UNIVERSITAT POMPEU FABRA

Modeling Instrumental Gestures:
An Analysis/Synthesis Framework for Violin Bowing

ESTEBAN MAESTRE GÓMEZ

TESI DOCTORAL UPF/2009

Directors de la tesi

Dr. Xavier Serra i Casals
Departament de Tecnologies de la Informació i la Comunicació
Universitat Pompeu Fabra, Barcelona

Dr. Julius O. Smith III
Music Department / Electrical Engineering Department
Stanford University

Dissertation submitted to the
DEPARTAMENT DE TECNOLOGIES DE LA INFORMACIÓ I LA COMUNICACIÓ
UNIVERSITAT POMPEU FABRA
in partial fulfilment of the requirements for the degree of

DOCTOR PER LA UNIVERSITAT POMPEU FABRA

Music Technology Group
Departament de Tecnologies de la Informació i la Comunicació
Universitat Pompeu Fabra
Roc Boronat, 138
08018 Barcelona, SPAIN
http://mtg.upf.edu
http://www.upf.edu/dtecn

This work has been supported by a doctoral scholarship from Universitat Pompeu Fabra, by the Spanish R&D Program projects PROMUSIC (TIC 2003-07776-C02) and PROSEMUS (TIN2006-14932-C02), by the EU-ICT 7th Framework Programme project SAME (no. 215749), and by YAMAHA Corporation.

Part of the research presented in this dissertation has been carried out at the Center for Computer Research in Music and Acoustics (CCRMA, Stanford University), during a research stay partially funded by the Agència de Gestió d'Ajuts Universitaris i de Recerca (AGAUR) through a BE Pre-doctoral Program scholarship.

A mi hermana,
Aurora.

"The whole of science is nothing more than a refinement of everyday thinking."

Albert Einstein

Abstract

This work presents a methodology for modeling instrumental gestures in excitation-continuous musical instruments. In particular, it approaches bowing control in violin classical performance. Nearly non-intrusive sensing techniques are introduced and applied for accurately acquiring relevant timbre-related bowing control parameter signals and constructing a performance database. By defining a vocabulary of bowing parameter envelopes, the contours of bow velocity, bow pressing force, and bow-bridge distance are modeled as sequences of Bézier cubic curve segments, yielding a robust parameterization that is well suited for reconstructing original contours with significant fidelity. An analysis/synthesis statistical modeling framework is constructed from a database of parameterized contours of bowing controls, enabling a flexible mapping between score annotations and bowing parameter envelopes. The framework is used for score-based generation of synthetic bowing parameter contours through a bow planning algorithm able to reproduce possible constraints imposed by the finite length of the bow. Rendered bowing control signals are successfully applied to automatic performance by being used for driving offline violin sound generation through two of the most extended techniques: digital waveguide physical modeling, and sample-based synthesis.

Resum

Aquest treball presenta una metodologia per modelar el gest instrumental en l'interpretació amb instruments musicals d'excitació contínua. En concret, la tesi tracta el control d'arc en interpretació clàssica de violí. S'hi introdueixen tècniques de mesura que presenten baixa intrusivitat, i són aplicades per a l'adquisició de senyals de paràmetres de control d'arc relacionats amb el timbre del so, i per a la construcció d'una base de dades d'interpretació. Mitjançant la definició d'un vocabulari d'envolupants, es fan servir seqüències de corbes paramètriques de Bézier per modelar els contorns de velocitat de l'arc, força aplicada a l'arc, i distància entre l'arc i el pont del violí. Així, s'obté una parametrització que permet reconstruir els contorns originals amb robustesa i fidelitat. A partir de la parametrització dels contorns continguts a la base de dades, es construeix un model estadístic per l'anàlisi i la síntesi d'envolupants de paràmetres de control d'arc. Aquest model permet un mapeig flexible entre anotacions de partitura i envolupants. L'entorn de modelat es fa servir per generar contorns sintétics a partir d'una representació textual de la partitura, mitjançant un algorisme de planificació de l'ús d'arc capaç de reproduir les limitacions imposades per les dimensions físiques de l'arc. Els paràmetres de control sintetitzats s'utilitzen amb éxit per generar interpretacions artificials de violí fent servir dues de les técniques de síntesi de so més exteses: models físics basats en guies digitals d'ona, i síntesi basada en mostres.

Acknowledgements

Even though you might not even notice it, I acknowledge this section as the most dissapointing of my dissertation. Believe me, I could keep working on it for ages, but I would never be happy about its ability to communicate how grateful I feel to every person who helped me and my research during the last five years. Let me give it a try...

Back in 2004, Xavier Serra opened me a door to what today I know as 'Computer Music Technology', if that sounds like an appropriate way of referring to it. Although I was coming from a different area (two years before I had finished my EE Master's thesis on resonant power conversion while at Philips Research Labs in Germany), my passion for Music and my previous experience with different sorts of sequencers and basic synthesizers drove me to decide that I needed to knock at that door. Now, after five years, I am proud to say that I was right. And I smile on Xavier for handing me such opportunity.

Being part of the Music Technology Group has been an incredibly enriching experience during which I had the oppotunity to work alongside wonderful people. Shortly after I joined the group, I realized Xavier's enormous capacity for getting different people's hands and beliefs into challenging, rewarding research projects. I acknowledge his efforts in setting the foundations of such a great enterprise. Notwithstanding that, I must tribute the invaluable deed of every fellow who has been instrumental for reaching the high human standards that once made possible to formulate the MTG receipt.

From my early days at MTG, I want to express my gratitude to my first three guiding colleagues. Emilia Gómez accompanied me during my very first steps, teaching me the introductory lessons to the group's *modus operandi*. Amaury Hazan started his PhD work at the same time as I did. He was my first partner, and the first person with whom I shared more than just work. Finally, I am specially thankful to my first mentor Rafael Ramírez, whom I still have the pleasure to work with. From him I learnt about transforming data into experiments, experiments into results, and results into publications. Somehow, they all grounded my further proceeding.

After two years working with them on saxophone expressive performance modeling from audio-extracted features, I moved to the 'Voice and Audio Processing Team' (or the 'Yamaha Team', as some MTG fellows commonly refer to). The combination of people and singing voice-related research topics I found

there made the team to be special. Indeed, personalities like Jordi Bonada, Àlex Loscos, Oscar Mayor, Jordi Janer, Lars Fabig, Merlijn Blaauw, and Maarten deBoer contributed to a very dynamic atmosphere always full of joy and enthusiasm. I could use their priceless help in any issue for which I eventually needed. I claim Jordi Bonada to be my second mentor. His openness to discussing new ideas, and the vast experience he acquired on spectral modeling after many years leading the team were always a source of inspiration. And I could claim Jordi Janer to be my second partner. His company, support and and availability have been fundamental for my advancement and comfort. Altogether, this particular group of people provided me with an environment in which I could easily combine joking and laughing with my work on saxophone sample-based sound synthesis and singing voice articulation modeling. Àlex and Oscar were always good at pulling out a worthy hee-haw from me. Lars's memorable funky tunes used to get everyone to the beat, and the always friendly arguments about rivalry between Real Madrid F.C. and F.C. Barcelona that I had with Maarten and Jordi Bonada made me enjoy even more their company.

A second stint with the 'Yamaha Team' started when Alfonso Pérez and Enric Guaus joined as co-workers in the two year-long 'Violin Project', another challenging, Yamaha-funded research project. Together with Jordi Bonada and Merlijn Blaauw, they joined me in what I could name the 'Violin Project Team'. All four importantly contributed to the work I am presenting in this dissertation. Without their collaboration, it would have been impossible to achieve it. Each of them was a key component of the team, and in my acknowledgements I would like to mention at least one of the major contributions from each one. I thank Jordi for his involvement in measurements, database construction, and sample-based sound synthesis algorithms; Merlijn for his remarkable work on the recording plug-in, sample database managing tools, and sample-based synthesizer structure; Enric for his commitment in overcoming the force measurement difficulties; and Alfonso for the score creation and the gesture-based violin timbre modeling. To finalize, I consider Alfonso to be the third of my partners, working alongside me during the latter stages of my MTG experience.

Along my path, some other people from MTG gained my respect in some special way. Only paired by his colossal experiece, the availability and willingness to help always shown by Perfecto Herrera made him becoming a reference in the group, and I want to thank him for every little advice and comment he has made regarding my work. I enjoyed very much the company and always useful, practical advice of Àlex Loscos and Pedro Cano. I would not want to dismiss from my mind the good times (both at work and off-work) that I had with Enric Guaus and Fernando Villavicencio. Likewise, the appealing companionship of Òscar Celma was always there wlthough we never worked together. In some way, he has also been some sort of a partner for me.

Towards the end of my research, I had the oppotunity of carrying out important research at the Center for Computer Research in Music and Acoustics, Stanford University. My stay at CCRMA was among the best periods of my PhD, and I am very happy to have enjoyed such a lucky chance. There, my experience

as a human being and as a researcher got greatly broadened, not just because of having the possibility of working or sharing interests with people like Jonathan Abel, Chris Chafe or Julius Smith, but also due to the inmense excitement and predisposition that one could breath in that fantastic space. Sasha Leitman and Carr Wilkerson are two marvelous persons who played an crucial role in my integration. I am proud to say that they are among my closest CCRMA fellows. A very welcoming Chris Chafe and his interest in my research significantly helped to augment the spiritness of my work at my arrival. Not to forget is the enormous benevolence and humanity of Jonathan Abel, who was always available and ready to provide help by means of enriching, inspiring advice.

I want to explicitly acknowledge the motivating enthusiasm and involvement that Julius always demonstrated towards my research. Somehow, he managed to extract the deepest of my beliefs while boosting my aspirations. He gave to my work the value that sometimes I missed from myself, especially during the latter stages of this journey. In that sense, I think that I am very lucky to have had such an experienced, human personality as director. The combination of his wisdom and fascination together with the institutional support I always received from Xavier formed an enviable, once-in-a-lifetime direction duo for a dissertation that I feel very proud of.

Talking about the manuscript itself, I am indebted to Julius Smith and Marcelo M. Wanderley for their inestimable help in reviewing it. Their feedback was critical for the improvement of its readability and effectiveness.

There are three more persons from Universitat Pompeu Fabra that deserve well to receive my gratitude. All three were very special partners for me. First, I want to express my recognition towards my friend Carlos Spa, who has been the closest and most sincere of my colleagues. Beyond research, he has helped me in many different ways and contributed to my development as a person. I honestly wish the best for him, as I honestly wish the best for Ping Xiao. Apart from some basics of Mandarin that I already forgot, she taught me important lessons about health and other basic human principles. Her support and disposition will always remain in my memories. Finally, I reckon Anna Sfairopoulou as the best teaching company I could ever have. My gratitude goes to her for always keeping a positive attitude, even when things were not looking that good.

My life in San Francisco would not have been the same if I had not met my roommates and friends Yorgos Sofianatos, Argyris Zymnis, and Ross Dunkel. I feel very happy about the friendship I started when moving to that amazing graduate student appartment located at the intersection of 21st and York streets. They all three significantly contributed to my optimism and prosperity while there. I want to make an special mention to Argyris, who was always willing to spend time on discussing and giving help whenever I had troubles with algebra and related issues. Likewise, I appreciate very much having met Natalie Schrik, whose good heart always raised my self-esteem.

Through my walk, many other people that I am not mentioning here stayed besides me in some way or another. I could fill several pages by dedicating some words to every of them, but I will just stick to a few. The three members of my

closest family supported me like nobody did. Although they possibly are the ones who got less insight about what the dissertation was actually about, they delivered themselves to it like if it was their own work. The older I get, the more I respect my father Juan and my mother Araceli. Without their wisdom, this would not have been possible. And my most heartfelt appreciation goes to my sister Aurora. She has always been there, cheering me up and providing me with invaluable help in clearing up my ideas about life. Having said that, I can tell you that a lot of friends also suffered from my 'PhD-ness'. Thanks to all of them. Finally, I would like to express gratitude towards my roommates and friends Ana and Bàrbara for their company and preoccupation during these months of being shut away in my room while writing this dissertation.

At this stage of my writing (this has been the section that I wrote last), I already feel the bittersweet sadness of missing most of what travelled alongside me in this journey: it is impossible to look ahead without expecting future flashbacks in the shape of sparkling melancholy.

Preface

– Mike, what do you think of the string section in this part here?

(Orchestra sound)

– Mmm... Which scene does it correspond to?

– Still out of the house. She turned around, and slowly started to approach the door.

– Can you play it again? But this time I want to watch the scene.

(Video plus orchestra sound)

– So?

– Well, I think the melody and bass lines are good, but to my understanding there should be more tension in the performance. It sounds as if your violinists got lethargic or something.

– Exactly! That's exactly what I feel.

– Did you input 'sleep' as a keyword for the suggestion?

They look at each other and, after a couple of seconds of silence, they release a laugh.

– Good one, Mike. No, I think the keywords are fine. They match those proposed in the script.

– I guess you haven't made changes to dynamics, articulations, or anything like that.

– Already checked all of them, and I have to admit that the suggestions concord pretty well with what I'd have annotated myself. You see? This crescendo here is just perfect. And these accented notes are just right. Look.

Mike approaches the screen, starts messing around with the different windows and toolbars, and goes:

– Mmm... Any special settings?

– I haven't digged much into settings yet. I just expected it to come out with a better result right away. For you it has worked no sweat all week long.

Mike keeps exploring.

– Let me see. Which performers are playing?

– You mean violinists? You didn't tell me anything about that. Can you choose them?

– Oh yeah. The software comes with different models from different performers. Mmm... Oh, you are using the default ones!

– Is that bad?

– They were created a couple of years ago for some education-oriented software tools from the same company. They play too dully for this! Let's try instead with... these ones!

– I see. Mmm... Shouldn't I have a way to somehow induce a particular intention to whatever the 'perfomer model' is in there, as opposed to changing it?

– Man, give it a break. This is a computer, not a human! How would you do that, anyway?

– Maybe with a 'virtual conductor mode' in which I'd use a 'virtual baton'?

– Yeah, right... But you would still miss the feedback from the performers.

– Mmm...

– Modeling music performance, including your 'virtual baton', has been a topic of study for many years now, and I personally believe that this software does a great job. There's still a long way to go before you'll get satisfied. In fact, perhaps you'll never be, Jim. You're too demanding!

Mike smiles at Jim, and Jim gives the smile back. Mike continues:

– Dude, I think you should go back to manually arranging compositions, and to rehearsing with 'real-world' performers. You'll feel less frustrated, and you'll probably be more confident of your music and more prepared to face final recording sessions.

– Maybe you're right, Mike. But let's get this scene finished. I'm starting to feel hungry...

The above dialog recounts on a quotidian situation showing how an instrumental music creation environment could be assisted by computer in some hypothetic future. Maybe some day, the success of instrumental sound synthesis applied to automatic performance will not be a matter of the quality of the virtual instrument (possibly a solved problem by then), but a matter of how virtual instruments are played. Obviously, modeling human performance represents a much more challenging pursuit than modeling a musical instrument, and this fact partly explains why state of the art computer research in music and acoustics has already contributed more know-how on modeling instrumental sound than on modeling instrumental playing, especially for those cases in which the human-instrument interaction is based on rich, complex energy transfers.

The subject of this work is the investigation of computational approaches for the study of instrumental gestures in music performance. The nature of instrumental gestures, understood as the physical actions applied by a musician to produce the sound conveying the musical message contained in the piece or composition being performed, is highly constrained by the sound production mechanisms of the instrument, and by the manner in which the instrument is excited. From the different types of musical instruments, those called *excitation-continuous* are often considered as to allow for a higher degree of expressivity,

in part due to the freedom that is available for the performer to continuously modulate input control parameters when navigating through the instrument's sonic space in the seek for pleasant timbre nuances.

Understanding input control patterns executed by a trained performer when playing an excitation-continuous musical instrument represents a challenging research pursuit that started to receive attention only during the past years. Availability of acquisition devices and techniques devoted to the measurement or estimation of musical instrument input controls is making real performance data to be accessible, bringing the opportunity of opening new paths for embarking the study of music performance from a closer, richer perspective. Performance analysis and modeling, instrumental sound synthesis, or musical pedagogy are amongst the research fields to get benefitted from the the insight provided by quantitatively analysing instrumental gesture control of musical sound.

This disseration proposes a systematic approach to the acquisition, analysis, modeling, and synthesis of instrumental gesture patterns in bowed-string instruments. More concretely, it deals with bowing parameters in violin classical performance. Only comparable to the singing voice, violin is regarded as one of the most expressive musical instruments, offering a prime opportunity for the investigation of instrumental control also because of a relative accessibility for the acquisition of input control parameter signals.

Although a number of applications arise from the possibilities brought by the instrumental control modeling framework proposed in this dissertation, violin sound synthesis from an input score is chosen as an assessment of the validity and perspectives of quantitatively analysing, modeling and synthesizing instrumental control parameters. In general, excitation-continuous instrumental control patterns are not explicitely represented in current sound synthesis paradigms when oriented towards automatic performance. Both physical model-based and sample-based sound synthesis approaches may benefit from a flexible and accurate instrumental control model, enabling the improvement of naturalness and realism of synthetic sound. In this work, gesture models obtained from real data drive the rendering of bowing parameter signals. Artificial bowing controls are successfully used for synthesizing sound from an input score, delivering a significant improvement in the perceived quality of obtained sound. In regard to physical modeling, the availability of input controls for driving a basic bowed-string model based on digital waveguides boosts the naturalness and realism of synthetic sound to an unprecedented level. For the case of sample-based synthesis, embedding sample retrieval and transformation functions that are specific and meaningful to instrumental control remarkably enhances timbre continuity, proving to help in overcoming one of the major drawbacks of concatenative synthesis frameworks.

This work presents a comprehensive study of bowing control in violin performance. The proposed methodologies aim at demonstrating the crucial importance of modeling instrumental control when carrying out computer research in music performance. Formulating an effective vocabulary for bowed-string

performance practice, estimating the use of this vocabulary in recorded playing, and addressing the problem of generating a performance from a written score represent valued new capabilities that will hopefully serve as a foundation for many more to come.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Instrumental gestures in music performance

The process of music performance can be roughly decomposed into three main elements: the composition, the performer, and the musical instrument. The composition can be understood as the basic musical creation (or 'make-up' of the musical piece) conveying a musical idea. In the need for exchanging or transferring such musical ideas, a well defined, structured representation arises as a necessity for enabling common understanding. Such representation, implemented by a time-sequence of discrete events (having the note as basic unity) is what is known as a musical score. The musical instrument is regarded as the physical object that allows executing the composition. By means of applying a variety of physical actions to the instrument, it produces sound that can interpreted as 'musical' as long as there is an intention (the composition itself) explaining those sets of physical actions and their characteristics and arrangements in time. Finally, the composition is interpreted and converted into musical sound by the performer, who is in charge of applying the physical actions to the musical instrument.

None of these three elements is independent from each other. Yet the score, which could have more sense as an isolated element, might still be constrained by the musical instrument or performance style it is intended for. Either, the performer and the instrument are not easily treated as separate elements when considering the complex interactions taking place during performance. Indeed, there exists an intimate relationship between the three elements: the composer composes for an instrument by expecting the performer to understand the musical message and execute a variety of physical actions or gestures for the instrument to produce the desired sound, i.e., the composer is assuming a 'function' from the performer.

Trained musicians are able to read an interpret a composition or musical piece that they are given in the form of a musical score. This musical document may get very different meanings depending on how it is performed, i.e., how

emotional content is converted into musical sound by the performer. In fact, music performance as the act of interpreting, structuring and physically realizing a composition is a complex human activity with many facets: physical, acoustic, physiological, psychological, social, artistic, etc. (Gabrielsson, 1999). Far from willing to enter a discussion on how each of these factors come into play, it is commonly acknowledged that there is an important part of expression or meaning already conveyed in the musical piece to be performed (e.g., the melody itself), and another -also important- part introduced by the performer, who executes a set of 'gestures' (e.g., the bowing patterns indicated by the composer) by the praxis habits for that particular instrument and her/his musical training, but by conveying particular meaning in the way she/he is producing the musical sound. Therefore, it results of crucial importance the form in which the performer acts over the instrument, i.e., the nature of the gestures and physical actions taking place during performance.

1.1.1 Defining *instrumental gesture*

Biologists define *gesture*, broadly stating, "the notion of gesture is to embrace all kinds of instances where an individual engages in movements whose communicative intent is paramount, manifest, and openly acknowledged" (Nespoulous et al., 1986). In its simplest sense, the term *gesture* has been mainly treated as to the way human beings move their body to communicate, although gestures are used for everything from pointing at a person to get their attention to conveying information about space and temporal characteristics (Kendon, 1990). Evidence indicates that, for instance, gesturing does not simply embellish spoken language, but is part of the language generation process (McNeill & Levy, 1982).

Although sometimes it results hard to differentiate them, musicians performing a particular piece make use of two types of gestures, according to an act-symbol dichotomy (Nespoulous et al., 1986). This dichotomy refers to the notion that some gestures are pure actions, while others are intended as symbols. For instance, an action gesture occurs when a person chops wood or counts money, while a symbolic gesture occurs when a person makes the 'okay' sign or puts their thumb out to hitch-hike. Examples of each one of these senses are, for an action gesture, purely playing a pair of notes present in the score; and for a symbol gesture, playing such pair of notes by means of a extreme *staccato* articulation. Naturally, some action gestures can also be interpreted as symbols (semiogenesis), as illustrated in a spy novel, when an agent carrying an object in one hand has important meaning; or in a musical performance, the intended articulation with which two or more notes are played.

Among the musical gestures, one can find composition gestures and performance gestures. Composition gestures, while not physical, are already at the music creation step, constituting the composition itself. They can be understood as isolated notes, groups of notes, or annotations (e.g., crescendo, legato) referring to the manner in which some notes are to be played. They convey a particular musical message and are expressed explicitly in the score.

Conversely, performance gestures may not be explicit or quantified, and lay on the physical domain. They are understood as the voluntary (or constrained) gestures produced by the performer during the transformation of the score into musical sound.

Within performance gestures, Wanderley & Depalle (2004) propose a classification of performance gestures into *ancillary gestures* and *instrumental gestures*. They refer to instrumental gestures as those involved in the sound production process (e.g., moving the bow in violin performance), while ancillary gestures are considered to be produced by the performer as additional body movements, not involved directly in the sound production mechanisms, but linked to the performance and being able to communicate some emotional content, or even to slightly modify the sound properties (e.g., body movements of pianists). Cadoz (1988) propose a definition of instrumental gestures as follows:

> *"Instrumental gesture will be considered as a communication modality specific to the gestural channel, complementary to free gestures, and characterized by the following: it is applied to a material object and there is physical interaction with the object; within this interaction, specific physical phenomena are produced whose forms and dynamical evolution can be controlled by the one applying the gesture; these phenomena can convey communicational messages (information)".*

He considers this transfer of information as a specificity of instrumental gestures, when compared to other ergodic or ancillary gestures. He reminds that *"not all gestures have the same function on an instrumental object"*. Therefore, he proposes an instrumental gesture typology (see Figure 1.1) according to their function:

- **Excitation gestures**: they convey the energy that will be found in the sonic result (e.g., blowing in wind instruments).
- **Modulation gestures**: used to modify the properties of the instrument but in which energy does not participate directly in the sonic result. These can be divided in two groups:
  - **Parametric modulation gestures**: those changing continuously a parameter (e.g., variation of bow pressure in violin performance) .
  - **Structural modulation gestures**: those modifying the structure of the object (e.g., opening or closing a hole in a flute).
- **Selection gestures**: used to perform a choice among different but equivalent structures to be used during a performance (e.g., the direction of bowing in violin performance). There is neither energy transfer (in the sense of excitation gestures) nor an object modification (as in the case of structural modulation gestures).

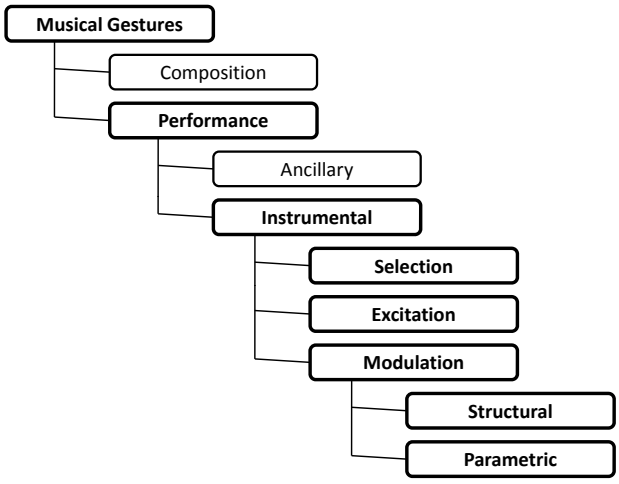


Figure 1.1: Classification of musical gestures, and classification of instrumental gestures according their function as proposed by Cadoz (1988)

Usually, none of these gestures come alone during performance, and it is often difficult to distinguish a single functionality of a gesture. An example of instrumental gesture combining excitation and modulation functionalities happens for instance during a vibrato in singing voice performance. Another example of mixed functionality could be take place during violin bowing, which can be seen as conveying a combination of excitation, selection, and modulation functionalities, because the performer bows in order to excite the body, chooses which direction to bow, and continuously changes the bowing speed.

The generality of this classification does not prevent instrumental gestures from being constrained by several factors. Some annotations present in the score help the performer to execute the musical piece and guides her in choosing some gestures or modulating some parameters, as it is for instance the case of a crescendo annotation, which makes the musician to progressively increae the airflow in saxophone performance. From the information not present explicitly in the score, instrumental gestures are constrained by physical limitations, instrument or performance habits, and performer wishes.

Some physical limitations force instrumental gestures to be executed in some particular form or at some particular time, like for instance changing the direction of the bow as it reaches the maximum displacement, or the pronunciation of some consonants in singing voice performance, which force a structural change that modifies or truncates perceptual parameters as for instance fundamental frequency. Other important factors are more related to the instrument being played or the performance praxis. This refers for instance to fingering, the bow releasing the string between successive bouncing strokes, or the bow stopping on the string between staccato notes. The third main group of

factors or constraints includes all variations and deviations that the performer introduces due to her own wishes (often more related to expression), like for instance vibrato or tremolo, or a particular style in articulating slurred notes.

1.1.2 Considerations on the instruments' sound production mechanisms

Depending on the complexity of the performer-instrument interaction taking place in the sound production process, approaching the study of instrumental gestures may bring different challenges when it comes to representing the gesture-sound relationship with certain completeness. Based on the the sound production mechanisms that characterize the instrument, a basic classification makes a major distinction into two classes of instruments: *excitation-instantaneous* musical instruments and *excitation-continuous* musical instruments.

In the case of excitation-instantaneous musical instruments, the performer excites the instrument mostly by means of instantaneous actions in the shape of impulsive hits or plucks, producing different sounds by the changing characteristics of the impulsive actions and the conditions in which they are produced. In general terms, this makes the analysis and modeling of instrumental gestures to become easier, while the gesture-sound relationship appears to be more accessible for pursuing research. Examples of this kind of instruments are, for instance, drums, piano, etc.

Conversely, in excitation-continuous musical instruments, the performer produces sound by exciting the instrument continuously during performance, and she achieves variations of sound and a richer navigation through the instruments' sonic space by continuous modulations of the physical actions involved. This fact, apart from making some excitation-continuous instruments to be claimed as more 'difficult' to play, raises the potential need for deeper study of instrumental gestures if the gesture-sound relationship is the central topic of investigation. Indeed, this makes harder to understand and model the dynamic characteristics of instrumental gestures, being often the case of having more difficulties when pursuing computer-aided synthesis of instrumental sound from a score if the instrument into consideration falls into this category. Examples of this class of instruments are woodwind and brass instruments, bowed strings, and the singing voice, among others.

Actually, excitation-continuous musical instruments are often considered as to allow for a higher degree of expressiveness, in part due to the freedom that is available for the performer to continuously modulate input control parameters when navigating through the instrument's sonic space in the seek for pleasant timbre nuances.

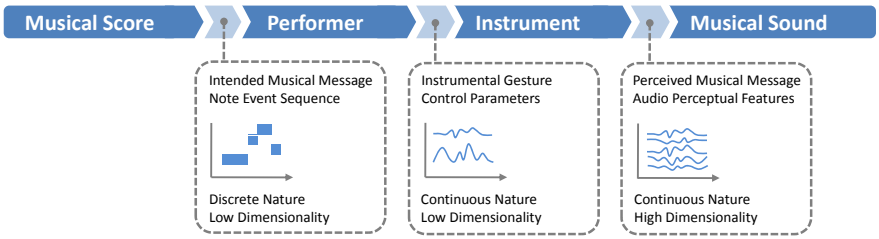


Figure 1.2: From an instrumental gesture perspective, musical score, instrumental gestures, and produced sound represent the three most accessible entities for providing valuable information on the music performance process.

1.2 Acquisition and analysis of instrumental gestures

The process of music performance offers opportunities for pursuing research on instrumental gestures when investigated from a computational approach based on data observation and analysis. Figure 1.2 shows the principal elements of music performance according to the main proposition made in Section 1.1. The intended musical message is represented in a written score containing an ordered sequence of note events and different annotations. While the information appearing is of discrete nature and low dimensionality (it contains only a reduced quantity of explicit information that is limited to that commonly agreed for supporting musical message and performance instruction exchange), it conveys a significant amount of instrumental gesture -related information that is relevant to instrumental gestures which appears implicitly embedded. As pointed out before, the set of annotations and instructions appearing in a score represent the composer's expectation of the performer's function for executing the gestures or physical actions needed for producing a desired musical sound. Hence, in the analysis process it is important to keep an 'alignment' (both logical and temporal) of the musical score to the rest of sources of information that can be accessed (e.g., gestures or sound).

The next step in the process provides more explicit evidences of instrumental gestures than the musical score per se. The trained musician reads and interprets the discrete information in the score and 'converts' it into a set of physical actions of continuous nature. Such physical actions (instrumental gestures) respond to a learnt set of patterns also expected from the composer for the instrument to produce the sound conveying the intended musical message. In the domain of control parameters, the representation of such habits (they are strongly linked to the musical score) can still be considered of a low dimensionality (the number of input controls to a musical instrument is relatively low), but of continuous nature. This fact makes it more challenging not only

because of the potential difficulties of systematically approach the analysis and modeling of continuous-nature signals, but also due to the fact that, at this level, an enriched and flexible representation of a number of continuous nature 'signals' should be able to provide means for representing higher-level aspects of musical performance which go beyond the restricted, explicit information that appears in the score.

As a result of the modulations of the instrument's control parameters (derived from the execution of the physical actions by the performer), a further point of access to instrumental control-related information is the sound produced by the instrument. Even though the time-varying magnitude of sound pressure at a point in the space can be seen as a one-dimensional signal, higher levels of perception-related descriptions linked to listener habits have traditionally dealt with spectral-domain approaches to the analysis of musical sound. The representations on which those analyses rely present a higher complexity due to the large amount of continuous-nature parameters needed for a complete representation of sound. Contrary to what should happen when studying score-gesture relationships, research on gesture-sound correspondences (that is, modeling the instrument itself) must give special attention to the information gathered from this last domain and less to the musical score.

1.2.1 On pursuing a study of score- instrumental gesture relationship

From the previous discussion, it remains clear that in the study of instrumental gestures as a key pursuit for abstracting a clear representation of the performer as the mediator between the musical score and the instrument, the observation and analysis of physical actions appears to be a more direct approach than the analysis of spectral-domain representations of produced sound.

Putting the focus on the performer, the investigation of means for constructing a model able to represent the complex relationships between the inputs to the performer (musical score) and the outputs delivered (instrumental controls through physical actions) can be approached, in general, by mostly paying attention to these two domains and not to the sound domain. In fact, a satisfying compromise between completeness and low complexity is difficult to reach when dealing with representation and analysis of audio perceptual attributes as related to the musical score.

If the interest of the pursuit resides on the study of instrumental gestures from a score perspective, why should one not attend to physical actions if they are available? Even though a musician learns and integrates playing habits or patterns also by listening to produced sound, the weakness of modeling sound perception may lead to incompleteness when trying to overcome the high complexity derived from spectral representations. This would bring an additional difficulty, as the problem would indeed become a two-stage question: modeling the relationship between the score and the perceived sound must assume a reliable and concise knowledge of the timbric behavior of the instrument as re-

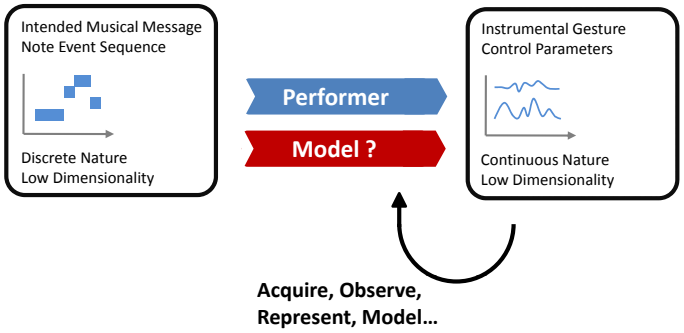


Figure 1.3: Pursuing a model that represents how the performer translate score events into instrumental gestures implies to acquire, observe, robustly represent and model the temporal contours of instrumental gesture parameters from a score perspective.

lated to instrumental gestures. Certainly, acquired understanding of the physics of instrumental sound production (Fletcher & Rossing, 1998) should provide means for approaching such a challenge, but the opportunity of avoiding the direct inclusion of sound perception into the score-gesture 'equation' secures a more affordable research.

The study of instrumental gestures in relation to the performed score could be seen as an approach to *modeling* the performer, as it is illustrated in Figure 1.3. Assuming the availability (i.e., access to measurements) of instrumental control parameter signals (when it results possible), the task involves several research challenges.

First, meaningfully observing instrumental gestures involves the acquisition of control parameters that are relevant to sound production. This implies two different sub-problems: identifying the relevant control parameters, and effectively measuring them. While the former is to be helped from studying the instrument's sound production mechanisms, the latter becomes a matter of design and implementation of acquisition devices and methodologies that provide accurate and reliable data while minimizing intrusion to the performance. A comprehensive overview of acquisition methodologies, mostly in the context of real-time control of sound synthesis and constructing musical controllers can be found in the works by Wanderley & Depalle (2004); Miranda & Wanderley (2006); Wanderley (2001).

A second problem resides on how to efficiently represent and quantify acquired instrumental gesture parameter signals. Although the reduced complexity (low dimensionality) of instrumental control eases the problem (as compared to the case of spectral representation of sound), a convenient 'coding' of the temporal evolution of instrumental gesture parameters arises as a crucial step. Such quantitative representation must be efficient, flexible, and robust enough

for setting the basis for posterior analysis and modeling of patterns involved in performance. Moreover, since instrumental control is to be modeled from the perspective of the musical score (the actual pursuit is to model how the performer translates score symbols and events into the continuum of sound-producing physical actions), a strong connection to the score must be kept when devising a representation scheme.

Finally, applying pertinent analysis to obtained representations of instrumental gesture parameter signals stands as the ultimate fundamental task in the search for a model of instrumental gestures. Estimating quantitative parametrizations of how the score and the instrumental gestures appear related in music performance becomes first a matter of structuring the representations of acquired data in two different spaces: a more *symbolic space* defined by the (limited) possibilities available to the composer (i.e., score events and annotations), and a more *continuum space* of quantitative descriptions of instrumental control parameter signals. From there, data observation or the use of appropriate analysis/modeling techniques is to reveal how one space maps to each other, i.e., how the musical score and instrumental gestures are related.

When looking into the literature, it results hard to find in-depth, data-driven studies of instrumental gesture from a score perspective (i.e., trying to explicitly model the performer), especially for the case of excitation-continuous musical instruments. Early works by Cadoz (1988); Cadoz & Ramstein (1990) pushed the formalisms and the relevance of the topic, giving definitions (see Section 1.1.1) and envisaging analysis approaches. Ramstein (1991) followed his lines, and his dissertation (dealing with piano performance) can be considered a pioneer work in treating the acquisition, analysis, and representation of instrumental gestures as a whole. Focusing more on acquisition techniques and control of sound synthesis, Wanderley (2001) continued later the research line started by his colleagues, making significant steps since the early days of gesture-controlled musical sound, when the idea of explicitly relating acquired motion to sound properties was made possible by the Theremin or Matthews' radiobaton (Mathews, 1989).

Since then, research efforts and enthusiasm has been put into computational approaches to the observation, understanding and modeling of instrumental gestures from real performance. Violin, as an excitation-continuous musical instrument, has received special attention (Young, 2008; Rasamimanana et al., 2006; Rasamimanana & Bevilacqua, 2008; Rasamimanana et al., 2009; Demoucron & Caussé, 2007; Demoucron, 2008). There are two reasons behind that. First, violin (and the other bowed-string instruments) is regarded as one of the most expressive musical instruments (only comparable to the singing voice); and second, the technique of violin playing (e.g., bowing technique) offers a unique opportunity for gathering instrumental gestures (as compared, for example, to the singing voice).

1.2.2 Instrumental gesture acquisition domains

Notwithstanding the advantages of directly attending to actual instrumental control signals instead of embarking the perceptual analysis of produced sound and try to relate that to the score, difficulties for observing physical actions in excitation-continuous musical instrument performance has lead to approach the acquisition of instrumental gestures from two different domains: the domain of physical actions and the domain of audio analysis.

Whenever the acquisition of instrumental gesture parameters brings obstacles that are difficult to overcome while keeping the measurement setup at a low intrusiveness, knowledge about the instrument's underlying physical phenomena during sound production (ant their effects on the perceptual attributes of the sonic result) is used for indirectly inducing the physical actions carried out by the musician during performance. Thus, still aligned towards the aim of 'measuring' instrumental gestures in order to pursue a study of how those are mapped to the score, the *indirect acquisition*, as it is referred to by Wanderley & Depalle (2004), arises as an alternative approach.

- **Indirect acquisition: audio analysis domain**

  Given the experience gained through years of research in the acoustical properties of musical instruments, the knowledge about audio analysis techniques, and the simplicity of acquiring produced sound (e.g., by using a microphone or pickup), indirect acquisition of instrumental gestures has traditionally received more attention. Several works, for instance, have approached the acquisition of the plucking point in guitar playing. Various techniques have been proposed, either in the spectral domain (Orio, 1999; Traube & Smith, 2001; Traube et al., 2003) or in the time domain (Penttinen & Välimäki, 2004). Wind instruments have also been object of research, especially the clarinet (Egozy, 1995; Smyth & Abel, 2009).

  While proving to be useful for obtaining certain instrumental gesture parameters in some particular cases, the success of these techniques in providing complete, reliable estimations of the physical actions used by the performer to control the timbre of the instrument falls upon overcoming a serious inconvenient: a surjection in the 'function' that transforms the instrument's input control parameters into those timbre properties preceived from produced sound (i.e., a surjection in the instrument's function). In other words, several points in the instrument's control parameter space may have a unique point in the instrument's timbre space.

- **Direct acquisition: physical action domain**

  Although ideally it provides the best possible signals, and no induction needs to be carried as a preprocessing step, the high difficulty of gathering information without intruding the performance makes the direct acquisition of instrumental gestures a less explored field. As already pointed out, among excitation-continuous instruments, the bowed-string family

has been the focus of a number of works that lead to successful capturing techniques, due to the characteristics of the instrument-performer interaction (mostly attending to bowing). The reader is referred to Section 1.4.4 for a review of direct acquisition methods for violin bowing.

1.3 Instrumental gestures for off-line sound synthesis

The nature of the sound perceived from music performance is the result of a (complex) combination of a musical instrument and a performer, especially for the case of excitation-continuous musical instruments (see Figure 1.2). From an automatic performance perspective (i.e., sound synthesis from an input score), the importance of instrumental gesture modeling and representation must be raised up to that of the performer in real music playing. Indeed, in spite of the various methods available to synthesize sound, the ultimate musical naturalness or expression of those sounds still falls upon the capture and modelling of (instrumental) gestures used for control and performance (Cadoz & Ramstein, 1990; Rovan et al., 1997; Demoucron, 2008).

1.3.1 Sound synthesis techniques: physical models versus sample-based

It represents a difficult task to define a clear taxonomy to classify existing instrumental sound synthesis techniques. Smith (2002b) divides synthesis into four categories: *abstract algorithms*, *processing of recorded samples*, *spectral models*, and *physical models*. Techniques falling into the *abstract algorithms* category, as they are understood there, are not of interest when applied to synthesize the sound of an existing instrument, because it is difficult to produce musically pleasing sounds by exploring the parameters of a mathematical expression. The spectral modeling category can be seen as an evolution of signal processing techniques applied to recorded samples, because the best way to improve transformations of recordings is to understand their effect on human hearing, and the closest way to the human hearing for representing the sound is by means of a spectral representation. By considering spectral models and processing of recorded samples to fall into the category of sample-based synthesis, one might be left out with two main categories of instrumental sound synthesis techniques: physical models and sample-base synthesis.

Physical model -based synthesis is founded on mathematical models or abstractions that describe the physical phenomena of instrumental sound production. Physical models rely on the production mechanisms of sound and not on the perception of sound. Since they make assumptions about the instrument they model, the estimation of the model parameters cannot be completely automated, and at least the model structure has to be determined 'in advance'. The model structure already describes the main features of the instrument, so only small numbers of parameters are needed. By appropriately modifying

some of these parameters (e.g., input controls), perceptually meaningful results can be obtained. For example, for the case of bowed-string physical models, one of the input controls is the bow force, rather than the loudness of a single partial. In general, only one parameter set is required, and the different playing styles (according to the musician controls) will be the result of feeding it with appropriate input parameters. As physical models describe the physical structure, the interaction of the different model parts are already taken into account. Several abstractions and mathematical formulations have been used for computationally representing sound production mechanisms, among which two main groups could be highlighted: vibrating mass-spring networks and digital waveguides. For a detailed overview of physical model-based synthesis techniques, the reader is referred to the works by Smith (2002a) and Tolonen et al. (1998). A clear advantage of physical modeling synthesis is the fact that the input controls are directly related to the physical actions or instrumental gestures (they are actually a copy), so that they do not need, in general, to be 'induced' from an analysis of the audio stream coming out from the instrument during performance. Nevertheless, the unavailability of those input controls represents a major drawback, and physical models traditionally suffered from certain discredit when applied in automatic performance contexts.

Sample-based sound synthesis has been traditionally understood as based on playback and transformations of recorded samples in the time domain, as an evolution of tape-based 'musique concréte'. Understood as such, the level of realism and sound quality of individual samples is the highest possible, because there is no synthetic sound produced but recordings. The problems appear when the samples are played in sequence: transitions and nuances not stored in the database will be missing. Recently, the term 'sample-based synthesis' acquired a broader meaning: spectral-domain sound modeling techniques, as for instance phase vocoder (Flanagan & Golden, 1966; Puckette, 1995), sinusoidal models (McAulay & Quatieri, 1986), or extended sinusoidal models (Serra & Smith, 1990), have brought transformation possibilities that make possible to shape certain perceptual attributes (e.g., pitch, duration, etc.) of samples in a database and smoothly concatenate them (Schwarz, 2000, 2004). In fact, sophisticated synthesizers using spectral representations of sounds are also called 'sample-based' because they are based on sampling sounds and storing them as sequences of spectra to which apply transformations. The advantage and success of sample-based synthesis resides in well defined analysis-transformation-resynthesis techniques, which enable high quality transformations while maintaining sound fidelity. However, limitations on the flexibility and meaning of transformations (mostly based on measuring distances of symbolic quantities at score-level) and a rather pale context representation still confine the naturalness and expressiveness perceived from synthetic sound, especially for the case of excitation-continuous instruments.

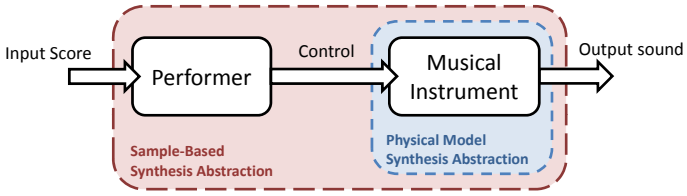


Figure 1.4: Simplified diagram of musical instrument sound production in human performance. Abstractions made both by physical modeling synthesis and by sample-based synthesis are illustrated.

1.3.2 On the need for modeling instrumental gestures

The complexity of control taking place in instrumental music performance stands out when dealing with excitation-continuous or *sustained* instruments (e.g., bowed-strings or winds), by which variations of sound are achieved with continuous modulations of the physical actions directly involved in sound production mechanisms, i.e., instrumental gestures. In contrast to the case of excitation-instantaneous musical instruments (e.g., drums or piano-like instruments), difficulties arising from quantitatively understanding the nature of control parameter profiles has kept excitation-continuous musical instrument sound synthesis from raising more success when oriented towards automatic performance, either by means of physical models or in sample-based synthesis approaches.

Figure 1.4 shows an schematic representation of musical instrument sound production by human performance. In general, a performer transforms the sequence of discrete events appearing in a musical score into the continuous-nature physical actions (or controls) needed for the instrument to produce a desired output sound. While the important role of the performer appears to be crucial in music performance, it is often the case of having off-line sound synthesizers not to feature an explicit representation of the actual control carried out by the musician.

In particular, the abstraction taking place in physical model-based synthesis puts the focus on the sound production mechanisms of the instrument. Even though the flexibility of control tends to be preserved, the lack of appropriate control parameter signals represents an important drawback when it comes to using them for automatic performance. For the case of sample-based sound synthesis, the abstraction usually includes the performer, thus a priori appearing to be more suited to automatic performance scenarios due to the pairing of score annotations and sound recordings. As already pointed out, although higher sound fidelity might be achieved (actual sound recordings are used as source

material), flexibility limitations and eventual timbre discontinuities often make it difficult to get the feeling of natural performance.

With the aim of improving naturalness of generated sound for excitation-continuous musical instrument synthesis, approaching the challenge of modeling instrumental control patterns should provide means for abstracting a proper representation of the performer. Recalling the active function of the musician, an explicit decoupling from the instrument would (1) free instrument sound modeling from control-based design constraints traditionally appearing in off-line synthesis scenarios in which the essential role of the performer is not present (e.g *NoteOn*-like events triggering sound rendering), and (2) make the flexibility and naturalness of instrumental control becoming a key component in off-line sound synthesis from an input score.

In Figure 1.5, the two sound synthesis techniques into consideration have been represented in a plane defined by two axis: sound realism and control flexibility. As claimed in Section 1.3.1, sample-based synthesizers provide in general the highest realism of sound because real performance recordings are used as source material. However, they lack of the flexibility of control that would be required for overcoming one of their major drawbacks: database coverage limitations. Even in case of being built around a carefully constructed database, it results impossible to represent the infinity of possible timbre nuances potentially produced by excitation-continuous musical instruments. Moreover, concatenating samples that were played in different gestural contexts leads to timbre discontinuities that negatively contribute to the feeling of a real and expressive performance. Given the *score-sound* pairing (instrumental gestures are absent) that in general grounds (1) database structure, (2) sample selection process, and (3) sample transformation, it appears to be tough to beat such inconveniences if no explicit presence of instrumental gesture information is introduced. Indeed, if sample annotation, search, and transformation is enriched with instrument control information, and the *score-sound* couple becomes a *score-gesture-sound* trio, synthesis control flexibility is to get significantly improved. Hence, the investigation of instrumental gestures from a score perspective arises as a crucial pursuit in the seek to significantly improve sample-based off-line sound synthesis, especially when dealing with excitation-continuous musical instrument sound.

For the case of physical models, the story is different. In principle, they represent the most promising and powerful technique when looking at flexibility of control. Differently to what happens in sample-based synthesis, the focus of physical models is put on the instrument itself. Since they model the sound production mechanisms, their input controls coincide with those of the instrument, and the flexibility of control is kept. However, their success in off-line sound synthesis applications has been traditionally constrained by the unavailability of appropriate input (instrumental) control signals, especially for the case of excitation-continuous musical instruments. Leaving apart their use in real-time scenarios in which the models are fed with signals coming from devices or *controllers* that resemble the actual control interface of the real instru-

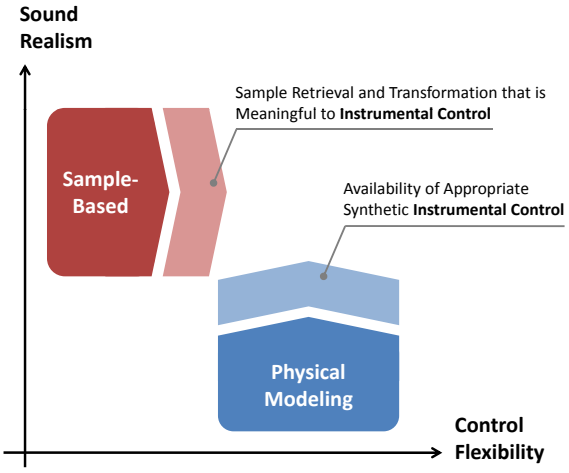


Figure 1.5: Benefits brought by synthetic instrumental control to physical modeling -based and sample-based instrumental sound synthesis in the context of off-line automatic performance.

ment, the perception of realism and naturalness when listening to generated sound mostly depends on how the input score is 'translated' into input control parameters (or synthetic instrumental gestures). Properly representing how the performer, through instrumental control, acts as a mediator between the input score and the musical instrument should boost the inherent power of physical models to deliver realistic sound when used for automatic performance.

Due to the practice-centered characteristics of the training process behind traditional musical instrument performance learning, it results difficult to quantitatively represent control patterns by attending to a *far-from-existing* human numeric gestural control tablature. Instead, only a learning-by-example approach involving (direct or indirect) acquisition of physical actions from real performances (see Figure 1.3) is to enable the construction of a quantitative model able to emulate control habits in excitation-continuous musical instrument performance.

1.4 Bowing control in violin performance

Bowed-string instruments are often regarded as to allow for a high musical expressiveness only comparable to the human voice. The space of control parameters, yet constrained, include sufficient freedom for the player to continuously modulate the timbre at a high level of detail. It is not only the notes themselves, but how the performer navigates from one note to another in the

control parameter space, which carries a large part of the expressiveness in performance (Woodhouse & Galluzzo, 2004; Schoonderwalt, 2008).

Both hands play an important role in the sound production phenomena behind violin. From a non-functional perspective (contrarily to the classification on instrumental gestures presented previously), instrumental gestures involved in violin performance can be divided into *left-hand* gestures and *right-hand* gestures. Basically, the left hand controls the length of the string that is played, and the right hand acts as the exciter, shaping the interaction between the bow hairs and the string. Such interaction leads to the characteristic vibration of the bowed-string, which is transmitted to the violin resonating body through the bridge end of the string.

In a first approximation, most of the playing techniques and expressive resources commonly available in classical violin performance are achieved through right-hand instrumental controls not involved in the selection of the string to play. These are known as *bowing* controls. During performance, the musician continuously modulates a number of parameters that are directly influencing the bow-string interaction characteristics (*bowing* parameters), with the aim of affecting the timbre properties of produced sound.

1.4.1 Basics of the the bowed string motion

The first study of string motion under bowing conditions is commonly attributed to Helmholtz, who observed, using a vibration microscope, that the motion of the string could be described by a sharp corner, traveling back and forth on the string along a parabola-shaped path (Helmholtz, 1862). The fundamental period of vibration is determined by the time it takes for the corner to make a single round trip, and is directly related to the length of the string. As a combination of the velocity of the bow and the bowing point on the string, two bow-string interaction phases happen during each vibration period. During the *sticking phase*, the string moves along with the bow at the same velocity. During the *slipping phase*, the string slips back in opposite direction. The traveling corner is responsible for keeping the transition times between the two phases, as well as for triggering slipping (release) and sticking (capture). As the string follows the motion of the bow during sticking, the amplitude of the string vibrations is mainly determined by the combination of bow velocity and the relative bow-bridge distance.

The transversal force exerted by the string on the bridge, which excites the violin body and produces the sound, is proportional to the angle of the string at the bridge. The energy losses, including internal losses in the string and at the string terminations, combined with dispersion due to stiffness (higher frequencies traveling slightly faster than lower frequencies) introduce a smoothing of the traveling corner. The net rounding of the corner is determined by a balance between this smoothing and a resharpening effect at the bow during release and capture. The effect of corner rounding and resharpening has been described by Cremer (1984). Sharpening takes place mainly during release, when changing

from sticking to slipping. If the perfectly sharp corner is replaced by a rounded corner of finite length, the string velocity no longer drops suddenly when the corner arrives at the bow. Instead, a gradual change in velocity takes place. Taking the frictional force between bow and string into account, the string is now prevented from slipping at first instance when the rounded corner arrives at the bow. The frictional force increases until the maximum static friction force is reached, and the bow eventually loses the grip of the string. The slipping phase is initiated, slightly delayed compared to the idealized Helmholtz motion. As a result of the build-up in frictional force, the rounded corner is sharpened as it passes under the bow. The balance between rounding and resharpening of the traveling corner explains the influence of bow force in playing. A higher bow force yields a higher maximum static friction, which in turn leads to a more pronounced sharpening during release. As a result, the energy of the higher partials will be boosted, leading to an increase in brilliance of the sound.

The maintenance of regular Helmholtz motion, characterized by a single slip and stick phase per fundamental period, involves two requirements on bow force: (1) during the sticking phase the bow force must be high enough to avoid premature slipping under influence of variations in friction force, and (2) the bow force must be low enough that the traveling corner can trigger release of the string when it arrives at the bow. The limits of the playable region have been formalized by Schelleng (1973).

Beyond the basics given here as a brief introduction, much research has been devoted to describe violin acoustics in general and how bow-string interaction characteristics affect produced sound in different, complex ways. These topics, however, are not addressed here. The reader is referred to the works by Helmholtz (1862); Schelleng (1973); Cremer (1984); Schumacher (1979); McIntyre & Woodhouse (1979); McIntyre et al. (1983); Woodhouse & Galluzzo (2004); Schoonderwalt (2008) for a thorough description of sound production phenomena taking place in bowed string instruments.

1.4.2 Bowing control parameters

The modulation of bowing control parameters is carefully planned and controled by the performer in order to reach the intended acoustical features of notes and phrases while respecting a number of patterns and constrains derived from the complex connection between physical actions exerted on the violin (mostly those involved in affecting bow-string interaction) and the timbre characteristics of sound. The string player needs to coordinate a number of bowing parameters continuously, and several of them may be in conflict with each other due to constrains of the following types: *physical* (bow-string interaction), *biomechanical* (the players build and level of performance technique), or *musical* (the score). Players learn and adapt early to common strategies for basic, frequent playing habits and, as experience is gained, bowing control becomes a *natural* task that might be perceived as less complex than it actually is.

The control parameters for the sound available to the player (the main bowing parameters) are basically three:

- **Bow velocity**: The velocity of the bow as imposed by the player's hand at the frog. The local velocity at the contact point with the string is not exactly the same due to small modulations in the bow hair and bending vibrations of the stick. Bow velocity sets the string amplitude together with the bow-bridge distance.
- **Bow-bridge distance**: The distance along the string between the contact point with the bow and the bridge. The bow-bridge distance sets the string amplitude in combination with the bow velocity.
- **Bow force**: The force with which the bow hair is pressed against the string at the contact point. The bow force determines the timbre (brightness) of the tone by controlling the high-frequency content in the string spectrum. In tones of normal quality (Helmholtz motion) the bow force needs to stay within a certain allowed range. The upper and lower limits for this range in bow force range increase with increasing bow velocity and decreasing bow-bridge distance.

In addition to these, three secondary bowing parameters allow the performer to facilitate the control of the three main parameters outlined before. The secondary parameters are:

- **Bow position**: The distance from the contact point with the string to the frog. The bow position does not influence the string vibrations per se, but has a profound influence on how the player organizes the bowing. The finite length of the bow hair represents one of the most important constraints in playing.
- **Bow tilt**: The rotation of the bow around the length axis. The bow is often tilted in playing in order to reduce the number of bow hairs in contact with the string. In classical violin playing, the bow is tilted with the stick towards the fingerboard. Changing the tilt angle helps the performer to modulate both the width of the hair ribbon and the pressing force applied on the string.
- **Bow inclination**: Pivoting angle of the bow relative to the strings. The inclination is mainly used to select the string played.

