Journal of New Music Research
2006, Vol. 35, No. 1, pp. 71 – 93

Mapping Strategies for Gestural and Adaptive Control of Digital Audio Effects

Vincent Verfaille,[1] Marcelo M. Wanderley[1,2] and Philippe Depalle[1]

[1]Sound Processing and Control Laboratory (SPCL), [2]Input Devices and Music Interaction Laboratory (IDMIL), McGill University, Canada

Abstract

This paper discusses explicit mapping strategies for gestural and adaptive control of digital audio effects. We address the problem of defining what is the control and what is the effect. We then propose a mapping strategy derived from mapping techniques used in sound synthesis. The explicit mapping strategy we developed has two levels and two layers for each level: the first level is the adaptive control with a feature combination layer and a control signal conditioning layer; the second level is the gestural control layer. We give musical examples that illustrate the interest of this strategy.

1. Introduction

When manipulating digital sound on computers, two main approaches are used: modifying digital sounds (audio effects) and creating digital sounds from scratch (sound synthesis). Both approaches were developed simultaneously from the beginning of computer music in order to create new sounds and simulate existing ones. As long as processes were applied offline, taking hours of computation, the control of the parameters was pre-established by the composer. In a real-time context, the control of synthesis or audio effect parameters offers auditory feedback and interaction with processes. Then, ways to manipulate several synthesis/effect parameters with a limited number of inputs are needed. This is what we call mapping.

The music technology community has recently started to investigate mapping in the context of sound synthesis (Fels, 1994; Wanderley & Battier, 2000; Wanderley & Depalle, 2004). Although somewhat limited when applied to the structure of complex interactive instruments (Chadabe, 2002), the mapping concept is useful when applied to the structure of electronic instruments modelled after traditional acoustic instruments, and as a result has produced several mapping toolkits (Bevilacqua et al., 2005; Steiner, 2005). However, mapping has not been thoroughly investigated in the context of sound effects, with gestural control (Wanderley & Depalle, 2000; Arfib, 2002), and with adaptive and gestural control (Arfib & Verfaille, 2003; Verfaille, 2003). It may be due to a difference in the complexity of mappings: a high quality synthesis such as additive synthesis may require several hundreds of coefficients, whereas an audio effect usually only requires a few parameters. This may also be due to the fact that sound synthesis is considered as something to control and perform, unlike audio effects. However, audio effects can also be performed (Todoroff, 2002): this is true for performers using interactive systems, and composers and sound engineers modifying the effects settings in real-time. Algorithmic composition also suffers from a lack of investigation from scientists: “Many composers use mapping either explicitly or implicitly, in their compositional practice [...], but well documented and detailed examples of exactly how mapping is used are very rare. This is probably because it has not been identified as a separate step in composition practice.” (Doornbusch, 2002, p. 146)

Correspondence: Vincent Verfaille, Sound Processing and Control Laboratory (SPCL), Music Technology Area, Schulich School of Music, McGill University, 555, Sherbrooke Street West, Montréal, Québec, Canada H3A 1E3. E-mail: vincent@music.mcgill.ca

DOI: 10.1080/09298210600696881 © 2006 Taylor & Francis

Our idea is that the mapping of audio effects has to be explored, ideally using techniques and vocabulary already defined for sound synthesis, since both sound synthesis and audio effects are performed in real-time. Recently, performers and composers have started to systematically explain the musical use they make of digital audio effects (Risset, 2002), and the mappings they used for controlling audio effects (Burtner, 2002; Myatt, 2002). However, no systematic effort has been carried out by the scientific community to propose a general mapping framework that encompasses the various control types of audio effects, and for the widest range of audio effects.

In the present article we propose such a framework by addressing mapping strategies for adaptive, automatic and gestural control of digital audio effects. We first give a definition of effect, control and mapping (Section 2). We then define a set of mapping strategies we developed for digital audio effects (Section 3), and specifically a two level and two layer explicit mapping strategy designed by separating adaptive from gestural control, and feature combination from control signal conditioning. We finally give six examples of musical use of our strategy (Section 4), develop the musical implications of our mapping model onto adaptive effects, discuss the advantages and drawbacks or our techniques, and give some perspective of this study (Section 5) before concluding (Section 6).

2. Basics about effects control and mapping

2.1 Digital audio effects (DAFx)

The acronym DAFx stands for Digital Audio Effects, defined as "boxes or software tools with input audio signals or sounds which are modified according to some sound control parameters and deliver output signal or sounds" (Zölzer, 2002a). These musical signal processing units are used either to modulate or to modify an audio signal (Verfaille et al., 2006). Modulations are applied to amplitude (tremolo), frequency (pseudo-vibrato), and timbre via a filter (flanger, wah-wah). Examples of modifications are time-scaling, pitch-shifting and timbre morphing.

Digital audio effects allow for transformations of a digital sound that are audible and may have a perceptual meaning. For example, an echo can be simulated as a reflection of the sound onto a distant wall, in a wide and non-distorting space by a single digital delay line. A digital audio effect processes sounds without a specific model, whereas a sound transformation is part of an analysis–synthesis model of sound. Such models perform block-by-block analysis to represent the digital sound either in the time-domain (SOLA, PSOLA, WSOLA), in the time-frequency domain (phase vocoder, with phase locking) or the frequency domain (additive model, or spectral model SMS, TMS). Refer to Zölzer (2002b) and Verfaille et al. (2006) for more details.

Once computed, the sound representation and the sound transformation processes consist of modifying the representation parameters before resynthesis. Even though digital audio effects and sound transformations differ in technical aspects, they share the same goal, which is to provide musically meaningful modifications of a sound. For that reason and in order to simplify vocabulary, we will not differentiate them and we will use DAFx for both throughout the whole paper.

2.2 Control of DAFx

DAFx are usually controlled via a few parameters that set the signal processing algorithm to affect perception in a specific way. Given a specific system, various DAFx can be implemented according to control. For example, a system with fractional delay lines can be used to perform various control strategies and effects, depending on the number of delay lines and their length:

- one delay line: frequency modulation (transposition, vibrato);
- original sound plus one delay line:
  - echo: constant length (delay time $\tau > 100$ ms);
  - comb filter: original sound plus one delay line ($\tau \leq 50$ ms); chorus (random variation and $\tau \in [1, 10]$ ms), flanger (sine wave modulation of $\tau \in [20, 30]$ ms);
- original sound plus several delay lines:
  - multi-echo: constant lengths, $\tau \geq 100$ ms;
  - phasing: constant/varying lengths, $\tau \leq 100$ ms;
  - reverberation: delay line networks.

Then, by combining these systems, other effects such as stereophonic chorus, 4-layer chorus, or complex phasing can be designed.

On the other hand, when controlling a sound transformation (based on an analysis–synthesis technique) such as the additive model, many parameters can be controlled, and various transformations can be performed, for instance pitch-shifting, time-scaling, auto-tuning or inharmonizing. Hence, the control must be clearly defined. In Table 1, we indicate for the main existing DAFx the attributes of sound perception that are modified and their control type.

2.3 Types of control

We consider five control categories (cf. Table 2): low frequency oscillator, gestural, automation, adaptive and algorithmic. They can be grouped into two categories of generators, namely wave generators and arbitrary generators. Arbitrary generators are divided into three sub-categories: user-defined, sound-defined and algorithmically-defined. User-defined control is provided by real-time gestural control, or offline gestural control (automation). Sound-defined control corresponds to adaptive control. In both user-defined and sound-defined cases, some features (or parameters, descriptors) are extracted and represent either gesture or sound. Systems like the UPIC designed by Iannis Xenakis combine automation, gesture and algorithmic control (Marino et al., 1993).

Table 1. Main digital audio effects modifying several perceptual 'attributes' (L: loudness, D: duration and rhythm, P: pitch and harmony, T: timbre and quality, S: spatialization). We indicate if real-time implementation is possible, and the control type.

