}}$, it has now been included into the regularization term for convenience reasons when writing the objective function later.

The regularization weight parameter $\lambda_{\mathbf{I}}$ may then be determined by applying eq. (4.30) for the resonance component of the harmonic model only and the adapted equation hence becomes:

$$\lambda_{\mathbf{I}} = \lambda_{\mathbf{I},0} \frac{\left|\left| \sum_i^N \left(\sum_p^P b_{(R)_p}(\hat{f}_{(k)}) \right)^2 \right|\right|_1}{\left|\left| \sum_j^J \left(\sum_p^P b_{(R)_p}(v_j) \right)^2 \right|\right|_1} \tag{7.12}$$

A possible value for the and data independent, initial coefficient $\lambda_{\mathbf{I},0}$ will be given in sec. 8.2.

The use of a gradient method for parameter estimation makes it eventually necessary to determine the first derivative of eq. (7.11) with respect to the weight parameters of the resonance function only:

$$\frac{\partial \mathcal{R}_{\mathbf{I}}(\mathbf{w})}{\partial w_p} = \lambda_{\mathbf{I}} \Big| \sum_j^J \sum_{p=1}^P b_p(v_j) w_p \Big| \sum_j^J b_p(v_j) \tag{7.13}$$

There is however another ambiguity, which arises from the use of the pitch parameter for the source filter function and the frequency dependent resonance filter, because the pitch parameter can be also regarded a log-frequency dependency. This means, that frequency dependent features of the instrument can either be represented by the source or the resonance, which is in fact a desired property of the model, however, ambiguous solutions are not.

We address this ambiguity issue of double frequency-dependency in the model by using the regularization term (4.29) introduced in sec. 4.4 for reasonably penalizing for slope of all $2K$ source filter functions along the pitch dimension using $z = 1$ and $d = P$. This shall enforce the source functions to exhibit less radical variations along the pitch dimension assuming that an excitation signal may be quite similar for nearby pitches. The resonance filter function will not be constrained in such terms and may therefore be able to represent resonances and anti-resonances in close proximity on the frequency axis.

In sec. 4.4 the regularization function $\mathcal{R}^{(z,d)}(\mathbf{w})$ has been introduced as a function of the weights of the B-spline together with its according balancing parameter $\lambda^{(z,d)}$ which depends on the data. To constrain the harmonic model using this regularization, both need to be represented independently for every partial k and temporal segment s to account for their independent weight vectors $\mathbf{w}_{(k,s)}$ and respective data.

To furthermore retain the possibility of adding regularization for other values of z or d, we thus write the second type of regularization $\mathcal{R}_{\mathbf{II}}$ as:

$$\mathcal{R}_{\mathbf{II}}(\mathfrak{w}) = \frac{1}{2} \sum_{k,s} \sum_{z,d} \lambda_{\mathbf{II},(k,s)}^{(z,d)} \mathcal{R}^{(z,d)}(\mathbf{w}_{(k,s)}) \tag{7.14}$$

The regularization parameter $\lambda_{\mathbf{II},(k,s)}^{(z,d)}$ is then also determined as for $\mathcal{R}_{\mathbf{I}}$ using an adaptation of eq. (4.30), which we write with an initial weighting

parameter $\lambda_{\mathbf{II},0}^{(z,d)}$ that is equal for all partial indexes and the temporal segments but may vary according to a selected penalty specified by z and d. The parameter may hence be determined individually for all its 4 parameters as follows:

$$\lambda_{\mathbf{II},(k,s)}^{(z,d)} = \lambda_{\mathbf{II},0}^{(z,d)} \frac{\left|\left| \sum_i^N \left(\sum_p^P \mathbf{b}_{(S)_p}^{(z,d)}(\mathbf{\Theta}_h(n)) \right)^2 \right|\right|_1}{\left|\left| \sum_j^J \left(\sum_p^P \mathbf{b}_{(S)_p}^{(z,d)}(\mathbf{v}_j) \right)^2 \right|\right|_1} \tag{7.15}$$

Values for the initial coefficient $\lambda_{\mathbf{II},0}^{(z,d)}$ for various z and d that have been used for constraining the harmonic model while training will be presented in sec. 8.2.

The first derivative of eq. (7.14) then needs to be taken individually for all partial indexes k and both temporal segments s, yielding:

$$\frac{\partial \mathcal{R}_{\mathbf{II}}(\mathfrak{w})}{\partial \mathbf{w}_{(k,s)}} = \frac{1}{2} \sum_{z,d} \lambda_{\mathbf{II},(k,s)}^{(z,d)} \frac{\partial \mathcal{R}_{\mathbf{II}}^{(z,d)}(\mathbf{w}_{(k,s)})}{\partial \mathbf{w}_{(k,s)}} \tag{7.16}$$

It may be derived that each individual derivative at the right hand side essentially equals eq. (4.33), though we are using the weight vector as derivative variable here rather than single vector entries.

The general objective function for estimating the free parameters of the harmonic model with respect to all given sound examples i and all possible regularizations using $\mathcal{R}_{\mathbf{I}}$ to refer to the above and $\mathcal{R}_{\mathbf{II}}$ to denote the multidimensional slope or curvature penalties is thus:

$$\mathcal{O}_h = E_h(\mathfrak{w}; \mathbf{A}|\mathbf{\Psi}_h) + \mathcal{R}_{\mathbf{I}}(\mathfrak{w}) + \mathcal{R}_{\mathbf{II}}(\mathfrak{w}) \tag{7.17}$$

Minimization of $\mathcal{O}_h$ using the iterative offline procedure SCG therefore not only estimates the parameter according to the given data, but also regarding certain desired behaviors determined by the additional regularization terms. The required values for the initial regularization parameters $\lambda_{\mathbf{I},0}$ and $\lambda_{\mathbf{II},0}^{(z,d)}$ will be given in ch. 8.

7.1.3 Preconditioning

Preconditioning has been thoroughly introduced in sec. 4.5, however, as the harmonic model is assembled using linear combinations of multiple source and one single filter function, the according equations need to be updated accordingly.

First of all, the weight vector $\mathfrak{w}$ of the harmonic model needs to get replaced by an element-wise product using notation $\circ$ of a new weight vector $\widetilde{\mathfrak{w}}$ and their respective scaling parameters $\mathbf{c}$ for the quadratic form:

$$\mathfrak{w} = \widetilde{\mathfrak{w}} \circ \mathbf{c} \tag{7.18}$$

whereas:

$$\widetilde{\mathfrak{w}} := [\widetilde{\mathbf{w}}_{(1,1)}, \ldots, \widetilde{\mathbf{w}}_{(K,1)}, \widetilde{\mathbf{w}}_{(1,2)}, \ldots, \widetilde{\mathbf{w}}_{(K,2)}, \widetilde{\mathbf{w}}_{(R)}]^T \tag{7.19}$$

$$\mathbf{c} := [\mathbf{c}_{(1,1)}, \ldots, \mathbf{c}_{(K,1)}, \mathbf{c}_{(1,2)}, \ldots, \mathbf{c}_{(K,2)}, \mathbf{c}_{(R)}]^T \tag{7.20}$$

The first derivatives of the objective function (7.17) with respect to either $\widetilde{\mathbf{w}}_{(k,s)}$ or $\widetilde{\mathbf{w}}_{(R)}$ required for their estimation are thus:

$$\frac{\partial \mathcal{O}_h}{\partial \widetilde{\mathbf{w}}_{(k,s)}} = \frac{\partial E_h}{\partial \widetilde{\mathbf{w}}_{(k,s)}} + \frac{\partial \mathcal{R}_{\mathbf{II}}(\mathfrak{w})}{\partial \widetilde{\mathbf{w}}_{(k,s)}} \tag{7.21}$$

$$\frac{\partial \mathcal{O}_h}{\partial \widetilde{\mathbf{w}}_{(R)}} = \frac{\partial E_h}{\partial \widetilde{\mathbf{w}}_{(R)}} + \frac{\partial \mathcal{R}_{\mathbf{I}}(\mathfrak{w})}{\partial \widetilde{\mathbf{w}}_{(R)}} \tag{7.22}$$

because:

$$\frac{\partial \mathcal{R}_{\mathbf{I}}(\mathfrak{w})}{\partial \widetilde{\mathbf{w}}_{(k,s)}} = \mathbf{0} \tag{7.23}$$

$$\frac{\partial \mathcal{R}_{\mathbf{II}}(\mathfrak{w})}{\partial \widetilde{\mathbf{w}}_{(R)}} = \mathbf{0} \tag{7.24}$$

The derivatives of the cost function E_h can be processed individually for every single recording i as follows:

$$\frac{\partial E_h}{\partial \widetilde{\mathbf{w}}_{(k,s)}} = \frac{1}{2} \sum_i \frac{\partial E_h^i}{\partial \widetilde{\mathbf{w}}_{(k,s)}} \tag{7.25}$$

$$\frac{\partial E_h}{\partial \widetilde{\mathbf{w}}_{(R)}} = \frac{1}{2} \sum_i \frac{\partial E_h^i}{\partial \widetilde{\mathbf{w}}_{(R)}} \tag{7.26}$$

$$\tag{7.27}$$

whereas the individual derivatives of E_h^i then eventually yield:

$$\frac{\partial E_h^i}{\partial \widetilde{\mathbf{w}}_{(k,s)}} = -\frac{2}{\nu_h} \sum_n \tau_{h,s}(n) \Big| A_{(k)}(n) - \hat{A}_{(k,s)}(\mathbf{\Psi}_h(n)) \Big| \mathbf{b}_{(S)}(\mathbf{\Theta}_h(n)) \circ \mathbf{c}_{(k,s)} \tag{7.28}$$

$$\frac{\partial E_h^i}{\partial \widetilde{\mathbf{w}}_{(R)}} = -\frac{2}{\nu_h} \sum_{s,n} \tau_{h,s}(n) \sum_{k=1}^{K} \Big| A_{(k)}(n) - \hat{A}_{(k,s)}(\mathbf{\Psi}_h(n)) \Big| \mathbf{b}_{(R)}(\hat{f}_{(k)}) \circ \mathbf{c}_{(R)} \tag{7.29}$$

using $\circ$ to denote the elemen-wise product.

The first derivatives of the regularization term $\mathcal{R}_{\mathbf{I}}$ with respect to $\widetilde{\mathbf{w}}_{(R)}$ resolves to eq. (7.30) using the univariate virtual data points v_j:

$$\frac{\partial \mathcal{R}_{\mathbf{I}}(\mathfrak{w})}{\partial \widetilde{\mathbf{w}}_{(R)}} = \lambda_{\mathbf{I}} \Big| \sum_j^J \sum_{p=1}^P b_p(v_j) \widetilde{w}_p c_p \Big| \sum_j^J \mathbf{b}_{(R)}(v_j) \circ \mathbf{c}_{(R)} \tag{7.30}$$

The first derivative of $\mathcal{R}_{\mathbf{II}}$ however needs to consider all directional derivatives determined by z and d using the multivariate virtual data points $\mathbf{v}_j$:

$$\frac{\partial \mathcal{R}_{\mathbf{II}}(\mathfrak{w})}{\partial \widetilde{\mathbf{w}}_{(k,s)}} = \sum_{z,d} \lambda_{\mathbf{II},(k,s)}^{(z,d)} \sum_{j}^{J} \eta^{(z,d)}(\mathbf{v}_j) \Big| \sum_{p=1}^{P} \left(b_p^{(z,d)}(\mathbf{v}_j) \widetilde{w}_p \, c_p \right) \Big| \mathbf{b}_{(S)}^{(z,d)}(\mathbf{v}_j) \circ \mathbf{c}_{(k,s)} \tag{7.31}$$

One may note, that the B-spline expressions in straight brackets in eq. (7.30) and (7.31) refer to the specific B-splines of the resonance and source filter function respectively, though their only difference in notation here is by using either univariate or multivariate parameters.

To determine the scaling parameters $\mathbf{c}_{(k,s)}$ and $\mathbf{c}_R$ the second derivatives of the objective function (7.17) are required with respect to their according weights:

$$\frac{\partial^2 \mathcal{O}_h}{\partial \widetilde{\mathbf{w}}_{(k,s)}^2} = \frac{\partial^2 E_h}{\partial \widetilde{\mathbf{w}}_{(k,s)}^2} + \frac{\partial^2 \mathcal{R}_{\mathbf{II}}(\mathfrak{w})}{\partial \widetilde{\mathbf{w}}_{(k,s)}^2} \tag{7.32}$$

$$\frac{\partial^2 \mathcal{O}_h}{\partial \widetilde{\mathbf{w}}_{(R)}^2} = \frac{\partial^2 E_h}{\partial \widetilde{\mathbf{w}}_{(R)}^2} + \frac{\partial^2 \mathcal{R}_{\mathbf{I}}(\mathfrak{w})}{\partial \widetilde{\mathbf{w}}_{(R)}^2} \tag{7.33}$$

The second derivatives of the cost function E_h can again be processed individually for every single recording i as follows:

$$\frac{\partial^2 E_h}{\partial \widetilde{\mathbf{w}}_{(k,s)}^2} = \frac{1}{2} \sum_{i} \frac{\partial^2 E_h^i}{\partial \widetilde{\mathbf{w}}_{(k,s)}^2} \tag{7.34}$$

$$\frac{\partial^2 E_h}{\partial \widetilde{\mathbf{w}}_{(R)}^2} = \frac{1}{2} \sum_{i} \frac{\partial^2 E_h^i}{\partial \widetilde{\mathbf{w}}_{(R)}^2} \tag{7.35}$$

$$\tag{7.36}$$

whereas the individual second derivatives of E_h^i then resolve to:

$$\frac{\partial^2 E_h^i}{\partial \widetilde{\mathbf{w}}_{(k,s)}^2} = \frac{1}{\nu_h} \sum_{n} \tau_{h,s}(n) \Big|\Big| \mathbf{b}_{(S)}(\mathbf{\Theta}_h(n)) \circ \mathbf{c}_{(k,s)} \Big|\Big|^2 \tag{7.37}$$

$$\frac{\partial^2 E_h^i}{\partial \widetilde{\mathbf{w}}_{(R)}^2} = \frac{1}{\nu_h} \sum_{s,n} \tau_{h,s}(n) \sum_{k=1}^{K} \Big|\Big| \mathbf{b}_{(R)}(\hat{f}_{(k)}) \circ \mathbf{c}_{(R)} \Big|\Big|^2 \tag{7.38}$$

using notation $||\cdot||^2$ to denote the element-wise square operator.
The according second derivatives of the regularization terms yield:

$$\frac{\partial^2 \mathcal{R}_{\mathbf{I}}(\mathfrak{w})}{\partial \widetilde{\mathbf{w}}_{(R)}^2} = \lambda_{\mathbf{I}} \Big|\Big| \sum_{j}^{J} \mathbf{b}_{(R)}(v_j) \circ \mathbf{c}_{(R)} \Big|\Big|^2 \tag{7.39}$$

$$\frac{\partial^2 \mathcal{R}_{\mathbf{II}}(\mathfrak{w})}{\partial \widetilde{\mathbf{w}}_{(k,s)}^2} = \sum_{z,d} \lambda_{\mathbf{II},(k,s)}^{(z,d)} \sum_{j}^{J} \eta^{(z,d)}(\mathbf{v}_j) \Big|\Big| \mathbf{b}_{(S)}^{(z,d)}(\mathbf{v}_j) \circ \mathbf{c}_{(k,s)} \Big|\Big|^2 \tag{7.40}$$

To eventually obtain the scaling parameter vectors $\mathbf{c}_{(k,s)}$ for all k and both s as well as $\mathbf{c}_{(R)}$, the second derivatives of the objective function (7.32) and (7.33) will be set equal to 1 and solved for $\mathbf{c}_{(k,s)}$ and $\mathbf{c}_{(R)}$ respectively. This yields the following equations:

$$\begin{aligned}\mathbf{c}_{(k,s)} = \Bigg(& \sum_i \frac{1}{\nu_h} \sum_n \tau_{h,s}(n) \Big|\Big| \mathbf{b}_{(S)}(\boldsymbol{\Theta}_h(n)) \Big|\Big|^2 \\ & + \sum_{z,d} \lambda^{(z,d)}_{\mathbf{II},(k,s)} \sum_j^J \eta^{(z,d)}(\mathbf{v}_j) \Big|\Big| \mathbf{b}^{(z,d)}_{(S)}(\mathbf{v}_j) \Big|\Big|^2 \Bigg)^{-\frac{1}{2}}\end{aligned} \tag{7.41}$$

$$\begin{aligned}\mathbf{c}_{(R)} = \Bigg(& \sum_i \frac{1}{\nu_h} \sum_{s,n} \tau_{h,s}(n) \sum_{k=1}^K \Big|\Big| \mathbf{b}_{(R)}(\hat{f}_{(k)}) \Big|\Big|^2 \\ & + \lambda_{\mathbf{I}} \Big|\Big| \sum_j^J \mathbf{b}_{(R)}(v_j) \Big|\Big|^2 \Bigg)^{-\frac{1}{2}}\end{aligned} \tag{7.42}$$

All required equations to eventually estimate the free parameters of the harmonic model of eq. (6.4) using the objective function (7.17) have been presented in this section. However, establishing the B-splines for the model's source and resonance component still requires several hyperparameters which are not part of the parameter estimation method, but need to be adjusted manually. These will be covered in ch. 8.

7.2 Residual Model

The parametric model to represent the cepstral coefficients of an instrument's residual characteristics presented in eq. (5.1.1) uses individual functions of the control parameters which do not exhibit any interdependency. The model is hence composed of separate and mutually independent B-splines for every cepstral coefficient l and temporal segment s:

$$H_{(l,s)}(\boldsymbol{\Theta}_r) = \sum_{p=1}^P b_p(\boldsymbol{\Theta}_r) \cdot w_p \tag{7.43}$$

Similarly to the B-splines of the excitation source of the harmonic model, the B-splines for the representation of the cepstral coefficients l and temporal segments s are established using identical, multivariate B-splines but each with its unique weight vector $\mathbf{w}_{(l,s)}$.

Compared to the model of the partial amplitudes within the harmonic component, the residual's representation is substantially simpler. Fig. 7.2 depicts the mutually independent multivariate B-splines used for representing the cepstral coefficients as functions of the gestural controls. Their parameters may then be estimated using the technique proposed for multivariate B-splines in ch. 4.

7.2.1 Parameter Estimation

Considering the independence of the individual B-spline representations for every cepstral coefficient and temporal segment, their estimation can also be done mutually independent. For the reasons given in ch. 4, we have chosen to estimate the free parameters of the model using the iterative method applying the SCG method. We hence express a cost function for every single cepstral coefficient and temporal segment similarly as in eq. (7.44) for all residual sounds i of the database of recordings using $\mathbf{C}_{(l)}$ to denote a data vector that contains the l-th cepstral coefficient of all sounds at all frames and $\mathbf{\Psi}_r$ to refer to their respective control parameters.

$$E_r(\mathbf{w}_{(l,s)}; \mathbf{C}_{(l)}|\mathbf{\Psi}_r) = \frac{1}{2} \sum_i E_r^i(\mathbf{w}_{(l,s)}; \mathbf{C}_{(l)}^i|\mathbf{\Psi}_r^i) \tag{7.44}$$

$$E_r^i(\mathbf{w}_{(l,s)}; \mathbf{C}_{(l)}^i|\mathbf{\Psi}_r^i) = \frac{1}{\nu_r} \sum_{s,n} \tau_{r,s}^i(n) \left| C_{(l)}^i(n) - \hat{C}_{(l,s)}(\mathbf{\Psi}_r^i(n)) \right|^2 \tag{7.45}$$

Eq. (7.45) then shows the cost function for a single sample in the least-mean-square sense, taking also the temporal segmentation into account as for within the harmonic model. An additional normalization factor ν_r has also been introduced for the same reason as for the harmonic model which constitutes itself as in eq. (7.46) using N to denote the amount of analysis frames within the signal:

$$\nu_r = 2N \tag{7.46}$$

The gradient as required for the parameter estimation using the SCG method then resolves to eq. (7.47) and (7.48) again omitting the parameter variables of for E_r and E_r^i for readability.

$$\frac{\partial E_r}{\partial \mathbf{w}_{(l,s)}} = \frac{1}{2} \sum_i \frac{\partial E_r^i}{\partial \mathbf{w}_{(l,s)}} \tag{7.47}$$

$$\frac{\partial E_r^i}{\partial \mathbf{w}_{(l,s)}} = -\frac{1}{\nu_r} \sum_n \tau_{r,s}(n) \left| C_{(l)}^i(n) - \hat{C}_{(l,s)}(\mathbf{\Psi}_r^i(n)) \right| \mathbf{b}(\mathbf{\Theta}_r(n)) \tag{7.48}$$

Since there is only one kind of a multivariate B-spline contained within the model for the residual component, only one gradient function needs to

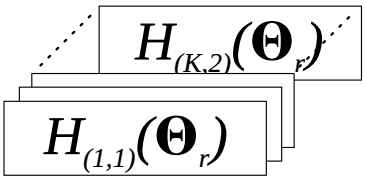


Figure 7.2: The internal representation of the residual component of the instrument model uses independent representation for every single cepstral coefficient and temporal segment as a function of the control parameters.

be provided and the estimation of the individual weight vectors $\mathbf{w}_{(l,s)}$ may eventually be done individually for all l and both s.

7.2.2 Regularization

In contrast to the harmonic model, there are no ambiguities within the model of the residual component, though overfitting might still occur. We therefore apply regularization as introduced in ch. 4 with respect to the individual weight vectors $\mathbf{w}_{(l,s)}$.

The objective function for a single pair of l and s taking its according cost value as well as some regularization into account may hence be written as:

$$\mathcal{O}_r = E_r(\mathbf{w}_{(l,s)}; \mathbf{C}_{(l)} | \mathbf{\Psi}_r) + \mathcal{R}_{\mathbf{II}}(\mathbf{w}_{(l,s)}) \tag{7.49}$$

The regularization term $\mathcal{R}_{\mathbf{II}}$ is defined as for the harmonic model but for a single cepstral coefficient and temporal segment. It reuses the definition of $\mathcal{R}$ in eq. (4.29), but incorporates all values of z and d:

$$\mathcal{R}_{\mathbf{II}}(\mathbf{w}_{(l,s)}) = \frac{1}{2} \sum_{z,d} \lambda_{\mathbf{II},(l,s)}^{(z,d)} \mathcal{R}_{\mathbf{II}}^{(z,d)}(\mathbf{w}_{(l,s)}) \tag{7.50}$$

One may note, that we reuse $\mathbf{II}$ for the regularization to indicate the similarity to the according regularization term for the harmonic model.

The regularization parameter $\lambda_{\mathbf{II},(l,s)}^{(z,d)}$ is then also determined as for the harmonic model using an adaptation of eq. (4.30). We thus write:

$$\lambda_{\mathbf{II},(l,s)}^{(z,d)} = \lambda_{\mathbf{II},0}^{(z,d)} \frac{\left|\left| \sum_i^N \left(\sum_p^P b_p^{(z,d)}(\mathbf{\Theta}_r(n)) \right)^2 \right|\right|_1}{\left|\left| \sum_j^J \left(\sum_p^P b_p^{(z,d)}(\mathbf{v}_j) \right)^2 \right|\right|_1} \tag{7.51}$$

Values for $\lambda_{\mathbf{II},0}^{(z,d)}$ that we have used for training the residual model for several instruments will be shown in sec. 8.2.

The required gradient of the regularization term then resolves to:

$$\frac{\partial \mathcal{R}_{\mathbf{II}}(\mathbf{w}_{(l,s)})}{\partial \mathbf{w}_{(l,s)}} = \frac{1}{2} \sum_{z,d} \lambda_{\mathbf{II},(l,s)}^{(z,d)} \frac{\partial \mathcal{R}_{\mathbf{II}}^{(z,d)}(\mathbf{w}_{(l,s)})}{\partial \mathbf{w}_{(l,s)}} \tag{7.52}$$

The derivative within the sum on the right hand side then equals the first derivative of the regularization term shown in eq. (4.33). As within the harmonic model, the required initial regularization parameter values $\lambda_{\mathbf{II},0}^{(z,d)}$ will be identical for all cepstral coefficients and temporal segments and its according values used for the present thesis will be given in ch. 8.

7.2.3 Preconditioning

We also apply preconditioning for the estimation of the parameters of the residual model component to substantially improve the convergence rate of the algorithm using the same method as introduced in ch. 4 and as already applied to the harmonic model. Therefore, we again substitute the weight vector $\mathbf{w}_{(l,s)}$ of the multivariate B-spline representation for every cepstral coefficient and the temporal segments by the element-wise product of a new weight vector and their according scaling coefficients:

$$\mathbf{w}_{(l,s)} = \widetilde{\mathbf{w}}_{(l,s)} \circ \mathbf{c}_{(l,s)} \tag{7.53}$$

The new derivative of the objective function (7.49) is thus:

$$\frac{\partial \mathcal{O}_r}{\partial \widetilde{\mathbf{w}}_{(l,s)}} = \frac{\partial E_r}{\partial \widetilde{\mathbf{w}}_{(l,s)}} + \frac{\partial \mathcal{R}_{\mathbf{II}}(\mathbf{w}_{(l,s)})}{\partial \widetilde{\mathbf{w}}_{(l,s)}} \tag{7.54}$$

whereas their partial derivatives can be derived similarly to the harmonic model for the data dependent cost function:

$$\frac{\partial E_r}{\partial \widetilde{\mathbf{w}}_{(l,s)}} = \frac{1}{2} \sum_i \frac{\partial E_r^i}{\partial \widetilde{\mathbf{w}}_{(l,s)}} \tag{7.55}$$

$$\frac{\partial E_r^i}{\partial \widetilde{\mathbf{w}}_{(l,s)}} = -\frac{1}{\nu_r} \sum_n \tau_{r,s}(n) \Big| C_{(l)}^i(n) - \hat{C}_{(l,s)}(\mathbf{\Psi}_r^i(n)) \Big| \mathbf{b}(\mathbf{\Theta}_r(n)) \circ \mathbf{c}_{(l,s)} \tag{7.56}$$

as well as for the regularization term which essentially equals the first derivative of the second regularization of the harmonic model apart from its parameterization using l instead of k :

$$\frac{\partial \mathcal{R}_{\mathbf{II}}(\mathbf{w}_{(l,s)})}{\partial \widetilde{\mathbf{w}}_{(l,s)}} = \sum_{z,d} \lambda_{\mathbf{II},(l,s)}^{(z,d)} \sum_j^J \eta^{(z,d)}(\mathbf{v}_j) \Big| \sum_{p=1}^P \left(b_p^{(z,d)}(\mathbf{v}_j) \widetilde{w}_p \, c_p \right) \Big| \mathbf{b}^{(z,d)}(\mathbf{v}_j) \circ \mathbf{c}_{(l,s)} \tag{7.57}$$

The second derivative of the objective function is then required again to eventually determine the scaling coefficients $\mathbf{c}_{(l,s)}$:

$$\frac{\partial^2 \mathcal{O}_r}{\partial \widetilde{\mathbf{w}}_{(l,s)}^2} = \frac{\partial^2 E_r}{\partial \widetilde{\mathbf{w}}_{(l,s)}^2} + \frac{\partial^2 \mathcal{R}_{\mathbf{II}}(\mathbf{w}_{(l,s)})}{\partial \widetilde{\mathbf{w}}_{(l,s)}^2} \tag{7.58}$$

The second partial derivative for the cost function may then be expressed as follows:

$$\frac{\partial^2 E_r}{\partial \widetilde{\mathbf{w}}_{(l,s)}^2} = \frac{1}{2} \sum_i \frac{\partial^2 E_r^i}{\partial \widetilde{\mathbf{w}}_{(l,s)}^2} \tag{7.59}$$

$$\frac{\partial^2 E_r^i}{\partial \widetilde{\mathbf{w}}_{(l,s)}^2} = -\frac{1}{\nu_r} \sum_n \tau_{r,s}(n) \Big| C_{(l)}^i(n) - \hat{C}_{(l,s)}(\mathbf{\Psi}_r^i(n)) \Big| \mathbf{b}(\mathbf{\Theta}_r(n)) \circ \mathbf{c}_{(l,s)} \tag{7.60}$$

whereas the second partial derivative of the regularization term becomes:

$$\frac{\partial^2 \mathcal{R}_{\mathbf{II}}(\mathfrak{w})}{\partial \widetilde{\mathbf{w}}_{(l,s)}^2} = \sum_{z,d} \lambda_{\mathbf{II},(k,s)}^{(z,d)} \sum_j^J \eta^{(z,d)}(\mathbf{v}_j) \Big| \Big| \mathbf{b}^{(z,d)}(\mathbf{v}_j) \circ \mathbf{c}_{(l,s)} \Big| \Big|^2 \tag{7.61}$$

Setting the second derivative of the objective function (7.58) to equal 1 and solving for the scaling coefficients eventually yields:

$$\mathbf{c}_{(l,s)} = \left(\sum_i \frac{1}{\nu_r} \sum_n \tau_{r,s}(n) \left|\left| \mathbf{b}(\boldsymbol{\Theta}_r(n)) \right|\right|^2 + \sum_{z,d} \lambda^{(z,d)}_{\mathbf{II},(l,s)} \sum_j^J \eta^{(z,d)}(\mathbf{v}_j) \left|\left| \mathbf{b}^{(z,d)}(\mathbf{v}_j) \right|\right|^2 \right)^{-\frac{1}{2}} \tag{7.62}$$

7.3 Conclusion

In this chapter we have thoroughly discussed the application of multivariate B-splines for representing either harmonic partials of a musical instrument or cepstral coefficients both as function of several control variables. These data trajectories have been assumed to be decisive for an instrument sound and the according control parameters have been selected assuming their substantial influence on its inherent sound features and hence as being suitable for transforming these.

The equations introduced in this chapter cover all required mathematical expressions to create the harmonic as well as residual component for the instrument model proposed within this thesis, whereas both are established to allow for estimating certain instrument characteristics either referring to features related to partial index or partial frequency in case of the harmonic model or to spectral envelope features represented by cepstral coefficients.

However, this chapter only covered the automatic estimation of the model's free parameters and left out the manual adjustment of the hyper parameters. The selection of these requires some assumptions which need to be made a priori to the automatic estimation and hence will be introduced in the next chapter alongside a thorough description of the instrument sound databases used for the application of the model.

Chapter 8

Model Selection

Introduction

Four different quasi-harmonic instruments have been chosen to apply and eventually evaluate the instrument model proposed in the previous chapters. These four instruments have been selected with regard to their belonging family to cover various types of sound production mechanisms and performer-instrument interactions, though only their standard playing techniques without any ornamentations are considered. The sound datasets hence exclude tremolo or vibrato techniques as well as pizzicato or glissandi and only contain recordings that are monophonic with constant pitch and global sound intensity. The sound datasets needed to contain separate recordings for all possible pitches along its respective pitch range and separate recordings for several levels of sound intensity. All recordings are also required to entail a complete instrument sound from its onset to some reasonable offset.

To further assure high-quality recordings with 24Bit resolution depth and 44.1kHz sample rate in a loss-less file format and without undocumented post-processing modifications we have taken the sound datasets from either the Ircam Solo Instruments library [uvib] or the Fazioli F278 Concert Grand piano database [uvia]. The selected instruments then are:

- Trumpet [uvib]
- B$\flat$ Clarinet [uvib]
- Violin [uvib]
- Fazioli Grand Piano [uvia]

The trumpet, clarinet and violin have been chosen to have one representative of the brass, woodwind and violin instrument family respectively. Though, as they all share a continuous excitation mechanism when played in a standard manner, the grand piano library has been picked to study characteristics which may be exclusive for impulsively excited instruments, but has also been selected for its assumably strong timbre variability.

The following tab. 8.1 lists some general statistics about the four different instruments used for the evaluation of the instrument model:

instrument	Num Files	Num Pitches	Num Intensity Levels
trumpet	92	33	3
clarinet	136	47	3
violin	279	see tab. 8.2	3
piano	535	88	≤ 8

Table 8.1: Some general stats about the used sound data sets.