1.4.3 Early studies on bowing control in performance

The study of bowing gestures in string players is not an extended field of research, having its origins linked to pedagogical interests. Even though extensive literature (mostly in the area of music education and performance training) has been devoted to a rather qualitative description of bowing patterns in classical

violin playing (Galamian, 1999; Garvey & Berman, 1968; Fischer, 1997), one finds early works that opened paths for future studies on bowing control based on data acquired from real violin performance.

At the beginning of the 20th century, Hodgson published the first results on visualizations of trajectories of the bow and bowing arm using cyclegraphs (Hodgson, 1958). Using this technique, which had been developed by the manufacturing industries for time studies of workers, he could record brief bowing patterns by attaching small electrical bulbs to the bow and arm and exposing the motions on a still-film plate. The controversial results showing that bow trajectories were always curved (crooked bowing), and that the bow was seldom drawn parallel to the bridge, caused an animated pedagogical debate. Some years before Hodgson published his results, Trendelenburg had been examining string players' bow motion from a physiological point of view (Trendelenburg, 1925). Without access to measurement equipment for recording the motions of the players arms and hands, he drew sensible conclusions on different aspects of suitable bowing techniques based on his expertise as a physician. Fifty years later Askenfelt studied basic aspects of bow motion using a bow equipped with custom-made sensors for calibrated measurements of all bowing parameters except the bow angles (Askenfelt, 1986, 1989). Apart from establishing typical ranges of the bowing parameters, basic bowing tasks as *détaché*, *crescendo-diminuendo*, *sforzando* and *spiccato* were investigated. A general conclusion was that it is the coordination of the bowing parameters which is the most interesting aspect. The result was not surprising in view of the many constraints which determine the player's decisions on when and how to change the bowing parameters. However, it was a reminder of that the control of bowed-string synthesis needs interfaces which easily can control several parameters simultaneously, like a regular bow (Schoonderwalt, 2008).

1.4.4 Direct acquisition methods for bowing parameters in violin performance

Because of the complex and continuous nature of physical actions involved in the control of bowed-string instruments (often considered among the most articulate and expressive), acquisition and analysis of bowed-string instrumental gestures (mostly bowing control parameters) has been an active and challenging topic of study for several years, leading to diverse successful approaches.

Askenfelt (1986, 1989) presents methods for measuring bow motion and bow force using diverse custom electronic devices attached to both the violin and the bow. The bow transversal position was measured by means of a thin resistance wire inserted among the bow hairs, while for the bow-bridge distance, the strings were electrified, so that the contact position with the resistance wire among the bow hairs was detected. For the bow pressure, four strain gages (two at the tip and two at the frog) were used. A different approach was taken by Paradiso & Gershenfeld (1997), who measured bow displacement by means of oscillators driving antennas (electric field sensing). In a first application carried

out for cello, a resistive strip attached to the bow was driven by a mounted antenna behind the bridge, resulting as well into a wired bow. Afterward, in the violin implementation of this methodology which resulted into a first wireless measurement system for bowing parameters, the antenna worked as the receiver, while two oscillators placed in the bow worked as drivers. There, what is referred to as bow pressure was measured by using a force-sensitive resistor below the forefinger (or between the bow hair and wood at the tip). These approaches, while providing means for measuring the relevant bowing parameters, did not allow tracking performer movements. Furthermore, the custom electronic devices that needed to be attached to the instrument resulted to be somehow intrusive, while not being easy to interchange the instrument at performerŠs demand.

More recent implementations of violin bowing parameter measurement introduced some important improvements, resulting in less intrusive systems than previous ones. Young (2002, 2007) measured downward and lateral bow pressure with foil strain gages, while bow position with respect to the bridge is measured in a similar way as it was previously carried out by Paradiso & Gershenfeld (1997). The strain gages were permanently mounted around the midpoint of the bow stick, and the force data were collected and sent to a remote computer via a wireless transmitter mounted at the frog, resulting in considerable intrusiveness to the performer. Goudeseune (2001) used a commercial EMF device for tracking some low-level momevement parameters and use them for controlling some synthesis features in a performance scenario. The procedure for extracting movement or gestural parameteres was not much ellaborated, as he just used speeds or positions/rotations of the sensors in the violin or bow without extracting relevant instrumental gesture parameters.

Rasamimanana et al. (2006) performed wireless measurements of acceleration of the bow by means of accelerometers attached to the bow, and used force sensitive resistors (FSRs) to obtain the strain of the bow hair as a measure of bow pressure. This system had the advantage that could be easily attached to any bow. Conversely, it needeed considerable post-processing in order to obtain motion information, since it was measuring only acceleration. This was carried out afterward by Schoonderwaldt et al. (2006), who combined the use of video cameras with the measurements given by the acceloremeters in order to reconstruct bow velocity profiles.

Accuracy and robustness in bow pressing force measurement was recently taken to a higher level (see the work by Demoucron & Caussé (2007); Demoucron et al. (2009) and extensions by Guaus et al. (2007, 2009), the latter two constituting part of the contributions of this dissertation) by using strain gages attached to the frog end of the hair ribbon, thus measuring ribbon deflection.

Also recent is the approach presented by Maestre et al. (2007) (important part of this dissertation's contribution on bowing parameter acquisition techniques), where bowing control parameters are very accurately measured by means of one of the commercially available electromagnetic field-based tracking devices. Afterward, this methodology (based on tracking positions of hair string and

ribbon ends) was adapted by Schoonderwaldt & Demoucron (2009) to a more expensive commercial camera-based motion capture system that needed a more difficult calibration system and post-processing. Latterly, research in capturing bowing parameters in real time led to the first commercial product, the K-Bow[1]. It consists on an augmented bow plus additional electronics attached to the violin, and is mostly intended for controling sound processing algotithms in stage performance.

1.4.5 Computational modeling of bowing parameter contours in violin performance

Despite the existence of diverse successful approaches to the acquisition of violin gesture parameters, just a few applications have been devoted to the analysis of bowing parameters from a computational perspective. Rasamimanana et al. (2006) used bow acceleration extrema for automatic bow stroke classification by applying linear discriminant analysis to a number of features extracted from the contour of bow acceleration. Later, authors continued their work in (Rasamimanana et al., 2009; Rasamimanana & Bevilacqua, 2008), where they presented an analysis of performer arm coordination (joint angles) under different bowing stroke frequencies, as well as a description of motion from a kinetic perspective looking at effort, showing anticipation effects. The work by Young (2008) extended bowing technique automatic classification to a larger number of bowing techniques across different performers by extracting the principal components of raw bow acceleration and data coming from a strain gage sensor attached to the bow.

None of these approaches is aligned towards a modeling framework able to effectively represent parameter contours as related to the score so that it results possible to embark the generation of synthetic performances. Notwithstanding that, the challenging problem of synthesizing bowing control parameters from a musical score has indeed been approached in the past.

A first attempt is found in the work by Chafe (1988), where the author presented an algorithm for rendering a number of violin performance gesture parameter contours (including both left and right hand parameters) by concatenating short segments following a number of hand-made rules. Following the same line, extensions dealing with left hand articulations and string changes were introduced by Jaffe & Smith (1995) for controlling a digital waveguide bowed-string physical model. While resulting in pioneering applications of observation-based analysis of performance habits or conceptions, both approaches lacked a real data-driven definition of contours parameters, as the algorithms' parameters were manually tuned. Similarly, Rank (1999) pursued some research on using standard ADSR (Bernstein & Cooper, 1976) envelopes in MIDI-controled synthesis of violin, leaving clear the limitations of chosen contour representation. A recent study working with real data is found in the

[1] http://www.keithmcmillen.com/kbow/

works by Demoucron & Caussé (2007); Demoucron (2008); Demoucron et al. (2008), where bow velocity and bow force contours of different bow strokes are quantitatively characterized and reconstructed mostly using sinusoidal segments. While an engaging analysis of different dynamics and bow strokes (mostly focused on isolated notes) is carried out, flexibility limitations of the proposed contour representation may impede (as pointed out by the author) to easily generalize its application to other bowing techniques not only based on bow strokes per se, but also on more sustained control situations (e.g., longer *détaché* notes or *legato* articulations).

1.4.6 Violin sound synthesis

Bowing control modeling applied to sound synthesis can only be found in the context of physical models, mostly based on the digital waveguide modeling framework introduced by Smith (1992, 2002a, accessed 2009). Young & Serafin (2003) successfully evaluated the combination of a real-time, bowing control parameter measurement system (Young, 2002, 2007) with Smith's digital waveguide bowed string model featuring a friction model (Serafin, 2004). For the case of off-line scenarios, Chafe (1988), Jaffe & Smith (1995), and Rank (1999) pursued the application of synthetic bowing controls to off-line synthesis, but none of the models was obtained from real performance data. Conversely, Demoucron & Caussé (2007); Demoucron (2008); Demoucron et al. (2008) used real bowing control data for reconstructing contours of several bow strokes and applying them for generation of violin sound through a modal synthesis approximation.

For the case of sample-based violin sound synthesis, explicit bowing control is still a missing component. While extended commercial samplers provide a vast number of possibilities in terms of articulations, picthes, etc. (e.g., *Vienna Symphonic Library*[2]), sample concatenation and control flexibility limitations represent their major drawback. Specialized spectral domain sound transformations oriented toward real-time expressivity control using traditional MIDI-based keyboard interfaces can also be found in the market. Both the *Garritan Solo Stradivari*[3] and the *Synful Orchestra*[4] (Lindemann, 2007) achieve such sample transformations. An interesting feature of *Garritan* is the fact that it makes use of a sample database of string vibration signals and, after sample transformation and concatenation, the body resonances are added by convolution with an estimated impusle response. Something to mention about *Synfyul* is a modeling component able to synthesize spectral envelopes given a low-speed varying perceptual attribute signals (e.g., pitch, loudness, etc.) generated from an input score.

[2] http://vsl.co.at/
[3] http://www.garritan.com/stradivari.html
[4] http://www.synful.com/

1.5 Objectives and outline of the dissertation

The main objective of this dissertation is to develop a comprehensive modeling framework suitable for the analysis and synthesis of insttrumental gestures in excitation-contnuous musical instruments. The instrument chosen is the violin, as it is considered among the most complex to play and it offers the performer a broad range of expressive resources and playing techniques. Moreover, the acquisition of instrumentals gesture parameters, in particular bowing gestures, becomes affordable in low-intrusiveness conditions. The focus is put into bowing techniques, and the modeling framework is devised by keeping a strong connection to the score being played.

The aim of this work could be seen as an attempt to propose a methodology for data-driven modeling the crucial function of the performer in transforming the discrete-nature information appearing in an annotated musical score into the continuous-nature physical actions driving sound production. This dissertation intends to introduce and validate a systematic approach to the acquisition, representation, modeling, and synthesis of bowing patterns in violin classical performance. This involves a number of research challenges:

- **Acquisition**

  Acquisition of timbre-related bowing parameter signals and construction of a performance database including gesture and sound data aligned to performed scores. The database should cover a representative set of bowing techniques. The acqusition techniques must be devised by attending to robustness, accuracy, low-intrusiveness, and portability. The construction of the database should be suited for automatization, enabling rapid post-processing of large amounts of data.

- **Representation**

  Design of a contour representation framework that is suitable for quantitatively supporting the definition of a bowing technique vocabulary in terms of temporal patterns of bowing parameter envelopes. The framework must be (1) flexible enough for providing contour representation fidelity, and (2) robust enough for ensuring contour parameterization consistency accross different executions of similar bowing techniques. The extraction of contour parameterizations should also be devised so that an automatization of the process is going to make further data analysis to be more feasible.

- **Modeling**

  Statistical analysis and modeling of the parameterizations obtained from acquired contours, supporting a meaningful representation of observed variability in bowing parameter contours. The approach followed for building the statistical model should enable a flexible mapping between bowing parameter contours and annotations of performed scores.

- **Synthesis**

  Generation of synthetic bowing parameter contours. The score-gesture mapping model developed shall ground a gesture rendering framework able to synthesize, from an annotated input score, envelopes of bowing control parameters. The synthesis method should take advantage of the flexibility brought by the contour representation and modeling schemes, so that it allows to smoothly navigate the space of score annotations while appropriately shaping the corresponding bowing parameter signals.

- **Validation**

  As an ideal validation for the modeling framework, the use of synthetic bowing parameter contours for controlling off-line sound synthesis shall provide means for obtaining realistic violin performances from an annotated input score. The improved naturalness of obtained sound should evidence that the explicit introduction of an instrumental gesture modeling component, which enables the emulation of the performer's role, significantly enhances instrumental sound synthesis when applied to automatic performance.

The contributions of this work are presented by attending to the (chrono-) logical order followed for succeeding in the approach to the challenges enumerated above. The disseration is structed as follows.

Chapter 2 starts by presenting the methods developed and applied for acquiring a number of bowing gesture parameters, with focus on timbre-related bowing control parameters (bow velocity, bow pressing force, and bow-bridge distance). Moreover, by means of a combined use of acquired bowing parameters and a number of audio features, an automatic score-performance alignment algorithm is introduced as the basis for constructing a synchronized database of audio and gesture data which covers a representatitve set of bowing techniques played in different contexts.

Chapter 3 introduces the quantitative representation of bowing parameter contours. First, a contour qualitative analysis of acquired bowing parameters is presented, uncovering the principal reasons that support the approach chosen for consistently representing contours of bowing parameters. Then, it is introduced the quantitative representation framework, along with an algorithm devised for automatically obtaining quantitative descriptions of bowing parameter contours.

Chapter 4 presents a statistical modeling framework that maps score annotations to bowing contour parameters. The model is used for building a synthesis framework able to render bowing controls from an input score. The chapter gives details on the statistical analysis of bowing parameter contours, and the construction of the rendering model (based on Gaussian mixtures). Finally, it is introduced a bow planning algorithm that integrates the rendering model while statistically accounting for the potential constraints imposed by the finite length of the bow.

Chapter 5 introduces an application use for validating the instrumental gesture modeling framework developed in this dissertation. Synthetic bowing parameter contours obtained from an annotated input score are used as a key component for synthesizing realistic, natural sounding violin sound through the explicit inclusion of instrumental control information in two of the most common sound synthesis approaches: physical modeling synthesis and sample-based synthesis.

Finally, Chapter 6 summarizes the achievements of the thesis in approaching each of the research challenges that motivated this work, analyzing the strengths and limitations of the introduced methodology. Forthcoming steps, improvements, and applications are also presented. Moreover, contributed new capabilities are shortly presented as a foundation for future directions both in the way of instrumental gesture modeling, and towards a further, significant improvement of instrumental sound synthesis.

Chapter 2

Bowing control data acquisition

The first set of contributions is enclosed in this chapter. It starts by presenting the methods developed and applied for acquiring a number of bowing gesture parameters, with focus on timbre-related bowing control parameters (bow velocity, bow pressing force, and bow-bridge distance), from real violin playing. Bow velocity and bow-bridge distance are measured by tracking the position and orientation of bow and violin using a commercial electromagnetic field sensing device, while an estimation of the bow force is obtained from strain gages attached to a bending plate laying against bow hairs. The string being played is automatically detected from bow and violin orientations. By means of a combined use of acquired bowing parameters and a number of audio features, automatic score-performance alignment is carried out for constructing a synchronized database of audio and gesture data which covers a representatitve set of bowing techniques played in different contexts.

Three publications by the author are directly related to the contents of Chapter 2. The first work (Maestre et al., 2007) introduces the framework and procedures needed for the acquisition of motion-related bowing parameters based on tracking position and orientation of a number of relevant points of both the violin and the bow. It also proposes the basis on which is constructed the score-performance alignment detailed at the end of the chapter. The suitability of acquiring bowing parameters from tracking the positions of the ends of strings and hair ribbon was proved in a later work by Schoonderwaldt & Demoucron (2009) in which authors pursue an implementation of similar ideas (motion-related parameters from position tracking, and string detection from relative orientations), but adapted to a position tracking setup based on infrared cameras.

The other two publications (Guaus et al., 2007, 2009) focus on the implementation and refinement of a methodology for estimating the bow pressing force by measuring the deflection of the hair ribbon (inspired by the work of Demoucron & Caussé (2007); Demoucron et al. (2009). Previously, in the first publication, it was proposed a method for estimating the bow pressing force

also from positions and orientations of a number of relevant points of both the violin and the bow. The ideas of such method, not detailed in this disseration, were succesfully applied by Schoonderwaldt & Demoucron (2009) in a posterior work.

2.1 Acquisition process overview

Several requirements need to be taken into account when devising the acquisition setup and methodology. Given the principal aim of capturing bowing control parameters that are relevant to timbre nuances occurring during *classical* performance, a first concern is to establish which parameters, apart from produced sound, will be captured from real performances. As already discussed in Section 1.4, extensive and detailed studies (Schelleng, 1973; Schumacher, 1979; Cremer, 1984; Askenfelt, 1986, 1989; Schoonderwalt, 2008) have shown that the three most important bowing control parameters related to timbre are *bow transversal velocity*, *bow-bridge distance*, and *bow pressing force*. Because of the objective of using acquired data for analyzing, modeling, and synthesizing bowing control parameter contours, accuracy and robustness of the measurement method is of crucial importance. Not less important is the intrusiveness of the setup: the performance process should remain unaffected, allowing the performer to play as naturally as in normal performance conditions. A last concern to remark is the need for a portable, non-complex system not requiring complicated installations or calibration processes, so that recordings can be carried out in realistic performance contexts.

After a survey of existing literature and available acquisition techniques (see Section 1.4.4), the requirements and concerns outlined above lead to the design of an acquisition process able to meet the application requirements through the following approaches:

- **Audio acquisition**: bridge piezoelectric pickup substituting violin original bridge, so that violin sound gets unaffected while being able to capture the important timbre characteristics of the string signal which are related to bowing control.
- **Bowing motion parameter acquisition**: tracking device that is portable, able to provide position and orientation of both violin and bow, and featuring enough accuracy at a sufficiently high sampling rate so that bow velocity and bow-bridge distance can be reliably estimated.
- **Bow force acquisition**: strain gage-based device able to accurately measure hair ribbon deflection so that bow pressing force can be estimated.

The acquisition process, in which audio, motion, and force data are synchronously captured, is sketched in Figure 2.1. Audio (string velocity signal) is acquired from the bridge pickup, and a number of derived audio descriptors are used during database creation and annotation. Position and orientation of two

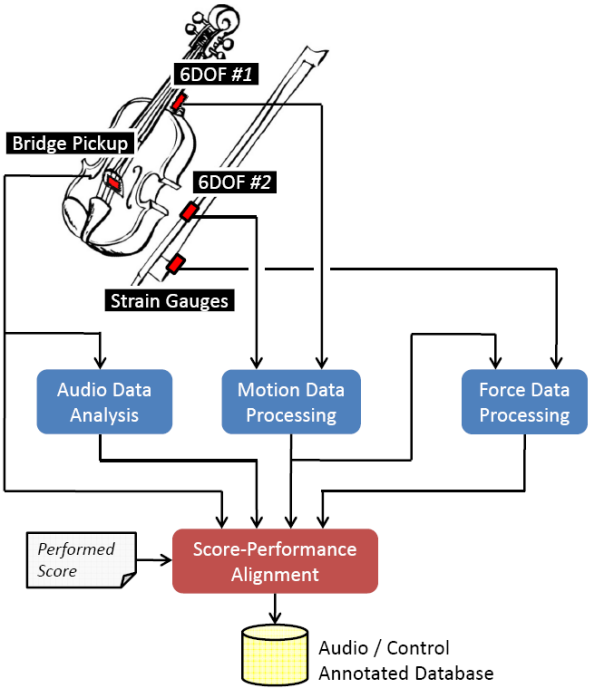


Figure 2.1: Overview of the data acquisition process. Audio analysis data (from the bridge pickup), motion-related processed data (from the 6DOF motion tracking sensors), and force processed data are used during score-performance alignment. Acquired data and score-based annotations extracted from the performed scores are included in performance database containing time-aligned audio and instrumental control signals.

six-degrees of freedom (6DOF) sensors (respectively attached to the violin and to the bow) are acquired and processed for obtaining relevant motion-related bowing control parameter signals (e.g., bow position, bow velocity, bow-bridge distance, etc.). Mounted on the bow frog, a strain gage-based device specifically developed for acquiring hair ribbon deflection provides means for capturing the bow pressing force by combining its output signal and some of the acquired motion-related parameters. Database creation and annotation is carried out off-line with the help of an automatic score-performance alignment procedure that uses both the audio descriptors and the acquired bowing control parameters.

2.2 Audio acquisition

The acquisition of bowing control parameter signals does not appear at first sight as strongly linked to capturing the sound produced by the violin. However, it results very important to acquire sound and synchronize it to bowing control acquired data. A first reason comes from the fact that audio analysis features are used during construction and annotation of the database of bowing control parameter contour (see Section 2.6). Second, the validation of the gesture modeling framework implies applying synthetic bowing control parameters to sample-based sound synthesis, which indeed needs of an annotated database including both audio and bowing control parameter streams (see Section 5.3).

Taking into account this, and also foreseeing upcoming needs concerning the timbric content of acquired sound, it is decided to carry out audio capturing by means of a piezoelectric bridge pickup replacing the original bridge. Several commercially available choices are considered for the pickup, having the *Yamaha© VNP1*[1] bridge pickup system as the best suited regarding its frequency response when compared to the others. The system, consisting on a hard maple bridge with a piezoelectric transducer attached, can be seen in Figure 2.2 when mounted on the violin.

Regarding the timbric content of the acquired sound signal, it is critical to avoid as much as possible the resonant and reverberating characteristics of the violin body. Getting a *clean* audio signal (close to that of string velocity or force) brings two main advantages in the context of application of this work. First, audio analysis provides better results when using extracted features (e.g., energy or fundamental frequency estimation) for automatic alignment purposes. Second, the acquired audio database is used in a sample-based synthesis approach by which the body effects on the audio signal are considered to be linear, therefore having sample transformations to be applied to the bridge pickup signal before concatenated samples are convolved with an estimation of the violin body impulse response (see Section 5.3). Moreover, the impulse response of the violin body (to be used during sound synthesis, see Section 5.3) is going to be estimated by deconvolving a microphone signal with the signal captured by the bridge pickup.

2.3 Bow motion data acquisition

After considering the aforementioned requirements and concerns, the choice for the acquisition of motion-related bowing parameters is the *Polhemus© Liberty*[2] commercial device, a 6DOF tracking system based on electromagnetic field (EMF) sensing. It consists of a transmitting source, and a set of receiving wired spheric sensors with sizes going down to ∅0.5cm and weights down to 6gr (see Figure 2.3). Each sensor provides 3DOF for translation and 3DOF for

[1] http://www.yamaha.co.jp/english/product/strings/v_pickup/index.html
[2] http://www.polhemus.com/?page=Motion_Liberty

Figure 2.2: Detail of the *Yamaha*© *VNP1* bridge piezoelectric pickup mounted on the violin.

rotation at 240Hz sampling rate, with translation and rotation static accuracies of 0.75mm and 0.15° RMS respectively within a range of 1.5m of distance to the source when using the source model *TX4*.

One sensor is attached to to the violin body, and the other one is attached to the bow. From the 6DOF data stream of the former, the exact position of the ends of each string can be estimated for any position or orientation of the violin. Analogously, the ends of the hair ribbon can be accurately estimated from the 6DOF data streams of the latter. String ends and hair ribbon ends, together with the position of the sensors, are used to obtain the bow transversal position and velocity, bow-bridge distance, and bow tilt. In order to minimize the intrusiveness of the setup, the bow sensor (model RX1-D, see Figures 2.3 and 2.4) is attached to the wood, close to the center of gravity of the bow. While the balance point of the bow remains unaffected, the weight is increased by 12gr including the wire. Despite not being a wireless setup, the intrusiveness achieved (including the bow force sensing device, see next section) improves that of existing configurations as the *Hyperbow* (Young, 2002).

2.3.1 Calibration procedure

The calibration process comprises two analogous steps. The first step provides the means for obtaining the positions of all eight string ends as relative to the coordinate system defined by the translation and rotation of the sensor attached to the violin (i.e., the position and orientation of the violin). The second step allows defining the the positions of the hair ribbon ends as relative to the coordinate system defined by the translation and rotation of the sensor attached to the bow [3].

[3] Along this section, and also during Section 2.3.2, points will be denoted with italics, vectors or segments will be noted with bold letters, and matrices will be referred to by means of capital italics.

(a) Sensor *Polhemus© RX2* (b) Sensor *Polhemus© RX1-D* (c) Probe *Polhemus© ST8*

Figure 2.3: Detail of the *Polhemus©* 6DOF sensors and probe used. The sensor *Polhemus© RX2* (referred to as s_{c1}) is the one chosen to be attached to the violin, while the sensor *Polhemus© RX1-D* (referred to as s_{c2}) is the one chosen to be attached to the bow due to its reduced size and weight (⌀0.5cm and 6gr respectively). The probe *Polhemus© ST8* is used during the calibration process.

For the subsequent explanations on the calibration procedure, the sensor attached to the violin is referred to as s_{c1}, and the sensor attached to the bow is referred to as s_{c2}. A probe is also used (see Figure 2.3), referred to as s_p. The three sensors provide translation and rotation data with respect to a common reference (or *world* coordinate system) defined by the position and orientation of the *Polhemus© Liberty* emitting source. The translation and orientation of s_{c1} define the *violin coordinate system* (i.e., vectorial space) in which the coordinates of the string ends remain unaltered when the violin is moving. Analogously, the translation and orientation of s_{c2} define the *bow coordinate system* in which the coordinates of the hair ribbon ends move and rotate along with the bow.

After calibration (i.e., during performance), the sensor s_{c1} remains attached to the violin body, and the sensor s_{c2} does so to the bow, so that the strings and the hair ribbon can be tracked. The probe s_p is used during calibration as a marker for annotating the positions of the hair string ends and ribbon ends points so that they can be expressed in their respective coordinate systems (i.e., obtain their respective coordinates). The placement of sensors s_{c1} and s_{c2} (see Figure 2.4) has been chosen as to minimize their effect on the performer's comfortability.

Once the coordinates of each of the relevant points (e.g., string ends and hair ribbon ends) in their respective vectorial spaces are obtained through calibration, they can be translated and rotated to the reference vectorial space (i.e., the *world* coordinate system) at any time by attending to the translation and rotation data coming from the their respective sensors. The coordinates to be obtained, named b_v, apply to:

- **Violin coordinate system**: position of each of the eight string ends, expressed in the basis defined by the vectorial space v_1 provided by the

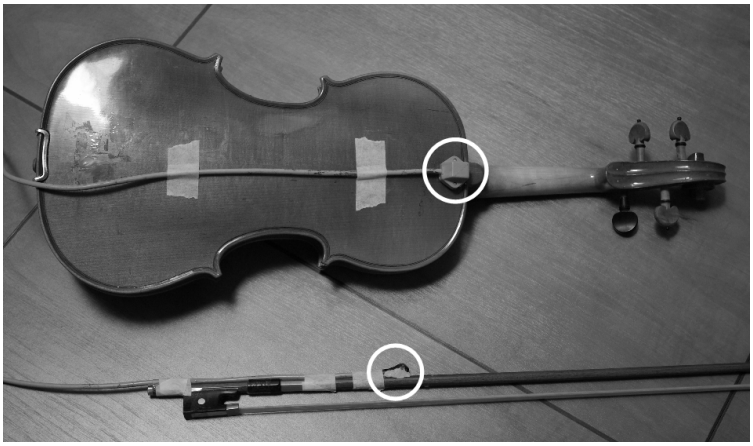


Figure 2.4: Detail of violin and bow placement of the 6DOF sensors during calibration process. The same exact position is kept during performance.

position and orientation (with respect to the reference vectorial space) of the sensor s_{c1} (attached to the violin body).

- **Bow coordinate system**: position of each of the two hair ribbon ends, expressed in the basis defined by the vectorial space v_2 provided by the position and orientation (with respect to the reference vectorial space) of the sensor s_{c2} (attached to the bow).

As already pointed out, tracking the positions of the relevant points during performance becomes a matter of simply returning each of the previously obtained b_v coordinates to the reference vectorial space. The process of obtaining the b_v coordinates for one of the relevant points is introduced next (it applies to any of the ten relevant points).

In Figure 2.5 appear represented two coordinate systems. The one on the left represents the reference (or *world*) coordinate system, and is defined by the point $(0,0,0)$ and the canonical base $\{\mathbf{x}_e, \mathbf{y}_e, \mathbf{z}_e\}$, both corresponding to the translation and orientation of the *Polhemus© Liberty* emitting source. The one on the right side represents the *mobile* vectorial space, defined by the point c (from the translation of the sensor) and the orthogonal base $\{\mathbf{x}_v, \mathbf{y}_v, \mathbf{z}_v\}$ (from the rotation of the sensor). The point p represents the relevant point for which its coordinates b_v are to be obtained, and corresponds to the position where the probe s_p is pointing during calibration. The first step is to obtain the vector $\mathbf{b}$ as

$$\mathbf{b} = \mathbf{p} - \mathbf{c} \tag{2.1}$$

Then, a change of basis is performed on the vector $\mathbf{b}$ so that it gets expressed in the *mobile* vectorial space (defined by c and $\{\mathbf{x}_v, \mathbf{y}_v, \mathbf{z}_v\}$). For doing so, an inverse linear rotation is applied as

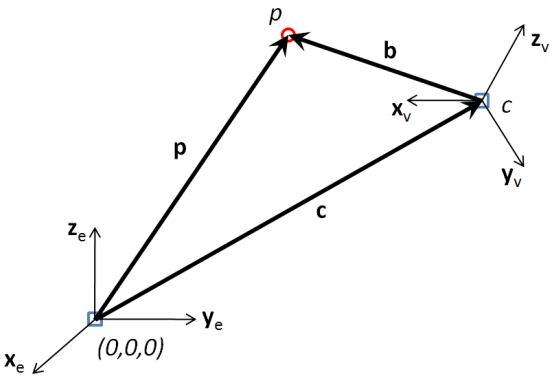


Figure 2.5: Graphical representation of the change of base needed both for obtaining the b_v coordinates during the calibration step, and for tracking relevant positions during performance. The point $(0,0,0)$ corresponds to the emitting source, the point c represents the position of the sensor, and the point p corresponds to the relevant point to be tracked (the latter maintains its relative position with respect to point c).

$$\mathbf{b_v} = R_v^{-1} \times \mathbf{b}, \tag{2.2}$$

where R_v represents the rotation matrix obtained from the Euler angles[4] derived from the relative rotation of $\{\mathbf{x}_v, \mathbf{y}_v, \mathbf{z}_v\}$ (*mobile* coordinate system) with respect to $\{\mathbf{x}_e, \mathbf{y}_e, \mathbf{z}_e\}$ (*world* coordinate system). Now, the vector $\mathbf{b_v}$ defines the coordinates b_v (in the *mobile coordinate system*) of the relevant point p.

During performance, any new position p' (defined by $\mathbf{p}'$) of a relevant point will be obtained from its b_v coordinates (defined by $\mathbf{b_v}$), and any new position c' (defined by $\mathbf{c}'$) and orthogonal basis $\{\mathbf{x}'_v, \mathbf{y}'_v, \mathbf{z}'_v\}$ corresponding the sensor's new position and orientation. This is expressed in equation (2.3), where R'_v is the rotation matrix obtained from the new Euler angles of sensor.

$$\mathbf{p}' = R_v^{'-1} \times \mathbf{b_v} + \mathbf{c}' \tag{2.3}$$

Step 1: String ends

The calibration of the string ends consists on obtaining the b_v coordinates of each of the eight string ends (four at the bridge and four at the nut), i.e., expressed in the *mobile* vectorial space defined by the translation and rotation of the sensor s_{c1} attached to the violin body. For doing so, the violin is kept still while (1) the probe s_p is used for annotating the position p (see Figure 2.6) of

[4]The *Polhemus*© *Liberty* system provides rotation data as three Euler angles.

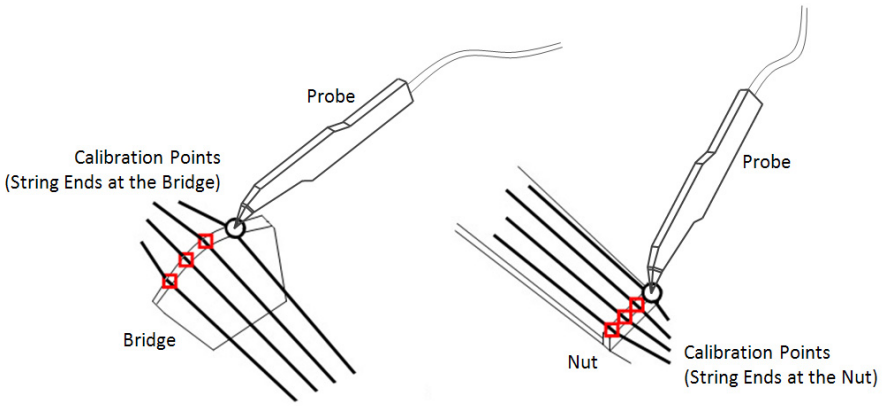


Figure 2.6: Schematic view of the string calibration step. Sensor s_{c1} remains attached to the violin body, and probe s_p is used to annotate the position of the string ends. The violin is kept still.

each eight string end, and (2) c and R_v are obtained from the translation and rotation of sensor s_{c1}. The change of basis given by equations (2.1) and (2.2) is then applied for obtaining the b_v coordinates of each string end.

Step 2: Hair ribbon ends

Analogously to step 1, the calibration of the two hair ribbon ends (one at the frog and another at the tip) consists on obtaining their corresponding b_v coordinates, this time expressed in the *mobile* vectorial space defined by the translation and rotation of the sensor s_{c2} attached to the bow. Again, the bow is kept still while (1) the probe s_p is used for annotating both the position p of the both hair ribbon ends, and (2) c and R_v are obtained from the translation and rotation of the sensor sensor s_{c2} (see Figure 2.7).

2.3.2 Extraction of bow motion-related parameters

By following the procedure outlined previously, the exact position of the string ends and the hair ribbon ends is tracked during performance at a sampling rate $s_r = 240$Hz (corresponding to the tracker's sample rate). From such positions, a series of computations are applied for accurately obtaining a number of parameters related to bow motion.

In a first step, the orientation of the bow is estimated from the eight string ends, leading to a vector $\mathbf{v}_n$ normal to the violin plane. In a similar manner, a the orientation of the bow is estimated from the plane formed by the two hair

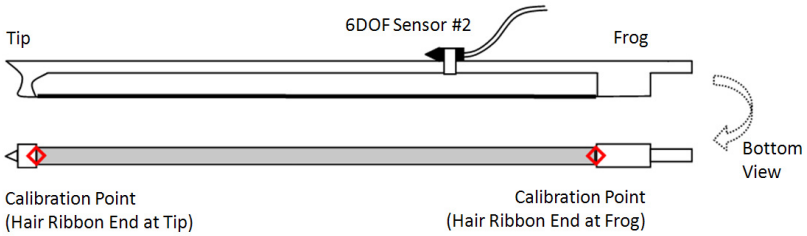


Figure 2.7: Schematic view of the hair ribbon calibration step. Sensor s_{c2} remains attached to the bow, and probe s_p is used to annotate the position of the hair ribbon ends ends. The bow is kept still.

ribbon ends and the sensor s_{c2} providing a vector $\mathbf{b}_n$ normal to the plane of the bow (see Figure 2.8).

Then, three relevant segments are defined. The first, representing the string being played, is obtained as going from the string end s_b at the bridge, and the string end s_n at the nut. The second segment corresponds to the hair ribbon, and is defined as going from the tip end h_t to the frog end h_f. Finally, a segment **P** is obtained as the shortest perpendicular to the former two, crossing at points p_s and p_h respectively (depicted in Figure 2.8).

Estimation of the string being played

The estimation of the string being played is based on measuring an angle α between vectors $\mathbf{v}_n$ and $\mathbf{h}$ (see Figure 2.8):

$$\phi = acos \frac{\mathbf{v}_n \cdot \mathbf{h}}{||\mathbf{v}_n|| \, ||\mathbf{h}||} \tag{2.4}$$

By comparing the angle α to a number of pre-calibrated angles indicating the characteristic inclination of the bow when playing each string, the string currently played is detected. These calibrated angles are schematically represented in Figure 2.9 (left). Red lines depict the characteristic angles mostly used for playing single strings (α_G,α_D,α_A, and α_E) and double strings (α_{GD},α_{DA}, and α_{AE}). Green lines depict the characteristic angles most likely defining the limit between strings being played: α_{T1} (limit between playing the *G* string, and both *G* and *D* strings), α_{T2} (limit between playing both *G* and *D* strings, and the *D* string), α_{T3} (limit between playing the *D* string, and both *D* and *A* strings), α_{T4} (limit between playing both *D* and *A* strings, and the *A* string), α_{T5} (limit between playing the *A* string, and both *A* and *E* strings), and α_{T6} (limit between playing both *A* and *E* strings, and the *E* string).

The angle pre-calibration consists on asking the performer to play a known sequence of notes for which the strings to play are scripted, and estimating

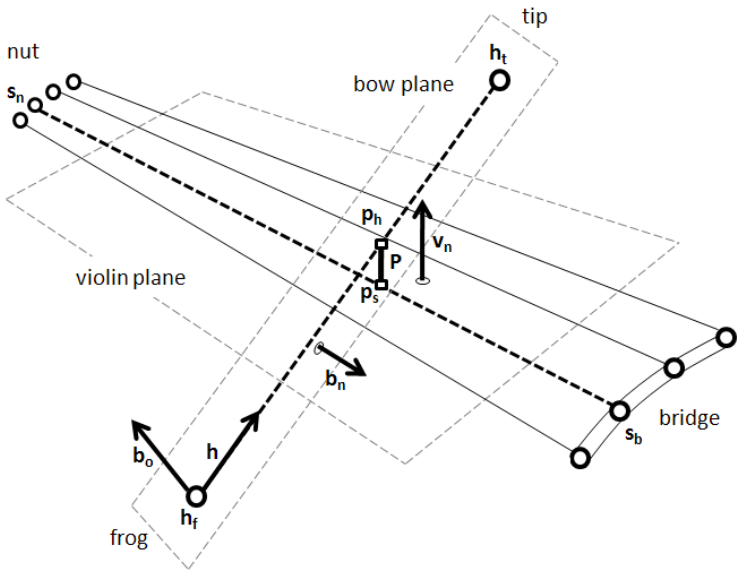


Figure 2.8: Schematic representation of the relevant positions and orientations relevant to the extraction of bowing motion parameters.

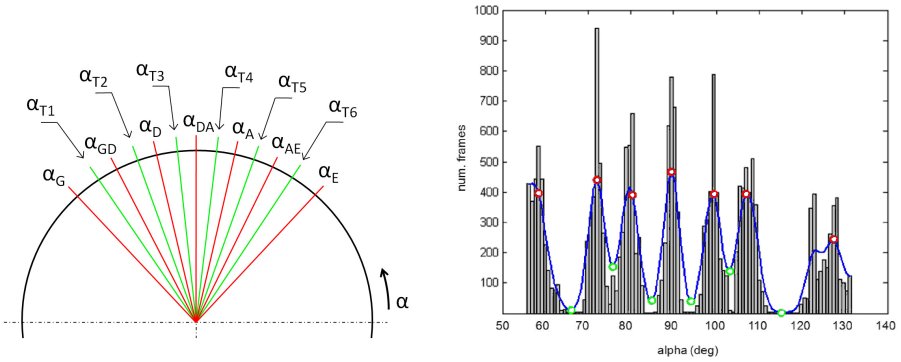
the characteristic angles from each of the segments. The sequence includes different dynamics and durations for notes played at each string, so that the complete length of the bow is used. The string sequence is as follows:

$$G \rightarrow GD \rightarrow D \rightarrow DA \rightarrow A \rightarrow AE \rightarrow E$$

Once the recording is finished, scripted string change times are aligned to the recorded performance by detecting the audio energy local minimum around each string change nominal transition time (i.e., a bow direction change is performed). Then, the values of the computed angle α corresponding to each segment (i.e., to each scripted string) are collected, and seven histograms are built each one corresponding to the angles α_G, α_{GD}, α_D, α_{DA}, α_A, α_{AE}, and α_E. Each of these seven characteristic angles (depicted in red in Figure 2.9) is estimated as maximum of its corresponding histogram.

In order to calibrate the limit angles α_{T1} to α_{T6}, a histogram is built from the angle data of all segments. After smoothing the histogram, each limit angle is set to the local minimum found between each consecutive pair of the maxima detected before. This is represented in Figure 2.9 (right), where the light blue line corresponds to the smoothed version of the final histogram, and red and green circles respectively represent the calibrated characteristic angles and limit angles.

The estimation of the string being played is performed by comparing the computed angle α with the calibrated limit angles. In order to avoid glitches



(a) Illustration of the characteristic α angles used for the estimation of the string being played.

(b) Violin plane - hair ribbon angle α histogram constructed during the string pre-calibration.

Figure 2.9: Relevant α angles used during the automatic estimation of the string being played

around decision boundaries, an hysteresis cycle of 1.5° is applied. Figure 2.10 shows the detection results for the fragment recorded for calibration. From top to bottom appear the audio signal with the audio-based segmentation limits superimposed, the computed angle α, and the string detection results. In the bottom plot, the estimated strings are represented as follows: string G, value of 4; double string GD, value of 3.5; string D, value of 3; double string DA, value of 2.5; string A, value of 2; double string AE, value of 1.5; string E, value of 1.

Although it is possible to detect both double and single strings, a 3-level angle detection configuration may be used when aiming at estimating only single strings. In scenarios where no double strings appear in the performances to be recorded, a new set of three transition angles α'_{T1}, α'_{T2}, and α'_{T3} is used, having their values respectively obtained from the characteristic angles α_{GD},α_{DA}, and α_{AE} of the original calibration.

Bowing control

Once the string being played is estimated, the relevant points and vectors introduced before are used for estimating the following bowing control-related parameters (c.f. Figure 2.8):

- The **bow transversal position** p_b is defined as the euclidean distance between the points p_h and h_f:

$$p_b = \sqrt{(p_{h,x} - h_{f,x})^2 + (p_{h,y} - h_{f,y})^2 + (p_{h,z} - h_{f,z})^2} \tag{2.5}$$

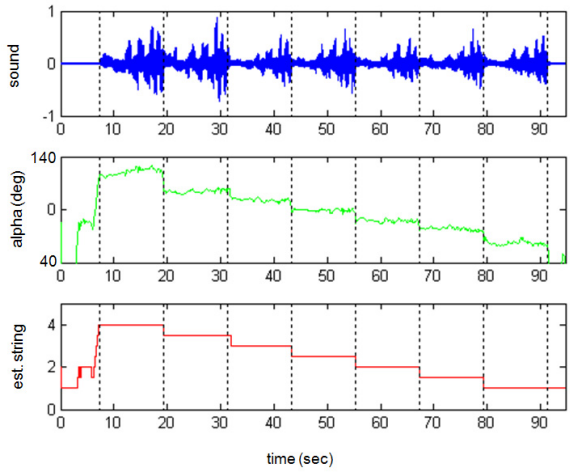


Figure 2.10: Results of the estimation of the string being played. From top to bottom: audio and nominal string change times, computed angle α, and estimation result.

- The **bow-bridge distance** d_{bb} is defined as the euclidean distance between p_s and s_b:

$$d_{bb} = \sqrt{(p_{s,x} - s_{b,x})^2 + (p_{s,y} - s_{b,y})^2 + (p_{s,z} - s_{b,z})^2} \tag{2.6}$$

- The **bow transversal velocity** v_b is obtained as the time derivative of the bow transversal position p_b:

$$v_b = \frac{\delta p_b}{\delta t} \tag{2.7}$$

- The **bow tilt angle** ϕ, providing an indication of the rotation of the bow around its main axis (given by **h**, see Figure 2.8), is obtained from computing the angle between vector $\mathbf{b}_n$ and vector $\mathbf{v}_n$:

$$\phi = acos \frac{\mathbf{b}_n \cdot \mathbf{v}_n}{||\mathbf{b}_n||\ ||\mathbf{v}_n||} \tag{2.8}$$

For the particular case of bow transversal velocity v_b, the ends of an hypothetical fifth string placed between string *A* and string *D* can be used instead. The reason for doing so resides on considering the bow velocity to be independent of the string being played, and also on willing to avoid artifacts in bow velocity velocity contour as it is especially sensible to sudden changes.

2.4 Bow force data acquisition

Measuring the bow pressing force in real violin performance is not a straightforward task. This sections outlines the approach taken for estimating the bow pressing force. This procedure introduced here results from the continuation and the improvements of previous works (Guaus et al., 2007, 2009). The reader is referred to these publications for getting further insight and details.

The acquisition of bow pressing force is carried out by means of a dual strain gage device mounted on the frog of the bow (see Figure 2.11), having one strain gage attached to each side of a metallic plate that is laying against the hair ribbon. Supported by a triangular piece attached to the wooden frog, the plate is permanently bended and suffers a deformation directly related to the deflection of the hair ribbon. When playing on a string, the force applied by the performer leads to such deflection, so an estimation of the pressing force can be pursued from the resistance changes caused by the deformation of the strain gages.

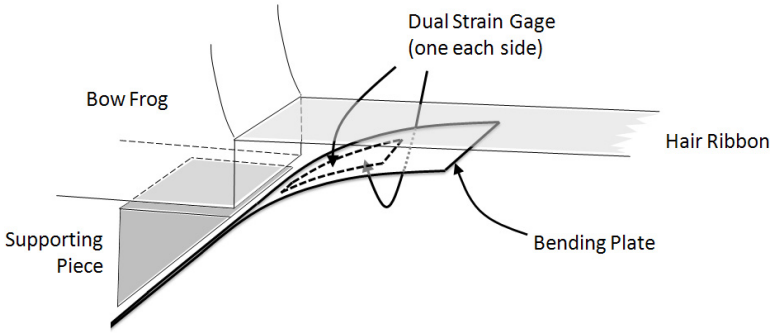
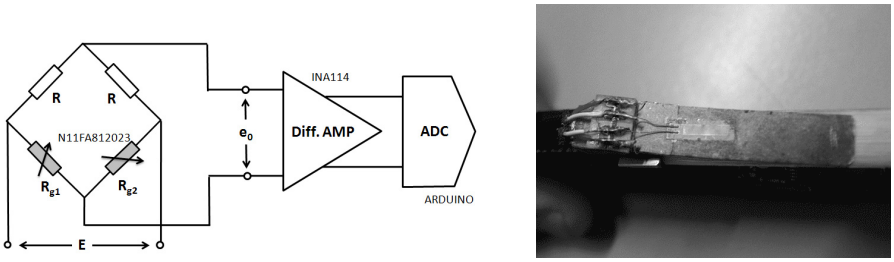


Figure 2.11: Illustration of the dual strain gage setup by means of which the bow pressing force is measured. Two strain gages are glued each one to one side of a metal bending plate.

At this point, one finds a first big difference between this approach and the proposal by Demoucron & Caussé (2007); Demoucron (2008); Demoucron et al. (2009) (on which this approach is based), where the author also introduces the use of two strain gages, but having one gage glued to a metallic bending piece at the frog, and another at the tip. Apart from needing additional hardware to be attached to the bow tip, having instead two strain gages mounted at opposite sides of the same metallic piece brings a number of advantages when configured using a Wheatstone bridge as shown in Figure 2.12 (left): it delivers twice the sensitivity while allowing for temperature compensation, having also eventual thermal effects on lead wires to be canceled. This implementation caused a slight reduction of the effective length of the bow, having eventual imbalances

because of the performer hitting the string with the plate. This problem, while not representing a critical drawback for a studio recording context, would need to be tackled in a similar way as in (Demoucron et al., 2009), where the author instead places the metallic plate in the inner side of the hair ribbon.

The voltage signal provided by the bridge is conditioned, normalized, and converted to digital samples (at a sample rate of 240Hz) by means of a differential amplifier (INA114) and the ARDUINO[5] prototyping board. Picture in Figure 2.12 (right) shows a close view of the metallic plate attached to the frog, where one of the strain gages is displayed.



(a) Schematics of the strain gage-based deformation measurement device

(b) Detail of the dual strain gage setup mounted on the bow frog (bottom view)

Figure 2.12: Detail of device constructed for measuring hair ribbon deflection

The following important step in pursuing an appropriate and reliable bow force acquisition method is the calibration process. It is crucial to account for the non-linear relationship between the actual force F applied by the performer and the hair ribbon deflection measure d_h (the latter given by the readings coming from the dual strain gage device). Because of the different behaviors that such relationship shows for different combinations of bow transversal displacement p_b and bow tilt angle ϕ (see previous section), calibration is approached by learning a non-linear function f able to provide a reliable value of bow pressing force F for any combination of bow displacement p_b, bow tilt ϕ, and hair ribbon deflection d_h to happen during performance:

$$F = f(d_h, p_b, \phi) \tag{2.9}$$

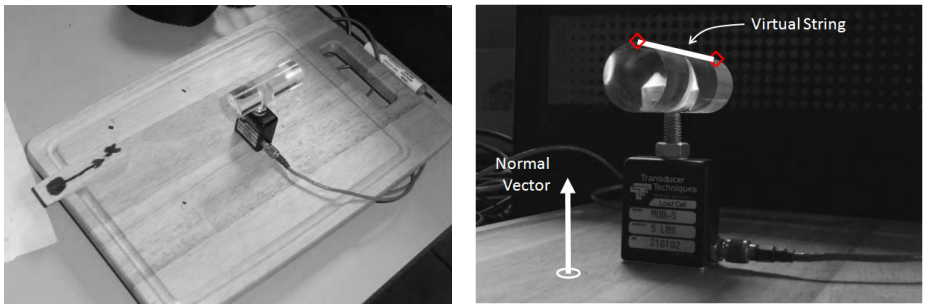
2.4.1 Bow force calibration procedure

The calibration procedure, thought for being carried out as a pre-recording step, implies the construction of a *bowing table* on top of which is attached a commercial force transducer (load cell *Transducer Techniques© MDB-5*)[6] hold-

[5] http://www.arduino.cc/en/

[6] http://www.transducertechniques.com/MDB-Load-Cell.cfm

ing a rolling cylinder (emulating the string) able to provide values of pressing force (see Figure 2.13 (left)). Prior to the performance recording, the performer is asked to hold the table and bow on the cylinder as if it were a violin string, having both the actual bow force F and the hair ribbon deflection d_h acquired.



(a) Bowing table

(b) Bowing cylinder mounted on top of the load cell *Transducer Techniques© MDB-5*

Figure 2.13: Detail of the constructed bowing table device for carrying out bow pressing force calibration.

One of the 6DOF sensors is attached to the bow, and the other is attached to the table. By following the motion tracking calibration described in Section 2.3.1, it is possible to acquire the position and orientation of the bowing table as if it were the actual violin when held by the performer. Therefore, it results straightforward to obtain (1) a vector that is normal to the plane of the table, and (2) the end points of a virtual string (top edges of the cylinder) where the performer is asked to bow. These three elements are displayed in Figure 2.13 (right). By following the methodology presented in Section 2.3.2, bow transversal displacement p_b and bow tilt angle ϕ are obtained as related to the virtual string. Now, the four values appearing in the function in equation (2.9) can be acquired and used for modeling such non-linear function.

In order to cover all possible combinations of bow force F, bow displacement p_b, and bow tilt ϕ to occur during real performances, the musician is asked to bow the cylinder for several minutes in all sort of bowing conditions. From the acquired data, a Support Vector Regression (SVR) model (Cristianini & Shawe-Taylor, 2003) available implementation[7] is trained to predict the value of F having as input the values of d_h, p_b, and ϕ (see Figure 2.14). Feeding the model with the set of acquired examples and evaluating it by means of a 10-fold cross validation procedure[8], a correlation coefficient well above 0.95 is obtained. Figure 2.15 shows part of the data used during one of the calibrations, along with the bow force predicted by means of the trained model.

[7] http://www.csie.ntu.edu.tw/~cjlin/libsvm/

[8] It consists on training the model ten times, having the examples shuffled and separated into 90% and 10% sets (training set and test set) each time, and averaging the obtained results.

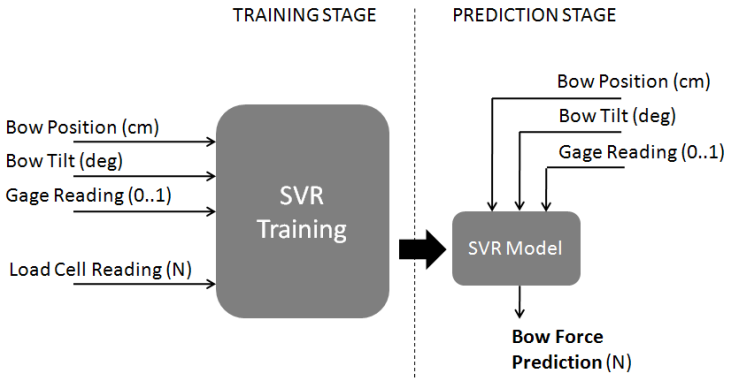
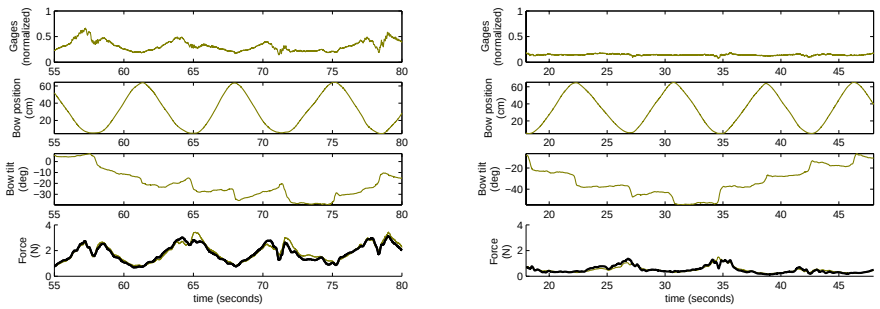


Figure 2.14: Schematic diagram of the Support Vector Regression (SVR)-based processes of training and prediction of bow pressing force. The model is trained with the actual bow force data acquired from a load cell in a pre-recording calibration step. During performance, the model is used for obtaining reliable values of bow force in absence of the force transducer data.

Because of the long duration of the recording sessions that are needed for constructing the database, hair ribbon tension often drifts, or is adjusted on purpose by the performer. Because of that, the calibration procedure needs to be repeated periodically, implying the design of an interpolation methodology for compensating those changes along the recording sessions. The reader is referred to (Guaus et al., 2009) for a detailed description.

2.5 Bowing parameter acquisition results

It remains difficult to formally assess the reliability of the whole acquisition methodology. Some of the partial results obtained along the calibration process (e.g., automatic estimation of the string being played) already validate accuracy and robustness, but in order to quantitatively evaluate the accuracy of some of the other magnitudes, a further test is carried out. In the test, real and measured values of bow transversal position p_b and bow-bridge distance d_{bb} are respectively compared in static conditions. For doing so, a number of ticks are marked on the strings (starting from each bridge end, having one each cm up to 5cm), and on the hair ribbon (starting from the frog end, having one each 15cm up to 60 cm). For every combination of p_b and d_{bb} average absolute errors below 0.20cm and below 0.25cm are respectively obtained for for bow transversal position and bow-bridge distance. These results prove to be sufficiently small given the intrinsic accuracy of the tracking device, and the possible error propagation happening during the computations performed as part of the calibration and measurement processes. Figure 2.16 shows acquired data for a couple of consecutive phrases.



(a) Higher pressing force (b) Lower pressing force

Figure 2.15: Bow force calibration data. The three plots at the top correspond to acquired gages deflection d_h, bow position p_b, and bow tilt angle ϕ. In the plot at the very bottom, thin and thick lines respectively correspond to the bow force acquired using the load cell, and the bow force predicted by the SVR model.

Synchronized recording of audio and bowing control data is integrated into the commercial production environment *Steinberg© Cubase* by means of using a dedicated VST[9,10] plug-in specifically developed for the purpose of this work[11] The plug-in is able to manage correction delay between data streams arriving from the bridge pickup, the motion tracker, and the strain gages conditioning circuit. It also provides specific working modes for carrying out the different calibration processes, while it gives visual feedback of captured data (see Figure 2.17).