	Perceptual Dim.			
Effect name	Main	Other	Real-time	Control type
compressor/limiter	L	T	✓	adapt.
tremolo	L		✓	LFO
time inversion	D	P,L,T	N/A	
time-scaling[1]	D		N/A	adapt.
transposition[2]	P		✓	
auto-tune	P		✓	adapt.
smart harmony	P		✓	adapt.
Doppler	S	L,P	✓	
echo	S	L	✓	
granular delay	S	L,D,P,T	✓	adapt.
panning	S		✓	
equalizer	T	L	✓	
wah-wah	T	L,P	✓	
chorus	T		✓	random
flanger	T	P	✓	LFO
phaser	T	P	✓	
distortion	T	L,P	✓	
cross-synthesis	T	L,P	✓	adapt.
prosody change	L,D,P		✓	adapt.
resampling	D,T	L,P	N/A	
ring modulation	P,T		✓	
robotisation	P,T	L	✓	
vibrato	L,P	T	✓	LFO

[1] Time-scaling with pitch, spectral envelope and attack preservation.
[2] Transposition with spectral envelope preservation.

We now further explain the five categories of control. Gestural control is given through a gesture transducer: a surface control in digital studio environment, a set of pedals used by guitarists, any transducer used in the context of sound synthesis. This control can be continuous (modulation/modification gestures) as well as discontinuous (selection gestures) (Cadoz & Wanderley, 2000).

Automation is a gestural control given through a Graphical User Interface (GUI): values of one effect control are represented by a segment-line curve. This curve is discontinuous, and can be made continuous by the host software that produces the automation. Even though digital multitrack recording and editing software allows for real-time modification of the segment-lines, it is not exactly a real-time control as is the gestural control using a transducer.

Low frequency oscillator (LFO) control consists of having an LFO that drives a control value, e.g. tremolo, vibrato or auto-wah rate, and length of a flanger's delay-line. It is a continuous control, used as well for real-time as non-real-time applications.

Adaptive control consists of using sound features as control values for the effect with non-linear warping laws (e.g. compressor/expander, auto-tune, cross-synthesis) (Verfaille & Arfib, 2001; Verfaille et al., 2006). Depending on the sound feature extracted, it can be continuous (e.g. energy, fundamental frequency) or discontinuous (e.g. onset/offset detection).

Algorithmic control consists of algorithmic description of control using mathematical formulae. A specific case is adaptive control, where the input of the algorithmic control are sound features. Score following and event-dependent control while score following are other examples. Sequences can be defined by the composer or the performer, where an identical control will be mapped with different rules, i.e. different algorithmic controls.

It is interesting to note that a specific control type is sometimes part of the effect itself, e.g. LFO control for tremolo, vibrato, flanger, and auto-wah, or adaptive for

Table 2. Controls of an effect: real-time context, types of control and control origin.

Control name	Control type	Real-time	Offline	Continuous	Discont.	From
	wave generator					
LFO		✓	✓	✓	–	oscillator
	arbitrary generators					
gestural	user-defined	✓	–	✓	✓	gesture
automation	user-defined	–	✓	(✓)	✓	gesture
adaptive	sound-defined	✓	✓	✓	✓	sound
algorithmic	algorithmic-defined	✓	✓	✓	✓	equations

auto-tune, distortion, compressor, expander and cross-synthesis. Nevertheless, another layer of LFO or adaptive control can be added on top of it, as it can be added to any effect. Indeed, all control types are not exclusive and can be combined: automation, LFO, and adaptive layers can be gesturally controlled.

2.4 Mapping input controls to the effect controls

From these five control categories, we can start to investigate mapping strategies from input controls to effect controls. This mapping is sometimes called *input mapping*, as it "translates user's actions into parameter values needed to drive the sound processing algorithms", whereas the output mapping does the reverse, "representing the algorithms parameters in a way that makes sense to the user" (Todoroff, 2002). Mapping can be explicit, i.e. defined by mathematical expressions such as $c[n] = \mathcal{F}(f_i[n])$, or implicit, i.e. defined using the "black-box" model.

A number of composers and performers explained the mapping they used to control DAFx. One example of documented control of DAFx is given in (Myatt, 2002), where the developed system controls amplitude modulation, frequency modulation, mixing, filter/wah-wah, multi-brassage, spatialization and filter/convolver. The mappings used are simple: trigger, exponential law, or binary law. Gestures are identified using neural networks analysis. The provided system was extensively used for performance, providing interesting cues on learning, expressiveness and quality of the musical result. Another example is the MetaSaxophone, an augmented instrument (Burtner, 2002) built by adding to an acoustic saxophone lots of pressure sensors on the keys to trigger or modulate processes. The MetaSaxophone allows for independent control of several effects: amplitude modulation, distortion, reverberation and sampling. An important issue raised is the control notation for such an instrument.

Mapping has to be clearly defined and structured, thus requiring two main steps: defining first what in a sound processing algorithm belongs to the control and to the effect, and second how to connect the control input into effect control values.

2.5 Frontier between effect and control

As already explained in specific cases of LFO and adaptive control, part of the signal processing system belongs to the effect and part belongs to its control (see Figure 1). This distinction clarifies what makes the effect specific, and what can be controlled. An example was given in Section 2.2 with delay lines, which are controlled directly or via a LFO to provide various DAFx. The following two examples develop this point further. When using a delay line to produce echoes, the user either directly controls delay length and reinjection gain, or controls the number of beats per minute (BPM) using one mapping layer on delay length, or controls the number of repetitions using one mapping layer on the delay reinjection gain. When pitch-shifting to produce vibrato, the user either controls vibrato frequency and depth, or directly controls the pitch-shifting ratio (Hunt et al., 2002; Arfib & Verfaille, 2003). LFO is a specific mapping layer of vibrato, but also tremolo, flanger, etc.

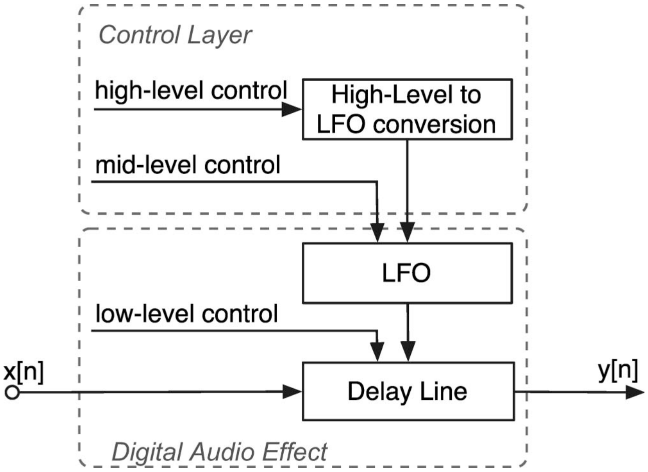

Fig. 1. Example diagram showing what we consider as the effect (delay-line modulation) and as control.

In order to define the frontier between processing and control mapping level, *we consider as part of the effect the control mapping layer that is specific to the way it sounds, whereas the other control layers belong to the control mapping level*. An effect is then the combination of a processing technique (e.g. delay-line modulation for flanger, pitch-shifting for vibrato) and a first control level (e.g. LFO, time to BPM conversion, amplitude to dB conversion). Then, control is given to the user at a low level (e.g. a vibrato with $f = 5$ Hz and $d = 6$ dB) or at a high level (e.g. a vibrato that automatically starts and stops, and adapts to note change). Studies about high level control of sound transformation (e.g. content-based, adaptive) use additional mapping layers. An effect directly affects perception with meaningful values with regards to perceptual cues. When the user manipulates the signal processing system itself, he/she often affects perception in a less transparent or intuitive way than when manipulating the effect integrated control level. What is direct from the perceptual modification point of view may be indirect from the signal processing point of view, and vice versa.

2.6 Adaptive and gestural control of DAFx

We now focus on gestural and adaptive control of an effect in music production, offline composition, and real-time performance contexts.