The second column of tab. 8.1 shows the amount of single recordings entailed within the respective sound database, the third shows how many discrete pitches are contained and the last column lists the amount of different intensity levels available for each pitch. The overall amount of sound files of a database should theoretically equal the product of the amount of available pitches and intensity levels, however, a few files are missing in all datasets for unknown reasons, though this does not effect the parameter estimation negatively since only very few are missing.

In case of the violin, the sound data set has been divided into subsets for every string whose statistics are shown in tab. 8.2.

String (Pitch)	Num Files	Num Pitches	Num Intensity Levels
1 (E)	71	25	3
2 (A)	74	25	3
3 (D)	68	23	3
4 (G)	66	22	3

Table 8.2: The general statistics about the violin data set divided into subsets for each string.

The division of the violin into subsets according to the string number is important, because we will create instrument models for every single string rather than for the whole instrument. This is due to the different spectral characteristics caused by the positioning of the bow above the string which effectuates a comb filter behavior that is specific for each string [JS83]. Therefore, equal pitches being played on different strings sound differently and as such a characteristic is not represented within our model, we need to establish individual models for each string.

The means we will eventually establish individual models for the trumpet, clarinet and piano as well as separate models for all four strings of the violin.

For each of the instrument models, a suitable configuration of the multivariate B-spline representations for the excitation and resonance component of the harmonic model as well as for the B-splines used within the residual model need to be set prior to the parameter estimation. Moreover, we have defined several constraints using regularization terms which require initial weighting parameters that also need to be set a priori. Both parts of the issue of finding an appropriate model configuration with properly adjusted constraint weights will be discussed in the next two sections.

8.1 Model Configurations

As thoroughly introduced in ch. 4, B-splines are characterized by their belonging knot sequence and B-spline order and both need to be adjusted manually. Of course this remains true for multivariate B-splines, but since we are using tensor-products of univariate B-splines their configuration may be done individually for each dimension. Hence, for choosing an appropriate configuration of a multivariate B-spline, their individual domain spaces I_g, P and $I_{l,\gamma}$ may be studied separately and the according B-spline parameters can eventually be adjusted individually. Moreover, certain assumptions and a priori knowledge about the inherent data characteristics may also be applied when choosing a knot sequence and B-spline order.

Prior to the selection of a suitable model configuration, a particular characteristic shared among all instrument sound data sets needs to be considered carefully: The discreteness of some of their control parameters, namely I_g, P and the partial frequency parameter $f_{(k)}$, because the property of a discrete distribution of a variable may eventually yield an underdetermined system when estimating the B-spline's weight parameters. This will happen if the amount of basis functions exceeds the available amount of discrete values of the variable within their neighborhood and represents the very extreme case of overfitting introduced in sec. 4.4.

However, an equal issue arises also with the non-discrete variable $I_{l,\gamma}$ if a B-spline's domain limits exceed the available values of $I_{l,\gamma}$ yielding one or more basis functions for which there is no data.

The first and obvious solution to the above issues is to employ a B-spline knot sequence and order which will always yield less basis functions when there will be values of the control variable for its whole domain. Certainly, this requires precise knowledge about the distribution of the variable and may hence not always be available. Another method to solve the issue of extreme overfitting which does not require knowledge about the distribution of the domain variable is to apply a smoothness constraint using the regularization presented in sec. 4.4. Smoothness can be be achieved by penalizing for slope or curvature of the B-spline. Both approaches will be applied for the model configurations depending on the availability of data knowledge or simply according to its ease of use.

We furthermore employ two distinct configurations of the B-spline parameters to retain the generality of the approach but equally accounting for disparate sound and signal characteristics. We hence establish a model configuration suitable for continuously driven sounds as within the trumpet and clarinet as well as within the four violin data sets. The second model configuration aims for impulsively excited sounds like within the piano sound set.

For the sake of simplicity of the approach and of the model description, the configurations of the multivariate B-splines $S_{(k,s)}(\boldsymbol{\Theta}_h)$ shown in eq. (7.1) and $H_{(l,s)}(\boldsymbol{\Theta}_r)$ depicted in eq. (7.43) will be equal for all partial and cepstral coefficients as well as for both temporal segments. Moreover, we establish the multivariate B-splines with equal knot sequences and orders for the harmonic as well as residual model due to their similar control parameters.

8.1.1 A Model for Continuously Driven Instruments

Since we may adjust the B-splines of all components of the instrument model for every control parameter individually, we may analyze them separately and deduce suitable B-splines configurations:

8.1.1.1 Configuring the B-spline for Parameter I_g

Within the trumpet, clarinet and all violin sound sets three different global intensity categories are contained for all pitches, denoted *pp*, *mf* and *ff* as shown in tab. 8.1. To represent these categorial values within our model representation we employ a linear mapping of these three categories to the MIDI velocity values: 1, 64 and 127 respectively to obtain I_g. This will eventually map all measured partial amplitudes and cepstral coefficients to either one out of 3 discrete positions along the global intensity dimension of the multivariate B-spline, which yields enough understanding to adjust the B-spline with respect to the distribution of the data along I_g.

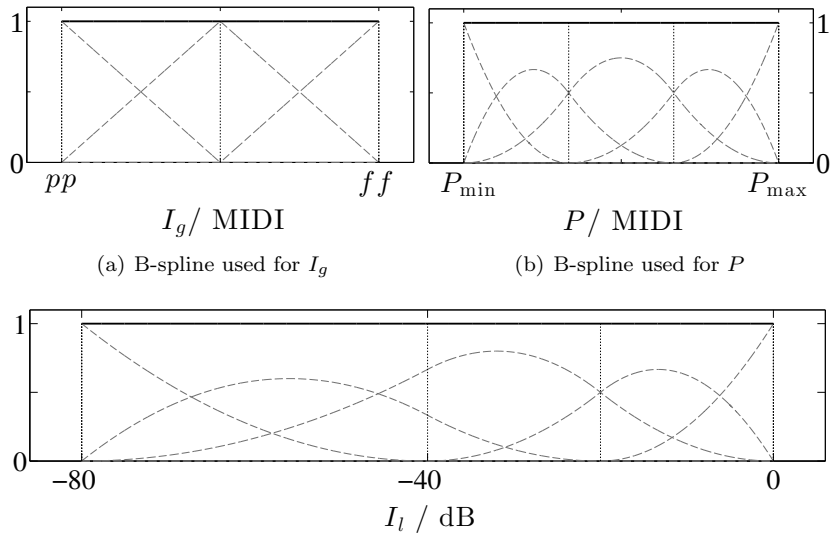
A possible configuration using only 3 basis functions for the univariate B-spline to represent the global intensity dimension is given in fig. 8.1(a). The B-spline is set up using a sequence of knots which are placed at the exact mapping positions for the three categories *pp*, *mf* and *ff* and the basis functions are established using a B-spline order 2, enabling linear interpolation along the global intensity dimension.

The total amount of free weight parameters for the univariate B-spline representing the global intensity hence equal 3.

8.1.1.2 Configuring the B-spline for Parameter P

Like the categorial values of the global intensity, the pitch parameter for all sounds becomes translated to its respective MIDI value yielding a discrete control parameter P which implies a log-frequency metric. In sec. 7.1.2 we have then discussed the model's inherent frequency ambiguity due to the simultaneous use of pitch and frequency parameters, though we have assumed the data to exhibit only minor variations along the pitch dimension of the excitation component of the harmonic model. A B-spline configuration for the pitch parameter may hence be setup again in the MIDI domain and for generality with equal B-spline configurations for all 3 instruments as shown in fig. 8.1(b), though adaptively regarding the instrument's actual pitch range using $P_{\min}$ and $P_{\max}$ to refer to their min and max values. The B-spline eventually exhibits an order of 3 to enable quadratic interpolation and consists of three homogeneously partitioned segments.

All instrument sound data sets with continuous excitation exhibit similar amounts of separately recorded pitches which will be translated to discrete positions in between $P_{\min}$ and $P_{\max}$ and hence underdetermination could happen for B-spline configurations with an amount of parameters larger than discrete pitches contained within the dataset. However, the B-spline configuration proposed in fig. 8.1(b) exhibits 5 basis functions and all sound data sets contain at least 25 pitches which we distributed at equi-distant positions. This eventually yields an amount of 5 free weight parameters for the pitch parameter.



(a) B-spline used for I_g

(b) B-spline used for P

(c) B-spline used for $I_{l,}$

Figure 8.1: The 3 univariate B-splines used to assemble the tensor-product B-spline $b_p(\mathbf{\Theta}_\gamma)$ which is used within the harmonic and residual component of all instrument models for continuously driven instrument sounds.

8.1.1.3 Configuring the B-spline for Parameter $I_{l,\gamma}$

The values of the local intensities $I_{l,\gamma}$ are in contrast to all other control variables not sampled at just a few discrete locations, but exhibit real values ranging from $-\infty$ to 0dB due to its normalization. To bound them reasonably for our internal representation we use a lower limit of -80dB for all instrument data sets and employ a non-uniform knot sequence for the B-spline as shown in fig. 8.1(c). It may be important to note, that the model's limitation of a maximum dynamic of 80dB does not necessarily delimit the dynamic range of the sound synthesis, but only its capability of representing variations of the sounds spectral characteristics for such a range. In the synthesis application, signal level values below -80dB from its signal maximum may still be synthesized, though their variations may not be considered different from the lower limit of the instrument model.

Also, in contrast to $I_{l,\gamma}$ and P, the B-spline configuration facilitates a non-uniform knot sequence to support more variations of the trajectories at higher values of $I_{l,\gamma}$ and a reduced modeling accuracy in the lower range. This accommodates the idea of having a lesser modeling error for signal segments in the signals sustain region with assumably most important perceptual impact, while keeping the model complexity reasonably low for signal segments of low energy and less impact. The total amount of parameters for the local intensity is hence 5 as can be seen from the amount of basis functions in the fig. 8.1(c)

8.1.1.4 Configuring the B-spline for Parameter $\hat{f}_{(k)}$

The generic B-spline representation for the resonance filter module has been introduced in eq. (7.2) where $R(f)$ is established as a continuous function of frequency, though the parameter estimation technique in eq. (7.6) uses discrete partial frequencies $\hat{f}_{(k)}$ to estimate a smooth filter envelope from which we infer the following implications:

As $R(f)$ represents sound features by partial frequency, we employ a Mel-scale based partitioning to create a non-uniform knot sequence for the B-spline. This shall allow resonances and formants at lower frequencies to be represented with higher accuracy when high frequency content, roughly following the human perception of sound [Moo12].

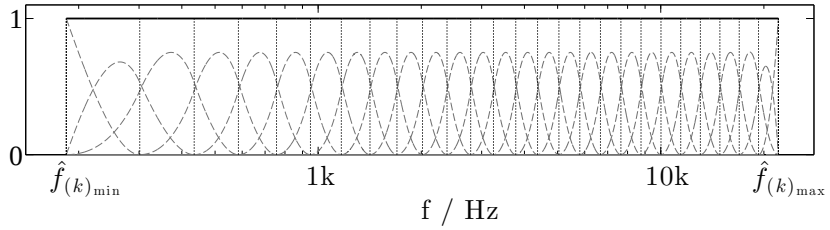


Figure 8.2: The B-spline used for the resonance filter component $R(f)$ within all instrument models with continuously excited sound signals. The B-spline is created using a non-uniform partitioning based on the Mel scale and a B-spline order 3.

During the parameter estimation procedure, the discrete partial frequencies $\hat{f}_{(k)}$ are assumed to reveal a sampled version of the instrument's resonance characteristics ranging from the lowest to the highest partial frequency contained within the dataset. We may therefore estimate the instrument's resonance characteristics only within these limits denoted $\hat{f}_{(k)_{\min}}$ and $\hat{f}_{(k)_{\max}}$ respectively and hence establish the B-spline exactly within these limits.

In the first low octaves of the frequency function, the partial frequencies will reveal a highly subsampled version of the resonance filter function and underdetermination may occur. We have therefore used 26 distinct segments for partitioning the frequency axis for all instruments obtaining B-spline segments that are always determined for the used data sets.

Figure 8.2 depicts such a B-spline with increasing segment sizes and lower and upper bounds to be determined by the frequency range of an instrument's harmonics. Its overall amount of free weight parameters eventually accumulates to 28.

8.1.1.5 Parameter Space Analysis

In eq. (4.20) we have given a formula how the amount of free parameters for a tensor-product B-spline can be determined given the amount of parameters of its univariate components. The amount of free parameters for a single multivariate B-spline $S_{(k,s)}(\boldsymbol{\Theta}_h)$ or $H_{(l,s)}(\boldsymbol{\Theta}_r)$ used to represent one coefficient of one temporal segment therefore becomes $3 \cdot 5 \cdot 5 = 75$.

8.1.2 A Model for Impulsively Driven Instruments

The piano sound database differs from the other data sets as it exhibits more than twice the amount of global intensities and a substantially bigger pitch range which also enlarges the frequency range of the quasi-harmonic partials. The increased amount of data is assumed to be accompanied by an increased amount of inherent sound features and also due to the specific characteristics of impulsively excited instruments, we employ more complex B-spline representations for the respective components of the instrument model.

8.1.2.1 Configuring the B-spline for Parameter I_g

The piano data set contains up to 8 levels of global intensity, whereas the amount per pitch decreases with increasing pitch. The upmost pitches within the data set contain only 3 intensities levels. As for the continuous model above, the global intensity values have been linearly mapped onto MIDI vecocity scale, though depending on the available amount of levels per pitch. The mapping has hence be made pitch dependent such that for every pitch the lowest intensity got mapped to the minimum and the highest intensity to the maximum velocity following the assumption, that the data set contained the minimum and maximum playable intensity for every pitch but with different amounts of gradual nuances.

We further assume the piano set to exhibit a higher variability for variations of the global intensity of the excitation component and therefore create a B-spline with an order 3 and 5 equal-sized segments shown in fig. 8.3(a) eventually yielding 7 basis functions with their respective free parameters.

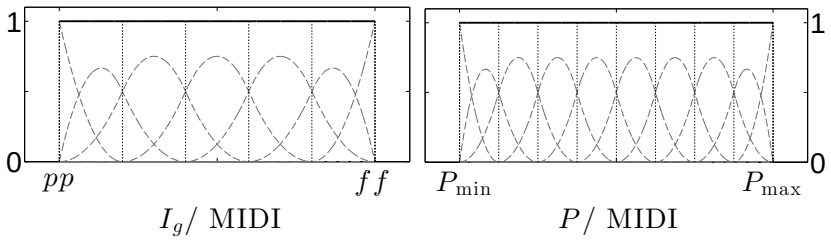
8.1.2.2 Configuring the B-spline for Parameter P

The pitch parameter for the piano sound data set is set up equal to the other data sets but as the piano entails about twice as many pitches as the other sets, we utilize a B-spline which allows for more complex data representations to account for an assumably higher variability of piano sounds of different pitches. The according B-spline with its underlying basis functions is shown in fig. 8.3(b) using 10 basis function and hence free parameters.

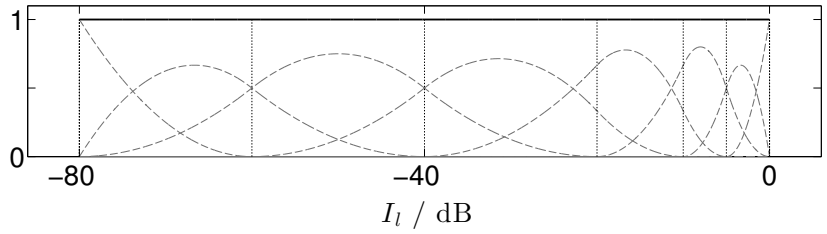
8.1.2.3 Configuring the B-spline for Parameter $I_{l,\gamma}$

The local intensity of impulsively excited instruments are characterized by an attack-release characteristic with a short attack slope and a longer release tail and in contrast to the continuously excited sounds, an impulsively excited signal may have its significant timbral variations for a much wider range of local intensities values. Therefore, we use a B-spline configuration for the local intensity dimension which allows to represent more complex data trajectories as for continuously excited sounds. The employed B-spline is shown in fig. 8.3(c) using again a non-uniform domain partitioning with a increasing density of basis functions at the maximum of the local intensity to allow for higher model precision at the signals amplitude peak.

The amount of free parameters for the local intensity hence increased to 8.



(a) 2nd order B-spline used for I_g (b) B-spline used for P



(c) B-spline used for $I_{l,}$

Figure 8.3: The 3 univariate B-splines used to assemble the tensor-product B-spline $b_p(\mathbf{\Theta}_\gamma)$ which is used within the harmonic and residual component of the piano instrument model.

8.1.2.4 Configuring the B-spline for Parameter $\hat{f}_{(k)}$

Again, we denote the lowest fundamental frequency of the data set $\hat{f}_{(k)_{\min}}$ and in case of the piano this frequency is much lower than for the other data sets. Therefore, to cover the very low frequency content and to still obtain a similar resolution across the whole frequency range as for the continuously excited instruments, we employ a different scale for a non-uniform partitioning.

The different partitioning is based on octaves as it subdivides every octave starting with $\hat{f}_{(k)_{\min}}$ into 5 subsegments which represent the knot sequence used to create the B-spline. In comparison to the Mel scale while using an equal amount of parameters, this partitioning of the frequency axis yields a more dense segmentation for the lower octaves and larger segments for the very high frequencies. This will allow are more precise representation of the resonance characteristics of the piano without increasing the amount of parameters to much.

Using 5 segments per octave, the B-spline eventually requires 80 basis functions for the whole frequency axis with an according amount of free parameters.

8.1.2.5 Parameter Space Analysis

The amount of parameters for the multivariate B-spline $S_{(k,s)}(\mathbf{\Theta}_h)$ or $H_{(l,s)}(\mathbf{\Theta}_r)$ for a single coefficient and for one temporal segment eventually equals $7 \cdot 10 \cdot 8 = 560$.

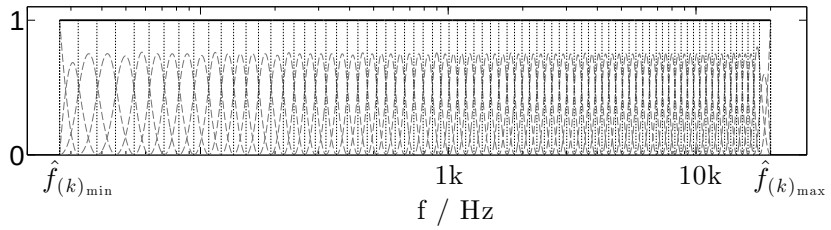


Figure 8.4: The B-spline used for the resonance filter component $R(f)$ for the piano sounds. The B-spline is created using a non-uniform partitioning based on an octave scale and a B-spline order 3.

8.2 Initial Regularization Weights

As already pointed out in sec. 4.4, the initial regularization parameters $\lambda_{\mathbf{I},0}$ and $\lambda_{\mathbf{II},0}^{(z,d)}$ as well as the respective scaling polynomials η need to be adjusted manually with respect to certain desired model characteristics. We have identified these desirable model properties to be smooth modeling trajectories even in subspaces of the model where no data is available and hence overfitting an apparent issue as well as the solution to the inherent ambiguities.

All values given below have been found by separately training instrument models for a large variety of possible regularization parameter values until a set of values had been found that yielded smooth surfaces for all partial and cepstral coefficients. The smoothness of the surfaces had been evaluated visually using figures as shown in ch. 9 and by that we were able to identify two general sets of parameters that have shown to be suitable for the two used classes of instrument sounds. Thus we present two distinct sets of parameter values which represent possible configurations for either one class of sound signals.

8.2.1 Initial Weights for Continuously Excited Instruments

For the harmonic model of the first type of instrument sounds we employ the initial regularization parameters for $\lambda_{\mathbf{I},0}$ and $\lambda_{\mathbf{II},0}^{(z,d)}$ shown in tab. 8.3. The regularization parameter $\lambda_{\mathbf{I},0}$ refers to the penalty of a potential DC offset of the resonance curve while all other parameters constrain the excitation component.

The excitation component for the continuously excited sounds hence receive slope and curvature penalties for its pitch and local intensity dimension indicated by $\lambda_{\mathbf{II},0}^{(P,1)}$ and $\lambda_{\mathbf{II},0}^{(P,2)}$ as well as $\lambda_{\mathbf{II},0}^{(I_{l,h},1)}$ and $\lambda_{\mathbf{II},0}^{(I_{l,h},2)}$ respectively. This selection of regularization terms not only solves the ambiguity of frequency and log-frequency dependency but also enforces smoothness of the estimated surface along both variables. This is an important requirement due to the use of equally constructed B-spline surfaces for all partials, though not all partials will be available for all pitches or may not be present at low values of the local intensity. The regularization using slope and curvature penalties will then enforce the surface to extrapolate smoothly from regions with partial data into regions without.

The polynomial η used to locally amplify a certain regularization is set to

I/II	λ_0	η	J
$\lambda_{\mathbf{I},0}$	.01		100
$\lambda_{\mathbf{II},0}^{(P,1)}$	.25	$[1, 0]$	25
$\lambda_{\mathbf{II},0}^{(P,2)}$	.01	$[1, 0]$	25
$\lambda_{\mathbf{II},0}^{(I_{l,h},1)}$	.01	$[1, 0]$	25
$\lambda_{\mathbf{II},0}^{(I_{l,h},2)}$	.001	$[-2.5, 0]$	25

Table 8.3: Specific values for the regularization parameter $\lambda_{\mathbf{I},0}$ and $\lambda_{\mathbf{II},0}$ used for the harmonic models of all continuously excited instruments as well as the polynomial coefficients for the local emphasis function η in decreasing order from left to right. The last column shows the amount of virtual data points used for the respective regularization.

the identity function for all but the last regularization nullifying its impact except for the curvature penalty along the local intensity dimension. Its coefficients have been chosen with respect to the definition of its domain range and hence accentuates the regularization parameter by a factor of 200 at -80dB to enforce a smooth fade of the surface with constant slope if no data is available at such low signal levels. At 0dB however, η will make the regularization become zero to enable the surface to only represent the data rather than the constraints penalty for a signals maximum value.

I/II	λ_0	$\eta()$	J
$\lambda_{\mathbf{II},0}^{(P,1)}$	.01	[10]	25
$\lambda_{\mathbf{II},0}^{(P,2)}$	.01	[10]	25
$\lambda_{\mathbf{II},0}^{(I_{l,r},1)}$	.01	[10]	25
$\lambda_{\mathbf{II},0}^{(I_{l,r},2)}$	.01	[10]	25

Table 8.4: Regularization weight values for the residual models of all continuously excited instruments and polynomial coefficients for the additional scaling function. The last column shows the amount of virtual data points used for the respective regularization.

The initial weights for slope and curvature penalties along P and $I_{l,r}$ for the residual model are presented in tab. 8.4. Their values are all set to .01 making the regularization becoming as influential as a 100th of the data and hence only effective in regions of very sparse data. Though, as the signal analysis yielded 15 cepstral coefficients from all recordings regardless of their pitch or global intensity, their distribution within the model space will be uniformly across these dimension and their distribution will only vary along the local intensity. It may therefore be assumed that these regularization weights will only affect the estimated surface at low values of the local intensity.

8.2.2 Initial Weights for Impulsively Excited Instruments

For impulsively excited signals we only employ curvature penalties for the excitation component as shown in tab. 8.5 to support the sloping character of the partial trajectories of such sound signals. Therefore, slope penalties have been removed and now all 3 dimensions of the excitation component got second order penalties only. The global intensity has been added to the regularizations because the piano sound data set exhibits more possible global intensity values as the data sets of continuously excited instrument sounds and we will therefore desire some additional smoothing.

For the second order penalties of the pitch and local intensity dimensions a third order polynomial has been introduced to emphasize the regularization using an S-shaped curve. Such a curve allows to emphasize the regularization in the lower range with similar strength while simultaneously reducing the impact of the regularization in the upper range or vice versa. The two polynomials given in tab. 8.5 hence provide a similar emphasis of the regularization of the local intensity in the lower half of the dynamic range while reducing its impact in the whole upper half. The polynomial for the pitch dimension strengthens the regularization in the region for the upper pitches where not always partial data will be present and at the same time reduces the influence of the regularization in the lower pitch range where partial data can be guaranteed for most pitches.

I/II	λ_0	$\eta()$	J
$\lambda_{\mathbf{I},0}$	.01	$[1, 0]$	200
$\lambda_{\mathbf{II},0}^{(P,2)}$	.1	$[1, .7, .3, .1]$	25
$\lambda_{\mathbf{II},0}^{(I_g,2)}$	.1	$[1, 0]$	25
$\lambda_{\mathbf{II},0}^{(I_{l,h},2)}$	.1	$[1, .7, .3, .1]$	25

Table 8.5: Values for the regularization parameter $\lambda_{\mathbf{I},0}$ and $\lambda_{\mathbf{II},0}$ used for the harmonic model of the piano sound set as well as their respective polynomial coefficients for the local emphasis function η in decreasing order from left to right. The last column shows the amount of virtual data points used for the respective regularization.

The residual component gets also constrained for curvature only to also support the sloping characteristics of the cepstral trajectories, though we do not require any additional scaling polynomial for some local emphasis of the regularization.

8.3 Model Training

Training of the instrument model is carried out by means of applying the SCG method introduced in ch. 4.2 to iteratively update the free model parameters of the harmonic and residual component of the model respectively. The parameters of the model's components are estimated independently since both components are established independently.

I/II	λ_0	$\eta()$	J
$\lambda_{\mathbf{II},0}^{(P,2)}$	.1	$[1,0]$	25
$\lambda_{\mathbf{II},0}^{(I_g,2)}$	.05	$[1,0]$	25
$\lambda_{\mathbf{II},0}^{(I_{l,h},2)}$	.05	$[1,0]$	25

Table 8.6: Regularization weight values for the residual model of the piano sound set and the polynomial coefficients for the additional scaling function all set to the identity function. The last column shows the amount of virtual data points used for the respective regularization.

For an overview over the computational complexity of the parameter estimation procedures, tab. 8.7 lists the amount of free parameters for the harmonic model whereas tab. 8.8 shows the amounts for the residual models. The total sum of free parameters in tab. 8.7 for each instrument is indicated by $\dim(\widetilde{\mathfrak{w}})$, whereas it is mainly being determined by the size of the excitation component $\dim(\widetilde{\mathbf{w}}_{(k,s)})$ which gets multiplied by the amount of partials K contained within the dataset. The piano model hence exhibits by far the most free parameters.

instrument	$\dim(\widetilde{\mathbf{w}}_{(k,s)})$	$\dim(\widetilde{\mathbf{w}}_{(R)})$	K	$\dim(\widetilde{\mathfrak{w}})$	$\dim(\mathbf{A})$
trumpet	75	28	109	16378	13011064
clarinet	75	28	144	21628	10795668
violin (str. 1)	75	28	33	4978	3674401
violin (str. 2)	75	28	49	7378	5588809
violin (str. 3)	75	28	73	10978	6786295
violin (str. 4)	75	28	109	16378	7337112
piano	560	80	230	257680	48237914

Table 8.7: The amount of free model parameters for the excitation and resonance component of each harmonic model as well as the amount of partials contained within the respective data set and the resulting amount of free parameters for the whole model additionally taking the temporal segmentation into account. The last column shows the amount of data used to estimate the parameters for the harmonic model.

The last column in tab. 8.7 lists the overall amount of discrete partial data values $A_{(k)}(n)$ of all recordings contained within the respective data set. A hypothetical transformation matrix for the linear system of equations for a single harmonic model would hence be as large as the product of the values of its last two columns.

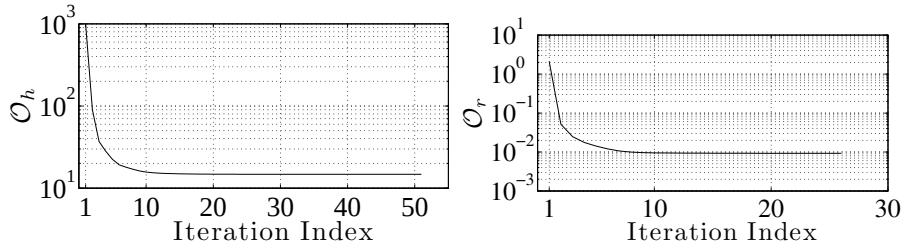
The amount of free parameters of the residual models only differs according to their signal excitation model and hence tab. 8.8 only discriminates between continuously and impulsively excited signals. Though their sizes of the multivariate B-spline are equal to the source component of the harmonic model,

instrument class	$\dim(\widetilde{\mathbf{w}}_{(l,s)})$	L	$\dim(\widetilde{\mathfrak{w}})$
continuously	75	16	1200
impulsively	560	16	8960

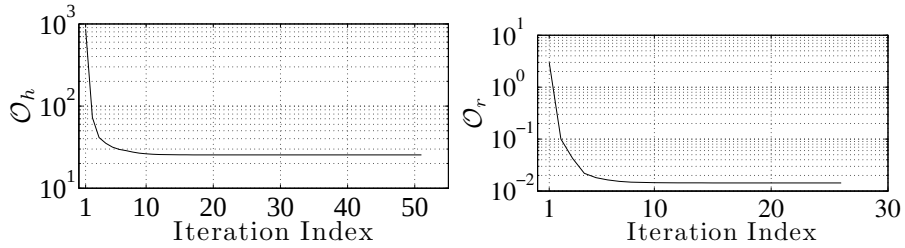
Table 8.8: The amount of free model parameters for the multivarate B-spline representation for every single cepstral coefficient, the amount of cepstral coefficients being modeled by the residual model and the overall amount of free parameters additionally taking the temporal segmentation into account.

their amounts of cepstral coefficients are much smaller and hence their overall amounts.

Estimation of the free parameters starts with all free parameters set to zero and iteratively updates their values using the SCG method. The figures in 8.5 display the convergence behavior while learning the free parameters of the harmonic and residual models using either the trumpet or clarinet dataset. For both model components convergence is reached already after about 10 to 20 iterations which is mainly due to the use of a preconditioning method as not using such a strategy for scaling the error surface required up to 100 times the amount of iterations for ensured convergence.



(a) Error of the harmonic model component (b) Error of the residual model component



(c) Error of the harmonic model component (d) Error of the residual model component

Figure 8.5: Convergence property of the objective functions $\mathcal{O}_h$ and $\mathcal{O}_r$ for the two model components of the trumpet (top) and B$\flat$-clarinet (bottom) sound data set.

8.4 Conclusion

The box below gives a short summary of the applied method for obtaining the parameters of an instrument model given a certain data set of recordings.

- Analysis Phase
  1. Get sound signal representations $A_{(k)}(n)$ and $C_{(l)}(n)$ for all sounds using the methods explained in sec. 5.1.
  2. Get control parameter signals $\mathbf{\Theta}_h(n)$ and $\mathbf{\Theta}_r(n)$ from all signals following the procedure in sec. 5.2 and get its derived controls $\tau_{\gamma,s}$ and $\varphi_{\gamma,s}$ for $\gamma \in \{h, r\}$ and $s \in \{1, 2\}$ as well as $\hat{f}_{(k)}$ to eventually establish $\mathbf{\Psi}_h$ and $\mathbf{\Psi}_r$ for every available recording.
- Pre-Training Phase
  4. Determine the B-spline values for the input control parameters of the harmonic and residual sound representations $b_{(S)}(\mathbf{\Theta}_h(n))$, $b_{(R)}(\hat{f}_{(k)})$ and $b(\mathbf{\Theta}_r(n))$ using either a multivariate or univariate model as presented in ch. 4.
  5. Determine $\lambda_{\mathbf{I}}$, $\lambda^{(z,d)}_{\mathbf{II},(k,s)}$ and $\lambda^{(z,d)}_{\mathbf{II},(l,s)}$ for all k, l and s as well as the selected values of z and d using eq. (7.12), (7.15) and (7.51) respectivley.
  6. Determine coefficients $\mathbf{c}_{(k,s)}$, $\mathbf{c}_{(R)}$ and $\mathbf{c}_{(l,s)}$ for preconditioning the matrixes of the harmonic and residual model using eq. (7.41), (7.42) or (7.62).
- Training Phase
  7. Estimate the weights $\widetilde{\mathbf{w}}_{(k,s)}$ and $\widetilde{\mathbf{w}}_{(R)}$ using the objective function (7.17)
  8. Estimate the weights $\widetilde{\mathbf{w}}_{(l,s)}$ using the objective function (7.49)

Chapter 9

Visual Evaluation

Introduction

To assess the selection of the initial constraint coefficients in terms for their impact on the smoothness of the estimated surfaces and curves for the harmonic as well as residual components of an instrument model we utilize visualizations of the internal representations. These are hence established by means of revealing the fit of the internal representation to its respective data. Therefore, in all our visualizations of the model's components we show the model as surfaces and its according data using point clouds such that the figures not only allow to assess the smoothness of the fit but also allow to discuss the selected B-spline configurations.