2.6 Database construction

2.6.1 Generation of recording scripts

The creation of a satisfactory set of recording scripts represents a difficult task. Several factors need to be taken into account: playing style, context coverage, and recording session duration. As already mentioned, the playing style that is chosen is *classical violin playing*. Context coverage is understood as the variety of note articulations, durations, dynamics, and fundamental frequency (and some of their combinations within a short sequence of notes) that is covered by the set of scores. The ultimate goal of the recording script creation process is to reach a satisfying compromise between the coverage achieved and the expected

[9] VST (Virtual Studio Technology) was developed by *Steinberg Media Technologies GmbH*.

[10] `http://ygrabit.steinberg.de/~ygrabit/public_html/index.html`

[11] The plug-in is property of *Yamaha Corporation*, and the implementation details are protected by a non-disclosure agreement reached during part of the research carried out within the work presented in this dissertation.

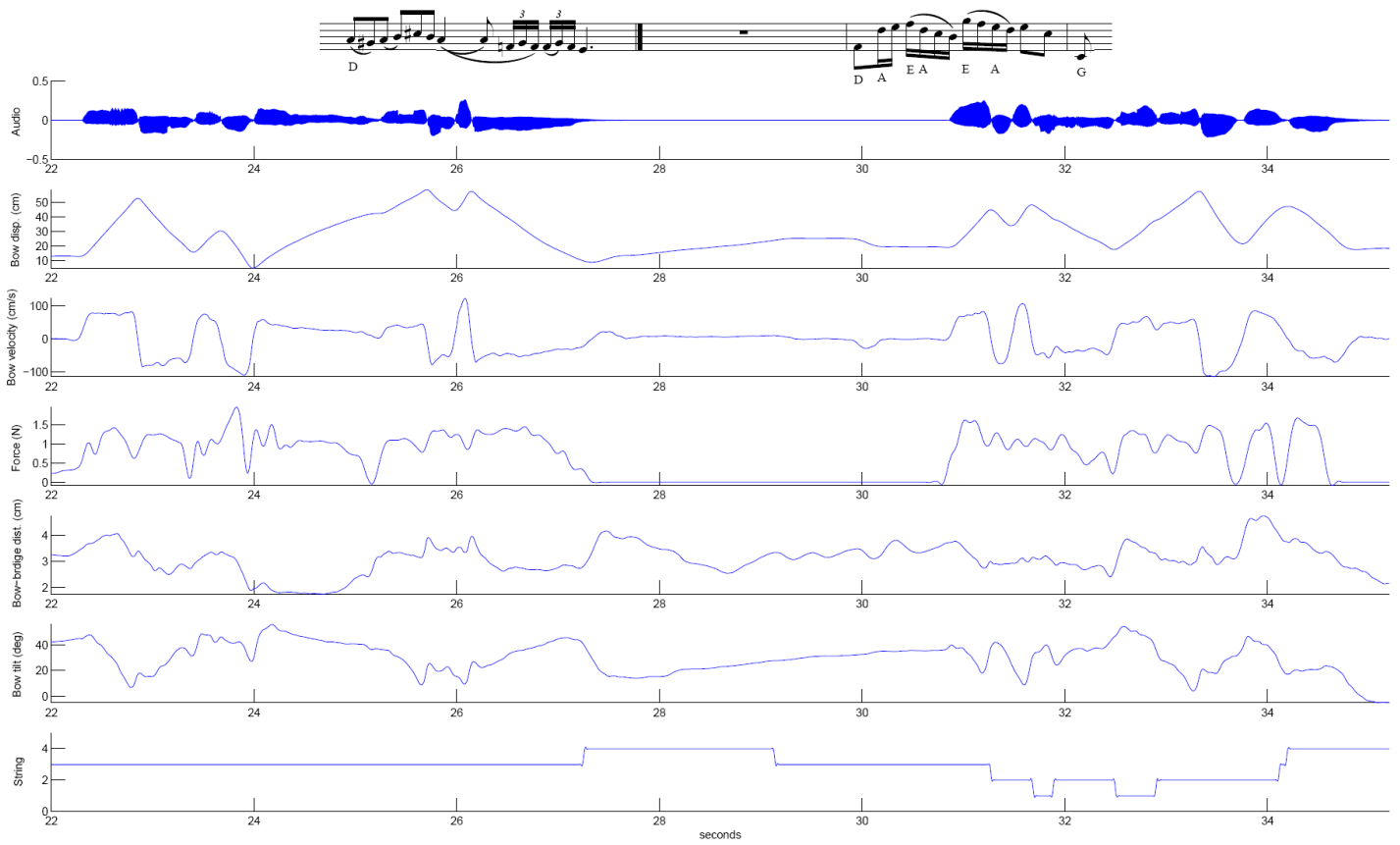


Figure 2.16: Acquired bowing control parameter contours for a pair of phrases recorded during the database construction process. From top to bottom: audio signal, bow transversal displacement p_b, bow transversal velocity v_b, bow pressing force F, bow-bridge distance d_{bb}, bow tilt angle ϕ, and estimated string (Strings E, A, D, and G are respectively represented by values 1, 2, 3 and 4.

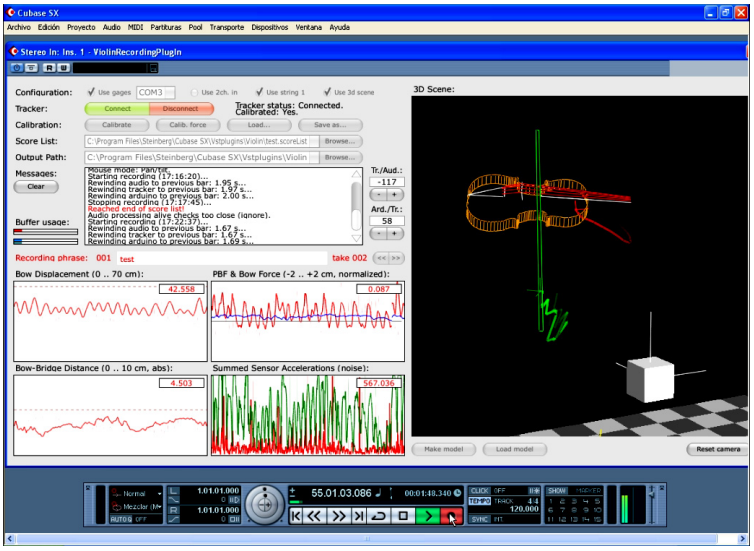


Figure 2.17: Screenshot of the VST plug-in developed for testing the acquisition process, and for carrying out synchronized recording of audio and bowing control data.

duration of the recording sessions (covering all possible contexts to appear in a piece results impossible), while maintaining a certain musical motif familiarity for the performer so that she/he is able to play more naturally.

In Figure 2.18 it is illustrated the script generation process. In order to fulfill the aforementioned constrains, a specialized software is devised for being a key component in the generation of recording scripts: the process is carried out in a semi-automatic manner by attending to context coverage basic statistic data. By starting with a basic set of classical motifs or exercises taken from existing violin pieces, the software is instructed to apply a number of transformations and generate a vast number of derived scripts including feasible variations on note duration (or tempo), string played, hand position (fundamental frequency), three different dynamics (all scripts are recorded three times: *piano*, *mezzoforte*, and *forte*), and a number of articulations or bowing techniques (only *détaché*, *legato*, *staccato*, and *saltato*[12] are used in this work[13]). The software is able to provide coverage summary information while giving an estimation of the

[12] Even though the labels chosen do not exactly distinguish or specify only articulation or bow stroke type, these four names have been used for referring to the playing techniques considered as a representative subset. For convenience in the work presented here, notes are grouped into these four *articulation types* by assuming that information about both the actual articulation or bow stroke is carried by the name given.

[13] The reader is referred to Section 3.1 for an overview of a series of considerations taken into account when deciding on the articulations or bow strokes selected for constructing the database. Such considerations have been contrasted by the literature (Galamian, 1999).

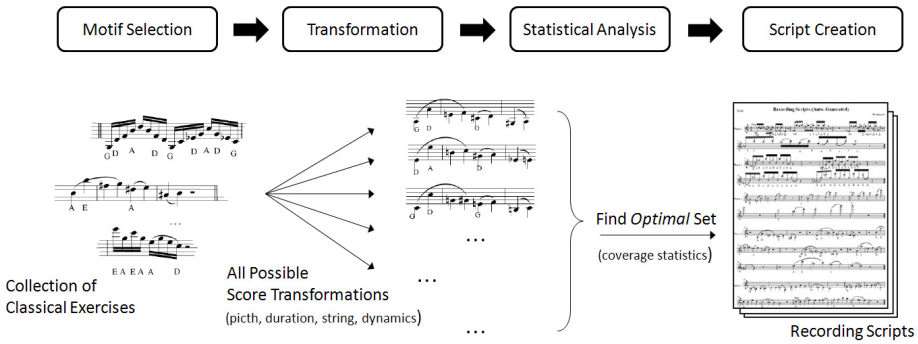


Figure 2.18: Schematic illustration of the recording script creation process.

duration of the recording session. This way, it is straightforward to drop any required subset of automatically generated scores in the search of a good length-coverage compromise.

2.6.2 Database structure

Score-based annotations, acquired audio, audio analysis files, and processed instrumental control data are stored in a structured database of files of different nature that enables the access to relevant information when required. Such file structure is depicted in Figure 2.19. For each phrase in the database, one finds three main categories of files: annotation-related files, instrumental control -related files, and audio-related files.

Annotation-related files

The set of scores to be performed is automatically parsed from its original *MusicXML* format[14,15] (provided by the script transformation process outlined in previous section) to obtain a set of text-based label files used through the database construction process, the subsequent steps along the development of the bowing control modeling framework, and for the final sample-based sound synthesis validation. Such segmentation files contain the nominal time segmentation of the different annotations present in the score. A summary is given next (scripted rests are represented by number `0`):

- **Pitch**. This label file includes a sequence the MIDI note numbers of all notes in the score.

  `[<MIDI note number>,<onset time>,<offset time>]`

[14] MusicXML was developed by *Recordare LLC.*
[15] http://www.recordare.com/xml.html

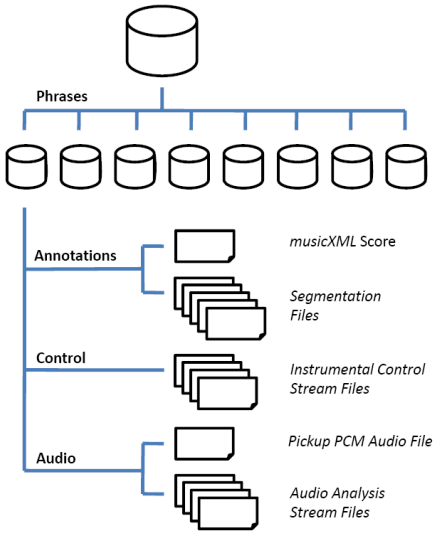


Figure 2.19: Performance database structure.

- **String**. An ordered sequence of scripted strings. Strings are represented by `E`,`A`,`D` and `G` letters.

  `[<String>,<start time>,<end time>]`

- **Dynamics**. Scripted dynamics. Again an ordered sequence of the different sections of the score where different dynamics are played. The three dynamics are represented by `pp`, `mf`, and `ff`.

  `[<Dynamics>,<start time>,<end time>]`

- **Bow direction**. The two possible bow directions are represented as `down` and `up`.

  `[<Bow direction>,<start time>,<end time>]`

- **Articulation type**. Four labels represent the four articulations considered in this study: `détaché`, `legato`, `staccato`, and `saltato`.

  `[<Articulation type>,<start time>,<end time>]`

Instrumental control -related files

Processing results of acquired sensor data are also stored into four different files, each one containing an array of values corresponding to each of the instrumental control streams. The content of all files is time-aligned and sampled at a rate $s_r = 240$Hz (corresponding to the *Polhemus Liberty* EMF position tracking

device used for capturing the motion data). The contents of each one of these files is:

- **Bow velocity**. Acquired bow transversal velocity, expressed in cm.
- **Bow force**. Processed bow pressing force, expressed in N. Before down-sampling, this signal is low-pass filtered using a 20ms-width Gaussian sliding window.
- **String played**. Estimation of the string being played, with strings `E`,`A`,`D` and `G` respectively represented by the values `1`,`2`,`3` and `4`.
- **Bow-bridge distance**. Distance from the bow hair-ribbon to the bridge (measured on the string being played), expressed in cm.

Audio-related files

Both the audio stream acquired by means of the bridge pickup, and the feature streams resulting from analyzing it are also stored into files for being accessible. Again, audio analysis results are re-sampled in order to match the tracker sampling rate $s_r = 240$Hz.

- **Bridge pickup audio signal**. Audio PCM file sampled at $f_s = 44100$Hz using 32 bits of coding depth.
- **Signal energy**. Estimated instantaneous RMS energy of the pickup signal using a 20ms Blackman Harris window at an overlap factor of 50%.
- **Estimated pitch**. Fundamental frequency estimation, expressed in cents, relative to a reference frequency $f_r = 440$Hz. This estimation, mostly based on time-domain auto-correlation, has been carried out by means of the algorithm described by de Cheveigné & Kawahara (2002).
- **Aperiodicity measure**. Measure of signal aperiodicity, again based on time-domain auto-correlation and computed by means of the algorithm described by de Cheveigné & Kawahara (2002).

2.6.3 On combining audio analysis and instrumental control data for automatic segmentation

Given the timing deviations introduced by the performer, nominal times appearing in the segmentation files are not reliable enough for carrying out the bowing control modeling research being pursued in this work. Since score-aligning the performance database results a very time-consuming task when carried out manually, the possibility of using acquired bowing data for automatically correcting onset and offset times appears as a good opportunity for speeding up the alignment process. The idea is based on attending to bow direction changes,

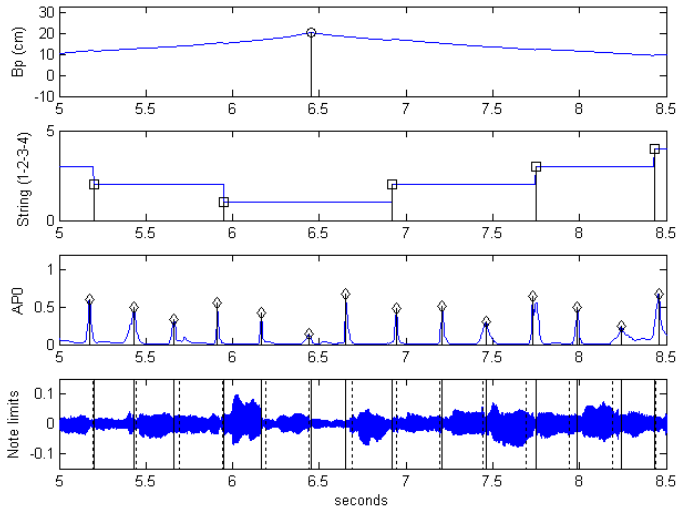


Figure 2.20: Illustration of the combination of audio analysis and instrumental control data for score-aligning recorded performances. From top to bottom: detected bow direction change times t_{bc} (circles) on top of the bow transversal position signal, string changes t_{stc} (squares) on top of the string estimation signal, and pitch transition times t_{f_0c} (diamonds) on top of the aperiodicity measure. Dashed and solid lines represent nominal and estimated transition times.

to string changes, and to the profile of some audio analysis features within a time window around the nominal note transition times of performed scores.

First, a window $\omega_{tc_i} = \{tc_i - \Delta_{l,i}\ ,\ tc_i + \Delta_{r,i}\}$ is defined around each transition time tc_i found in the score transcription. Assuming that the performer deviations on transition times do not exceed half the duration of their respective preceding and subsequent notes, the left-side width $\Delta_{l,i}$ and the right-side width $\Delta_{r,i}$ of ω_{tc_i} are defined accordingly. Then, within the time window ω_{tc_i}, an estimation of the times of three relevant events potentially indicating a note transition are carried out. First, the time t_{bc} of an eventual bow direction change is obtained from any zero-crossing of the bow velocity contour. Similarly, any string change time t_{stc} is derived from localizing eventual steps in the detected string signal (see Section 2.3.2). A potential pitch transition time t_{f_0c} (implying a note transition) is obtained from the local maximum of the aperiodicity measure signal. Due to the possibility of having three detections in one window, the estimated note transition time is assigned to one of the detected event times by giving first priority to the bow direction changes t_{bc}, followed by string changes t_{stc}, and finally by pitch transitions times t_{f_0c}. In Figure 2.20, the detections obtained from a recording excerpt are shown.

After carefully observing preliminary estimated transition times for different

excerpts, it is learned that a more sophisticated approach is needed. Considerable asynchrony between detected bow direction or string change times, and the actual note transition times often appears. Also, significant transition time estimation errors happen in notes preceded by or followed by rests (onset and offset times respectively).

A couple of major problems are found regarding the use of detected bow direction changes for estimating note transition times. For the case of *saltato* articulation, the bow direction is changed while the bow is in off-string conditions, leading to a spurious *pre-onset* detections. For the case of successive *staccato* notes, the bow remains stopped on top of the string for a considerable amount of time, so the estimation of a unique bow direction change gets difficult.

Detected string change times show in general to be more reliable, although they do not serve as the only basis for transition time estimation. Also, maxima of the aperiodicity measure contour does not represent a good source for estimating transition times by itself when, for instance the bow remains stopped between notes (e.g., in *staccato*), and no periodicity is observed for a certain amount of time (thus impeding the localization of a clear maximum of aperiodicity). Moreover, rest segments often suffer a significant duration variation with respect to scripted nominal times, so the limitations imposed by defining a time window around nominal transition times leads to bad segmentation results when involving rest segments, specially when these appear at the beginning or at the end of phrases.

A number of non-positive conclusions is drawn from the results obtained by these preliminary tests on combining audio and instrumental control data for segmenting the database in a simple way. Consequently, a more effective and reliable score-performance alignment procedure is devised in such a way that extended audio analysis data are used, bow direction changes are not considered, and robustness against time deviations is improved.

2.6.4 Score-performance alignment

With the aim overcoming the limitations pointed out previously, automatic score-performance alignment is approached by means of dynamic programming techniques, leading to much more robust results that need to be manually corrected only in very few occasions. The problem of aligning N notes is set as finding an optimal transition frame index matrix K^* defined by an array of N frame index pairs $\mathbf{k}_n^*$ each one corresponding to the optimal beginning and ending frame indexes of a n-th note of the performed score (see equations (2.10) and (2.11), where the super-indexes L and R respectively denote onset and offset frame indexes). Scripted rest segments are considered as note segments.

$$\mathbf{k}_n^* = \{k_n^{L*}, k_n^{R*}\} \tag{2.10}$$

$$K^* = \left[\, \mathbf{k}_1^*, \ldots, \mathbf{k}_n^*, \ldots, \mathbf{k}_N^* \,\right] \tag{2.11}$$

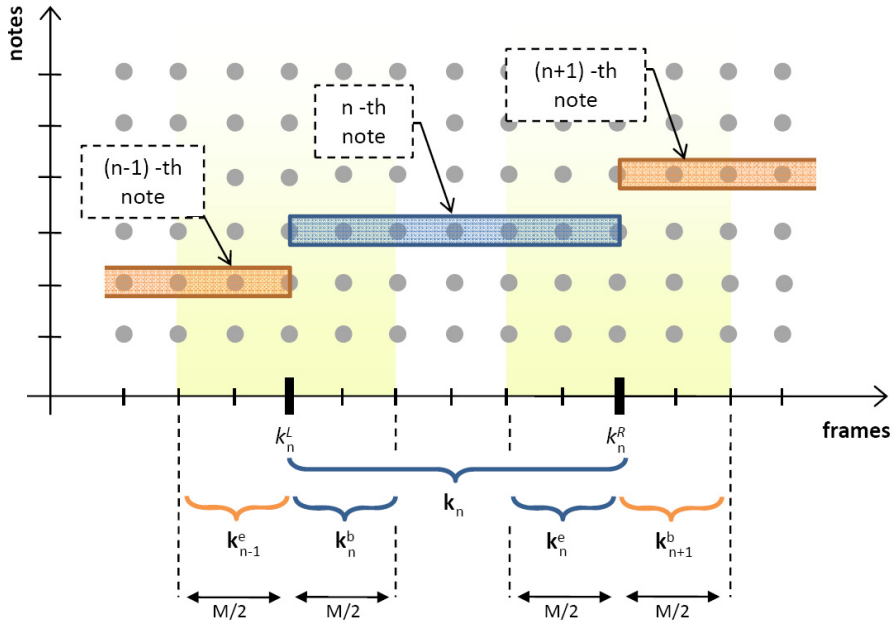


Figure 2.21: Illustration of the relevant segments involved in the automatic score-performance alignment algorithm. The regions where transition costs are evaluated appear highlighted.

Each offset frame index k_n^R of each n-th note must coincide with the onset frame index k_{n+1}^L of its successor $(n+1)$-th note in order to ensure time continuity.

$$k_n^R = k_{n+1}^L \;\; \forall n = 1,\ldots,N-1 \tag{2.12}$$

The dynamic-programming procedure used for alignment is based on the *Viterbi* algorithm (Viterbi, 1967), and focuses into three main regions of each note: the note body and two transition segments (onset and offset). Different costs are computed for each of the different segments, and the optimal note segmentation K^* is obtained so that a total cost (computed as the sum of the costs corresponding to the complete sequence of note segments) is minimized.

In order to make clear how the costs are computed, the different segments have been illustrated in Figure 2.21. Given a pair $\mathbf{k}_n$ of transition frame indexes of an n-th note in the score, its onset transition region $\mathbf{k}_n^b$ is defined around the candidate onset frame index k_n^L as expressed in equation (2.13), where M is a parameter defining the width of the transition segment from which transition-related costs will be computed. The offset transition region $\mathbf{k}_n^e$ is defined analogously:

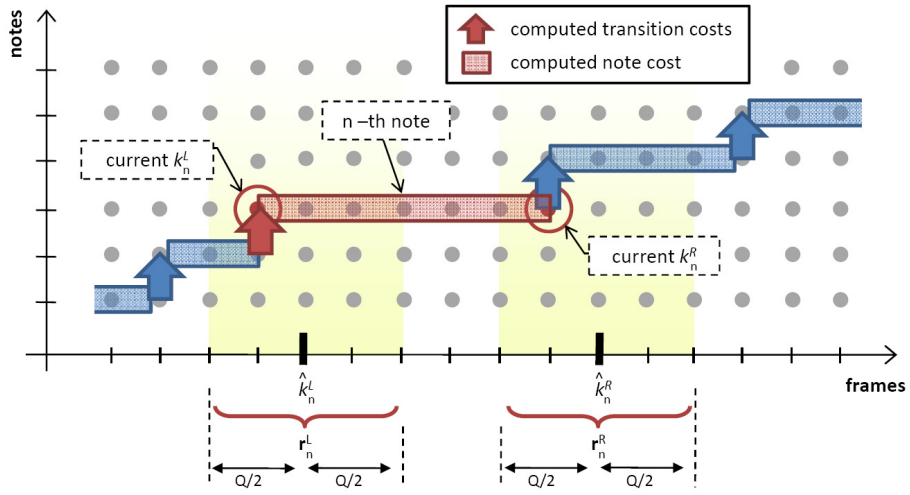


Figure 2.22: Schematic illustration of one of the steps of the dynamic programming approach that is followed for carrying out automatic score-performance alignment. The regions corresponding to candidate onset and offset frame indexes for an n-th note appear highlighted.

$$\mathbf{k}_n^b = \{k_n^L, k_n^L + \tfrac{M}{2}\} \tag{2.13}$$

$$\mathbf{k}_n^e = \{k_n^R - \tfrac{M}{2}, k_n^R\} \tag{2.14}$$

Next it is introduced the computation of the different costs involved in the alignment procedure. As pointed out before, the task consists on finding the optimal segmentation matrix K^* that minimizes a total cost $C(K)$, as expressed in (2.15).

$$K^* = \underset{K}{\operatorname{argmin}}\, C(K) \tag{2.15}$$

Alignment cost computation

The total cost $C(K)$ is computed as the weighted sum of the note body costs $C_\eta(\mathbf{k}_n)$ and the onset and offset transition costs $C_{\tau b}(\mathbf{k}_n^b)$ and $C_{\tau e}(\mathbf{k}_n^e)$ attached to all notes in the sequence. At each step of the *Viterbi* algorithm, an optimal onset frame index k_n^L is found for each candidate offset frame index k_n^R of each note so that a partial cost (up to the n-th note) is minimized. One of the steps of the algorithm is illustrated in Figure 2.22. For every n-th note, two sets of candidate onset and offset frame indexes are defined as those respectively falling within

an onset candidate region $\mathbf{r}_n^L$ and an offset candidate region $\mathbf{r}_n^R$, as expressed in equations (2.16) and (2.17), where $\hat{k}_n^L$ and $\hat{k}_n^R$ respectively represent the nominal onset and offset frame indexes (from the nominal times of the performed score), and Q is a deviation parameter[16] used for defining the candidate regions by attending to the duration of the notes involved.

$$\mathbf{r}_n^L = \{\hat{k}_n^L - \tfrac{Q}{2}, \hat{k}_n^L + \tfrac{Q}{2}\} \tag{2.16}$$

$$\mathbf{r}_n^R = \{\hat{k}_n^R - \tfrac{Q}{2}, \hat{k}_n^R + \tfrac{Q}{2}\} \tag{2.17}$$

Let's introduce now the formulation of the total cost $C(K)$, expressed in equation (2.18). Because of the partial cost computation taking place in the algorithm, the total cost $C(K)$ is defined as an addition of three different terms.

$$\begin{aligned} C(K) \quad = \quad & w_{\tau b} C_{\tau b}(\mathbf{k}_1^b) + w_\eta C_\eta(\mathbf{k}_1) + \\ & \sum_{n=2}^{N} \left(w_{\tau e} C_{\tau e}(\mathbf{k}_{n-1}^e) \; + \; w_\eta C_\eta(\mathbf{k}_n) \; + \; w_{\tau b} C_{\tau b}(\mathbf{k}_n^b) \right) + \\ & w_{\tau e} C_{\tau e}(\mathbf{k}_N^e) \end{aligned} \tag{2.18}$$

The first term accounts for an onset cost $C_{\tau b}$ evaluated at the first note's onset transition region $\mathbf{k}_1^b$, weighted by an onset transition cost weight $w_{\tau b}$; and for a note body cost C_η evaluated at the region defined by the candidate onset and offset frame index pair $\mathbf{k}_1$, and weighted by a note body weight w_η. The second term is a weighed sum of the costs computed for the rest of notes in the score. For each note, its body cost C_η is evaluated at $\mathbf{k}_n$ and weighted by w_η, its onset transition cost C_τ is evaluated at $\mathbf{k}_n^b$ and weighted by the onset transition cost $w_{\tau b}$, and its preceding note's offset transition cost $C_{\tau e}$ is evaluated at $\mathbf{k}_{n-1}^e$ and weighted by the offset transition cost $w_{\tau e}$. The third term consists on the offset transition cost C_τ of the last note, and is evaluated at $\mathbf{k}_N^e$ and weighted by $w_{\tau e}$.

Note body cost computation

Each note body cost C_η is broken down into a weighted sum of different note-related sub-costs (see equation (2.19)). Depending on whether the note segment corresponds to a scripted rest or to a note in the original score, a different computation is used.

$$C_\eta(\mathbf{k}_n) = \begin{cases} w_D C_D(\mathbf{k}_n) \; + \; w_e C_{e^-}(\mathbf{k}_n) \; + \; w_E C_{E^-}(\mathbf{k}_n) & \text{if silence,} \\ w_D C_D(\mathbf{k}_n) \; + \; w_E C_{E^+}(\mathbf{k}_n) \; + \; w_{f_0} C_{f_0}(\mathbf{k}_n) \; + \; w_a C_{a^-}(\mathbf{k}_n) & \text{otherwise.} \end{cases} \tag{2.19}$$

[16] Q is used for speeding up the algorithm, assuming an upper bound for the time deviations occurring during performance.

Let f_d be the number of frames of the note segment into consideration (defined as $f_d = k_n^R - k_n^L + 1$) and used through the definition of all sub-costs:

- **Duration**. A note duration cost is defined as in (2.20), where $\hat{f}_D$ is the number of frames corresponding to the nominal duration of the note, and σ_D is defined as $0.7\hat{f}_D$.

$$C_D = 1 - e^{-\left(\frac{(f_d - \hat{f}_D)^2}{\sigma_D^2}\right)} \tag{2.20}$$

- **Low Excitation**. It is computed as (2.21), where f_e is the number of frames for which an excitation measure $e(k)$ defined as $e(k) = |v_b(k)|\, F_b(k)$, with $v_b(k)$ and $F_b(k)$ respectively being the acquired bow velocity and bow force at the k-th frame, is above a threshold e_{th} that is tuned manually.

$$C_{e^-} = \frac{f_e}{f_d} \tag{2.21}$$

- **Low Energy**. Computed as (2.22), it relates the total number of frames f_d to the number of frames f_E for which the energy $E_{XX}(k)$ of the pickup audio signal is above a threshold E_{th} that is tuned manually.

$$C_{E^-} = \frac{f_E}{f_d} \tag{2.22}$$

- **High Energy**. Complementary to the *Low Energy* cost, this penalizes low energy segments. It is computed as (2.23).

$$C_{E^+} = 1 - \frac{f_E}{f_d} \tag{2.23}$$

- **Fundamental Frequency**. A fundamental frequency cost C_{f_0} is computed as the weighted average of the likelihoods of the fundamental frequency $f_0(k)$ of the f_d frames, given the expected fundamental frequency of the note. This is expressed in (2.24), where μ_{f_0} corresponds to the expected fundamental frequency of the note (expressed in cents), and $\sigma_{f_0} = 300$cents. The weights w_k applied to each frame are configured in a window fashion so that they sum up to one while giving less importance to the frames close to the edge frames k_n^L and k_n^R.

$$C_{f_0} = 1 - \frac{\sum_{k=k^L}^{k^R} w_k e^{-\left(\frac{(f_0(k) - \mu_{f_0})^2}{\sigma_{f_0}^2}\right)}}{f_d} \tag{2.24}$$

- **Low Aperiodicity**. It is defined by (2.25), where f_a corresponds to the number of frames for which an aperiodicity measure $a(k)$ (computed as described in de Cheveigné & Kawahara (2002)) is above a threshold a_{th} that is tuned manually.

$$C_{a^-} = \frac{f_a}{f_d} \tag{2.25}$$

Transition costs computation

The onset transition cost $C_{\tau b}$ and offset transition cost $C_{\tau e}$ are broken down into a weighted sum of several sub-costs:

$$C_{\tau b}(\mathbf{k}_n^b) = w_s C_s(\mathbf{k}_n^b) + w_e C_{e^+}(\mathbf{k}_n^b) + w_{\Delta E} C_{\Delta E^+}(\mathbf{k}_n^b) + w_a C_{\Delta a^-}(\mathbf{k}_n^b) \tag{2.26}$$

$$C_{\tau e}(\mathbf{k}_n^e) = w_s C_s(\mathbf{k}_n^e) + w_e C_{e^-}(\mathbf{k}_n^e) + w_{\Delta E} C_{\Delta E^-}(\mathbf{k}_n^e) + w_a C_{\Delta a^+}(\mathbf{k}_n^e) \tag{2.27}$$

Both transition costs are set to zero if the note segment corresponds to a scripted rest in the performed score. Again, let f_d be the number of frames of the region, defined as $f_d = M/2$.

- **String Change**. A string change cost is defined in order to account for eventual string changes happening in the transition region. Depending on whether the segment into consideration corresponds to an onset ($\mathbf{k_n^b}$) or to an offset ($\mathbf{k_n^e}$), the string change cost is computed differently, as it is expressed in (2.28). In the equations, k^s represents the frame index where a string change is detected (see previous Sections for details). If no string change is detected within the interval, the cost is set to $C_s = 1$.

$$C_s = \begin{cases} \frac{2|k^s - k_n^L|}{M} & \text{if onset,} \\ \frac{2|k_n^R - k^s|}{M} & \text{if offset.} \end{cases} \tag{2.28}$$

- **High Excitation**. It is computed as (2.29), where f_e corresponds (as for the *High Excitation* cost) to the number of frames for which an excitation measure $e(k)$ defined as $e(k) = |v_b(k)|\, F_b(k)$, with $v_b(k)$ and $F_b(k)$ respectively being the acquired bow velocity and bow force at the k-th frame, is above a threshold e_{th} that is tuned manually.

$$C_{e^+} = 1 - \frac{f_e}{f_d} \tag{2.29}$$

- **Low Energy Derivative**. This cost penalizes low decrease rates of the audio energy at note offsets (see (2.30), where $f_{\Delta E}$ is the number of frames

for which the energy derivative $\Delta E(k)$ (computed over a 100ms sliding window) is above a threshold ΔE_{th} that is tuned manually.

$$C_{\Delta E^-} = \frac{f_{\Delta E}}{f_d} \tag{2.30}$$

- **High Energy Derivative**. Complementary to the *Low Energy Derivative* cost, energy increase at note onsets lowers this cost:

$$C_{\Delta E^+} = 1 - \frac{f_{\Delta E}}{f_d} \tag{2.31}$$

- **Low Aperiodicity Derivative**. In order to favor aperiodicity decrease at note onsets, this cost is computed as (2.30), where $f_{\Delta a}$ is the number of frames for which the aperiodicity derivative Δa (computed over a 100ms sliding window) is above a threshold Δa_{th} that is tuned manually.

$$C_{\Delta a^-} = \frac{f_{\Delta a}}{f_d} \tag{2.32}$$

- **High Aperiodicity Derivative**. This cost is complementary to high aperiodicity derivative cost, and it favors aperiodicity increase at note offsets (see (2.33).

$$C_{\Delta a^+} = 1 - \frac{f_{\Delta a}}{f_d} \tag{2.33}$$

Results

While proving to be reliable for the vast majority of cases, and therefore enabling the acceleration of the database creation process (see Figure 2.23 for an example of the obtained segmentation), results coming from the automatic score-alignment algorithm need to be confirmed in order to avoid inconsistencies when using the obtained segmentations in a gesture modeling context. For doing so, nominal note durations are compared to the duration of performed notes, respectively extracted from the musical scores and from the alignment results. Whenever an important duration difference appears, a flag is set so that the segmentation of that recording is manually corrected in a post-processing step.
As it can be observed in the figure, performance onset/offset times often happened to appear ahead of nominal times for the last phrases of a set of recordings. One of the main reasons for such phenomenon resides on the tiredness of the performer after studio sessions lasting several hours.

Among the possible mistakes made by the performer (with respect to the specific annotations present in the input score to be played), wrong choices of

bow direction or string stand out. In order to overcome them and avoid repeating recordings (otherwise, mismatches between acquired data and annotations would appear), an automatic correction procedure is performed as a last step. Once the segmentation process is finished, annotations of bow direction and string played are automatically corrected by picking the corresponding maxima of histograms constructed from the acquired data (e.g., sign of the bow velocity, and estimated string) within the performance limits of each note.

2.7 Summary

This chapter gave details on methods for real-time acquisition of violin instrumental gesture parameters. A number of previous works already addressed the capture of bowing control parameters from real violin playing using different techniques (Askenfelt, 1986, 1989; Paradiso & Gershenfeld, 1997; Goudeseune, 2001; Young, 2002, 2007; Schoonderwaldt et al., 2006; Demoucron & Caussé, 2007; Demoucron, 2008). In terms of motion-related parameters, the main contribution resides on tracking the positions of both the ends of the strings and the ends of the hair ribbon, and on constructing a thorough calibration methodolgy for (1) estimating bow position, bow velocity, bow-bridge distance, bow tilt, bow inclination, and (2) automatically detecting the string being played. The implementation, built on top of a commercial tracking device based on electromagnetic field sensing providing 6DOF with high accuracy and sample rate, led to an acquisition framework that outperforms previous approaches (Askenfelt, 1986, 1989; Paradiso & Gershenfeld, 1997; Goudeseune, 2001; Young, 2002; Schoonderwaldt et al., 2006; Young, 2007) in terms of intrusiveness (only two small-size, wired sensors are present, one attached to the wood of the bow and the other attached to the violin back plate), portability (it allows the musician to use her own instrument, and it does not require a complex setup), and robustness (it provides accurate estimations in a straightforward manner and does not need of complex post-processing).

As already pointed out at the beginning of this chapter, the suitability of acquiring bowing parameters from tracking the positions of the ends of strings and hair ribbon was proved by a later work by Schoonderwaldt & Demoucron (2009) in which authors pursue an implementation of similar ideas (motion-related parameters from position tracking, and string detection from relative orientations), but adapted to a more complex, less portable, and more expensive position tracking setup based on infrared cameras.

For the case of bow pressing force, the measurement techniques described here are inspired in previous works by Demoucron & Caussé (2007); Demoucron (2008); Demoucron et al. (2009). The principal contributions or improvements are the following. Firstly, two strain gages are attached to opposite sides of bending plate, providing double sensibility and allowing for temperature compensation. And secondly, an incremental calibration procedure is devised in order to compensate bow tension and temperature changes happening during

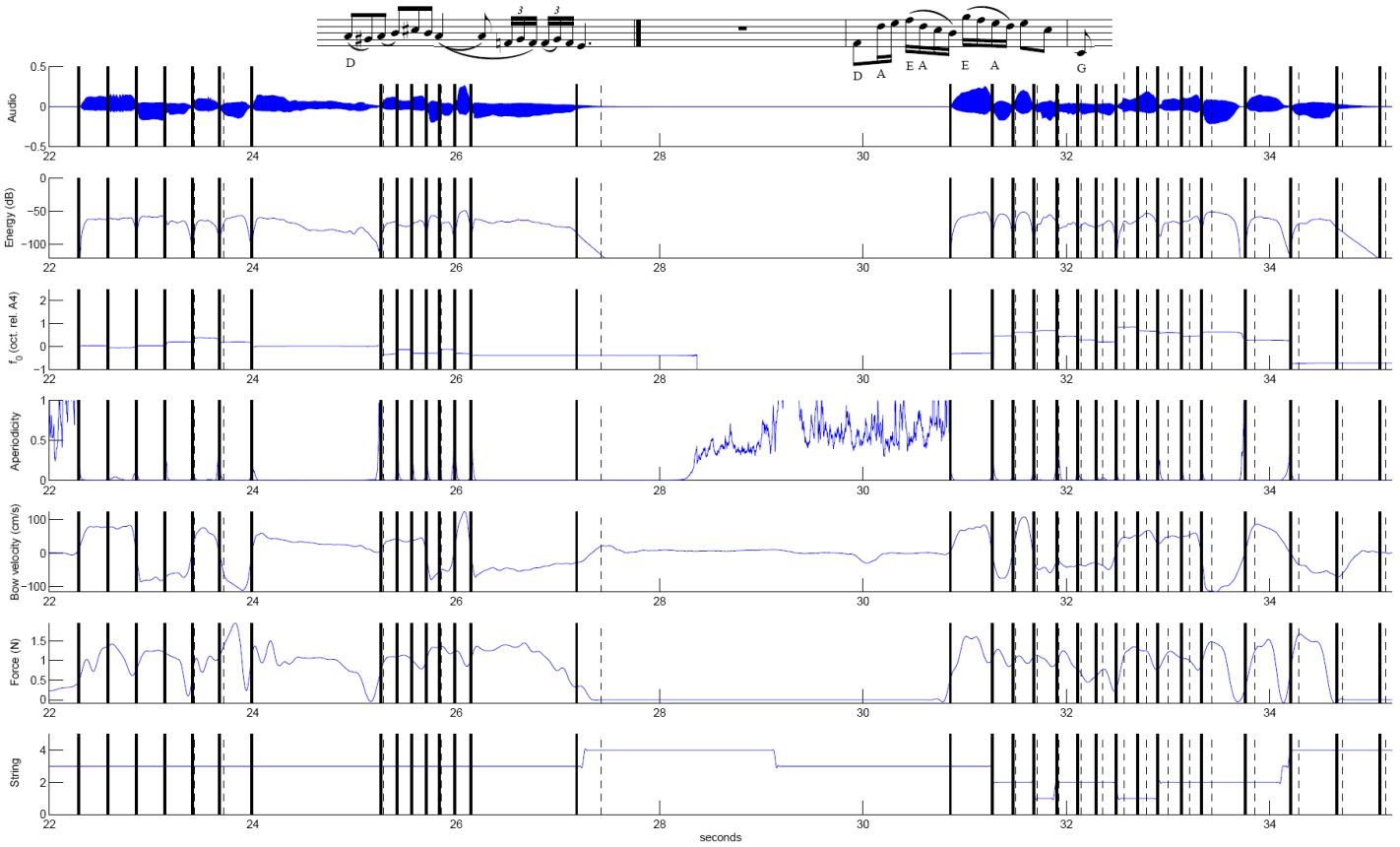


Figure 2.23: Results of the score-alignment process for a pair of recorded phrases. From top to bottom: audio signal, audio energy, estimated f_0, aperiodicity measure a, bow transversal velocity v_B, bow pressing force F_B, and estimated string. Vertical dashed and solid lines respectively depict nominal and performance transition times.

long recording sessions.

The last part of the chapter was devoted to the description of a score-performance alignment agorithm. The algorithm, based on dynamic programming techniques, makes use of both acquired instrumental gesture parameters and audio descriptors for automatically providing note onset/offset times of a considerable amount of violin performance recordings in which both audio and gesture data are acquired synchronously. Along with annotations and segmentation times, acquired data streams are stored in a performance database that is used for pursuing further analysis of bowing gestures, as detailed in the next chapter.

Chapter 3

Analysis of bowing parameter contours

This chapter introduces the analysis of bowing parameter contours as a first important step towards the development of a computational model for bowing control. First, a number of preliminary considerations are oulined, dealing with particular aspects of the performance database as related to the analysis to be carried out. Then, a contour qualitative analysis of acquired bowing parameters is presented, uncovering the principal reasons behind the chosen approach for consistently representing contours of bowing parameters. Finally, the representation framework is introduced, along with an algorithm devised for automatically obtaining quantitative descriptions of bowing parameter contours.

The author's publications related to the contents of this chapter of the disseration are time-aligned with the developments of the representation framework. In a first publication (Maestre & Gómez, 2005) it is introduced a basic coding scheme (linear segments) for modeling energy and fundamental frequency contours from saxophone audio recordings, represents a first attempt to quantitatively representing time-evolving sound-related features. Closer to the contents of this chapter, two more publications (Maestre et al., 2006; Maestre, 2006) respectively address the use of Bézier curves for statistical analysis of note articulations in singing voice performance, and the proposition of a framework (based on Bézier curves) fro modeling violin bowing parameters. The most recent publications (Maestre, 2009; Maestre et al., 2010) report on data-driven analysis of bowing parameter contours in violin performance, and represent an important part of the work to which this chapter is devoted.

3.1 Preliminary considerations

Once the acquisition of bowing parameters is carried out, a first and crucial step is to examine parameter contours with the aim of foreseeing an appropriate

framework for quantitatively representing them. One of the main premises is that the representation framework must be flexible and general, so that it can be used to parameterize bowing control patterns found in different playing contexts (duration of notes, bow stroke type, articulation type, bow direction, etc.). When constructing the database used as source material for carrying out this research, a subset of playing techniques and contexts was chosen so that it resulted representative enough for covering a variety of musically meaningful motifs and playing patterns that could serve as a test-bed for proving the flexibility and potential of the modeling framework. It is of course not the intention of the analysis presented in this dissertation to reduct the vast and complex assortment of violin playing techniques and terms (Galamian, 1999; Garvey & Berman, 1968) to only the ones treated and referred here. Instead the aim is to prove the feasibility of approaching the quantitative representation and modeling of bowing control by selecting a subset of bowing patterns or techniques and, by mostly looking into acquired bowing parameters, devise a common and flexible representation scheme that (1) successfully applies to them, and (2) can be extended to other bowing techniques and playing contexts. The eventual application of the representation scheme to other excitation-continuous musical instruments, though interesting, is out of the scope of this work.

By looking into the literature (Garvey & Berman, 1968), one realizes how difficult it is to find total agreement on the terminology and definition of existing violin bowing techniques, bow stroke types, or note articulations. Therefore, it remains hard to decide on a theoretical or foundational basis that serves as a starting point for approaching the analysis of bowing control. In fact, it is impossible to find quantitative descriptions of bowing patterns, having rather qualitative definitions as the only source of foundational information. This is indeed justified by the *practical* nature of violin (or any other instrument) playing.

The case of human speech represents a more than appropriate analogy. During the process of learning spoken language, having the only reference of a finite set of accustomed written symbols (letters) and a number of basic rules defining how to combine them (spelling or syllabling) into a higher-level entity (word) is far from providing a complete and unique method that solves the 'easy' task of pronunciation. It is in fact a 'learn-by-example' approach that makes it possible, having the conventionalism of written language derived as an 'accepted' mean for -roughly speaking- exchanging and storing messages. Hence, and going back to violin playing, the objective of this research pursuit to obtain *non-existing* quantitative representations[1] of bowing parameters by observing data, so that a direct relation between musical scores (the written messages) and bowing patterns (the actual pronunciation) can be defined and generalized to some extent. It is therefore important, in the context of a data-

[1] This kind of representations exists in some *excitation-instantaneous* instruments, like it is the case of tablatures for guitar playing.

driven representation pursuit like the one being introduced in this dissertation, to agree on the definition a basic set of score symbols (letters) and contextual rules (spelling or syllabling) to which start relating acquired and observed data.

3.1.1 Bowing techniques

One of the most important aspects to which a performer pays attention when playing a given score is the type of bow stroke used for playing notes, and how notes are articulated. It is sometimes difficult to completely separate definitions for bow strokes from definitions of articulations (Galamian, 1999), since they often appear linked in the context of a particular bowing technique. When constructing the database, four labels were given to the four bowing techniques that were selected to form a representative subset of those appearing in violin classical performance. Even though not necessarily referring only to an articulation or to a bow stroke per se, the four playing techniques have been labeled here as 'articulations'. The four labels are *détaché*, *legato*, *staccato*, and *saltato*. While *détaché* and *legato* can be considered more as *sustained-excitation* bowing techniques, *staccato* and *saltato* present more *discontinuous-excitation* characteristics:

- *Sustained-excitation*
  - *détaché*
  - *legato*
- *Discontinuous-excitation*
  - *staccato*
  - *saltato*

For the pair of *sustained-excitation* bowing techniques, a simplification is made when agreeing with the performer on how to distinguish *détaché* from *legato*. On one hand, any sequence of notes appearing as slurred in the annotated scores must be played with *legato* articulation. On the other, any sequence of notes not appearing as slurred in the score, and not presenting additional annotations related to ather bowing techniques (e.g., *staccato* or *saltato*), must be played with *détaché* articulation, changing the bow direction every new note in the sequence. Thus, it is excluded from the current database any *legato*-articulated pair of notes for which a bow direction change happens in between.

Regarding the two *discontinue-excitation* techniques, it is important to agree with the performer on the bow direction changes. It is decided that in a succession of non-slurred notes, the performer must change the bow direction each note. Therefore, a distinction between *staccato* (present in the current database) and *slurred staccato* (not present in the current database) can be easily made by looking into slurs appearing in the score. Similarly, a distinction between *saltato* (present in the current database) and *ricochet* (not present in the current database) can be also made. We refer the reader to the works by Galamian

(1999) and Garvey & Berman (1968) for more detailed definitions of the bowing techniques considered here.

3.1.2 Duration, dynamics and bow direction

Notes of different durations are scripted when designing the scores. While a broader range of durations is present for the two *sustained-excitation* bowing techniques, physical constraints that are characteristic to the two *discontinuous-excitation* techniques result into a reduced note duration variety for the second group, which comprises *saltato* and *staccato*. For the case of *saltato* bowing technique, an upper bound for note duration both delimits its practicality, and explains an essential difference from *spiccato* bowing technique (not present in the current database): the hand or arm does not percuss on every note in order to produce each bounce (intentional in *spiccato*), which occurs naturally for shorter note durations through the resilience of the bow stick. Conversely, a lower bound in note duration is derived from the characteristics of *staccato*: playing sequences of very short *staccato* notes becomes impractical, having the bowing technique to otherwise become *firm staccato* (not present in the current database): in distinction to *staccato* where individual application and release of pressure is needed to produce each articulation, a reflexive, cyclical motion produces the articulations of *firm staccato*.

In terms of dynamics it also results difficult to establish an agreement on absolute levels, having dynamics defined always as relative. Because of the expected length of the recordings to be carried out, it is decided to consider only three different dynamics, so that performer deviations are minimized along the different performances[2]. The three different dynamic levels are labeled as *piano*, *mezzoforte* and *forte*.

For the bow direction, the *rule of the down-bow* was followed: downwards bow direction is used on the first beat of each measure, unless the measure begins with a rest or otherwise specified. As already pointed out when introducing the bowing techniques, a bow direction change is performed every new note, unless having successions of notes in a slur (indicated in the score).

3.1.3 Contextual aspects

Two main contextual aspects are considered here. The first is dealing with the position of notes within slur. It is clear that notes are performed differently when appearing in different positions within a slurred sequence. Given the fact that in the current database slurs are only present in the case of *legato* articulation, an explicit treatment (by paying attention to adjacent bow direction changes) is made to this bowing technique (see next Section). The other contextual aspect takes into account whether a notes are following or preceding scripted rests. When preparing the database, these two contextual aspects are considered, so

[2] During the recording sessions, scripts containing different levels of dynamics were performed in alternating order.

that a certain variety of notes appearing in different slur and silence contexts is recorded. Of course, more contextual characteristics could be taken into account, like it is the case of the preceding and following articulations, the metrical strength of a note, or higher-level features resulting from elaborated musicological analysis of performed pieces, among others.

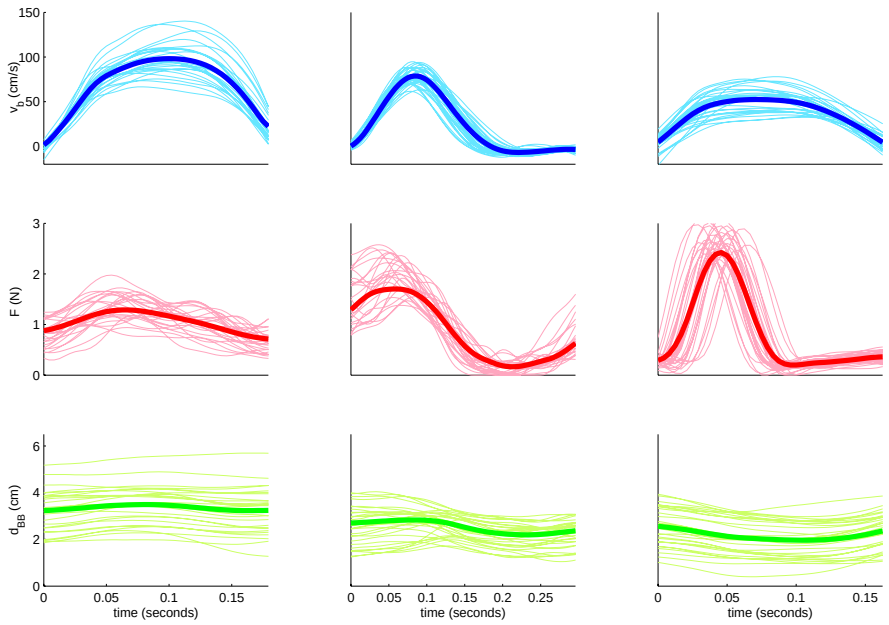
3.2 Qualitative analysis of bowing parameter contours

Contours of acquired bowing parameters must be first observed in order to devise an appropriate strategy for approaching an efficient representation that guarantees reliability in further analysis and modeling stages. As already pointed out, the objective is to find a flexible representation model that allows to quantitatively characterize the temporal envelope of the bowing control parameters that are relevant to timbre (bow velocity v_b, bow pressing force F, and bow-bridge distance d_{bb}). Such representation model must indeed set up the basis for further analyses of contours, so it is important to aim at finding sets of quantitative descriptors that can be used both for the analysis, modeling, and generation of contours (i.e., for constructing an analysis-synthesis framework based on parametric representation of contours). Hence, it results crucial to identify contour patterns that can be easily described.

Starting from the segmentation results of recorded phrases, and also from the annotation files accompanying the in the database (see Section 2.6), acquired contours are cut into note segments, having each three cut signals as a set of contour *samples*. Each note contour sample corresponds to a note in the database, so the annotations attached to it serve as the basis for the identification of different patterns. Different patterns can be associated to articulations, dynamics, bow direction, etc., so a separation based on those annotations is already revealing such association.

3.2.1 Articulation type

A first clear separation into articulation types allows to visually perceive strong differences in the temporal evolution of contours. Figure 3.1 shows the contours of various samples corresponding to notes played with *détaché*, *staccato*, and *saltato* articulations. For each of the three articulations, samples of similar durations have been chosen. It is easy to clearly identify specific patterns shown by each articulation, especially when considering bow velocity and bow force contours. Smaller differences observed in the contour of bow-bridge distance (in average, a few tenths of cm) inform about how the performer is using this parameter: it gets affected mostly at note-level, leading to different overall values for the different techniques. Note that *legato* articulation is not included in this qualitative analysis of contours. Even though *legato*-articulated notes also present a clear pattern easily recognized as different from the other articulations (a major difference is that no bow direction changes happen between them), some contextual issues make the treatment of *legato* articulation



(a) *détaché* articulation (b) *staccato* articulation (c) *saltato* articulation

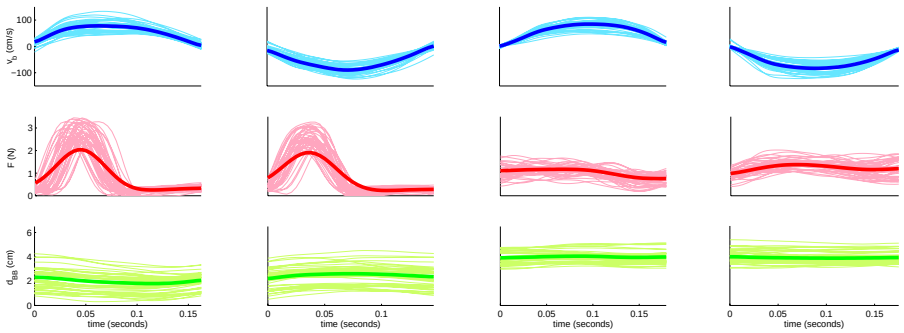
Figure 3.1: Subsets of acquired bowing parameter contours of notes performed with three different articulation types (*détaché*, *staccato*, and *saltato*), all of them played in *downwards* bow direction. From top to bottom: bow velocity v_b, bow force F, and bow-bridge distance d_{bb}. Raw contours are represented by thin curves, while mean envelopes are depicted with thick curves.

to be more complex. A short discussion on structural aspects of contours of *legato*-articulated notes is given later.

One can learn from these three sets of examples that the articulation type leads to strong, structural differences in the shape of bowing parameter contours, as it was already reported in the literature (Askenfelt, 1986, 1989; Guettler & Askenfelt, 1998; Demoucron & Caussé, 2007). This confirms the articulation type as a fundamental factor when devising a representation strategy, envisaging a possible parameterization scheme to be based on the structure observed contours.

3.2.2 Bow direction

In bowed-string instrument performance, the finite length of the bow represents a physical constrain that becomes an essential aspect. The fact that bow direction must change in the course of playing forces the musician to become capable of executing notes no matter which bow direction is used. Thus, the bowing control parameter that mostly gets affected by this is of course the bow velocity, whose contour becomes inverted (the bow velocity is negative) while maintaining its principal aspects. This is shown for *saltato* and *détaché* articulations in Figure 3.2, where the contours of acquired bowing parameters corresponding to large sets of similar notes are displayed for *downwards* bow direction (the bow tip is approaching the string, i.e., positive velocity) and *upwards* bow direction (the frog is approaching the string, i.e., negative velocity).



(a) *saltato*: *downwards* bow direction (b) *saltato*: *upwnwards* bow direction (c) *détaché*: *downwards* bow direction (d) *détaché*: *upwards* bow direction

Figure 3.2: Subsets of acquired bowing parameter contours of *saltato* and *détaché* - articulated notes, each played in *downwards* and *upwards* bow direction. From top to bottom: bow velocity v_b, bow force F, and bow-bridge distance d_{bb}. Raw contours are represented by thin curves, while mean envelopes are depicted with thick curves.