The direct control of an effect is the simplest case of gestural control. For instance, a surface control may be manipulated by the user (musician, sound engineer) using knobs, circular or linear potentiometers, or other gestures (Wanderley & Depalle, 2000), and gives control values to the effect, as depicted in Figure 2. In this case, mapping is direct, explicit and usually 1 to 1.

The direct control with gesture analysis consists in extracting the intention in the gesture through an analysis layer, as depicted Figure 3. An example is score following, where the gestures of a conductor are analysed in order to synchronize pre-recorded segments of audio to the beat. In this case, the mapping is divided into two successive complex mapping layers: the gesture feature extraction layer and the gesture features to effect control values mapping layer. The gesture feature extraction consists of computing a set of features/descriptors $f_i[n]$ from the gesture transducer data with mathematical operators, e.g. position, speed, acceleration, a form that is recognized, data segmentation into elementary gestures. The mapping layer between gesture features and effect control values is often considered as part of the instrument definition (Arfib et al., 2002b; Hunt et al., 2002), since it defines the transformation of gesture features into ‘musically interesting’ control values, via its bounds, variation type and behaviour.

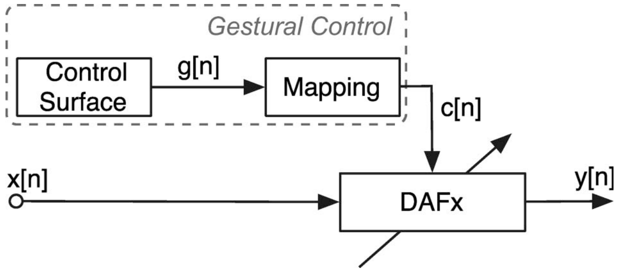

Fig. 2. Diagram showing direct gestural control of an effect.

The adaptive control of an effect consists of using sound features as DAFx control parameters (Verfaille & Arfib, 2001; Verfaille et al., 2006). Typical examples are compressor/expander, auto-tune, cross-synthesis, or score following. As depicted in Figure 4, two successive complex mapping layers are needed: a sound feature extraction layer and a layer that transforms features into effect control values. The sound feature extraction layer is designed in the same way as the extracted gesture feature, but specifically for sound properties: mathematical operators (e.g. Fourier transform, gravity centre, mean, power, fundamental frequency estimation) transform wave forms into low level and high level features. Adaptive control appears as a way to enhance the correlation between output sound features.

By adding gestural control to adaptive control as depicted in Figure 5, two types of features have to be mapped to the effect controls: sound features and gesture features. Then, which mapping can be used between sound and gesture features, and effect controls? Is this mapping explicit or implicit? Do we need to separate or to merge sound feature mapping and gesture feature mapping? In the next section (Section 2.7) a short review of mapping techniques usually used in the context of

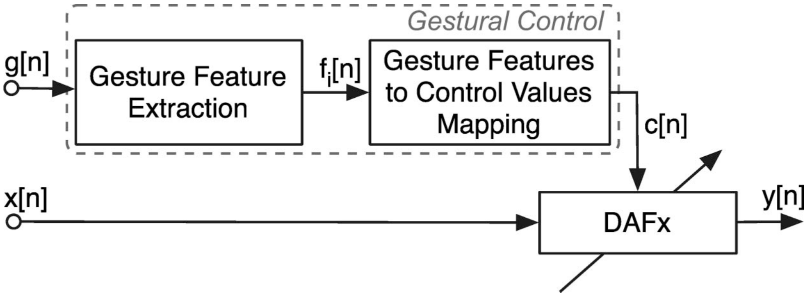
Gestural Control
g[n]
Gesture Feature Extraction
$f_i[n]$
Gesture Features to Control Values Mapping
c[n]
x[n]
DAFx
y[n]

Fig. 3. Diagram showing the gestural control of an effect with gesture feature extraction.

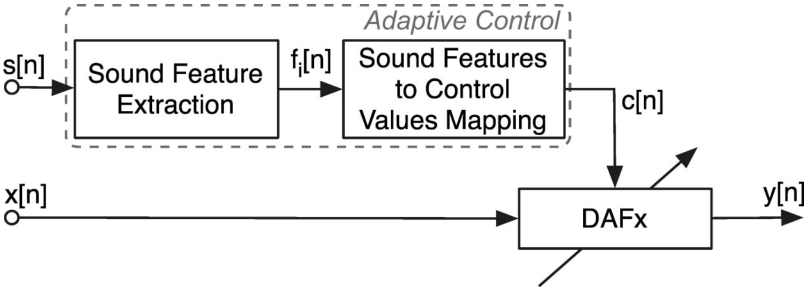

Fig. 4. Diagram showing the adaptive control of an effect: sound features are extracted and mapped to effect controls.

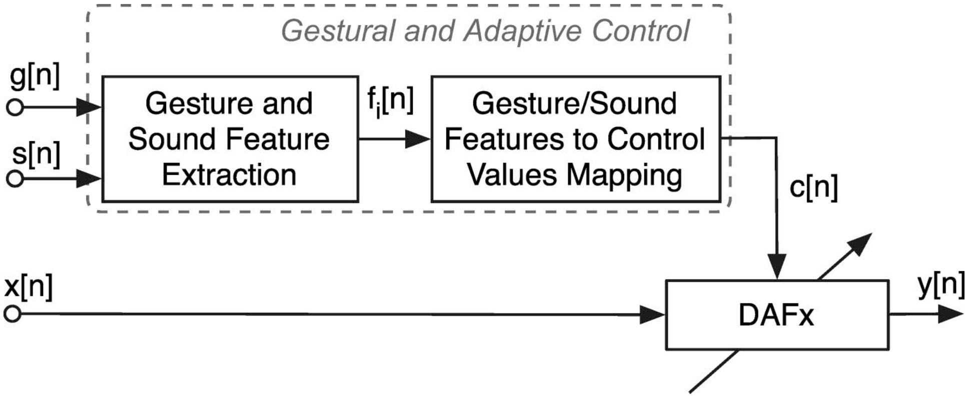
Gestural and Adaptive Control
g[n]
s[n]
Gesture and Sound Feature Extraction
$f_i[n]$
Gesture/Sound Features to Control Values Mapping
c[n]
x[n]
DAFx
y[n]

Fig. 5. Adaptive digital audio effect with gesture control: an input signal from one of the three possible signals is used for feature extraction.

sound synthesis is proposed, as it is a domain where many of these questions have been widely addressed (Wanderley & Battier, 2000; Wanderley, 2002). It is then compared with mapping for adaptive and gestural control of DAFx.

2.7 Mapping comparison in DAFx control, DMI control and algorithmic composition

Similiar issues are addressed by adaptive control of DAFx and by gestural control of sound synthesis (see Table 3), due to the fact that adaptive control consists in using musical gestures directly extracted from signals (Bailey et al., 1993; Egozy, 1995; Métois, 1998; Marrin-Nakra, 2000; Traube & Smith, 2000). Musical gestures are correlated to the extracted sound features. A review of interesting features for adaptive control of DAFx was proposed in Verfaille et al., (2006). The feature extraction process is a mapping by itself (see Arfib et al. (2002b) and Section 3.1), however it is not usual to modify its control, since it is finely tuned to perform a good analysis. From the observation of digital music instrument (DMI) design and mapping in algorithmic composition, we can propose clear definitions of mapping that are also valid for DAFx.

Digital musical "instrument design [is] the putting of 'physical handles on phantom models', discovering which controls ('handles') work well with mapping into a synthesizer" (Ryan, 1992). Then, *adaptive effect design is the putting of sound information handles as well as gestural handles on digital audio effects, discovering which sound features and gestural controls work well with mappings and sounds into a digital audio effect unit*.