To visualize the threevariate B-splines $S_{(k,s)}(\mathbf{\Theta}_h)$ and $H_{(l,s)}(\mathbf{\Theta}_r)$ we make cutouts and create 3D figures. This requires the selection of a single variable to leave out for every figure and for the generation of the model's surface this variable needs to be set to a constant value. Though, for showing the corresponding data values we will not only show data with that specific value of the left out variable, as due to the use of a B-spline for all dimensions, not only data values with that specific variable value will influence the current model's representation. Therefore, every 3D figure will exhibit an additional graph on top to indicate the position of the left out variable within its domain together with the basis functions of the B-spline used for its representation.

This graph also contains a color gradient used to refer to the data's influence on the currently shown internal representation. In this scheme red color refers to data that has a similar value for the left-out variable as the currently shown surface. Purple and yellow colored point clouds are used to refer to data whose value of the left-out variable is either below or above the value used to generate the surface. The range of the color gradient depends on the B-spline configuration of the left-out dimension and is determined by the range which has some reasonable impact on the currently shown surface and therefore data that does no influence on the current surface exhibits no color in the top graph and is hence not shown in the figure. Though, there is one exception in the figures with constant global intensity values for instrument sound sets which only exhibit the 3 discrete global intensity values *pp*, *mf* and *ff* for which we show neighboring variable values even though they to not have any impact on the current surface. These subfigures are always located in the right column of

the figures.

The model's representation based on multivariate B-splines is shown as a semi-transparent surface in the figures. It is created using a constant value for the left-out variable which is always shown on top of the top graph where also a rectangle with the color map of the surface is given at its position of the left-out variable. The surface is created along the complete domain ranges of the 2 remaining variables.

On top of each semi-transparent surface every figure depicts the knot grid created by the tensor-product of univariate B-splines depicted as black lines connecting the intersections of the segments of the univariate B-splines. The black lines represent linearly interpolated intersection points, though the model's surface may exhibit non-linear characteristics between them.

9.1 Harmonic Model Component

The harmonic model represents partial amplitude data $A_{(k)}(n)$ separately by an excitation $S_{(k,s)}(\mathbf{\Theta}_h)$ and a resonance component $R(\hat{f}_{(k)})$ each contributing to the data by an estimated amount. To visualize either component individually but jointly displaying the data it actually represents requires the contribution of the other component to be subtracted from the data. This can be seen from some simplified math neglecting the aspect of temporal segmentation:

Assuming the estimate of the partial data to be $\hat{A}_{(k,s)}(\mathbf{\Psi}_h)$ representing an instrument sounds datum $A_{(k)}(n)$ with some residual error ϵ:

$$A_{(k)}(n) = \hat{A}_{(k,s)}(\mathbf{\Psi}_h(n)) + \epsilon \tag{9.1}$$

The estimate of the partial data is established by means of the harmonic model which contains the excitation and resonance component as in:

$$A_{(k)}(n) = S_{(k,s)}(\mathbf{\Theta}_h) + R(\hat{f}_{(k)}) + \epsilon \tag{9.2}$$

To eventually visualize either one component and to display the data it represents will hence require the other component to be subtracted from the original data yielding the respective partial amplitude represented by the excitation component:

$$A_{(k)}(n) - R(\hat{f}_{(k)}) = S_{(k,s)}(\mathbf{\Theta}_h) + \epsilon \tag{9.3}$$

and similarly for the resonance component:

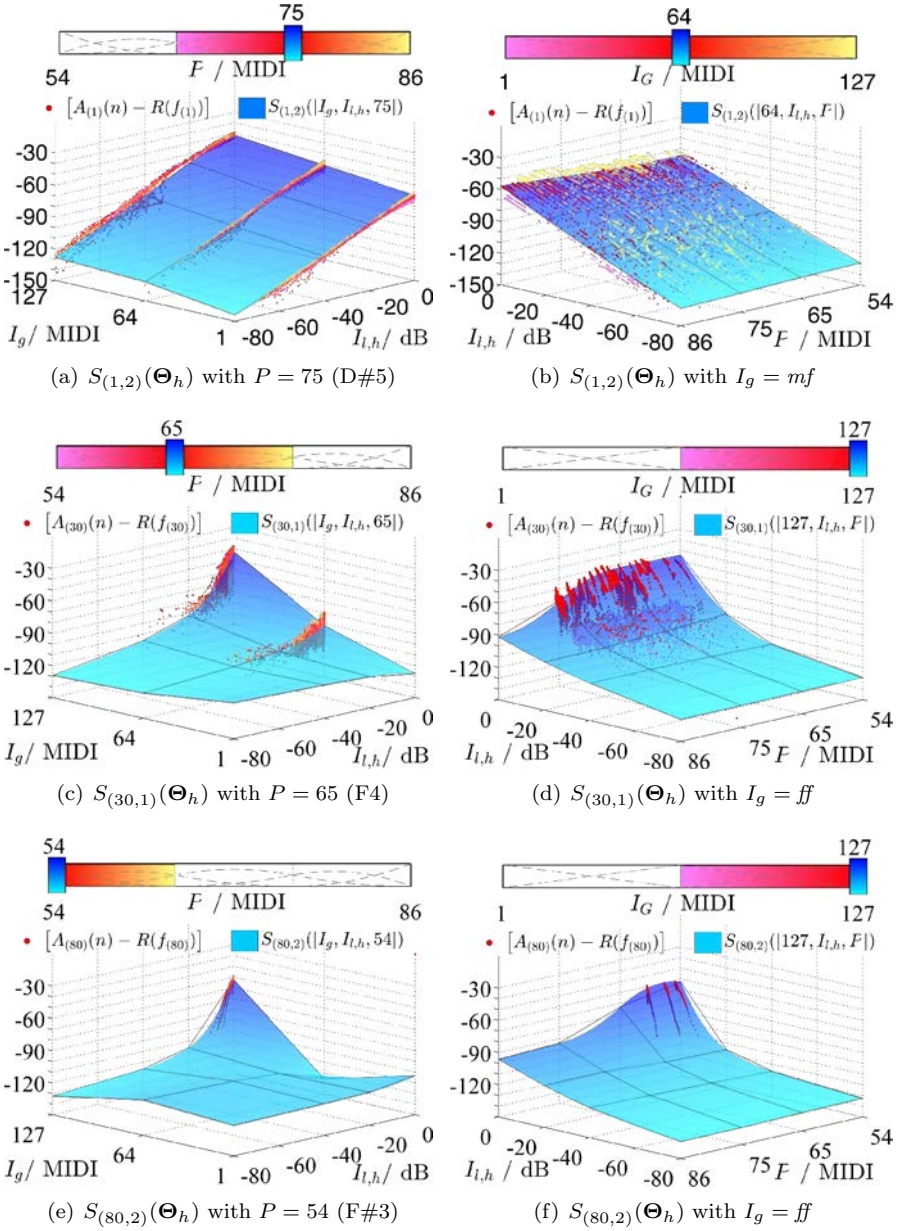
$$A_{(k)}(n) - S_{(k,s)}(\mathbf{\Theta}_h) = R(\hat{f}_{(k)}) + \epsilon \tag{9.4}$$

This also infers that the same residual error from the estimation procedure will be present in both visualizations of the harmonic model.

9.1.1 Trumpet

Six subfigures are shown in fig. 9.1 each depicting a part of the excitation component of the harmonic model $S_{(k,s)}(\mathbf{\Theta}_h)$ learned from the trumpet data set together with its respective data obtained using eq. (9.3). Subfigures 9.1(a) and 9.1(b) display the models estimated surface for the trajectory of the first

partial index hence fundamental and second temporal segment. In the left subfigure the surface is plotted using a constant pitch MIDI of 75 (D#5) while the right figure shows the surface for a constant global intensity MIDI value 64 referring to *mf*. In the subfigures 9.1(c) and 9.1(d) surfaces for partial index $k = 30$ and $s = 1$ are shown while subfigures 9.1(e) and 9.1(f) show such for $k = 80$ and $s = 2$.



(a) $S_{(1,2)}(\boldsymbol{\Theta}_h)$ with $P = 75$ (D#5)

(b) $S_{(1,2)}(\boldsymbol{\Theta}_h)$ with $I_g = mf$

(c) $S_{(30,1)}(\boldsymbol{\Theta}_h)$ with $P = 65$ (F4)

(d) $S_{(30,1)}(\boldsymbol{\Theta}_h)$ with $I_g = ff$

(e) $S_{(80,2)}(\boldsymbol{\Theta}_h)$ with $P = 54$ (F#3)

(f) $S_{(80,2)}(\boldsymbol{\Theta}_h)$ with $I_g = ff$

Figure 9.1: Visualizations of the excitation component $S_{(k,s)}(\boldsymbol{\Theta}_h)$ of the harmonic model for the trumpet data set together with its respective partial amplitude data.

All figures reveal the discrete property of the used values for the signals global intensity variable I_g as well as their pitch P by the strictly directional arrangement of the partial amplitude data displayed as dots in the figures. The partial amplitudes of a single recording hence align along a single global intensity and pitch value and only exhibit varying real values for its local intensity $I_{l,h}$. For any non-redundant set of recordings like the trumpet database every sound sample will hence have a trajectory within the model space at a location which is non-overlapping with any other recording. But as we display data for several values of the left-out dimension some trajectories may still overlap though having unique colors to indicate their disparate position in the model's space.

From the figures one may also observe the absence of partials with index 30 and 80 at upper pitches and low intensity values indicating a similar behavior for all other partial indexes which are not displayed. The absence of partials with higher index at upper pitches is due to the Nyquist frequency as such partials have not been present in the whole data set of recordings. The fact why partials with high index are not present at low values of the global as well as local intensity results from the signal analysis strategy, where spectral data has been rejected from the sinusoidal model which exhibited very low energy values or values that were too close to the residual noise.

Within all figures shown, the estimated multivariate trajectories of the excitation source model of the harmonic component represented as semi-transparent surfaces adapt to the given data as closely as possible regarding the chosen knot grid. The surfaces also smoothly extrapolate into regions where no partial data is present according to the chosen regularization using first and second order derivative penalties along the pitch and local intensity dimension.

The estimated characteristics for the resonance component $R(f)$ as well as the according partial amplitude values of the trumpet data set are shown jointly in fig. 9.2. As for the source component, the contribution of the not displayed model component has been subtracted from the partial amplitude values as in eq. (9.4) and the resulting partial amplitudes have been placed at their ideal frequency values $\hat{f}_{(k)}$ in the figure.

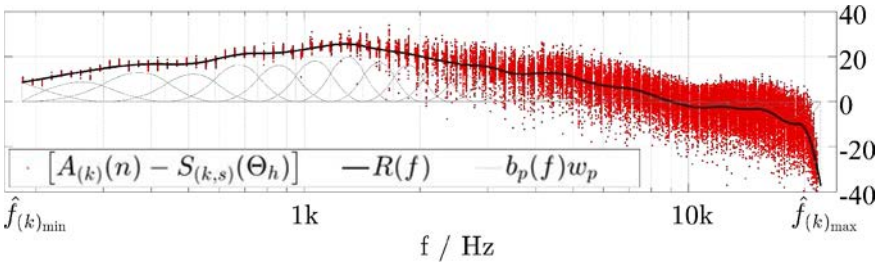


Figure 9.2: Visualization of the resonance component $R(f)$ of the harmonic model for the trumpet data set together with its respective partial amplitude data located at their ideal frequency locations $\hat{f}_{(k)}$.

It can be observed, that the partial frequency values in the low registers are placed at discrete positions with uniform distances which is due to the use of the ideal partial frequency values. The actual distance in the lowest 2 octaves

between neighboring partial frequency values refers to the semitone distance of the fundamental frequency of adjacent pitches. This becomes an important issue when choosing a certain B-spline configuration with high fidelity in the low register as it may result in overfitting and hence would require additional regularization treatment.

From the figure we may further observe that the parameter estimation procedure obtained a curve for the resonance characteristics that closely aligns to the mean of the data and it may further be concluded, that the variance of the data is increasing with increasing frequency.

Eventually, all figures for the source and resonance component reveal the non-uniform data distribution within the generated model space and hence prove the necessity for the various regularization strategies and also the use of a preconditioning method for an acceleration of the parameter estimation procedure.

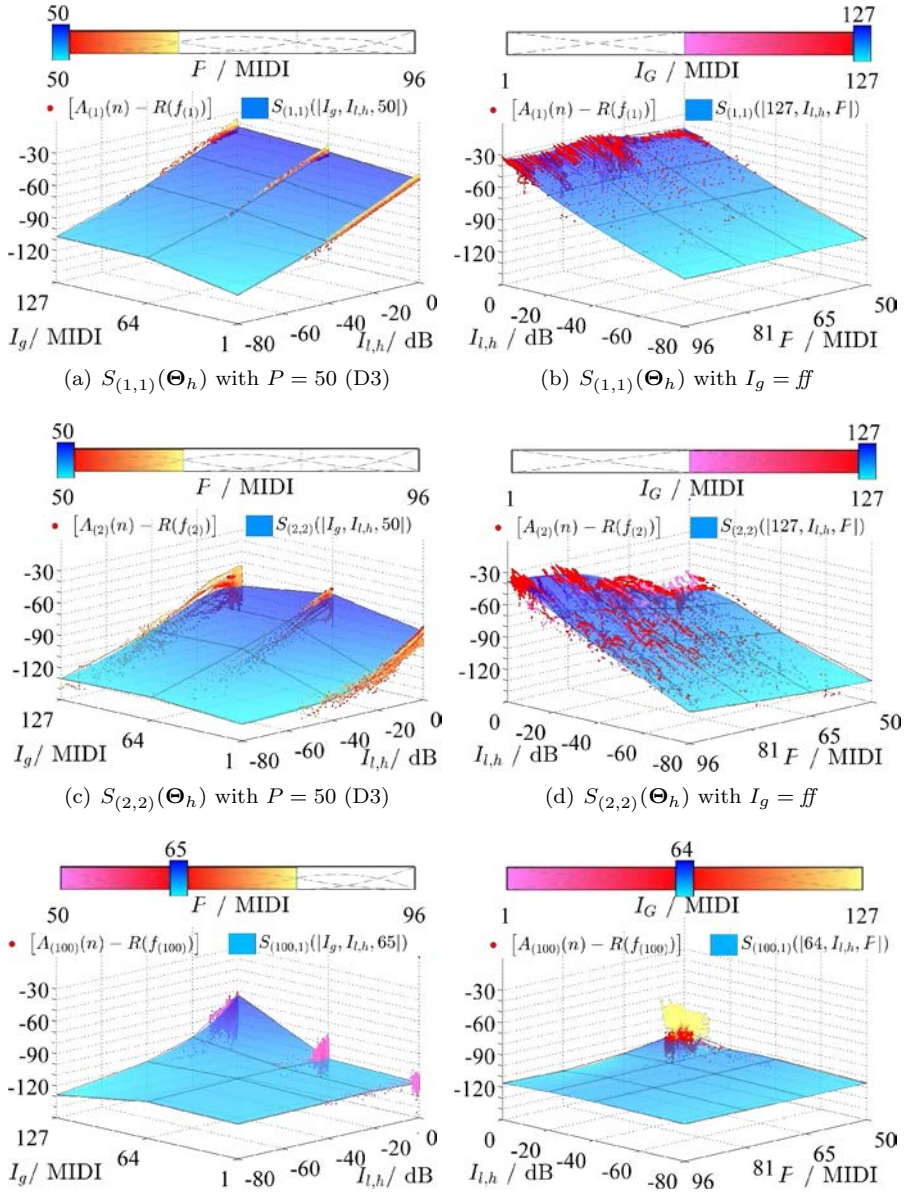
It may also derived from the figures, that the used B-spline configuration to setup the excitation component of the harmonic model represents a suitable trade-off balancing data fit and computational efficiency.

9.1.2 Clarinet

To visualize the excitation component of the instrument model trained using the clarinet data set, cutouts for partial indexes $k = \{1, 2\}$ and $k = 100$ are selected as shown in fig. 9.3. The figures exhibit a lot of features similar to the ones of the model and data of the trumpet, but we may also derive a sound signal property specific for the clarinet instrument. The well-known characteristic of the clarinet of exhibiting weak amplitudes for partials with an even index can be observed by comparing the estimated excitation surfaces of the first and second partial index. There, a level difference of about 40dB between the first and second partial can be observed at low pitch and high global intensity values. Interestingly, this characteristic property of clarinets diminishes with increasing pitch and essentially disappears for its highest pitches.

Similarly to the trumpet data set, partial amplitude data with high index is not present in subspaces of low local or global intensity as well as for high pitch values.

The estimated resonance function of the clarinet set is shown in fig. 9.4 showing more variance in the mid range than for the trumpet set.



(a) $S_{(1,1)}(\boldsymbol{\Theta}_h)$ with $P = 50$ (D3)

(b) $S_{(1,1)}(\boldsymbol{\Theta}_h)$ with $I_g = \mathit{ff}$

(c) $S_{(2,2)}(\boldsymbol{\Theta}_h)$ with $P = 50$ (D3)

(d) $S_{(2,2)}(\boldsymbol{\Theta}_h)$ with $I_g = \mathit{ff}$

(e) $S_{(100,1)}(\boldsymbol{\Theta}_h)$ with $P = 65$ (F4)

(f) $S_{(100,2)}(\boldsymbol{\Theta}_h)$ with $I_g = \mathit{mf}$

Figure 9.3: Visualizations of the excitation component $S_{(k,s)}(\boldsymbol{\Theta}_h)$ of the harmonic model for the clarinet data set together with its respective partial amplitude data.

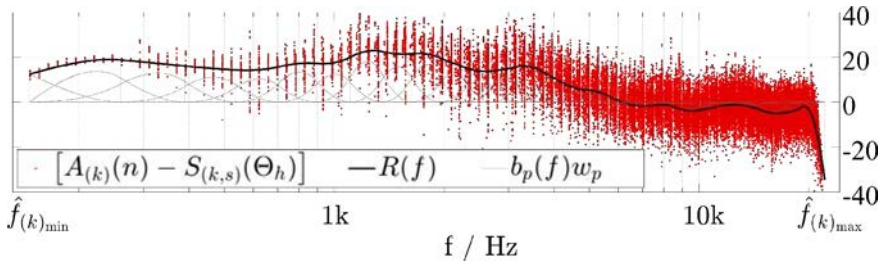


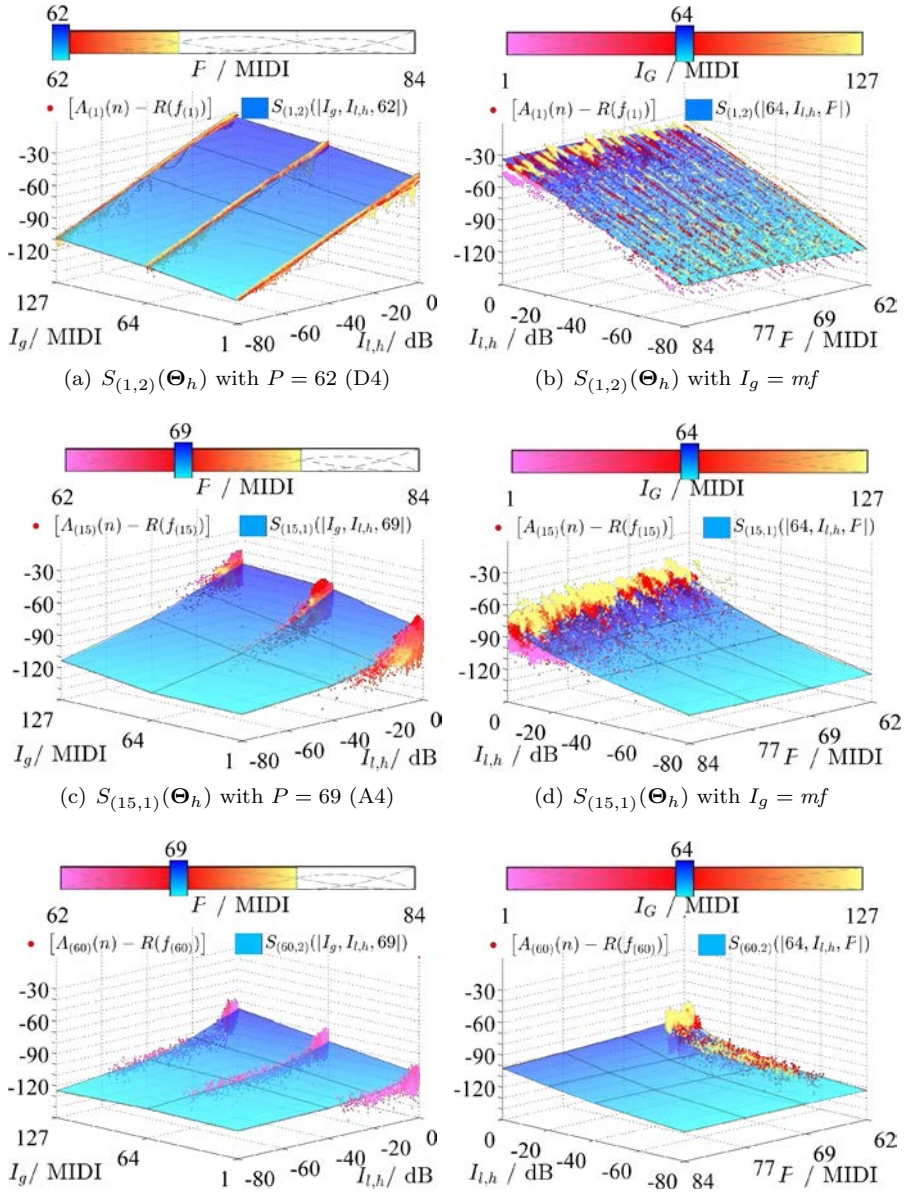
Figure 9.4: Visualization of the resonance component $R(f)$ of the harmonic model for the clarinet data set together with its respective data located at their ideal frequency locations $\hat{f}_{(k)}$.

9.1.3 Violin

The violin data set represents a special case since we have divided the database into separate sets according to the string used to play the various pitches. In the subfigures of fig. 9.5 we only show cutouts of the excitation module of second lowest string denoted as the third.

Again the data and surface properties presented in fig. 9.5 are similar to the previous models for continuously excited instruments, though the data set for the third string of a violin exhibits some more variance for indexes above 10 qs may be observed in the subfigures 9.5(c) - 9.5(f).

The fig. 9.6 contains the estimated resonance functions $R(f)$ for all 4 harmonic models of the violin sound data set. The four estimated curves for $R(f)$ differ in their frequency range since the strings have different pitch ranges and hence the lowest partial frequency contained within each subset is different. Further differences in the curves may result from the different attachments of the strings on its resonating body yielding different positions at which the violins corpus gets excited. However, certain features of the estimated resonance characteristics appear in all curves making them looking fairly similar.



(a) $S_{(1,2)}(\boldsymbol{\Theta}_h)$ with $P = 62$ (D4)

(b) $S_{(1,2)}(\boldsymbol{\Theta}_h)$ with $I_g = mf$

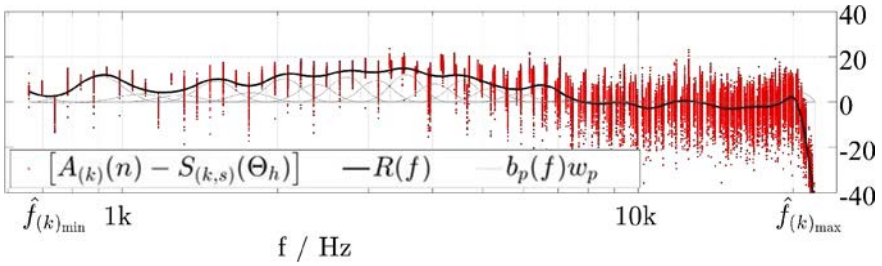
(c) $S_{(15,1)}(\boldsymbol{\Theta}_h)$ with $P = 69$ (A4)

(d) $S_{(15,1)}(\boldsymbol{\Theta}_h)$ with $I_g = mf$

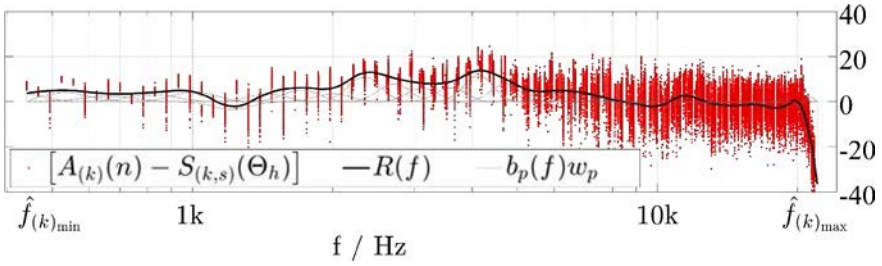
(e) $S_{(60,2)}(\boldsymbol{\Theta}_h)$ with $P = 69$ (A4)

(f) $S_{(60,2)}(\boldsymbol{\Theta}_h)$ with $I_g = mf$

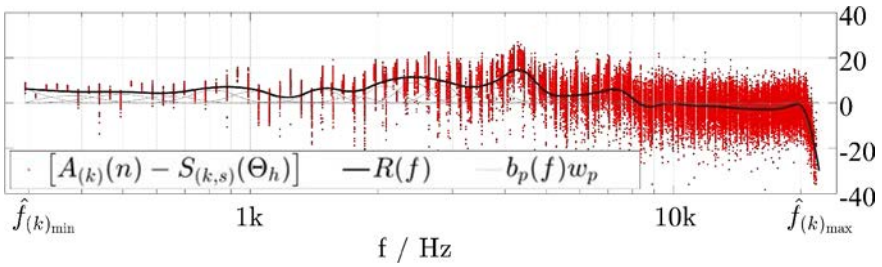
Figure 9.5: Visualizations of the excitation component $S_{(k,s)}(\boldsymbol{\Theta}_h)$ of the harmonic model for the third string subset of the violin data set together with its respective partial amplitude data.



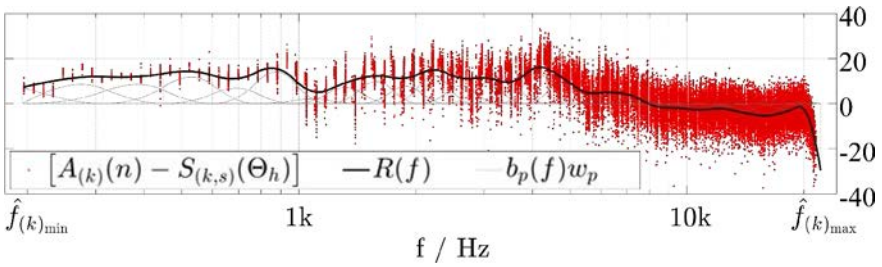
(a) String 1 (E)



(b) String 2 (A)



(c) String 3 (D)



(d) String 4 (G)

Figure 9.6: Visualization of the resonance components $R(f)$ together with its respective data located at their ideal frequency locations $\hat{f}_{(k)}$ of the harmonic models of all 4 strings of the violin data set.

9.1.4 Piano

The grand piano sound data set exhibits by far the most sound examples due to its large pitch range and the fact that every pitch had been recorded at up to 8 different velocities. This fact can be observed within all subfigures of fig. 9.8 as the presented sound data exhibts up to 8 discrete trajectories along its gobal intensity dimension best presented in the left column and hardly distinguishable trajectories along its pitch despite their discreteness.

We may further observe that the partial amplitudes decay more rapidly with increasing partial index and similar to the other sound data sets, partial data is absent for higher indexes and upper pitches as well as for lower global and local intensity values. Furthermore, the data appears to have more variability especially for varying pitch values.

Due to the increased amount of segments to create the knot sequences for the tensor-product B-spline for the piano model, the surface is capable to adapt to the strong data variability and still extrapolates smoothly from subspaces with data into areas without.

The resonance component of the harmonic model trained using the grand piano data set is presented in fig. 9.7 representing the filter envelope with the largest frequency range of all instruments and most complex B-spline configuration.

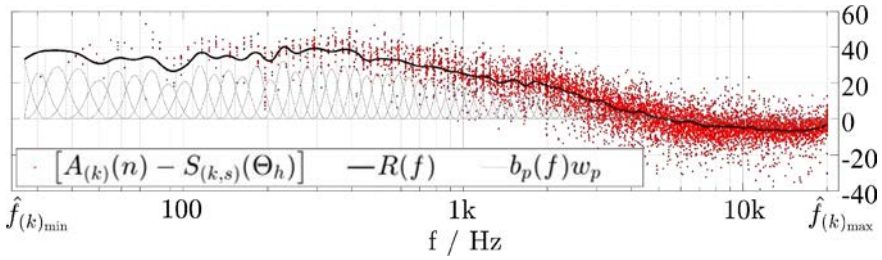
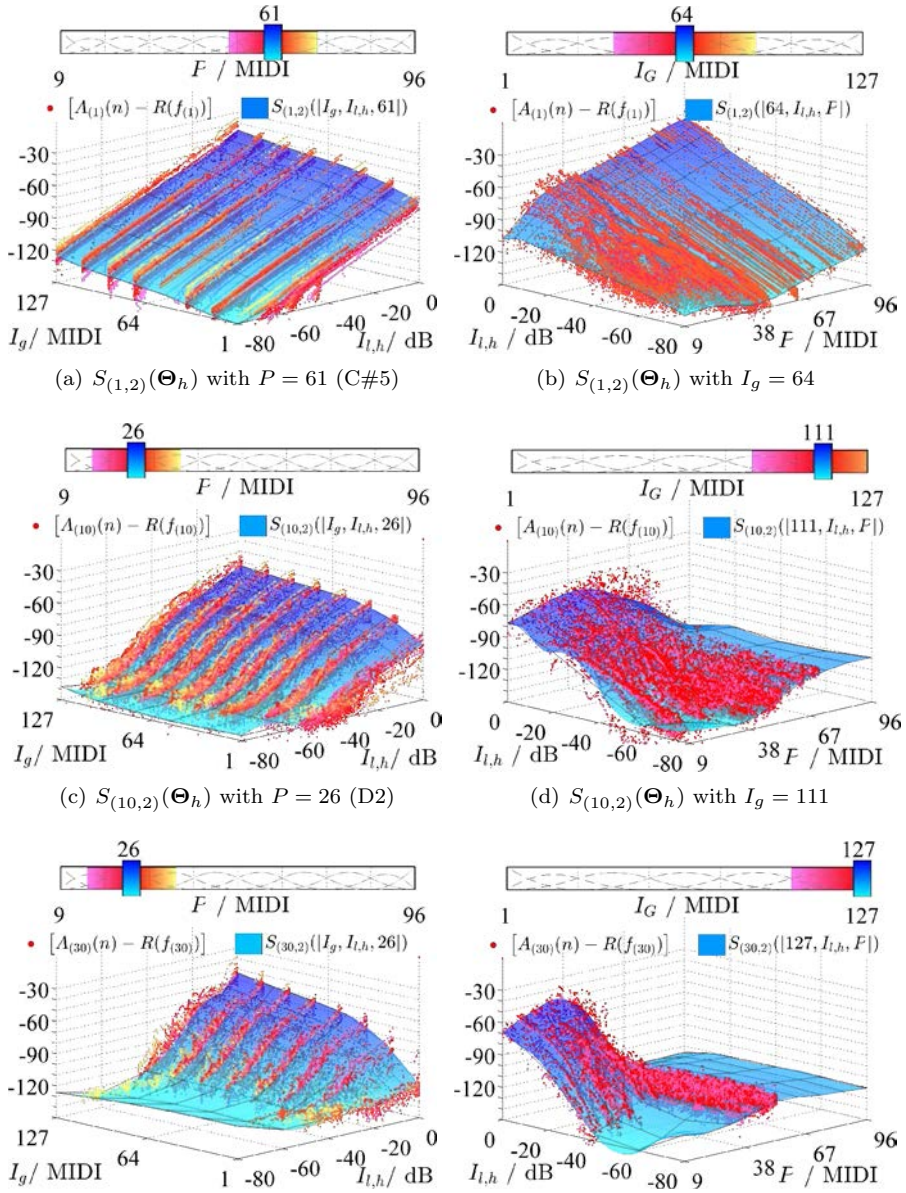


Figure 9.7: Visualization of the resonance component $R(f)$ of the harmonic model for the Fazioli grand piano data set together with its respective data located at their ideal frequency locations $\hat{f}_{(k)}$.