From the data displayed, one cannot perceive remarkable differences in the contour of bow force. Its pattern is maintained for both articulations, and only some changes non related to the structure of the shape can be observed especially for the case of *détaché* articulation (in *upwards* bow direction, the overall value along the contour is higher). A little more noticeable, though light is the change that suffers the contour of bow-bridge distance for the case of *saltato* articulation, going from a *fall-rise* to a *rise-fall* shape, due to the cyclic motion of the bow when performing successive bow direction changes (Hodgson, 1958; Schoonderwalt & Wanderley, 2007).

A straightforward approach to the representation of the two bow directions would be not considering them to be different, by simply keeping the absolute

value of instantaneous bow velocity. However, given the possibility that some of the little differences found in the contours of bow force and bow-bridge distance get more important, such a simplification might imply a lose of generality.

3.2.3 Dynamics

When looking into samples of notes that were played with different dynamics, it results also straightforward to observe how the performer changes certain characteristics of the bowing parameter contours. Differently to what was concluded for the case of different articulation types, playing a note with different dynamics does not lead to structural changes in the shape of bowing parameter contours, but just to some magnitude-related changes. Figure 3.3 illustrates this phenomenon for the case of *staccato* articulation by displaying subsets of acquired contours corresponding to three different dynamics, all of them performed in *upwards* bow direction. While the main pattern (see also Figure 3.1) is kept when increasing dynamics, one can identify a significant increase of the amplitude of the most prominent peak of both bow velocity and bow force contours. Also, a decrease of the overall value of bow-bridge is observed.

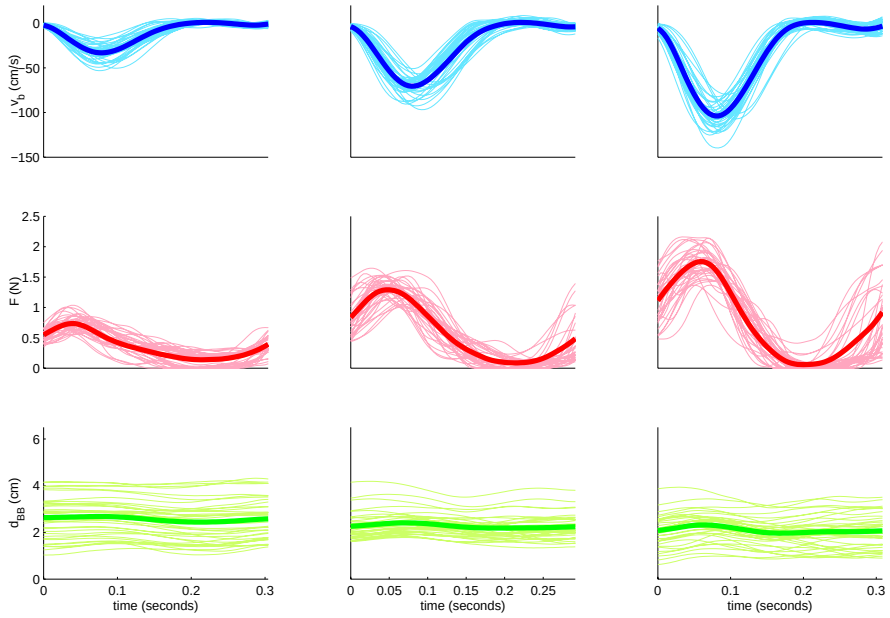
Given the fact that the performer indeed makes the difference by applying some magnitude modulations to the bowing parameters (e.g., reaching a higher value of bow velocity or bow force) while keeping the quality of the pattern, these 'obvious' dynamics-related observations are not very revealing per se. However, it is important to consider them when designing the representation model: it should allow to flexibly hold such magnitude variations within its structure-based parameterization scheme.

3.2.4 Duration

Another factor to take into account when approaching the representation problem is the duration of notes. Important conclusions can be drawn from visualizing bowing parameter contours of groups of notes of different duration. In general, non-linear time warping of bowing parameter contours happens when a note is played with different lengths. Figure 3.4 displays three groups of acquired bowing control parameters of *détaché*-articulated notes. Each group respectively contains notes of duration around $0.2s$, around $0.4s$, and $0.8s$.

If one looks at the characteristics of bow velocity, it is clearly seen how the beginning and ending segments (the portions where bow velocity is respectively *rising* and *falling*) maintain their length (between $0.05s$ and $0.1s$), while the length of inner part (the *sustained* segment) appears to be much more correlated to the note duration[3]. This indicates that the musician is indeed respecting both how the bow velocity reaches a stable value at the beginning, and how the bow velocity leaves its prior stable value by falling down to zero at the end of the note. This effect can also be perceived (though less clearly) from in the shape of

[3]Note that since the thick lines correspond to the averaged contour, these effects appear smoothed if only looking at the average behavior.



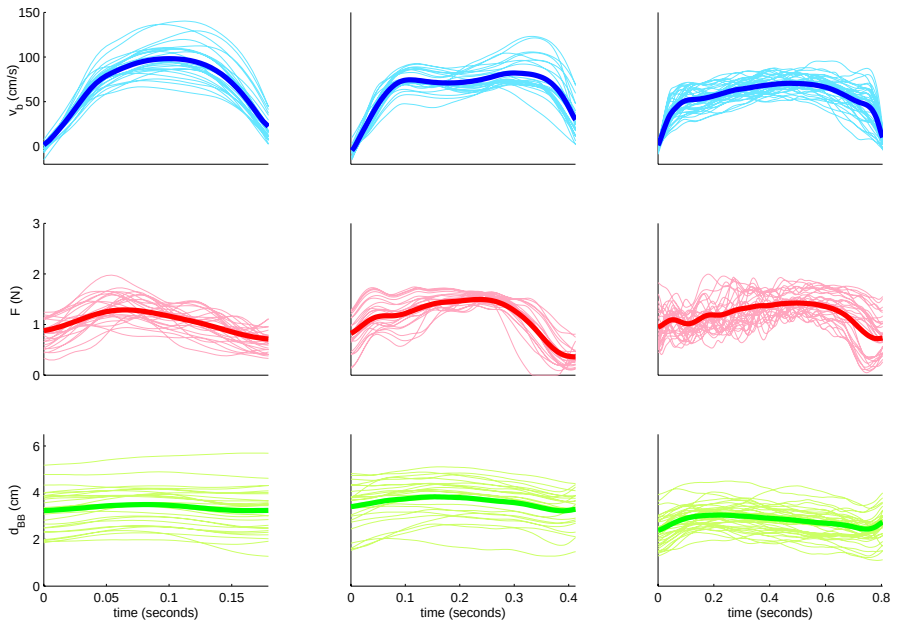
(a) *staccato* articulation: *pp* dynamics (b) *staccato* articulation: *mf* dynamics (c) *staccato* articulation: *ff* dynamics

Figure 3.3: Subsets of acquired bowing parameter contours of *staccato*-articulated notes performed with three different dynamics (*pp*, *mf*, *ff*), all of them played in *upwards* bow direction. From top to bottom: bow velocity v_b, bow force F, and bow-bridge distance d_{bb}. Raw contours are represented by thin curves, while mean envelopes are depicted with thick curves.

bow force, for which the rising part at the beginning of the note gets stretched so that it becomes longer for longer duration.

Like for the case of dynamics, structural changes are not observed when qualitatively analysing the contours of bow velocity and bow force. The main difference here is that modulations are now more importantly happening in the time axis. Given a qualitative segmentation of the contours of bow velocity and bow force into three and two meaningful parts respectively, a clear correlation can be observed between the duration of some of these parts and the duration of the note. Therefore, providing the representation scheme with flexibility enough for efficiently holding information about the time-axis modulation (time warping) is crucial.

So far, and also by going back to the discussion about structural differences observed in contours of different articulations, a representation scheme built

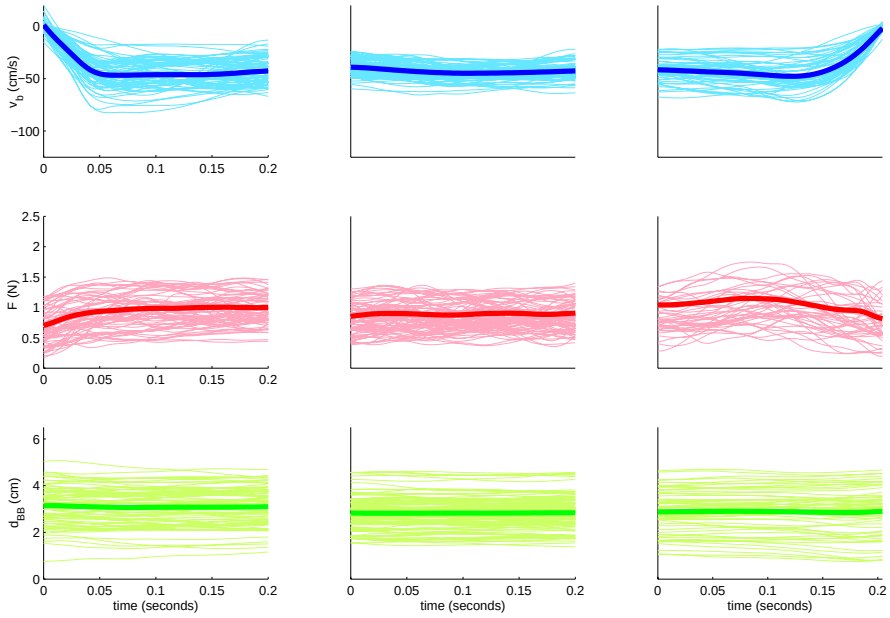


(a) *détaché*: duration group 1 (b) *détaché*: duration group 2 (c) *détaché*: duration group 3

Figure 3.4: Subsets of acquired bowing parameter contours of *détaché*-articulated notes performed with three different durations, all of them played in *downwards* bow direction. From top to bottom: bow velocity v_b, bow force F, and bow-bridge distance d_{bb}. Raw contours are represented by thin curves, while mean envelopes are depicted with thick curves.

around the idea of segmenting contours into meaningful portions becomes a promising framework, providing:

- A direct and meaningful relationship between the parameterization and the actual temporal structure of the contour (or *gesture*).
- Enough flexibility for holding information about non-structural variations (e.g., duration or magnitude modulations of the different segments of the contour) in a natural way, while respecting the original structure.
- A smooth correspondence between the representation parameters and the characteristics of the shapes, so that the robustness of the parameterization allows for pursuing further statistical analysis.



(a) *legato*: starting a slur (b) *legato*: within a slur (c) *legato*: ending a slur

Figure 3.5: Subsets of acquired bowing parameter contours of three different executions of *legato*-articulated notes (starting a slur, within a slur, ending a slur), all of them played in *upwards* bow direction. From top to bottom: bow velocity v_b, bow force F, and bow-bridge distance d_{bb}. Raw contours are represented by thin curves, while mean envelopes are depicted with thick curves.

3.2.5 Position in a slur

As pointed out before, the contours of *legato* articulations are highly dependent on the context in which the note appears in the score. More concretely, the position of the note in a slur configures the shape of the contours in very different manners. A main characteristic of *legato* articulation is a smooth transition between consecutive notes that is often achieved by not performing bow direction changes between notes. This makes a big difference (especially in the shape of the contour of bow velocity) between the first note of the slur, the last note of the slur, and the notes appearing between them. This can be clearly perceived by looking at Figure 3.5, where acquired contours are again displayed for different groups of notes, this time separating *legato*-articulated notes into notes starting a slur, within a slur, and ending slur (all of them played in *upwards* bow direction). While less important, yet considerable differences

are also observed in the contour of bow force, bow-bridge distance remains also for this case as more 'independent' on the position of the note in the slur.

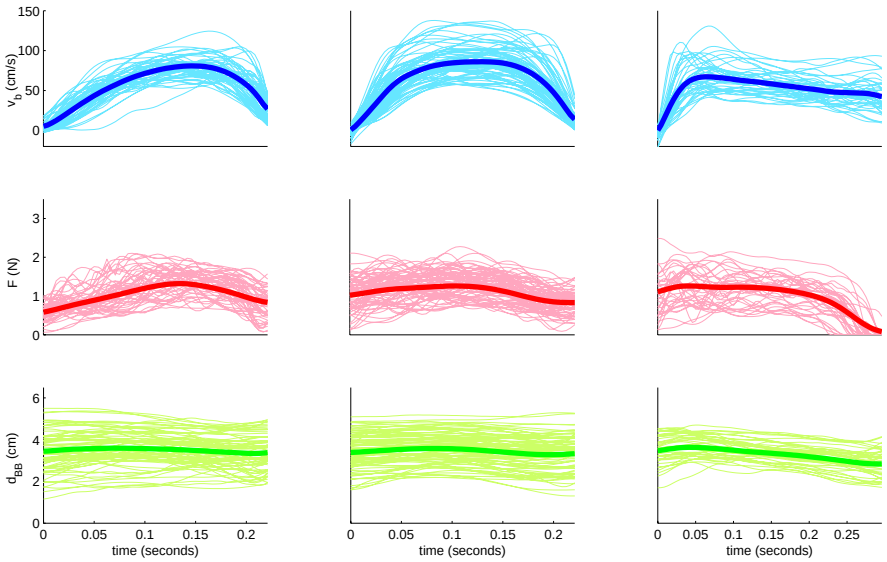
The structure of the shape of bow velocity is undoubtedly distinct in all three slur contexts. In the first context (starting a slur) the contour presents two main parts (or portions): first, the bow velocity is increasing towards a stable value (as similar to the first portion of a *détaché*-articulated note); then, once this stable value is reached, a stable stage starts, until the end of the note. A similar but converse behavior happens when the notes are ending a slur. For the case of notes within a slur which appear neither in first nor in last position, just a *sustained* segment is observed in the contour. Variations of similar nature are observed in the contour of bow force.

Again, the need of a representation scheme centered around the parameterization of the different portions or segments of a given contour stands out when dealing with contextual issues that make notes of the same articulation to present structural changes in contours of bowing control parameters. For the case analyzed here, different structures could be used for the different *slur* contexts, as it is also supported by a qualitative description the differences perceived in the shape of the bow velocity contours. As an example of such a qualitative description, when a *legato* articulated note is starting a slur, the bow velocity contour presents two main segments: during the first one, the absolute bow velocity increases until reaching a certain value; then, in the second one, the bow velocity follows a more stable behavior until the end of the note.

Following the aspects highlighted at the end of the discussion on the duration of the notes (see previous section), an appropriate contour representation scheme should not only provide means for defining a number of meaningful segments or some qualitative characteristics of them (e.g., *rising* or *falling*), but also a quantitative description of each one (e.g., the exact duration of a numeric parameterization of its shape). In some way, contours are suitable for being coded as sequences of short *units* that respond to the different 'states' (e.g., *rising*, *falling*, etc.), having a quantitative representation of each of such *units*.

3.2.6 Adjacent silences

Should a note be following or preceding a scripted silence, differences in the contours (i.e., in the execution of the notes) may consistently happen in some particular cases. By grouping *détaché* notes based on their context with respect to adjacent silences, one finds important differences, as it is illustrated in Figure 3.6 for notes of duration around $0.25s$. As an example, comparing the contours of the left column (notes following a silence) to the ones in the middle column (notes not preceded or followed by silences), a structure *rise-stable-fall* is maintained, but the characteristics and duration of each of the meaningful segments significantly change. This clear example, together to similar ones easily extracted from further qualitative comparisons of contours in Figure 3.6, illustrates the possibility of including another factor in quantitative analysis of bowing parameter contours: the context of notes regarding adjacent silences.



(a) *détaché*: following a silence (b) *détaché*: not surrounded by silences (c) *détaché*: preceded by silence

Figure 3.6: Subsets of acquired bowing parameter contours of *détaché*-articulated notes performed (a) following a scripted silence, (b) not surrounded by any scripted silence, and (c) followed by a scripted silence. All of them were played in *downwards* bow direction. From top to bottom: bow velocity v_b, bow force F, and bow-bridge distance d_{bb}. Raw contours are represented by thin curves, while mean envelopes are depicted with thick curves.

3.3 Selected approach

Attending to the conclusions drawn along the qualitative analysis outlined in previous section, contours of bow transversal velocity v_b, bow pressing force F, and bow-bridge distance d_{bb} are represented as sequences of small segments or units that configure the temporal structure of the corresponding envelopes. Segments must respond to common patterns in the instrumental control taking place in violin bowing, so the unit granularity derived from contour segmentation is in acccordance with that of bowing control. The main characteristics of each basic unit (or *building block*) can be enumerated as:

1. It follows a simple and monotonic temporal behavior.
2. It is consistent across different executions of equivalent notes.
3. It can be easily parameterized in a low-dimensional space.

An schematic illustration of the representation scheme that is devised for quantitatively analyzing contours is depicted in Figure 3.7. Contours of each note are segmented into meaningful units, and each unit is represented by small set of parameters defining its shape and duration. The number of units used for each contour is linked to its structural characteristics. Both contour segmentation and unit representation methods are crucial for ensuring consistency in the resulting quantitative parameterization.

For carrying out the quantitative analysis, the instantaneous value of bow-bridge distance d_{bb} is first transformed into a normalized ratio β, computed as the proportion between the bow-bridge distance d_{bb} and the effective length l_e of the string being played (see equation (3.1)). The length l_e is defined as the distance between the bridge and the finger position, having the finger position of each note estimated from the detected string and fundamental frequency (see Section 2.3.2).

$$\beta = \frac{d_{bb}}{l_e} \tag{3.1}$$

During the next sections, details are given on how the selected approach is implemented. Considering different articulations, dynamics, and contexts extracted from score annotations, a number of note classes (representing 'equivalent' notes) is defined based on the qualitative analysis outlined in Section 3.2. From exhaustive observation of acquired contours, constrained Bézier cubic curves are chosen as the basic representation unit. A predefined *grammar* dictating the structural properties of contours is defined for each note class (e.g., how the units are arranged), including both the number of units used for representing each contour, and a set of constraints related to the envelope of the contour. The grammar serves as a guide for both performing automatic segmentation of contours into units, and fitting segmented contours to Bézier curve parameters. Obtained curve parameters are enough for reconstructing original bowing control parameter contours with significant fidelity while ensuring representation robustness in a relatively low dimensionality space.

3.4 Note classification

Attending to different score-annotation -based characteristics of the notes in the corpus, and taking into account both the preliminary considerations introduced in Section 3.1, and the qualitative analysis outlined in Section 3.2, notes are divided into different classes for which a specific contour analysis is carried out. For each of the note classes, a specific grammar entry is defined, so that an adapted contour parameterization is performed.

In order to set up the classification basis, *intrinsic* note characteristics (based on score annotations attached to the note, see Section 2.6.2) are considered first, leaving three categories: articulation type (ART), dynamics (DYN) and bow direction (BD). In addition, two *contextual* characteristics (based on some

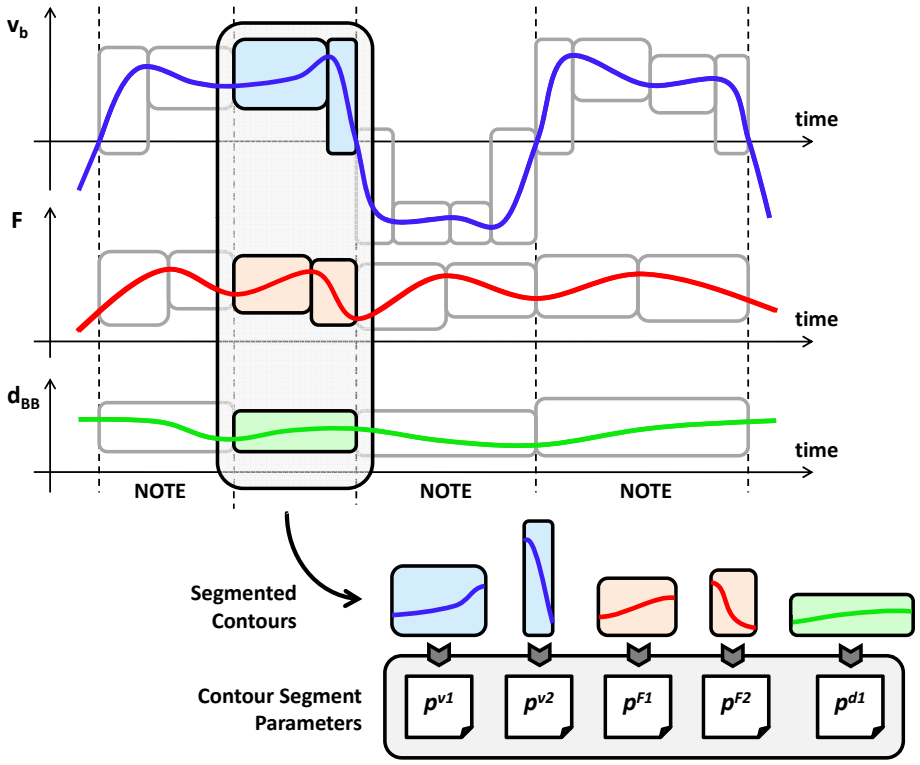


Figure 3.7: Schematic illustration of the approach selected for quantitatively representing contours of relevant bowing parameters. The numerical parameters used for describing the contours of each segment are represented as $p^b n$, with b denoting the bowing contour, and n the segment number.

characteristics of the surrounding notes) are considered: slur context (SC), and silence context (PC). The possible labels for each of the five characteristics are:

- Intrinsic characteristics:
  - Articulation type (ART):
    - *détaché*
    - *legato*
    - *staccato*
    - *saltato*
  - Dynamics (DYN):
    - *piano*

        - *mezzoforte*
        - *forte*
    - Bow direction (BD):
        - *downwards*
        - *upwards*

- Contextual characteristics:
    - Slur context (SC):
        - *init*
        - *mid*
        - *end*
        - *iso*
    - Silence context (PC):
        - *init*
        - *mid*
        - *end*
        - *iso*

Considering *intrinsic* note characteristics, first and most important is the articulation type. Four different articulation types are considered: *détaché*, *legato*, *saltato*, and *saltato*. Three different dynamics are present in the corpus: *piano*, *mezzoforte*, or *forte*. The possible bow directions are *downwards* and *upwards* (see Figures 3.1 through Figures 3.3 for examples of acquired bowing parameter contours).

In terms of *contextual* characteristics, two main aspects are considered: the context of the note within a slur (e.g., in *legato* articulation, several notes are played successively without any bow direction change), the context of the note with respect to scripted *rest* segments (e.g., silences).

For the case of slur context (SC), a note is labeled as *init* when, in a succession of notes sharing the same bow direction (e.g., a slurred sequence in *legato*-articulated notes), is played first. A note is labeled as *mid* when is played neither first nor last. The label *end* corresponds to notes played last, while notes appearing as the only notes within a slur (e.g., in *détaché* articulation) are classified as *iso*. This is illustrated in Figure 3.8, where a excerpt of a musical score showing different slur contexts is displayed.

Analogously, for the silence context (PC) it is paid attention to successions of notes with no *rest* segments or silences in between. Notes are classified as *init* when preceded by a silence and followed by another note, as *mid* when preceded and followed by a note, as *end* when preceded by a note and followed by a silence, and as *iso* those surrounded by silences. See Figure 3.9 for an illustration of an excerpt of a musical score showing different silence contexts.

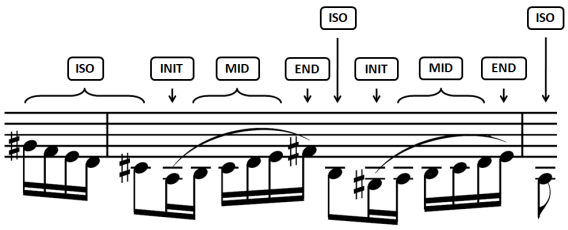


Figure 3.8: Musical excerpt corresponding to one of the performed scores when constructing the database. Regarding their slur context, different labels are given to notes.

As already mentioned, more contextual characteristics could be taken into account when building a classification scheme: preceding and following articulations (i.e., co-articulations as referred by Galamian (1999)), metrical strength, higher-level features resulting from more elaborated musicological analysis of performed pieces, etc. From the five characteristics considered in this work, each feasible combination of them leads to a note class C characterized by:

$$C = [\, ART \;\; DYN \;\; BD \;\; SC \;\; PC \,] \tag{3.2}$$

In fact, not every possible combination of the aforementioned note characteristics is feasible in practice. For instance, in the database used, the class [*legato piano downwards iso init*] is not feasible because no *legato*-articulated note was performed as being the only note in a slur. Given the nature of the articulations (i.e., bowing techniques, see Section 3.1) considered when constructing the database used in this work, constraints derived from the definition of both the slur context and the silence context lead to define a number of rules that reduce the possible combinations of articulation type (ART), slur context (SC), and silence context (PC):

- All *détaché*-articulated notes are played in *iso* slur context.
- All *staccato*-articulated notes are played in *iso* slur context.
- All *saltato*-articulated notes are played in *iso* slur context.
- No notes played in *init* slur context are played either in *end* silence context or in *iso* silence context.
- All notes played in *mid* slur context are played in *mid* silence context.
- No notes played in *end* slur context are played either in *init* silence context or in *iso* silence context.
- All notes played in *iso* silence context are played in *iso* slur context.

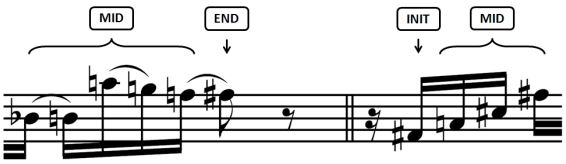


Figure 3.9: Musical excerpt corresponding to one of the performed scores when constructing the database. In terms of its silence context (SC), a different label is given to each note.

As pointed out above, this set of rules is derived from the nature of the database (see preliminary considerations of Section 3.1). Therefore, the set of rules should change whenever the characteristics of corpus change. As an example, extending the current database by including the *ricochet* bowing technique would lead to the elimination of the third rule, since several *saltato* notes could be played as grouped in the same slur (see Section 3.1.1).

Each note in the corpus is classified into one of a total of 102 classes. For each class, a specific grammar entry is defined. The grammar entries define the number of segments or units used for representing each contour, and serve as a guide for carrying out automatic segmentation and fitting (as already introduced in Section 3.3. As a result, it is obtained a segmentation of each contour into a predefined number of units, leading to a reduced set of parameters defining the temporal evolution of the contour within each unit.

3.5 Contour representation

Bowing parameter contours of recorded notes are modeled by concatenating curve segments, forming sequences of a predefined number of *units*. In particular, constrained cubic Bézier curves are chosen as the basic units. Bézier curves are extensively used in computer graphic applcations, due to their robustness and flexibility. Recent applications to speech prosody characterization, like the work by Escudero et al. (2002), showed the potential of this kind of curves for time-series modeling purposes.

On the music side, Battey (2004) used this kind of parametric curve representation in the past, when dealing with contours of perceptual audio parameter in singing voice performance. His work, although represeting an important inspiration for the choice of Bézier curves when devising a parametric representation of contours, showed a clear lack of structured schemata on the basis of consistently supporting the time constraints imposed by the performed score (e.g., score-performance alignment). Therefore, with his framework it results hard to provide a representation that is specific to one or various types of notes and consistent across different executions encountered. This makes difficult to use of numerical representation for further data analysis.

Conversely, the approach introduced in this work defines a concise representation applying at note-level. Here, performed notes are segmented, and the corresponding contours are represented by a known set of parameters directly linked to the number of basic units used for modeling them, hence enlightening the possibility of posterior statistical studies of contour parameters across different note characteristics and score annotations, and also across different performance contexts.

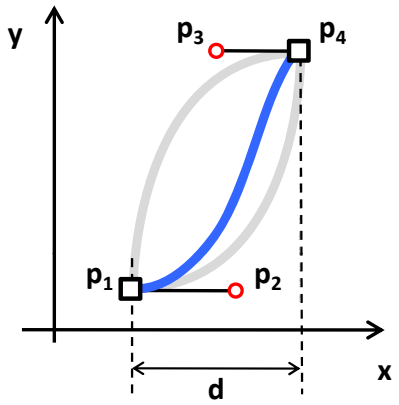


Figure 3.10: Constrained Bézier cubic segment used as the basic unit in the representation of bowing control parameter contours.

The basic unit is schematically represented in Figure 3.10. Even though it responds to a parametric cubic curve defined by the x-y points p_1, p_2, p_3, and p_4, the constrains found appearing in equations

$$p_{1y} = p_{2y} = v_s \tag{3.3}$$
$$p_{3y} = p_{4y} = v_e \tag{3.4}$$
$$r_1 = \frac{p_{2x} - p_{1x}}{d} \tag{3.5}$$
$$r_2 = \frac{p_{4x} - p_{3x}}{d} \tag{3.6}$$
$$0 \leq r_1 \leq 1 \tag{3.7}$$
$$0 \leq r_2 \leq 1 \tag{3.8}$$

allow defining its shape by the vector

$$u = [d\ v_s\ v_e\ r_1\ r_2], \tag{3.9}$$

where d represents the segment duration, v_s represents the starting y-value, v_e represents the ending y-value, and r_1 and r_2 represent the relative length of the attractors p_2 and p_3 respectively.

Among the reasons supporting the choice of this parametric curve as the building block for representing contours, it is important to emphasize two main aspects:

- It provides flexibility, so a diverse number of shapes can be represented by different values of the attractors r_1 and r_2, as it is illustrated by gray curves in Figure 3.10, which correspond to rather extreme values of r_1 and r_2.
- A smooth behavior of the shape of the curve is observed when parameters in u are changed. This way, similar values of curve parameters (control points) lead to similar shapes. This is an important advantage over working with pure cubic polynomials of the kind

$$y = a_0 + a_1 x + a_2 x^2 + a_3 x^3 \tag{3.10}$$

  for which (1) small changes in the coefficients may lead to drastic changes in the shape of the curve, and (2) there is no direct control over the edges of the curve.

The reader is referred to Appendix A (and references therein) for the basics of Bézier curves, and the polynomial parameterization behind them.

From one of the acquired contours (e.g., bow velocity), given a segment (or portion) $q(t)$ with $t \in [0,d]$, its starting value $q(0) = v_s$, and its ending value $q(d) = v_e$, optimal values of r_1^* and r_2^* leading to an optimal Bézier approximation $\sigma^*(t)$ of the segment can be found via constrained optimization (Battey, 2004). Hence, for a given bowing parameter contour, it is important to appropriately choose:

- The number of units used for representing the bowing parameter contour.
- The required duration (e.g., temporal segmentation) of each unit.

It is then a clear next step to provide means for automatically segmenting acquired contours into an appropriate, predefined number of segments, so that an optimal representation vector u^* can be obtained for each one.

3.6 Grammar definition

Contours of bowing parameters (bow velocity, bow force, and β ratio) of the samples belonging to each note class have been carefully observed in order to foresee an optimal representation scheme when using sequences of the constrained Bézier curve segments introduced in previous section. While deciding on the scheme, the aim was to minimize the length of the sequences while preserving representation fidelity.

The grammar defines the expected structural characteristics of each contour, and each grammar entry (expressed by a tuple containing the relevant structure-related constrains) includes information on the number of units and how they are arranged. For a bowing parameter b, the tuple ρ^b contains the number N^b of segments, and a slope sequence constraint vector Δs^{b*}, both used during segmentation and fitting[4]. The vector Δs^{b*} is composed of $N^b - 1$ slope constraints δs_i^{b*}, with $i = 1 \dots N^b$. This is represented as:

$$\rho^b = \{N^b, \Delta s^{b*}\} \tag{3.11}$$

$$\Delta s^{b*} = [\delta s_1^{b*} \cdots \delta s_{N^b-1}^{b*}] \tag{3.12}$$

The slope sequence constraint vector Δs^{b*} defines the expected sequence of slope changes for the bowing parameter contour b. If each i-th segment is approximated linearly, a contour slope sequence $s^b = [s_i^b \cdots s_{N^b}^b]$ is obtained. Each of the $(N^b - 1)$-th pairs of successive slopes entail a slope change that can be either positive ($s_i^b < s_{i+1}^b$) or negative ($s_i^b > s_{i+1}^b$). In order to express an expectancy on the sequence of slope changes, a parameter $\delta s_i^{b*} \in \{-1, +1, 0\}$ is defined for each of the $(N^b - 1)$-th pairs of successive segments. A value of $\delta s_i^{b*} = 0$ denotes no clear expectancy in the relationship between successive slopes s_i^b and s_{i+1}^b. A value of $\delta s_i^{b*} = 1$ denotes expecting an increase in the slope value (i.e., $s_i^b < s_{i+1}^b$), while a value of $\delta s_i^{b*} = -1$ denotes the opposite. This can be summarized as follows:

$$\delta s_i^{b*} = \begin{cases} 0 & \text{if no expectancy on slope change,} \\ 1 & \text{if expected } s_i^b < s_{i+1}^b, \\ -1 & \text{if expected } s_i^b > s_{i+1}^b. \end{cases} \tag{3.13}$$

Exhaustive observation of contours of the different classes led to the definition of a grammar entry for each note class. Analyzing other kinds of articulations or contexts (or even working with control parameter contours of other musical instruments) would of course lead to different grammar entries.

In Figure 3.11 it is sketched an example representation applied to the bowing parameters of an hypothetic note sample. The bow velocity contour is modeled by a sequence of three Bézier segments, having the slopes $s_1^{v_b} \dots s_3^{v_b}$ of their linear approximations to monotonically decrease from segment to segment. For the contour of bow force, a sequence of three Bézier segments is also used, but this time having the slopes $s_1^F \dots s_3^F$ to follow an alternating sequence of changes (first decreasing and then increasing). The β ratio is modeled by two segments for which the slopes of their linear approximations must accomplish $s_1^\beta < s_2^\beta$.

[4] Eventually, the letter b will be used during explanations for denoting any of the three bowing parameters, since the procedures apply identically to bow velocity, bow force, and β ratio. Thus, in the need to refer to them, or to provide formulae supporting the discourse, a letter b must be understood as interchangeable either with v_b (bow velocity), with F (bow force), or with β (β ratio).

The grammar entries needed for defining both the number of segments and their arrangement are expressed by:

$$\begin{aligned} \rho^{v_b} &= \{N^{v_b}, \Delta s^{v_b *}\} && (3.14)\\ & N^{v_b} = 3 && (3.15)\\ & \Delta s^{v_b *} = [-1\ -1] && (3.16) \end{aligned}$$

$$\begin{aligned} \rho^{F} &= \{N^{F}, \Delta s^{F *}\} && (3.17)\\ & N^{F} = 3 && (3.18)\\ & \Delta s^{F *} = [-1\ +1] && (3.19) \end{aligned}$$

$$\begin{aligned} \rho^{\beta} &= \{N^{\beta}, \Delta s^{\beta *}\} && (3.20)\\ & N^{\beta} = 2 && (3.21)\\ & \Delta s^{\beta *} = [+1] && (3.22) \end{aligned}$$

Tables 3.1 and 3.2 provide a detailed list of the grammar entries finally used for carrying out automatic segmentation and fitting of contours of all segmented notes in the database. Segmentation and fitting of bow velocity contours of notes played with *upwards* bow velocity is carried out on the absolute bow velocity. This means that the grammar entries are defined as if the bow velocity contour was always positive-valued, leading to an inversion of the sign of the y-axis values of the obtained Bézier parameters once the fitting process is finished.

Note that only 17 different grammar entries are included, each one corresponding to one of the feasible combinations of articulation type (ART), slur context (SC) and silence context (PC) (see Section 3.4). The reason for this is that the corpus covers all feasible combinations of dynamics (DYN) and bow direction (BD) for each of the 17 feasible combinations of articulation type (ART), slur context (SC) and silence context (PC). Given that observed contours of every combination of articulation type (ART), slur context (SC) and silence context (PC) keep their structural characteristics (i.e., arrangement of Bézier segments) across different combinations (in total 6 combinations) of dynamics (DYN) and bow direction (BD), it is decided to use a unique grammar entry for each of the 17 sets. This leads to a total of 102 classes resulting from 17 principal grammar entries (corresponding to feasible combinations of ART, BC, and PC) for which a different grammar entry is defined, multiplied by 6 final grammar entries (defined by DYN, and BD) that are inherited from each of them. Therefore, tables do not display dynamics (DYN) and bow direction (BD).

3.7 Contour automatic segmentation and fitting

By attending to the previously defined grammar entries, acquired bowing parameter contours of each note sample are automatically segmented and fitted

Grammar entries		
	[*ART SC PC*]	[*ART SC PC*]
	[*détaché iso init*]	[*détaché iso mid*]
N^{v_b}	4	4
$\Delta s^{v_b *}$	[-1 0 -1]	[-1 0 -1]
N^{F}	3	3
$\Delta s^{F *}$	[-1 -1]	[-1 -1]
N^{β}	2	2
$\Delta s^{\beta *}$	[0]	[0]
	[*détaché iso end*]	[*détaché iso iso*]
N^{v_b}	4	4
$\Delta s^{v_b *}$	[-1 0 -1]	[-1 0 -1]
N^{F}	3	3
$\Delta s^{F *}$	[-1 -1]	[-1 -1]
N^{β}	2	2
$\Delta s^{\beta *}$	[0]	[0]
	[*legato init init*]	[*legato init mid*]
N^{v_b}	2	2
$\Delta s^{v_b *}$	[-1]	[-1]
N^{F}	2	2
$\Delta s^{F *}$	[-1]	[-1]
N^{β}	2	2
$\Delta s^{\beta *}$	[0]	[0]
	[*legato mid mid*]	[*legato end mid*]
N^{v_b}	2	2
$\Delta s^{v_b *}$	[-1]	[-1]
N^{F}	2	2
$\Delta s^{F *}$	[0]	[-1]
N^{β}	2	2
$\Delta s^{\beta *}$	[0]	[0]
	[*legato end end*]	[*staccato iso init*]
N^{v_b}	2	3
$\Delta s^{v_b *}$	[-1]	[-1 +1]
N^{F}	2	3
$\Delta s^{F *}$	[-1]	[-1 +1]
N^{β}	2	2
$\Delta s^{\beta *}$	[0]	[0]

Table 3.1: Grammar entries defined for each of the different combinations of articulation type (ART), slur context (SC), and silence context (PC) [Part 1].

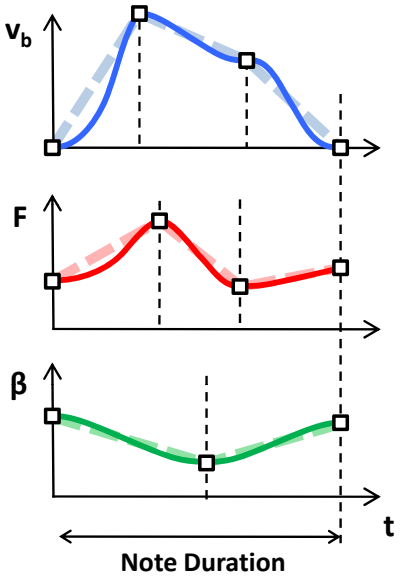


Figure 3.11: Schematic illustration of the bowing parameter contours of an hypothetic note. For each bowing parameter b, thick solid curves represent the Bézier approximation for each one of the N^b segments, while thick light lines laying behind represent the linear approximation of each segment. Squares represent junction points between adjacent Bézier segments.

by appropriate sequences of Bézier cubic curve segments as depicted in Figure 3.11. First, each note in the corpus is classified as one of the 102 note classes into consideration. Then, the corresponding grammar entry is retrieved. Finally, automatic segmentation and fitting of contours of bow velocity, bow force, and β ratio is carried out as detailed next. For clarity, the method is described in general, i.e., not referred to any bowing parameter b, so the letters b are omitted.

Segmentation and fitting of each contour is approached by automatically searching for an optimal duration vector $d^* = [d_1^* \cdots d_N^*]$ (see Section 3.5) such that a total approximation cost C is minimized while satisfying that the sum of all components of the duration vector must be equal to the duration D of the note. This is expressed in equations (3.23) and (3.24), where the approximation error ξ_i for the i-th segment is computed as the mean squared error between the real contour $q_i(t)$ and its optimal Bézier approximation $\sigma_i^*(t)$ (see Section 3.5), and w_i corresponds to a weight applied to each ξ_i except to ξ_N.

$$d^* = [d_1^* \cdots d_N^*] = \underset{d, \sum_{i=1}^{N} d_i = D}{\operatorname{argmin}} \; C(d) \tag{3.23}$$

Grammar entries (continued)		
	[*ART SC PC*]	[*ART SC PC*]
	[*staccato iso mid*]	[*staccato iso end*]
N^{v_b}	3	3
Δs^{v_b*}	[-1 +1]	[-1 +1]
N^{F}	3	3
Δs^{F*}	[-1 +1]	[-1 +1]
N^{β}	2	2
$\Delta s^{\beta*}$	[0]	[0]
	[*staccato iso iso*]	[*saltato iso init*]
N^{v_b}	3	3
Δs^{v_b*}	[-1 +1]	[-1 -1]
N^{F}	3	3
Δs^{F*}	[-1 +1]	[-1 +1]
N^{β}	2	2
$\Delta s^{\beta*}$	[0]	[0]
	[*saltato iso mid*]	[*saltato iso mid*]
N^{v_b}	3	3
Δs^{v_b*}	[-1 -1]	[-1 -1]
N^{F}	3	3
Δs^{F*}	[-1 +1]	[-1 +1]
N^{β}	2	2
$\Delta s^{\beta*}$	[0]	[0]
	[*saltato iso iso*]	
N^{v_b}	3	
Δs^{v_b*}	[-1 -1]	
N^{F}	3	
Δs^{F*}	[-1 +1]	
N^{β}	2	
$\Delta s^{\beta*}$	[0]	

Table 3.2: Grammar entries defined for each of the different combinations of articulation type (ART), slur context (SC), and silence context (PC) [Part 2].

$$C(d) = \Big(\sum_{i=1}^{N-1} w_i \xi_i\Big) + \xi_N \tag{3.24}$$

The weight w_i applied to each of the first $N-1$ computed ξ_i is set as penalty, and depends on the fulfillment of the slope sequence constraints defined by Δs^* (see Section 3.6). For each pair of successive i-th and $(i+1)$-th segments derived from a candidate duration vector d, a parameter δs_i is computed from the slopes s_i and s_{i+1} of their respective linear approximation as:

$$\delta s_i = \text{sign}(s_{i+1} - s_i) \tag{3.25}$$

The weight w_i is set to an arbitrary value $W \gg 1$ in case δs_i does not match its corresponding δs_i^* in the grammar entry (see Section 3.6), only when δs_i^* was

defined as non-zero. This can be expressed as:

$$w_i = \begin{cases} W \gg 1 & \text{if } \frac{\delta s_i}{\delta s_i^*} < 1 \text{ and } \delta s_i^* \neq 0, \\ 1 & \text{otherwise.} \end{cases} \tag{3.26}$$

The solution to this problem, for which one of the search steps is schematically represented in Figure 3.12, is approached by using dynamic programming (Viterbi, 1967). Acquired bowing parameters are sampled at a rate $s_r = 240$Hz (corresponding to the sampling rate of the tracking device, see previous chapter), therefore having an array of M frames $f_1 \dots f_M$ for each contour of a segmented note (details on score-performance alignment are given in Section 2.6.4). The optimal duration vector d^* (see equation (3.23)) can be also seen as the sequence of frames where the junctions of successive Bézier curves fall, i.e., the contour segmentation times. Thus, the starting frame $f_{s,i}$ and the ending frame $f_{e,i}$ of each i-th segment lead to a duration d_i of the form:

$$d_i = \frac{f_{e,i} - f_{s,i}}{s_r} \tag{3.27}$$

This way, instead of searching for the optimal duration vector d^*, the search is performed over a vector f of segmentation frames in which the ending frame $f_{e,i}$ of the i-th segment matches the starting frame $f_{s,i+1}$ of the $(i+1)$-th segment. This can be written as in equation (3.28), where N corresponds to the number of segments dictated by the corresponding grammar entry.

$$f = [\, 1 \;\; f_{e,1} \;\; \dots \;\; f_{e,N-1} \;\; M \,] \tag{3.28}$$

At each step of the algorithm (see Figure 3.12), all possible starting frames f_s are considered as candidates for setting, together with a candidate ending frame f_e, the segmentation limits that will define the duration d_i of the i-th Bézier segment (see equation (3.27)). For each pair of candidates f_s and f_e of the i-th segment, an approximation error ξ_i is computed, and a slope constraint penalty w_i is given (see equations (3.24) and (3.26)).

As a result, it is obtained the set of parameters defining the Bézier curve segments that best model each of the contours of each note in the corpus. Some examples of the results on automatic segmentation and fitting are shown in Figures 3.13 and 3.14, where acquired bowing parameter contours are compared to their corresponding Bézier approximations for *détaché*, *legato*, *staccato*, and *saltato* articulations.

3.8 Contour parameter space

The set of curve parameters of each note is represented as a vector p resulting from the concatenation of three curve parameter vectors, each one respectively holding obtained curve parameters for the contours of bow velocity, bow force, and β ratio. The dimensionality of each vector depends on the number of

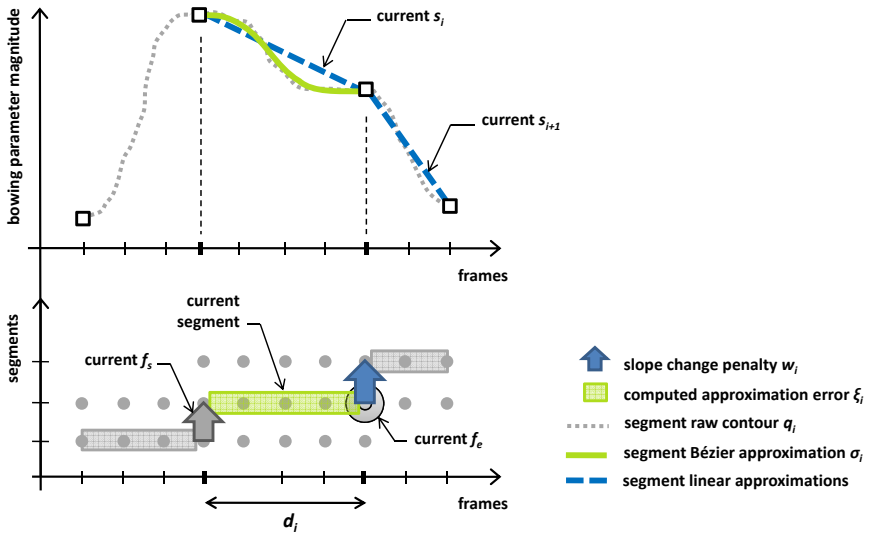


Figure 3.12: Schematic illustration of a search step of the dynamic programming algorithm used for automatically segmenting and fitting contours. The optimal duration vector d^* is found by searching for the sequence of segment starting and ending frames f_s and f_e that lead to a best Bézier approximation of the contour while respecting the slope change constraints vector Δs.

segments used for modeling the corresponding contour, and is pre-defined by the corresponding grammar entry (see Section 3.6). Due to the fact that the same grammar entry is used for carrying out segmentation and fitting of contours of notes belonging to the same note class (see Section 3.6), contour parameters of the same class reside in the same space.

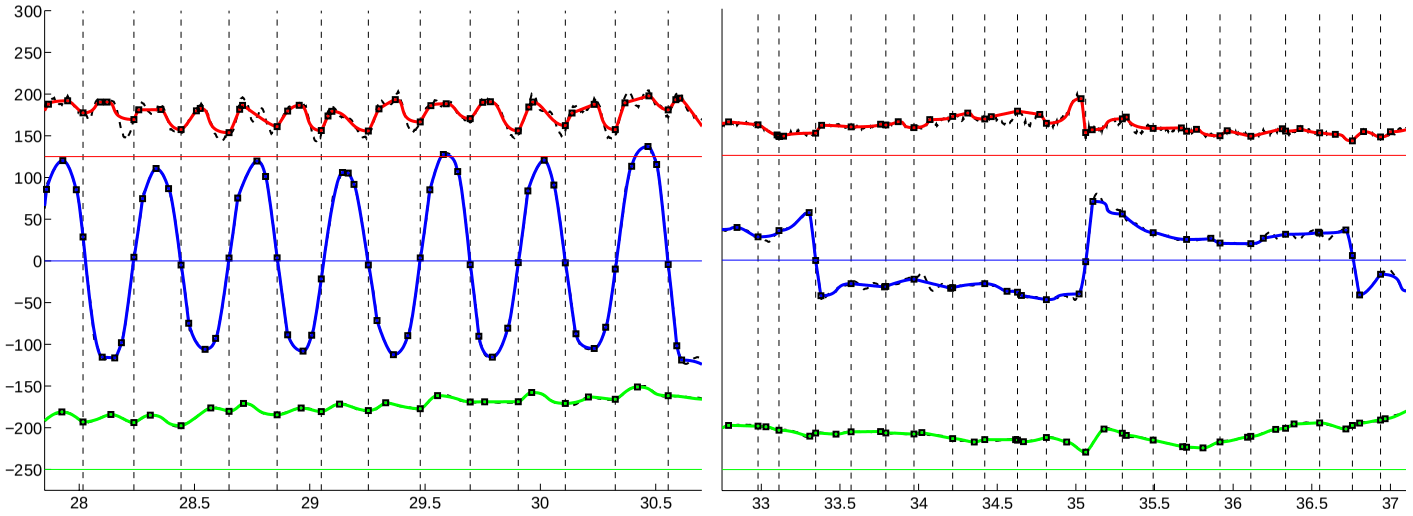
For a bowing parameter b, each of the three parameter vectors contains three different sub-vectors. A first sub-vector $p^{b,d}$ contains the relative durations d_i^b/D of each i-th of the N^b segments. A second sub-vector $p^{b,v}$ contains the the inter-segment y-axis values, i.e., staring values $v_{s,i}^b$ or ending values $v_{e,i}^b$ of each segment, having $v_{e,i}^b = v_{s,i+1}^b$. A third sub-vector $p^{b,r}$ contains the pairs of attractor ratios $r_{1,i}^b$ and $r_{2,i}^b$ (see Figure 3.15). The organization of the parameters of a bowing parameter b is therefore summarized as follows:

$$p^b = \{p^{b,d}, p^{b,v}, p^{b,r}\} \tag{3.29}$$

$$p^{b,d} = [d_1^b/D \cdots d_{N^b}^b/D] \tag{3.30}$$

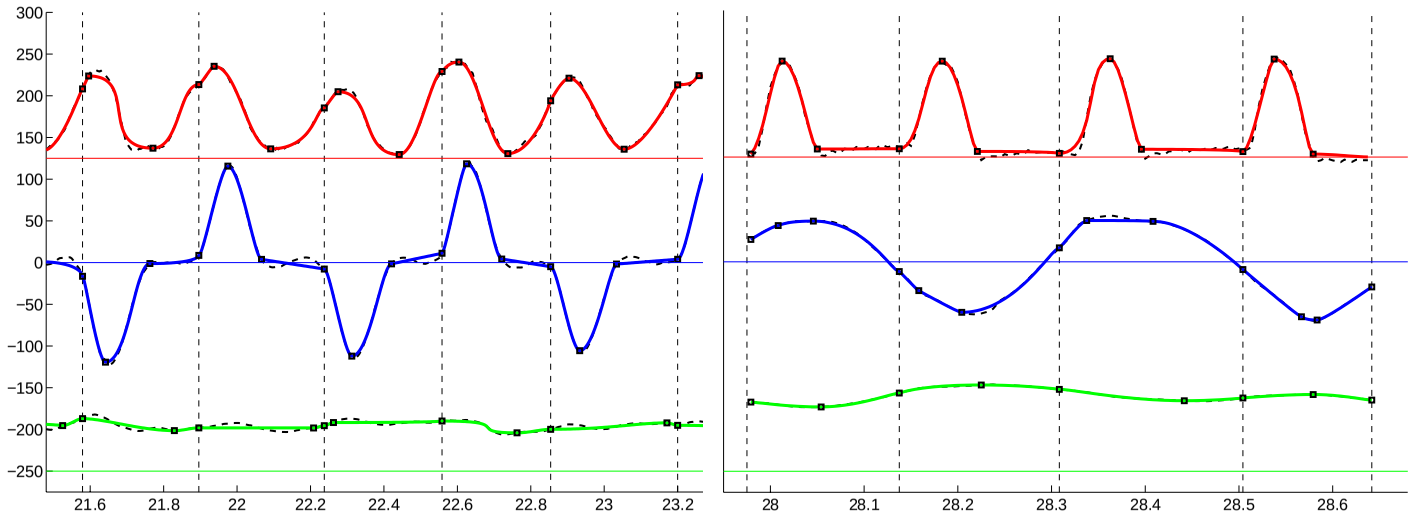
$$p^{b,v} = [v_{s,1}^b \cdots v_{s,N^b}^b \; v_{e,N^b}^b] \tag{3.31}$$

$$p^{b,r} = [r_{1,1}^b \; r_{2,1}^b \cdots r_{1,N^b}^b \; r_{2,N^b}^b] \tag{3.32}$$



(a) *Détaché* articulation (b) *Legato* articulation

Figure 3.13: Results of the automatic segmentation and fitting of bowing parameter contours. In each figure, from top to bottom: acquired bow force ($0.02N/unit$), bow velocity (cm/s), and bow-bridge distance ($0.04cm/unit$) are depicted with thick dashed curves laying behind the modeled contours, represented by solid thick curves. Given that the y-axis is shared among the three magnitudes, solid horizontal lines represent the respective zero levels. Junction points between successive Bézier segments are represented by black squares, while vertical dashed lines represent note onset/offset times (seconds).



(a) *Staccato* articulation

(b) *Saltato* articulation

Figure 3.14: Results of the automatic segmentation and fitting of bowing parameter contours. In each figure, from top to bottom: acquired bow force ($0.02N/unit$), bow velocity (cm/s), and bow-bridge distance ($0.04cm/unit$) are depicted with thick dashed curves laying behind the modeled contours, represented by solid thick curves. Given that the y-axis is shared among the three magnitudes, solid horizontal lines represent the respective zero levels. Junction points between successive Bézier segments are represented by black squares, while vertical dashed lines represent note onset/offset times (seconds).

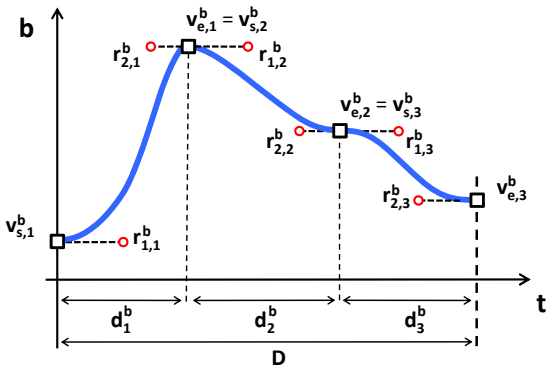


Figure 3.15: Illustrative example of the model parameters contained in each of the curve parameter vectors p^{v_b}, p^F, or p^β (see equations (3.29) through (3.32)).

3.9 Summary

This chapter reported on the contour analysis of acquired bowing parameters, in particular bow velocity, bow force, and bow-bridge distance. The interesting research pursuit of quantitatively representing bowing control parameters from real data had motivated a number of works in the past. Askenfelt (1986, 1989) carried out the first measurements of bowing parameters from real performance, identifying the main characteristics of a number of bowing techniques. His work already envisaged the need of an appropiate representation in order to quantitatively study bowing patterns. Later, Guettler & Askenfelt (1998) also introduced some further analyses of the anatomy of different bowing techniques, although an explicit, quantitative representation suitable for approaching data-driven analysis was not proposed. A more recent approach to modeling the contour of bow velocity and bow force in violin performance was carried out by Demoucron & Caussé (2007); Demoucron (2008). For the first time, a representation scheme was proposed by attending to data observations, and a piecewise sinusoidal model of contours was pursued. Some representation limitations inherent to the chosen parameterization constrained its application to the analysis of *sustained-excitation* bowing patterns. The low flexibility of sinusoidal signals (e.g., shape constrains) makes difficult to build a 'general' representation framework able to account for parameter modulations (happening in different performance contexts) that lead to differences in the shapes of acquired parameters. However, his results already demonstrated that, in the field of instrumental sound synthesis, modeling control parameters is as important as modeling the instrument itself.