When designing a digital musical instrument (DMI) (Wanderley & Depalle, 2004), the following questions: "from what? to what? by what means?" (Goudeseune, 2002) help to define mapping. Answers consider "the interaction between individual controls and individual synthesis parameters". Mapping in sound synthesis is defined as the extraction of physical gestures and the connection of this set of data to the synthesis parameters. Mapping in composition is defined as the extraction of musical gestures (Doornbusch, 2002) and the connection of this set of data to composition parameters. Mapping in gesturally and adaptively controlled audio effects is defined as the extraction of physical and musical gestures and the mapping of this set of data to the effect parameters. Mapping sound features to effect controls is quite similar to mapping gesture parameters to synthesis control parameters, except in the particular case where musical events are segmented and the mapping is adapted to each segment (Levitin et al., 2002). The three definitions are similar and rephrased as follows: *mapping in composition/sound synthesis/audio effects is the extraction of a set of musical/physical gestures and its connection to a set of control parameters*.

When designing mapping, one may define what sounds a given controller might produce. Then, technical means are derived from the specific musical goals of the DMI, DAFx or algorithmic composition control. Examples of DAFx with specifically designed mapping are a compressor that compresses the dynamic range (Dutilleux & Zölzer, 2002), an auto-tune that quantizes fundamental frequency to the tempered scale (Amatriain et al., 2003), cross-synthesis that for instance makes a cello talk (Arfib et al., 2002c), and voice morphing (Depalle et al., 1995).

Conversely, exploration of mapping can reveal new musical possibilities. From the composers' point of view (Doornbusch, 2002), it appears that mapping is often a creative approach, part of an exploration process, using both linear and non-linear techniques. It controls higher level structures and perceptual dimensions. Even though complex, it may be linear in perception, and thus non-linear in the signal. In previous works, the first author showed that a systematic investigation and thorough evaluation of mapping 65 sound features to one of 25 DAFx reveals new audio effects, such as adaptive

Table 3. Comparison of synthesis and digital audio effects control in the context of mapping.

Context & Concern	Synthesis	Effects/Transformations
Mapping definition		
from what? (input)	individual (gestural) controls	individual sound feature controls
to what? (output)	synthesis parameters	effect parameters
how to define it?	musical goals → technique investigation → new musical instruments	musical goals → technique investigation → new musical effects
Mapping description		
acquisition	by input devices	by feature extraction algorithms
controller	complete interface/set of commands available to the performer	complete set of sound features available to the performer
control	single indivisible part of the controller	single sound feature
	scalar/switch, continuous/discrete	
dimension	synthesis parameters, degrees of freedom	audio effect parameter (differs from sound perceptual attribute)
dimension value	scalar, dimension's realisation	
driving graph	arrows between a set of controls and a set of dimensions	
gain of a scalar control	how strongly inputs affect outputs	
order[1]	−1, 0, 1	−3, −2, −1, 0, 1, 2, 3...
mapping techniques	interpolation (from discretized space) neural networks, etc.	interpolation linear combination
Control Properties		
primary control	gesture	sound and/or gesture
secondary control	gesture	
type	absolute/relative	absolute
control bounds	bounded/unbounded in its motion	bounded

[1] Order 0 corresponds to proportional control, order 1 to its first order integral, order −1 to its first order derivative, and so on.

equalizer, adaptive spectral and tremolo, intonation change, and adaptive robotization (Verfaille, 2003; Verfaille et al., 2006). Going from techniques (combining sound feature extraction, audio effect and mapping modules) to musical aspects (selecting interesting solutions, from the perceptual and cognitive points of view) is a common attitude in music composition: Xenakis, for one, defined mathematical models of composition, and only selected "good sounding" results.

Two particular techniques proposed for sound synthesis are particularly adapted to DAFx control: interpolation (Allouis, 1984; Favreau, 2001; Terrugi, 2001; Goudeseune, 2002), and multiple-layer mappings (Fels, 1994; Arfib et al., 2002b; Hunt et al., 2002).

In tangible systems such as real and digital musical instruments, tactile information made of feedback signals enriches the control abilities by informing the performer about instrument and sound behaviour. Adaptive control of DAFx provides information coming from input or output/processed sound to the effect; however this feedback loop does not particularly involve the performer.

Mapping in composition is an inherent part of the algorithm (see Table 4). It is set before being used in real-time, and reflects the underlying conception, structure and planning of the music (Doornbusch, 2002). Mapping is controlled by the musical gestures the composer wants to use during the composition. Those musical gestures are related to the macro-scale of the structural model of the piece. This differs from DMI, where mapping is integrated to the instrument, designed by instrument makers who may be the performer or the composer, and concerns real-time music production. DMIs use physical gestures to control the micro-scale properties of sound. Its gestural control is also considered as part of the instrument. DAFx are also concerned with real-time production of music. The effect is controlled either by physical gestures (external gestural control) and musical gestures indirectly acquired by sound feature extraction (integrated adaptive control).

3. Mapping sound and gesture features into DAFx controls

In this section, we review some mapping strategies between sound and gestures features, and DAFx controls. Then, we discuss an explicit multi-layer mapping strategy that clearly separates adaptive and gestural control.

Table 4. Comparison of mapping between algorithmic composition, sound synthesis and audio effects.

Context	Mapping used for	Control scale	Gesture	Integration to	Designer
composition	process of planning	macro	compositional	software	composer
sound synthesis	RT music prod.	micro	physical	instrument	luthier, performer
audio effects	RT music prod.	micro	physical, musical	DAFx	composer, performer

3.1 Mapping strategies

After obtaining sound and gesture features such as curve input via a GUI, how do we combine them to obtain effect control signals? Several mapping strategies are possible, depending on how modular the mapping has to be. The most general strategy consists of having only one mapping level with two layers, that does not separate sound and gesture feature extraction: features are extracted and then combined, as depicted in Figure 6.

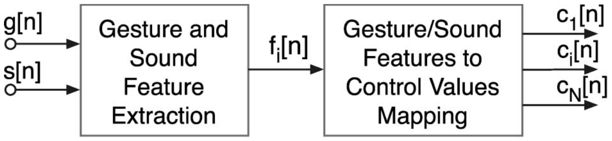
g[n]
s[n]
Gesture and Sound Feature Extraction
f_i[n]
Gesture/Sound Features to Control Values Mapping
c_1[n]
c_i[n]
c_N[n]

Fig. 6. General mapping strategy between sound/gesture features and effect controls.

Separating sound feature extraction from gesture extraction as depicted in Figure 7 allows for taking into account the following constraints:

- having a clear view about what is controlled, and
- possibility to disable the adaptive control, or
- possibility to disable the gestural control.

The gestural control mapping offers a control of the sound feature mapping. This clarifies the roles of adaptive and gestural control, and makes it a higher-level than direct control onto DAFx parameters. For instance, it allows one to use gestures to navigate between presets given by configurations of the adaptive control mapping layer. The use of modular design for this mapping strategy was inspired by modular mapping in sound synthesizers (Wanderley et al., 1998).

A more complex mapping strategy is given in Figure 8: sound features can modify the gesture feature mapping, which finally modify the sound feature mapping. The separation between the sound feature level and the gesture feature level does not exist anymore. This mapping strategy will be further developed in Section 3.2.

Features are always computed from a sound/gesture signal with mathematical formulae. This means that it is an explicit mapping, most of the time an *M*-to-*N* complex mapping. Some features are derived from others. For instance high-level features such as brightness are derived from low-level features such as the spectrum. The feature extraction mapping layer has generally no gestural control. The only input parameters of this mapping layer are generally set as constants, e.g. a STFT size or a window type, or adaptively changed, e.g. a window size set according to the fundamental frequency estimation (usually called "pitch-synchronous" analysis). Some musical applications can, however, benefit from gestural control of feature extraction. For example when analysing a sound with a subtractive model, sound is assumed to be a white noise affected by a given filter. By controlling the number of autoregressive coefficients, one can impose more or less perfect harmonicity and breathiness to the resynthesized sound.