(a) $S_{(1,2)}(\boldsymbol{\Theta}_h)$ with $P = 61$ (C#5)

(b) $S_{(1,2)}(\boldsymbol{\Theta}_h)$ with $I_g = 64$

(c) $S_{(10,2)}(\boldsymbol{\Theta}_h)$ with $P = 26$ (D2)

(d) $S_{(10,2)}(\boldsymbol{\Theta}_h)$ with $I_g = 111$

(e) $S_{(20,2)}(\boldsymbol{\Theta}_h)$ with $P = 26$ (D2)

(f) $S_{(20,2)}(\boldsymbol{\Theta}_h)$ with $I_g = 127$

Figure 9.8: Visualizations of the excitation component $S_{(k,s)}(\boldsymbol{\Theta}_h)$ of the harmonic model for the Fazioli grand piano data set together with its respective partial amplitude data.

9.2 Residual Model Component

The residual component represents the cepstral coefficients of the residual signals as functions of the control variables using only a single three-variate tensor-product B-spline for every cepstral coefficient. Therefore, to visually assess the fit of the data we utilize similar 3D cutouts for some selected cepstral coefficients as for the partial amplitude data and harmonic model surfaces.

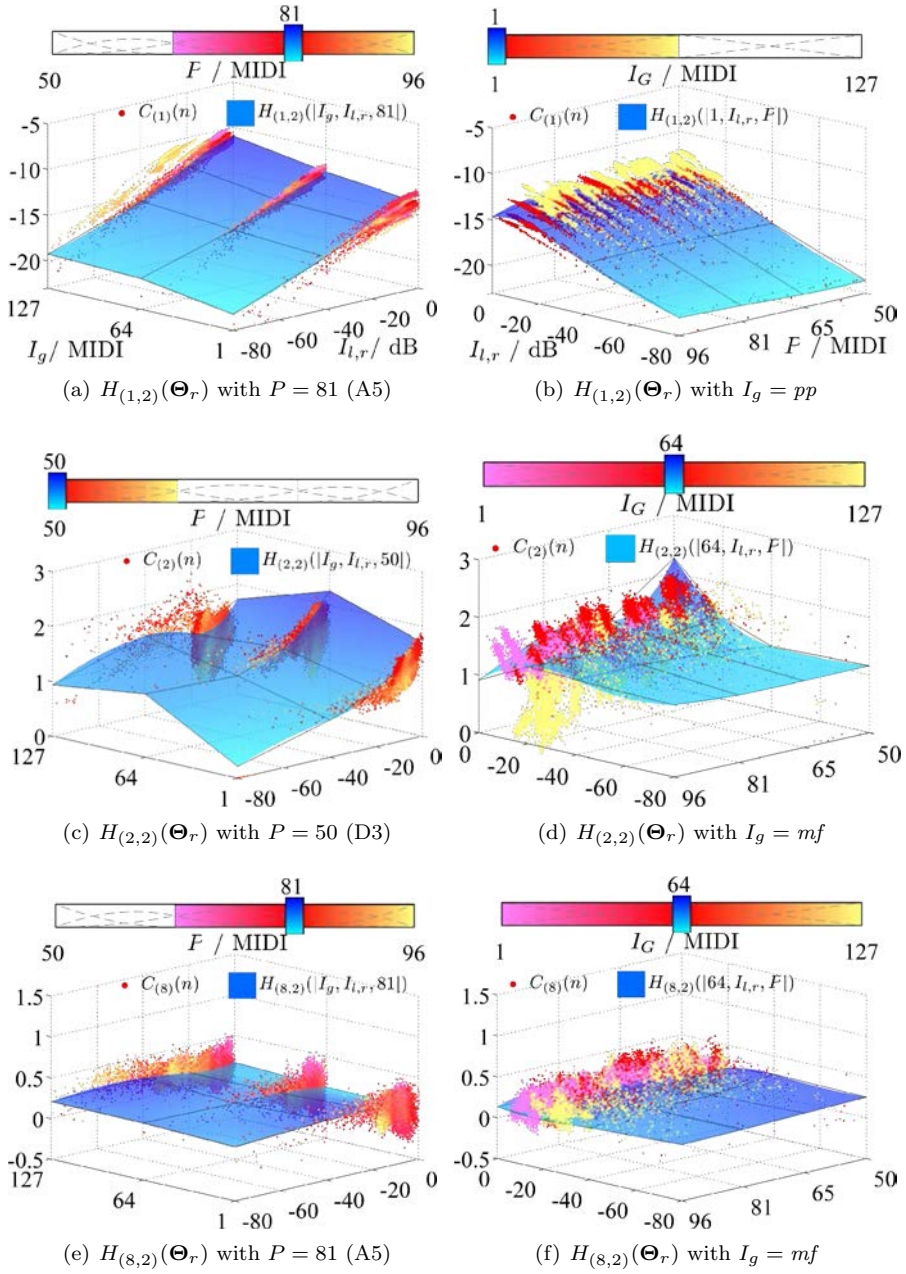
The cepstral coefficients represented as functions of the control variables do not necessarily resemble data trajectories similar to the partial amplitudes since all coefficients with an index $l > 1$ refer to modulation amounts of the spectral envelope rather than spectral sound features directly. The point clouds obtained from the cepstral data of the residual signals are hence assumed to exhibit raising and falling slopes along all axes.

Though, to keep the overall readability of the thesis only figures for the residual models of the clarinet and piano data set are presented since they refer to either one class of sound signal excitation and insights into the models of the trumpet and violin sets do not yield significant additional information.

9.2.1 Clarinet

The visualizations of the residual model of the clarinet sound data set is presented in fig. 9.9 showing their respective cepstral data and adapted surfaces for $l = 1, 2$ and 8 in its respective subfigures. The two top figures 9.9(a) and 9.9(b) present the data and model for the first cepstral coefficient representing the overall offset of the spectral envelope of the residual signal as a function of the control variables. Hence, a decreasing slope can be observed along the axis of its local intensity $I_{l,r}$ referring to a continuously decaying residual level with decreasing amplitude envelope.

The figures 9.9(c) to 9.9(f) depict model and data properties for the cepstral coefficients $l = 2$ and 8 which are therefore referring to modulation amounts of the spectral envelope and hence do not exhibit a clear tendency for a decaying trajectory with decreasing local intensity.



(a) $H_{(1,2)}(\boldsymbol{\Theta}_r)$ with $P = 81$ (A5)

(b) $H_{(1,2)}(\boldsymbol{\Theta}_r)$ with $I_g = pp$

(c) $H_{(2,2)}(\boldsymbol{\Theta}_r)$ with $P = 50$ (D3)

(d) $H_{(2,2)}(\boldsymbol{\Theta}_r)$ with $I_g = mf$

(e) $H_{(8,2)}(\boldsymbol{\Theta}_r)$ with $P = 81$ (A5)

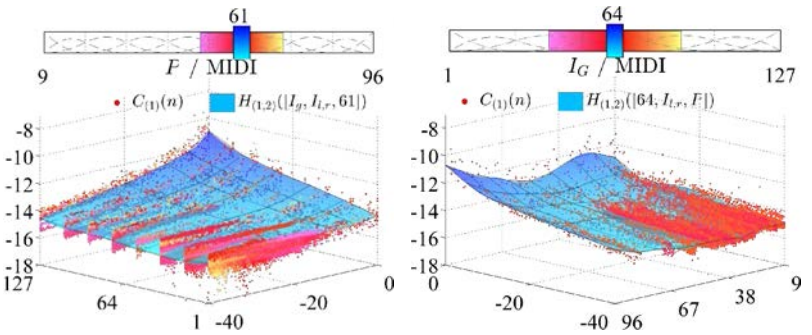
(f) $H_{(8,2)}(\boldsymbol{\Theta}_r)$ with $I_g = mf$

Figure 9.9: Selected visualizations for several tensor-product B-spline surfaces of the trained residual model of the clarinet data set.

It may however be observed, that the variance in the data is quite high and reasonable surface shapes are not always easy to derive.

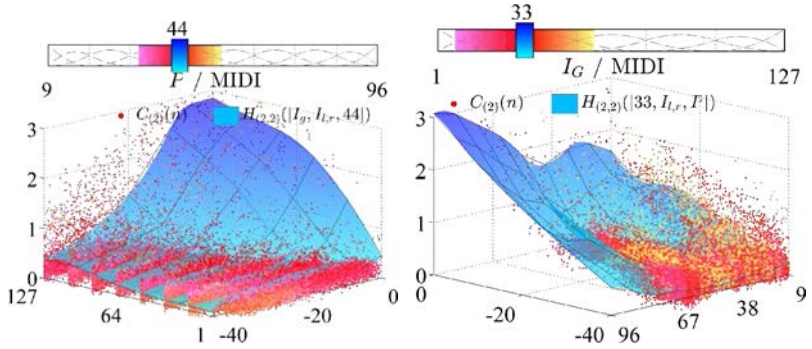
9.2.2 Piano

The residual model component for the Fazioli piano sound data set is also visualized for $l = 1, 2$ and 8 whereas always the second temporal segment $s = 2$ has been selected for display. As may be observed the distribution of the cepstral data of the residual signals of the piano exhibits strong variance, though the estimated surfaces represent this data follow their average as closely as possible regarding the selected knot grid.



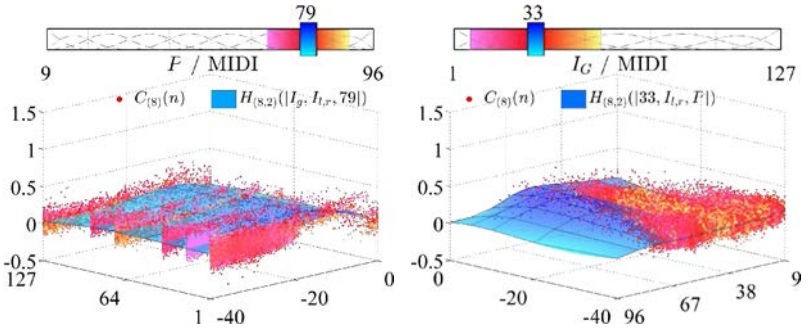
(a) $H_{(1,2)}(\boldsymbol{\Theta}_r)$ with $P = 61$ (C#4)

(b) $H_{(1,2)}(\boldsymbol{\Theta}_r)$ with $I_g = 64$



(c) $H_{(2,2)}(\boldsymbol{\Theta}_r)$ with $P = 44$ (G#2)

(d) $H_{(2,2)}(\boldsymbol{\Theta}_r)$ with $I_g = 33$



(e) $H_{(8,2)}(\boldsymbol{\Theta}_r)$ with $P = 79$ (G5)

(f) $H_{(8,2)}(\boldsymbol{\Theta}_r)$ with $I_g = 33$

Figure 9.10: Selected visualizations for several tensor-product B-spline surfaces of the trained residual model of the Fazioli piano data set.

Chapter 10

Model–Based Sound Synthesis

Introduction

Expressive sound synthesis using the presented extended-source-filter model and its respective control parameters needs to be done in parallel for the two signal components whereas synthesis of the harmonic component can be done in two ways: The component can be synthesized additively using the signals partial representation while performing all transformations directly in the sinusoidal domain or it can be carried out in a subtractive manner as shown in eq. (6.1) using the spectral representation of the whitened signal components $\bar{x}_h(t)$ and $\bar{x}_r(t)$. The residual component however can only be processed subtractively due the lack of a generative model for $x_r(t)$.

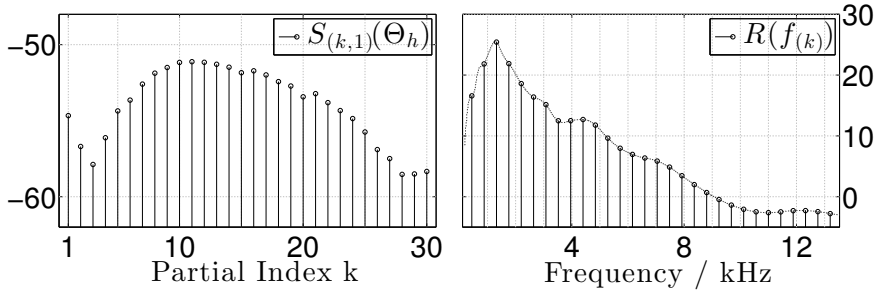
As already depicted in the schematic in fig. 6.3, we have chosen a subtractive method for our synthesis system for both signal components and we utilize a phase vocoder with support for implicit sinusoidal modeling as introduced in sec. 3.4 to process all signal transformations in the frequency domain. We hence still require a method to generate the time-varying filter envelopes $F_h\big(f, \mathbf{\Psi}_h(n)\big)$ and $F_r\big(f, \mathbf{\Psi}_r(n)\big)$ to create the whitened source signals $\bar{x}_h(t)$ and $\bar{x}_r(t)$ and their spectral representations to eventually perform sample-based sound synthesis using the control parameters P, I_g and $I_{l,\gamma}$.

In the following two sections we will therefore introduce the synthesis methods for both signal components separately and explain the respective filter envelope generation methods and whitening procedures as well as recapitulate the synthesis scheme for each component.

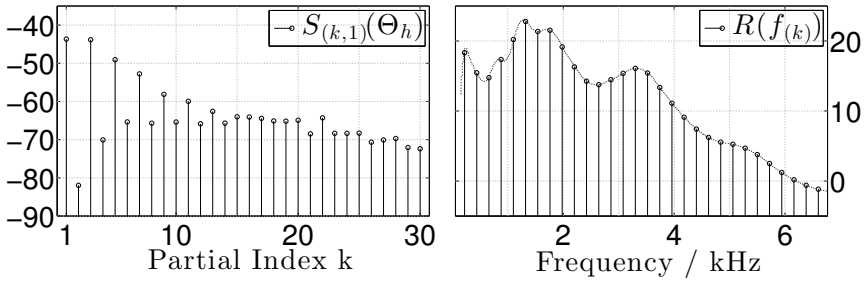
10.1 Subtractive Harmonic Synthesis

The application of subtractive synthesis essentially refers to filtering and as we aim for spectral domain processing an appropriate filter refers to a quasi-continuous function of frequency which gets multiplied with a spectral frame of a signal. The harmonic model as proposed in sec. 6.1 however estimates partial amplitude values only at their discrete frequency positions and for turning amplitudes values at discrete frequency positions into a continuous function we will apply an interpolation technique which takes the spectral leakage of the Fourier transform into account.

Though, prior to the interpolation method we first need to process all required estimates of the partial amplitudes using the harmonic model. Figures 10.1 and 10.2 depict the partial amplitude estimates separately for the excitation and resonance component of the harmonic model for the selected instruments indicating their independent contributions to the summed estimate.



(a) Trumpet, P = A4, I_g = *ff*, $I_{l,h}$ = 0dB



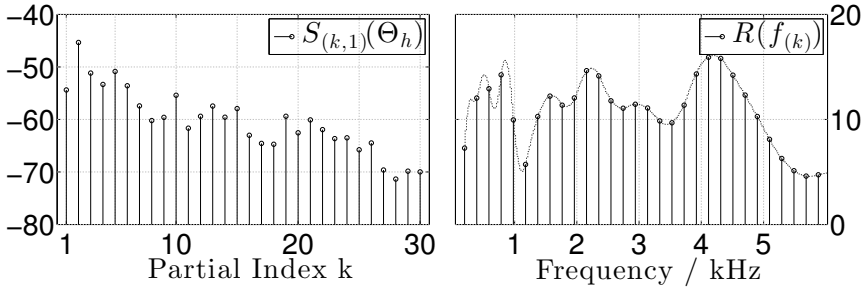
(b) B$\flat$-Clarinet, P = A3, I_g = *ff*, $I_{l,h}$ = 0dB

Figure 10.1: Trumpet and clarinet examples for the extended-source-filter model adapted to their respective sound data sets. The first 30 partial amplitude estimates for the $S_{(k,1)}$ and $R(k)$ component are displayed w.r.t. the specified control parameter values in dB values. The archetypical property of the clarinet exhibiting weak amplitudes at even harmonic indexes can be observed clearly.

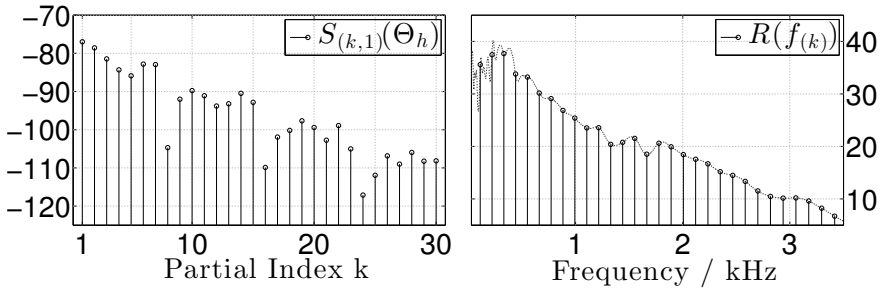
Eventually, the model's estimates $\hat{A}_{(k,s)}(\mathbf{\Psi}_h(n))$ shown by their individual source and resonance contributions in fig. 10.1 and 10.2 need to be transformed into continuous-valued functions of frequency. The filter generation method then will be used to create the white source signals as well as for sound synthesis.

10.1.1 Filter Envelope Generation

For the creation of a time-varying filter function to be used in the Phase Vocoder filtering method, we employ an interpolation technique of the partial amplitude estimates that alternates between constant and linear interpolation. The constant segments for the interpolation are centered around each frequency location with a certain bandwidth and linear interpolation in amplitude domain is used to concatenate these segments to create the continuous



(a) Violin, $\mathbf{\Theta}_h = (P = \text{G3}, I_g = ff, I_{l,h} = 0\text{dB})$



(b) Grand Piano, $P = \text{A1}$, $I_g = pp$, $I_{l,h} = 0\text{dB}$

Figure 10.2: Estimated partial amplitude values for the first 30 k of the models for the piano and violin data set. The figures show the estimates for the excitation source $S_{(k,1)}$ and resonance component $R(f_{(k)})$ separately regarding a certain set of control parameters. Several strong formant-like amplifications may be observed in the resonance component of the violin as well as distinct attenuations of partial amplitudes with index multiples of 8 within the piano excitation component which is due to their excitation position.

filter envelope. The constant bands of the filter envelope surrounding each partial are delimited by their according lower and upper limit using b_k^l and b_k^u respectively. Their exact values are determined such that the main lobe corresponding to that partial has dropped by 18dB from its peak. To ensure sufficient separation of the individual main lobes of all partials of a harmonic signal $X_h(f, n)$ the analysis window size needs of course to be adjusted with respect to the signals fundamental frequency as discussed in sec. 3.1.

The filter generation method $\mathfrak{F}_h(f)$ may hence be summarized as in the following equations using inequality (10.1) to emphasize the non-overlapping property of the constant segments and definition (10.2) to obtain the abbreviated linear partial amplitude per temporal segment $\hat{a}_{(k,s)}$ estimated using the harmonic component of the instrument model with respect to its control parameters.

$$b_{k-1}^u < b_k^l \quad \forall\, k \tag{10.1}$$

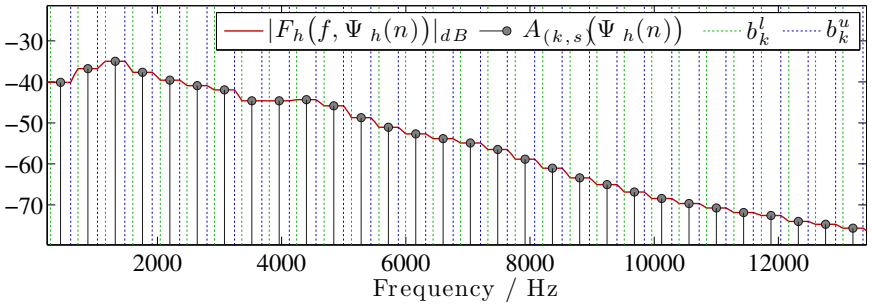
$$\hat{a}_{(k)} := 10^{\left(\frac{\hat{A}_{(k,s)}(\mathbf{\Psi}_h(n)}{20}\right)} \tag{10.2}$$

Using the constraint and definition above we may derive the equation (10.3) for obtaining the filter envelope $\mathfrak{F}_{h,s}\big(f, \mathbf{\Psi}_h(n)\big)$ to convert the partial amplitude estimates at discrete frequency locations to an envelope which can be evaluated at arbitrary frequencies.

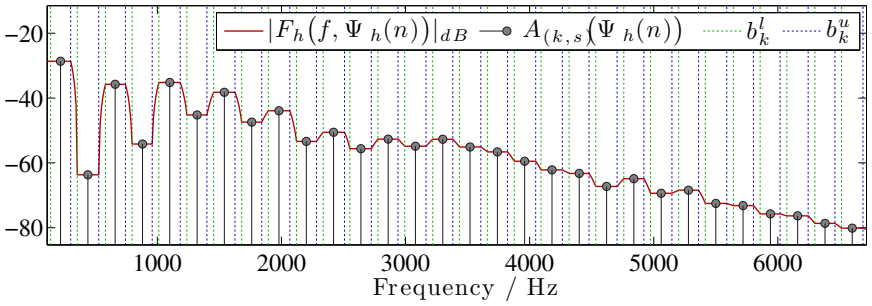
$$\mathfrak{F}_{h,s}\big(f, \mathbf{\Psi}_h(n)\big) = \begin{cases} \hat{a}_{(1)} & f \leq b_1^l \\ \hat{a}_{(k)} & b_k^l \leq f \leq b_k^u \\ \dfrac{\hat{a}_{(k)} - \hat{a}_{(k-1)}}{b_k^l - b_{k-1}^u}\big(f - b_{k-1}^u\big) + \hat{a}_{(k-1)} & b_{k-1}^u \leq f \leq b_k^l \\ \hat{a}_{(K)} & b_K^u \leq f \end{cases} \tag{10.3}$$

The first and last case of eq. (10.3) refer to boundary conditions at frequencies below the lowest of above the highest partial amplitude while case two defines the segment of constant interpolation and the third segment refers to the linear interpolation of neighboring segments of constant interpolation.

Examples for filter envelopes generated using eq. (10.3) using the exact partial amplitude estimates presented in fig. 10.1 and fig. 10.2 are shown in the figures 10.3 and 10.3. Though the filter envelopes are established in amplitude domain, the figures display them using decibel values for readability.

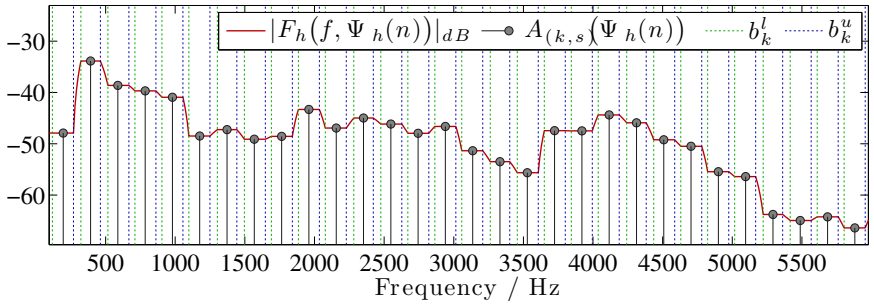


(a) Trumpet, P = A4, I_g = *ff*, $I_{l,h}$ = 0dB, s=1



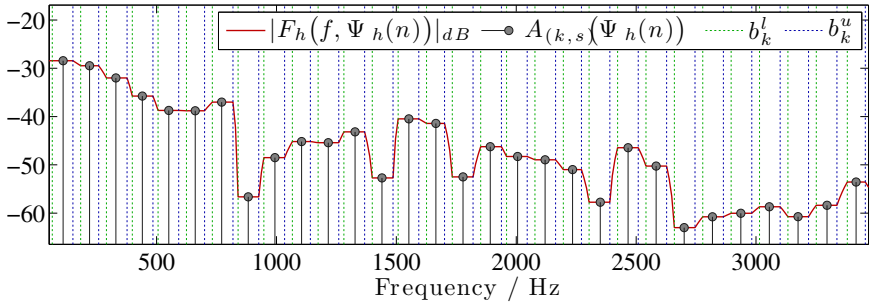
(b) B$\flat$-Clarinet, P = A3, I_g = *ff*, $I_{l,h}$ = 0dB, s=1

Figure 10.3: Generated filter envelopes $F_f\big(f, \mathbf{\Psi}_h(n)\big)$ using partial amplitude estimates $\hat{a}_{(k,s)}(\mathbf{\Psi}_h(n))$ of the trumpet and clarinet model with the alternating piecewise constant/linear interpolation method.



(a) Violin, $P = \text{G3}, I_g = ff, I_{l,h} = 0\text{dB}, s = 1$

Grand Piano, $P = \text{A1}$, $I_g = pp$, $I_{l,h} = 0\text{dB}$



(b) Fazioli Grand Piano, $P = \text{A1}, I_g = pp, I_{l,h} = 0\text{dB}, s = 1$

Figure 10.4: Generated filter envelopes $F_f\big(f, \mathbf{\Psi}_h(n)\big)$ using partial amplitude estimates $\hat{a}_{(k,s)}(\mathbf{\Psi}_h(n))$ of the trumpet and clarinet model with the alternating piecewise constant/linear interpolation method.

One may note the discontinuities of the first derivative of the filter envelopes at to the joints of the linearly and constantly interpolated segments. In our experiments these joints did not introduce any audible artifacts as the harmonic signal component $x_h(t)$ does not contain signal data apart from sinusoids whose frequencies are located at the center of the constant segments. We however also applied a cepstral smoothing method to obtain filter envelope whose first derivatives are smooth but due to their modulation property around the partials exact frequencies they introduced audible artifacts. We concluded the artifacts being caused by the non-constant attenuation of a partials main lobe which introduced a distortion of its bell-shape and therefore led to audible artifacts and we have hence not used cepstral smoothing for the work of this thesis.

For obtaining the final time-varying filter $F_h\big(f, \mathbf{\Psi}_h(n)\big)$ we need to account for the individual signal's attack-sustain and sustain-release phases and their discrete representations within the instrument model using either $s = 1$ or $s = 2$. Therefore, we utilize the fusion function $\varphi_{h,s}(n)$ introduced in sec. 5.2.3.2 to linearly cross-fade 2 discretely generated filter envelopes $\mathfrak{F}_h$ using eq. (10.3) for $s = 1$ and $s = 1$ respectively as shown in eq. (10.4).

$$F_h\big(f, \mathbf{\Psi}_h(n)\big) = \sum_{s=1}^{2} \varphi_{h,s}(n) \cdot \mathfrak{F}_{h,s}\big(f, \mathbf{\Psi}_h(n)\big) \tag{10.4}$$

The filter envelope generation technique as well as the linear cross-fade will eventually be used to create all time-varying filter functions of the harmonic signal component required by the whitening and all sound synthesis procedures.

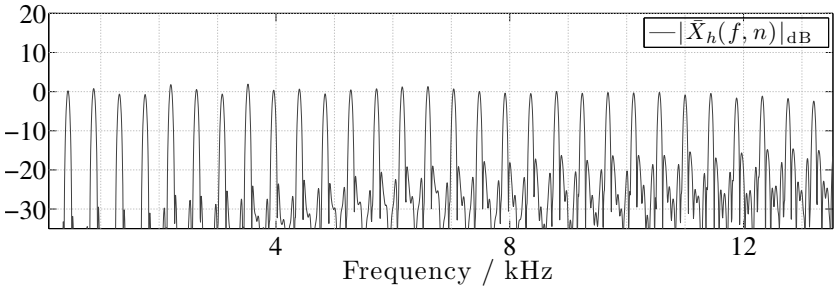
10.1.2 Harmonic Signal Whitening

A whitening procedure refers to the removal of the spectral envelope from a signal yielding a new signal that exhibits a uniform distribution of its spectral content hence the term white. Such a whitened signal may then be used as a source signal in a source-filter or extended source-filter model where an arbitrary spectral envelope may be applied to obtain a target signal as represented by eq. (6.1). In the approach presented within this thesis the spectral envelopes for whitening are obtained using the estimated instrument models and the according filter generation procedure.

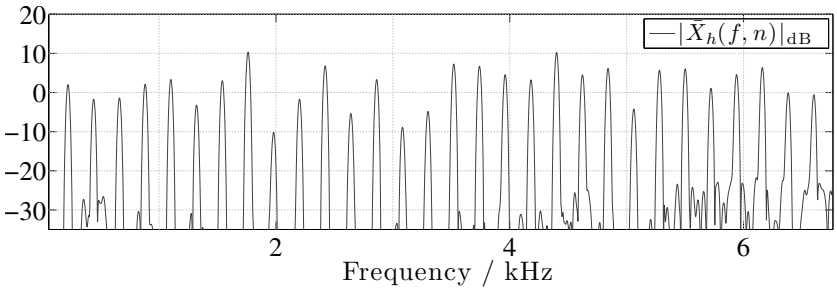
The whitening procedure to obtain the required source signal $\bar{x}_h(t)$ and its spectral representation for eq. (6.1) can be carried out by means of inverse filtering its respective signal component $x_h(t)$ as shown in eq. (10.5) using the estimated filter $F_h\big(f, \mathbf{\Psi}_h(n)\big)$ for its unaltered control variables $\mathbf{\Psi}_h(n)$.

$$\bar{X}_h(f, n) = X_h(f, n) \cdot F_h\big(f, \mathbf{\Psi}_h(n)\big)^{-1} \tag{10.5}$$

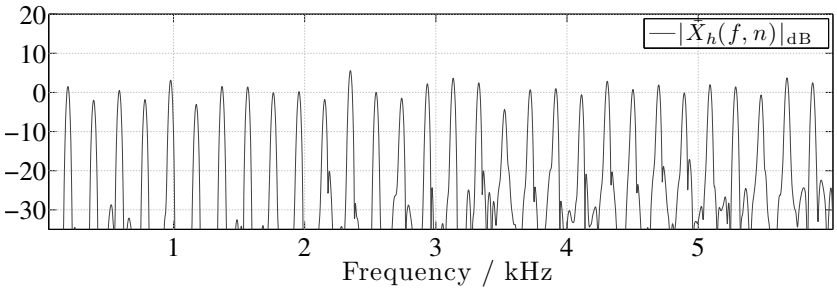
Several spectra of whitened harmonic signals are shown in fig. 10.5. It can be observed that all spectra do not entail a perfectly flat spectral shape which is due to the remaining model error which eventually retains as minor variations within the source signals used for sound synthesis thereafter.



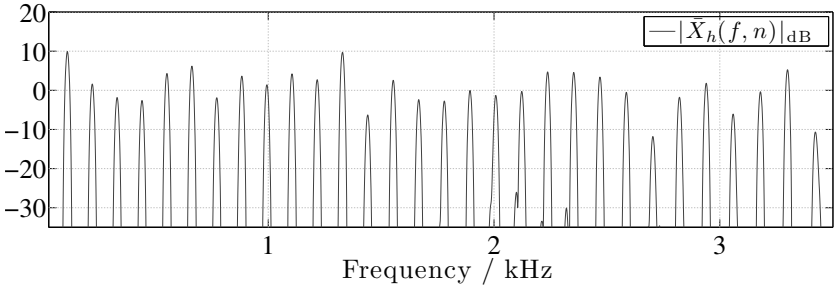
(a) Trumpet, P = A4, $I_g = ff$, $I_{l,h}$ = 0dB, s=1



(b) Bb-Clarinet, P = A3, $I_g = ff$, $I_{l,h}$ = 0dB, s=1



(c) Violin, $P = \mathrm{G3}, I_g = ff, I_{l,h} = 0\mathrm{dB}, s = 1$



(d) Fazioli Grand Piano, $P = \mathrm{A1}, I_g = pp, I_{l,h} = 0\mathrm{dB}, s = 1$

Figure 10.5: Flattened spectra of selected harmonic sounds of the trumpet, clarinet, violin and piano from their respective data sets obtained using the harmonic whitening procedure.