Following these lines, the quantitative representation framework introduced in this section establish a further step that goes beyond the aforementioned

limitations. The first main difference is the use of Bézier curves as the representation unit, which represent a more powerful tool given their flexibility, and robustness. Secondly, the schemes that define segment sequences and characterisitics for the different bowing techniques are enclosed within a general and extensible representation framework that is devised after observation of data. Finally, a segmentation and fitting algorithm allows for the automatic characterization and analysis of large amounts of data, enabling further applications like statistical modeling applied to the generation of synthetic bowing parameters from an input score, as it is presented in the next chapter.

Chapter 4

Synthesis of bowing parameter contours

The bowing parameter contour characterization framework presented in previous chapter is successfully applied to construct a statistical model intended for setting the basis for building a synthesis framework able to render bowing controls from an input score. This chapter gives details on the statistical analysis of bowing parameter contours, and the construction of the rendering model (based on Gaussian mixtures). The chapter finalizes by introducing a bow planning algorithm that integrates the rendering model while statistically accounting for the potential constraints imposed by the finite length of the bow.

The contributions presented in this chapter are related to three publications by the author (Maestre, 2009; Maestre et al., 2010; Maestre & Ramírez, 2010). The first one presents the basics of the statistical modeling of bowing parameter contours. The second publication extends the statistical model and introduces the bow planning algorithm. Although not reported in this dissertation, the main contribution of the third publication is, firstly, the application of principal component analysis to the space of curve parameters; and secondly, the rendering of bowing controls by using inductive logic programming techniques.

4.1 Overview

From the notes falling within one of the classes introduced in Section 3.4, one finds notes that were played in different performance contexts (e.g., what was the duration of the note, how close to the bridge was the stroke executed, etc.). The main idea behind the synthesis framework is to statistically model how such performance situations may lead to different shape characteristics of the contour of bowing parameters. For approaching this problem, each note sample within a note class K is represented by two vectors: a vector of *performance context parameters* and a vector of *bowing contour parameters*. While the vector containing the description of the bowing parameter contours (i.e., the vector of

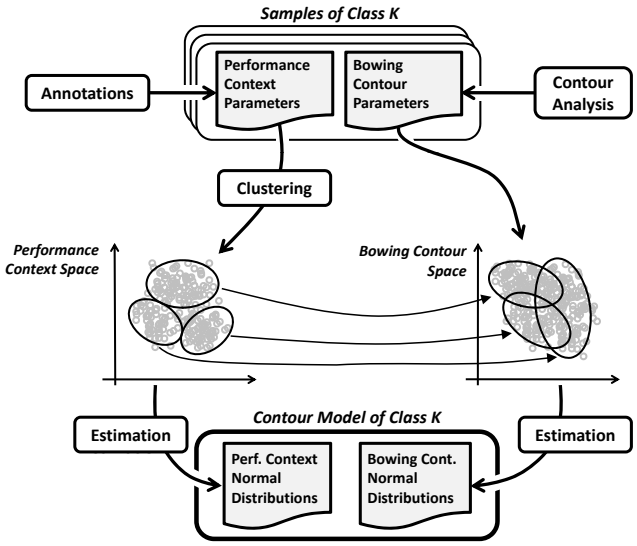


Figure 4.1: Overview of the contour modeling framework.

curve parameters obtained from the automatic analysis introduced in previous chapter) lives in a space whose dimensionality is determined by the amount of curve segments used for modeling each of the three contours (see Section 3.8), the space of performance context parameters is to be built by attending to different annotations coming either from the performed score, or from relevant measurements of the bow transversal position. Four main performance context parameters are considered: the duration of the note, the effective string length (derived from the string played and the pitch of the note, see Section 3.3),the bow starting position (e.g., transversal position of the bow when the note started), and the transversal displacement suffered by the bow during the execution of the note.

In the database, each note sample is represented by a point in the *performance context space* and by a another point in the *bowing contour space* (see Figure 4.1). Clustering of points in the performance context space leads to groups of notes that were executed in similar performance contexts. By correspondence of the note samples, each of the clusters is represented as a cloud of points in the bowing contour space. Then, the parameters of a pair of normal distributions are estimated for each of the original clusters: a normal distribution statistically describing the performance context parameters of the notes in the corresponding cloud (in the performance context space), and a normal distribution of the curve parameters of the bowing contours of the notes in the group (in the bowing contour space).

By means of such statistical modeling, for a new note to be played (rep-

resented by a new performance context vector) it is possible to estimate its different cluster memberships in the performance context space. Then, based on such memberships, a weighted mixture of the corresponding normal distributions in the bowing contour space will statistically describe the bowing contour parameters of the new note. Finally, the new distribution resulting from the mix can be used to obtain synthetic contours, thus leading to a flexible, generative modeling framework.

Starting by a preliminary analysis of the performance context parameters that is followed by some considerations on the clustering procedure, next sections introduce the details of the different steps of the modeling framework.

4.2 Preliminary analysis

It represents a challenging pursuit to quantitatively model possible relationships between the different performance contexts and the way that notes are executed by the performer. With the aim of using the obtained relationships for the generation of synthetic bowing parameter contours from an input score, such input-output (score to bowing contours) problem can be seen as the task of finding a mapping function that goes from a lower-dimensional space (the performance context space) to a high-dimensional space (the bowing contour parameter space). The dimensions of the space of performance context parameters mostly correspond to those of the type that one finds in an annotated score, whereas the space of bowing contour parameters is, in principle, far from being 'naturally' related to a score. This essential difference makes more attractive the approach of dividing note executions by attending to the performance context of the note and not by attending to the actual contours. Indeed, since the input to the system will be based on performance context parameters extracted from a score, a proper discretization of the space where they live is to provide a convenient distance measure.

So far, an approximation to quantitatively representing bowing contours (note executions) has been presented in the previous chapter, having notes of different nature to be classified and represented in different spaces. This first step of grouping notes (e.g., classification) remained easier because of the discrete nature of the characteristics on which the classification was carried out (see Section 3.4). However, within each one of the classes, a continuum of executions is found, raising the need of a different, fuzzier approach to further separating the space into representative areas. Through a second step of separation, a more specialized representation of the bowing contours of notes of each class is to be obtained, leading to a finer-grain statistical description of bowing parameter contours. It is therefore important, also by accounting for the possible space coverage limitations imposed by the database used, (1) to appropriately select the performance context parameters on which such separation is to be based, and (2) to design a procedure that is appropriate for the separation (i.e., clustering or grouping) of the different note samples

while enabling an optimal representation of the relationship between context parameters and contour parameters.

4.2.1 Performance context parameters

From the score-based annotations not used for pursuing note classification (see Section 3.4), two important dimensions are present: note duration and pitch. Given their continuous nature, they form a first set of parameters to become *performance context* -related dimensions of the space where notes of the same class live. It is important to recall here that even though the annotated dynamics can be seen as a continuous-nature dimension, it was considered as discrete because of the difficulties for constructing a database including a continuum of dynamics levels. As it was quantitatively analyzed in Section 3.1, both duration and pitch (the latter expressed as the effective string length) significantly affect the shape of the bowing contours (and so the contour representation parameters).

Two more performance context parameters are considered in this study: the starting bow position and the bow displacement. The bow starting (transversal) position is understood as the distance from the bow frog to the string-hair contact point when the note starts (see Section 2.3.2). The bow (transversal) displacement is computed as the distance (along the bow hair ribbon axis) traveled by the bow during the execution of the note, thus measured as the absolute value of the difference between the bow starting position and the bow ending position. Although these two dimensions are not explicitly annotated in the score, they can be clearly seen as performance context parameters since, for a given note sequence and the constrains imposed by both the bow direction changes and the finite length of the bow, they are chosen by the performer during the performance.

The set of performance context parameters considered in this work can be summarized as:

- Derived from score annotations:
  - Note duration D
  - Effective string length l_e
- Derived from the finite length of the bow:
  - Bow starting position BP_{ON}
  - Bow displacement ΔBP

Two important factors are going to condition the decisions taken when building the modeling framework. On one hand, the performance context space coverage limitations imposed by the database may lead to a particular clustering strategy. On the other, correlations found between some performance context

parameters (for example, the natural relationship between the duration of a note and the bow displacement occurring during its execution) may imply dropping some of the performance context parameters, having its relationship with the other(s) to be statistically modeled in a latter stage of the modeling process.

Further parameters could enlarge the dimensionality of the performance context space: duration of surrounding notes, metrical strength, or higher-level continuous-nature descriptors obtained from a musicological analysis of the score.

For some representative note classes, a preliminary qualitative analysis of observed correlations between different performance context parameters and some relevant contour parameters is outlined next. It remains difficult to pursue a more comprehensive qualitative analysis that can be visually represented, given the high dimensionality of the contour parameter space (ranging from 25-dimensional to 50-dimensional, depending on the note class), and the large number of note classes. Instead, aiming at being able to come out with clarifying arguments that support the selected approach, the focus is put into some particular contour parameters that appear to be relevant, leaving out cross-correlation information (to be latter represented by the covariance of the normal distributions), and keying on representative note classes, in particular corresponding to sustained bowing techniques (see Section 3.1.1).

In the following sections, the different correlations are displayed in a configuration of graphs like the one shown in Figure 4.2. Although the relative lengths of the attractors r_1 and r_2 are included in the analysis, no significant conclusions can be drawn from observing the results of the correlation analysis. A possible reason for this is that they play a more important role in the 'signature' of the shape than in explaining shape variations due to differences in the performance context of notes.

Note duration

The first and most important of the performance context parameters considered in this work is the duration of the note. In the qualitative analysis of bowing contours that was carried out while devising the contour representation scheme (outlined in Section 3.2.4), the role of the note duration was highlighted as an important aspect to be taken into account. Although not carrying relevant structural information about the shape of the contours, the note duration determines the relative durations of the segments used for modeling each of the contours. In order to carry out the analysis of the effects of note duration in the curve parameters, the focus is put into a representative note class of each of two *sustained-excitation* bowing techniques that are considered here: *détaché* and *legato*.

Figure 4.3 depicts, for the note class defined by the tuple [*détaché ff downwards iso mid*][1], the individual correlation coefficient of the note duration and

[1] See Section 3.4 for a description of the meaning of each of the variables of the tuple.

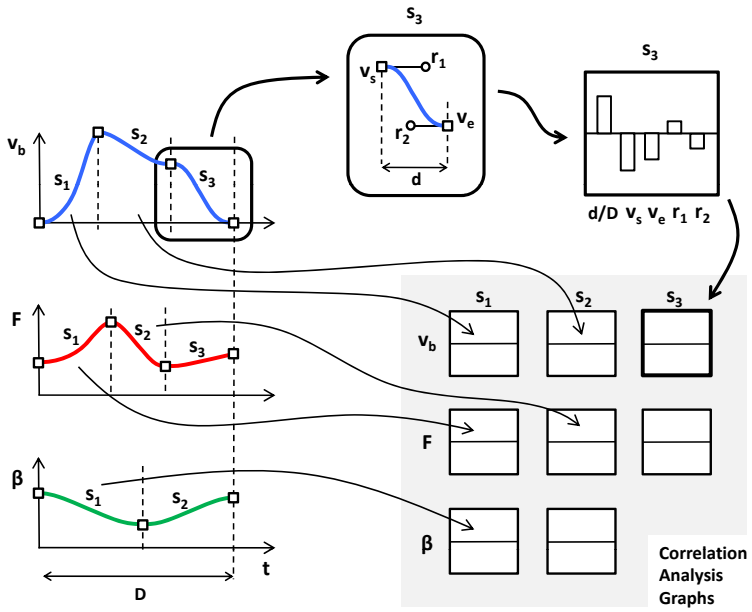


Figure 4.2: Schematic explanation of the meaning of the graphs used in the preliminary correlation analysis, for an hypothetic. Rows correspond to bowing parameters, while columns correspond to contour segments. In each graph are displayed the correlations of all five curve parameters and the performance context parameter under analysis.

each Bézier curve parameter used for representing the contours. The curve parameters d/D, v_s, v_e, r_1 and r_2 used for representing each segment have been respectively collected for the $N^{v_b} = 4$ segments of the bow velocity contour, the $N^F = 3$ segments of the bow force contour, and the $N^\beta = 2$ segments used for representing the bow-bridge distance contour (see Sections 3.5 and 3.8). Then, an array has been constructed from the collected values of each of the curve parameters, having a duration correlation coefficient computed by using this array and the array of note durations of the samples falling into such class.

By paying attention to the correlation coefficients obtained for the bow velocity contour (appearing in the top row), the strongest correlation is observed between the note duration D and the relative durations d/D of the Bézier segments (appearing the first column of each subplot). Let's recall how the bow velocity contour is modeled. Four segments are used (see the example contours in Figure 3.4 and the grammar entry in Table 3.1). The first and the last segments (s_1 and s_4 in the upper row of plots of the figure) correspond to the portions where the bow velocity is, respectively, quickly increasing and decreasing (a bow direction change is preceding and following the note). The two segments

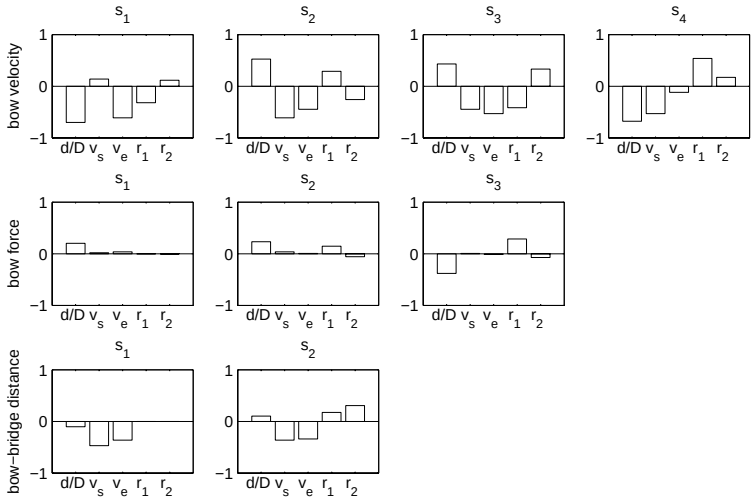


Figure 4.3: Correlation coefficient between the different curve parameters and note duration, obtained for the note class [*détaché ff downwards iso mid*]. Each subplot shows the correlation of the five parameters defining a Bézier segment (d/D, v_s, v_e, r_1 and r_2). Subplots in each row correspond to the segments used for modeling the contour of each bowing parameter, while the columns are arranged so that the segments appear in their temporal order. For this class, 4 Bézier segments were used for modeling the bow velocity contour, 3 for the bow force, and 2 for the bow-bridge distance.

segments appearing in the middle (s_2 and s_3 in the upper row of plots) correspond to the portion where the bow velocity follows a more sustained behavior, a priori carrying most of the duration information. The modulation that the relative duration d/D of the different segments is clearly perceived from the correlation depicted in the figure. While the first and the last segment present a strong negative correlation coefficient (their relative duration gets shorter when the duration of the note increases), the two middle segments show a positive correlation, thus proving to fetch the temporal expansion of the contour. Also by looking to the subplots corresponding to the bow velocity contour, it is possible to observe an evident negative correlation of the starting and ending values v_s and v_e of the middle segments, thus indicating that the bow velocity reaches higher values values for shorter notes (this can be visually confirmed by the contours plotted in Figure 3.4).

Less clear, though still existing is the analogous phenomenon that can be observed in the bow force contour duration parameters (segment relative durations d/D). While the relative duration of the first two segments appear as positively correlated to note duration, the relative duration of the third segment

(representing the decrease of bow force at the final portion of the note, see Table 3.1) shows the opposite behavior. The reader is referred to Figure 3.4 for qualitative visual inspection.

The starting and ending values v_s and v_e of the segments used for modeling the contour of bow-bridge distance also show here a negative correlation to note duration, thus indicating that the overall level of bow-bridge distance decreases with note duration (also perceived from observing Figure 3.4).

In order to check whether these behaviors are consistent across different sustained bowing techniques, a similar analysis is carried out now for the case of *legato*-articulated notes. The note class chosen is the one represented by the tuple [*legato ff downwards end mid*]. The duration correlation coefficients for this note class are depicted in Figure 4.4. In this case, the contour of bow velocity is represented by two segments, the first one corresponding to the more sustained portion of the envelope, and the second one corresponding to the portion of the envelope where the bow velocity quickly decreases towards zero (it is preceding a bow direction change, see its grammar entry in Table 3.1, and the example contours plotted in Figure 3.5[2]). Like it happens with the previous articulation type, the relative duration of the more sustained segment (s_1 in the upper row of plots in the figure) presents a positive correlation, hence confirming that it suffers the most important temporal stretch (carries most of the duration transformation). In contrast, the second segment (s_2 in the upper row of plots) gets a shorter relative duration for longer notes.

Again, the last segment of the bow force contour is negatively correlated to note duration (contrary to the behavior of the fist segment), and a similar explanation is found. Parallel to what is observed for the *détaché* note class presented before, the overall level bow-bridge distance decreases with note duration (negative correlations of v_s and v_e for both segments in the lower row of plots), as it happens with the bow velocity (v_s and v_e in the upper row of plots).

From the above analysis, it remains clear the necessity of carrying out a separation of notes based on their duration (possibly as part of a clustering procedure). Otherwise, a statistical description of the curve parameters of a note class will be less informative if note samples of very different durations are used for estimating means and variances, thus leading to a less realistic representation that will provide worse results during the contour synthesis stage.

Effective string length

The effective length of the string is obtained from the score by attending to the annotated string and the pitch of the note (e.g., finger position). It is well known (Schelleng, 1973; Cremer, 1984; Guettler et al., 2003) that a change in the

[2]The example contours shown in Figure 3.5 correspond to the opposite bow direction, thus the negative values of bow velocity. However, the main temporal characteristics perceived from each of the two segments help to visually understand the observed difference in correlation to note duration.

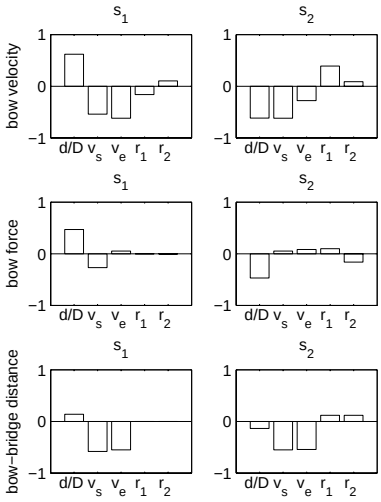


Figure 4.4: Correlation coefficient between the different curve parameters and note duration, obtained for the note class [*legato ff downwards end mid*]. Each subplot shows the correlation of the five parameters defining a Bézier segment (d/D, v_s, v_e, r_1 and r_2). Subplots in each row correspond to the segments used for modeling the contour of each bowing parameter, while the columns are arranged so that the segments appear in their temporal order. For this class, 2 Bézier segments were used for modeling the bow velocity contour, 2 for the bow force, and 2 for the bow-bridge distance.

length of the string strongly affects the manner in which the performer combines the three bowing parameters to achieve the desired timbre characteristics of the produced sound. Although the details of this effect are not obvious, it remains particularly clearer for the case of the bow-bridge distance. Variations of the string length change the relative durations of the stick-slip phases (directly linked to the relative lengths of the *bow-bridge* and *bow-finger* sections of the string) that describe the Helmholtz regime Helmholtz (1862). In general, this causes the performer to adjust the distance to the bridge, so that such changes in length get compensated. As the bow-bridge distance is not the only parameter that drives timbre characteristics, further, less clearly explained modulations may happen with regard to the bow velocity and bow force contours.

In order to account for this type of effect in the modeling framework, the effective length of the string is included as a performance context parameter. In fact, it remains very clear to observe the phenomenon by performing a qualitative analysis of correlation as introduced in previous section, but computing the correlation coefficients by using the effective string length of each note. The same two note classes have been chosen, having the results depicted in

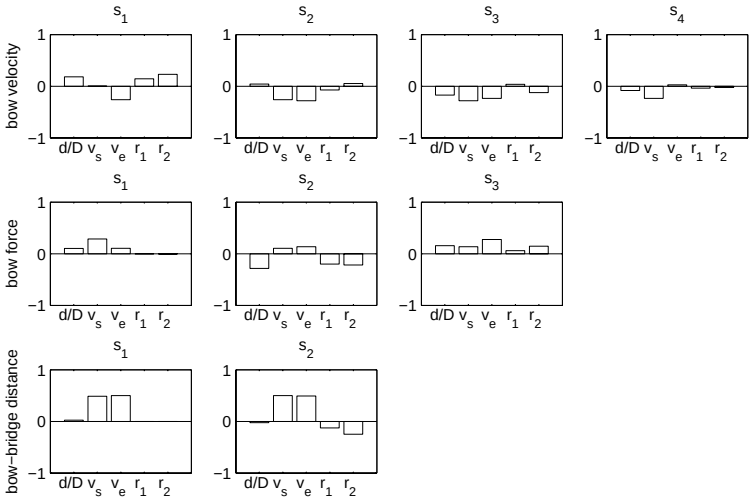


Figure 4.5: Correlation of bowing contour parameters to effective string length, obtained for the note class [*détaché ff downwards iso mid*]. Each subplot shows the correlation of the five parameters defining a Bézier segment (d/D, v_s, v_e, r_1 and r_2). Subplots in each row correspond to the segments used for modeling the contour of each bowing parameter, while the columns are arranged so that the segments appear in their temporal order. For this class, 4 Bézier segments were used for modeling the bow velocity contour, 3 for the bow force, and 2 for the bow-bridge distance.

Figures 4.5 and 4.6, each one corresponding to the note classes respectively represented by the tuplas [*détaché ff downwards iso mid*] and [*legato ff downwards end mid*].

While most of the curve parameters show a lower correlation to the effective string length, the starting and ending values v_s and v_e of the Bézier segments used for modeling the contour of the bow-bridge distance present a strong, positive correlation that confirms the fact that the performer is increasing the distance to the bridge as the string gets longer (lower pitch), as it has been explained above. It is therefore important to use the effective length of the string as a separation parameter, leading to different normal distributions in which the bow-bridge distance variance is clearly delimited.

Bow starting position and bow displacement

The bow section that is used for executing a note does not explicitly appear annotated in the score, as it is strongly linked to a bow planning anticipatory process that the musician follows as a consequence of the constraints imposed

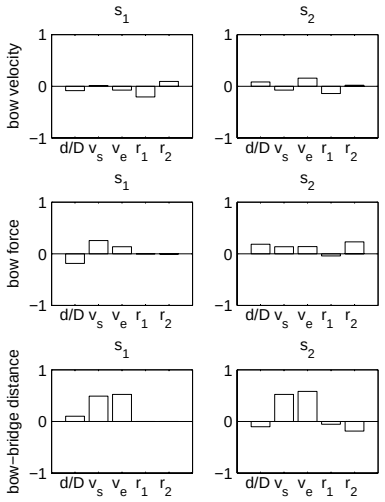


Figure 4.6: Correlation of bowing contour parameters to effective string length, obtained for the note class [*legato ff downwards end mid*]. Each subplot shows the correlation of the five parameters defining a Bézier segment (d/D, v_s, v_e, r_1 and r_2). Subplots in each row correspond to the segments used for modeling the contour of each bowing parameter, while the columns are arranged so that the segments appear in their temporal order. For this class, 2 Bézier segments were used for modeling the bow velocity contour, 2 for the bow force, and 2 for the bow-bridge distance.

by the finite length of the bow. For a sequence of notes appearing in the score to be played, the performer chooses a corresponding sequence of bow sectors (each one expressed as a bow starting position and a bow displacement) by attending to certain preferences or habits related to the characteristics of the note (e.g., bowing technique or articulation, duration, dynamics), to some contextual aspects (e.g., position in a slur, bow direction changes), and to the limitations derived from the length of the bow. In general, a skilled violinist should be able to adapt her bowing so that the sound of the notes remains independent on the bow sector being used. While being a obvious habit that musicians have incorporated by training, building an appropriate bowing control model that accounts for this phenomenon represents a challenging pursuit.

While the bow starting position is not linked to the contour characteristics of bowing parameters in an apparent way, the relation of the bow displacement with the bowing parameters is rather more direct: the bow displacement can be seen as the integration of the bow velocity contour over time. This distinction could lead to follow a different approach for including each of them in the modeling framework.

For the case of bow starting position, its non-obvious influence on the shape of the contours, an statistical approach based on previous sample separation (by performing note clustering also based on the bow starting position) like the one introduced in previous subsections could provide a representation of any existing relevant behavior.

In terms of bow displacement, it would be straightforward to derive an analytical expression in which it appears as a function of the curve parameters used for modeling the bow velocity contour. Such function, however, would not represent a bijection[3], thus not being possible to deterministically obtain the set of curve parameters for a desired bow displacement. Even in the hypothetical case that an approximation to the equivalent inverse of such function (from bow displacement to curve parameters of bow velocity) could be estimated, the non-obvious relationship between the bow velocity contour and the contours of bow force and bow-bridge distance should not be eluded, hence ending up in a complex model that would be more easily constructed by statistical analysis of collected contours than by deriving a set of analytical formulae.

In this work, the inclusion of these two performance context parameters into the modeling framework is approached from an statistical perspective. While the bow starting position is used for clustering note samples in a preprocessing step during the construction of the model (as it happens with the note duration and the effective string length), the bow displacement is used in a posterior stage, when the synthetic contours of bowing parameters are rendered (see Figure 4.7 and Figure 4.13). During that stage, any statistical dependence that exists between the bow displacement and the contour shape of the bowing parameters (not only bow velocity, but also bow force and bow-bridge distance) is accounted for meaningfully modifying the curve parameters used for synthesizing contours (see Section 4.4.2 and 4.5).

Two main reasons motivated this decision. The first one resides both on the coverage limitations and on the amount of samples of the database used for constructing the model. In the case that an optimal coverage had been achieved, the four performance context parameters could have been used for clustering, but when having a look at the distribution of the different performance context parameters, it was observed that the bow displacement was more correlated to note duration than the other parameters. This fact, obvious from the perspective of the performer (in general, the longer the note, the longer the distance that the bow travels during its execution), indicated that including bow displacement was not bringing relevant information at that stage of the modeling process, and would possibly cause a worset space separation leading to an uneven distribution of samples into clusters[4].

The second one inherently derives from the proposed framework architecture, which is devised with the aim of representing the bow planning process

[3] A known of set of curve parameters of the bowing contour would lead to a known bow displacement, but not the other way around.

[4] Due to the limited number of samples, some of the clusters would be containing too few notes, thus leading to less reliable statistical descriptions.

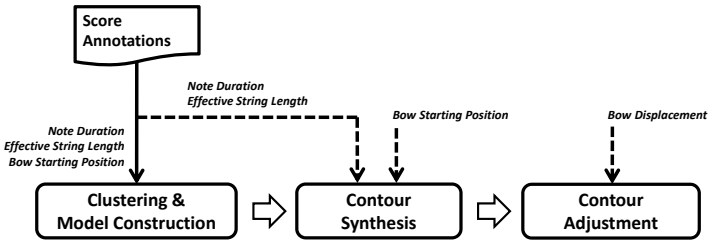


Figure 4.7: Preview of the contour synthesis framework. Solid arrows corespond to performance context parameters used during model construction, while dashed arrows indicate parameters used during contour synthesis.

that is carried out by the performer. The sequence of note durations and the effective string lengths are obtained from the score, while the *optimal* sequence[5] of bow starting positions and bow displacements is obtained by means of a bow planning algorithm conceived for such purpose. For each note in the input score, the desired bow starting position, in conjunction with the scripted note duration and the effective string length, is used for obtaining the curve parameters of the three bowing contours. Then, these curve parameters are adjusted so that a desired bow displacement is matched (see Section 4.4.2). As it has been discussed above, the relationship between the bow displacement and the bowing parameter contours appears more direct (especially for the case of bow velocity), so it is decided that a more effective adjustment of obtained curve parameters can be applied in the last stages of the contour synthesis process. Therefore, the bow displacement is not used for driving the clustering process, but as an extra dimension of the contour parameter vectors that are used for obtaining the normal distributions in the contour parameter space (see Figure 4.1). This way, any modification that needs to be applied to the obtained curve parameters in order to force a desired bow displacement can be performed by following the statistical dependence between them and the bow displacement itself (see Section 4.4.2).

In Figures 4.8 and 4.9, the correlation coefficients between the different curve parameters and the bow starting position appear displayed for notes of the classes respectively defined by the tuples [*détaché ff downwards iso mid*] and [*legato ff downwards end mid*], for durations between 0.8 and 1.0 seconds. A reduced duration range has been selected for visually inspecting correlations. This decision is based on the fact that in general, the duration of the database notes is correlated to the bow displacement annotations acquired from measurements (as it has been pointed out earlier in the discussion). Since the bow displacement is indeed constraining the bow starting position (especially for

[5] In this case, the *optimal* sequence is understood as the *most likely* sequence to happen, see Section 4.5.

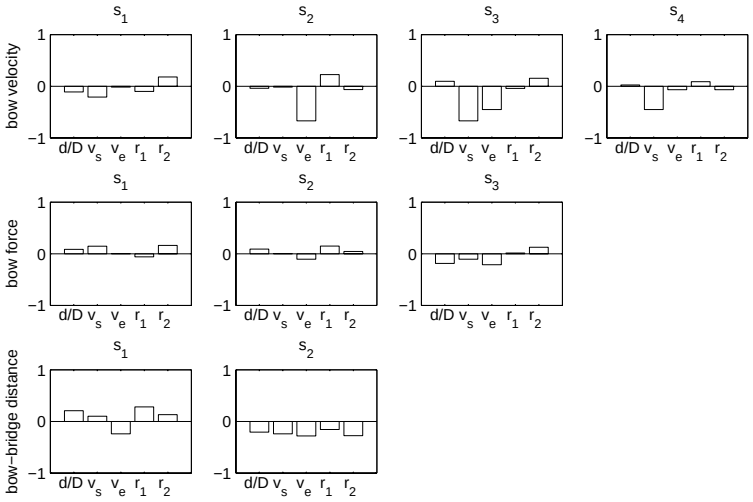


Figure 4.8: Correlation of bowing contour parameters to starting bow position, obtained for the note class [*détaché ff downwards iso mid*] by attending to note durations ranging from 0.8 to 1.0 seconds. Each subplot shows the correlation of the five parameters defining a Bézier segment (d/D, v_s, v_e, r_1 and r_2). Subplots in each row correspond to the segments used for modeling the contour of each bowing parameter, while the columns are arranged so that the segments appear in their temporal order. For this class, 2 Bézier segments were used for modeling the bow velocity contour, 2 for the bow force, and 2 for the bow-bridge distance.

longer notes, due to the finite length of the bow), including a broad range of note durations would lead this visual analysis to become parallel to that devoted to note duration.

Let's recall again how the bow velocity contours are modeled for these two classes, since the correlation coefficients standing out are, for both note classes, those obtained from the starting and ending values v_s and v_e of some particular segments of the bow velocity contour. For the case of the *détaché* note class (Figure 4.8), four segments are used (see the example contours in Figure 3.4 and the grammar entry in Table 3.1). The first and the last segments (s_1 and s_4 in the upper row of plots of the figure) correspond to the portions where the bow velocity is, respectively, quickly increasing and decreasing (a bow direction change is preceding and following the note). The two segments appearing in the middle (s_2 and s_3 in the upper row of plots) correspond to the portion where the bow velocity follows a more sustained behavior. It is precisely happening in these two segments that their starting and ending values show a significant (negative) correlation to the starting bow position, roughly indicating that a

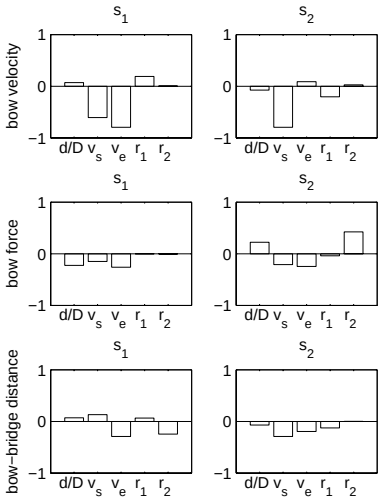


Figure 4.9: Correlation of bowing contour parameters to starting bow position, obtained for the note class [*legato ff downwards end mid*] by attending to note durations ranging from 0.8 to 1.0 seconds. Each subplot shows the correlation of the five parameters defining a Bézier segment (d/D, v_s, v_e, r_1 and r_2). Subplots in each row correspond to the segments used for modeling the contour of each bowing parameter, while the columns are arranged so that the segments appear in their temporal order. For this class, 2 Bézier segments were used for modeling the bow velocity contour, 2 for the bow force, and 2 for the bow-bridge distance.

higher velocity is reached when the note starts by bowing from a position closer to the frog, i.e., closer to zero. Since the note class under analysis presents *downwards* bow direction (the bow position increases along the note), this explains that the bow moves faster if a considerable portion of bow is still remaining.

A similar behavior is observed for the *legato* note class in Figure 4.9, thus serving as a proof of consistency. The starting and ending values v_s and v_e of the first segment (s_1 in the lower row of plots) also show a considerable (negative) correlation. Since this segment represents the more stable part of the envelope (see the corresponding grammar entry in Table 3.1, and the example contours plotted in Figure 3.5[6]), a higher bow velocity is reached when the note is starting at a bow position for which a portion of bow is still left, i.e., closer to the frog for *downwards* bow direction.

[6]The example contours shown in Figure 3.5 correspond to the opposite bow direction, thus the negative values of bow velocity. However, displayed help in a visual, qualitative segmentation of the contour.

4.2.2 Considerations on clustering

As already introduced in previous sections, note samples must be clustered by attending to their representation vectors in the space of performance context parameters (see Figure 4.1), so that it is possible to obtain a more specialized statistical description of the curve parameters used for modeling the bowing contours. This will enable a flexible modeling framework in which different distributions of curve parameters can be mixed for obtaining synthetic contours by attending to a desired set of performance context parameters (note duration D, effective string length l_e, and bow starting position BP_{ON}). Given the importance of note sample clustering in the process of building the model, some considerations must be taken into account regarding how the obtained clusters represent the data.

A first concern resides on the clustering algorithm to be used. One can find in the literature a vast number of clustering techniques and variants, from which two algorithms could be considered among the most popular: the *k-means* algorithm (MacQueen, 1967) and the Expectation-Maximization (EM) algorithm (Dempster et al., 1977). A main difference between them resides on the type of assignment made to data samples. While the former comes out with hard assignments (a sample belongs just to one cluster), the latter provides fuzzy memberships to samples (it is based on Maximum Likelihood estimation). For the framework proposed in this dissertation, in which a 3-dimensional normal distribution is estimated a posteriori from each of the resulting clusters, note samples are not used any longer through the process (it is not a classification pursuit), so partial memberships of a sample to the different clusters do not provide relevant information. A second difference deals with convergence. None of the two algorithms guarantee to converge at an absolute minimum, although the EM algorithm shows in general better results. Possibly given by the simplicity of the clustering problem that represents separating note samples by attending to *well known* performance context parameters, the results obtained using both algorithms were, notwithstanding, almost identical in a number of tests carried out with several note classes. Based on the above considerations, it is decided to use the simpler *k-means* algorithm.

The second consideration deals with database nature. Given the limited size and coverage of the database, not all the areas of the performance context space are densely populated for every note class, especially in terms of note duration[7]. In fact, from the analysis of correlation that has been outlined in the previous section, the note duration could be unveiled as the most important of the performance context parameters (curve parameters showed in general a greater correlation to note duration that to the other performance context parameters). Because of the uneven distribution of note samples in the space, it often happens to obtain clusters for which the statistical properties of the

[7] For the construction of the database, the variety in note durations was achieved by setting a fixed tempo and changing note duration values, hence the sparseness in the coverage of note durations.

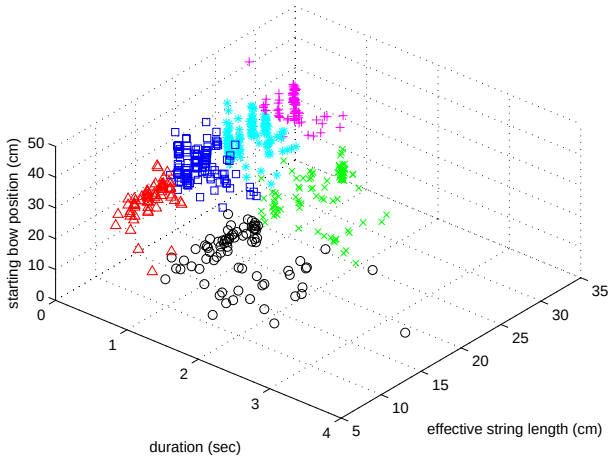


Figure 4.10: Performance context -based clustering of note samples of the note class defined by the tuple [*détaché mf downwards iso init*]. Each of the 6 clusters presents its samples represented by a a different symbol.

duration (e.g., variance) result deficient (i.e., not representative) as compared to what one could expect given an a priori *known* distribution of note durations. This is particularly true for note classes that include a relatively small number of samples, like the example in Figure 4.10, where the clustering results of the samples of the class [*détaché mf downwards iso init*] have been illustrated for the case of 6 clusters[8].

In the initial results displayed in Figure 4.10, two clusters are obtained in the range of longer note durations (roughly between one and two seconds). In each of these two clusters, note samples of very different durations are included, implying a large variance of curve parameters (see the discussion on note duration in previous section) in the normal distributions to be estimated from the corresponding groups of vectors in the bowing contour space (see Figure 4.1). By attending to the distribution of note duration of samples in this note class, it is clear from the plot that three main groups are present. If one aims at giving preference to the quality of the statistical representation of note duration, it would be preferred not to obtain any cluster to enclose notes from such significantly different durations.

A simple but effective solution to this problem can be achieved by devising a two-step, hierarchical clustering process. In a first step, note samples can be clustered by only attending to note duration (first dimension of the performance context parameter vectors), thus performing an initial separation into a lower number of duration-based groups. Then, within each duration group, an arbi-

[8] A minimum amount of samples is required to fall into each cluster with the aim of securing a reliable statistical description of each group, hence a sufficiently low number of clusters must be used.

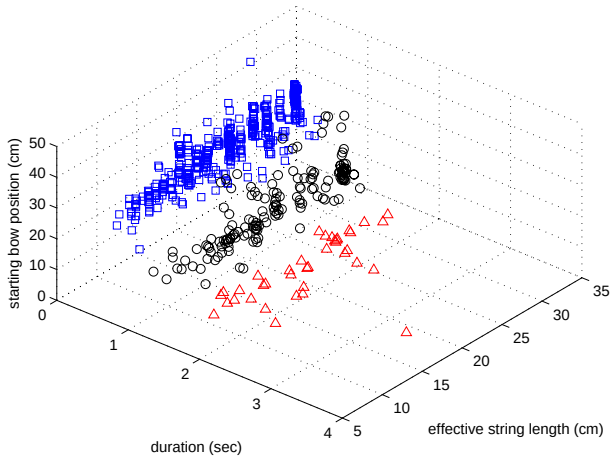


Figure 4.11: First step of a two-step hierarchical clustering of note samples, applied to samples of the note class defined by the tuple [*détaché mf downwards iso init*]. Three different clusters appear, obtained by attending only to note duration. Samples corresponding to each clusters are represented by different symbols.

trary number of sub-clusters can be obtained by attending to the distribution of the performance context parameters (the three dimensions of the performance context parameter vectors) of the separated samples. Figures 4.11 and 4.12 display how the duration variance in each of the final clusters is improved by applying the proposed method to the samples of the same class as in Figure 4.10 , i.e., defined by [*détaché mf downwards iso init*]. Figure 4.11 illustrates the duration clusters obtained at the first step, while Figure 4.12 displays the final results, having two clusters obtained from each of the previously obtained duration clusters. This time, while still obtaining 6 clusters (Figure 4.12), the duration distribution of the final clusters significantly improved the original duration distributions (Figure 4.10), leading to a better separation in terms of note duration.

Data clustering represents a difficult issue for which an *optimal* solution is in general very dependant on the nature and meaning of each of dimensions of the data used, the size and sparseness of the space, and the final application of the obtained results. In the ideal case of being in possession of a densely covered space of performance context parameters, ad-hoc fixings like the duration-based 2-step clustering process proposed here would probably be unnecessary. Yet, different and more complex clustering strategies could be further studied, but such pursuit falls out of the scope of this work.

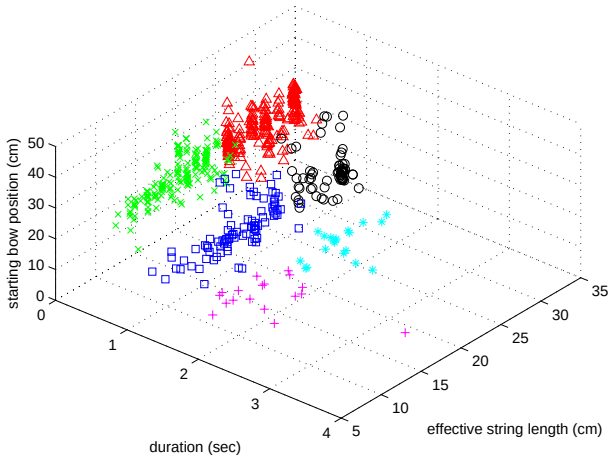


Figure 4.12: Second step of a two-step hierarchical clustering of note samples, applied to samples of the note class defined by the tuple [*détaché mf downwards iso init*]. A total of 6 clusters appear, resulting from separating note samples of each of the previously obtained duration clusters (see Figure 4.11) into 2 sub-clusters.

4.2.3 Selected approach

An introductory description of the proposed approach to the synthesis of bowing parameter contours is given next. The generative modeling framework presented here is based on statistical modeling of bowing parameter contours, specifically using a methodology of the kind *finite mixtures of normal distributions*, by which different distributions of curve parameter vectors are combined for rendering synthetic bowing controls from an annotated input score.

Based on the initial representation depicted in Figure 4.1, a contour rendering -oriented illustration of the proposed method is sketched in Figure 4.13. The first step is to obtain a contour model for each of the note classes into which note samples have been previously classified (see Section 3.4). As already outlined, the contour model of each class consists on (1) clustering note samples by attending to their performance context parameters (note duration D, effective string length l_e, and bow starting position BP_{ON}), and (2) estimating the parameters of a pair of normal distributions for each of the obtained clusters, the first from the performance context vectors (in the performance context space), and the second from the curve parameters vectors corresponding to the samples of the cluster (in the bowing contour space). Clustering of note samples is set as a two-step process, by which first is obtained an arbitrary number of duration-based clusters, each one containing an (also) arbitrary number of performance context -based clusters.

Given the annotations of a note in the input sequence, the note is classi-

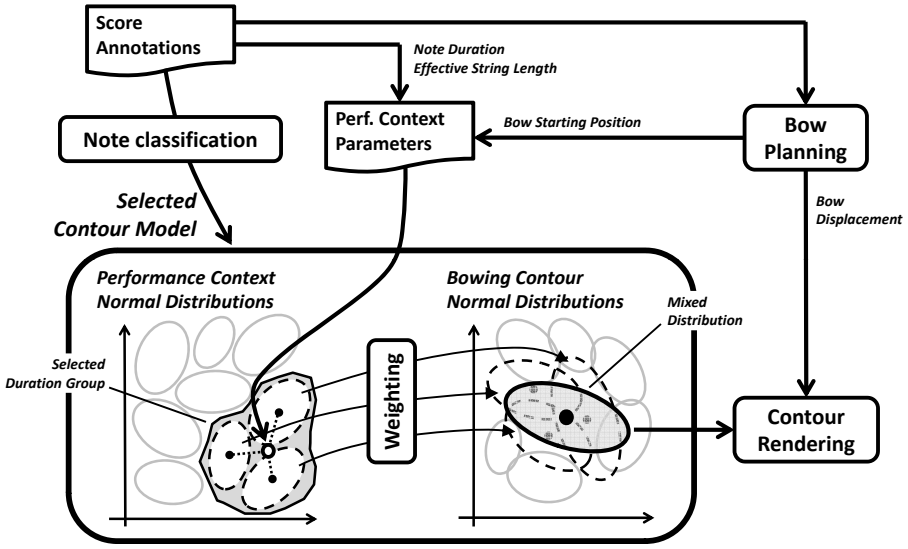


Figure 4.13: Overview of approach selected for rendering bowing contours.

fied into one of the note classes considered in the representation scheme (see Section 3.4), and the corresponding contour model is selected. The note duration D and the effective string length l_e are extracted from the input score, while the bow starting position BP_{ON} is provided by a bow planning emulation algorithm, also fed by the input score annotations. From the resulting three-dimensional vector, the memberships of the note's context to the performance context normal distributions (see Figure 4.13) are computed my means of a normalized distance measure related to conditional probability estimation (or likelihood estimation). First, a duration group is selected by attending to the duration of the note. Each duration group is characterized by a duration normal distribution, so that a duration likelihood estimation is used for the selection. Then, a normalized distance of the performance context is used to compute the memberships corresponding to the normal distributions inside the selected duration cluster.

Estimated cluster memberships are used for obtaining a weighted mixture of the bowing contour normal distributions to which each of the selected clusters correspond in the curve parameter space (see Figure 4.13). A synthetic curve parameter vector can be obtained by generating a random vector with a probability density defined by the mean and covariance of the curve parameter distribution resulting from the mix.

Before using obtained curve parameters for rendering the actual bowing

contours, they need to be tuned in order to match certain constraints. A first constraint comes from a target bow displacement imposed by the bow planning emulation. As pointed out during the discussion on how to include the bow starting position BP_{ON} and bow displacement ΔBP in the modeling framework (see previous sections), the statistical dependence between the contour parameters and the bow displacement is used for fulfilling this first constrain. By including the bow displacement as an additional dimension in the curve parameter space (see Section 3.8 and equation (4.1), where p^{v_b}, p^F, and p^β correspond to the curve vectors of curve parameter vectors or the bow velocity, the bow force, and β ratio), each of the bowing contour estimated distributions (so does the mixed distribution) carries such dependence in its covariance. Based on such covariance, the values of a subset of curve parameters (all except the bow displacement) of the randomly generated vector can be tuned for matching the target value of a desired, complementary subset (the bow displacement) of the same vector by using least-squares, i.e., by minimizing the likelihood difference between new vector and the originally generated vector. The fulfillment of a number of other constraints also implies certain adjustments of synthetic curve parameters. For further details, see Section 4.4.2.

$$p = \{p^{v_b}, p^F, p^\beta, \Delta BP\} \tag{4.1}$$

The framework proposed here represents one of many possible approaches that could be devised for generating synthetic bowing controls by parameterizing acquired contours following a structured scheme. It is not the intention to claim these ideas to be founding the optimal approach to modeling violin bowing control, but to transmit a methodology and a research experience, and to inspire the application of similar approaches to a number of related problems.

A number of the decisions taken along the way are significantly based on the nature of the problem, and also on the nature of the available data. In particular, the reduced number of samples falling into certain classes led to drop the possibility of constructing different models for different strings. This is an important drawback derived from the size and coverage of the database. However, the flexibility of the framework easily allows further, meaningful extensions that can be easily integrated into the existing architecture.

Modeling the stochasticity of a real performance

For a note in the input score, curve parameters of bowing contours are obtained by generating random vectors with a certain probability density. Such probability density is defined by the mean and covariance of an appropriate mixture of curve parameter normal distributions obtained in the bowing contour space (see Figure 4.13). Hence, notes defined by identical performance context parameters will yield to obtaining different curve parameter vectors even though mixture weights will be identical, leading to a certain stochastic behavior that indeed responds to the original stochasticity appearing in the data used for constructing the model.

While one could think of such *noise* as mainly derived from data acquisition or segmentation errors, or from contour representation errors, it remains clear that a significant stochastic component appearing in human performance is indeed providing a degree of naturalness to perceived sound (the musician plays identical notes by means of slightly different bowing control parameter profiles). Assuming that the probability density of such *human* noise is indeed shaping the covariance of the mixed normal distribution from which random vectors are generated, it represents an good opportunity for meaningfully parameterizing the stochasticity of bowing control. Thus, by scaling the variance of the mixed distribution, the stochasticity of the randomly generated vectors gets modulated while maintaining the cross-correlation between components (i.e., curve parameters). In practice, such scaling can be performed in three steps:

1. Decompose the covariance matrix into eigenvalues and eigenvectors.
2. Scale all the eigenvalues by a desired factor.
3. Recompose the covariance matrix.

Synthetic bowing contours obtained for two different variance scaling factors are compared for different note classes in Appendix B, in particular in Figures B.19, B.20, B.21 and B.22. The columns on the right display contours obtained by scaling the variances by a factor of 2.0 with respect to that on the left.

Overcoming problems derived from non-normality of variables

In this statistical framework, the components of the curve parameter vectors are assumed to follow a normal behavior, modeled by a Gaussian distributions. Whereas the actual probability density of most of the components responds in general to a Gaussian shape, the nature of the magnitude measured by some components make their distribution to appear as significantly *non-normal*. As an example, this is particularly true for the relative lengths r_1 and r_2 of the curve attractors of each segment (see Section 3.5), which in a number of cases respond to an asymmetric probability density: the values of r_1 and r_2 live either close to zero or close to one, mostly because of the upper and lower bounds imposed by the nature of the constrained Bézier segments used for modeling contours.

When obtaining a synthetic curve parameter vector from a mixture of normal distributions, it is rather probable that the generated value of some of the *non-normal* components (e.g., r_1 and r_2) falls out of its valid range (e.g., r_1 or r_2 being out of $[0,1]$). In that case, a correction of such component is needed before using the curve parameters for rendering the corresponding contours. By using the covariance of the normal distribution from which the random vector was generated, such correction can be performed while preserving the likelihood of the originally generated vector. Therefore, required adjustments to the other (not constrained) components of p will represent how variables are statistically

correlated to each other. The reader is referred to the end of Section 4.4.2 for further details on this issue.

4.3 Statistical modeling of bowing parameter contours

This section gives details on the statistical description of bowing parameter contours. To recall, a note sample (from now on, a *sample*) is represented by two vectors, each one in a different space. The first vector is denoted as s, and lives in the space of performance context parameters, of lower dimensionality. The second vector is denoted as p, and lives in the space of Bézier curve parameters, of higher dimensionality[9]. As already introduced, samples are clustered by attending to their performance context parameters (in vector s. Statistical descriptors (e.g., probability distributions) are estimated (1) in the performance context space, from vectors s of each cluster; and (2) in the bowing contour space (i.e., curve parameter space) from the curve parameter vectors p of each cluster. The procedure applies to any of the note classes (see Section 3.4).

4.3.1 Sample clustering

Samples are clustered into different groups based on a set of performance context parameters, consisting in note duration D, bow starting position BP_{ON} (distance from the frog to the string contact point), and effective string length l_e (distance from the bridge to the finger). A performance context vector s is constructed from these three parameters (see equation (4.2)) and attached to each sample. From the set of vectors s, a two-step, hierarchical clustering process is carried out. Due to database limitations (see Section 4.2.2), samples are first grouped into N different duration groups by only attending to their note duration. Then, M performance context sub-clusters are obtained within each of the N duration clusters, this time by attending to the vectors s.

$$s = [\, D \; BP_{ON} \; l_e \,] \tag{4.2}$$

Duration-based clustering

In a first step, N duration groups (clusters) of samples are obtained by applying the *k-means* clustering algorithm to the samples of a note class, based on the first component of the performance context vector s, i.e., the note duration D.

Performance context -based clustering

In a second step, M performance context sub-clusters of samples are obtained by applying again the *k-means* clustering algorithm to the notes within each of the previously obtained N duration clusters, but this time attending to the

[9] The number of dimensions is determined by the number of curve segments used for modeling each contour.

3-dimensional context vector s. Ideally, this leads to $N \times M$ performance context sub-clusters $c^{n,m}$ per note class. Each of these sub-clusters contains a number of samples, each one represented by a performance context vector s, and a curve parameter vector p. Thus, each cluster is represented in the performance context space by a set $s^{n,m}$ of performance context vectors, and in the bowing contour space by a set $p^{n,m}$ of curve parameter vectors.

4.3.2 Statistical description

The parameters included in the statistical model of bowing contours of a note class consist on (a) statistical descriptors of duration, bow starting position, and bow displacement attributes for each of the N duration clusters, and (b) statistical descriptors of curve parameter vectors and of performance context attributes for each of the M performance context sub-clusters within each of the duration clusters.

Duration cluster description

In a first step, a note duration normal distribution d^n, defined by a mean duration τ^n and a duration variance Φ^n (see equation 4.3), is estimated from the duration D of the notes contained in the n-th duration cluster.

$$d^n = \{\tau^n, \Phi^n\} \tag{4.3}$$

Secondly, for each duration cluster and each bowing parameter b (i.e., either bow velocity, bow force, or bow-bridge distance), it is performed an analysis of the correlation between the absolute durations $[d_1^b \dots d_{N_b}^b]$ of the N^b Bézier segments, and the note duration D. This information is included in the model in order to be able to adequately adjust segment relative durations when reconstructing contours (see Section 4.4). The task of this analysis is to find which of the N^b segments presents the highest correlation with the note duration D[10]. For doing so, all note samples belonging to the duration cluster under analysis are first collected. Then, it is computed the Pearson correlation coefficient $r(d_i^b, D)$ between the absolute duration d_i^b of each i-th segment of the total N_b segments, and the note duration D. Finally, it is selected the segment number $q^{n,b}$, corresponding to the segment presenting the highest correlation (see equation (4.4)). As a result, a duration correlation vector q^n containing obtained segment numbers $q^{n,V}$, $q^{n,F}$, $q^{n,\beta}$ for each of the three contours (see equation (4.5)) is attached to each n-th duration cluster:

$$q^{n,b} = \underset{i, i=1\cdots N^b}{\operatorname{argmax}}\ r(d_i^b, D) \tag{4.4}$$

$$q^n = \{q^{n,V}, q^{n,F}, q^{n,\beta}\} \tag{4.5}$$

[10] The segment presenting the highest correlation will be chosen for applying any time transformation that is needed during synthesis.

Next, a bow starting position distribution a^n, defined by a mean bow starting position α^n and a bow starting position variance A^n, is estimated from the bow starting position BP_{ON} annotated for each sample in the n-th duration cluster, as expressed in equation 4.7. This information will be used in the emulation of bow planning, as described in Section 4.5.

$$a^n = \{\alpha^n, A^n\} \tag{4.6}$$

Finally, a bow displacement distribution b^n, defined by a mean bow displacement π^n and a bow displacement variance Π^n, is estimated from the bow displacement ΔBP annotated for each sample in the n-th duration cluster, as expressed in equation 4.7. This information will also be used in the emulation of bow planning, as described in Section 4.5.

$$b^n = \{\pi^n, \Pi^n\} \tag{4.7}$$

Performance context sub-cluster description

Assuming that both the curve parameter vectors $p^{n,m}$ and the context vectors $s^{n,m}$ to follow a normal distribution, $N \times M$ pairs of normal distributions $g^{n,m}$ and $v^{n,m}$ are estimated, each pair corresponding to one sub-cluster. The distribution $g^{n,m}$ is estimated from the set $p^{n,m}$ of curve parameter vectors belonging to the sub-cluster $c^{n,m}$, and is defined by a mean vector $\mu^{n,m}$ and covariance matrix $\Sigma^{n,m}$. Analogously, the distribution $v^{n,m}$ is estimated from the set $s^{n,m}$ of performance context vectors belonging to the sub-cluster $c^{n,m}$, and is defined by a mean vector $\gamma^{n,m}$ and covariance matrix $\Omega^{n,m}$. The two normal distributions describing each sub-cluster are expressed as

$$g^{n,m} = \{\mu^{n,m}, \Sigma^{n,m}\} \tag{4.8}$$

$$v^{n,m} = \{\gamma^{n,m}, \Omega^{n,m}\} \tag{4.9}$$

Model parameters overview

The set of parameters describing the bowing contour model for each note class contain:

- N duration clusters, each one defined by:
  - A duration normal distribution d^n
  - A segment duration correlation vector q^n
  - A bow starting position normal distribution a^n
  - A bow displacement normal distribution b^n
- M performance context sub-clusters within each of the N duration clusters. Each of the $N \times M$ performance context sub-clusters is defined by:
  - A performance context normal distribution $v^{n,m}$
  - A curve parameter normal distribution $g^{n,m}$

4.3.3 Discussion

Given the database size and coverage limitations, the clustering procedure is set up as an iterative process, having the initial parameters to be $N = 3$ (three duration clusters) and $M = 9$ (nine performance context sub-clusters). Because of design principles followed when creating the set of exercises used as recording scripts, having enough notes of at least three different durations was assured for almost every combination of articulation type, bow direction, and dynamics. However, some classes derived from such combinations (e.g., different silence contexts) contained only a few samples. Thus, whenever too few samples are found in any of the M performance clusters, the parameter M is automatically reduced and the second-step clustering (performance context -based clustering within each duration cluster) is repeated, yielding to values of $2 \leq M \leq 9$. For the specific case of *staccato* and *saltato* articulations, some of the note classes are better represented by just two duration clusters ($N = 2$). Clustering parameters (and hence the modeling capability of the clusters) is of course dependent on the size and nature of the database.