With the set of 65 features previously used, it rapidly appears that there is a correlation (and so redundancies) between features that can be seen as slight variations of control values. In this case, the redundancies offer extra refinement to adaptive control curves. Redundancies can also appear, such as too-similar features that will lengthen the list of features proposed to the user and then reduce the readibility of the GUI. In that case, the user may want to reduce the size of the mapping set. A solution consists of using Principal Component Analysis (PCA) (Jolliffe, 1986), also known as the Karlhunen-Loeve transform. It has been widely used, and for instance in spectral data reduction (Rochebois, 1997). Using PCA, the user can decide how many components the reduced set should contain. Moreover, the components are independent dimensions, so no correlations exist between the components, as depicted in Figure 9. The main drawback of PCA is that new 'features' may be difficult to understand, since they are linear combinations of features from the feature set, and do not always have a signal or perceptive sense. In the example given in Figure 10, we computed various "spectral centroid"-like measures from an excerpt of Pierre Schaeffer's voice (Schaeffer & Reibel, 1998, CD1, track 3): spectral centroid on spectrum, energy spectrum, with Beauchamp's correction computation (Beauchamp, 1982) for $c_0 = 1$, $c_0 = 10$ and $c_0 = 100$, low-high frequency balance and zero-crossing. See Appendix A for details about feature computation. As we can see in Figure 11, the first two components represent 73.92% and 22.76% of the features, respectively. There is a correlation between the second component and the energy by RMS (cf. upper left feature of Figure 10). This means that centroid is

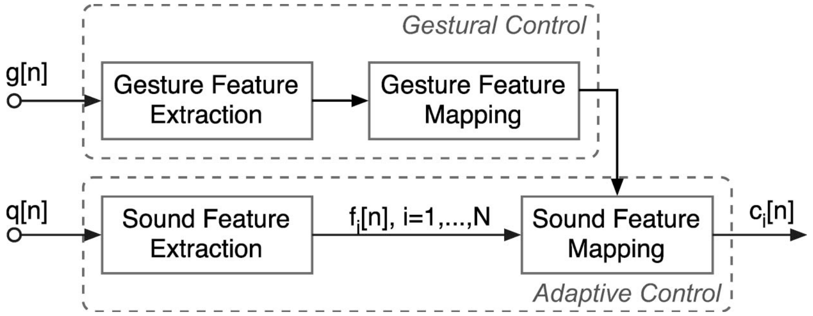

Fig. 7. Modular mapping strategy between sound/gesture features and effect controls that separates adaptive and gestural controls.

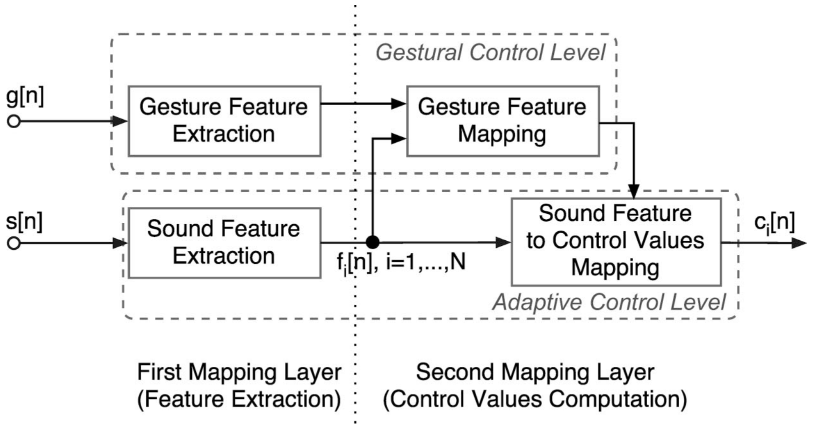
Gestural Control Level
g[n]
Gesture Feature Extraction
Gesture Feature Mapping
s[n]
Sound Feature Extraction
$f_i[n]$, i=1,...,N
Sound Feature to Control Values Mapping
$c_i[n]$
Adaptive Control Level
First Mapping Layer (Feature Extraction)
Second Mapping Layer (Control Values Computation)

Fig. 8. Mapping strategy between sound/gesture features and effect controls that allows sound features to modify gesture feature combination.

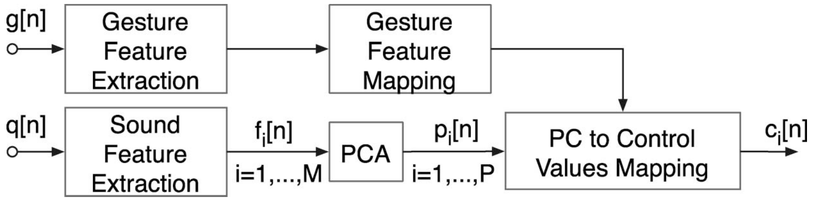

Fig. 9. Diagram of the mapping between feature and effect controls using principle component analysis for reducing the set of features.

represented by the first component for very low-level signals with big signal-to-noise ratios, and the second component for the signals with higher sound levels. Beauchamp's definition of centroid (Beauchamp, 1982) helps the user to choose between both centroid behaviours.

3.2 A multi-layer mapping strategy for gestural and adaptive control of DAFx

In previous works, the first author and colleagues proposed a mapping strategy derived from the three-layer mapping that uses a perceptive layer (Verfaille & Arfib, 2002; Verfaille, 2003; Verfaille et al., 2006). We now give more details about the structure of this mapping strategy, as well as examples of musical use.

The effect and its mapping between features and controls (level 1) can alternatively be modified by gestural control (level 2). To convert sound features $f_i(n)$, $i = 1, \ldots, M$ into effect control parameters $c_j(n)$, $j = 1, \ldots, N$, we use an M-to-N explicit mapping scheme divided into two layers: sound feature combination and control signal conditioning (see Figure 12). M is the