10.2 Subtractive Residual Synthesis

Synthesis of the residual component is setup similar to its harmonic counterpart as shown in the overall synthesis equation (6.1) using the spectral representation $\bar{X}_r(f,n)$ of a white source signal of the sound signals residual component and a belonging parametric filter with time-varying parameters $F_h\big(f, \mathbf{\Psi}_h(n)\big)$. As for the harmonic component, the residual component of the instrument model does not represent spectral envelopes and hence usable filter functions directly but cepstral coefficients. Obtaining white source signals as well as synthesis results henceforth requires a method to convert cepstral coefficients back to spectral envelopes.

10.2.1 Filter Envelope Generation

The Conversion of the estimated cepstral coefficients $\hat{C}_{(l,s)}(\mathbf{\Psi}_r)$ of the residual component of the instrument model to a spectral envelope is straightforward as shown in sec. 3.3.5, though as windowing in cepstral domain has already been applied and with using the parametric model of the cepstral coefficients, eq. (3.15) needs to be adapted appropriately to yield a spectral envelope suitable for filtering in the frequency domain:

$$\mathfrak{F}_{r,s}\big(f, \mathbf{\Psi}_r(n)\big) = \exp\left(\sum_{l=0}^{L} \hat{C}_{(l,s)}(\mathbf{\Psi}_r(n)) \cos(\pi l f/N)\right) \tag{10.6}$$

As for the harmonic component, linear cross-fades are applied to smoothly perform the transition between the estimates for the spectral envelopes of the attack-sustain and sustain-release segments using the respective fusion function $\varphi_{r,n}$. Eq. (10.7) shows the fade between the different filter envelopes using $\mathfrak{F}_r$ to refer to the method expressed in eq. (10.6).

$$F_r\big(f, \mathbf{\Psi}_r(n)\big) = \sum_{s=1}^{2} \varphi_{r,s}(n) \cdot \mathfrak{F}_{r,s}\big(f, \mathbf{\Psi}_r(n)\big) \tag{10.7}$$

In contrast to the generation of the filter envelopes for the harmonic filter envelope, envelopes generated using eq. (10.6) exhibit a continuous first derivative by definition of the windowed cepstrum used to represent the spectral envelope of the residual signal component. This does however not introduce any artifacts since we assume the absence of sinusoidal components in the residual signal whose main lobes could become distorted.

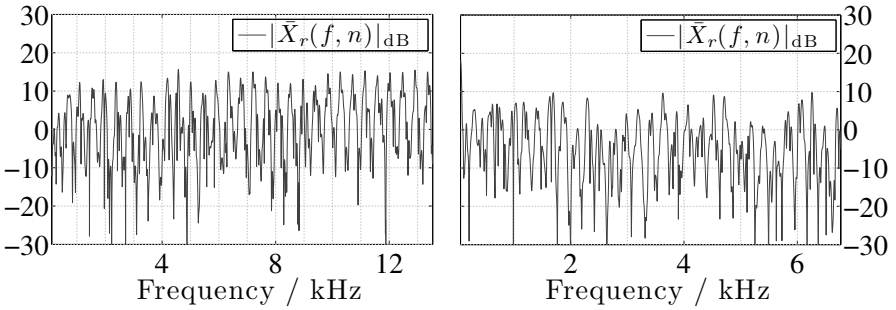
10.2.2 Residual Signal Whitening

Whitening of the residual component is eventually straightforward and done in an equal manner as for the harmonic signal component as show in (10.8). The residual component $x_r(t)$ of a signal gets filtered with the inverse of the filter which has been generated using its unaltered control variables to remove the spectral envelope which has been estimated by the instrument model according to its parameters.

$$\bar{X}_r(f,n) = X_r(f,n) \cdot F_r\big(f, f, \mathbf{\Psi}_r(n)\big)^{-1} \tag{10.8}$$

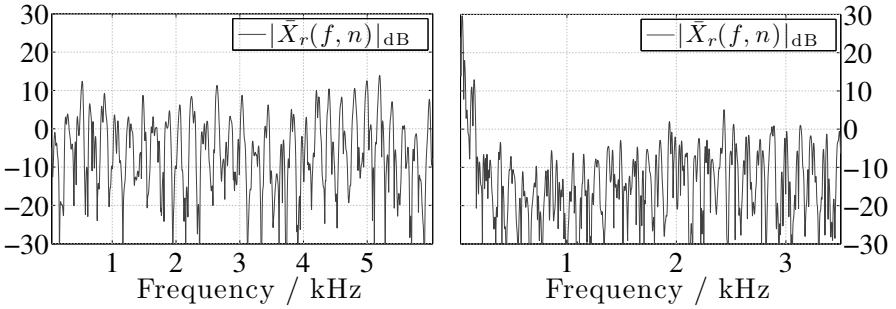
Whitening of the residual signal component may eventually yield a signal with almost white spectral distribution and one could argue that such a source signal could be generated using a white noise generator. This is however not the case for two reasons: First, the spectral envelope estimated by the residual component of the instrument model is not necessarily the same envelope of the residual signal due to possible limitations of the modeling capabilities of the internal representation. Therefore its removal may not yield a perfect uniform distribution in the whitened signal. Second, the assumption of a residual signal being only filtered white noise may be too approximate and as the authors of [CKD+13] have shown, substituting the residual signal by filtered white noise yielded perceptually recognizable differences.

Therefore, for our synthesis method we keep the whitened residual source signals and use them rather than artificial white noise signals for the synthesis method. Some example spectra are shown in fig. 10.6.



(a) Trumpet, P = A4, I_g = *ff*, $I_{l,r}$ = 0dB, s=1

(b) B$\flat$-Clarinet, P = A3, I_g = *ff*, $I_{l,r}$ = 0dB, s=1



(c) Violin, $P = \mathrm{G3}, I_g = \mathit{ff}, I_{l,r} = 0\mathrm{dB}, s = 1$

(d) Fazioli Grand Piano, P = A1, I_g = *pp*, $I_{l,r} = 0\mathrm{dB}, s = 1$

Figure 10.6: Flattened spectra of selected residual sounds of the trumpet, clarinet, violin and piano from their respective data sets obtained using the residual whitening procedure.

10.3 Dual Component Synthesis

Sound Synthesis is eventually done by altering the control variables $\mathbf{\Psi}$ to modify the resulting filter envelopes and hence create target sound signals using the source signals that correspond to the adjusted rather than their original variables. We may hence rewrite the synthesis eq. (6.1) to account for altering its parameters as follows:

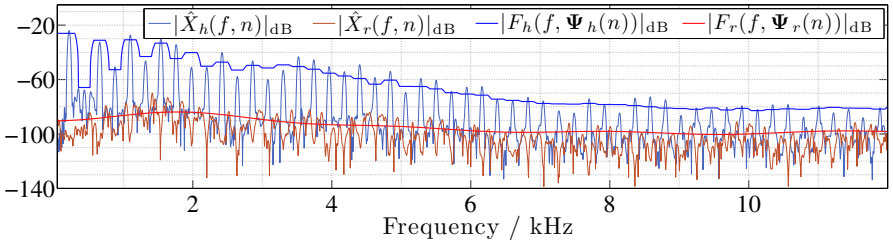
$$\begin{aligned} X((,f),n) &= \bar{X}_h(f,n) \cdot F_h\big(f, f, \mathbf{\Psi}'_h(n)\big) \\ &+ \bar{X}_r(f,n) \cdot F_r\big(f, f, \mathbf{\Psi}'_r(n)\big) \end{aligned} \tag{10.9}$$

In eq. (10.9) the variables $\mathbf{\Psi}'_h$ and $\mathbf{\Psi}'_r$ are used to refer to modified versions of the control variables belonging to the selected source signals $\bar{X}_h(f,n)$ and $\bar{X}_r(f,n)$. These modifications essentially refer to alterations of a signals global intensity I_g or its pitch P in the first place. Modifications of the local intensity $I_{l,\gamma}$ of either source signal is beyond the scope of this chapter.

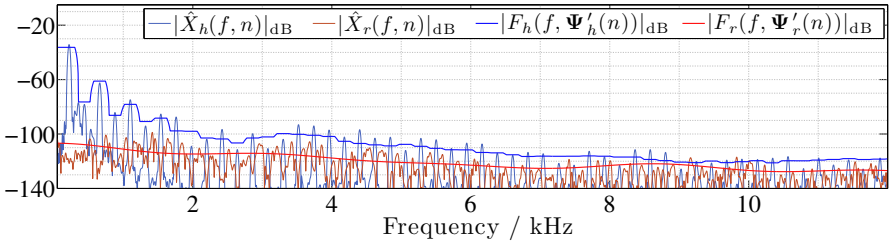
Though, as the instrument model components allow for amplitude envelope modifications only, modifications to the pitch control variable need to be accompanied by an additional transposition step. The frequency shifting procedure in our approach is applied prior to filtering of the source signal onto the harmonic signal only using its spectral representation. For transposition we apply the advanced Phase Vocoder method taking into account transient preservation as well as the partials vertical phase coherence as introduced in sec. 3.4. However, for piano sound signals vertical phase coherence can not be retained due to the inherent inharmonicity of its sound signals.

In fig. 10.7 three spectra are presented to illustrate some synthesis results for the clarinet data set. All three examples are synthesized using the same source signals which have been created from the clarinet sound signal which exhibits pitch $P = \text{A3}$ and a global intensity value of $I_g = \textit{ff}$. The top graph 10.7(a) shows a single spectrum from the resynthesis using unaltered control variables and hence represents the original signal without any modifications, though the signal has been filtered twice. The center graph 10.7(b) depicts the synthesis result for applying the filter envelopes obtained using a modified value of the signals global intensity. Both spectra for the harmonic as well as residual signal have changed significantly whereas the harmonic component now exhibits a much stronger narrow-band property and the residual component has been significantly reduced in terms of its energy. Both signal components exhibit a spectral shape following the estimated filter envelopes quite closely. This allows to assume that the obtained synthesis results match the features learned by the instrument model regarding the selected control parameters.

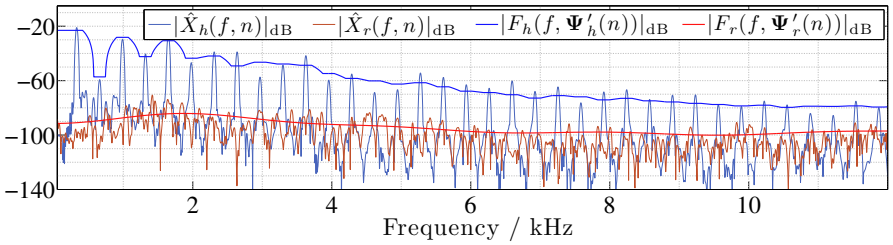
The result for a pitch transformation of 6 semitones is presented in 10.7(c) showing that the property of weak even partials is retained during the transposition procedure.



(a) Bb-Clarinet, P = A3, $I_g = \mathit{ff}$, $I_{l,h} \approx 0$dB



(b) Bb-Clarinet, P = A3, $I_g = \mathit{pp}$, $I_{l,h} \approx 0$dB



(c) Bb-Clarinet, P = E4, $I_g = \mathit{ff}$, $I_{l,h} \approx 0$dB

Figure 10.7: Synthesis results for the Bb-Clarinet based on the same source signals created from the sound with P = A3, $I_g = \mathit{ff}$ using unaltered control parameters in the top graph but altered global intensity in the center and transformed pitch in the lower graph.

Chapter 11

Sound Intensity Estimation

Introduction

So far the instrument model as well as the synthesis scheme do not account for a signal's actual note intensity level, though the instrument model learns the timbral differences for the various different note intensity values. This is due to the fact, that the sounds that were included in the sound data sets of the continuously driven instruments have all been normalized to a maximum RMS value and hence do not exhibit any level variations apart from side-effects which we neglect here. The instrument model therefore only covers relative timbral differences rather than RMS differences but for the target of imitating acoustic instrument using a digital sound synthesis approach it appears to be a necessary component to adjust the sound level according to actual level variations of the respective instrument when playing with varying global intensity values. This becomes even more evident for variations of the global intensity while a note is being played to obtain dynamic variations like crescendo or decrescendo.

The sounds of the piano data set though have not been normalized and exhibit their original level differences for the various values of global note intensity. No further analysis and processing for the piano sounds is hence required and the piano sound set will hence be neglected for the intensity modeling technique.

In digital sound synthesizers the peak sound level for different global intensity values needs to be adjusted manually in a dedicated instrument design process using either a linear or non-linear scaling [Dan06]. Especially for synthesizers that aim for imitating acoustic instruments, this is very likely to be not coherent with actual characteristics of an instrument. Therefore, we present a method to estimate RMS levels for varying values of global note intensity utilizing specific recordings entailing crescendo and decrescendo variations and an instrument model trained using recordings of the same instrument as introduced in the previous chapters.

However, this requires some preliminary assumptions about the sound signals with dynamic variations:

For all recordings with dynamic changes we assume them to start with the lowest possible note intensity and that they end with its highest or vice versa and any normalization applied to the recording does not effect the relative difference between the dynamic levels. We further assume that these recordings

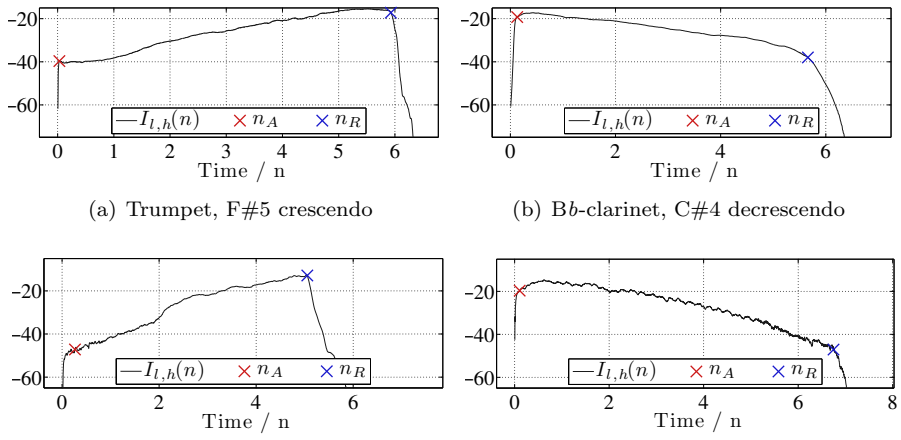
not only entail dynamic but also timbral variations, whereas these timbral changes are assumed to be related to the dynamic changes and hence derive that it is possible to establish a link between timbral variations and respective RMS values.

As the instrument model presented in this thesis represents the timbral properties of musical instruments as a function of the global intensity we develop a method that allows to establish the link between global intensity values represented as MIDI velocity to RMS signal values.

11.1 Analysis of Crescendo/Decrescendo Signals

To establish a link between timbral characteristics and RMS values we only use the harmonic component of signals with dynamic variations whereas the signal's attack and release phases will be neglected. We assume that the variations of the RMS value represented as local intensity throughout this thesis within that portion of such a signal refer to the variations of the played global intensity. The residual component is not considered for this procedure and therefore we omit the $_h$ identifier in all equations since it is implied in all cases.

The signal segment which is assumed to exhibit the RMS changes due to variations of global intensity value can however not easily be obtained with an automatic method. We therefore annotated the end of the attack and the begin of the release manually in the local intensity function $I_{l,}(n)$ of the harmonic component of these signals. Fig. 11.1 shows four examples of the local intensity $I_{l,}(n)$ for several instruments sounds with either a raising or decreasing intensity slope together with the hand-picked inflection points. Note that the functions have not been normalized to 0dB.



(a) Trumpet, F#5 crescendo

(b) B$\flat$-clarinet, C#4 decrescendo

(c) Violin, C#4 crescendo

(d) Violin, E5 decrescendo

Figure 11.1: 4 Examples for the evolution of the local intensity $I_{l,}(n)$ of 4 recordings playing with dynamic intensity changes.

The signal portion of the signal's local intensity bounded by the chosen delimiters n_A and n_R is hence specified as follows:

$$A_{(k)}(n) \quad , n_{\mathrm{A}} \leq n \leq n_{\mathrm{R}} \tag{11.1}$$

In a succeeding step the local intensity function $I_{l,}(n)$ gets sorted in an ascending order to obtain a function of monotonously increasing values. This unifies the remaining procedure as it can then be carried out equally for crescendo and decrescendo signals but also reduces complexity for the procedure of assigning global intensities to the signals RMS progression.

$$I_{l,}(n') = \mathrm{sort}(I_{l,}(n)) \tag{11.2}$$

The index n' is used to refer to the new ordering and we will further utilize that reordering for the partial amplitudes of the signal under consideration as well using $A_{(k)}(n')$ which now exhibits N' frames in an order that refers to a monotonously increasing local intensity value.

11.2 Generation of Prototypical Partial Envelopes

For an assignment of the values of the local intensity function to global intensity values of a recording of an instrument we calculate the partial amplitude values estimated by an accordingly trained instrument model using an artificial set of control parameters. A reasonable amount of global intensity values $numEnvs$ is chosen and a control variable vector $I_g(z)$ is created containing monotonously increasing values covering the whole range of possible intensity values. Since $I_g(z)$ is represented on the MIDI scale a stepwidth of 1 is used yielding an overall amount of $numEnvs = 127$:

$$I_g = [1, \ldots, 127] \in \mathbb{R}^{1 \times Z} \tag{11.3}$$

The pitch vector $P(Z)$ is set to the respective MIDI pitch of the recording for all z and the local intensity vector $I_{l,}(Z)$ is set to 0 following the assumption about the bounded local intensity of the signal referring to its sustain phase even though it exhibits changes of the RMS. The according control parameter matrix $\mathbf{\Theta}$ is hence expressed as in eq. (11.4). In the following we will use $\mathbf{\Theta}(z)$ to refer to a single column of $\mathbf{\Theta}$.

$$\mathbf{\Theta} = \begin{bmatrix} P(1) & , \ldots , & P(Z) \\ I_g(1) & , \ldots , & I_g(Z) \\ I_{l,}(1) & , \ldots , & I_{l,}(Z) \end{bmatrix} \in \mathbb{R}^{3 \times Z} \tag{11.4}$$

To eventually establish $\mathbf{\Psi}_h$ to generate partial amplitude estimates using eq. (6.4) we furthermore require an idealized partial frequency vector $\hat{f}_{(k)}$ and a fusion scheme. The generation of the partial frequency vector does not change in comparison to the classic analysis of instrument signals and is hence established as in sec. 5.2.4.

The fusion scheme is created using eq. (11.5) and eq. (11.6) as inflection points for the linear cross-fade between the partial amplitude estimates of the instrument model.

$$\xi_1 = \frac{1}{3} Z \tag{11.5}$$

$$\xi_2 = \frac{2}{3} Z \tag{11.6}$$

The fusion functions $\varphi_{,1}(z)$ and $\varphi_{,2}(z)$ are then created using the according formulations from sec. 5.2.3.2. The partial amplitude estimates may eventually be processed as in eq. (11.7) using eq. (6.4) to process the estimates of each temporal segment.

$$\hat{A}_{(k)}(z) = \sum_{s=1}^{2} \varphi_{,s}(z) \hat{A}_{(k,s)}(\boldsymbol{\Theta}(z)) \tag{11.7}$$

11.3 Intensity Assignment using Dynamic Programming

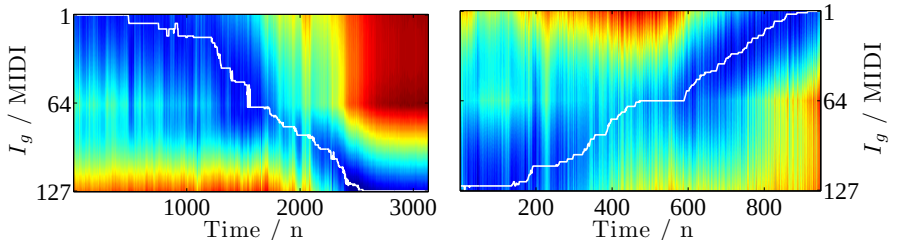
Using the partial amplitude estimates $\hat{A}_{(k)}(z)$ of the instrument model and the partial amplitudes of the signal $A_{(k)}(n')$ allows to create a partial amplitude cost matrix $\mathbf{C}_A \in \mathbb{R}^{Z \times N'}$ using the summed squared-error of the partial amplitudes for every frame n' and prototypical envelope z:

$$c_A(z, n') = \sum_{k}^{K} \left| A_{(k)}(n') - \left(\hat{A}_{(k)}(z) - \mu(n', z) \right) \right|^2 \tag{11.8}$$

Within the cost function (11.8) we use $\mu(n', z)$ to remove the mean difference between the two envelopes spanned by the discrete partial amplitudes to account for the overall level offset of the envelopes and to ensure that the cost function actually measures their relative difference rather than their offset. The difference between the two partial envelopes is hence processed as follows:

$$\mu(n', z) = \frac{1}{K} \sum_{k}^{K} \left(A_{(k)}(n') - \hat{A}_{(k)}(z) \right) \tag{11.9}$$

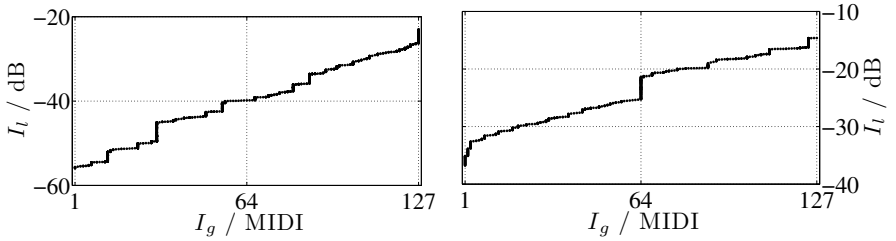
Using the dynamic programming approach from [SC78] allows to estimate an optimal path with minimum cost which yields an assignment of local intensity to global intensity values. Two examples for the paths finding algorithm within such a cost matrix are shown in fig. 11.3, though the columns have been reordered into their original sequence referred to by using n. Therefore, within fig. 11.2(b), the decrescendo appears as a path from a global intensity value 127 referring to *ff* to a value of 1 referring to *pp* and within fig. 11.2(a) an inverse path can be observed.



(a) Trumpet, A#3 crescendo. (b) B♭-Clarinet, A3 decrescendo.

Figure 11.2: Error Surface and optimal path from *pp* to *ff* for a trumpet crescendo (left) and a B♭-clarinet decrescendo signal (right).

The estimated path can now be used to associate every frame n of the signal with a value of the global intensity I_g and as such we may inversely assign the according local intensity value $I_{l,}(n)$ to its respective global intensity. The assignment hence yields a tuple $(I_{l,}|I_g)$ for every frame n'. Several local intensity values may hence be assigned onto one value of the global intensity and therefore the tuples appear as data clouds in fig. 11.3 for the two examples above. The applied ordering of the local intensity function in eq. 11.2 now effectuates the monotonous increase of the values of the tuples.



(a) Trumpet, A#3 crescendo.

(b) B$\flat$-Clarinet, A3 decrescendo.

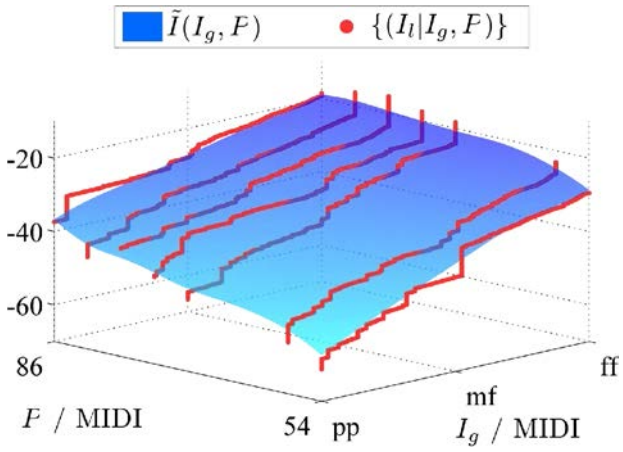
Figure 11.3: Tuples $(I_{l,}|I_g)$ of local intensity assignments for two selected recordings .

This procedure is being repeated for all recordings with dynamic variations i available within each dataset and we eventually transform the set of tuples of every single recording into triplets $(I_{l,}|I_g, P)$ by accounting for their respective pitch value as well.

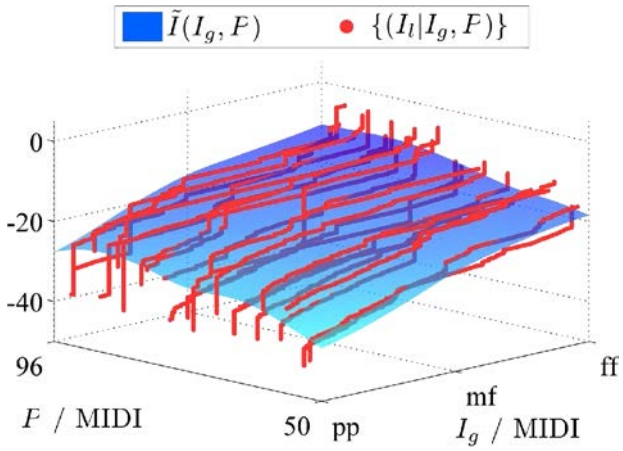
11.4 A Model for Global Note Intensity Levels

For every instrument such a set of data triplets will eventually be used to create a model of note intensity which is specific for the instrument and represents the estimated RMS values with respect to the timbral characteristics. The model is created as a surface in the space spanned by the pitch and global intensity variable using a Tensor-Product-B-Spline model as introduced in ch. 4 and shall be denoted $\tilde{I}(I_g, P)$. It represents intensity values in dB as a function of pitch and global intensity, both in MIDI scale. The model for the violin is created in a unified manner taking recordings from all strings into account to establish a single model.

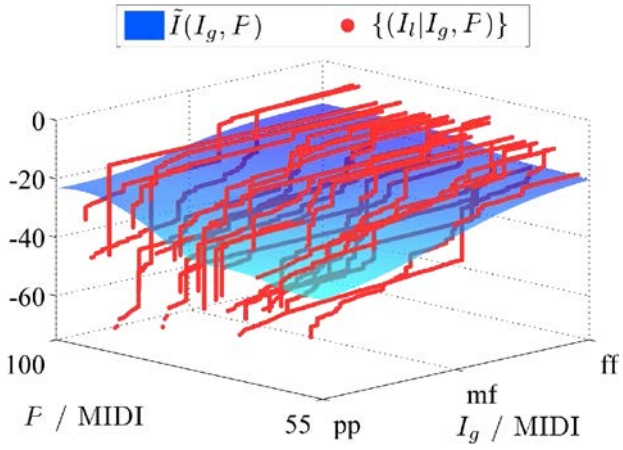
These models are then used to normalize all original input signals $x(t)$ of the continuously excited instrument sounds using the RMS values estimated by $\tilde{I}(I_g, P)$ and the complete procedure of estimating the deterministic and residual component, transforming the signals into a representation suitable for further processing and learning the parameters of the instrument models had been done again. The instrument model then not only estimate the timbral characteristics, but also their inherent sound level variations and therefore we call the obtained instrument models calibrated. With rerunning the analysis and instrument model parameter estimation methods anew, we circumvent the use of the model of the sound intensity level explicitly in the synthesis phase as we have shifted the additional processing into the offline analysis and modeling.



(a) Intensity model for Trumpet



(b) Intensity model for B$\flat$-Clarinet



(c) Intensity model for Violin

Figure 11.4: Three Models for the note intensity level as a function of pitch and global intensity.

Chapter 12

Subjective Evaluation

Intro

To finally assess the synthesis quality of the proposed synthesis system using the calibrated instrument model based on an extended source filter approach, we conducted a subjective evaluation to eventually determine to what extend instrument recordings may be transformed until audible artifacts or deviations from the listeners aural expectations are introduced.

The MUSHRA [itu03] as well as the ABX method [Cla82] have been developed to assess the subjective quality of audio systems in broadcasting environments and its design is thus meant for systems focusing on a perceptually perfect reconstruction of a reference signal rather than perceptual coherence according to the timbral variety of a musical instrument. Speaking in musical terms, a musician that is playing a note with equal pitch and volume twice does not necessarily create two indistinguishable sounds, but both will be perceived as correct realizations of a listeners expectation according to the timbre of the instrument. The MUSHRA as well as ABX method are hence not suitable for the subjective evaluation of a signal processors that aims for musical coherence rather than perfect reconstruction.

In [GSV11], Gabrielli et. al. also pointed out that ABX and MUSHRA are not viable for this purpose and hence developed an evaluation method and metric to assess the distinguishability between recordings of an acoustic or electric instrument and an emulation algorithm. Their method however also does not apply in the case of the expressive manipulation of the sound samples of the dataset as presented in this thesis.

The R-S method presented by Gabrielli [GSV11] et. al. targets for instrument emulation algorithms that do not exactly reproduce the timbral properties of the instrument used for the analysis and parameter estimation procuedure. This marks the main difference to the approach of this thesis where we are aiming for a perceptual coherent reproduction of the timbre and performance properties of a music instrument represented by a dataset of recordings. As Gabrielli et. al. also developed their metrics according to this fundamental assumption prohibits the use of their test mechanics as well as their measures. The R-S method by Gabrielli et. al. furthermore utilizes an instrument emulation algorithm that does not support the residual component of a sound signal which requires particular addition of some recorded background noise in the

R-S method.

12.1 Method

For our evaluation method to assess the perceptual coherence with an instrument's timbre variety, sound variations caused by slight modifications of the playing style need to be tolerated as long as a listeners expectation with respect to the sound of the overall instrument is fulfilled. We employ a similar terminology as for the MUSHRA or ABX procedures to refer to the recordings presented to the test subjects and hence all tests are constituted of 3 different kinds of sound signals:

- *Known Reference* : An unmodified sound signal those position in the presented sound samples is known to the test subject
- *Hidden Reference* : An unmodified sound signal those position in the presented sound samples is not known to the test subject
- *Probe* : The modified sound signal those position in the presented sound samples is not known to the test subject

In the case of the MUSHRA and ABX method, the known as well as the hidden reference signal refer to the exact same sound signal whereas the probe refers to a processed version of that signal and their perceptual similarity gets investigated. Such a setup is not suitable to assess a synthesis method that aims for actual sound transformations. For an expressive sound synthesis method, we therefore aim for an investigation that tests the coherence of the hidden reference and the probe with the overall timbre of an instrument allowing subtle variations and hence only requiring the control parameters to be equal.

Therefore within all tests, the probe will be generated using a source signal different from the hidden reference but transformed in such a way that it exhibits the same pitch and global intensity value. This shall enable a comparison of a listeners aural expectation for that specific set of control parameters regarding the instrument.

In the optimal case, the known reference should also exhibit the same pitch and global intensity as the hidden reference and the probe, though this requires the sound data sets to entail redundant recordings to prevent the use of identical signals for the hidden and known reference which would make the probe easily identifiable. The sound data sets used within the work of this thesis however do not contain multiple recordings for equal pitch and global intensity values and hence a recording with either a different pitch or global intensity will have to be picked for the known reference.

In the evaluation conducted for this thesis we have chosen the known reference to always exhibit an equal global intensity as the hidden reference and probe and only differ in pitch. We have decided that way as we assume the task of perceptual extrapolation along varying pitches to be an easier task for a test

subject as of varying global intensity. The interval for the pitch difference has chosen to be a major third for all tests of all instruments which is further assumed to be a familiar interval to western listeners that exhibit similar timbres for both pitches [HE01].