Indeed, in the ideal situation of being in possession of an optimally sized database providing a satisfactory coverage of the performance context space, the first step of the clustering process (duration -based clustering) would not be needed. In such case, the segment duration correlation vectors and the bow displacement normal distributions (see above) could be computed from each sub-cluster, thus having more specific statistical descriptions that could be combined, as it is carried out with the curve parameter distributions attached to each performance context sub-cluster (see next Section).

4.4 Obtaining synthetic contours

By using the model parameters, curve parameter values corresponding to each bowing contour are obtained for a given note in a score context. First, the note class to which each note belongs is determined by following the principles outlined in Section 3.4. Then, a target performance context vector s^t of performance context parameters (expressed as in equation (4.10), see Section 4.3.1) is determined. Based on s^t, a mix g^* of curve parameter normal distributions is obtained from the distributions g corresponding to the model of the class into consideration. From g^*, a curve parameter vector p (see equation (3.32)) is randomly generated. Finally, some components of generated p are tuned before p can be used for rendering the bowing parameters contours of the given note.

$$s^t = [\, D^t \; BP^t_{ON} \; l^t_e \,] \tag{4.10}$$

4.4.1 Combining curve parameter distributions

This section details the steps followed to obtain the mix g^* of curve parameter distributions attending to a target performance context vector s^t. The vector s^t

is determined by the target duration D^t, the effective string length l_e^t (obtained from the pitch of the note), and a desired bow starting position BP_{ON}^t. In case that the process of synthesizing bowing parameter contours is used within the bow planning emulation algorithm described in Section 4.5.1, BP_{ON}^t is provided by the planning emulation. Conversely, if contours are to be synthesized without using the bow planning algorithm, BP_{ON}^t is set to the mean bow starting position α^{n^*} (see equation (4.6)) of the samples contained the selected duration cluster n^*.

Duration cluster selection

First, the appropriate duration cluster n^* (see Section 4.3.1) is selected. For doing so, a normalized Euclidean distance between the target duration D^t and each of the N cluster duration distributions d^n centroid is computed as

$$n^* = \underset{n}{\operatorname{argmin}} \sqrt{\frac{(D^t - \tau^n)^2}{(\Phi^n)^2}} \tag{4.11}$$

Performance context sub-cluster selection

Within the selected duration cluster n^*, the closest K performance context clusters (see Section 4.3.1) to the target context vector s^t are selected from the M sub-clusters in cluster n^*. For doing so, it is computed the Mahalanobis distance D_M between s^t and each m-th context vector distribution $v^{n^*,m}$ in n^* (see equation (4.12)), and a vector $h = [h_1 \dots h_K]$ is filled with indexes h_i of the performance context sub-clusters in decreasing order by attending to the computed distances.

$$D_M(s^t, v^{n^*,m}) = \sqrt{(s^t - \gamma^{n^*,m})^T (\Omega^{n^*,m})^{-1} (s^t - \gamma^{n^*,m})} \tag{4.12}$$

Obtaining the mixture

The mix g^* of curve parameter distributions is obtained as a weighted average of K *source* curve parameter distributions in the bowing contour space, each one corresponding to one to the closest K performance context distributions to the performance target s^t in the performance context space. The parameters μ^* and Σ^* of the mixed curve parameter distribution g^* respectively correspond to the weighted average of the means and covariance matrices of the K *source* curve parameter distributions. This is expressed by:

$$g^* = \{\mu^*, \Sigma^*\} \tag{4.13}$$

$$\mu^* = \sum_{i=1}^{K} w_i \, \mu^{n^*,h_i} \tag{4.14}$$

$$\Sigma^* = \sum_{i=1}^{K} w_i \, \Sigma^{n^*,h_i} \tag{4.15}$$

For establishing the weights w_i corresponding to each distribution in the mix, the previously computed Mahalanobis distances are used. The weight for the i-th Gaussian component is written as

$$w_i = \frac{D_M^{-1}(s^t, v^{n^*,h_i})}{\sum_{j=1}^{K} D_M^{-1}(s^t, v^{n^*,h_j})} \tag{4.16}$$

Note that choosing a good value for K is again very dependent on how data is distributed. It is more convenient to keep a low value for K, so that averaging of source distributions g does not imply a possible loss of variance information. In general, a choice of $3 \leq K \leq 4$ is preferred if enough performance context sub-clusters are present.

4.4.2 Contour rendering

The final step is to randomly generate a vector p that follows the probability distribution determined by μ^* and Σ^*. Ideally, the components of such vector (curve parameters) could be directly used for rendering the contours by rendering the corresponding Bézier curve segments (see Section 3.5 and Appendix A). However, a number of components of p must be tuned in order to fulfill certain constraints. Some of such constraints are derived from the nature of the model (e.g., duration of segments is coded as relative, so the sum must equal one); others come from modeling as *normal* some variables that are *non-normally* distributed (the relative lengths r_1 and r_2 of the Bézier attractors must be valued between zero and one, see Section 4.2.3); others come from a number of aspects particular to intrinsic or contextual characteristics of the notes (e.g., the starting value of a note's bow velocity must be zero if the note is *détcahé*-articulated and follows a bow direction change); other come from inherent errors in the data acquisition process (e.g., in off-string bowing conditions the acquired force is, in general, slightly over zero); and a number of them are derived from the application use of the synthetic contours (e.g., note concatenation). In the next subsections, it is introduced a formalization of the the different adjustments to be performed for fulfilling such constraints, followed by a methodology for preserving the original likelihood of generated p while satisfying them.

Segment relative durations

The relative segment durations d_i must sum to the unity for each of the three contours. In order to perform the adjustments for a bowing contour b (either bow velocity, bow force, or bow-bridge distance), the duration of the segment given by $q^{n^*,b}$ (which corresponds to the one found presenting the highest correlation with the note duration, see Section 4.3.2) is modified. This applies to any of the three bowing parameters. In the contour parameter vector p, the value of the relative duration $d^b_{q^{n^*,b}}$ (indeed corresponding to the segment $q^{n^*,b}$ of the

bowing parameter b) is set to a value that, given the other relative durations, makes relative durations to sum to the unity. This is expressed in equation (4.17), where D corresponds to the note target duration, and N^b corresponds to the number of segments used for modeling the contour of the bowing parameter b.

$$d^b_{q^{n^*,b}}/D = 1 - \sum_{\substack{i=1 \\ i \neq q^{n^*,b}}}^{N^b} d^b_i/D \tag{4.17}$$

In case that either some of the segment durations is out of the range $[0,1]$, its value is set to zero or to one respectively. In case the sum of the durations of the segments distinct of $q^{n^*,b}$ is already bigger than one, a new curve parameter vector p is generated.

Segment attractor x-value ratios

Due to the nature of the Bézier cubic curve segments used for modeling any bowing contour b, the attractor x-value ratios r_1 and r_2 must always be in the range $[0,1]$ (see equations (4.18) and (4.19)). In case that any of them is out of the range, its value is set to zero or to one respectively.

$$0 \leq r^b_{1,i} \leq 1 \quad \forall i = 1,\ldots,N^b \tag{4.18}$$

$$0 \leq r^b_{2,i} \leq 1 \quad \forall i = 1,\ldots,N^b \tag{4.19}$$

$$\tag{4.20}$$

Negative values of bow force and bow-bridge distance

Obviously, only non-negative values of bow force F and bow-bridge distance d_{BB} (the latter represented as the β ratio) are allowed. If any of the inter-segment y-axis values (corresponding to starting or ending segment y-axis values v_s or v_e, see equations (4.21) through (4.24)) appears as negative in the generated curve parameter vector p, its value is set to zero.

$$v^F_{s,i} \geq 0 \quad \forall i = 1,\ldots,N^F \tag{4.21}$$

$$v^F_{e,i} \geq 0 \quad \forall i = 1,\ldots,N^F \tag{4.22}$$

$$v^\beta_{s,i} \geq 0 \quad \forall i = 1,\ldots,N^\beta \tag{4.23}$$

$$v^\beta_{e,i} \geq 0 \quad \forall i = 1,\ldots,N^\beta \tag{4.24}$$

Note class -specific constrains

The following set of constraints are derived from intrinsic characteristics of the notes, or from bowing data acquisition imperfections. They are organized into bowing technique -related, and slur context -related.

- **Bowing technique**

  Bow velocity and bow force contours present some particularities specific to both *staccato* and *saltato* bowing techniques:

  - *Staccato* bowing technique

    The nature of *staccato* (as it is agreed in this work, see Section 3.1.1) implies that the bow stops for certain time at the end of the note (so that notes are *separated*). Therefore, given that the bow velocity contour is modeled by using three segments, and the third segment represents such state (see Table 3.2), the starting and ending y-axis values v_s and v_e of the segment number 3 must be set to zero. This is expressed as:

    $$v_{s,3}^{v_b} = 0 \tag{4.25}$$
    $$v_{e,2}^{v_b} = 0 \tag{4.26}$$
    $$v_{e,3}^{v_b} = 0 \tag{4.27}$$

  - *saltato* articulation

    In *saltato* (as the literal translation of its name -*jumped*- suggests), the bow releases the string for certain time at the end of the note, so the bow force stays at its (theoretical) minimum value of zero Newtons. Based on the fact that the bow force contour is modeled using three segments, and the third segment represents such state (see Table 3.2), the starting and ending y-axis values v_s and v_e of the segment number 3 must be set to zero. This is expressed as:

    $$v_{s,3}^{F} = 0 \tag{4.28}$$
    $$v_{e,2}^{F} = 0 \tag{4.29}$$
    $$v_{e,3}^{F} = 0 \tag{4.30}$$

- **Slur context**

  Starting and ending bow velocity values of a note must respect a number of constrains depending on the slur context (see Section 3.4) of the note to be rendered:

  - *init* slur context

    * *downwards* bow direction:

      Any note starting a slur (e.g., first in a sequence of *legato* notes) played with *downwards* bow direction must never present a negative bow velocity starting value. Thus the starting value v_s of the first segment of the bow velocity contour will have to be greater than or equal to zero (see equation (4.31)), otherwise it will be set to zero.

      $$v_{s,1}^{v_b} \geq 0 \tag{4.31}$$

* *upwards* bow direction:
  Any note starting a slur (e.g., first in a sequence of *legato* notes) played with *upwards* bow direction must never present a positive bow velocity starting value. Therefore, the starting value v_s of the first segment of the bow velocity contour will have to be less than or equal to zero (see equation (4.32)), otherwise it will be set to zero.

$$v_{s,1}^{v_b} \leq 0 \tag{4.32}$$

– *end* slur context

* *downwards* bow direction:
  Any note ending a slur (e.g., last in a sequence of *legato* notes) played with *downwards* bow direction must never present a negative bow velocity ending value. Hence, the ending value v_e of the last segment of the bow velocity contour will have to be greater than or equal to zero (see equation (4.33), where N^{v_b} corresponds to the number of segments used), otherwise it will be set to zero.

$$v_{e,N^{v_b}}^{v_b} \geq 0 \tag{4.33}$$

* *upwards* bow direction:
  Any note ending a slur (e.g., last in a sequence of *legato* notes) played with *upwards* bow direction must never present a positive bow velocity ending value. Thus, the starting value v_s of the last segment of the bow velocity contour will have to be less than or equal to zero (see equation (4.33), where N^{v_b} corresponds to the number of segments used), otherwise it will be set to zero.

$$v_{e,N^{v_b}}^{v_b} \leq 0 \tag{4.34}$$

– *iso* slur context

* *downwards* bow direction:
  Any note performed as the only note of a slur (i.e., not a slur, but a single note in the stroke) played with *downwards* bow direction must never present a negative bow velocity starting value. Therefore, the starting value v_s of the first segment must be greater than or equal to zero (see equation (4.35)), otherwise it must be set to zero.

$$v_{s,1}^{v_b} \geq 0 \tag{4.35}$$

$$\tag{4.36}$$

* *upwards* bow direction:
  Any note performed as the only note of a slur (i.e., not a slur, but a single note in the stroke) played with *upwards* bow direction

must never present a positive bow velocity starting value. Therefore, the starting value v_s of the first segment of the contour must be less than or equal to zero (see equation (4.37)), otherwise it must be set to zero.

$$v_{s,1}^{v_b} \leq 0 \tag{4.37}$$

$$\tag{4.38}$$

Application-specific constraints

When rendering bowing contours in the context of automatic performance (e.g., used for violin sound synthesis), two main constraints appear. The first one is derived from solving discontinuities between synthetic contours of successive notes. The second one is a special constraint that deals with the adjustment of obtained curve parameters for matching a certain bow displacement (already introduced in Section 4.2.1, and performed within the algorithm for bow planning emulation presented in next section).

- **Note concatenation**

  Possible bowing contour discontinuities between successive notes are solved by setting the starting value v_s of the first segment of each of the three segment sequences (bow velocity, bow force, and bow-bridge distance) to the ending value v_s of the last segment of their corresponding sequence in the preceding note (already rendered). This is expressed in equations (4.39) through (4.41), where n represents the sequence order of the current note in the score, and $N^{v_b}(n-1)$, $N^{F}(n-1)$, and $N^{\beta}(n-1)$ represent the total number of segments used in the $(n-1)$-th note for respectively modeling the contours of bow velocity, bow force and bow-bridge distance (the latter represented as the β ratio).

$$v_{s,1}^{v_b}(n) = v_{\{e,N^{v_b}(n-1)\}}^{v_b}(n-1) \tag{4.39}$$

$$v_{s,1}^{F}(n) = v_{\{e,N^{F}(n-1)\}}^{F}(n-1) \tag{4.40}$$

$$v_{s,1}^{\beta}(n) = v_{\{e,N^{\beta}(n-1)\}}^{\beta}(n-1) \tag{4.41}$$

- **Bow transversal displacement**

  The desired bow parameter displacement ΔBP^t of the note to be rendered is to substitute the generated value in the sample curve parameter vector p (recall that the bow displacement is added as an extra contour parameter, see Section 4.2.1 and equation (4.1)). Just substituting (adjusting) such value does not affect the curve parameters (the bow displacement is indeed determined by the contour of the bow velocity). In fact, if a particular bow displacement adjustment is desired, curve parameters corresponding to the bow velocity contour (and possibly, by inherent cross-correlation,

the curve parameters corresponding to the other contours) need to be adjusted. By using the covariance of the mixed curve parameter distribution g^* from which the initial curve parameter vector p was generated (i.e., by using Σ^*), such required changes in components of the curve parameter vector other than ΔBP can be estimated by minimizing the likelihood difference between the new curve parameter vector p_f and that of the initially generated p. This method, that can be expressed roughly by equation (4.42), is also applicable to any other constraint, and will be detailed in next subsection.

$$p_f = f(p, \Delta BP^t, \Sigma^*) \tag{4.42}$$

An illustrative example of the adjustment of bow displacement is shown in Figure 4.14. An initial curve parameter vector p is generated for a *détaché* note of duration $D^t = 0.9sec$, obtaining a bow displacement $\Delta BP = 49.91cm$. Four different bow displacements are set as target, two above and two below the originally generated value. Such target values are $\Delta BP = \{40cm, 45cm, 55cm, 60cm\}$, each one yielding a different adjustment of the curve parameters of original p. The contours rendered from original p are depicted in the figure by thick lines, while the contours rendered from the adjustments of p are depicted by thin lines. For longer bow displacements (darker colors), it is possible to observe overall increases in bow velocity and bow bridge distance, and an overall decrease in bow force.

In order to check the validity of the adjustments, the actual values of bow displacements are computed by integrating each rendered bow velocity signal over time. Obtained values are shown in Table 4.1, where the originally generated bow displacement is displayed in bold characters.

ΔBP (cm)	
target	actual
40.00	41.81
45.00	46.13
49.91	50.47
55.00	54.96
60.00	59.49

Table 4.1: Adjustment of bow displacement for a *détaché* note of duration $D^t = 0.9sec$. Different target values are compared to actual (computed by integration of bow velocity over time) values.

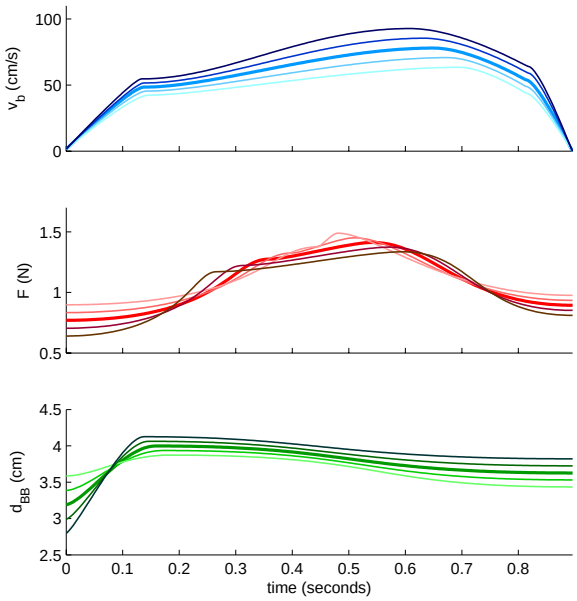


Figure 4.14: Rendered bowing contours for five different bow displacements ΔBP. Synthesized note is [*détaché ff downwards iso mid*] with a target duration $D^t = 0.9sec$. Darker colors represent longer bow displacements. Originally generated contours are displayed by thick lines, while thin lines display contours obtained from tuning curve parameters for matching a target bow displacement. The original bow displacement was $\Delta BP = 49.91cm$, while the target bow displacements are $\Delta BP = \{40cm, 45cm, 55cm, 60cm\}$. In each subplot, lighter colors correspond to shorter bow displacements (i.e., 40cm and 45cm), and darker colors are used to depict contours adjusted to match shorter bow displacemnts(i.e., 55cm and 60cm).

Tuning the contour parameter vectors while preserving likelihood

Being Q the dimensionality of vectors in the space where the distribution g^* is defined, once an initial curve parameter vector $p_i \in \mathbb{R}^Q$ is generated from g^*, a subset of R components (with $R < Q$) contained in vector p_i must be tuned by adding a constraint vector $\delta p_i \in \mathbb{R}^Q$ of the form $\delta p_i = [\delta p_{i_1}, \cdots, \delta p_{i_R}, 0, \cdots, 0]^T$ in order to satisfy a set of constraints (see earlier in the section), obtaining a final vector p_f. In order to maintain the likelihood of the resulting parameter vector p_i in the new p_f, the non-fixed $Q-R$ components of vector p_i will be modified by adding a vector $\delta p_f \in \mathbb{R}^Q$ of the form $\delta p_f = [0, \cdots, 0, \delta p_{f_{R+1}}, \cdots, \delta p_{f_Q}]^T$, so that the squared Mahalanobis distance ${D^2}_M(p_f, p_i)$ between the initial and final vectors p_i and p_f is minimized.

Expressing δp_f as the product of a $Q \times (Q-R)$ selector matrix A and a vector $\delta p_{f_0} \in \mathbb{R}^{Q-R}$ of the form $\delta p_{f_0} = [\delta p_{f_{01}}, \cdots, \delta p_{f_{0Q-R}}]^T$, p_f and A are written as

$$p_f = p_i + \delta p_i + A\delta p_{f_0} \tag{4.43}$$

$$A = \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \tag{4.44}$$

The problem is to obtain the final vector p_f by means of finding the optimal ${\delta p_{f_0}}^*$. Being fixed the initial vector p_i, the constraint vector δp_i, and the selector matrix A, the search of ${\delta p_{f_0}}^*$ can be expressed as in (4.45), where Σ^* represents the covariance matrix of g^*.

$$\begin{aligned} {\delta p_{f_0}}^* &= \underset{\delta p_{f_0}}{\operatorname{argmin}}\, f(\delta p_{f_0}) \\ &= \underset{\delta p_{f_0}}{\operatorname{argmin}}\, {D^2}_M(p_f, p_i) \\ &= \underset{\delta p_{f_0}}{\operatorname{argmin}}\, (p_f - p_i)^T {\Sigma^*}^{-1} (p_f - p_i) \end{aligned} \tag{4.45}$$

By using equation (4.43), equation (4.45) can be rewritten as

$${\delta p_{f_0}}^* = \underset{\delta p_{f_0}}{\operatorname{argmin}}\, (\delta p_i + A\delta p_{f_0})^T {\Sigma^*}^{-1} (\delta p_i + A\delta p_{f_0}) \tag{4.46}$$

In order to solve the problem, the gradient $\nabla_{\delta p_{f_0}} f(p_{f_0})$ must equal zero, yielding

$$\nabla_{\delta p_{f_0}} f(\delta p_{f_0}) = 2{\delta p_i}^T {\Sigma^*}^{-1} A + 2A^T {\Sigma^*}^{-1} A\delta p_{f_0} = 0 \tag{4.47}$$

By solving for δp_{f_0}, the solution for ${\delta p_{f_0}}^*$ is obtained as

$${\delta p_{f_0}}^* = -(A^T {\Sigma^*}^{-1} A)^{-1} A^T {\Sigma^*}^{-1} \delta p_i \tag{4.48}$$

4.4.3 Rendering results

The flexibility and robustness of the proposed framework is successfully tested was rendering bowing contours for a number of notes spanning all four bowing

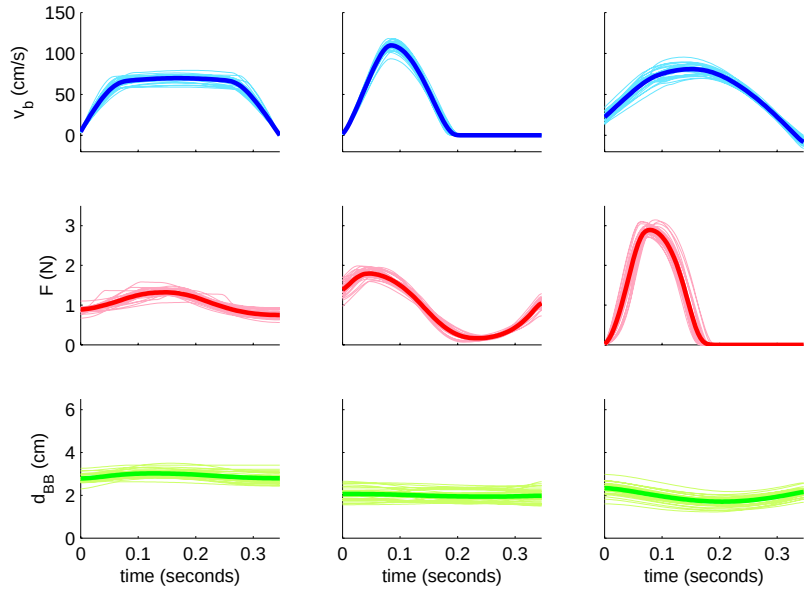


Figure 4.15: Synthetic bowing contours for different bowing techniques. From left to right, [*détaché ff downwards iso mid*], [*staccato ff downwards iso mid*], and [*saltato ff downwards iso mid*]. Thin lines correspond to 30 generated contours, while the thick line in each plot corresponds to the time-average contour.

techniques, the two bow directions, the three dynamics, and different slur and silence contexts. Some representative synthetic bowing contours, corresponding to the four different bowing techniques, are depicted in Figures 4.15 and 4.16, where contours of *legato*-articulated notes are shown for three different slur contexts. In order to demonstrate the consistency of the model even while keeping the original stochasticity, the contour synthesis process was repeated 30 times for each note. Also, when comparing synthetic curves to acquired signals for a given note class, the fidelity of the model is clearly noticed (compare for instance the contours of Figure 4.16 with those displayed in Figure 3.5, corresponding to analogous note classes, but only differing in bow direction).

An extended set of synthetic contours, including most of the note classes, can be visualized in Appendix B. Apart from showing sets of rendered contours obtained with two variance scaling factors (see Section 4.2.3), rendering is also carried out for different values of note duration, effective string length (pitch), and bow starting position. While the response of the model to different durations and effective string lengths is satisfactory (the shapes of contour smoothly

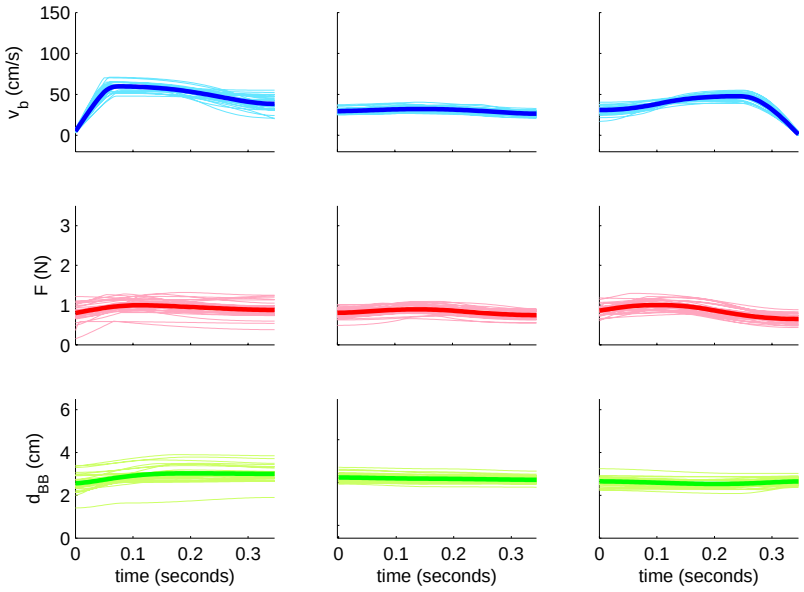


Figure 4.16: Synthetic bowing contours for different slur contexts of *legato*-articulated notes. From left to right, [*legato ff downwards init mid*], [*legato ff downwards mid mid*], and [*legato ff downwards end mid*]. Thin lines correspond to 30 generated contours, while the thick line in each plot corresponds to the time-average contour.

vary with them), changes in contours due to the different bow starting positions are in general less importantly reflected in rendered contours. This could possibly be caused by the database size and coverage limitations, combined with the clustering process. A lack of variety of bow starting positions within the clusters may lead to this. It often happens that, because a low number of note samples is found for some duration clusters, the second step of the clustering process further divides the space into sub-clusters for which bow starting position coverage is poor. If this happens to clusters surrounding the target performance context point (see Figure 4.13), the correlation between curve parameters and bow starting position is not significant, as contrary to what was hypothesized in Section 4.2.1.

The modeling framework proves to be flexible, and provides high fidelity in most cases. However, it presents an important drawback that can be seen as going beyond its performance dependency on the database size and coverage limitations: the model is limited to the convex hull defined by the normal distributions that populate both the performance context space and the bowing

contour space, making it impossible to generalize out of such limits.

4.5 An approach to the emulation of bow planning

For a note to be executed in the context of sequence found in a score, the starting bow position and the bow displacement are chosen by the performer, who is able to plan the sequence of starting bow positions BP_{ON} and bow displacements ΔBP, based on the constraints imposed by the finite length of the bow and on own preferences. In order to represent possible particularities in note executions given the different bow starting position and bow displacement possibilities, a bow planning algorithm is devised in an attempt to take those into account.

Up to this point, no bow planning considerations were present during the process of rendering contours of isolated notes. On one hand, the target bow starting position BP^t_{ON} used for obtaining the curve parameter distribution g^* was set to the mean bow starting position α^{n^*} (see equation (4.6)) of the samples contained the selected duration cluster n^* (see Section 4.4.1).

The bow displacement is not used as a performance context parameter, but as an extra dimension of the curve parameter space. This enables the modification of the rest of curve parameters of a generated vector p for matching a desired ΔBP^t by attending to the curve parameter distribution g^* (see equation 4.42).

By means of the algorithm introduced in this section, a sequence of BP^t_{ON} and ΔBP^t corresponding the notes appearing in an annotated input score is obtained so that (1) the finite length of the bow is respected and (2) the original distributions of bow starting position and bow displacement for each note class and duration (respectively represented by a and b, see equations (4.6) and (4.7)) are taken into account. The obtained values for BP^t_{ON} and ΔBP^t are used for rendering individual notes of the sequence, therefore contributing to the synthesis of continuous contours of bow velocity, bow force, and bow-bridge distance for a given input score.

4.5.1 Algorithm description

The bow-planning problem is set as the task of finding an optimal sequence of note bow starting and ending positions, represented by the vector

$$BP^* = [{BP_{ON,1}}^* \, {BP_{OFF,1}}^* \cdots {BP_{ON,N}}^* \, {BP_{OFF,N}}^*] \tag{4.49}$$

in which the starting bow position of each note matches the ending bow position of the previous note in the sequence:

$$BP_{ON,n} = BP_{OFF,n-1} \;\; \forall n = 2, \ldots, N \tag{4.50}$$

For doing so, it is devised the state transition matrix represented in Figure 4.17, for which the number of columns is fixed by the number of notes in the se-

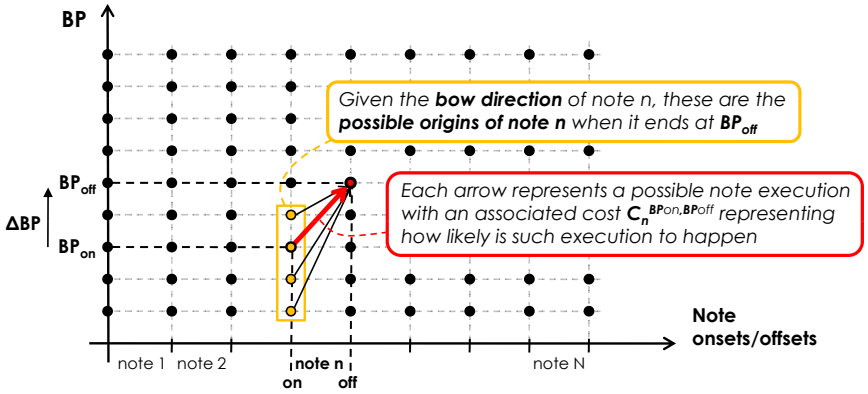


Figure 4.17: Schematic illustration of the state transition matrix on which the bow planning algorithm is based. For each note in the input sequence, all possible combinations of starting and ending bow positions BP_{ON} and BP_{OFF} are assigned an execution cost C.

quence, and the number of rows is arbitrarily set by the desired resolution used for representing the bow position $BP \in [0, l_B]$, with the bow length l_B being equal to 63cm.

In Figure 4.17, each n-th note is represented by an arrow going (1) from its onset to its offset in the x-axis (rows), and (2) from a starting bow position BP^n_{ON} to a ending bow position BP^n_{OFF} in the y-axis (rows). All possible executions of a note (associated to possible combinations of starting and ending bow positions) are assigned an execution cost C_n (see Section 4.5.3). The optimal path, represented by the vector BP^*, is found as the path minimizing the total cost $\Psi(BP)$,

$$BP^* = \underset{BP}{\operatorname{argmin}}\, \Psi(BP) \tag{4.51}$$

having the the total cost $\Psi(BP)$ defined as the sum of the execution costs of the notes:

$$\Psi(BP) = \sum_{n=1}^{N} C_n^{BP^n_{ON}, BP^n_{OFF}} \tag{4.52}$$

For each note in the sequence, the first step is to assign it a note class. Then, scripted note duration D^t and the effective string length l^t_e (the latter computed from the scripted pitch and string) are obtained from the input score. In order to complete the performance context target vector s^t, the starting bow position BP^t_{ON} is needed. Also, a bow displacement ΔBP^t is needed for rendering the

contours of a note. These two values come determined from each possible execution of the note, i.e., each possible combination of $\{BP_{ON}, BP_{ON}\}$ with $BP_{ON}, BP_{OFF} \in [0, 63]$.

In the algorithm, any value of BP_{OFF} is allowed for every note, and the bow direction of the note will define the possible values of BP_{ON}, as it is expressed in:

$$BP_{ON} \in \begin{cases} [\,0, BP_{OFF}\,) & \text{if } \textit{downwards} \text{ bow direction,} \\ (\,BP_{OFF}, L_B\,] & \text{if } \textit{upwards} \text{ bow direction.} \end{cases} \tag{4.53}$$

The solution for BP^* is found by using dynamic programming techniques (Viterbi, 1967). Relatively short rests or silences present in a note sequence are considered as if they were normal notes, with the aim of representing the bow traveling distance constraints associated to them. For that, they are treated as short *détaché* notes, using a *détaché* articulation model with duration-dependent dynamics for emulating different bow velocities. When needed, any given input score might be divided into several sequences of notes depending on the length of the rests or silences found.

4.5.2 Contour concatenation

Bowing contour of successive notes are concatenated within the bow planning framework by taking advantage of the partial optimal path search that characterizes the *Viterbi* algorithm. For a given i-th note and a given ending bow position $BP_{OFF,i}$, the curve parameter vectors p_f (each associated to a candidate $BP_{ON,i}$ and used for computing its associated execution cost, see Section 4.5.3) are known when computing the cost associated to the $(l+1)$-th note. For those candidate executions of the $(i+1)$-th note which present their starting bow positions $BP_{ON,i+1}$ matching $BP_{OFF,i}$, the starting values of the three contours of the $i+1$-th note will be set to the ending values obtained for the i-th note ending at $BP_{OFF,i}$. Setting these values is therefore considered as a contour parameter constraint to be adjusted before computing the cost associated to the note (see end of Section 4.4.2).

4.5.3 Cost computation

Once a curve parameter vector p_f has been obtained from original p and g^* by applying the set of modifications needed for fulfilling the rendering constrains presented in previous section (including those carried out for matching the desired $\Delta BP'$) for a candidate execution, its associated cost C is computed as follows. Given the set of K *source* distributions $g^{n^*,h(i)}$ used for obtaining the mixed distribution g^* (see Section 4.4.1), the cost C is computed as a weighted sum of K negative log-likelihoods of the vector p_f, each one computed given the corresponding i-th original distribution, plus two additional costs. This is expressed in equation (4.54), where ω_i represents the weight applied to each of the likelihoods, and η and λ are additional costs respectively related to the

bow displacement ΔBP and bow starting position BP_{ON} of the current execution candidate.

$$C = \eta + \lambda + \sum_{i=1}^{K} \omega_i \, n \log L(p_f \mid g^{n^*,h(i)}) \tag{4.54}$$

The value of ω_i used for weighting each i-th likelihood is computed from the Mahalanobis distance from the target performance context point $s^t = [\, D^t \; BP^t_{on} \; L^t_{st} \,]$ to the centroid $\gamma^{n^*,h(i)}$ of the i-th *source* distribution $\nu^{n^*,h(i)}$ (i.e., the original weights applied for combining the source distributions, see Section 4.4.1), computed as in equation (4.16).

In order penalize note executions for which the bow displacement ΔBP (determined by the candidate execution) is not likely to happen, it is introduced the cost η, computed as the negative log-likelihood of ΔBP given the bow displacement distribution b^{n^*} associated to the selected duration cluster n^* from the model being used (see Section 4.3.2). The penalty cost η is expressed as:

$$\eta = n \log L(\Delta BP \mid b^{n^*}) \tag{4.55}$$

Analogously, note executions for which the bow starting position BP_{ON} (determined by the candidate execution) are unlikely to occur are penalized by λ, computed as the negative log-likelihood of BP_{ON} given the distribution a^{n^*} of bow starting position, associated to the selected duration cluster n^* from the model being used (see Section 4.3.2). The penalty cost λ is expressed as:

$$\lambda = n \log L(BP_{ON} \mid a^{n^*}) \tag{4.56}$$

Of course, the introduction of these costs makes the algorithm results to depend on the contents of database. In general, given the mix distribution g^*, the likelihood of a tuned curve parameter vector p_f should match that of the originally generated p because of applying the procedure introduced at the end of Section 4.4.2. Therefore, the third cost term in (4.54) (comprising the sum) is very sensible to the likelihood of the originally generated p. However, including the penalties η and λ enables finding an optimal path (bowing plan) for which (1) bow starting positions of notes respond to the original performer preferences, and (2) unfeasible bow displacements (which indeed imply extreme transformations of p into p_f) are discarded. Different weights could be applied to each of the three terms in the need of emphasizing one particular aspect.

Results and discussion

The sequence of bow starting/ending positions for a group of motifs presents in the database are compared in Figure 4.18 to their performed performance values. The actual onset/offset times of the recorded notes were used for the input score (displayed at the top). In general, results of the bow planning

algorithm proved to be consistent reliable. Apart from lacking certain flexibility in the representations of bow starting position or bow displacement preferences for notes of different classes and durations (they are inferred from the database), a problem is found when the curve parameter distributions g are estimated from clusters presenting a small number of samples, leading to a bad model of the correlation of the different curve parameters. In such cases, whenever an extreme bowing displacement is forced (e.g., rendering a long *forte détaché* note while imposing a very short bow displacement), obtained curve parameters are totally unreliable, leading to unrealistic bowing contours. Fortunately, those situations are avoided by the introduction of the bow displacement penalty in the cost computation.

4.5.4 Rendering results

By combining the bow planning algorithm to the contour rendering method for isolated notes (introduced in previous Section), synthetic bowing controls can be obtained from an annotated input score. Although it represents a first approximation founded on solid principles strongly linked to the physical constrains introduced by the finite length of the bow, it is difficult to evaluate the actual necessity of the bow planning algorithm for the modeling framework, since its performance is strongly linked to the database contents. Even when leaving out a database phrase during the construction of the models, the results of the bow planning algorithm proved to be reliable and realistic (see Figure 4.18).

Some representative results of bowing contour synthesis are shown in Figures 4.19, 4.20, and 4.21, for which existing scores in the corpus were used as input to the rendering framework (contours displayed in Figure 4.21 correspond to the first motive in Figure 4.18). By using note onset/offset times of the recorded performances instead of the nominal times, it is possible to visually compare the obtained contours to the acquired ones.

Due to unbalanced database note distribution, the number of notes belonging to some of the note classes happened to be much smaller than that belonging to other classes, thus having some performance context sub-clusters to contain very few curve parameter vectors. Accounting for possible errors during contour analysis (segmentation and fitting), the statistical description of such clusters remains less reliable. In these cases, a specific treatment is given, since the policy of iteratively reducing the number of performance context sub-clusters (see Section 4.3.3) often leads to obtaining a single cluster. This solution, which is designed to artificially avoid problems derived from high variances in estimated curve parameter distributions, is founded on the same basis as the possibility of modulating the stochasticity of contour shapes: covariance matrices of less populated sub-clusters are decomposed into eigenvalues and eigenvectors, and the eigenvalues are scaled by a factor inversely proportional to a relative measure of the number of vectors used when estimating the distribution parameters.

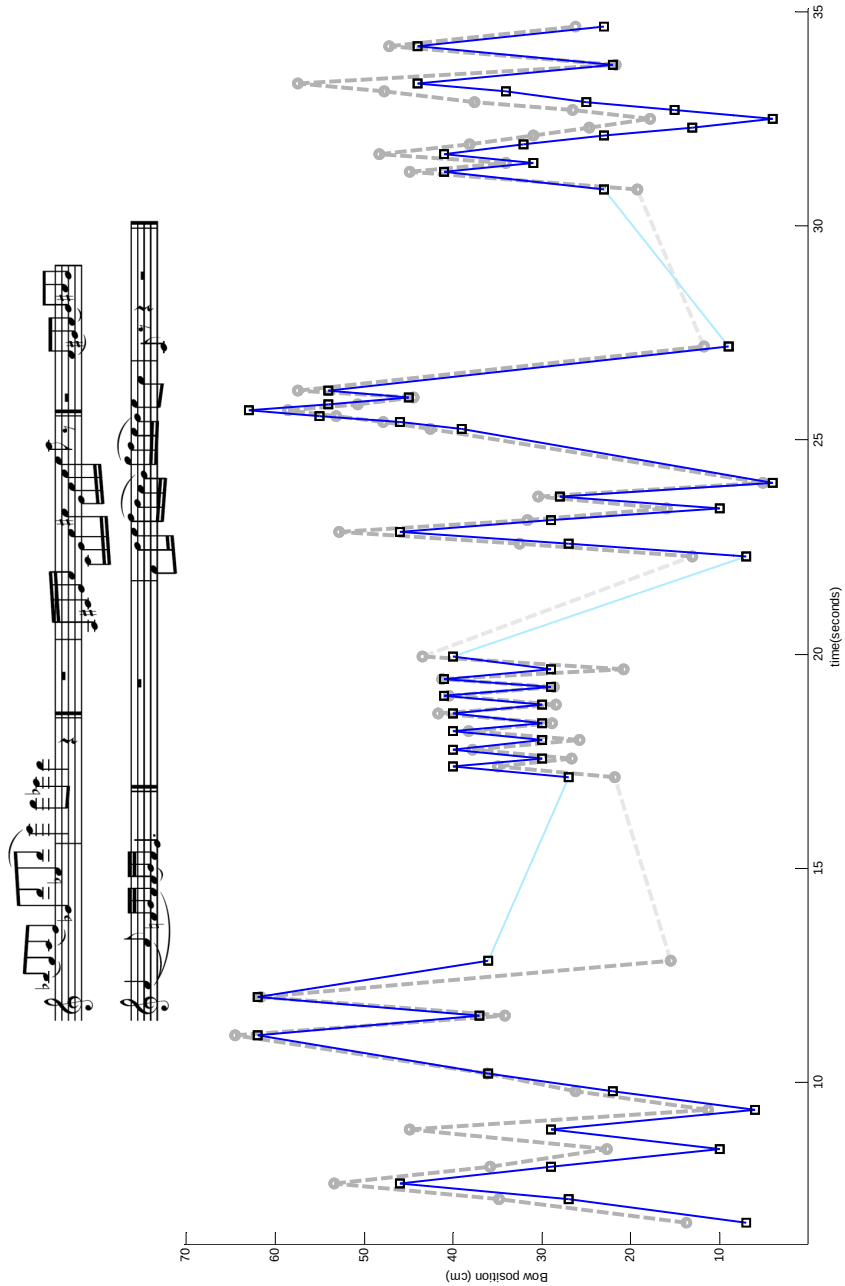
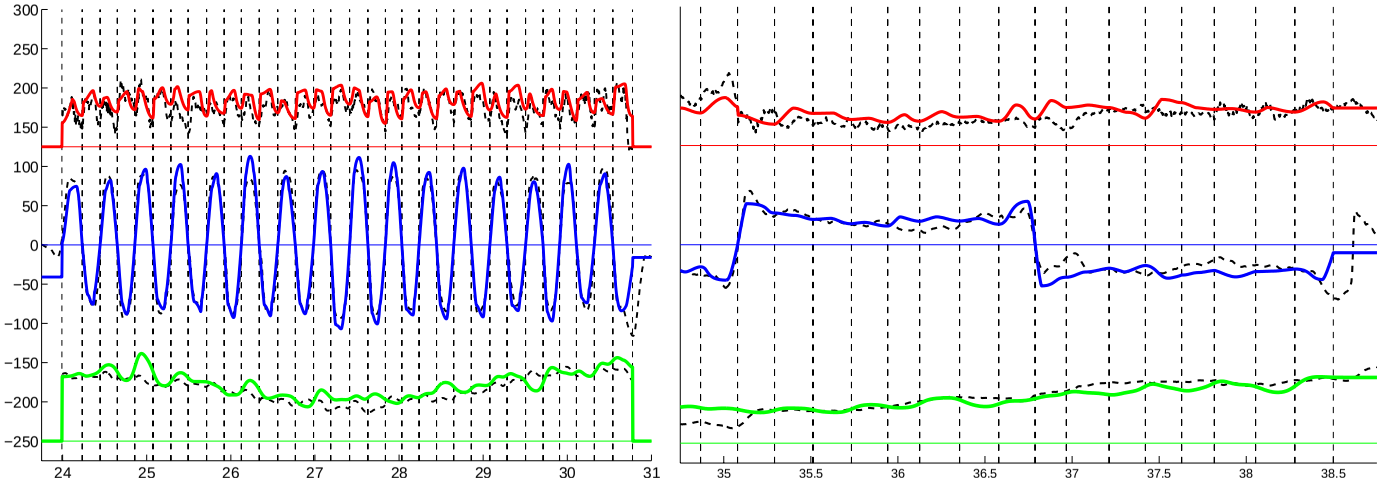


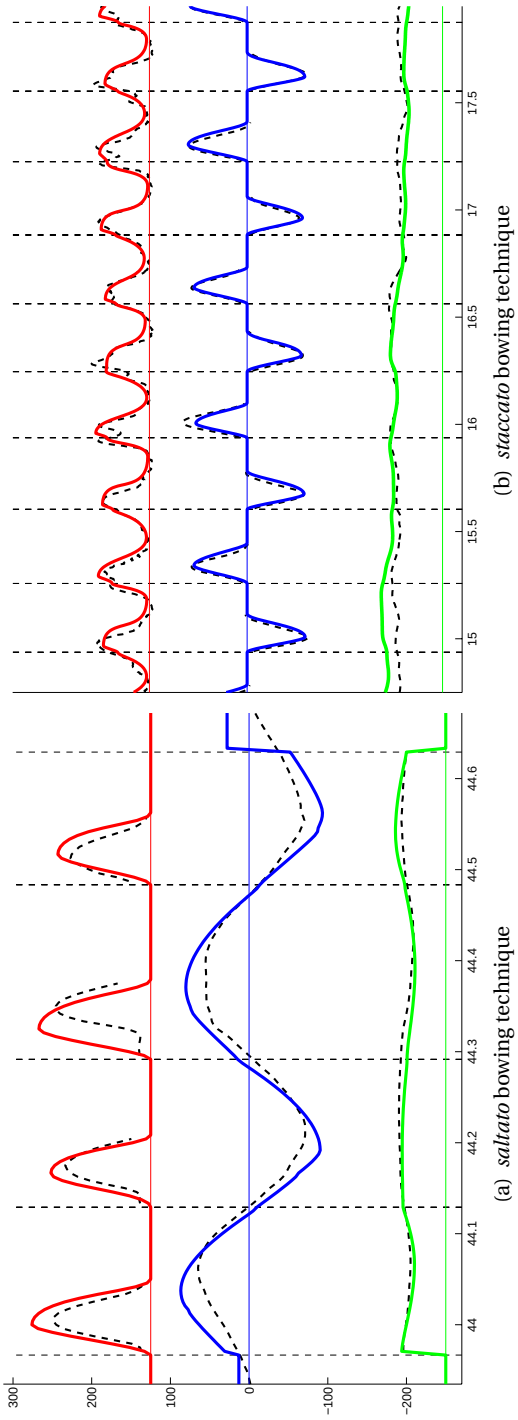
Figure 4.18: Results of the emulation of bow planning for a group of four consecutive motifs presents in the database, including *détaché*- and *legato*- articulated notes, played at *forte* dynamics. Thick dashed grey segments represent the sequence of bow displacements of the original performance, with note onsets/offsets marked with circles. Thin solid blue segments correspond to the sequence of bow displacements obtained by the algorithm, with note onsets/offsets marked with squares. Lighter segments correspond to scripted silences. The original phrase was left out when constructing the models.



(a) *détaché*-articulated notes

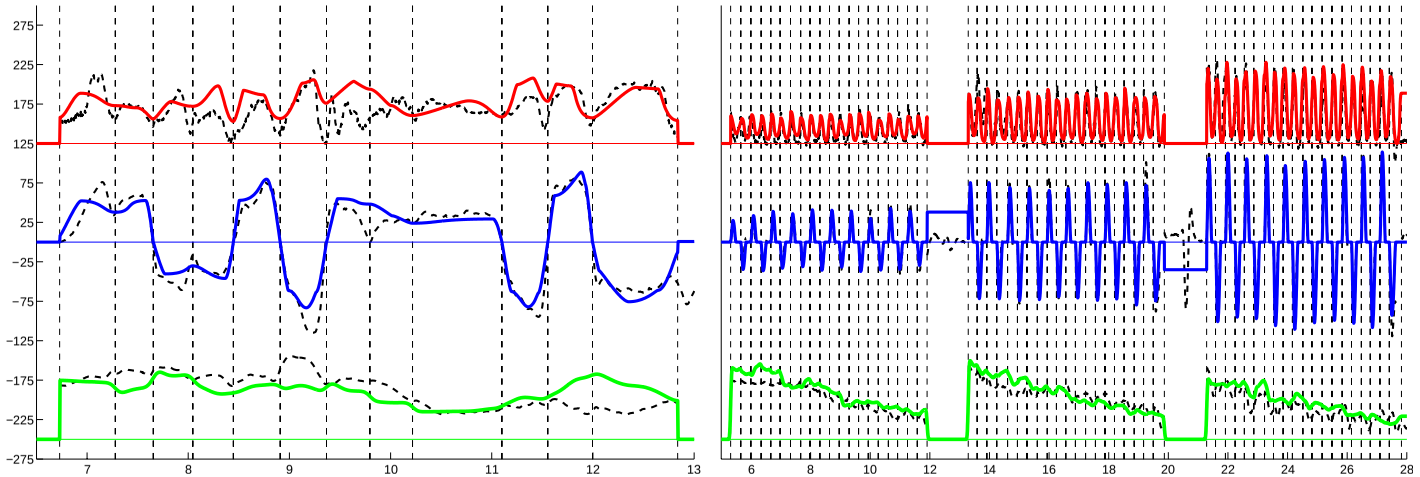
(b) *legato*-articulated notes

Figure 4.19: Rendering results of bowing parameter contours. From top to bottom: bow force ($0.02N/unit$), bow velocity (cm/s), and bow-bridge distance ($0.04cm/units$). Horizontal thin lines correspond to zero levels, solid thick curves represent rendered contours, and dashed thin curves represent acquired contours. Vertical dashed lines represent note onset/offset times (seconds).



(a) *saltato* bowing technique (b) *staccato* bowing technique

Figure 4.20: Rendering results of bowing parameter contours. From top to bottom: bow force ($0.02N/units$), bow velocity (cm/s), and bow-bridge distance ($0.04cm/unit$). Horizontal thin lines correspond to zero levels, solid thick curves represent rendered contours, and dashed thin curves represent acquired contours. Vertical dashed lines represent note onset/offset times (seconds).



(a) Alternating *détaché* and *legato* articulations

(b) Ascending scales of *staccato* notes played with *piano*, *mezzoforte*, and *forte* dynamics

Figure 4.21: Rendering results of bowing parameter contours. From top to bottom: bow force ($0.02N/units$), bow velocity (cm/s), and bow-bridge distance ($0.04cm/units$). Horizontal thin lines correspond to zero levels, solid thick curves represent rendered contours, and dashed thin curves represent acquired contours. Vertical dashed lines represent note onset/offset times (seconds).

4.6 Summary

A novel method for synthesizing violin bowing controls from an input score has been presented in this chapter. The methodology is based on statistical analysis and modeling of bowing parameter quantitative descriptions (i.e., Bézier curve parameters) that have been obtained from an annotated performance database as introduced in Chapter 3. The first part of the chapter is devoted to the synthesis of bowing parameter contours from a series of annotations of an isolated note. Each note in the database is represented in two different spaces: a contour parameter space and a performance context space. While the former is defined by the obtained Bézier curve parameter values, the latter is constructed from contextual note characteristics not dealing with the bowing parameter contours per se (e.g., note duration, finger position, etc.). A generative model based on multivariate gaussian mixtures is constructed so that both spaces are mapped, and a probability distribution of curve parameter values can be synthesized from an input vector of context parameters. From the obtained probability distribution, random vectors of curve parameters can be generated so that rendered curves mirror the shape of orginally acquired bowing parameter contours.

The chapter follows with a bow planning algorithm that integrates the statistical model for contour rendering while explicitely accounting, again by means of statistical modeling of acquired data, for the implications brough by one of the most important constrains in violin bowing: the finite length of the bow. From an input score annotated with bow direction changes, articulations or bowing techniques, and dynamics, the algorithm proposes the most probable sequence of bow starting and ending positions, and accordingly renders and coherently concatenates contours of bow velocity, bow pressing force and bow-bridge distance. Both the bow planning results, and synthetic bowing controls significanlty resemble original contours.

The instrumental gesture (bowing control) synthesis framework presented in this disseration outdoes any of the (few) previous attempts to render violin control parameters from an annotated input score. Having the generative model constructed from real performance data brings more fidelity and an objective groundtruth as compared to the earlier works by Chafe (1988), Jaffe & Smith (1995), and Rank (1999). A recent work by Demoucron & Caussé (2007); Demoucron (2008); Demoucron et al. (2008) constructs and modifies bowing parameter contour models from real data. In his work, bow velocity and bow force contours of different bow strokes are quantitatively characterized and partly reconstructed mostly using sinusoidal segments. While a simple univariate statistical analysis (mostly focused on isolated notes) is carried out, flexibility limitations of the proposed contour representation kept it (as pointed out by the author) to easily generalize its application to other bowing techniques not only based on bow strokes per se, but also on more sustained control situations (e.g., longer *détaché* notes or *legato* articulations).

The data-driven statistical models introduced in this chapter bring new pos-

sibilities for studying instrumental gestures (bowing control in particular), and open paths for different analysis applications (e.g., bowing technique automatic recognition). One of the most straightforward applications is, however, the use of renderd controls for driving instrumental sound synthesis. Next chapter is devoted to the application of synthetic bowing parameters for generating sound using the two most extended sound synthesis techniques: physical models and sample-based concatenative synthesis.

Chapter 5

Application to Sound Synthesis

An application use and validation of the developed instrumental gesture modeling framework is presented in this chapter. Synthetic bowing parameter contours obtained from an annotated input score are used as a key component for synthesizing realistic, natural sounding violin sound through the explicit inclusion of instrumental control information in two of the most common sound synthesis approaches: physical modeling synthesis and sample-based synthesis. The author's article that is most related to this chapter was recently accepted for publication (Maestre et al., 2010), and it presents the bowing control modeling framework applied to sound synthesis. Its contents are extended by the details given in the sections below.

As already outlined in Section 1.3, the introduction of gestural control as a key component for the synthesis of excitation-continuous musical instrumental sound brings significant benefits along the line drawn from the particular limitations that are inherent to physical modeling and to sample-based synthesis (see Figure 1.5). On one hand, the availability of appropriate instrumental control signals improves the sound realism and naturalness that can be achieved by means of physical models, traditionally constrained by the lack of means for their automatic control. On the other hand, the limited control flexibility of sample-based approaches is improved by enriching the sample retrieval and transformation processes with meaningful gesture-based functions, leading to an enhancement of the timbre continuity that helps the obtained sound to be perceived as more natural.

The general block diagram of the application of synthetic bowing parameters for synthesizing sound is depicted in Figure 5.1. For the case of physical modeling (upper part), synthetic bowing parameters are directly used to drive a digital waveguide physical model, having the string vibration synthetic signal convolved with a violin body impulse response estimated from real data. In the sample-based framework (lower part), rendered contours are explicitely used both during sample retrieval and during sample transformation (crucial stages of the synthesis process) of a spectral-domain synthesizer that makes use of a

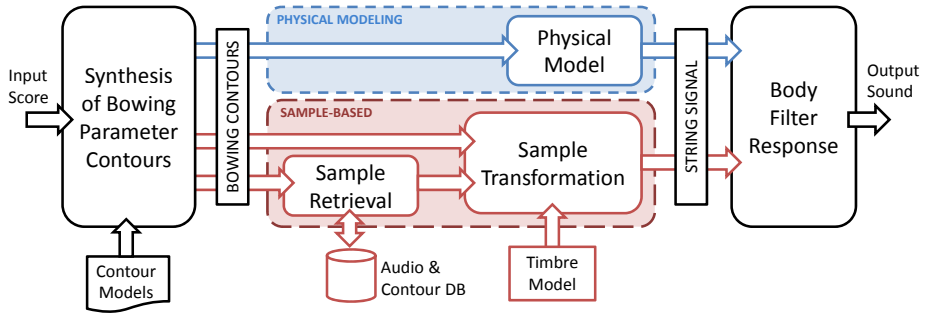


Figure 5.1: Schematic illustration of the two main sound synthesis applications of the bowing parameter modeling framework presented in this work.

synchronized audio/gesture database, having the resulting sound signal (also corresponding to the string vibration[1]) also convolved with the estimated body impulse response.