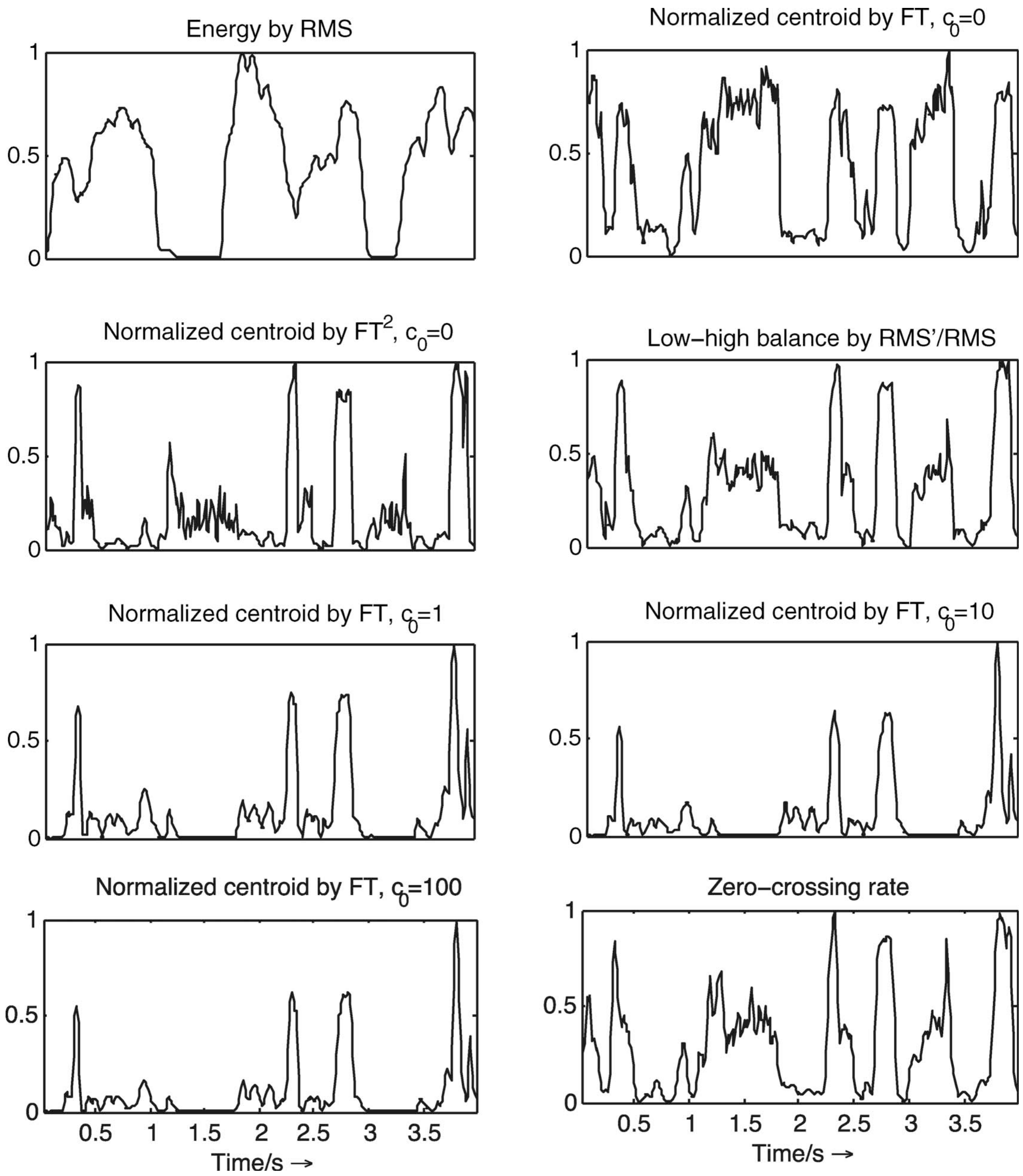

Fig. 10. Energy by RMS, and feature set for PCA: spectral centroid computed with various formulae and correlated features (low-high frequency balance and zero-crossing rate).

number of features we use, usually between 1 and 5; N is the number of effect control parameters, usually between 1 and 20. Details will be given in Section 3.3. Sound features frequently vary rapidly and with a constant sampling rate (synchronous data), whereas gestural controls used in sound synthesis vary less frequently and sometimes in a discontinuous and asynchronous mode. For that reason, we chose sound features for direct control of the effect and optional gestural control for modifications of the mapping between sound features and effect control parameters (Verfaille, 2003), thus providing navigation by interpolation between presets.

3.3 First level mapping between sound features and DAFx controls (adaptive control)

We now detail the mapping layers and sublayers of the first mapping level, which only deals with adaptive control. It can be divided into two main layers: a feature combination layer (four sublayers) and a signal conditioning layer (three sublayers).

3.3.1 Explicit mapping for feature combination

The first mapping layer of level 1 (adaptive control) deals with the sound feature combination. As depicted in

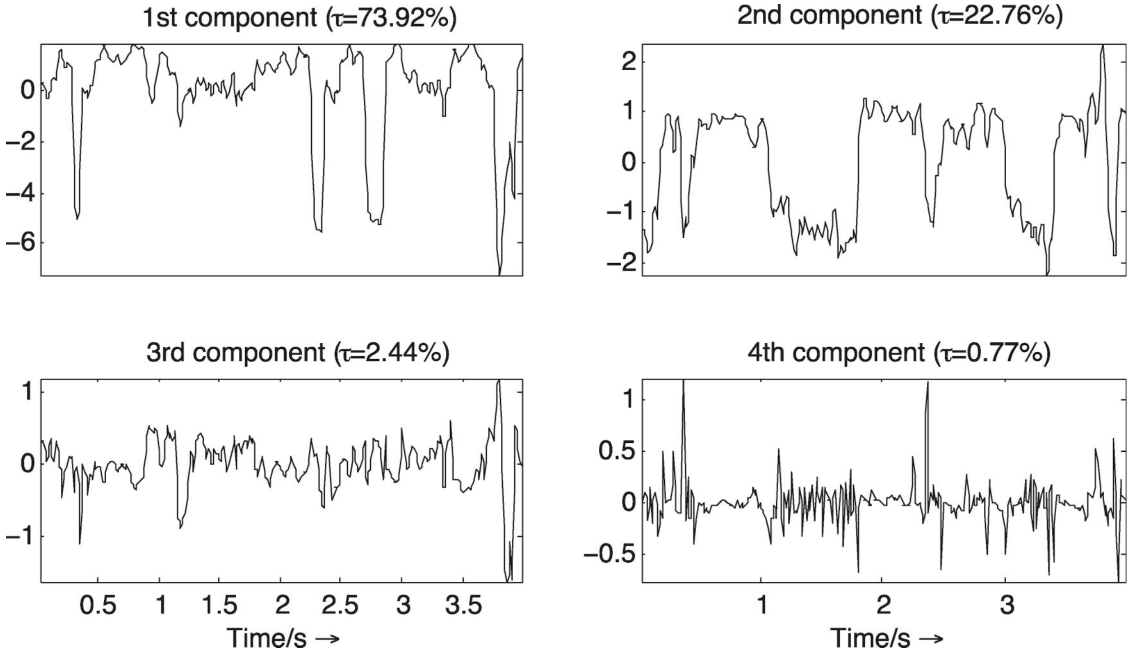

Fig. 11. Principal components from the PCA analysis of the feature set depicted in Figure 10.

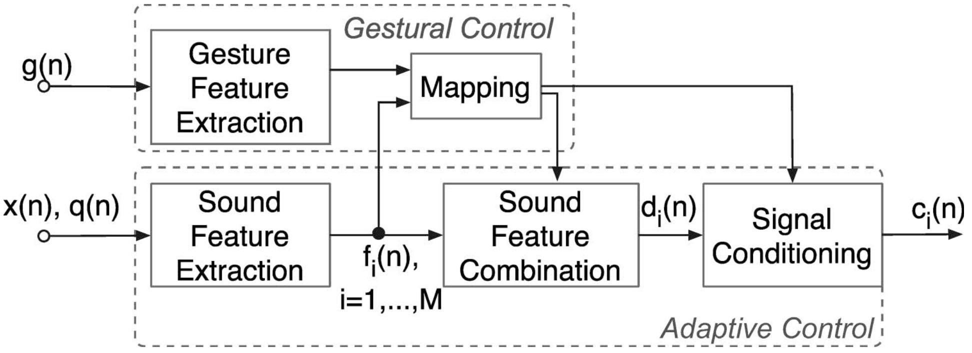

Fig. 12. Diagram of the mapping between sound features and one effect control $c_i(n)$. Sound features are first combined, and then conditioned in order to provide a valid effect control.

Figure 13, it consists of combining features after a normalization sublayer and using warping functions to modify the features behaviour before and after combination.

First, all the features are normalized in order to have the same variation range. This step can be considered as signal conditioning, even though it is applied in the feature combination mapping layer. This normalization step is also called "scaling" in other works (Todoroff, 2002), but we prefer using "scaling" for the last step, where abstract controls are mapped to the effect control bounds. Let us consider $f_k[n]$, $k = 1, \ldots, M$, the M features we want to normalize. We note $f_k^M = \max_{n\in[1;N]} f_k[n]$ and $f_k^m = \min_{n\in[1;N]} f_k[n]$ the feature extrema. We consider two feature normalization functions. The first normalization $\mathcal{N}_1[f_k, n]$ sets the features range to [0,1] by shifting and scaling:

$$\mathcal{N}_1[f_k, n] = \frac{f_k[n] - f_k^m}{f_k^M - f_k^m}. \qquad (1)$$

This normalization ensures that any unsigned parameter will reach its new bounds in the interval [0,1]. The second normalization $\mathcal{N}_2[f_k, n]$ divides all feature values by the maximum absolute value:

$$\mathcal{N}_2[f_k, n] = \frac{f_k[n]}{\max_{n\in[1;N]} |f_k[n]|}. \qquad (2)$$

Any signed parameter keeps its sign, it however does not ensure that the normalized feature will reach its two bounds except in the case $|f_k^m| = f_k^M$.