To further assist a listeners perception of an expected instrument timbre, the known reference will always be played before the hidden reference and the probe and therefore the listener will always make a judgement between two samples each containing two notes differing by a major third. This requires the test subject to perceptually extrapolate the sound of the known reference to a major third above when listening to a single sample.

The piano however is assumed to exhibit more drastic changes of its timbre for varying pitches due to its inherent inharmonicity among other reasons. We will hence provide the test subject an additional known reference played after the probe or hidden reference. This second known reference is always chosen to be a minor third above the hidden reference or probe hence creating a major chord for the whole sample. Using two known references is meant to support the test subject by transforming the task of extrapolation to a task of aural interpolation assuming it to be easier to accomplish.

For a single transformed sound, every test subject will hence be presented two sound samples each containing either two or three notes played consecutively. The subjects will then prompted to decide in which sound sample they perceived the second note played to be synthetic sounding or modified in any way and hence assess the interval or chord as not being natural sounding for the according instrument.

For the statistical evaluation we employ a test of statistical significance [BS10], for which we postulate the null hypothesis for each sound transformation in the way, that the listeners are not able to distinguish the hidden reference and the probe reliably, whereas the directed alternative hypothesis states, that the transformed sound is perceived as being artificial and not natural to the respective musical instrument.

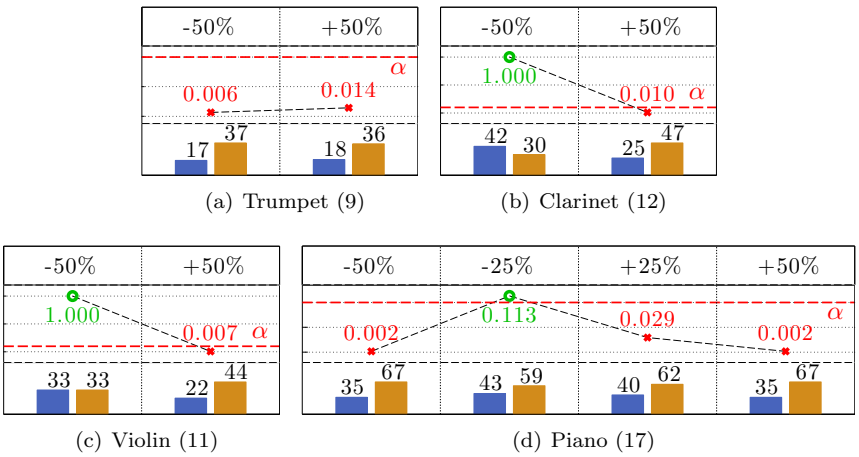
We evaluate modifications of the control parameters I_g and P independently to assess the transformation capabilities of our synthesis method separately for the two control parameters and created sets of equal modifications extends of either parameter to simplify and generalize the evaluation procedure. We hence specified several modification amounts for each transformation and created 6 sound pairs for each amount to approximately cover the instruments pitch and intensity range for each extend. The results for each extend of modification has been summed to obtain the observations for a χ^2 Goodness–of–fit test which yields a p-value to allow a decision in accordance to a specified level of significance α [BS10].

12.2 Results

The evaluation procedure had been setup as an online survey and the test subjects had been invited to participate voluntarily without compensation. The subjects have been asked to perform the test in a quiet environement using headphones and the exact amount of participants is given in brackets for each instrument in the respective captions of fig. 12.1 and fig. 12.2.

Fig. 12.1 depicts the results for the subjective evaluation of transformations of the signals global intensity values of the selected instrument sound data sets. The top row indicates the amount of change with respect to the instruments full range representing 100% is presented. For the trumpet, clarinet and violin, shown in fig. 12.1(a), 12.1(b) and fig. 12.1(c) respectively only half range changes have been made reflecting a transformation of I_g from either *ff* to *mf* or *mf* to *pp* in case of -50% or vice versa for +50%. The limited amount of discrete global intensity steps prevents a more granular analysis of global intensity steps as only three levels of global intensity are available for the data sets of continuously excited instrument signals. For the piano sound set up to 8 levels are contained and hence 4 discrete transformation steps have been selected as shown in the top graph of sub fig. 12.1(d).

The summed observations from all subjects are given as bar plots at the bottom for each instrument showing the distribution of the listeners selections. The blue bar refers to the amount of selections of the hidden reference for being the less natural sounding example, while the yellow bar refers to selections of our transformed sound for being artificial and incoherent with an expected timbre. The χ^2 test statistics had been used to calculate the p-value shown in the center graph, whereas different colors are used to indicate if its value is above (green) or below (red) the selected level of significance which has been set to the standard value of $\alpha = .05$.



(a) Trumpet (9) (b) Clarinet (12)

(c) Violin (11) (d) Piano (17)

Figure 12.1: Subjective evaluation results for the transformation of the global intensity value of the sound signals of the selected instrument sound data sets. The amount of test subjects for each instruments is given in brackets in the according subfigure caption.

As can be seen in fig. 12.1(b) and 12.1(c) an intensity reduction of half the possible dynamic range of either the clarinet or violin can be made while retaining the instruments sound characteristics, since the listeners were not able to identify the transformed sound above random probability. Contrarily, an increase of 50% dynamic range can be recognized much more likely by the listeners. We identified this as being caused by the limited amount of partials present within the signal of lower intensity in comparison to the upper.

Hence, even though the modification for the limited set of partials may yield an appropriate result, its narrower bandwidth makes the final transformation result distinguishable from an expected instrument sound.

The same reason seems to account for the results of the trumpet data set. There not only the increase of the signals global intensities has been identified by the test subjects as less natural sounding but also its attenuation. We assume this as being caused by a not sufficient attenuation of the trumpets upper partials while decreasing its global intensity values which keeps them audible even though their amplitudes are at about the same level as the residual signal.

The evaluation results for intensity transformation for the grand piano in fig. 12.1(d) shows a less optimal performance. Apart from the well working intensity attenuation of about -25% all the other modifications had been identified by the test subjects. An analysis of the spectral components of piano signals shows a particular difference to other instruments which consists of sympathetic resonances. These are not treated by either the signal or the instrument model and hence are not modified accordingly by the signal transformation method. However, being able to perceptually reliable modify the global intensity of up to -25% enables a possible synthesizer to rely on 4 discrete intensity recordings, while retaining full expressivity throughout synthesis.

Transposition results, thus modifications of a signals pitch are shown in fig. 12.2 for all instruments, whereas in the top row of each subfigure, the amount of transposition is given in semitones. Again, the results notably differ for impulsively and continuously driven instruments. For the trumpet, clarinet and violin sound data set, transpositions of up to a full octave yield results indistinguishable from their original equivalent, whereas for the piano this holds until a pitch shift of +5 semitones, which we identified to be caused by its varying inharmonicity and untreated sympathetic resonances. We did not conduct an evaluation on negative pitch shift, since we expect this to suffer from the same issue, which arises when increasing the global intensity.

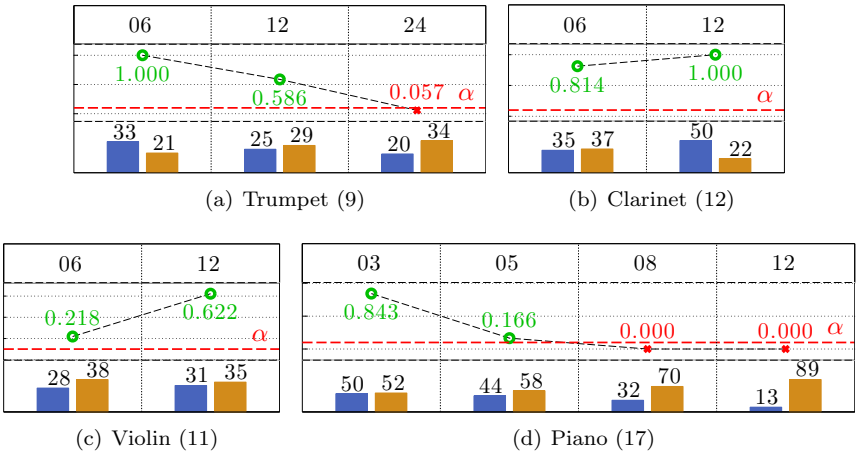
Conclusion

We may derive from the evaluation results that the calibrated instrument model allows to largely modify an instrument's sound signal coherently with the overall timbre of that instrument while retaining the quality of the recordings.

Pitch transpositions of up to a full octave can be achieved which are indistinguishable from their unmodified equivalents for the three presented continuously excited instruments and even for the piano, pitch alterations of almost half an octave are possible.

Modifications of the signals global intensity can also be achieved in a manner coherent with the overall instrument timbre for attenuations of either 25 or 50% regarding the instrument's complete dynamic range.

We may however also identify some limitations of the approach: The method is missing a strategy for adding partials not present in the harmonic source signal as well as for removing partials present in the source signal but not desired within the target sound. Eventually for piano sound signals two further limitations became apparent referring either to the limitations of the signal model



(a) Trumpet (9) (b) Clarinet (12)

(c) Violin (11) (d) Piano (17)

Figure 12.2: Subjective evaluation results for pitch transformations of the sound signals of the selected instrument sound data sets. The amount of test subjects for each instruments is given in brackets in the according subfigure caption.

not supporting sympathetic sinusoidal signal components as well as the missing modifications of the signal's inherent inharmonicity during the transposition.

A more detailed conclusion of the subjective evaluation will be discussed in the overall discussion of the thesis.

This page is intentionally left blank.

Part III

Conclusion

Chapter 13

General Conclusions

In this thesis, a parametric model suitable for the representation of the sounds of quasi-harmonic instruments has been presented that can be used to modify recordings of a particular instrument in a manner such that the synthesis result is coherent with the spectral properties of the respective acoustic instrument. The model uses general high-level control parameters which are available for most quasi-harmonic instruments to perform signal modifications which refer to standard control variables of sound synthesizers aiming for the imitation of acoustic instruments in a sample-based approach.

The instrument model developed within this thesis represents the deterministic and residual components of the sounds of an instrument separately using individual filter functions for each with suitable signal representation respectively. The distinct signal representation have been chosen with respect to the signal's properties and a unified approach for their internal representation for simplified modeling and parameter estimation has been presented.

A general and universally applicable arbitrary-order regression model using B-splines had been introduced for the purpose of fitting of multi-dimensional surfaces. The model utilizes Tensor-Product-B-splines to represent the multivariate data and a generic regularization scheme has been introduced to manually control the smoothness of the fit. A preconditioning method has further also been introduced for a significant acceleration of the iterative parameter learning procedure.

For the harmonic component of the instrument model an extended source-filter model has been employed referring to an individual excitation source and a resonance filter function. It has been shown, that both filter functions can be estimated jointly from an instrument's sound database using a dedicated parameterization of the filter functions. In the visualization of the estimated filter functions of the extended source-filter model several distinctive instrument features could be observed which lets us assume the successful separation of their individual contributions to the analyzed sound signals.

A new technique for the estimation of relative level differences of global note intensity values had been presented using dedicated recordings of crescendo and decrescendo instrument sound signals for continuously excited instruments. The estimated level values had further been used to calibrate the instrument model to eventually obtain models that account for spectral variations as well as level adaptations.

A subjective evaluation has been conducted to asses the synthesis results obtained by applying estimated filter functions generated using the calibrated instrument model. Within the subjective evaluation we were able to identify a range of possible parameter transformations which yielded results indistinguishable from their untransformed counterparts.

Pitch transpositions of up to an octave have shown to be possible using the proposed instrument model while retaining the coherence with the spectral properties of the respective instrument. To the authors knowledge, no other method so far has proven such a result for a similar transposition range. Changes to the note intensity have also been applied successfully with intensity attenuations of about a quarter to a half of the whole instrument's dynamic range without yielding audible differences to original recordings. Such a result has also not been proven by another method in the literature to our knowledge.

However, several insufficiencies and imperfections as well as aspects that require refinement and further improvement became apparent while conducting the research for this thesis which will be summarized in the next chapter.

Chapter 14

Future Work

This chapter lists some key points for revision and future development to further enhance the presented approach for expressive transformations of instrument recordings.

14.1 Sound Signal Transitions and Modulations

A topic of gaining interest in the domain of imitative sound synthesis of acoustic instruments is the reproduction of note transitions and modulations. Such transitions may refer to pitch glides as well as to vibrato or tremolo playing styles and such variations of the way acoustics instruments are being played may entail sound properties that are not covered by an instrument model trained using quasi-stationary instrument recordings.

Future version of an instrument model for imitative sound synthesis should hence also incorporate features of instrument sounds that refer to changes of expressive control parameters by taking their first or second order derivatives into account, though this would require additional recordings of such an instrument with respective parameter changes.

In the original proposition of the Sample Orchestrator 2 project such parameter transitions have already been considered, though have been rejected due to time and effort constraints. Future enhancements of the instrument model should certainly consider the incorporation of such parameter transitions as such are assumed to have a huge impact on the perceived naturalness of a digital sound synthesis method.

14.2 Signal Model Enhancements

The signal model being used separates an instrument sound signal into a sinusoidal and residual component. It has however been shown that the transient of the signals might be better represented using a dedicated model that emphasizes the temporal structure rather then its spectral envelope [Lev98, Ver99, Tho05]. This becomes quite apparent for impulsively excited signals as for piano sounds, whose residual component in the current setup mainly contains the signal being produced by the striking hammer. A parametric transient signal model with a more compact representation of the temporal shape of the transient seems worth considering and the incorporation into the instrument model

could be achieved by a third component using the same multivariate regression model for its representation and the selected expressive control parameters.

14.3 Expressive Control Enhancements

The introduced instrument model supports two parameters for expressive control that are assumed to be provided by some other source. This might either be a performer playing with some MIDI equipment or some sequencing device sending MIDI control parameters automatically.

Dedicated control models for fully expressive performances could though enhance the performative capabilities and expressiveness of the approach as they could automatically generate complex sequences of control parameters to transform and modify sound sources. The authors in [MBB+10] have created a control model for violin bowing gestures which may also be applied to other gestural music instrument interactions and could hence provide an expressive control layer for the presented instrument model.

14.4 Regression Model Enhancements

The multivariate regression model using Tensor-Product-B-splines is a very convenient methodology once all its components are in place. It is however by far not the only option to represent the multivariate data in a surface-fitting approach.

Within the current approach of using equally designed Tensor-Product-B-splines for all partial indices we encountered serious overfitting issues due to the resulting large regions of sparse data for high partial indexes. A better localization of the data could be achieved using different knot sequences and unequal control parameter boundaries for the various filter functions of the source excitation of the harmonic model component. For example, it does not seem to be an appropriate solution to represent a partial trajectory for all values of the control parameter space if the partial had only been present at very low pitches and high global intensity values.

Using models that are locally limited to the actual distribution of the data could hence significantly reduce the amount of parameters needed for the whole model while the local representation could even become more complex simultaneously without introducing more parameters for all partial indexes.

As discussed in sec. 4.3 the rectilinear knot grid may become inappropriate once instrument sound signals exhibit a less directional orientation in the parameter space. This may occur if recordings are used for the parameter estimation method that contain pitch or global intensity variations. More suitable models that also use linear superposition of basis function may be multivariate B-splines [Dah80, Hoe82, DM83] but also T-Splines [Sed07, Sed14] or recent advances in Truncated Hierarchical B-splines [GJS12, KGJ12] would allow for non-rectilinear knot grids.

There are also other methods that are not based on basis functions that allow for surface-fitting in arbitrary dimensions. Self-Organizing Maps [Hay09] or Support Vector Regression [SS02] could be applied, though these methods come with their own learning methods and regularization schemes and their

incorporation appears to be quite difficult even though there are various free and open source implementations available.

Last but not least, recent advances in Deep Learning architectures [Ben09, DD14] could enable the estimation of structural patterns rather than estimating the parameters of a predefined structure determined by the extended source-filter model. This could open a lot of possibilities as a model based on deep learning machines could automatically adapt itself to the physical structure of the instrument and learn their individual contributions.

14.5 Adaptive Model Selection

The current model selection strategy for choosing an appropriate configuration of the knot sequences and B-spline orders does not incorporate any automatic methods but solely manually adjustments. This might be improved by using Multivariate Adaptive Regression Splines [Fri91]. This method allows to begin parameter estimation with the simplest knot sequence and gradually improves the model by incrementally increasing the model complexity with respect to the local variance of the data regarding the model. We may assume that this method could be adopted to the presented instrument model to achieve significant improvements in terms of modeling accuracy as well as delimiting the amount of parameters required for the model to be sufficient.

14.6 Data Selection Improvements

Currently the amount of training data for each instrument model is tremendously large and even though nowadays computational capacity allows to work with such large amounts of data it seems plausible to introduce a smart data selection strategy to further accelerate the learning procedure. Such a data selection procedure should incorporate knowledge about the distribution of the data with respect to the control parameters. This could be done by using the inverse of probability density function of the distribution of the data regarding their parameters to increase the likelihood for choosing data from sparse regions and decrease for dense regions.

14.7 Subjective Evaluation Improvements

The task of conducting subjective evaluations of the coherence of the sounds of digital imitative synthesis methods with their acoustic equivalents has got only little attention in the past. The proposed method can hence only be regarded a starting point for the development of a general purpose subjective assessment method to evaluate such sound synthesis approaches. We hope to have raised interest into this general topic with the presented approach and expect advances from other researchers and graduate students as more and better imitative sound synthesis methods will be published in the future.

14.8 Improvements to the Global Intensity Model

The estimation of the level differences of instrument recordings with varying global note intensity values is certainly an important topic for designers and

developers of various kinds of digital sound synthesizers as it allows to apply attenuations and amplifications of the sound level that are coherent with actual instrument characteristics. As has been shown in the presented figures 11.4 , these level variations are by far not linear and also exhibit different behaviors for different pitches.

However, it can be derived from the figures 11.4 that the current approach tends to be more flat than i may be justified by the data and hence the level differences may tend to be too equalized. An improved method could for example incorporate the Kullback-Leibler divergence as a spectral distance metric as it has been shown to yield better performance for spectral distance measurements in source-separation tasks in the non-negative-matrix factorization framework.

Chapter 15

Final Remarks

Imitative digital sound synthesis remains an intriguing topic of research as it combines several academic disciplines including physics, music cognition, musicology, digital signal processing, computer science as well as machine learning and statistics. We believe this thesis represents a step forward in the progression of sample-based sound synthesis methods making them perceptually closer to their acoustic pendants.

We furthermore hope that topics covered within this thesis may reach out into other research directions within the audio domain and possibly beyond and that this thesis may help others developing better tools for people to make and enjoy music in the future.

Henrik Hahn, Paris/Berlin, 07.08.2015

This page is intentionally left blank.

Bibliography

[ABLS11] Xavier Amatriain, Jordi Bonada, Alex Loscos, and Xavier Serra. Spectral processing. In Udo Zölzer, editor, *Digital Audio Effects (eds. U. Zölzer)*, chapter 10, pages 293 – 446. John Wiley & Sons, 2 edition, 2011.

[Adr88] Jean-Marie Adrien. *Étude de Structures Complexes Vibrantes, Applications à la Synthèse par Modèles Physiques*. PhD thesis, Universit'e Paris VI, Pierre et Marie Curie (UPMC), 1988.

[Adr91] Jean-Marie Adrien. The Missing Link: Modal Synthesis. In Giovanni De Poli, Aldo Piccialli, and Curtis Roads, editors, *Representations of Musical Signals*, pages 269 – 298. MIT Press, Cambridge, 1991.

[AE03] Marios Athineos and Daniel P. W. Ellis. Frequency–Domain Linear Prediction for Temporal Features. In *IEEE Workshop on Automatic Speech Recognition and Understanding, ASRU´03*, pages 261–266, 2003.

[AF95] François Auger and Patrick Flandrin. Improving the Readability of Time–Frequency and Time–Scale Representations by the Reassignment Method. *IEEE Transactions on Signal Processing*, 43(5):1068 – 1089, May 1995.

[AKZV11] Daniel Arfib, Florian Keiler, Udo Zölzer, and Vincent Verfaille. Source-filter processing. In Udo Zölzer, editor, *Digital Audio Effects (eds. U. Zölzer)*, chapter 8, pages 279 – 320. John Wiley & Sons, 2 edition, 2011.

[All77] Jont B. Allen. Short Term Spectral Analysis, Synthesis and Modification by Discrete Fourier Transform. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-25(3):235 – 238, June 1977.

[AR77] Jont B. Allen and Lawrence R. Rabiner. A Unified Approach to Short–Time Fourier Analysis and Synthesis. *Proceedings of the IEEE*, 65(11):1558 – 1564, November 1977.

[AS04] Mototsugu Abe and Julius O. Smith. CQIFFT: Correcting Bias in a Sinusoidal Parameter Estimator based on Quadratic Interpolation of FFT Magnitude Peaks. Technical Report STAN-M-117, Center for Computer Research in Music and

Acoustics (CCRMA), Department of Music, Stanford University, Stanford, California 94305 USA, 2004.

[AT82] Luis B. Almeida and José M. Tribolet. Harmonic Coding: A Low Bit–Rate, Good–Quality Speech Coding Technique. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 1664 – 1667, 1982.

[AT83] Luis B. Almeida and José M. Tribolet. Nonstationary Spectral Modeling of Voiced Speech. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-31(3):664 – 678, June 1983.

[Bar13] Daniele Barchiesi. *Sparse Approximations and Dictionary Learning with Applications to Audio Signals*. PhD thesis, Queen Mary, University of London, 2013.

[BBC+94] R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. Van der Vorst. *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods, 2nd Edition*. SIAM, Philadelphia, PA, 1994.

[BDA+05] Juan Pablo Bello, Laurent Daudet, Samer Abdallah, Chris Duxbury, Mike Davies, and Mark B. Sandler. A Tutorial on Onset Detection in Music Signals. *IEEE Transactions on Speech and Audio Processing*, 13(5):1035–1047, Sept 2005.

[BDR06] Roland Badeau, Bertrand David, and Gaël Richard. High-Resolution Spectral Analysis of Mixtures of Complex Exponentials Modulated by Polynomials. *IEEE Transactions On Signal Processing*, 54(4):1341 – 1350, April 2006.

[Ben09] Yoshua Bengio. Learning Deep Architectures for AI. *Foundations and Trends in Machine Learning*, 2(1):1 – 127, 2009.

[Bil09] Stefan Bilbao. *Numerical Sound Synthesis*. John Wiley & Sons, 2009.

[Bis06] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[BS10] J. Bortz and C. Schuster. *Statistik für Human- und Sozialwissenschaftler*. Springer, 2010.

[Cae11] Marcelo Caetano. *Morphing Isolated Quasi-Harmonic Acoustic Musical Instrument Sounds Guided by Perceptually Motivated Features*. PhD thesis, École Doctorale EDITE, Universit'e Paris VI, Pierre et Marie Curie (UPMC), 2011.

[Car09] Alfonso Antonio Pérez Carillo. *Enhancing Spectral Synthesis Techniques with Performance Gestures using the Violin as a Case Study*. PhD thesis, Universitat Pompeu Fabra - Music Technology Group, 2009.

[CBM+12] Alfonso P. Carrillo, Jordi Bonada, Esteban Maestre, Enric Guaus, and Merlijn Blaauw. Performance Control Driven Violin Timbre Model Based on Neural Networks. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(3):1007 – 1021, 2012.

[CBR10] Marcelo Caetano, Juan José Burred, and Xavier Rodet. Automatic Segmentation of the Temporal Evolution of Isolated Acoustic Musical Instrument Sounds using Spectro–Temporal Cues. In *Proc. of the 13th Int. Conference on Digital Audio Effects (DAFx-10)*, Graz, Austria, September 2010.

[CDM14] Tian Cheng, Simon Dixon, and Matthias Mauch. A Comparison of Extended Source–Filter Models for Music Signal Reconstruction. In *Proc. of the 17th Int. Conference on Digital Audio Effects (DAFx-14)*, September 2014.

[CDS01] Scott Shaobing Chen, David L. Donoho, and Michael A. Saunders. Atomic Decomposition by Basis Pursuit. *SIAM Journal on Scientific Computing*, 20(1):129 – 159, 2001.

[Chi95] D. G. Childers. Glottal source modeling for voice conversion. *Speech Communication*, 16(2):127–138, 1995.

[Cho73] John M. Chowning. The Synthesis of Complex Audio Spectra by Means of Frequency Modulation. *Journal of the Audio Engineering Society*, 21(7):526 – 534, September 1973.

[Chu91] Charles K. Chui. *Multivariate Splines*. Society For Industrial And Applied Mathematics, 1991. Second printing.

[CKD+13] Marcelo Caetano, George Kafentzis, Gilles Degottex, Athanasios Mouchtaris, and Yannis Stylianou. Evaluating how well filtered White Noise Models the Residual from Sinusoidal Modeling of Musical Instrument Sounds. In *IEEE ASSP Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2013.

[CLA+63] Melville Clark, David Luce, Robert Abrams, Howard Schlossberg, and James Rome. Preliminary Experiments on the Aural Significance of Parts of Tones of Orchestral Instruments and on Choral Tones. *Journal of the Audio Engineering Society*, 11(1):45 – 54, January 1963.

[Cla82] David Clark. High-Resolution Subjective Testing Using a Double-Blind Comparator. *Journal of the Audio Engineering Society*, 30(5):330 – 338, May 1982.

[CLFC03] Claude Cadoz, Annie Luciani, Jean-Loup Florens, and Nicolas Castagné. ACROE - ICA, Artistic Creation and Computer Interactive Multisensory Simulation Force Feedback Gesture Transducers. In *Conference on New Interfaces for Musical Expression (NIME-03), Proceedings of the 2003*, pages 235 – 246, Montreal, Canada, May 2003.

[CLM95] Olivier Cappé, Jean Laroche, and Eric Moulines. Regularized Estimation of Cepstrum Envelope from Discrete Frequency Points. In *IEEE ASSP Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 1995.

[CM96] Olivier Cappé and Eric Moulines. Regularization Techniques for Discrete Cepstrum Estimation. *IEEE Signal Processing Letters*, 3(4):100 – 102, 1996.

[COCM01] Marine Campedel-Oudot, Olivier Cappé, and Eric Moulines. Estimation of the Spectral Envelope of Voiced Sounds Using a Penalized Likelihood Approach. *IEEE Transactions on Speech and Audio Processing*, 9(5):469 – 481, July 2001.

[COCVCRS13] Julio J. Carabias-Orti, Máximo Cobos, Pedro Vera-Candeas, and Francisco J. Rodriguez-Serranoa. Nonnegative Signal Factorization with learnt Instrument Models for Sound Source Separation in Close-Microphone Recordings. *EURASIP Journal on Advances in Signal Processing*, 184:16, 2013.

[Coh95] Leon Cohen. *Time–Frequency Analysis*. Prentice Hall, Upper Saddle River, NJ, 1995.

[COVVC+11] J.J. Carabias-Orti, T. Virtanen, P. Vera-Candeas, N. Ruiz-Reyes, and F.J. Canadas-Quesada. Musical Instrument Sound Multi-Excitation Model for Non-Negative Spectrogram Factorization. *IEEE Journal of Selected Topics in Signal Processing*, 5(6):1144 – 1158, October 2011.

[CR11] Marcelo Caetano and Xavier Rodet. Improved Estimation of the Amplitude Envelope of Time–Domain Signals Using True Envelope Cepstral Smoothing. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 4244 – 4247, Prague, Czech, May 2011.

[CRYR14] J.P. Cabral, K. Richmond, J. Yamagishi, and S. Renals. Glottal Spectral Separation for Speech Synthesis. *IEEE Journal of Selected Topics in Signal Processing*, 8(2):195 – 208, April 2014.

[CS11] Nick Collins and Bob L. Sturm. Sound Cross–Synthesis and Morphing using Dictionary–Based Methods. In *Proceedings of the International Computer Music Conference (ICMC)*, Huddersfield, UK, 2011.

[CT65] James W. Cooley and John W. Tukey. An algorithm for the Machine Calculation of Complex Fourier Series. *Mathematics of Computation*, 19(90):297 – 301, April 1965.

[CW00] Claude Cadoz and Marcelo M. Wanderley. Gesture-music. In M. Wanderley and M. Battier, editors, *Trends in Gestural Control of Music*. Paris: IRCAM - Centre Pompidou, 2000.

[DAAY14] Thomas Drugman, Paavo Alku, Abeer Alwan, and Bayya Yegnanarayana. Glottal source processing: From analysis to applications. *Computer Speech and Language*, 28(5):1117 – 1138, September 2014.

[Dah80] Wolfgang A. Dahmen. On Multivariate B–splines. *SIAM Journal on Numerical Analysis*, 17(2):179 – 191, April 1980.

[Dan06] Roger B. Dannenberg. The Interpretation of MIDI Velocity. In *Proceedings of the International Computer Music Conference (ICMC)*, 2006.

[dB76] Carl de Boor. Splines as linear combinations of B–splines. A Survey. *Approximation Theory*, II, 1976.

[dB01] Carl de Boor. *A Practical Guide to Splines*. Springer, revised edition edition, 2001.

[DBD11] Thomas Drugman, Baris Bozkurt, and Thierry Dutoit. A comparative study of glottal source estimation techniques. *Computer Speech and Language*, 26(1):20 – 34, January 2011.

[DBR06] Bertrand David, Roland Badeau, and Gaël Richard. HRHA-TRAC Algorithm for Spectral Line Tracking of Musical Signals. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages III–45 – III–48, 2006.

[dCK02] Alain de Cheveigné and Hideki Kawahara. YIN, a fundamental frequency estimator for speech and music. *Journal of the Acoustical Society of America*, 111(4):1917–1930, 2002.

[DCM98] Myriam Desainte-Catherine and Sylvain Marchand. High-Precision Fourier Analysis of Sounds Using Signal Derivatives. *Journal of the Audio Engineering Society*, 48(7/8):654 – 667, July 1998.

[DD14] Li Deng and Yu Dong. *Deep Learning: Methods and Applications [Draft]*. Now Publishers, 2014.

[DDR11] Jean-Louis Durrieu, Bertrand David, and Gaël Richard. Musical Instrument Sound Multi-Excitation Model for Non-Negative Spectrogram Factorization. *IEEE Journal of Selected Topics in Signal Processing*, 5(6):1180 – 1191, October 2011.

[DDS02] Chris Duxbury, Mike Davies, and Mark Sandler. Improved Time-Scaling of Musical Audio Using Phase Locking at Transients. In *112th Audio Engineering Society Convention*, May 2002.

[Deg10] Gilles Degottex. *Glottal Source and Vocal–Tract Separation - Estimation of glottal parameters, voice transformation and synthesis using a glottal model*. PhD thesis, École Doctorale EDITE, Universit'e Paris VI, Pierre et Marie Curie (UPMC), 2010.

[DGR93] Philippe Depalle, Guillermo Garcia, and Xavier Rodet. Tracking of Partials for Additive Sound Synthesis using Hidden Markov Models. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages I 225 – I 228, Minneapolis, MN, USA, April 1993.

[DHS00] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification*. John Wiley & Sons, second edition edition, 2000.

[DLRR13] Gilles Degottex, Pierre Lanchantin, Axel Roebel, and Xavier Rodet. Mixed Source Model and its adapted Vocal Tract Filter Estimate for Voice Transformation and Synthesis. *Speech Communication*, 55(2):278 – 294, 2013.

[DM83] Wolfgang A. Dahmen and Charles A. Micchelli. On the Linear Independence of Multivariate B–splines. II: Complete Configurations. *Mathematics of Computation*, 41(163):143–163, July 1983.

[Dol86] Mark Dolson. The Phase Vocoder: A Tutorial. *Computer Music Journal*, 10(4):14 – 27, 1986.

[dP08] Arantza del Pozo. *Voice Source and Duration Modelling for Voice Conversion and Speech Repair*. PhD thesis, Cambridge University, Engineering Department, April 2008.

[DRR10] Gilles Degottex, Axel Roebel, and Xavier Rodet. Joint Estimate of Shape and Time-Synchronization of a Glottal Source Model by Phase Flatness. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 5058 – 5061, May 2010.

[DRR11a] Gilles Degottex, Axel Roebel, and Xavier Rodet. Phase Minimization for Glottal Model Estimation. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(5):1080 – 1090, 2011.