5.1 Estimation of the body filter impulse response

In the application context of this work, the body filtering effects taking place during violin performance are considered to be linear and invariant (Cremer, 1984). For the case of physical modeling synthesis, the obtained sound signal corresponds to the string velocity at the bridge, so it needs to be convolved with an estimated impulse response respresenting the bridge and the body of the instrument. For the case of sample-based synthesis, the database is constructed from the signal acquired with a bridge pickup, so an estimation of the impulse response is needed.

A body filter response (understood as the bridge pickup to radiated sound transfer function response) is estimated by means of deconvolving acquired radiated sound signal (microphone) with acquired string vibration signal (bridge pickup) sound signal similarly as described by Karjalainen et al. (2000), but in a frame-by-frame time-averaging fashion.

For doing so, it is recorded a long-duration *glissando* played on the G string (lower pitch) in order to capture low frequency -content excitation happening during violin performance. For the acquisition of the radiated sound, the microphone is placed in front of the violin and the performer is asked to keep in the same position and orientation while playing the *glissando*, so that a low variabil-

[1]The audio in the database is acquired from a bridge piezoelectric pickup as described in Section 2.2.

ity on distance and orientation minimizes inconsistencies in the time-average bin-per-bin estimation.

The magnitude for each frequency bin is estimated as the average of individual frame estimations (one per frame), each one weighted by the RMS energy of the frame. Conversely, the phase value for each bin is estimated as the maximum of a phase histogram that is constructed from individual phase estimations (one per frame), each one also weighted by the RMS energy of the frame.

5.2 Physical modeling synthesis

Bowing control parameter synthetic contours are used for driving a modified version of the *Synthesis Toolkit in C++* (STK)[2] implementation of the digital waveguide bowed-string physical model introduced by Smith (1992, accessed 2009), sketched in Figure 5.2. A single string model with a low-pass one-pole implementation of the loss filter is chosen as a proof-of-concept physical model for validating the bowing parameter synthesis framework.

The digital waveguide bowed string model used is illustrated in Figure 5.2. The right delay-line pair carries left-going and right-going velocity waves samples $v^+_{s,r}$ and $v^-_{s,r}$, respectively, which sample the traveling-wave components within the string to the right of the bow, and similarly for the section of string to the left of the bow. The '+' superscript refers to waves traveling into the bow. String velocity v_s at any point is obtained by adding a left-going velocity sample to the right-going velocity sample immediately opposite in the other delay line (this happens in Figure 5.2 at the bowing point). The loss filter $h_s(t)$ represents the losses at the bridge, bow, nut or finger-terminations, and the total attenuation from traveling back and forth on the string. The bow-string non-linear interaction is driven by means of the differential velocity v^+_Δ (bow transversal velocity v_b minus string velocity v_s) and the bow force F which, by modifying the shape of the bow table function providing the real-valued reflection coefficient ρ, define the proportion between waves being reflected by and traveling through the bow. Bow-bridge distance modifies the proportion of the delay line length L (corresponding to a given pitch) that goes into the bow-nut delay line length L_N and the bow-bridge delay line length L_B.

The output string velocity signal is convolved with a body filter impulse response. The impulse response can be obtained by different means. A first alternative is to estimate it as described Section 5.1. Radiated sound can be measured with a microphone in front of the violin after exciting the bridge with an impulse hammer, as described by Karjalainen & Smith (1996). Also, by inducing a force or velocity on the string, close to the bridge, it is possible to deconvolve radiated sound with the signal used to induce such force or velocity, as described by Farina et al. (1995); Farina (2007). If there is no need for using

[2] http://ccrma.stanford.edu/software/stk/

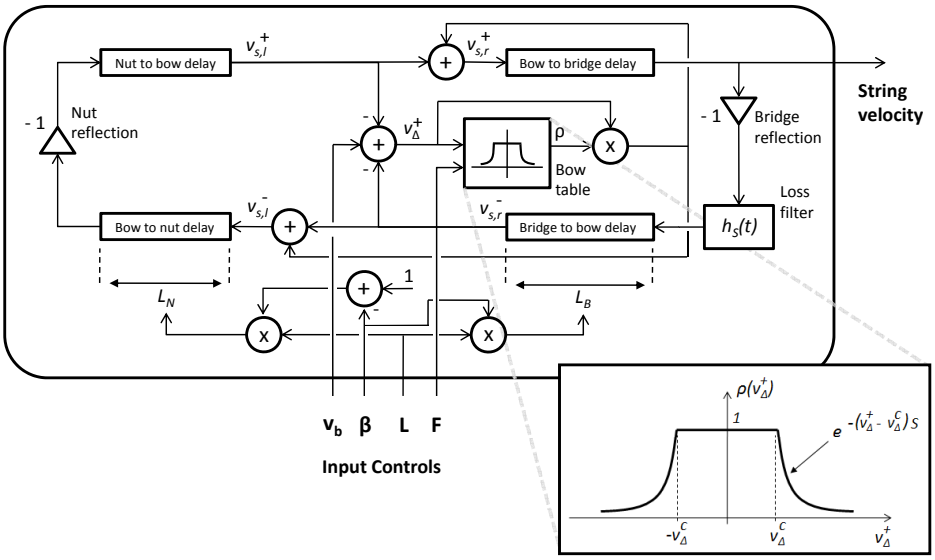


Figure 5.2: Smith's digital waveguide bowed-string physical model. Performance controls include bowing control parameters (bow velocity v_b, bow force F, β ratio), and delay line lengths L_N and L_B (derived from the string length for a given pitch). The zoom provides a closer view of the look-up function in charge of providing the bow reflection coefficient ρ.

the exact body of the violin used for acquiring bowing data, a number of body impulse responses can be accessed and tested online (Cook & Trueman, 1998). Note that the actual magnitude measured by the bridge pickup (first alternative) does not exactly correspond either to a string velocity-like or to a string force-like signal, so using an impulse response obtained from the bridge pickup yields particular timbre characteristics that differ from those perceived in the output sound when convolving the string velocity signal with an impulse response obtained from a string velocity signal as described by Farina et al. (1995); Farina (2007), which results into a more realistic sound. In orther to overcome such issue, the transfer function between the string velocity signal and the signal measured by the pickup could be estimated by deconvolution of the pickup signal with the string velocity signal, in case the latter was available.

5.2.1 Calibration issues

Digital waveguide physical model calibration represents a challenging problem not addressed here. Left-hand articulation-related control parameters (e.g.

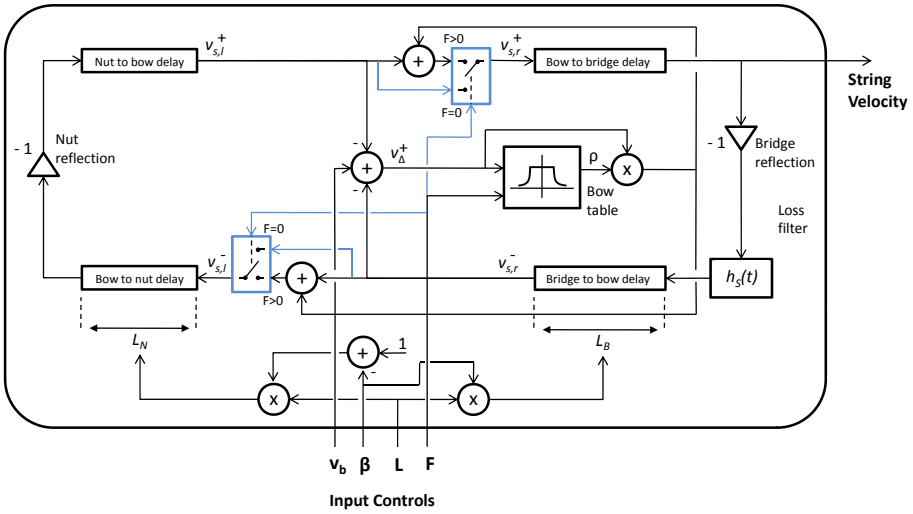


Figure 5.3: Modified structure of the digital waveguide physical model (additions are depicted in blue), in order to allow for off-string bowing conditions derived from zero-valued bow force as synthesized through the bowing modeling framework.

vibrato) are ignored, so pitch transitions are considered to be instantaneous (implying sudden changes on the delay line length L). String terminations are each represented by a reflection coefficient $\rho_0 = -1$. For the loss filter $h_s(t)$ of the single-string model, gain value and pole positioning (single pole) are set manually. While both bow transversal velocity v_b and β ratio are used directly in the model, bow force F is used to configure the shape of bow table. The bow table is defined (see Figure 5.2) by a break-away differential velocity v_Δ^c and an exponential decaying factor S, both depending on F (Smith, 1992, accessed 2009). Two linear mapping functions $v_\Delta^c = f(F)$ and $S = f(F)$ are manually tuned.

Given the availability of bowing control parameter signals aligned to audio captured by the pickup, an automatic calibration procedure might be developed by means of optimization techniques. Preliminary statements for solving the problem are shortly introduced here, although a successful implementation has not been achieved yet.

The main idea consists on finding the set of calibration parameters (i.e., the two linear mapping functions $v_\Delta^c = f(F)$ and $S = f(F)$, the gain value and pole of the loss filter $h_s(t)$, and the reflection coefficients corresponding to the bow table and to the string terminations) that minimizes a cost composed by a weighted sum of spectral envelope distances between synthetic sound and

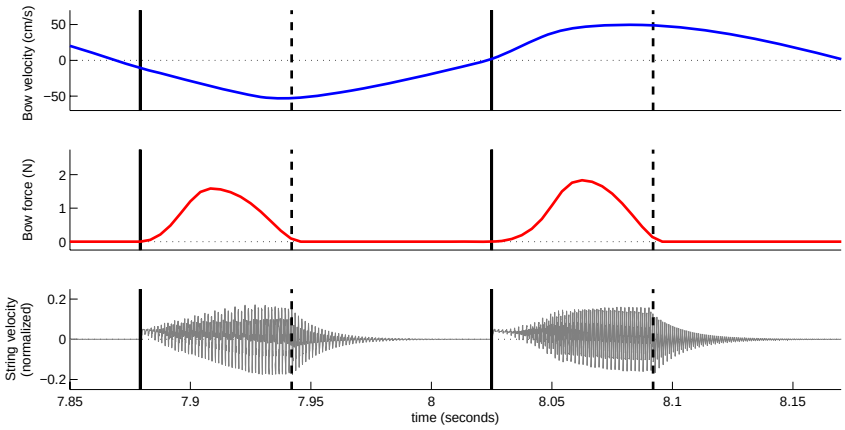


Figure 5.4: Obtained string velocity signal for alternating on-bow and off-bow bowing conditions (in particular, successive *saltato*-articulated notes). Vertical solid lines represent note onsets, while vertical dashed lines correspond to the times of bow release (bow force reaches zero).

recorded sound, having each distance computed in a different point of the space of bowing control parameters. However, since the magnitude measured by the pickup does not correspond to the string velocity, the use of spectral envelope distances computed from these two audio signals appears as invalid. Hopefully, this could be tackled by assuming that there exists a linear, time-invariant filter whose magnitude response can be used to approximate that of the transfer function that relates the string velocity signal and the signal captured by the pickup. Since this filtering effect (recall that it is assumed to be linear and time-invariant) would come into play by equally distorting spectral envelope distances computed in all points of the space of bowing control parameters, the optimization process would involve, at each step, finding the parameters that approximate the magnitude response of such filter, and a posterior frequency domain-filtering of the synthesized frames before computing the spectral envelope distances. At each step of the optimization procedure, the process of estimating the spectral envelope of the filter can be approached by means of least-squares techniques, as it is detailed in the Appendix C.

5.2.2 Incorporating off-string bowing conditions

In the digital waveguide configuration shown in Figure 5.2, the reflection coefficient returned by the lookup table for differential velocities $|\, v_\Delta^+ \,| < v_\Delta^c$ is always $\rho = 1$. From a physical-modeling point of view, such reflection coefficient should morph to zero as the bow is lifted (i.e., as the force approaches zero), regardless of the value of the differential velocity v_Δ^+ (in other words, the table

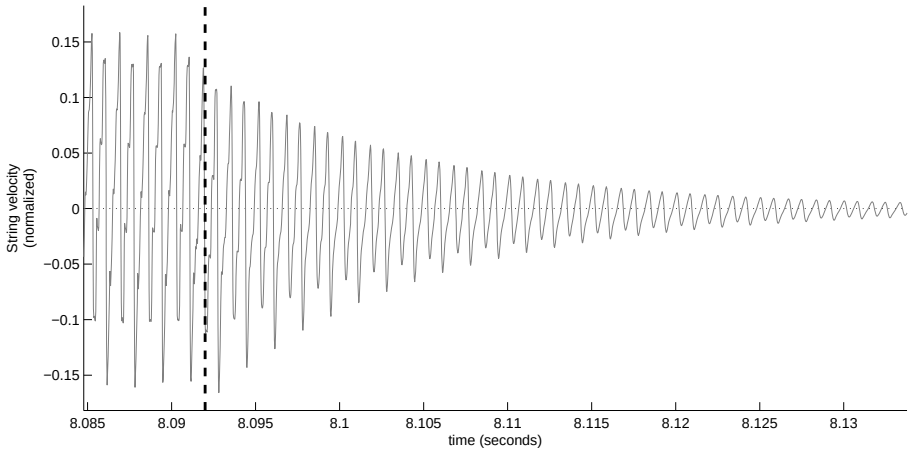


Figure 5.5: Detail of the damping of the string velocity signal after the bow releases the string. The vertical dashed line corresponds to the bow release time.

should morph to zero, towards a no operation table). That implies the incorporation of an additional mapping function $\rho_T = f(F)$ (also to be calibrated) providing the reflection coefficient $\rho = \rho_T$ to be returned by the table when $| v_\Delta^+ | < v_\Delta^c$, instead of the original $\rho = 1$. An ideal calibration of such function should provide smooth transitions between on-string and off-string bowing conditions.

As a practical implementation alternative to the physical-modeling approach described above for incorporating off-string bowing conditions (e.g. *saltato* articulation), bow-string interaction (represented by the bow table) is instead bypassed when bow force becomes non-positive. This implies that the left and right delay lines are connected, leading to a string damped (whose damping characteristics are due to the loss filter defined by $h_s(t)$) ringing when the bow is not in contact with the string. The digital waveguide configuration is slightly modified as shown in blue color in Figure 5.3, having the incoming force value compared to zero in order to override the non-linear effect introduced by the bow table.

Figures 5.4 and 5.5 (the latter showing a detail of the former) depict the damping suffered by the string velocity signal during after the bow releases the string. In Figure 5.5 it can also be observed how the high frequency content of the string velocity gets attenuated due to the low-pass nature of the string loss filter $h_s(t)$.

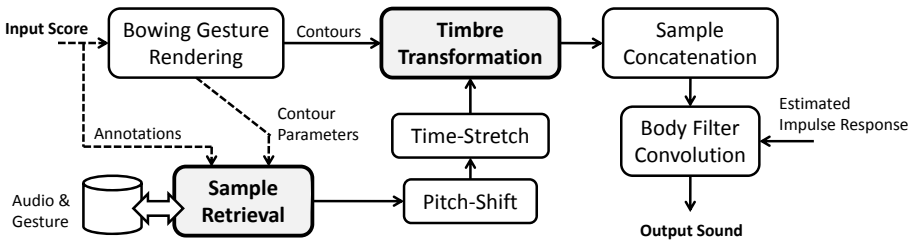


Figure 5.6: Overview of the sample-based synthesis framework. Synthesizer components making use of synthetic gesture data are highlighted. Dashed lines are used to indicate symbolic data flows.

5.2.3 Results

Despite having used the simplest expression of a digital waveguide bowed string model in which a proper calibration was not addressed, the perceived quality of obtained sounds raised unprecedented levels of realism and naturalness. As an exercise of subjective evaluation which also served for validating the bowing control modeling framework presented in this dissertation, sounds obtained from bowing parameter signals synthesized from a set of scores present in the database were compared to sounds obtained from the recorded bowing parameter signals corresponding to the same set of phrases. The subjects, including both trained and non trained musicians, were in general unable to consistently distinguish sounds obtained through recorded controls from those obtained by means of synthetic controls. Moreover, it often happened that sound excerpts obtained from synthetic bowing control were perceived as more realistic, especially for the case of discontinuous-excitation bowing techniques. One of the reasons behind this relies on certain inaccuracy in recorded bow force signal, which often presented a positive value during off-string bowing conditions (the bow is lifted) as opposed to the ideal value of zero (no force is being applied). Results and comparative examples can be found online[3].

5.3 Sample-based synthesis

A preliminary sample-based spectral-domain concatenative synthesizer is designed for incorporating the benefits of having synthetic bowing control parameter signals obtained from an input score. Conversely to the case of physical modeling, the classical sample-based synthesis structure suffers imporant modifications (see Figure 5.6).

[3] http://www.iua.upf.edu/~emaestre/phd/sounds/

The first and most important modification is the nature of the database: it contains samples of synchronized audio and gesture data. Secondly, a gesture-based timbre transformation component is introduced for performing audio modifications that are meaningful to the synthetic bowing gesture parameters obtained from an input score. Moreover, the generated curve parameters (from which the bowing parameter contours are synthesized at the first stage) take part in the sample retrieval process, leading to a selection of samples also based on instrumental control, going beyond classical search methods based only on symbolic annotations (e.g. pitch, duration, etc.) which do not account for any instrumental control-based timbre transformation cost or complex representations of context.

The database is constructed from the same samples used for modeling bowing control parameter contours. Along with the audio data, each sample holds the acquired bowing parameter signals and a curve parameter vector that describes the contour of such signals in a compact and robust manner (see Section 3.8).

Sample transformations are performed in the spectral domain to acquired bridge pickup audio signal, thus avoiding potential problems brought by body resonances and reverberation. Assuming the body filtering effect to be linear, the resulting audio signal is convolved with the body impulse response previously obtained by deconvolution as briefly described in Section 5.1. Concatenation is applied as a last step, following the spectral domain techniques described by Bonada & Serra (2007).

5.3.1 Sample retrieval

By attending both to the score-performance alignment carried out when constructing the database (see Section 2.6), and to the results provided by the analysis of bowing parameter contours presented in Chapter 3, each note in the database is annotated with the following information:

- **Symbolic annotations**:
  - String being played.
  - Note duration, obtained from score-performance alignment (see Section 2.6.4).
  - Note pitch, obtained from the input score.
  - Pitch interval to preceding and following note, if applies.
  - Note class (see Section 3.4).
- **Bowing annotations**:
  - Curve parameter vector p_s (see Section 3.8).

In a first step, a sample candidate list is generated for each of the N notes in the input score by attending to different sample annotations. The candidate list for a given note in the input sequence is populated with note samples matching (1) played string, and (2) the note class into which they were classifying during the contour analysis stage, i.e., articulation, bow direction, dynamics, slur context, and silence context (see Section 3.4).
Then, sample retrieval is performed by optimal path search in a similar fashion as presented by Maestre et al. (2009) after Schwarz (2004) and Aucouturier & Pachet (2006). A best sequence of N candidate indexes[4] γ^* is found by minimizing a total cost path C, composed of a symbolic cost C_S, a bowing cost C_b, and a sample continuity cost C_c as expressed in equations (5.1) and (5.2) ω_s, ω_b, ω_c respectively represent a corresponding weight that needs to be adjusted manually. The solution is found by using dynamic programming (Viterbi, 1967).

$$\begin{aligned} \gamma^* &= [\gamma_1^* \cdots \gamma_N^*] \\ &= \underset{\gamma}{\operatorname{argmin}}\, C(\gamma) \end{aligned} \tag{5.1}$$

$$C(\gamma) = \sum_{i=1}^{N} \big(\omega_s C_s(\gamma_i) + \omega_b C_b(\gamma_i)\big) + \sum_{i=1}^{N-1} \omega_c C_c(\gamma_i, \gamma_{i+1}) \tag{5.2}$$

Symbolic cost

The symbolic cost C_s is computed by comparing target note and candidate's sample duration, fundamental frequency, and fundamental frequency intervals to preceding and following notes. It is computed by a weighted sum of a duration cost C_d, a fundamental frequency cost C_f, and a fundamental frequency interval cost C_i. Whilst the former two are computed for each note, the latter intervenes in the sum whenever a note-to-note transition is involved.

- **Duration cost**

  The cost corresponding to duration is computed as the time-average of a time stretch transformation cost C_{ts}, computed from the time stretch factor F_{ts} values to be applied along the note sample in order to match the target duration. This is expressed in equation (5.3), where M corresponds to number of frames of the database sample. Time stretch is not to be applied linearly along the length of the note, but by means of a variable-shape function. The shape of such function is set so that it is avoided stretching the edge segments of the note.

$$C_d = \frac{1}{M} \sum_{m=1}^{M} C_{ts}^2(m) \tag{5.3}$$

[4] Each index corresponds to the selected sample from each candidate list.

In equation (5.4), a logarithmic function is used for computing the time-stretch transformation cost C_{ts} in order to make its value equal to one for time-stretch factors of two or point five. This decision is based on prior knowledge on the quality of the time stretch transformation algorithm used (Bonada, 2000), assuming near lossless transformations for stretching factors between point five and two.

$$C_{ts}(m) = \mid log_2(F_{ts}(m)) \mid \tag{5.4}$$

- **Fundamental frequency cost**

  The frequency transformation cost C_f is computed by means of a logarithmic function that depends on the relation of target and sample fundamental frequencies expressed in Hz, as it is expressed in equation (5.5). A transposition of one octave up or down would correspond to a cost equal to one. Again, the decision adapting the cost formula to arbitrary limits of transposition transformation (one octave) is based on prior knowledge of the quality of the pitch shifting technique used (Laroche, 2003) .

$$C_f = \left| log_2\left(\frac{f_{in}}{f_{out}}\right)\right| \tag{5.5}$$

- **Fundamental frequency interval cost**

  A fundamental frequency interval cost C_i is computed for each note-to-note transition (see Figure 5.7) as expressed in equation (5.6), where i_in corresponds to the target interval, $i_{R,left}$ corresponds to interval from the candidate sample at the right side of the transition to its predecessor sample in the database, and $i_{L,right}$ corresponds to interval from the candidate sample at the left side of the transition to its successor sample in the database. All intervals are expressed in *cents*.

$$C_i = \left| \frac{\mid i_{in} - i_{L,right} \mid + \mid i_{in} - i_{R,left} \mid}{i_{in}} \right| \tag{5.6}$$

Bowing cost

For the bowing contour cost C_b, it is computed the Mahalanobis $D_M(p_f, p_s)$ distance between contour parameters of the rendered bowing controls, contained in the vector p_f (see Section 4.4.2); and the contour parameters of the candidate sample (obtained during the analysis stage), contained in a vector p_s. Since samples used for populating each candidate list must match the note class to which belongs the corresponding note in the input score (see Section 3.4), p_f and p_s reside in the same space. The computation of $D_M(p_f, p_s)$ is based on the mixed covariance matrix Σ^* previously used for obtaining p_f (see Section 4.4.1).

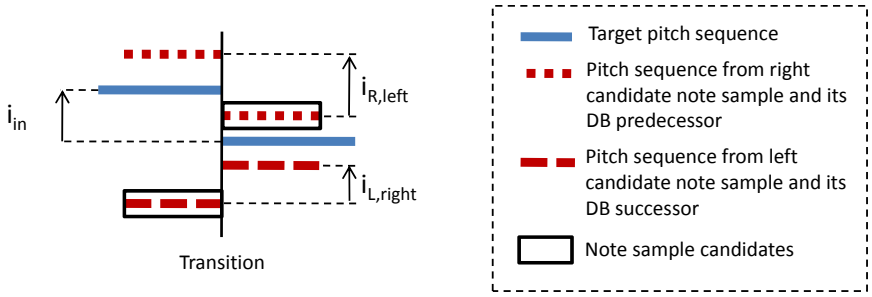


Figure 5.7: Illustration of the different pitch intervals taking part in the computation of the fundamental frequency interval cost C_i corresponding to each note-to-note transition.

$$C_b = D_M(p_f, p_s) = \sqrt{(p_f - p_s)^T \Sigma^{*-1} (p_f - p_s)} \tag{5.7}$$

Continuity cost

The continuity cost C_c between i-th and $(i+1)$-th notes is set as a penalty for encouraging the algorithm to retrieve samples that appear contiguously in the database.

Discussion

The different weights appearing in equation (5.2) define the importance given to each sub-cost with relation to the others. It is a difficult task to formally find an optimal compromise when choosing the weight values, as it strongly depends both on the nature and coverage of the database used, and on the quality of the different spectral-domain transformations to be applied to retrieved samples.

The continuity cost C_c is intended to avoid sample concatenations, and the symbolic cost C_s is directly related to the sound quality loss derived from time-stretch and transposition (classical transformations present in sample-based synthesis schemes). The bowing contour cost C_b is linked to the time-varying spectral envelope alteration applied for the timbre of the retrieved sample to match that of a prediction (indeed expressed as harmonic and residual spectral envelopes, see next section) obtained, frame-by-frame, from the synthetic bowing parameter signals.

Due to the difficulties arising from trying to qualitatively compare the sound quality losses caused by the different transformations, i.e. duration/pitch versus timbre, searching for a satisfactory set of weighting factors ends up becoming

a trial-and-error process that needs to be carried out manually. However, it is proved that the introduction of the bowing cost C_b enrichs the search process and provides better selections of samples in terms of similarity between bowing parameter contours that have been synthesized from an input score and those corresponding to retrieved samples.

Figures 5.8 and 5.9 depict each one an example excerpt showing sample selection results obtained by only attending to symbolic cost (left) and by including bowing cost (right). The first example comprises seven notes, with sample selections for notes no.1, no.3, no.4, and no.7 changed after introducing the bowing cost. By comparing syntetic contours (thick solid curves) to contours of retrieved samples (thick dashed curves), it is clearly perceived an improvement in shape similarity, especially for the case of bow velocity and bow force. In the second example, all four selections change when introducing the bowing cost, greatly improving similarity and continuity of contours, again especially noticeable for bow velocity and bow force.

5.3.2 Sample transformation

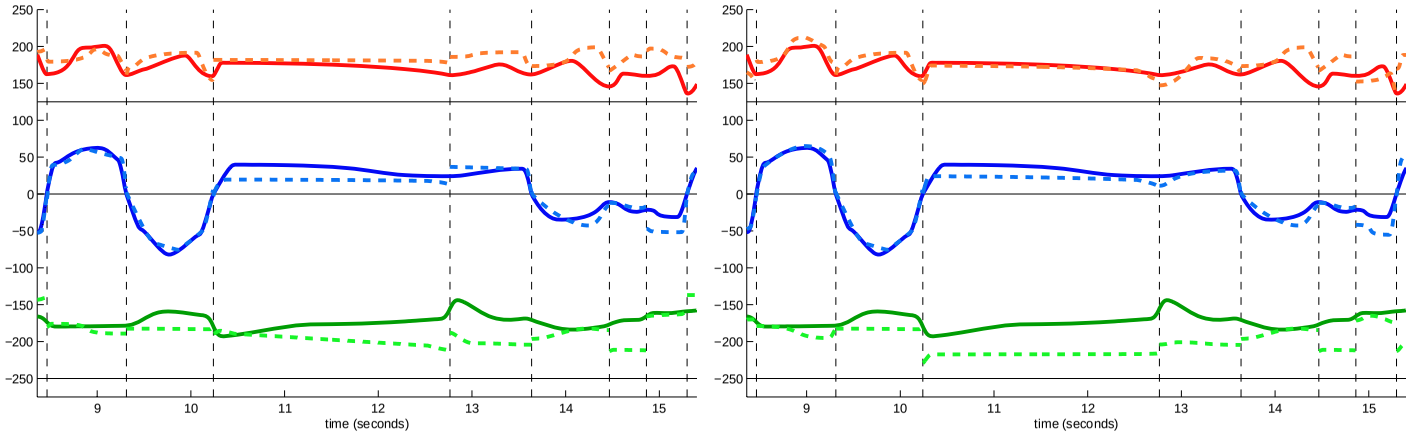
Once the best sequence of samples has been selected, transposition and time-stretch are applied in a first step so that target values coming from the input score are matched. Then, synthesized bowing parameter techniques are used for shaping the timbre evolution of samples by means of a gesture-based, non-linear timbre model obtained from real data.

Duration and tranposition

Fundamental frequency and duration mismatches between input score values and retrieved sample annotations are transformed in the spectral domain by means of an implementation of the phase-locked vocoder techniques described by Bonada (2000) and Laroche (2003). While transposition is applied uniformly to the whole sample, time-scaling is applied non-uniformly along the note so that its central part carries most of duration transformation. An overview of the application of these techniques for singing voice sample-based synthesis is given by Bonada & Serra (2007).

Timbre transformation based on instrumental control

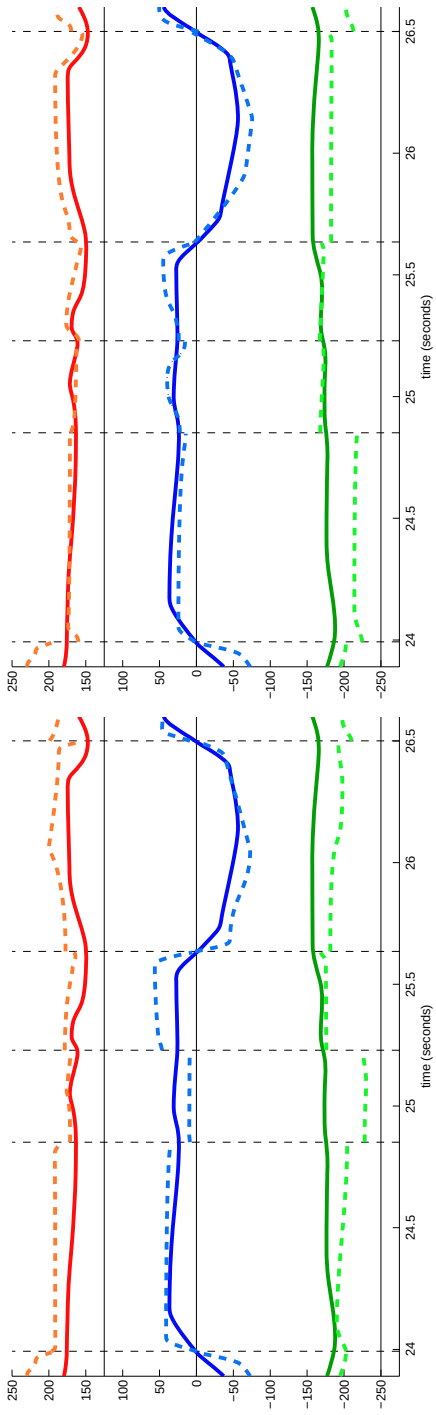
Retrieved samples are transformed in the spectral domain by independently alterating the envelopes of the harmonic and non-harmonic components of the sound at each frame. Such alteration is performed, by attending to synthetic bowing contours, by means of two timbre models able to independently predict, for a given set of bowing parameter values (e.g. bow velocity, bow force, etc.) and derivatives, the RMS energy level of a number of linearly spaced spectral bands modeling the amplitude of harmonic and non-harmonic componets of the spectrum. This is expressed in equations (5.8) and (5.9), where E_d^i and E_s^i



(a) Symbolic cost only

(b) Including bowing cost

Figure 5.8: Sample selection results (example 1). From top to bottom: bow force ($0.02N/unit$), bow velocity (cm/s), and bow-bridge distance ($0.04cm/unit$). Horizontal thin lines correspond to zero levels, solid thick curves correspond to rendered contours, dashed think curves represent the Bézier approximation of the acquired contours corresponding to retrieved samples, and thin dotted lines correspond to the actual bowing parameter signals of retrieved samples. Vertical dashed lines represent note onset/offset times.



(a) Symbolic cost only

(b) Including bowing cost

Figure 5.9: Sample selection results (example 2). From top to bottom: bow force ($0.02N/unit$), bow velocity (cm/s), and bow-bridge distance ($0.04cm/unit$). Horizontal thin lines correspond to zero levels, solid thick curves correspond to rendered contours, dashed think curves represent the Bézier approximation of the acquired contours corresponding to retrieved samples, and thin dotted lines correspond to the actual bowing parameter signals of retrieved samples. Vertical dashed lines represent note onset/offset times.

respectively correspond to the harmonic and non-harmonic energy of the i-th band, obtained by means of estimated non-linear functions f_d^i and f_d^i, and the non-harmonic energy of band.

$$E_d^i = f_d^i(p_b, v_b, F, d_{BB}, l_e, s, v_b', F', d_{BB}') \tag{5.8}$$

$$E_s^i = f_s^i(p_b, v_b, F, d_{BB}, l_e, s, v_b', F', d_{BB}') \tag{5.9}$$

Even though the timbre models are used here for performing sample transformations that are meaningful to instrumental control (indeed driven by synthetic contours of bowing gesture parameters which are obtained from an annotated input score), the details of their conception or construction are not addressed here. For further details, the reader is referred to previous publications (Pérez et al., 2007, 2010) and to the PhD work by Pérez (2009).

Prediction of spectral envelope parameters from continuous bowing control signals has been pursued in the past in the works by Schoner et al. (1999, 2000), where a predictive framework was built from real data. Streams of bow velocity, estimated bow force, finger position and bow-bridge distance were acquired from real violin playing, together with radiated violin sound. Acquired audio was analyzed in the spectral domain, having the partial amplitudes and frequencies at a given frame used as the target vector, and the bowing parameter instantaneous values as the input vector. Example input-target pairs were used to train a probabilistic model based on Gaussian mixtures, capable of providing predictions of partial ampitudes and frequencies. While mostly intented for real-time sound synthesis via sinusoidal modeling and not for sample transformation, the idea of explicitely relating spectral representations of sound to physical instrumental control parameters partly inspired the gesture-based timbre modeling framework applied here.

Two main aspects make the prediction framework used here to be different from Schoner's approach. The first one deals with the audio signal: the audio signal is acquired from a bridge pickup, thus avoiding the resonances and reverberation of the violin bridge. The second one is the way in which sound is modeled: both harmonic and noise parts of the sound signal are treated separately, and represented as independent of the fundamental frequency (spectral band positions are fixed). Next, a brief overview of the construction of the model and its application is given.

The data used for constructing the model correspond to the aligned audio-gesture data contained in the database used for pursuing gesture analysis and modeling (see Section 2.6), separated by strings. The audio signal captured by means of the bridge pickup is framewise analysed via Sinusoidal Modeing Synthesis (SMS)[5] as introduced by Serra (1989), yielding a parameterization of the harmonic component (partial amplitudes, frequencies and phases) and a separated residual part of each frame. From each of the obtained components

[5] http://mtg.upf.edu/technologies/sms

parts, it is computed the RMS energy of 40 overlapping bands linearly spaced along the frequency axis.

From the collected data, each pair i-th non-linear functions f_d^i and f_s^i in equations (5.8) and (5.9) is estimated by means of Artificial Neural Networks (ANNs), in a similar manner as Lindemann (2007) did for predicting harmonic amplitudes from low-varying pitch and energy curves. The data are separated into strings in order to accound for the different timbre characteristics of each of them. This yields a total number of $40\text{bands} \times 2\text{components} \times 4\text{strings} = 320$ ANNs to be trained. Given the large amount of recorded frames contained in the database, enough data are avaible for carrying out the training, achieving high correlation coefficients and low predition errors. Training of the ANNs (Witten & Frank, 2005) is carried out by using the free software *WEKA* [6].

Figure 5.10 displays an overview of the timbre transformation, applied to each frame of the retrieved sample (after applying time-stretch and transposition) by attending to the synthetic instrumental control (bowing) parameters generated from the input score. The harmonic part (partial amplitudes) of the frame is set so that it matches the harmonic spectral envelope prediction provided by the artificial neural networks previously trained with real data (upper part of the Figure). Analogously, the original stochastic component of the audio frame is shaped by its corresponding predicted envelope (lower part of the Figure). For both components, the spectral envelope shaping is achieved by dividing the spectral content into bands, extracting the original spectral envelope by spline interpolation of the obtained band energy values, and computing a differential spectral envelope from the spectral envelope obtained and the predicted band energy values also by spline interpolation. This differential spectral envelope is applied to the original component.

5.3.3 Results

Like it happened for the case of physical models, the results attained by explicitly introducing instrumental control into sample-based concatenative sound synthesis architectures also demonstrated the validity of the instrumental gesture modeling framework. This corroborates the ideas supporting this dissertation.

Sample-based gains in control flexibility, leading to a substantial improvement of timbre continuity while helping in overcoming database coverage limitations. However, other problems inherent to sample-based synthesis still remain, as it is the case of sample selection errors derived from weight calibration difficulties, or quality limitations of sample concatenation, pitch-shifting, and time-stretch processes. Moreover, certain defficiencies detected in the timbre modeling component when predicting spectral envelopes in off-string bowing conditions or during transients lowered the performance of the system in particular situtations.

[6] http://www.cs.waikato.ac.nz/ml/weka

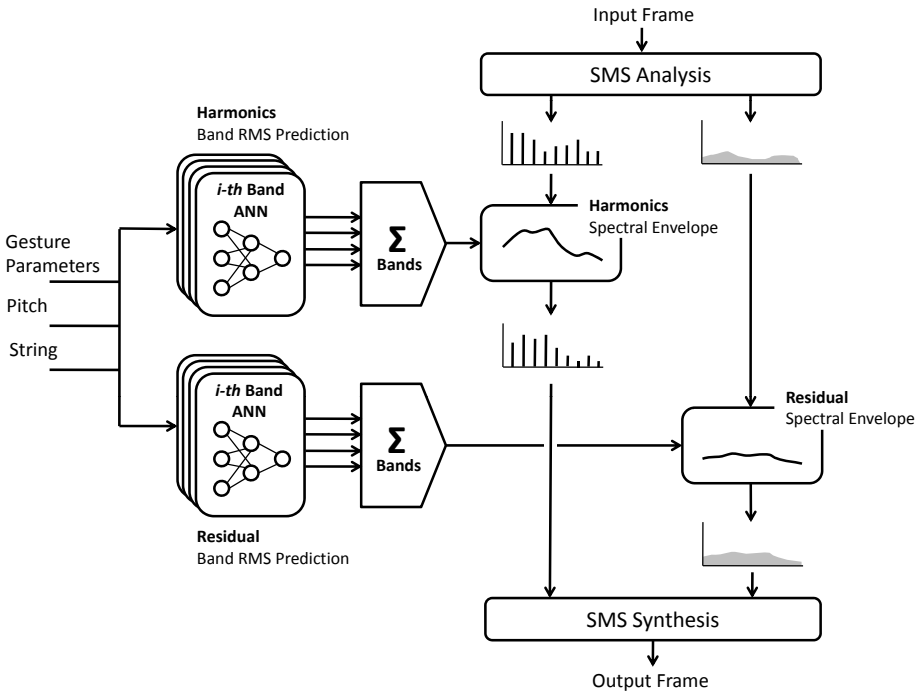


Figure 5.10: Overview of the gesture-based timbre transformation.

As for physical models, subjective tests were carried out with the aim of demonstrating the advantageous effects of informing sample-based synthesis with a rich, meaningful representation of instrumental control. Subjects listened to synthesis results obtained by means of the proposed architecture and by means of a classical architecture in which no synthetic bowing controls were used either during sample selection nor during sample transformation. In general, subjects agreed on an overal improvement of the timbre continuity, perceived naturalness and quality, except for those cases in which the timbre modeling limitations led to worst results than in the classical architecture. Results and comparative examples can be found online[7].

5.4 Summary

As reported in this chapter, the sucessful application of synthetic bowing parameter signals to violin sound generation proved to be an optimal evaluation

[7] http://www.iua.upf.edu/~emaestre/phd/sounds/

test-bed for demonstrating the validity of the instrumental gesture modeling framework introduced in this dissertation. At the same time, this chapter served to indicate future directions in instrumental sound synthesis.

In terms of physical model-based sound synthesis, obtained sounds delivered impressive realism and naturalness by only using a very simplified, non-calibrated digital waveguide model. The fact that listeners were unable to distinuish sounds obtained through recorded controls from those obtained by means of synthetic controls is a clear evidence of the efficacy provided by the bowing control modeling framework developed in this work. The potential of physical modeling sound synthesis, when combined with appropriately generated input control signals, has been confirmed by the results attained. Going for more complex physical models while ensuring reliable calibration is only to corroborate real possibilities of the most powerful instrumental sound synthesis technique in terms of control flexibility.

The second evaluation framework, consisting in explicitly introducing instrumental control into sample-based synthesis, also resulted in a great success. The subjective evaluation made clear the benefits brought by instrumental gesture modeling for overcoming one of the major drawbacks of sample-based synthesis: database coverage. Despite the constrains imposed by the basic concepts around sample-based concatenative synthesis (limitations of sample transformation and concatenation), achieved improvements in timbre continuity validated the deep structural changes proposed for the architecture. Fine adjustments of sample selection weighting and improvements on the performance of the timbre model can only improve results in less preliminary developments.

Chapter 6

Conclusion

Aligned to the challenging research pursuit that represents the computational exploration of music performance, the main objective of this dissertation was to introduce a computational modeling framework suitable for the analysis and synthesis of instrumental gestures in excitation-contnuous musical instruments.

Regarded among the most complex to play, the violin provides the performer with a broad range of expressive resources and playing techniques. Moreover, it offers a relative ease for capturing instrumentals gesture parameters (in particular bowing gestures) in low-intrusiveness conditions. These two reasons support the decision of choosing the violin as an optimal case of study. The focus of the research was put into bowing techniques, and an instrumental gesture modeling framework was devised so that it was ensured a strong connection with the score.

This work supposes the first comprehensive data-driven study for computationally addressing the problem of modeling the performer's role of transforming the discrete-nature information appearing in an annotated musical score into the continuous-nature physical actions driving sound production in bowed string practice. The dissertation introduced and validated a systematic approach to the acquisition, representation, modeling, and synthesis of bowing patterns in violin classical performance.

6.1 Achievements

Aiming at contributing with a general methodology that can be extended to other case studies, the accomplishment of the thesis objectives has involved the successful attainment of diverse research challenges, as Section 1.5 intially stated. The thesis achievements and contributions are summarized and criticized in the next sections.

6.1.1 Acquisition of bowing control parameter signals

The first achivement of this disseration is the proposition of methods for real-time acquisition of violin instrumental gesture parameters. In terms of motion-related parameters, the main contribution resides on tracking the positions of both the ends of the strings and the ends of the hair ribbon, and on constructing a thorough calibration methodolgy for (1) estimating bow position, bow velocity, bow-bridge distance, bow tilt, bow inclination, and (2) automatically detecting the string being played. The acquisition framework outperforms previous approaches in terms of intrusiveness (only two small-size, wired sensors are present, one attached to the wood of the bow and the other attached to the violin back plate), portability (it allows the musician to use her own instrument, and it does not require a complex setup), and robustness (it provides accurate estimations in a straightforward manner and does not need of complex post-processing).

For the case of bow pressing force, two main improvements have been introduced to previously existing techniques for measuring hair ribbon deflection using strain gages. First, two strain gages are attached to opposite sides of the bending plate, providing double sensibility and allowing for temperature compensation. Second, an incremental calibration procedure has been devised in order to compensate bow tension and temperature changes happening during long recording sessions.

Moreover, a score-performance alignment agorithm has been applied for the rapid construction of a large database of segmented audio and gesture data. The algorithm, based on dynamic programming techniques, makes use of both acquired instrumental gesture parameters and audio descriptors for automatically providing note onset/offset times of a considerable amount of violin performance recordings in which both audio and gesture data are acquired synchronously.

Critique

The main drawback of the acquisition methodology resides on the fact that sensor connections are wired. Despite the fact that the weight of the cables did not represent a consequential issue, a musician requires in general some time to get adapted to such playing conditions. Even though this inconvenient did not bring significant difficulties, an improvement on portability could be achieved if, yet allowing the musician to play her own instrument, the use of wired connections was avoided while keeping the low complexity of the measurement setup. A similar problem can be drawn from the device installed in the bow for measuring hair ribbon deflection. Although not implemented in this work, the estimation of bow force by carrying out appropriate post-processing of the data coming from the trackers would eliminate part of the potential playability problems brought by augmenting the bow with a metallic plate that indeed reduces the effective bow length by approximately two centimeters.

Certain segmentation inconsistencies arose for particular recordings in which off-string bowing conditions were common, demonstrating that some improvements could be achieved in more developed versions of the score-alignment algorithm. An objective evaluation of the algorithm was not carried out, but the obtained results demonstrated that the manual tuning of costs' weighting factors, combined with a small number of manual corrections, were satisfactory enough for the application context.

6.1.2 Representation of bowing control parameter signals

A second major contribution of this work is the design of a contour representation scheme that is suitable for quantitatively supporting the definition of a bowing technique vocabulary in terms of temporal patterns of acquired bowing parameter envelopes. Contours are modeled at note level by means of sequences of cubic Bézier curves, which represent a powerful tool given their flexibility and robustness. The arrangement and main characteristics of segments are adapted to each bowing technique, and determined by a set of rules defined from observation of contours. The proposed contour representation framework is (1) flexible enough for providing contour representation fidelity, and (2) robust enough for ensuring contour parameterization consistency accross different executions of similar bowing techniques. The extraction of contour parameterizations, i.e. estimation of the use of the curve vocabulary, is carried out by means of a segmentation and fitting algorithm that allows for the automatic characterization and analysis of large amounts of data, enabling further statistical observation.

Critique

Even though the envelope representation framework proved to be reliable and effective, it suffers from two main inconvenients. The first one deals with the methodology followed for determining the Bézier segment sequences' arrangement (i.e., how the grammar is defined): in spite of the excellent results provided by the segmentation and fitting algorithm, the definition of grammar entries requires human observation. An improvement could be brought by devising an optimization process in which grammar rules were defined automatically.

A second problem has to do with the space of curve parametrizations. The fact that parameter contours of different bowing techniques or note classes are coded with curve segment sequences of different lengths makes difficult to find a common space in which parametrizations of different bowing patterns can be analysed. This impedes, for instance, an otherwise potentially easy study of morphing between bowing techniques.

Moreover, another possible criticism could be thrown by claiming that the envelope representation schemes, which have been devised by pure observation of contour morphology, do not respond to any physical principle that supports the actual mechanics of performer-instrument interaction.

6.1.3 Modeling of bowing parameters in violin performance

One of the most significant achievements of the thesis is the proposition of an analysis/synthesis framework for violin bowing control parameter envelopes as related to the performed score. The introduced methodology is based on statistical modeling of bowing parameter quantitative descriptions (i.e., Bézier curve parameters) obtained from an annotated performance database. Each note in the database is represented in two different spaces: a contour parameter space and a performance context space. While the former is defined by the obtained Bézier curve parameter values, the latter is constructed from contextual note characteristics not dealing with the bowing parameter contours per se (e.g., note duration, finger position, etc.). A statistical model based on multivariate gaussian mixtures is constructed so that both spaces are mapped, and a probability distribution of curve parameter values can be synthesized from an input vector of context parameters. From the obtained probability distribution, random vectors of curve parameters can be generated so that rendered curves mirror the shape of orginally acquired bowing parameter contours.

On top of the note-level contour modeling framework, a bow planning algorithm has been introduced for integrating the envelope synthesis capabilities and successfully constitute a system able to generate automatic performances from an input score. The algorithm, again by means of statistical modeling of acquired data, explicitely accounts for the implications brough by one of the most important constrains in violin bowing: the finite length of the bow. From an input score annotated with bow techniques and direction changes, articulations, and dynamics, the algorithm proposes a sequence of bow starting and ending positions, and accordingly renders and coherently concatenates contours of bow velocity, bow pressing force and bow-bridge distance. Both the bow planning results and the synthetic bowing controls significanlty resembled original recordings.

Critique

Certain limitations of the modeling framework implementation were caused by database coverage problems. An eventual unreliability of the normal disitributions as a basic tool for statistically describing the behavior of contour parameters was often originated from an unbalanced coverage of the performance context parameters by the database, which in turn resulted in poorly populated clusters. Also related to this, a major simplification of the model was made my not separating data into different strings. If in possession of a larger amount of recordings, the space of performance context parameters could be expanded by including the bow displacement, or incorporating interesting dimensions to the description of each note's circumstances, like for instance the preceding and following bowing techniques, or higher level musicological features.

As previously mentioned, some characteristics inherent to the envelope char-

acterization scheme bring potential limitations regarding the space of curve parametrizations. Since the statistical modeling framework proposed in this work is based on relating the space of score-based contextual parameters and the space of curve parameters defining the envelopes of the different bowing controls, the fact that different bowing techniques are represented in different spaces somehow cuts down the potential of the analysis/synthesis framework. Finding a common representation space would be of great interest for incorporating new capabilities.

Although succesfully managed during bowing control rendering, the contour concatenation of adjacent notes does not consider explicit continuity constraints. This minor issue, while not conditioning the performance of the current implementation, could be tackled at the rendering stage by including a set of concatenation constrains in charge of guaranteeing that the values of the involved Bézier attractors ensure first-order continutity.

Regarding the bow planning algorithm, no exhaustive evaluation was carried out. However, obtained results always showed to be consistent with original recordings, mainly due to the fact that the algorithm is constructed from statistical descriptions of the performer exections. Since the modeling framework does not incorporate any physical principle dealing with the actual mechanics of performer-instrument interaction, it results difficult to formally assess its validity from a physical point of view.

Finally, it is important to remark that no objective evaluation was carried out with the aim of quantifying the performance of the modeling framework as a whole. In the implementation, a significant number of model parameter configurations (e.g. grammar definitions, clustering parameters, etc.) were tested, searching for a best setting by both observing rendering results (as comparing to acquired data) and by listening to synthesized sound. As it is usually done in speech processing frameworks, a proper assessment of the configuration of model parameters could be implemented, like for instance by means of some kind of perceptual measure like the spectral distortion between sound obtained from original bowing parameters (or even recorded sound) and that generated from synthetic bowing controls. This interesting option would nevertheless bring additional difficulties derived from known limitations of the sound synthesis techniques to be utilized.

6.1.4 Automatic performance applied to sound generation

The sucessful application of synthetic bowing parameter signals to violin sound generation represents by itself a major contribution, although it also served as an optimal test-bed for demonstrating the validity of the instrumental gesture modeling framework introduced in this work. Instrumental control was explicitely introduced into the two most extended sound synthesis techniques: physical modeling and sample-based synthesis. While for the former it resulted more evident, embedding gesture information into sample-based synthesis implied to contribute with two major modifications of the classical synthesizer

structure. The first was the enrichment of the database samples. Samples became a combination of audio and gesture signals, having both the bowing control envelopes and their parametrization to play an important role during the sample selection process. The second was to meaningfully enhance sample treatment by means of including a gesture-controlled timbre transformation component.

In terms of physical modeling, obtained sounds delivered impressive realism and naturalness by only using a very simplified, non-calibrated digital waveguide model. The unability of listeners when trying to distinuish sounds obtained through recorded controls from those obtained by means of synthetic controls represents a clear demonstration of the efficacy provided by the bowing control modeling framework developed in this work. When combined with appropriately generated input control signals, the potential of physical modeling synthesis as a tool for obtaining realistic sounds is confirmed by the attained results.

The case of sample-based synthesis also represented a great achievement. The subjective evaluation made clear the benefits brought by instrumental gesture modeling for overcoming two of the major drawbacks of sample-based synthesis: database coverage and control flexibility. Despite the constrains imposed by the basic concepts around sample-based concatenative synthesis (e.g., limitations of sample transformation and concatenation), achieved improvements in timbre continuity validated the structural and functional changes proposed for the architecture.

Critique

The application to sound synthesis was intially devised as an evaluation test-bed for validating the instrumental gesture models introduced in this dissertation. Even though it resulted in a success, the fact that the assessment only relayed on perceptual tests (i.e., listenting to synthetic sounds) could be seen as a weakness of the evaluation itself. As already commented, a quantitative evaluation technique could be devised by means of measuring spectral distortion between sound obtained from original bowing parameters (or even recorded sound) and that generated from synthetic bowing controls. Such technique, apart from bringing the opportunity of pursuing an objective exploration of the model parameters, might reveal strengths and limitations of the sound synthesis techniques.

Regarding sound generation per se, some criticisms could be drawn from the two implementations. For the physical modeling application, the sound quality was inherently limited both by the simplicity of the model and by the lack of an appropriate calibration. Likewise, an objectively determined sample-search weight calibration combined with an improved performance of the timbre model in both transient regions and low-force condtions are only to boost the quality of synthetic sound.

6.2 Future directions

Yet a success, the results attained in this work represent only a preliminary demonstration of the possibilities brough by computationally modeling excitation-continuous instrumental gestures in music performance. Thus, contributed capabilities constitute a starting point for a number of improvements and research directions. This section outlines future lines of study, both in the shape of tackling limitations of those exposed above, as well as regarding application possibilities.

6.2.1 Instrumental gesture modeling

Regarding the instrumental gesture modeling framework, an obvious forthcoming step would be the incorporation of left-hand controls in violin practice. Beyond the acquisition of the finger position (which can be easily managed by attending to the string being played and to the fundamental frequency of the sound), the implementation of a technique (either direct or indirect) able to provide a reliable estimation of the finger clamping force is to provide insightful, complementary information. Regarding this topic, the methodology recently introduced by Kinoshita & Obata (2009) appears as an interesting approach.

Once the validity of the modeling framework has been proved for a reduced but representative set of bowing techniques, the construction of an extended, complete database will provide means for building a more comprehensive model. The inclusion of a broader range of performance resources (e.g., adding more bowing techniques, crescendo/decrescendo, etc.) while ensuring a better database population in terms of note context paramters (recall that, for instance, unevenly distributed note durations forced the introduction of a hierarchical clustering scheme) is going to bring several advantages. A first important benefit is to be able to overcome the problems brought by poorly populated clusters. Indeed, the study of sample clustering might also become more central to the modeling process. Also, the possibility of separating data into strings, as well as the introduction of the bow displacement as one of the performance context parameters, will contribute positively to the improvement of the model. In general, a smart increment of available data will help to further develop the techniques introduced.

The availability of a database of aligned audio and gesture parameter signals looks as an optimal ground-truth for pursuing indirect acquisition of bowing control parameters from audio analysis. This, possibly to be carried out by using known characteristics of sound together with machine learning techniques, would interestingly contribute to ease the construction of further databases by avoiding the utilization of sensors.

An important characteristic of the contour representation framework is the fact that the grammar rules defining curve segment sequences were manually defined after observation of envelopes. An automatic procedure in charge of looking for the most suitable grammar entries would represent an inmense

improvement. Related to that, the design of an objective evaluation scheme (as pointed out in the previous section) appears as one of the mandatory next steps, and will definitely enable the automatic optimization of the model parameters, including the definition of grammar entries. Once in possession of an extended database, the idea is to choose physical models as the synthesis technique, and to iteratively search for the model parameters (e.g., the number of clusters) and grammar entries that minimize a total error between the spectral envelope of sounds generated through recorded controls and through recorded controls. This iterative procedure should ensure that the error is computed over a representative set of recordings.

Not to leave out as a future path is the study of alternative contour representations that allow the parametrizations of completely different bowing techniques to be expressed in a common space, an therefore boosting the flexibility of the models. Continuous morphing between bowing techniques, dynamics, and articulations are to be studied. The interesting idea of introducing physical principles (i.e. mechanics of the gestures) into instrumental control representation and modeling my also contribute to flexibility and generality (Rasamimanana & Bevilacqua, 2008; Bouënard et al., 2009).

6.2.2 Sound synthesis

In general terms, the instrumental control modeling framework presented in this disseration shows future directions in musical instrument sound synthesis. As appropriate techniques for measuring control parameters become available, emulating playing habits in human performance can be made feasible, hence taking instrumental sound synthesis to a next stage. Current sample-based synthesis techniques may already start benefiting from the ideas on performer-instrument decoupling introduced here: the combination of automatic control rendering with control-based spectral-domain sample transformation may provide them with a degree of controllability close to that of physical models. For the case of physical-models, next-generation synthesizers able to represent the most complex sound production mechanisms will nevertheless require a control model in order to exploit the great flexibility derived from their nature.