In the second mapping sublayer, a warping function is used either to change the variation scale of a parameter: e.g. from linear to exponential, or to logarithmic (useful for transforming a linear control into a frequency, or a magnitude in dB), or to impose a specific non-linear behaviour to a curve: inversion, clipping, soft-clipping, a truncation of the feature in order to select an interesting part, a low pass filtering, a monotonous or a non-monotonous variation. The normalized feature is then mapped into the same interval ([0,1] or $[-1,1]$). Parameters of the warping function can also be derived from sound features, for instance truncation boundaries (Verfaille, 2003). The choice of warping functions that are perceptively interesting can only be done thanks to knowledge of psychoacoustics (e.g. using a linear to exponential warping function to provide a fundamental frequency control), or by experimentation, since it concerns aesthetics and not mathematics. Then, the user can get used to the effect of a warping curve onto the adaptive control of an effect. Examples of warping functions are given in Figure 14 and in Appendix B.

The third mapping sublayer combines several features to provide one control parameter. This corresponds to an M-to-1 mapping, and the set of features combined differs for each control parameter. Combination can be non-linear, applied by multiplying warped features $\mathcal{J}[k,n]$

$$\mathcal{L}_m[n] = \prod_{k=1}^{M} a_k \mathcal{J}[k,n] \tag{3}$$

or a linear combination of warped features

$$\mathcal{L}_a[n] = \sum_{k=1}^{M} \frac{a_k}{\sum_{k=1}^{M} a_k} \mathcal{J}[k,n] \tag{4}$$

with $a_k \in [-1,1]$ the kth feature weight which can be given by other features. Since $\mathcal{J}[k,n] \in [-1,1]\ \forall n$, both combinations are normalized: $\mathcal{L}_a[n] \in [-1,1]\ \forall n$ and $\mathcal{L}_m[n] \in [-1,1]\ \forall n$.

In the fourth and last mapping sublayer of the feature combination, another warping function is applied to the feature combination output and symmetrically provides feature modifications before and after combination.

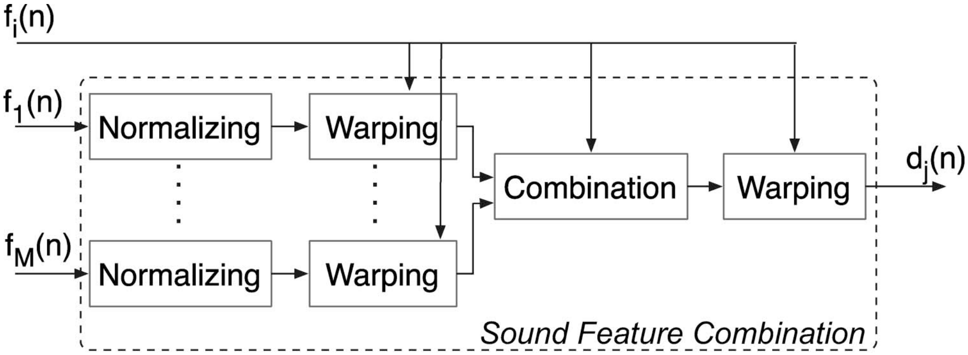

Fig. 13. Diagram of the first layer of sound feature mapping, performing sound feature combination. $f_i(n)$, $i = 1, \ldots, M$ are the sound features, and $d_j(n)$, $j = 1, \ldots, N$ are the combined features.

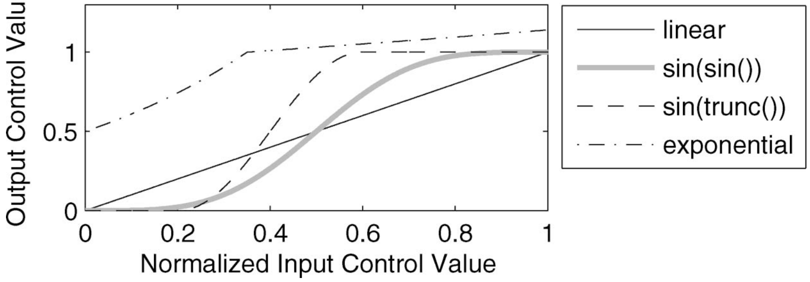

Fig. 14. Four examples of warping functions: (i) linear, (ii) sine wave, that increases the proximity to 0 or to 1 of the control value, (iii) truncation, e.g. between $t_m = 0.2$ and $t_M = 0.6$, that allows one to focus on an interesting range of the control curve, (iv) exponential for time-scaling, contracted from $s_m = 1/4$ to 1 for a control value between 0 and $\alpha = 0.35$, and dilated from 1 to $s_M = 2$ for a control value between α and 1.

3.3.2 Control signal conditioning

The second mapping layer of the adaptive control conditions the control signal $d_i(n)$ coming out of the feature combination box, as shown in Figure 15. Conditioning a signal consists of modifying it so that it fits to prerequisite behaviour in terms of boundaries and bandwidth. This mapping layer is made of three sublayers: effect-specific warping, low pass filter and scaling. There is no special order for the specific control curve warping sublayer (scaling or zooming, discretizing or quantifying), and for the lowpass filter sublayer. Fitting the control bounds is always the last sublayer.

The effect-specific warping sublayer is necessary to take into account the specific behaviour of each effect control. Let us note $\mathcal{W}$ the specific warping function defined as

$$\mathcal{W}[n] = \mathcal{W}_{\text{type}}(\mathcal{C}[n]) \tag{5}$$

and $d_i[n]$ the first mapping layer output parameter. We now develop three kinds of warping functions: time-warping, scaling and discretization. Time-warping a control signal (second sublayer) is useful for desynchronizing a control and the sound it is extracted from (Verfaille, 2003). It can also be used for time-scaling the sound signal, as the synthesis time index curve. We also used it in order to repeat a control curve, when extracted from a short sound and controlling the effect applied on a longer sound (Verfaille, 2003). A dynamic time warping is useful to provide hysteresis effects on a control curve, as usual in the compressor mapping. In the case where a curve has sudden range changes, it provides a focus on a reduced control value range (Verfaille, 2003). It is also useful when the gestural control becomes confined in a small area that the user explores with a gesture transducer. Examples of scaling functions are given in Appendix C, using low-pass filters, local minimum and maximum, and exponential weighting.

The discretization sublayer is useful for any effect control with a limited number of values, such as harmonizing and adaptive granular delay. A harmonizer is a pitch-shifter with discrete pitch-shifting ratios that provides chords (with respect to a harmonic context) in a tempered scale. Adaptive granular delay has a limited number of delay lines, with different length and feedback gain. Then the two control signals have to be discretized to a limited number of values in order to route each grain to the corresponding delay line (Verfaille & Arfib, 2001; Verfaille, 2003). Generally speaking, low-pass filtering is useless when discretizating. Several types of discretization can be used (Verfaille & Arfib, 2002; Verfaille, 2003), such as uniform discretization, eventually logarithmically or exponentially mapped, or non-uniform discretization for offline applications with control signals analysed before processing. Such discretization uses histograms and information about extrema in order to preserve some of the behaviour of the control signal after discretization (Verfaille, 2003).

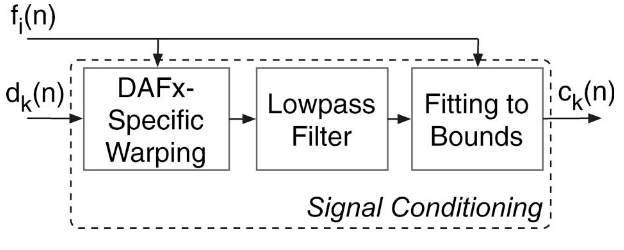

Fig. 15. Diagram of the second layer of sound feature mapping: control signal conditioning, applying specific mappings and fitting to control parameter bounds. $c_i(n)$, $n = 1, \ldots, N$ are the effect controls derived from sound features $f_i(n)$, $i = 1, \ldots, M$. The DAFx-specific warping and the fitting to boundaries can be controlled by other sound features.