[DRR11b] Gilles Degottex, Axel Roebel, and Xavier Rodet. Pitch Transposition and Breathiness Modification using a Glottal Source Model and its adapted Vocal–Tract Filter. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 5128 – 5131, Prague, Czech, May 2011.

[Dud39a] Homer Dudley. Remaking Speech. *Journal of the Acoustical Society of America*, 11(2):169 – 177, 1939.

[Dud39b] Homer Dudley. The Vocoder. *Bell Labs Rec.*, 18:122 – 126, 1939.

[EJM91] Amro El-Jaroudi and John Makhoul. Discrete All–Pole Modeling. *IEEE Transactions On Signal Processing*, 39(2):411 – 423, February 1991.

[EM96] Paul H. C. Eilers and Brian D. Marx. Flexible smoothing with B-splines and penalties. *Statistical Science*, 11(2):89–121, 1996.

[Fan79a] Gunnar Fant. Glottal source and excitation analysis. Technical Report 1, Department for Speech, Music and Hearing, KTH, Stockholm, Sweden, 1979. STL – Quarterly Progress and Status Report.

[Fan79b] Gunnar Fant. Vocal source analysis - a progress report. Technical Report 3–4, Department for Speech, Music and Hearing, KTH, Stockholm, Sweden, 1979. STL – Quarterly Progress and Status Report.

[Fan81] Gunnar Fant. The source filter concept in voice production. Technical Report 1, Department for Speech, Music and Hearing, KTH, Stockholm, Sweden, 1981. STL – Quarterly Progress and Status Report.

[FB88] David R. Forsey and Richard H. Bartels. Hierarchical B-Spline Refinement. *Computer Graphics*, 22(4):205 – 212, August 1988.

[FBS62] Harvey Fletcher, E. Donnell Blackham, and Richard Stratton. Quality of Piano Tones. *Journal of the Acoustical Society of America*, 34(6):749 – 761, 1962.

[Fet86] Alfred Fettweis. Wave Digital Filters: Theory and Practice. *Proceedings of the IEEE*, 74(2):14 – 27, 1986.

[FG66] J. L. Flanagan and R. M. Golden. Phase Vocoder. *The Bell System Technical Journal*, 45:1493 – 1509, 1966.

[FHLO02] Kelly Fitz, Lippold Haken, Susanne Lefvert, and Mike O`Donnell. Sound Morphing using Loris and the Reassigned Bandwidth–Enhanced Additive Sound Model: Practice and Applications. In *Proceedings of the International Computer Music Conference (ICMC)*, 2002.

[FLL85] Gunnar Fant, Johan Liljencrants, and Qi-guaq Lin. A four-parameter model of glottal flow. Technical Report 4, Department for Speech, Music and Hearing, KTH, Stockholm, Sweden, 1985. STL – Quarterly Progress and Status Report.

[FLL95] Gunnar Fant, Johan Liljencrants, and Qi-guaq Lin. The LF-model revisited. Transformations and frequency domain analysis. Technical Report 2 – 3, Department for Speech, Music and Hearing, KTH, Stockholm, Sweden, 1995. STL – Quarterly Progress and Status Report.

[FM06] Qiang Fu and Peter Murphy. Robust Glottal Source Estimation Based on Joint Source–Filter Model Optimization. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(2):492 – 501, March 2006.

[FODR15] Xavier Favory, Nicolas Obin, Gilles Degottex, and Axel Roebel. The Role Of Glottal Source Parameters For High–Quality Transformation Of Perceptual Age. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, April 2015.

[FR64] R. Fletcher and C. M. Reeves. Function minimization by conjugate gradients. *The Computer Journal*, 7:149 – 154, 1964.

[FR98] Neville H. Fletcher and Thomas Rossing. *The Physics of Musical Instruments*. Springer-Verlag New York, 2 edition, 1998.

[Fri91] Jerome H. Friedman. Multivariate Adaptive Regression Splines. *The Annals of Statistics*, 19(1):1 – 141, 1991.

[FV13] Andreas Franck and Vesa Välimäki. Higher–order Integrated Wavetable and Sampling Synthesis. *Journal of the Audio Engineering Society*, 61(9):624 – 636, September 2013.

[GA99] Alexander Galembo and Anders Askenfelt. Signal Representation And Estimation Of Spectral Parameters By Inharmonic Comb Filters With Application To The Piano. *IEEE Transactions On Speech And Audio Processing*, 7(2):197 – 203, March 1999.

[gar] Garritan Virtual Instruments. accessed: 2015-02-06.

[GHNO03] Masataka Goto, Hiroki Hashiguchi, Takuichi Nishimura, and Ryuichi Oka. RWC Music Database: Music Genre Database and Musical Instrument Sound Database. In *4th International Society for Music Information Retrieval Conference (ISMIR)*, pages 229 – 230, October 2003.

[GJS12] Carlotta Gianelli, Bert Jüttler, and Hendrik Speleers. THB–splines: The truncated basis for hierarchical splines. *Computer Aided Geometric Design*, 29:485 – 498, 2012.

[GL84] Daniel W. Griffin and Jae S. Lim. Signal Estimation from Modified Short-Time Fourier Transform. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-32(2):236 – 243, april 1984.

[Glo12] John Glover. *Sinusoids, noise and transients: spectral analysis, feature detection and real–time transformations of audio signals for musical applications*. PhD thesis, Department of Music, National University of Ireland, Maynooth, October 2012.

[GMdM+03] Laurent Girin, Sylvain Marchand, Joseph di Martino, Axel Roebel, and Geoffroy Peeters. Comparing the Order of a Polynomial Phase Model for the Synthesis of Quasi–Harmonic Audio Signals. In *IEEE ASSP Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2003.

[GR87] Thierry Galas and Xavier Rodet. An improved cepstral method for deconvolution of source–filer systemms with discrete spectra: Application to musical sound signals. In *Proceedings of the International Computer Music Conference (ICMC)*, Champaign-Urbana, Illinois, 1987.

[GSV11] Leonardo Gabrielli, Stefano Squartini, and Vesa Välimäki. A Subjective Validation Method for Musical Instrument Emulation. In *Proceedings of the 131st Convention of the Audio Engineering Society (AES)*, New York, USA, October 2011.

[Haj96] John M. Hajda. A New Model for Segmenting the Envelope of Musical Signals: The Relative Salience of Steady Sate versus Attack, Revisited. In *Proceedings of the 101st Convention of the Audio Engineering Society (AES)*, Los Angeles, California, USA, November 1996.

[Haj98] John M. Hajda. The Effect of Amplitude and Centroid Trajectories on the Timbre of Percussive and Nonpercussive Orchestral Instruments. *Journal of the Acoustical Society of America*, 103(5):2966 – 2967, May 1998.

[Har78] Frederic J. Harris. On the Use of Windows for Harmonic Analysis with the Discrete Fourier Transform. *Proceedings of the IEEE*, 66(1):51 – 83, January 1978.

[Hay09] Simon Haykin. *Neural Networks and Learning Machines*. Pearson / Prentice Hall, 2009.

[HB96] Andrew Horner and James Beauchamp. Piecewise-Linear Approximation of Additive Synthesis Envelopes: A Comparison of Various Methods. *Computer Music Journal*, 20(2):72–95, 1996.

[HD12] Brian Hamilton and Phillippe Depalle. Comparisons of Parameter Estimation Methods for an Exponential Polynomial Sound Signal Model. In *Audio Engineering Society Conference: 45th International Conference: Applications of Time-Frequency Processing in Audio*, Helsinki, Finland, March 2012.

[HDR11] Romain Hennequin, Bertrand David, and Gaël Richard. NMF With Time–Frequency Activations to Model Nonstationary Audio Events. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(4):744 – 753, May 2011.

[HE01] Stephen Handel and Molly L. Erickson. A Rule of Thumb: The Bandwidth for Timbre Invariance Is One Octave. *Music Perception: An Interdisciplinary Journal*, 19(1):121 – 126, 2001.

[Hel70] Hermann L. F. von Helmholtz. *Die Lehre von den Tonempfindungen als Physiologische Grundlage für die Theorie der Musik*. Druck und Verlag von Friedrich Vieweg und Sohn, Braunschweig, dritte umgearbeitete auflage edition, 1870.

[HKV09] Toni Heittola, Anssi Klapuri, and Tuomas Virtanen. Musical instrument recognition in polyphonic audio using source-filter model for sound separation. In *10th International Society for Music Information Retrieval Conference (ISMIR)*, pages 327 – 332, October 2009.

[HM03] Stephen Hainsworth and Malcolm Macleod. On Sinusoidal Parameter Estimation. In *Proc. of the 6th Int. Conference on Digital Audio Effects (DAFx-03)*, pages 1 – 6, London, United Kingdom, September 2003.

[Hod12] Matthieu Hodgkinson. *Physically Informed Subtraction of a String´s Resonances from Monophonic, Discretely Attacked Tones : a Phase Vocoder Approach*. PhD thesis, Department of Computer Science, Faculty of Science and Engineering, National University of Ireland, Maynooth, May 2012.

[Hoe82] Klaus Hoellig. Multivariate Splines. *SIAM Journal on Numerical Analysis*, 19(5):1013 – 1031, October 1982.

[Hoe03] Klaus Hoellig. *Finite Element Methods with B-Splines*. Society for Industrial and Applied Mathematics, SIAM, 2003.

[HR71a] Lejaren Hiller and Pierre Ruiz. Synthesizing Musical Sounds by Solving the Wave Equation for Vibrating Objects - Part I. *Journal of the Audio Engineering Society*, 19(6):462 – 470, June 1971.

[HR71b] Lejaren Hiller and Pierre Ruiz. Synthesizing Musical Sounds by Solving the Wave Equation for Vibrating Objects - Part II. *Journal of the Audio Engineering Society*, 19(7):542 – 551, July/August 1971.

[HR13] Henrik Hahn and Axel Roebel. Joint f0 and inharmonicity estimation using second order optimization. In *Proceedings of the Sound and Music Computing Conference (SMC)*, Stockholm, Sweden, August 2013.

[HRBW10] Henrik Hahn, Axel Roebel, Juan José Burred, and Stefan Weinzierl. Source–Filter Model For Quasi-Harmonic Instruments. In *Proc. of the 13th Int. Conference on Digital Audio Effects (DAFx-10)*, September 2010.

[HRD12] Stefan Huber, Axel Roebel, and Gilles Degottex. Glottal source shape parameter estimation using phase minimization variants. *Computer Speech and Language*, 28(5):1170 – 1194, September 2012.

[HS52] Magnus R. Hestenes and Eduard Stiefel. Methods of Conjugate Gradients for Solving Linear Systems. *Journal of Research of the National Bureau of Standards*, 49(6):409 – 436, December 1952.

[HWTL09] Matthieu Hodgkinson, Jian Wang, Joseph Timoney, and Victor Lazzarini. Handling inharmonic Series with Median-Adjustive Trajectories. In *Proc. of the 12th Int. Conference on Digital Audio Effects (DAFx-09)*, September 2009.

[itu03] Method for the subjective assessment of intermediate quality level of coding systems, 2003.

[Jen99] Kristoffer Jensen. *Timbre Models of Musical Sounds*. PhD thesis, Department of Computer Science, University of Copenhagen, 1999.

[JS83] David A. Jaffe and Julius O. Smith. Extensions of the Karplus–Strong Algorithm. *Computer Music Journal*, 7(2):56–69, 1983.

[JVK01] H. Järveläinen, Vesa Välimäki, and Matti Karjalainen. Audibility of the timbral effects of inharmonicity in stringed instrument tones. *Acoustics Research Letters Online*, 2(3):79 – 84, 2001.

[KCOL14] Changhyun Kim, Wonil Chang, Sang-Hoon Oh, and Soo-Young Lee. Joint Estimation Multiple Notes and Inharmonicity Coefficient based on f0-Triplet for Automatic Piano Transcription. *IEEE Signal Processing Letters*, 21(12):1536 – 1540, December 2014.

[KD11] Corey Kereliuk and Philippe Depalle. Sparse Atomic Modeling of Audio: A Review. In *Proc. of the 14th Int. Conference on Digital Audio Effects (DAFx-11)*, September 2011.

[KDK13] Holger Kirchoff, Simon Dixon, and Anssi Klapuri. Missing Template Estimation For User-Assisted Music Transcription. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 26 – 30, Vancouver, BC, Canada, May 2013.

[KGJ12] Gábor Kiss, Carlotta Gianelli, and Bert Jüttler. Algorithms and Data Structures for Truncated Hierarchical B–splines. In *Proceedings of the 8th International Conference, Mathematical Methods for Curves and Surfaces (MMCS 2012)*, Oslo, Norway, June/July 2012.

[Kla07] Anssi Klapuri. Analysis of Musical Instrument Sounds by Source-Filter-Decay Model. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, volume 1, pages I–53 – I–56, April 2007.

[Kli09] Michael Kateley Klingbeil. *Spectral Analysis, Editing and Resynthesis: Methods and Applications*. PhD thesis, Columbia University, Graduate School of Arts and Sciences, 2009.

[KM02] Florian Keiler and Sylvain Marchand. Survey on Extraction of Sinusoids in Stationary Sounds. In *Proc. of the 5th Int. Conference on Digital Audio Effects (DAFx-02)*, pages 51 – 58, Hamburg, Germany, September 2002.

[kon] Native Instruments - Kontakt Library. accessed: 2015-02-06.

[KP12] Stefan Kersten and Hendrik Purwins. Fire Texture Sound Re–Synthesis Using Sparse Decomposition and Noise Modelling. In *Proc. of the 15th Int. Conference on Digital Audio Effects (DAFx-12)*, York, UK, September 2012.

[Kra94] Rainer Kraft. Hierarchical b-splines. Technical report, Mathematisches Institut A, Universität Stuttgart, 1994.

[Kra97] Rainer Kraft. *Adaptive and Linearly Independent Multilevel B-splines*. Bericht, Sonderforschungsbereich Mehrfeldprobleme in der Kontinuumsmechanik. Stuttgart SFB 404, Geschäftsstelle, 1997.

[KS83] Kevin Karplus and Alex Strong. Digital Synthesis of plucked String and Drum Timbres. *Computer Music Journal*, 7(2):43–55, 1983.

[KT78] Werner Kaegi and Stan Templaars. VOSIM - A New Sound Synthesis System. *Journal of the Audio Engineering Society*, 26(6):418 – 425, June 1978.

[KVH10] Anssi Klapuri, Tuomas Virtanen, and Toni Heittola. Sound source separation in monaural music signals using excitation–filter model and EM algorithm. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 5510 – 5513, May 2010.

[KVT98] Matti Karjalainen, Vesa Välimäki, and Tero Tolonen. Plucked–String Models: From the Karplus–Strong Algorithm to Digital Waveguides and Beyond. *Computer Music Journal*, 22(3):17 – 32, 1998.

[KZ10] Adrian von dem Knesebeck and Udo Zölzer. Comparison of pitch trackers for real–time guitar effects. In *Proc. of the 13th Int. Conference on Digital Audio Effects (DAFx-10)*, Graz, Austria, September 2010.

[Lar89a] Jean Laroche. A New Analysis/Synthesis System of Musical Signals Using Prony´s Method. Application to Heavily Damped Percussive Sounds. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 2053 – 2056. IEEE, May 1989.

[Lar89b] Jean Laroche. *Etude d`un système d`analyse et de synthèse utilisant la méthode de prony. Application aux instruments de musique de type percussif*. PhD thesis, Telecom Paris 89 E 009, 1989.

[LB78] Marc Le Brun. Digital Waveshaping Synthesis. *Journal of the Audio Engineering Society*, 27(4):250 – 266, April 1978.

[LD97] Jean Laroche and Mark Dolson. Phase Vocoder - About This Phasiness Business. In *IEEE ASSP Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 1997.

[LD99a] Jean Laroche and Mark Dolson. Improved Phase Vocoder Time-Scale Modification of Audio. *IEEE Transactions On Speech and Audio Processing*, 7(3):323 – 332, May 1999.

[LD99b] Jean Laroche and Mark Dolson. New Phase-Vocoder Techniques For Pitch-Shifting, Harmonizing and other exotic effects. In *IEEE ASSP Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 91 – 94, 1999.

[Lev98] Scott Nathan Levine. *Audio Representations for Data Compression and Compressed Domain Processing*. PhD thesis, Center for Computer Research in Music and Acoustics (CCRMA), Department of Music, Stanford University, Stanford, California 94305 USA, 1998.

[Lin07] Eric Lindemann. Music Synthesis with Reconstructive Phrase Modeling. *IEEE Signal Processing Magazine*, 24(2):80 – 91, March 2007.

[LMR07] Mathieu Lagrange, Sylvain Marchand, and Jean-Bernard Rault. Enhancing the Tracking of Partials for the Sinusoidal Modeling of Polyphonic Sounds. *IEEE Transactions on Audio, Speech and Language Processing*, 15(5):1625 – 1634, July 2007.

[LR13] Marco Liuni and Axel Roebel. Phase Vocoder and beyond. *Musica/Tecnologia*, 7-2013:73 – 89, 2013.

[LSM93] Jean Laroche, Yannis Stylianou, and Eric Moulines. HNM - A Simple, Efficient Harmonic + Noise Model for Speech. In *IEEE ASSP Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 169 –172, 1993.

[Lue99] Hans Dieter Lueke. The origins of the sampling theorem. *IEEE Communications Magazine*, 37(4):106 – 108, April 1999.

[Mae09] Esteban Maestre. *Modeling Instrumental Gestures - An Analysis/Synthesis Framework for Violin Bowing*. PhD thesis, Universitat Pompeu Fabra - Music Technology Group, 2009.

[Mag10] Nick Magnus. Native Instruments Kontakt 4. *Sound on Sound*, February 2010.

[Mak75] John Makhoul. Linear Prediction: A Tutorial Review. *Proceedings of the IEEE*, 63(4):561 – 580, April 1975.

[Mal93] Stéphane G. Mallat. Matching Pursuit With Time–Frequency Dictionaries. *IEEE Transactions on Signal Processing*, 41(12):3397 – 3415, December 1993.

[Mal09] Stèphane Mallat. *A Wavelet Tour of Signal Processing, The Sparse Way*. Academic Press, September 2009.

[Mas02] Dana C. Massie. Wavetable sampling synthesis. In Mark [Kahrs and Karlheinz] Brandenburg, editors, *Applications of digital signal processing to audio and acoustics*, pages 311 – 341. Kluwer Academic Publishers, 2002.

[Mat63] Max Vernon Mathews. The Digital Computer as a Musical Instrument. *Science*, 142(3592):553–557, November 1963.

[Mat69] Max Vernon Mathews. *The Technology of Computer Music*. The M.I.T. Press, 1969.

[May12] Javier Perez Mayos. *Voice Source Characterization for Prosodic and Spectral Manipulation*. PhD thesis, Universitat Politecnica de Catalunya, July 2012.

[MB10] Sašo Muševiv and Jordi Bonada. Comparison of Non-Stationary Sinusoid Estimation Methods using Reassignment and Derivatives. In *Proceedings of the Sound and Music Computing Conference 2010 (SMC 2010)*, Barcelona, Spain, July/August 2010.

[MB11] Sašo Muševič and Jordi Bonada. Generalized Reassignment with an Adaptive Polynomial-Phase Fourier Kernel for the Estimation of Non–Stationary Sinusoidal Parameters. In *Proc. of the 14th Int. Conference on Digital Audio Effects (DAFx-11)*, September 2011.

[MBB+10] Esteban Maestre, Merlijn Blaauw, Jordi Bonada, Enric Guaus, and Alfonso Pérez. Statistical Modeling of Bowing Control Applied to Violin Sound Synthesis. *IEEE Transactions on Audio, Speech and Language Processing*, 18(4):855 – 871, 2010.

[McL08] Philip McLeod. *Fast, Accurate Pitch Detection Tools for Music Analysis*. PhD thesis, University of Otago, Dunedin, New Zealand, 2008.

[MD14] Matthias Mauch and Simon Dixon. PYIN: A Fundamental Frequency Estimator Using Probabilistic Threshold Distributions. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 659 – 663, May 2014.

[MEKR11] Meinard Müller, Daniel P. W. Ellis, Anssi Klapuri, and Gaël" Richard. Signal Processing for Music Analysis. *IEEE Journal of Selected Topics in Signal Processing*, 5(6):1088 – 1110, October 2011.

[MLV13] Rémi Mignot, Heidi-Maria Lehtonen, and Vesa Välimäki. Warped low-order Modeling of Musical Tones. In *Proceedings of the Sound and Music Computing Conference 203 (SMC 2013)*, pages 622 – 627, July/August 2013.

[Mø93] Martin Fodslette Møller. A scaled Conjugate Gradient Algorithm for fast supervised Learning. *Neural Networks*, 6(4):525 – 533, 1993.

[Moo76] James A. Moorer. The Use of the Phase Vocoder in Computer Music Applications. In *55th Audio Engineering Society Convention*, October/November 1976.

[Moo12] Brian C. J. Moore. *An Introduction to the Psychology of Hearing*. Emerald Group Publishing Ltd., 6th edition, january 2012.

[MQ85] Robert J. McAulay and Thomas F. Quatieri. Mid–Rate Coding based on a sinusoidal representation of speech. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 744 – 754, Tampa, Florida, 1985. IEEE.

[MQ86] Robert J. McAulay and Thomas F. Quatieri. Speech Analysis/Synthesis based on a Sinusoidal Representation. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 34(4):744 – 754, August 1986.

[MV13] Rémi Mignot and Vesa Välimäki. Perceptual Cepstral Filters for Speech and Music Processing. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pages 1 – 4, New Paltz, NY, October 2013.

[MV14a] Rémi Mignot and Vesa Välimäki. Extended Subtractive Synthesis of Harmonic Musical Tones. In *136th Audio Engineering Society Convention*, April 2014.

[MV14b] Rémi Mignot and Vesa Välimäki. Low-order ARMA approximation using a perceptually-based criterion, for sound synthesis. Technical report, Aalto University, Department of Signal Processing and Acoustics, Espoo, Finland, 2014.

[MV14c] Rémi Mignot and Vesa Välimäki. Perceptual Linear Filters – Low-order ARMA Approximations for Sound Synthesis. In *Proc. of the 17th Int. Conference on Digital Audio Effects (DAFx-14)*, Erlangen, Germany, September 2014.

[MV14d] Rémi Mignot and Vesa Välimäki. True Discrete Cepstrum: An Accurate and Smooth Spectral Envelope Estimation for Music Processing. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 7465 – 7469, May 2014.

[MW05] Philip McLeod and Geoff Wyvill. A smarter Way to find Pitch. In *Proceedings of the International Computer Music Conference (ICMC)*, Barcelona, Spain, 2005.

[Nol67] A. Michael Noll. Cepstrum Pitch Determination. *Journal of the Acoustical Society of America*, 41(2):293 – 309, February 1967.

[Nol70] A. Michael Noll. Pitch Determination of Human Speech by the Harmonic Product Spectrum, the Harmonic Sum Spectrum and a Maximum Likelihood Estimate. In *Proceedings of the Symposium on Computer Processing in Communications, Vol. XIX, Polytechnic Press*, pages 779 – 797, Brooklyn, New York, 1970.

[Nut81] Albert H. Nuttal. Some Windows with Very Good Sidelobe Behavior. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-29(1):84 – 91, February 1981.

[OCM97] Marine Oudot, Olivier Cappé, and Eric Moulines. Robust estimation of the spectral envelope for "harmonics + noise" models. In *Speech Coding For Telecommunications, Proceeding, IEEE Workshop on*, pages 11 – 12, 1997.

[O'L09] Sean O'Leary. *Physically Informed Spectral Modelling of Musical Instrument Tones*. PhD thesis, The University of Limerick, 2009.

[OS10] Alan V. Oppenheim and Ronald W. Schafer. *Discrete-Time Signal Processing*. Prentice Hall, 3rd edition, 2010.

[Oud98] Marine Campedel Oudot. *Étude du modèle 'sinusoïdes et bruit' pour le traitement des signaux de parole Estimation robuste de l'enveloppe spectrale*. PhD thesis, Telecom Paris, 1998.

[PCBPV11] Alfonso Perez Carrillo, Jordi Bonada, Jukka Pätynen, and Vesa Välimäki. Method for measuring violin sound radiation based on bowed glissandi and its application to sound synthesis. *Journal of the Acoustical Society of America*, 130(2):1020 – 1029, August 2011.

[Pee04] Geoffroy Peeters. A large set of audio features for sound descryiption (similarity and classification) in the CUIDADO project. Technical report, Ircam, Analysis/Synthesis Team, 2004.

[Por76] Michael R. Portnoff. Implementation of the Digital Phase Vocoder using the Fast Fourier Transform. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-24(3):243 – 248, June 1976.

[Por78] Michael Rodney Portnoff. *Time-Scale Modification of Speech Based on Short-Time Fourier Analysis*. PhD thesis, Massachusetts Institute of Technology, 1978.

[Por80] Michael R. Portnoff. Time-Frequency Representation of Digital Signals and Systems Based on Short-Time Fourier Analysis. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-28(1):55 – 69, February 1980.

[Por81] Michael R. Portnoff. Time-Scale Modification of Speech Based on Short-Time Fourier Analysis. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-29(3):374 – 390, June 1981.

[PR69] E. Polak and G. Ribiere. Note sur la convergence de méthodes de directions conjugueés. *Revue Française d´Informatique et de Recherche Opérationnelle*, 16:35 – 43, 1969.

[Puc95] Miller Puckette. Phase-Locked Vocoder. In *Proceedings of the IEEE Conference on Applications of Signal Processing to Audio and Acoustics*, 1995.

[QM86] Thomas F. Quatieri and Robert J. McAulay. Speech Transformations Based on a Sinusoidal Representation. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP–34(6):1449 – 1464, December 1986.

[Rab68] Lawrence R. Rabiner. Digital Formant Synthesizer for Speech–Synthesis Studies. *Journal of the Acoustical Society of America*, 43(4):720 – 734, 1968.

[RDD11] François Rigaud, Bertrand David, and Laurent Daudet. A Parametric Model of Piano Tuning. In *Proc. of the 14th Int. Conference on Digital Audio Effects (DAFx-11)*, September 2011.

[RDD12] François Rigaud, Bertrand David, and Laurent Daudet. Piano Sound Analysis Using Non-negative Matrix Factorization with Inharmonicity Constraint. In *Proceedings of the 20th European Signal Processing Conference (EUSIPCO)*, pages 2462 – 2466, August 2012.

[RDD13] François Rigaud, Bertrand David, and Laurent Daudet. A Parametric Model and Estimation Techniques for the Inharmonicity and Tuning of the Piano. *Journal of the Acoustical Society of America*, 133(5):3107 – 3118, May 2013.

[Rei67] Christian H. Reinsch. Smoothing by Spline Functions. *Numerische Mathematik*, 10(3):177 – 183, 1967.

[RHRD12] Axel Roebel, Stefan Huber, Xavier Rodet, and Gilles Degottex. Analysis and Modification of Excitation Source Characteristics for Singing Voice Synthesis. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 5381 – 5384, March 2012.

[Ris85] Jean-Claude Risset. Computer Music Experiments 1964 - ... *Computer Music Journal*, 9(1):11–18, 1985.

[Ris07] Jean-Claude Risset. Fifty Years of Digital Sound for Music. In *Proceedings of the 4th Sound and Music Computing Conference (SMC)*, July 2007.

[RK89] Richard Roy and Thomas Kailath. ESPRIT – Estimation of Signal Parameters Via Rotational Invariance Techniques. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(7):984 – 995, July 1989.

[RLV07] Jukka Rauhala, Heidi-Maria Lehtonen, and Vesa Välimäki. Fast Automatic Inharmonicity Estimation Algorithm. *Journal of the Acoustical Society of America*, 121(5 Pt1):EL 184 – 189, May 2007.

[Roa96] Curtis Roads. *The Computer Music Tutorial*. The MIT Press, Philadelphia, PA, February 1996.

[Roa04] Curtis Roads. *Microsound*. The MIT Press, August 2004.

[Roe03a] Axel Roebel. A new Approach to Transient Processing in the Phase Vocoder. In *Proc. of the 6th Int. Conference on Digital Audio Effects (DAFx-03)*, pages 344 – 349, London, United Kingdom, September 2003.

[Roe03b] Axel Roebel. Transient detection and preservation in the phase vocoder. In *Proceedings of the International Computer Music Conference (ICMC)*, Singapore, September/October 2003.

[Roe06] Axel Roebel. Adaptive Additive Modeling With Continuous Parameter Trajectories. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(4):1440 – 1453, July 2006.

[Roe10a] Axel Roebel. A Shape-Invariant Phase Vocoder for Speech Transformation. In *Proc. of the 13th Int. Conference on Digital Audio Effects (DAFx-10)*, Graz, Austria, September 2010.

[Roe10b] Axel Roebel. Between Physics and Perception - Signal Models For High-Level Audio Processing. In *Proc. of the 13th Int. Conference on Digital Audio Effects (DAFx-10)*, Graz, Austria, September 2010.

[Roe10c] Axel Roebel. Shape-Invariant Speech Transformation With The Phase Vocoder. In *Proc. International Conf. on Spoken Language Processing (InterSpeech)*, page 2146–2149, September 2010.

[RR05a] Axel Roebel and Xavier Rodet. Efficient Spectral Envelope Estimation and its application to pitch shifting and envelope preservation. In *Proc. of the 8th Int. Conference on Digital Audio Effects (DAFx-05)*, pages 30 – 35, Madrid, Spain, Septembre 2005.

[RR05b] Axel Roebel and Xavier Rodet. Real Time Signal Transposition with Envelope Preservation in the Phase Vocoder. In *Proceedings of the International Computer Music Conference (ICMC)*, pages 672 – 675, Barcelona, Spain, 2005.

[RS78] Lawrence R. Rabiner and R.W. Schafer. *Digital Processing of Speech Signals*. Prentice Hall, 1978.

[RS07] Xavier Rodet and Diemo Schwarz. Spectral envelopes and additive + residual analysis/synthesis. In James W. Beauchamp, editor, *Analysis, Synthesis, and Perception of Musical Sounds*, pages 175 – 227. Springer, 2007.

[RSDVC+15] Francisco J. Rodriguez-Serranoa, Zhiyao Duan, Pedro Vera-Candeas, Bryan Pardo, and Julio J. Carabias-Orti. Online Score–Informed Source Separation with Adaptive Instrument Models. *Journal of New Music Research*, page 14, January 2015.

[RV07] Jukka Rauhala and Vesa Välimäki. F0 Estimation of Inharmonic Piano Tones using Partial Frequencies Deviations Method. In *Proceedings of the International Computer Music Conference (ICMC)*, March 2007.

[RVR07] Axel Roebel, Fernando Villavicencio, and Xavier Rodet. On cepstral and all-pole based spectral envelope modeling with unknown model order. *Pattern Recognition Letters*, 28:1343 – 1350, March 2007.

[Saa03] Yousef Saad. *Iterative Methods for Sparse Linear Systems*. Society for Industrial and Applied Mathematics, SIAM, 2003.

[SC78] Hiroaki Sakoe and Seibi Chiba. Dynamic Programming Algorithm Optimization for Spoken Word Recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-2(1):43 – 49, February 1978.

[Sch66] Manfred R. Schroeder. Vocoders: Analysis and Synthesis of Speech. *Proceedings of the IEEE*, 54(5):720 – 734, May 1966.

[Sch86] Ralph Schmidt. Multiple Emitter Location and Signal Parameter Estimation. *IEEE Transactions on Antennas and Propagation*, AP-34(3):276 – 280, March 1986.

[Sch04] Diemo Schwarz. *Data-Driven Concatenative Sound Synthesis*. PhD thesis, École Doctorale d'Informatique, Université Paris VI, Pierre et Marie Curie (UPMC), 2004.

[SD11] Nicolas Sturmel and Laurent Daudet. Signal reconstruction from STFT magnitude: A state of the art. In *Proc. of the 14th Int. Conference on Digital Audio Effects (DAFx-11)*, September 2011.

[SdP99] Augusto Sarti and Giovanni de Poli. Toward Nonlinear Wave Digital Filters. *IEEE Transactions On Signal Processing*, 47(6):1654 – 1668, June 1999.