The use of more sophisticated physical models (Demoucron, 2008; Serafin, 2004) will shortly be pursued. Moreover, an implementation of an automatic procedure for calibrating the physical model (see Section 5.2.1 and Appendix C) is currenlty under development. For the sample-based synthesizer, research should be dedicated to (1) a better adjustment of sample selection weights, (2) the incorporation of a non-linear time-stretch technique that account for instrumental control information, and (3) the improvement of the timbre model performance in non-sustained conditions. An spectral modeling alternative to synthesize sound is, as presented by Pérez (2009) after Schoner et al. (1999, 2000), the direct synthesis of harmonic and residual parts, driven by rendered controls. Such approach would avoid using sound samples, so many related problems would therefore be eluded.

The combination of a gesture sample database (annotated segments of instrumental control parameter signals) with an appropriate distance measure may lead to devising and implementing a possibly interesting approach to sound synthesis: Gesture-Sampling Sound Synthesis (GSSS). Out of an input score, a gesture prediction engine (maybe similar to what has been presented in this dissertation) would provide a set of target gesture features to be used for selecting and concatenating segments of an annotated performance database including instrumental control parameters. Then, for synthesizing sound, the resulting control signals could be applied either to physical modeling or to spectral modeling techniques. Given the availability of data, this option also constitutes one of the inminent research lines.

6.2.3 Application possibilites

As it has been demonstrated by the results attained, the use of synthetic instrumental controls for driving off-line sound synthesis represents the most direct application of the proposed framework, bringing an intersting opportunity for virtual orchestration. The systematic approach presented here for modeling instrumental control (including the acquisition techniques) can be easily applied to the rest of instruments of the bowed-string family. Going for other excitation-continuous musical instruments (e.g., wind instruments) would imply, however, a number of challenges related to the acquisition of gesture parameters. In those cases, indirect acquisition techniques may contribute to maintain the necessary low intrusiveness needed for the performer to play naturally.

In this work, the scores performed by the musician were annotated, so she was not allowed to freely choose bowing tecniques, bow directions, articulations, or dynamics or duration variations. In order to approach the challenging problem of studying expressive performance, the constructed models could be used for automatically recognizing which expressive resources are chosen by the performer when she is given a plain score with no annotations and, at the same time, for providing a coherent parametrization of those resources. Thus, modeling a performance style of one or different performers could be approached. Related to this, and assuming that acquisition techniques can be made more affordable, the use of performance models in pedagogical scenarios also appears as a potential application.

Even though the contour representation and modeling framework was initially devised to the study of instrumental gesture parameters, its application to model the temporal signature of perceptual parameters (e.g., pitch or energy) in music performance (Danneberg & Derenyi, 1998) or even speech (Fujisaki & Ohno, 1996; Escudero et al., 2002) could be undertaken and compared to exisiting solutions.

6.3 Closing

This dissertation has introduced and validated a novel methodology for modeling instrumental gestures in excitation-continuous musical instrument performance practice. The promising results obtained for the case of violin bowing constitute an evidence of the many possibities that offers to computationally address the challenging problem of modeling music performance. Hopefully, forthcoming studies will contribute with new capabilities that significantly extend those introduced in this work. Then, the existence of Mike's computer-assisted music creation tool will appear as less remote.

Bibliography

Askenfelt, A. (1986). Measurement of bow motion and bow force in violin playing. *Journal of the Acoustical Society of America*, 80(4):1007–1015.

Askenfelt, A. (1989). Measurement of the bowing parameters in violin playing. II. bow-bridge distance, dynamic range, and limits of bow force. *Journal of the Acoustical Society of America*, 86(2):503–516.

Aucouturier, J. & Pachet, F. (2006). Jamming with plunderphonics: Interactive concatenative synthesis of music. *Journal of New Music Research*, 35(1):35–50.

Battey, B. (2004). Bézier spline modeling of pitch-continuous melodic expression and ornamentation. *Computer Music Journal*, 28(4):25–39.

Bernstein, A. D. & Cooper, E. D. (1976). The piecewiselinear technique of electronic music synthesis. *Journal of the Audio Engineering Society*, 24(6):446–545.

Bonada, J. (2000). Automatic technique in frequency domain for near-lossless time-scale modification of audio. In *Proceedings of the 2000 International Computer Music Conference*. Berlin.

Bonada, J. & Serra, X. (2007). Synthesis of the singing voice by performance sampling and spectral models. *IEEE Signal Processing Magazine*, 24(2):67–78.

Bouënard, A., Wanderley, M. M., & Gibet, S. (2009). Advantages and limitations of simulating percussion gestures for sound synthesis. In *Proceedings of the International Computer Music Conference (ICMC09)*. Montréal.

Bourke, P. (2000). *Computer Graphics (Lecture notes)*. Centre for Astrophysics and Supercomputing, Swinburne University of Technology, Melbourne.

Cadoz, C. (1988). Instrumental gesture and musical composition. In *Proceedings of the 1988 International Conputer Music Conference*, pages 1–12. köln.

Cadoz, C. & Ramstein, C. (1990). Capture, representation and composition of the instrumental gesture. In *Proceedings of the 1990 International Computer Music Conference*, pages 53–56. Glasgow.

Chafe, C. (1988). Simulating performance on a bowed instrument. *CCRMA Tech. Rep. STAN-M48, Stanford University*.

Cook, P. & Trueman, D. (1998). A database of measured musical instrument body radiation impulse responses, and computer applications for exploring and utilizing the measured filter functions. In *Proceedings of the 1998 International Symposium on Musical Acoustics*. Leavenworth.

Cremer, L. (1984). *The physics of the violin*. MIT Press.

Cristianini, N. & Shawe-Taylor, J. (2003). *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge University Press, New York.

Danneberg, R. & Derenyi, I. (1998). Combining instrument and performance models for high quality music synthesis. *Journal of New Music Research*, 27(3):211–238.

de Cheveigné, A. & Kawahara, H. (2002). YIN, a fundamental frequency estimator for speech and music. *Journal of the Acoustical Society of America*, 111(4):1917–1930.

Demoucron, M. (2008). *On the control of virtual violins: Physical modelling and control of bowed string instruments*. Phd Thesis, Université Pierre et Marie Curie (Paris 6) and the Royal Institute of Technology (KTH, Stockholm).

Demoucron, M., Askenfelt, A., & Caussé, R. (2009). Measuring bow force in bowed string performance: Theory and implementation of a bow force sensor. *Acta Acustica united with Acustica*, 95(4):718–732.

Demoucron, M., Askenfelt, A., & Caussé, R. E. (2008). Observations on bow changes in violin performance. *The Journal of the Acoustical Society of America*, 123(5):3123–3123.

Demoucron, M. & Caussé, R. (2007). Sound synthesis of bowed string instruments using a gesture based control of a physical model. In *Proceedings of the 2007 International Symposium on Musical Acoustics*. Barcelona.

Dempster, A., Laird, N., & Rubin, D. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society*, 39:1–38.

Egozy, E. B. (1995). *Deriving Musical Control Features from a Real-Time Timbre Analysis of the Clarinet*. Master Thesis, Massachussetts Institute of Technology.

Escudero, D., Cardeñoso, V., & Bonafonte, A. (2002). Corpus-based extraction of quantitative prosodic parameters of stress groups in spanish. In *Proceedings of the 2002 International Conference on Acoustics, Speech and Signal Processing*. Orlando.

Farina, A. (2007). Advancements in impulse response measurements by sine sweeps. In *Preprints of the Audio Engineering Society 122th Convention*. Vienna.

Farina, A., Langhoff, A., & Tronchin, L. (1995). Realisation of virtual musical instruments: measurements of the impulse response of violins using mls technique. In *Proceedings of 2nd International Conference on Acoustics and Musical Research (CIARM)*. Ferrara.

Fischer, S. (1997). Basics: 300 exercises and practice routines for the violin. *Edition Peters, London*.

Flanagan, J. L. & Golden, R. M. (1966). Phase vocoder. *Bell System Technical Journal*, 45:1493–1509.

Fletcher, N. H. & Rossing, T. D. (1998). *The Physics of Musical Instruments, 2nd Edition*. Springer-Verlag.

Fujisaki, H. & Ohno, S. (1996). Prosodic parameterization of spoken japanese based on a model of the generation process of f0 contours. In *Proceedeings of the 1996 International Conference on Spoken Language Processing*.

Gabrielsson, A. (1999). *The performance of Music*. Academic Press.

Galamian, I. (1999). *Principles of Violin Playing and Teaching, 3rd edition*. Shar Products Co.

Garvey, B. & Berman, J. (1968). *Dictionary of Bowing Terms for Stringed Instruments*. American String Teachers Association.

Goudeseune, C. M. A. (2001). *Composing with parameters for synthetic instruments*. Phd Thesis, University of Illinois at Urbana-Champaign.

Guaus, E., Blaauw, M., Bonada, J., Maestre, E., & Pérez (2009). A calibration method for accurately measuring bow force in real violin performance. In *Proceedings of the 2009 International Computer Music Conference*. Montréal.

Guaus, E., Bonada, J., Pérez, A., Maestre, E., & Blaauw, M. (2007). Measuring the bow pressing force in a real violin performance. In *Proceedings of the 2007 International Symposium in Musical Acoustics*. Barcelona.

Guettler, K. & Askenfelt, A. (1998). On the kinematics of spiccato and ricochet bowing. *Catgut Acoustical Society Journal*, 3(6):9-15.

Guettler, K., Schoonderwaldt, E., & Askenfelt, A. (2003). Bow speed or bowing position-which one influences the spectrum the most? In *Proceedings of 2003 Stockholm Music Acoustics Conference*, pages 67–70. Stockholm.

Helmholtz, H. v. (1862). *Lehre von den Tonempfindungen*. Braunschweig, Vieweg.

Hodgson, P. (1958). *Motion study and violin bowing*. American String Teachers Association.

Jaffe, D. & Smith, J. (1995). Performance expression in commuted waveguide synthesis of bowed strings. In *Proceedings of the 1995 International Computer Music Conference*. Alberta.

Jaun, A. (2003). *Numerical methods (Lecture notes)*. Royal Institute of Technology, Stockholm.

Karjalainen, M. & Smith, J. (1996). Body modeling techniques for string instrument synthesis. In *Proceedings of the 1996 International Computer Music Conference*, pages 232–239. Hong Kong.

Karjalainen, M., Välimäki, V., Penttinen, H., & Saastamoinen, H. (2000). Dsp equalization of electret film pickup for the acoustic guitar. *Journal of the Audio Engineering Society*, 48(12):1183–1193.

Kendon, A. (1990). *Conducting Interaction: Patterns of behavior in focused encounters*. Cambridge University Press.

Kinoshita, H. & Obata, S. (2009). Left hand finger force in violin playing: Tempo, loudness, and finger differences. *Journal of the Acoustical Society of America*, 126(1):388–395.

Laroche, J. (2003). Frequency-domain techniques for high-quality voice modifications. In *Proceedings of the 2003 International Conference on Digital Audio Effects*. London.

Lindemann, E. (2007). Music synthesis with reconstructive phrase modeling. *IEEE Signal Processing Magazine*, 24(2):80–91.

MacQueen, J. B. (1967). Some methods of classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297.

Maestre, E. (2006). Coding instrumental gestures: towards automatic characterization of instrumental gestures in excitation-continuous musical instruments. *DEA Doctoral pre-Thesis work, Universitat Pompeu Fabra*.

Maestre, E. (2009). Data-driven statistical modeling of violin bowing gesture parameter contours. In *Proceedings of the 2009 International Computer Music Conference*. Montréal.

Maestre, E., Blaauw, M., Bonada, J., Guaus, E., & Pérez, A. (2010). Modeling bowing control applied to violin sound synthesis. *IEEE Transactions on Audio, Speech, and Language Processing (In Press)*.

Maestre, E., Bonada, J., Blaauw, M., Guaus, E., & Pérez, A. (2007). Acquisition of violin instrumental gestures using a commercial emf device. In *Proceedings of the 2007 International Computer Music Conference*, volume 1, pages 386–393. Copenhagen.

Maestre, E., Bonada, J., & Mayor, O. (2006). Modeling voice articulation gestures in singing voice performance. In *Preprints of the Audio Engeenering Society 121st Convention*. San Francisco.

Maestre, E. & Gómez, E. (2005). Automatic characterization of dynamics and articulation of expressive monophonic recordings. In *Preprints of the Audio Engeenering Society 118th Convention*. Barcelona.

Maestre, E. & Ramírez, R. (2010). An approach to predicting bowing control parameter contours in violin performance. *Intelligent Data Analysis (In Press)*.

Maestre, E., Ramírez, R., Kersten, S., & Serra, X. (2009). Expressive concatenative synthesis by reusing samples from real performance recordings. *Computer Music Journal (in press)*.

Mathews, M. (1989). The radio drum as a synthesizer controller. *Proceedings of the 1989 International Computer Music Conference*, pages 42–45.

McAulay, R. J. & Quatieri, T. F. (1986). Speech analysis/synthesis based on a sinusoidal representation. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 34(4):744–754.

McIntyre, M. E., Schumacher, R., & Woodhouse, J. (1983). On the oscillations of musical instruments. *Journal of the Acoustical Society of America*, 75(5):1325-1345.

McIntyre, M. E. & Woodhouse, J. (1979). On the fundamentals of bowed string dynamics. *Acustica*, 43(2):93-108.

McNeill, D. & Levy, E. (1982). *Conceptual Representations in Language Activity and Gesture*. John Wiley and Sons Ltd.

Miranda, E. R. & Wanderley, M. M. (2006). *New Digital Musical Instruments: Control and Interaction beyond the Keyboard*. A-R Editions.

Nespoulous, J., Perron, P., & Lecours, A. R. (1986). *Biological Foundations of Gestures: Motor and Semiotic Aspects*. Lawrence Erlbaum Associates.

Orio, N. (1999). The timbre space of the classical guitar and its relationship with the plucking techniques. In *Proceedings of the 1999 International Computer Music Conference*. Beijing.

Paradiso, J. A. & Gershenfeld, N. A. (1997). Musical applications of electric field sensing. *Computer Music Journal*, 21(2):69–89.

Penttinen, H. & Välimäki, V. (2004). A time-domain approach to estimating the plucking point of guitar tones obtained with an under-saddle pickup. *Applied Acoustics*, 65(12):1207–1220.

Pérez, A. (2009). *Enhancing Spectral Synthesis Techniques with Performance Gestures using the Violin as a Case Study*. Phd Thesis, Universitat Pompeu Fabra, Barcelona.

Pérez, A., Blaauw, M., Bonada, J., Guaus, E., & Maestre, E. (2010). Violin timbre model driven by performance controls. *IEEE Transactions on Audio, Speech, and Language Processing (In Press)*.

Pérez, A., Bonada, J., Maestre, E., Guaus, E., & Blaauw, M. (2007). Combining performance actions with spectral models for violin sound transformation. In *Proceedings of 2007 International Congress on Acoustics*. Madrid.

Puckette, M. (1995). Phase-locked vocoder. In *Proceedings of the 1995 IEEE International Conference on Applications of Signal Processing to Audio and Acoustics*, pages 222–225.

Ramstein, C. (1991). *Analyse, représentation et traitement du geste instrumental*. Phd Thesis, Institut National Polytechnique de Grenoble.

Rank, E. (1999). A player model for MIDI control of synthetic bowed strings. In *DIDEROT Forum on Mathematics and Music*. Wien.

Rasamimanana, N., Fléty, E., & Bevilacqua, F. (2006). Gesture analysis of violin bow strokes. *Lecture Notes in Computer Science*, 3881:145–155.

Rasamimanana, N. H., Bernardin, D., Wanderley, M., & Bevilacqua, F. (2009). String bowing gestures at varying bow stroke frequencies: A case study. In *Advances in Gesture-Based Human-Computer Interaction and Simulation*, volume 5085 of *Lecture Notes in Computer Science*, pages 216–226. Springer Verlag.

Rasamimanana, N. H. & Bevilacqua, F. (2008). Effort-based analysis of bowing movements: evidence of anticipation effects. *Journal of New Music Research*, 37(4):339–351.

Rovan, J. B., Wanderley, M. M., Dubnov, S., & Depalle, P. (1997). Instrumental gestural mapping strategies as expressivity determinants in computer music performance. In *Proceedings of Kansei - The Technology of Emotion*. Genova.

Schelleng, J. C. (1973). The bowed string and the player. *Journal of the Acoustical Society of America*, 53:26–41.

Schoner, B., Cooper, C., Douglas, C., & Gershenfeld, N. (1999). Data-driven modeling of acoustical instruments. *Journal for New Music Research*, 28:28–42.

Schoner, B., Cooper, C., & Gershenfeld, N. (2000). Cluster weighted sampling for syntheis and cross-synthesis of violin family instruments. In *Proceedings of the 2000 International Computer Music Conference*. Berlin.

Schoonderwaldt, E. & Demoucron, M. (2009). Extraction of bowing parameters from violin performance combining motion capture and sensors. *Journal of the Acoustical Society of America (In Press)*.

Schoonderwaldt, E., Rasamimanana, N., & Bevilacqua, F. (2006). Combining accelerometer and video camera: reconstruction of bow velocity profiles. In *Proceedings of the 2006 International Conference on New Interfaces for Musical Expression*, pages 200–203. Paris.

Schoonderwalt, E. (2008). *Mechanics and acoustics of violin bowing*. Phd Thesis, The Royal Institute of Technology (KTH, Stockholm).

Schoonderwalt, E. & Wanderley, M. (2007). Visualization of bowing gestures for feedback: The Hodgson plot. In *Proceedings of the 2007 International Symposium in Musical Acoustics*. Barcelona.

Schumacher, R. T. (1979). Self-sustained oscillations of the bowed string. *Acustica*, 43:109–120.

Schwarz, D. (2000). A system for data-driven concatenative sound synthesis. In *Proceedings of the 2000 International Conference on Digital Audio Effects*. Verona.

Schwarz, D. (2004). *Data-driven concatenative sound synthesis*. Phd Thesis, IRCAM Centre Popidou, Université Pierre et Marie Curie (Paris 6).

Serafin, S. (2004). *The sound of friction : real-time models, playability and musical applications*. Phd Thesis, Stanford University.

Serra, X. (1989). *A System for Sound Analysis/Transformation/Synthesis based on a Deterministic plus Stochastic Decomposition*. Phd Thesis, Stanford University.

Serra, X. & Smith, J. (1990). Spectral modeling synthesis: A sound analysis/synthesis system based on a deterministic plus stochastic decomposition. *Computer Music Journal*, 14(4):14–24.

Smith, J. (1992). Physical modeling using digital waveguides. *Computer Music Journal*, 16(4):74-91.

Smith, J. (2002a). Physical modeling synthesis update. Technical Report. Stanford University.

Smith, J. (2002b). Viewpoints on the history of digital synthesis. Technical Report. Stanford University.

Smith, J. O. (accessed 2009). *Physical Audio Signal Processing, December 2008 Edition*. `http://ccrma.stanford.edu/ jos/pasp/`. Online book.

Smyth, T. & Abel, J. (2009). Estimating the reed pulse from clarinet recordings. *Proceedings of the 2009 International Computer Music Conference*.

Tolonen, T., Välimäki, V., & Karjalainen, M. (1998). Evaluation of modern sound synthesis methods. Technical Report. Helsinki University of Technology.

Traube, C., Depalle, P., & Wanderley, M. M. (2003). Indirect acquisition of instrumental gesture based on signal, physical and perceptual information. *Proceedings of the 2003 International Conference on New interfaces for Musical Expression*.

Traube, C. & Smith, J. O. (2001). Extracting the fingering and the plucking points on a guitar string from a recording. In *Proceedings of the 2001 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*. New York.

Trendelenburg, W. (1925). Die natürlichen grundlagen der kunst des streichinstrumentspiels. *Verlag von Julius Springer, Berlin*.

Viterbi, A. J. (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13(2):260–269.

Wanderley, M. M. (2001). *Perfomer-Instrument Interaction: Applications to Gestural Control of Sound Synthesis*. Phd Thesis, Université Pierre et Marie Curie (Paris 6).

Wanderley, M. M. & Depalle, P. (2004). Gestural control of sound synthesis. *Proceedings of the IEEE*, 92(4):632–644.

Witten, I. H. & Frank, E. (2005). *Data Mining: Practical machine learning tools and techniques, 2nd Edition*. Morgan Kaufmann.

Woodhouse, J. & Galluzzo, P. (2004). The bowed string as we know it today. *Acustica - Acta Acustica*, 90(4):579-589.

Young, D. (2002). The hyperbow: A precision violin interface. *Proceedings of the 2002 International Computer Music Conference*.

Young, D. (2007). *A Methodology for Investigation of Bowed String Performance Through Measurement of Violin Bowing Technique*. Phd Thesis, Massachusetts Institute of Technology.

Young, D. (2008). Classification of common violin bowing techniques using gesture data from a playable measurement system. In *Proceedings of the 2008 International conference on New Interfaces for Musical Expression*. Genova.

Young, D. & Serafin, S. (2003). Playability evaluation of a virtual bowed string instrument. In *Proceedings of the 2006 International Conference on New Interfaces for musical Expression*, pages 104–108. Montréal.

Appendix A

A brief overview of Bézier curves

This appendix briefly describes the basic foundations of the so called *Bézier curves*. They are attributed to and named after a French engineer, Pierre Bézier, who used them for the body design of the Renault car makes back in the 1970s. They have since obtained dominance in the typesetting industry, as well as in other computer graphic applications. The overview given here is based on notes by Bourke (2000) and Jaun (2003). We refer the reader to references therein and to the work by Battey (2004) for further information.

Consider $N+1$ control points p_k, $k \in \{0,1,2,..,N\}$ in a space of arbitrary dimensionality. The Bézier parametric curve function $B(u)$ is of the form

$$B(u) = \sum_{0 \leq k \leq N} p_k \frac{N!}{k!(N-k)!} u^k (1-u)^{N-k}, \quad 0 \leq u \leq 1 \tag{A.1}$$

$B(u)$ is a continuous function in n-dimensional space defining a curve with $N+1$ discrete control points $P_k = \{p_0, p_1, .., p_N\}$. $u = 0$ at the first control point $(k = 0)$ and $u = 1$ at the last control point $(k = N)$. Some of the most important properties of this function are outlined next:

- The curve in general does no pass through any of the control points except the first and last. From the formula, $B(0) = p_0$ and $B(1) = p_N$.
- The curve is always contained within the convex hull of the control points, so it never oscillates wildly away from the control points.
- If there is only a control point p_0 (so $N = 0$), then $B(u) = P_0 \; \forall u$.
- If there are only two control points p_0 and p_1, the the formula reduces to a line segment between the two control points:

$$B(u) = \sum_{0 \leq k \leq 1} p_k \frac{N!}{k!(1-k)!} u^k (1-u)^{1-k} = p_0 + u(p_1 - p_0), \quad 0 \leq u \leq 1 \tag{A.2}$$

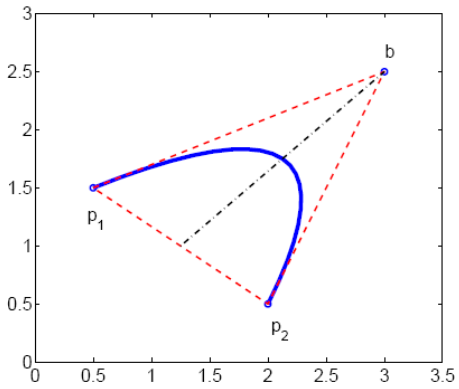


Figure A.1: Two-dimensional quadractic Bézier curve defined by the start and end points $p_1 = \{0.5, 1.5\}$ and $p_2 = \{2, 0.5\}$, and a single control point $b = \{3, 2.5\}$

- For the case of three control points, curves are usually called *quadratic Bézier curves*, and the formula reduces to:

$$B(u) = p_o(1-u)^2 + 2p_1u(1-u) + p_2u^2, \quad 0 \leq u \leq 1 \tag{A.3}$$

  An example showing a two-dimensional Bézier curve with three control points is shown in Figure A.1.

- For the case of four control points, curves are usually called *cubic Bézier curves*, and the formula reduces to:

$$B(u) = p_o(1-u)^3 + 3p_1u(1-u)^2 + 3p_2u^2(1-u) + p_3u^3, \quad 0 \leq u \leq 1 \tag{A.4}$$

  An example showing a two-dimensional Bézier curve with four control points is shown in Figure A.2.

- The term

$$\frac{N!}{k!(N-k)!}u^k(1-u)^{N-k} \tag{A.5}$$

  is called *blending function* since it blends the control points to form a Bézier curve.

- The blending function is always a polynomial one degree less than the number of control points. Thus, three control points results in a parabola, four control points a cubic curve, and so on.

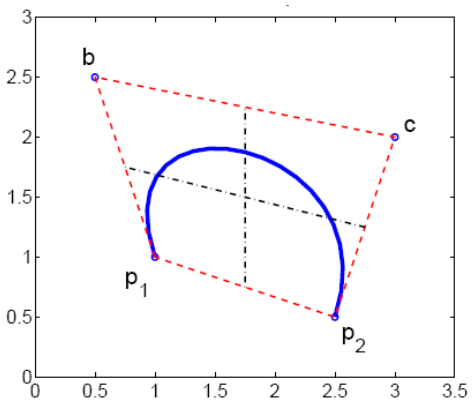


Figure A.2: Two-dimensional cubic Bézier curve defined by the start and end points $p_1 = \{1,1\}$ and $p_2 = \{2.5,0.5\}$, and two control points $b = \{0.5,2.5\}$ and $c = \{3,2\}$.

- When it results necessary, first order continuity between concatenated Bézier curves (splines) can be achieved by ensuring that the tangent between the last two control points of one curve matches the tangent between the first two control points of its successor curve.

Appendix B

Synthetic bowing contours

An extended collection of synthetic contours is displayed here. The contour rendering process 30 times for each note. The four different bowing techniques are shown first, together with different slur contexts for *legato* notes, and different silence contexts for *détaché* notes. Then, results showing the two bow directions and three dynamics are displayed for the four bowing techniques. Then, three different durations and four different pitches (effective string lengths) are respectively shown for *legato* and *détaché* notes, and for the four bowing techniques. Three different bow starting positions were used to render the same *détaché* note, showing very little differences. Finally contours rendered with two variance scaling factors (see Section 4.2.3) are compared for the four bowing techniques.

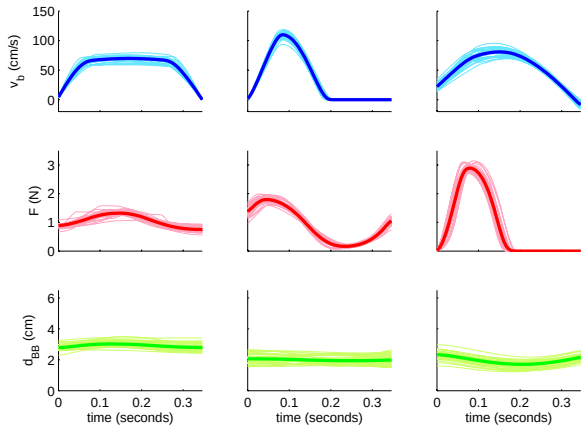


Figure B.1: Synthetic bowing contours for different bowing techniques. From left to right, [*détaché ff downwards iso mid*], [*staccato ff downwards iso mid*], and [*saltato ff downwards iso mid*]. Thin lines correspond to 30 generated contours, while the thick line in each plot corresponds to the time-average contour.

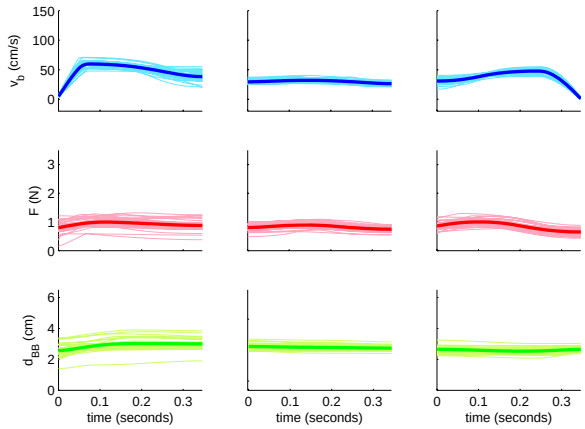


Figure B.2: Synthetic bowing contours for different slur contexts of *legato*-articulated notes. From left to right, [*legato ff downwards init mid*], [*legato ff downwards mid mid*], and [*legato ff downwards end mid*]. Thin lines correspond to 30 generated contours, while the thick line in each plot corresponds to the time-average contour.

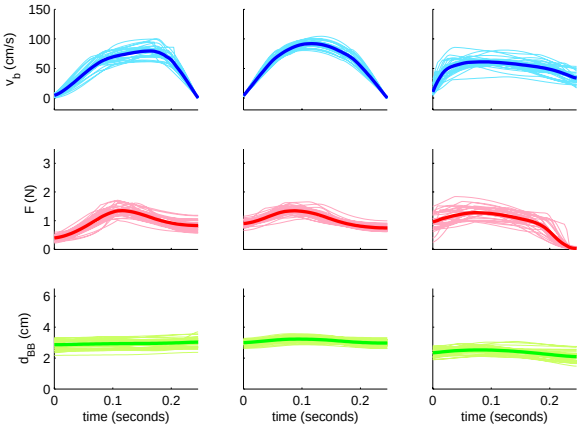


Figure B.3: Synthetic bowing contours for different silence contexts of *détaché*-articulated notes. From left to right, [*détaché ff downwards iso init*], [*détaché ff downwards iso mid*], and [*détaché ff downwards iso end*]. Thin lines correspond to 30 generated contours, while the thick line in each plot corresponds to the time-average contour.

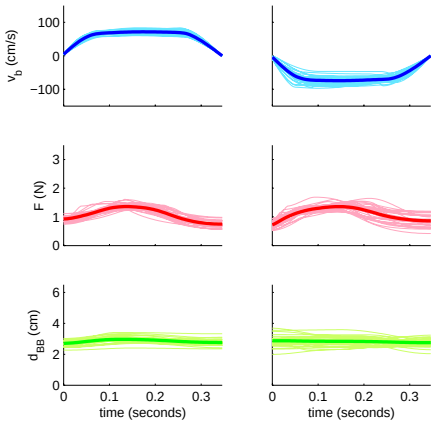


Figure B.4: Synthetic bowing contours of *détaché*-articulated notes, obtained for both bow directions. On the left, [*détaché ff downwards iso mid*]; on the left, [*détaché ff upwards iso mid*]. Thin lines correspond to 30 generated contours, while the thick line in each plot corresponds to the time-average contour.

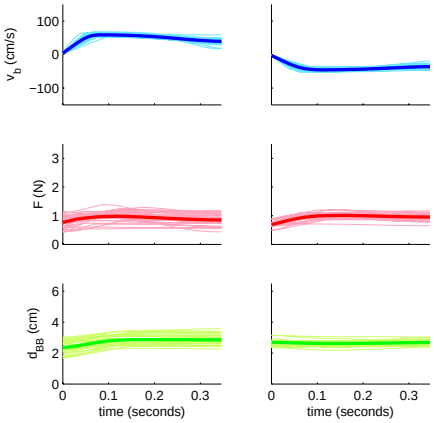


Figure B.5: Synthetic bowing contours of *legato*-articulated notes (starting a slur), obtained for both bow directions. On the left, [*legato ff downwards init mid*]; on the left, [*legato ff upwards init mid*]. Thin lines correspond to 30 generated contours, while the thick line in each plot corresponds to the time-average contour.

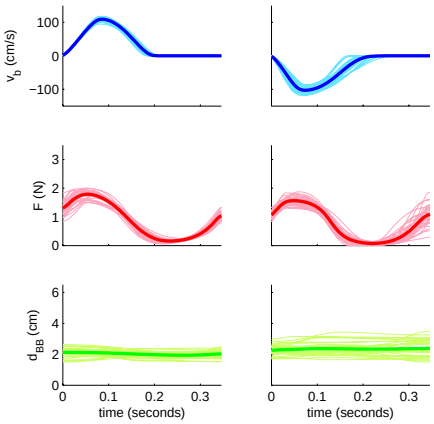


Figure B.6: Synthetic bowing contours of *staccato* notes, obtained for both bow directions. On the left, [*staccato ff downwards iso mid*]; on the left, [*staccato ff upwards iso mid*]. Thin lines correspond to 30 generated contours, while the thick line in each plot corresponds to the time-average contour.

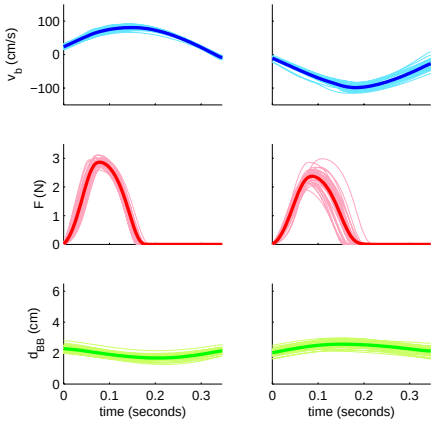


Figure B.7: Synthetic bowing contours of *saltato* notes, obtained for both bow directions. On the left, [*saltato ff downwards iso mid*]; on the left, [*saltato ff upwards iso mid*]. Thin lines correspond to 30 generated contours, while the thick line in each plot corresponds to the time-average contour.

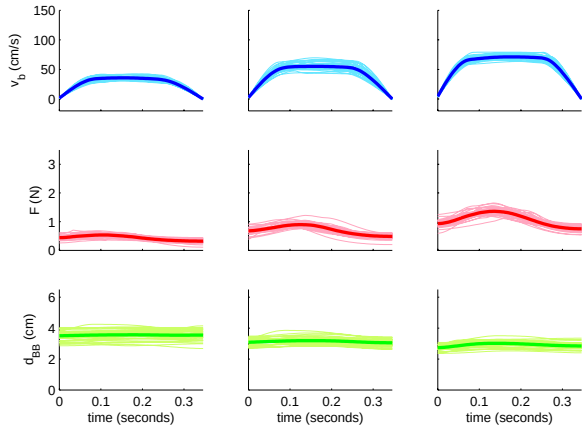


Figure B.8: Synthetic bowing contours of *détaché*-articulated notes, obtained for three different dynamics. From left to right, [*détaché pp downwards iso mid*], [*détaché mf downwards iso mid*], and [*détaché ff downwards iso mid*] . Thin lines correspond to 30 generated contours, while the thick line in each plot corresponds to the time-average contour.

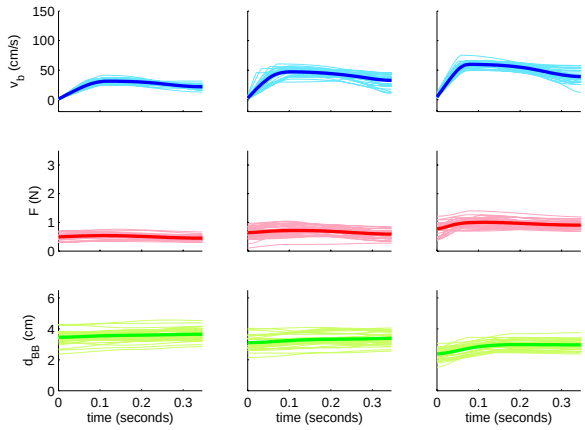


Figure B.9: Synthetic bowing contours of *legato*-articulated notes (starting a slur), obtained for three different dynamics. From left to right, [*legato pp downwards init mid*], [*legato mf downwards init mid*], and [*legato ff downwards init mid*] . Thin lines correspond to 30 generated contours, while the thick line in each plot corresponds to the time-average contour.

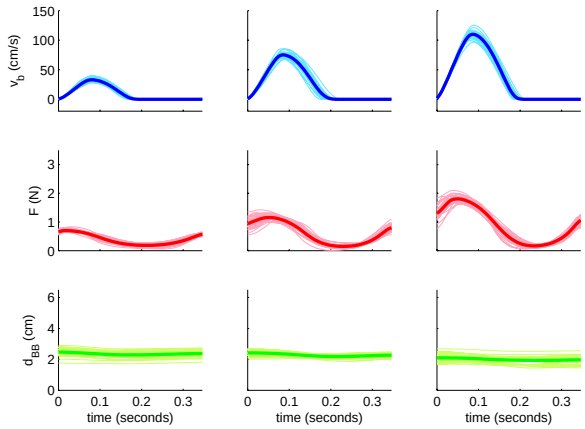


Figure B.10: Synthetic bowing contours of *staccato* notes, obtained for three different dynamics. From left to right, [*staccato pp downwards iso mid*], [*staccato mf downwards iso mid*], and [*staccato ff downwards iso mid*] . Thin lines correspond to 30 generated contours, while the thick line in each plot corresponds to the time-average contour.

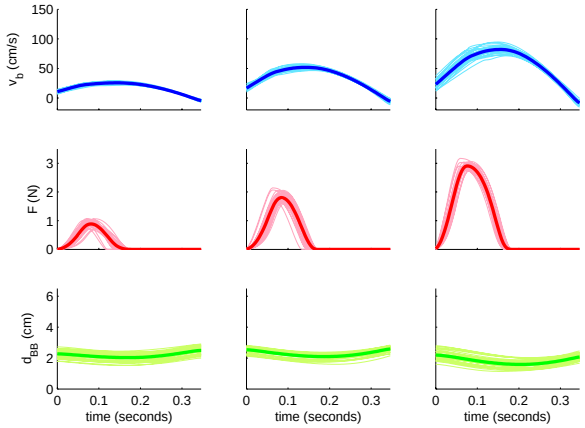


Figure B.11: Synthetic bowing contours of *saltato* notes, obtained for three different dynamics. From left to right, [*saltato pp downwards iso mid*], [*saltato mf downwards iso mid*], and [*saltato ff downwards iso mid*] . Thin lines correspond to 30 generated contours, while the thick line in each plot corresponds to the time-average contour.

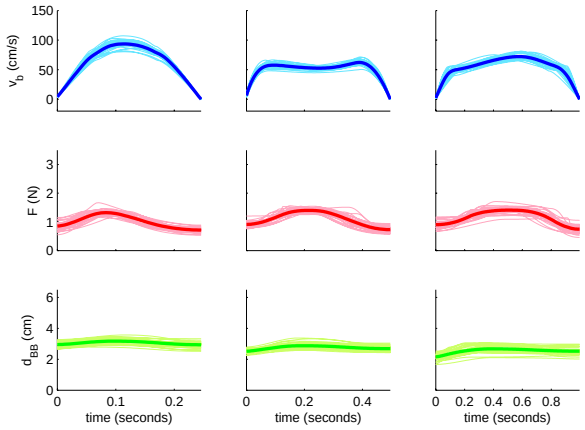


Figure B.12: Synthetic bowing contours of *détaché*-articulated notes, obtained for three different durations. All three columns correspond to [*détaché ff downwards iso mid*]. Thin lines correspond to 30 generated contours, while the thick line in each plot corresponds to the time-average contour.

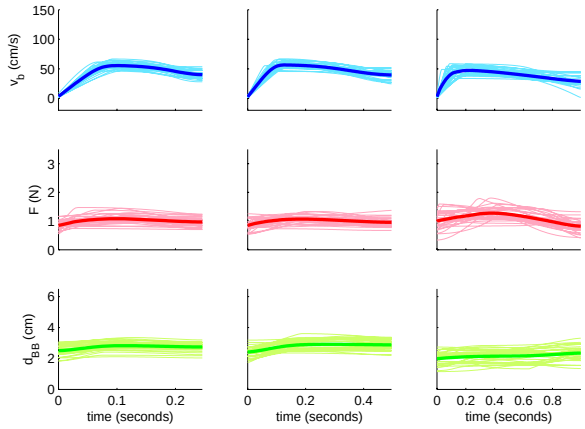


Figure B.13: Synthetic bowing contours of *legato*-articulated notes (starting a slur), obtained for three different durations. All three columns correspond to [*legato ff downwards iso mid*]. Thin lines correspond to 30 generated contours, while the thick line in each plot corresponds to the time-average contour.

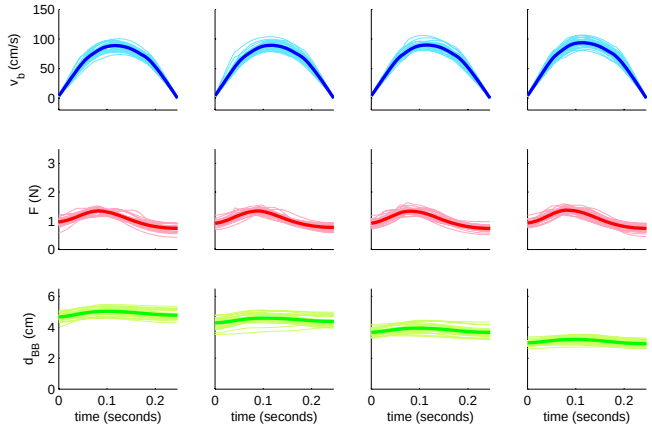


Figure B.14: Synthetic bowing contours of *détaché*-articulated notes, obtained for four different effective string lengths (finger positions). If played on the D string, contours shown in each row (from left to right) would respectively correspond to notes D (open string), E, F♯, and A. All four columns correspond to [*détaché ff downwards iso mid*]. Thin lines correspond to 30 generated contours, while the thick line in each plot corresponds to the time-average contour.

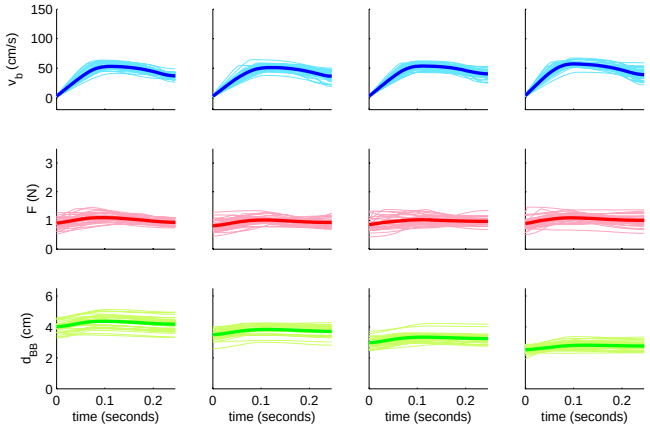


Figure B.15: Synthetic bowing contours of *legato*-articulated notes (bein, obtained for four different effective string lengths (finger positions). If played on the D string, contours shown in each row (from left to right) would respectively correspond to notes D (open string), E, F♯, and A. All four columns correspond to [*legato ff downwards iso mid*]. Thin lines correspond to 30 generated contours, while the thick line in each plot corresponds to the time-average contour.

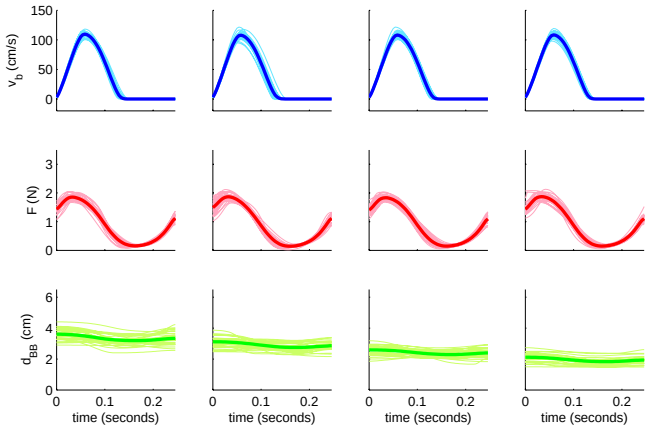


Figure B.16: Synthetic bowing contours of *staccato* notes, obtained for four different effective string lengths (finger positions). If played on the D string, contours shown in each row (from left to right) would respectively correspond to notes D (open string), E, F♯, and A. All four columns correspond to [*staccato ff downwards iso mid*]. Thin lines correspond to 30 generated contours, while the thick line in each plot corresponds to the time-average contour.

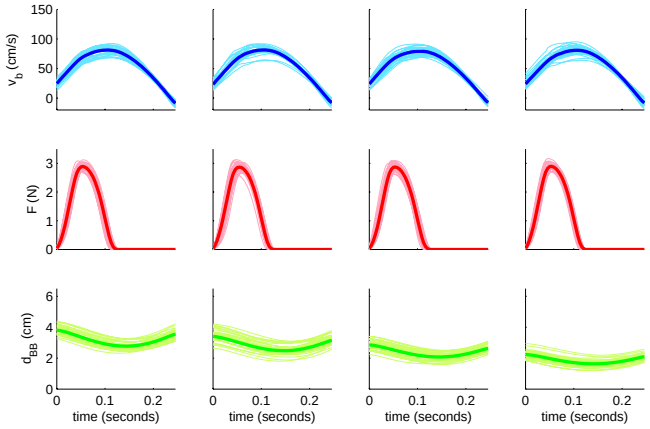


Figure B.17: Synthetic bowing contours of *saltato* notes, obtained for four different effective string lengths (finger positions). If played on the D string, contours shown in each row (from left to right) would respectively correspond to notes D (open string), E, F♯, and A. All four columns correspond to [*saltato ff downwards iso mid*]. Thin lines correspond to 30 generated contours, while the thick line in each plot corresponds to the time-average contour.

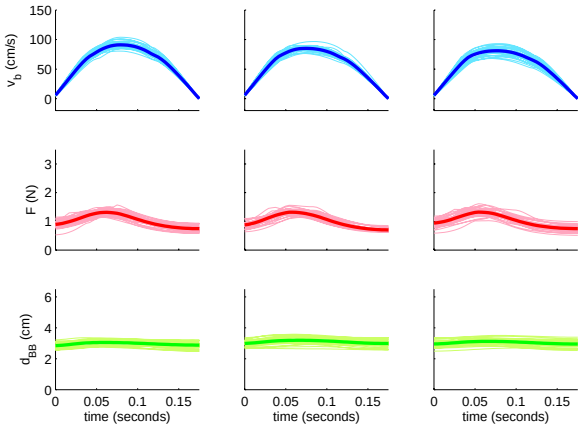


Figure B.18: Synthetic bowing contours of *détaché*-articulated notes, obtained for four bow starting positions BP_{ON} (measured from the frog). From left to right, contours displayed correspond to bow starting positions of 10cm, 20cm, and 30cm respectively. All four columns correspond to [*détaché ff downwards iso mid*]. Thin lines correspond to 30 generated contours, while the thick line in each plot corresponds to the time-average contour.

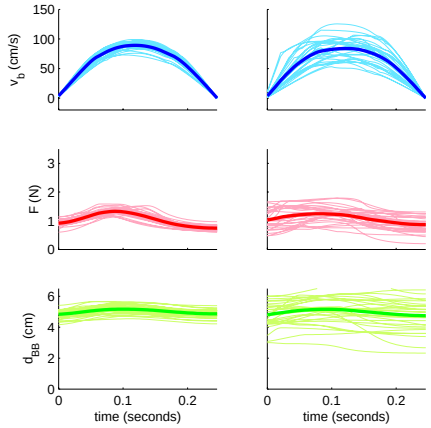


Figure B.19: Synthetic bowing contours of *détaché*-articulated notes, obtained for two different variance scaling factors (the column on the right displays contours by scaling the variances by a factor of 2.0. Both columns correspond to [*détaché ff downwards iso mid*]. Thin lines correspond to 30 generated contours, while the thick line in each plot corresponds to the time-average contour.

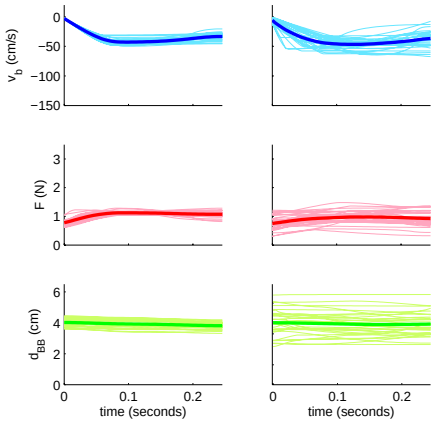


Figure B.20: Synthetic bowing contours of *legato*-articulated notes (starting a slur), obtained for two different variance scaling factors (the column on the right displays contours by scaling the variances by a factor of 2.0. Both columns correspond to [*legato ff downwards iso mid*]. Thin lines correspond to 30 generated contours, while the thick line in each plot corresponds to the time-average contour.

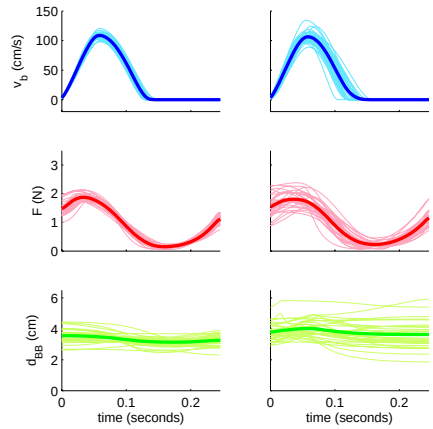


Figure B.21: Synthetic bowing contours of *staccato* notes, obtained for two different variance scaling factors (the column on the right displays contours by scaling the variances by a factor of 2.0. Both columns correspond to [*staccato ff downwards iso mid*]. Thin lines correspond to 30 generated contours, while the thick line in each plot corresponds to the time-average contour.

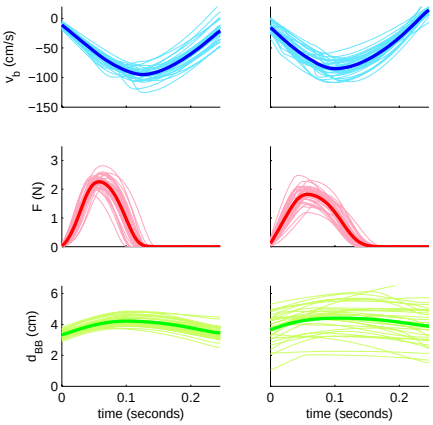


Figure B.22: Synthetic bowing contours of *saltatao* notes, obtained for two different variance scaling factors (the column on the right displays contours by scaling the variances by a factor of 2.0. Both columns correspond to [*saltato ff downwards iso mid*]. Thin lines correspond to 30 generated contours, while the thick line in each plot corresponds to the time-average contour.

Appendix C

On estimating the string velocity - bridge pickup filter

This appendix introduces a least-squares approach to the estimation of an equalization filter that approximates the transfer function relating the string velocity and the magnitude capture by the bridge pickup. Solving this problem is a requirement for implementing an interative optimization procedure devoted to the automatic calibration of the physical model, as introduced in Section 5.2.1.

For a given gestural context C_i (i.e., a point in the bowing parameter space defined by the effective string length L , the bow velocity v_b, the bow force F, and the β ratio), let $x_i \in \mathbb{R}^m$ be the MFCC vector containing the first m mel-cepstrum coefficients of an audio frame $W_k s_i(t)$ synthesized using Smith's digital waveguide-based physical model (Smith, 1992, accessed 2009). Assuming the existence of a linear time-invariant equalization filter $P(f)$ that relates the string velocity to the magnitude measured by the bridge pickup, its approximated spectral magnitude envelope can be represented as a MFCC vector $\eta \in \mathbb{R}^m$. At each step of the optimization procedure devoted to automatic calibration of the physical model, the equalization filter $P(f)$ will be applied to any synthesized audio frame $W_k s_i(t)$ in order to model the string-pickup filtering effect, and therefore be able to rely on spectral envelope distance computation as a basis for driving the optimization behind automatic calibration. In the MFCC space, the spectral envelope representation of the synthesized frame after applying equalization can be expressed as the addition of the vectors x_i and η, resulting into a vector γ_i. This is written as:

$$\gamma_i = x_i + \eta \tag{C.1}$$

In the MFCC space, the MFCC vectors obtained from the recorded audio frames are clustered by attending to bowing control parameters, having each gestural context C_i corresponding to one of the obtained clusters and represented as a normal distribution m_i of its respective MFCC vectors, given by the mean vector

ξ_i and the covariance matrix Ω_i.

$$m_i = \{\xi_i, \Omega_i\} \tag{C.2}$$

Now, for a particular gesture context C_i, an error ε_i between a MFCC vector γ_i and an expected MFCC vector $\hat{\xi}_i$ is defined as the squared Mahalanobis distance between γ_i and the mean MFCC vector ξ_i of the corresponding recorded frames, defined as ${D^2}_i(\gamma_i, \xi_i)$ in the Ω_i norm. Note that the error ε_i can be seen as the negative log-likelihood of γ_i given c_i, expressed as $nlogL(\gamma_i \mid c_i)$. This can be written as:

$$\begin{aligned} \varepsilon_i &= n \log L(\gamma_i \mid c_i) \\ &= {D^2}_i(\gamma_i, \xi_i) \\ &= (\gamma_i - \xi_i)^T \Omega_i^{-1} (\gamma_i - \xi_i) \\ &= (x_i + \eta - \xi_i)^T \Omega_i^{-1} (x_i + \eta - \xi_i) \end{aligned} \tag{C.3}$$

The problem is to find an optimum η^* that, applied to all x_i MFCC vectors, with $i = \{1, 2, \cdots, N\}$ being N the total number of gestural contexts into consideration, leads to a minimum weighted average error $\varepsilon_{min}(\eta)$ over all gesture contexts C_i, as expressed in equation (C.4), where w_i represents the weight applied to each error ε_i (details about the weighting are given later on).

$$\begin{aligned} \eta^* &= \underset{\eta}{\operatorname{argmin}}\ \varepsilon(\eta) \\ &= \underset{\eta}{\operatorname{argmin}} \sum_{i=1}^{N} w_i \varepsilon_i \\ &= \underset{\eta}{\operatorname{argmin}} \sum_{i=1}^{N} w_i (x_i + \eta - \xi_i)^T \Omega_i^{-1} (x_i + \eta - \xi_i) \end{aligned} \tag{C.4}$$

In order to solve this problem, classical least-squares can applied by making the gradient $\nabla_\eta\ \varepsilon(\eta)$ equal to zero, and solving for η. This is expressed in equation (C.5) (where the variable change $\psi_i = x_i - \xi_i$ has been applied before deriving the gradient) and in equation (C.6). The solution for η^* is written in equation (C.7).

$$\eta^* = \underset{\eta}{\operatorname{argmin}}\ \varepsilon(\eta) = \underset{\eta}{\operatorname{argmin}} \sum_{i=1}^{N} w_i (\psi_i + \eta)^T \Omega_i^{-1} (\psi_i + \eta) \tag{C.5}$$

$$\nabla_\eta\ \varepsilon(\eta) = 2 \sum_{i=1}^{N} w_i {\Omega_i}^{-1} \psi_i + 2 \sum_{i=1}^{N} w_i {\Omega_i}^{-1} \eta = 0 \tag{C.6}$$

$$\eta^* = -\left(\sum_{i=1}^{N} w_i {\Omega_i}^{-1} \right)^{-1} \sum_{i=1}^{N} w_i {\Omega_i}^{-1} (x_i - \xi_i) \tag{C.7}$$

In order to give each error ε_i an appropriate weigthing, one might take into account the number of samples describing each distribution c_i, which correspond

to the number of frames found for a particular context C_i when performing clustering of the recorded data. This way, the most populated gestural contexts will be given more importance. For doing so, we weight each ε_i with a ratio relating the number of frames M_{C_i} used for describing a particular context C_i, and the total number of frames M in the gesture space. This can be written as:

$$w_i = \frac{M_{C_i}}{M} \tag{C.8}$$

An alternative method for weigthing ε_i is to consider how the gesture contexts C_i are distributed in the gesture space. For doing so, the centroids ρ_i of the clusters representing the N gesture contexts C_i are collected, and the parameters of a normal distribution $p_{C_i} = \{\phi_{C_i}, \Phi_{C_i}\}$ are estimated. Then each ε_i is weighted by the log-likelihood $\log L(\rho_i \mid p_{C_i})$ of its corresponding ρ_i given the distribution p_{C_i} (see equation (C.9)). This way, gestural contexts that are more likely to happen will be given more importance.

$$w_i = \log L(\rho_i \mid p_{C_i}) = [(\rho_i - \phi_{C_i})^T \Phi_{C_i}^{-1} (\rho_i - \phi_{C_i})]^{-1} \tag{C.9}$$

This dissertation was written using LaTeX .
Data graphics were produced with *The Mathworks© MATLAB*.
Art was designed with the help of *Microsoft© PowerPoint*.