[Sed07] Thomas W. Sederberg. System and method for defining T–spline and T–NURCC surfaces using local refinements, September 2007. US Patent 7,274,364.

[Sed14] Thomas W. Sederberg. *Computer Aided Geometric Design*. Course Notes, October 2014.

[Ser89] Xavier Serra. *A System for Sound Analysis/Transformation/Synthesis based on a Deterministic plus Stochastic Decomposition*. PhD thesis, Center for Computer Research in Music and Acoustics (CCRMA), Department of Music, Stanford University, Stanford, California 94305 USA, 1989.

[Ser97a] Marie-Hélène Serra. Introducing the Phase Vocoder. In Curtis Roads, Steven Travis Pope, Aldo Piccialli, and Giovanni de Poli, editors, *Musical Signal Processing*, chapter 2, pages 31 – 90. Swets & Zeitlinger B. V., 1997.

[Ser97b] Xavier Serra. Current perspectives in the digital synthesis of musical sounds. Technical Report 1, Universitat Pompeu Fabra, Barcelona, Spain, 1997. Formats.

[Ser97c] Xavier Serra. Musical Sound Modeling with Sinusoids Plus Noise. In Curtis Roads, Steven Travis Pope, Aldo Piccialli, and Giovanni de Poli, editors, *Musical Signal Processing*, chapter 3, pages 91 – 122. Swets & Zeitlinger B. V., 1997.

[SG84] Julius O. Smith and Phil Gosset. A flexible Sampling Rate Conversion Method. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 1941 – 1944, 1984.

[She94] Jonathan R. Shewchuk. An introduction to the conjugate gradient method without the agonizing pain. Technical report, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, 1994.

[Smi87] Julius O. Smith. Music Applications of Digital Waveguides. Technical Report STAN-M-39, Center for Computer Research in Music and Acoustics (CCRMA), Department of Music, Stanford University, Stanford, California 94305 USA, 1987.

[Smi91] Julius O. Smith. Viewpoints on the History of Digital Synthesis. In *Proceedings of the International Computer Music Conference (ICMC)*, pages 1–10, Montreal, Canada, October 1991.

[Smi92a] Julius O. Smith. Physical Modeling using Digital Waveguides. *Computer Music Journal*, 16(4):74 – 91, 1992. Special Issue on Physical Modeling, Part I.

[Smi92b] Julius O. Smith. Viewpoints on the History of Digital Synthesis. *Cahier de l´IRCAM*, 1992.

[Smi05] Julius O. Smith. Viewpoints on the history of digital synthesis. Technical report, Center for Computer Research in Music and Acoustics (CCRMA), Department of Music, Stanford University, Stanford, California 94305 USA, 2005.

[Smi10a] Julius O. Smith. *Physical Audio Signal Processing*. `http://-ccrma.stanford.edu/~jos/pasp/`, December 2010. online book.

[Smi10b] Julius O. Smith. *Spectral Audio Signal Processing (Draft)*. `http://ccrma.stanford.edu/~jos/sasp/`, March 2010. online book.

[SS87] Julius O. Smith and Xavier Serra. PARSHL: An Analysis/Synthesis Program for Non–Harmonic Sounds Based on a Sinusoidal Representation. In *Proceedings of the International Computer Music Conference (ICMC)*, Champaign-Urbana, Illinois, 1987.

[SS90] Xavier Serra and Julius O. Smith. Spectral Modeling Synthesis - A Sound Analysis-Synthesis System Based on a Deterministic and Stochastic Decomposition. *Computer Music Journal*, 14(4):12 – 24, 1990.

[SS02] Bernhard Schölkopf and Alexander J. Smola. *Learning with Kernels*. The MIT Press, Cambridge, Massachusetts, 2002.

[Stu09] Bobby Lee Townsend Sturm. *Sparse Approximation and Atomic Decomposition: Considering Atom Interactions in Evaluating and Building Signal Representations*. PhD thesis, University of California, Santa Barbara, California, 2009.

[Sty96] Yannis Stylianou. *Harmonic plus Noise Models for Speech combined with Statistical Methods for Speech and Speaker Modification*. PhD thesis, Telecom Paris, 1996.

[Tho05] Harvey Thornburg. *Detection and Modeling of Transient Audio Signals with Prior Information*. PhD thesis, Department of Electrical Engineering, Stanford University, Stanford, California 94305 USA, 2005.

[TR03] Lutz Trautmann and Rudi Rabenstein. *Digital Sound Synthesis by Physical Modeling Using the Functional Transformation Method*. Kluwer Academic/Plenum Publishers, New York, 2003.

[uvia] Grand Piano Collection. accessed: 2014-06-26.

[uvib] Ircam Solo Instruments. accessed: 2014-06-26.

[Ver99] Tony S. Verma. *A Perceptually Based Audio Signal Model With Applications To Scalable Audio Processing*. PhD thesis, Center for Computer Research in Music and Acoustics (CCRMA), Department of Music, Stanford University, Stanford, California 94305 USA, 1999.

[vie] Vienna Symphonic Library. accessed: 2015-02-06.

[VK06] Tuomas Virtanen and Anssi Klapuri. Analysis Of Polyphonic Audio Using Source-Filter Model And Non-Negative Matrix Factorization. In *Advances in Models for Acoustic Processing, Neural Information Processing Systems Workshop*, December 2006.

[VPEK06] Vesa Välimäki, Jyri Pakarinen, Cumhur Erkut, and Matti Karjalainen. Discrete–time modelling of musical instruments. *Reports on Progress in Physics*, 69(1):1 – 78, Jan. 2006.

[VRR08] Fernando Villaviciencio, Axel Roebel, and Xavier Rodet. Extending Efficient Spectral Envelope Modeling To Mel-Frequency Based Representation. In *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 1625 – 1628, Las Vegas, NV, USA, March/April 2008.

[Wan01] Marcelo M. Wanderley. Gestural Control of Music. In *Proceedings of the International Workshop Human Supervision and Control in Engineering and Music*, Kassel, Germany, September 2001.

[WM07] Jacqueline Walker and Peter Murphy. A review of glottal waveform analysis. In Yannis Stylianou, Marcos Faundez-Zanuy, and Anna Esposito, editors, *Progress in nonlinear speech processing*, Lecture Notes in Computer Science, pages 1 – 21. Springer Berlin Heidelberg, 2007.

[XS09] Wen Xue and Mark Sandler. Notes on Model–Based Non–Stationary Sinusoid Estimation Methods using Derivatives. In *Proc. of the 12th Int. Conference on Digital Audio Effects (DAFx-09)*, September 2009.

[You52] Robert W. Young. Inharmonicity of Plain Wire Piano Strings. *Journal of the Acoustical Society of America*, 24(3):267 – 273, May 1952.

[YR09] Chunghsin Yeh and Axel Roebel. Multiple-F0 Estimation For Mirex 2009. In *MIREX Evaluation*, 2009.

[YR10] Chunghsin Yeh and Axel Roebel. Multiple-F0 Estimation For Mirex 2010. In *MIREX Evaluation*, 2010.

[YR11] Chunghsin Yeh and Axel Roebel. Multiple-F0 Estimation For Mirex 2011. In *MIREX Evaluation*, 2011.

[YRR10] Chunghsin Yeh, Axel Roebel, and Xavier Rodet. Multiple Fundamental Frequency Estimation and Polyphony Inference of Polyphonic Signals. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(6):1116 – 1126, August 2010.

[ZRR04] Miroslav Zivanovic, Axel Roebel, and Xavier Rodet. A new approach to spectral peak classification. In *Proceedings of the 12th European Signal Processing Conference (EUSIPCO)*, pages 1277–1280, Vienna, Austria, Septembre 2004.

This page is intentionally left blank.

Part IV

Appendix

Appendix A

Inharmonicity Estimation

A.1 The Method

The method for the joint estimation of a signals fundamental frequency and inharmonicity coefficient has been presented in the conference proceeding [HR13]. This appendix contains a thorough summary of the method and lists all evaluation results.

Overview

The proposed method jointly estimates the inharmonicity coefficient β and the fundamental frequency f_0 in an iterative manner which can be used on several frames at once and is illustrated in figure A.1. For the algorithm a signal segment $y(t)$ behind the signals attack frame is selected to ensure that the algorithm analyses no transient components. A standard f_0 estimation [dCK02] is applied and this initial value of f_0 is then being used to set the analysis parameters for the STFT adaptively to guarantee a suitable analysis window length according to the coarse estimate of the signal's fundamental. The STFT is taken for N overlapping frames n yielding $Y(f, n)$ and all spectral bins are classified into the 3 classes: main lobe, side lobe or noise component using the peak classification method proposed by Zivanovic et al. [ZRR04].

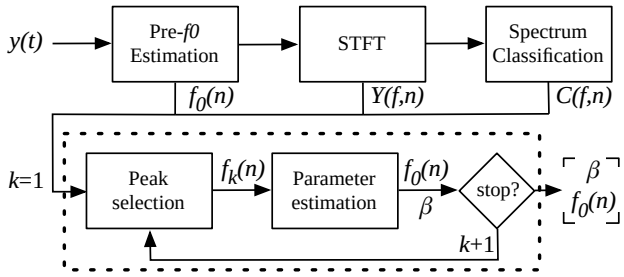


Figure A.1: General scheme of proposed iterative method.

The algorithms main loop identifies a valid peak for the current partial index within each frame and estimates a new f_0 for each frame n and a new β for all frames within each iteration until some abort criterion has been reached. With

increasing partial index the estimated parameters are assumed to converge and to their optimal values.

Peak selection step

The selection of a valid peak within the spectrum is done in 4 steps:

1. Estimation the frequency of the current partial $\hat{f}_k(n)$ by using eq. (3.11) and using the initial $f_0(n)$ and $\beta = 0$ for the first iteration, and the updated values in all later ones.
2. Selection of all spectral peaks classified as main lobe within a narrow band f_b around the estimated partials frequency $\hat{f}_k(n)$: $\hat{f}_k(n) - pf_0(n) \leq f_b \leq \hat{f}_k(n) + pf_0(n), p = .25$
3. If two or more peak candidates have been found within at least one frame we apply a logarithmic amplitude weighting function using a Hann window, centered at the estimated position $\hat{f}_k(n)$ with window length f_b and the peak with the strongest logarithmic amplitude after weighting gets selected.
4. Refinement of the frequency value of the selected peaks using the QIFFT [Ser89] with additional bias correction [AS04].

Estimation step

With at least 3 partials within one frame, we can estimate the parameters β and $f_0(n)$ for all frames n. As shown in eq. (A.1) we use the squared deviation of our estimated values from the measured partial frequencies normalized with the fundamental frequency to achieve equal error surface scalings for all possible fundamental frequencies. The final objective function with normalizations according to the number of frames N and amount of partials per frame $K(n)$ is given in eq. (A.2).

$$R = \frac{1}{2}\left(\frac{f_k(n) - kf_0(n)\sqrt{1+k^2\beta}}{f_0(n)}\right)^2 \tag{A.1}$$

$$O_1 = \frac{1}{N}\sum_{n=1}^{N}\frac{1}{K(n)}\sum_{k=1}^{K(n)} R \tag{A.2}$$

Since the objective function (A.2) reflects the least-mean-squared (LMS) error of all f_0-normalized deviations of our partial frequency estimations with their measured peak frequency counterparts, optimization reflects a fitting of eq. (3.11) to the measured data in the LMS sense. The optimization is being done by a gradient descent approach, whereas we utilize the method of the scaled conjugate gradient [Mø93], denoted CG throughout this document, for faster convergence compared with other methods. The gradient functions for both parameters are shown in eq. (A.3) and eq. (A.4).

$$\frac{\partial R}{\partial \beta} = -\frac{k^3}{2\sqrt{1+k^2\beta}} \tag{A.3}$$

$$\frac{\partial R}{\partial f_0(n)} = -\frac{f_k(n)}{f_0(n)^2} \tag{A.4}$$

Stop criterion

We only use two disjunctive abort criteria: If the next partial $\hat{f}_k(n)$ in the peak selection process would raise above the Nyquist frequency within one frame n or if no valid partial has been found for 3 consecutive iterations in at least one frame of the main loop. This means, the algorithm tries to use as much partials as possible of the signal, since it only stops, if the signals maximum bandwidth or some supposed noise level has been reached.

A.2 Evaluation

For the evaluation we will compare the results of our proposed method with the results of 3 methods briefly discussed in chapter 3.3.4.1: Inharmonic Comb Filters (ICF), median-adjustive trajectories (MAT) and the on-negative matrix factorization based method (NMF). Our proposed method will be denoted CG in the following figures.

We will use an artificial data sound of inharmonic sounds, created using an additive synthesis model and inharmonicity values taken from the tessitura model for the β coefficient shown in [RDD11] as well as the 3 piano data sets from the RWC library [GHNO03] and a piano sound set taken from the IRCAM Solo Instruments library which had also been used for the instrument modeling method.

The artificial data set will be used to compare all β coefficient estimation algorithms with a given ground truth. For the general evaluation of all data sets we will establish a tessitura model for the evolution of the coefficient for all sound samples contained in each data set. The tessitura model for the evolution of β over the MIDI index is derived from [RDD11] and will be used to measure the variance of each estimation algorithm to quantify its accuracy. Furthermore, we will compare the computational efficiency of all algorithms by measuring their realtime factors. For each algorithm a MATLAB$^{\text{TM}}$ implementation has been used therefore the realtime factors are more suitable for a comparison in between the algorithms rather than to give an indication for the performance of native implementations.

For all algorithms we used equal analysis parameters to ensure all algorithms analyze exactly the same frames of the signals and as most other algorithms also need a pre-f_0 estimation, we used the same pre-f_0 for all of them. The window length for the STFT was set to 6 times the coarse estimation of the signals fundamental period with 4 times spectral oversampling and a Blackman window. As our algorithm works on several frames, we took 3 consecutive frames with a hopsize of 1/8 of the analysis window length, whereas the other algorithms analyzed the 3 frames independently.

Tessitura model of the β coefficient

The tessitura model for the β coefficient introduced in [RDD11] is a function of the MIDI value m representing its evolution for the whole keyboard of a piano. It can be represented as the sum of two linear asymptotes in the logarithmic scale, whereas these two asymptotes are being described as Treble (b_T) and Bass bridge (b_B) and are characterized as linear functions, parametrized by its slope and constant value, such that the model $\beta_\phi(m)$ can be described as:

$$\beta_\phi(m) = e^{b_B(m)} + e^{b_T(m)} \tag{A.5}$$

$$= e^{(\phi_1 m+\phi_2)} + e^{(\phi_3 m+\phi_4)} \tag{A.6}$$

with ϕ being a vector of four elements containing the slope and constant parameters of the linear functions b_B and b_T respectively. All algorithms apart from ours estimate 3 coefficients, denoted $\hat{\beta}$, for each input sound file according to the 3 signal frames which are being used by our algorithm to estimate a single value. A curve fitting is done in a least-squares sense by minimizing the variance of the model $\beta_\phi(m)$ according to (A.7) with M^* representing the estimates of a single algorithm for one data set. We are using the logarithm of β as well as $\hat{\beta}$ for the objective function to account for the logarithmic behavior of the β coefficient.

$$O_2 = \frac{1}{2}\sum_{m}^{M^*} |\log(\hat{\beta}(m)) - \log(\beta_\phi(m))|^2 \tag{A.7}$$

Again we are using the scaled Conjugate Gradient method [Mø93] to obtain the tessitura model $\beta_\phi(m)$ with minimum variance using the gradients (A.8) and (A.9) for optimizing the parameters for the functions b_B and b_T with i either being set to 1 or 3 for eq. (A.8) or set to 2 or 4 for eq. (A.9). The four initial values for the vector ϕ are chosen as $[-0.09, -6.87, 0.09, -13.70]^T$.

$$\frac{\partial O_2}{\partial \phi_{1|3}} = \sum_{m}^{M^*} |\log(\hat{\beta}(m)) - \log(\beta_\phi(m))| \frac{m e^{(\phi_i m+\phi_{(i+1)})}}{\beta_\phi(m)} \tag{A.8}$$

$$\frac{\partial O_2}{\partial \phi_{2|4}} = \sum_{m}^{M^*} |\log(\hat{\beta}(m)) - \log(\beta_\phi(m))| \frac{e^{(\phi_{(i-1)} m+\phi_{(i)})}}{\beta_\phi(m)} \tag{A.9}$$

As the estimation algorithms may give fairly noisy results especially for the upper pitch range we delimit the usage of $\hat{\beta}$ values to a range which is logarithmically close to the initial value by accepting only values which are smaller than ten times the initial function value and bigger than one tenth of it. This is demonstrated in fig. A.2, but to finally compute the variance $\sigma^2 = 2N^{-1}O_2$ we take all N estimations of $\hat{\beta}$ into account.

The variance according to all estimations of $\hat{\beta}$ of one algorithm on data set can be used to determine its estimation accuracy, because we can assume the inharmonicity coefficient of one piano to roughly follow our tessitura model for β. We can further state, that the instruments original β coefficient is equal for all recordings of the same note of this instrument and constant along time. Therefore, each instrument exhibits a certain variance due to slight tuning

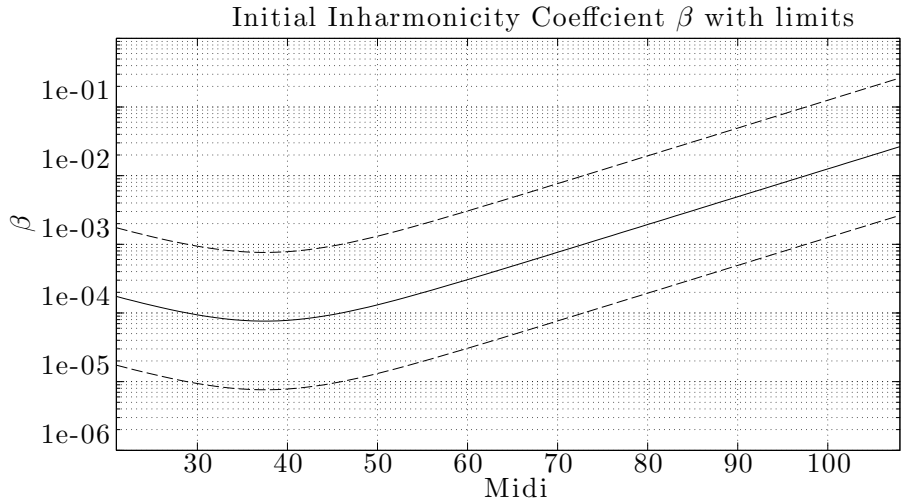


Figure A.2: The initial model $\beta_\phi(m)$ (solid) and limits (dashed) for adaptation

errors of its inharmonicity. This variance is unknown and reflects the lower boundary for every estimation algorithm. As all our algorithms estimate either a single inharmonicity value per frame of each sound sample (MAT, ICF, NMF) or a single value per sound sample (CG), the more these values are varying, the less accurate this algorithm has to be. Therefore, we can use the overall variance of the inharmonicity estimations of one algorithm for one data set to determine its accuracy performance.

Evaluation on artificial data

The sounds have been generated by additive synthesis using eq. (3.11) to generate the partials frequencies with the β coefficients taken from the initial tessitura model $\beta_\phi(m)$ for each corresponding fundamental frequency, a decaying spectral envelope as well as a simple Attack-Release temporal envelope. The sounds do not contain additional noise.

We estimated the β values with all methods for all synthesized sounds and measured their deviations from the original values used for synthesis. Fig. A.3 shows the resulting relative errors as percentage of the original β value denoted $\bar{\beta}$.

As can be seen in fig. A.3 the MAT, NMF and CG methods outperform the ICF method with relative errors below 0.1% until MIDI index 86 (D6). Above that index, only the NMF and CG method stay below 0.1% or even drop further down.

The estimated tessitura models of all algorithms for the artificial set are shown in fig. A.4 and their resulting overall variance of the estimated $\hat{\beta}$ is depicted in fig. A.5. The extremely high variance of the results for the MAT and ICF is especially caused by the low estimation accuracy for high pitches (MIDI index values above 85). The increased variance of the NMF method is due to estimation errors around MIDI index 35 at which the inharmonicity coefficient reaches its absolute minimum. Hence, our proposed CG outperforms

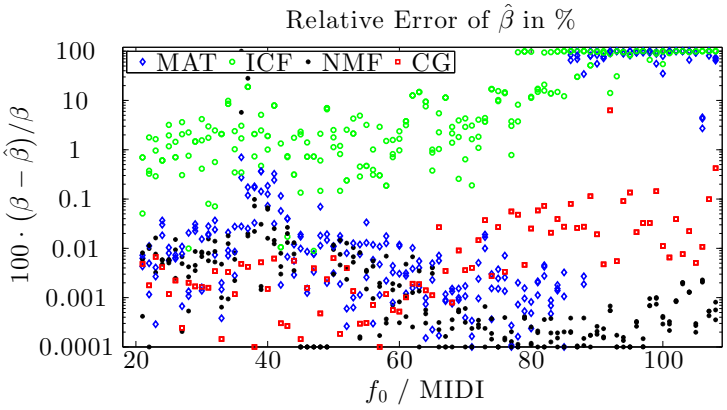


Figure A.3: Error in estimation of β given as percentage.

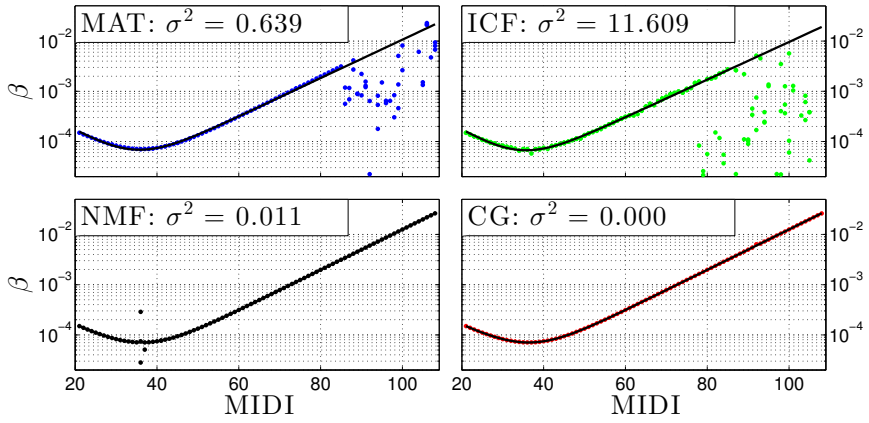


Figure A.4: Estimated $\hat{\beta}$ for the artificial data set.

the MAT and ICF methods significantly in terms of overall variance as it almost never shows an accuracy error of more than 0.1%.

Evaluation on recorded data

The RWC piano library contains recordings of 3 different grand pianos. Each piano has been recorded for all pitches in 3 different intensity levels (*pp*, *mf* and *ff*). The piano set of the IRCAM Solo instruments library also contains recordings for all pitches but with up to 8 intensity levels per pitch.

It can be seen in the figures A.6 to A.9, that the NMF as well as our proposed CG method show especially in the upper pitch range significantly less noise in the estimation of $\hat{\beta}$ compared to the ICF and MAT methods. This seems to be caused by the adaptive noise level used by the NMF method and the peak classification used by CG for selecting reasonable partials.

Also, the use of a Kullback-Leibler-divergence with euclidean distance (NMF) and a minimum variance method (CG) for estimating β shows to be clearly superior to a heuristic grid search (ICF) or a median method (MAT). The CG

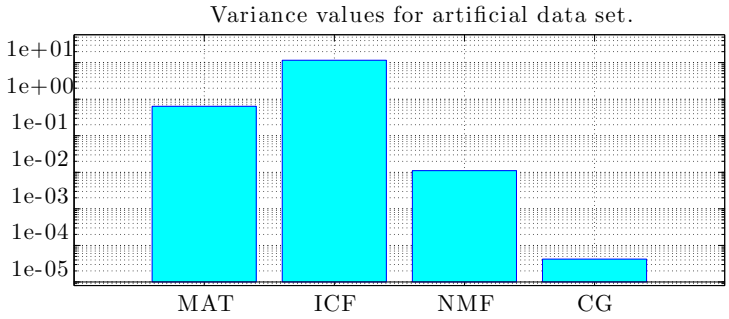


Figure A.5: Variance of measurements on artificial data.

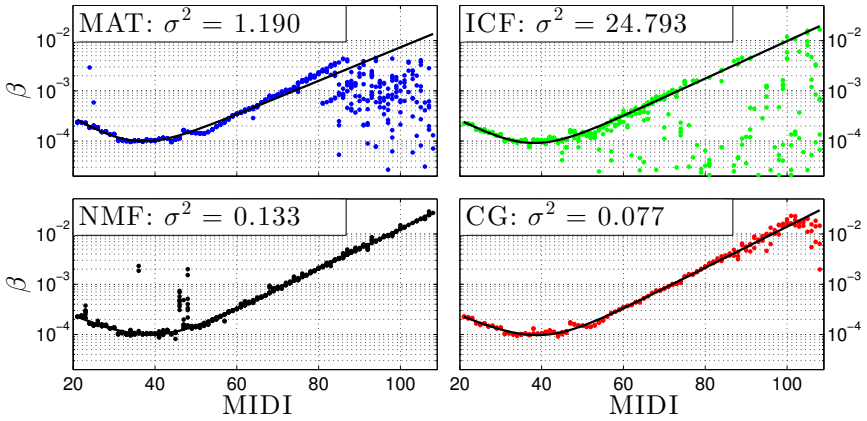


Figure A.6: Estimated $\hat{\beta}$ for RWC piano 1

method only shows a slightly higher variance for the RWC 2 data set, whereas it outperforms NMF on all other data sets up to a factor of 20 for the RWC 3 data set.

The overall estimation performance is demonstrated in fig. A.10. Here, the averaged variance values from all data sets are shown as bars, whereas their minimum and maximum values are given as error bars. It can be observed, that the CG method has the least variance closely followed by the NMF method. The ICF method is far from being accurate, whereas the MAT method rates third.

In terms of computational performance, as shown in A.11, the MAT method is by far the fastest method, but it clearly lacks in estimation accuracy in the upper pitch range, whereas our proposed method CG outperforms NMF which showed similar estimation results as well as the ICF method.

As can be seen from the above given result, the proposed method for joint inharmonicity estimation shows that a peak selection algorithm with adaptive noise and sidelobe rejection paired with a minimum variance based parameter estimation is a suitable strategy for a robust detection of the inharmonicity coefficient and the signals fundamental frequency.

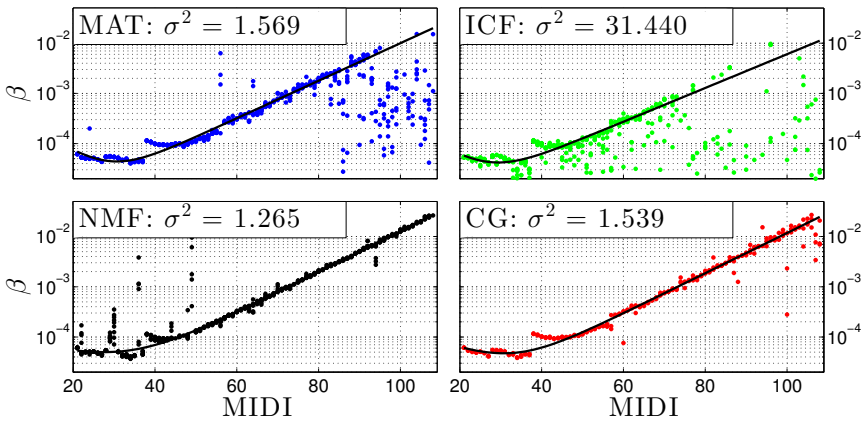


Figure A.7: Estimated $\hat{\beta}$ for RWC piano 2

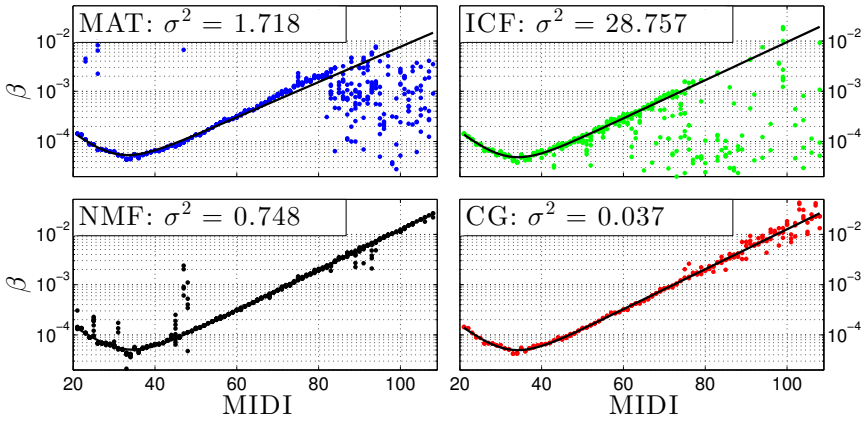


Figure A.8: Estimated $\hat{\beta}$ for RWC piano 3

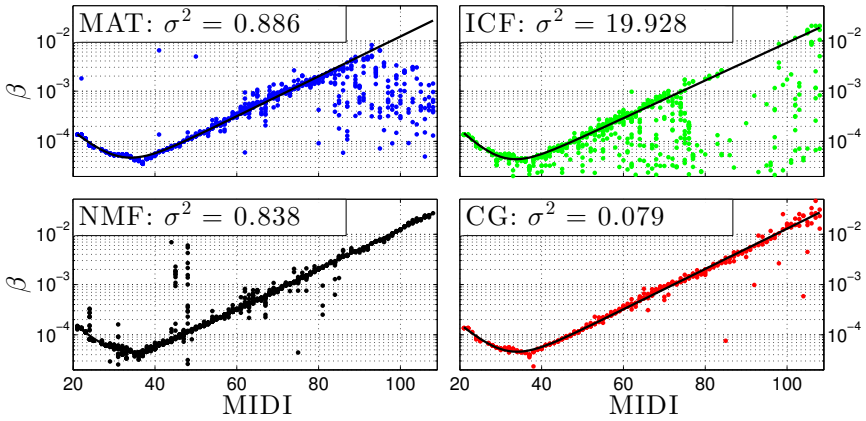


Figure A.9: Estimated $\hat{\beta}$ for IRCAM Solo Instrument piano

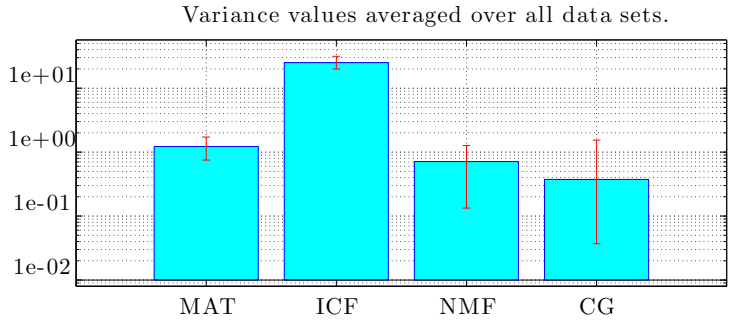


Figure A.10: Averaged variance of measurements on real world data according to the tessitura model. The error bars indicate the minimum and maximum variance values among all data sets.

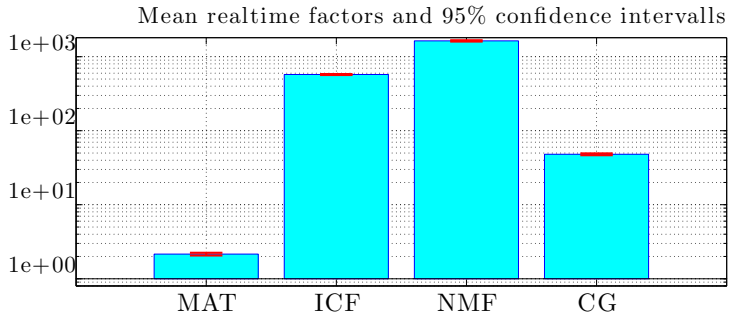


Figure A.11: Processing real-time factors for all 4 algorithms averaged for all data sets with 95% confidence intervals.