Low-pass filtering a control signal ensures its suitability for the selected DAFx. The interests of using a low-pass filter sublayer is outlined with the two following examples: amplitude modulation effect (adaptive tremolo), in order to avoid clicks due to irregularities of the control signal (e.g. due to truncation in the first mapping layer); adaptive panning, where motion speed should be under 12 Hz if one wants it to be perceived as panning (and not a dual amplitude modulation in phase opposition). It is better to lowpass filter before the bound fitting sublayer in order to be sure the control reaches the bounds given by the user, since lowpass filtering may change slightly the extrema of the curves.

Third and lastly, the control signal is scaled to the effect control boundaries given by the user, which are eventually adaptively controlled. We fit the mapped curve between the minimum Δ_m and maximum Δ_M values of the effect control parameter. When necessary the control signal, sampled at the block rate F_B, is resampled at the audio sampling rate F_A. The effect control is given by:

$$\mathcal{C}_{fx}[n] = \Delta_m + (\Delta_M - \Delta_m)\ \mathcal{W}_i(\mathcal{L}[n]). \tag{6}$$

The first two examples of Section 4 use our mapping structure to provide simple adaptive control of audio effects. They do not use all the mapping sublayers for sake of simplicity.

3.4 Second level mapping between gesture features and the first level mapping (gestural control)

The second level of mapping is much simpler than the first level. It consists of routing gesture features to parameters of the sound feature mapping. Indeed, we consider that adaptive effects provide meaningful perceptual modifications of sound. The use of perceptually relevant features (e.g. high-level features describing the auditive perception) simplifies the whole mapping, by using the perceptual layer. Then, the gestural control can be extremely simplified, e.g. using 1-to-1 connections between a gestural control and any control of the first mapping level (adaptive control). Depending on which entry point is used, the gestural control will be low-level or high-level control. We also consider the case where no adaptive layer is used, in order to encompass all existing effects. Examples of low-level gestural control are:

- a direct control of an effect parameter;
- a control over non-musical mapping parameters.

A direct mapping between gesture parameters and effect control is basic and corresponds to the usual case where MIDI controllers (knobs, potentiometers) are used as effects controls (as well as synthesis controls).

When gesturally controlling an adaptive effect, higher-level control is provided by modifications at a local level on the adaptive control, by:

- warping a control signal (layer 1, sublayer 2 or 4);
- weighting linear combination (layer 1, sublayer 3);
- modifying bounds (layer 2, sublayer 3);

or at a global level, by:

- interpolating between two presets (continuous control: modification gesture), as in GRM Tools (Favreau, 2001);
- extrapolating from presets (continuous control: modification gesture);
- changing the preset (discrete control: decision gesture).

A control upon the mapping corresponds to a high level of control. A basic example is the vibrato effect: to apply a vibrato to a flat sound, one can control directly the pitch (low level control) or control the vibrato parameters (rate and depth) as higher-level parameters (Hunt et al., 2002; Arfib & Verfaille, 2003). This control upon the mapping also permits navigation between effects: e.g. with delay line modulations, changing between a chorus, a flanger and a phaser. This high-level control also allows navigation among presets. A first example is with an adaptive granular delay, morphing from "keep the transient" to "keep the steady parts" preset (Verfaille et al., 2006). A second example is the gestural control of adaptive time-scaling curves: several time-scaling ratio curves are pre-computed (e.g. a curve only slowing down vowels, another speeding up consonants, a third one slowing down low pitches and speeding up high pitches), and the user gesturally interpolates between these curves (Arfib & Verfaille, 2003). A third example using adaptive spatialization consists in changing the warping function to make the sound position change only during silence, or to make it stop moving during silence (see Section 4.3).

4. Examples of musical use

Six musical examples using our mapping structure are discussed. The first two only concern adaptive control (robotization and granular delay), whereas the other four concern both adaptive and gestural controls (spatialization, prosody change, equalizer and spectral tremolo).

4.1 Adaptive robotization

Consider the simple case where the user wants to control the pitch of a robot, using adaptive robotization driven by the energy of a voice sound. The control value that has to be set is the fundamental frequency $f_0 \in [100,200]$ Hz, and the algorithm is controlled via the varying hop size $R_a = F_{sr}/f_0$ with F_{sr} the sampling rate (Verfaille & Arfib, 2001). A sound feature is extracted, for example the energy using the RMS. We focus on the combination layer first. This feature varies in [0, 1], so the normalization is useless. Note that when the feature is nul, no processing has to be done. The first warping function can be useless (and replaced by $y(n) = x(n)$), as well as the combination, since only one feature is used. Finally, we use an inversion as the second warping function: $y(n) = (\min\ x(n))/x(n) \in]0,1]$. Concerning the signal conditioning layer, low-pass filtering is useless, the only necessary mapping step is fitting R_A to the bounds $R_a \in [F_{sr}/\max(f_0),\ F_{sr}/\min(f_0)] = [220.5,441]$ samples.

4.2 Adaptive granular delay

Granular delay is a set of delay lines with various length and gains. The sound is decomposed in grains, which are sent to one of the delay lines (Verfaille & Arfib, 2002; Verfaille, 2003). Consider the case where only the gains of delay lines differ from one line to another. When the feature that drives the effect concerns the attack detection and/or the brightness of the grain, then this effect provides an echo with attack morphing, where the attack dissolves rapidly with repetitions, or inversely where only the attack does not dissolve with repetitions. A non-real-time adaptive granular delay can implement any number of delay length and gains, using fractional delays and overlap-adding the delayed grains in the output sound file. However, a real-time implementation of adaptive granular delay is limited in the number of delay lines, implying to discretize the control values. Extracting the spectral centroid feature (that varies in $[0,F_{sr}/2]$), we first normalize it in [0,1], and a logarithmic warping is applied to make it vary more linearly. No combination nor second warping function are required. Then, the control is quantized, in order to provide a limited number of values (Verfaille & Arfib, 2002). These quantized values are then mapped in the delay gain range [0,1[.

4.3 Adaptive spatialization

The first author designed a series of Max/MSP patches for gestural control of adaptive spatialization during a collaboration with Anne Sédès (Paris VIII University) (Sédès et al., 2003). A dancer controls parameters of a sound production system using granular synthesis, his/her position being captured by ultra-sound beam sensors. Among spatialization control parameters (distance from listener, elevation, azimuth, source distance to room walls, etc.), we focused on azimuth and distance, using a multi-channel system (8 loudspeakers) and vector base amplitude panning (Pulkki, 1997).

Sound is synthesized using granular synthesis, delay-lines and filters: a first mapping level defines how gestural control on the vertical ultra-beam sensor affects synthesis. Vertical position corresponds to grain position in sound, and speed and acceleration are mapped to pitch-shifting of grain as well as delay-line lengths, inducing comb filtering and delay effects. A minimal set of features were extracted: energy (by RMS) and spectral centroid. Those two features are then linearly combined and warped (feature combination layer). The parameter obtained can be mapped to sound position, speed or acceleration upon a given trajectory (allowing for non-linear motion upon this trajectory), to trajectory shape modification, or to both (see mapping in Figure 16).

Gestural and adaptive controls were assigned specific roles as follows: gesture controls feature combination, trajectory shape modification (by warping and shifting), and motion type change (position, speed, acceleration); sound features control sound position/speed/acceleration onto the gesturally modified curve, and trajectory shape modifications.

We defined with the dancer six main presets or configurations: automatic motion, gesture positioning, motion arc length, transfer function, source width, and trajectory.

A. *Automatic motion:* sound position x, speed v or acceleration a upon a trajectory $\mathcal{T}$, is given by a sound feature. For example, using a quasi-rectangular window transfer function as depicted in Figure 17, sound moves on a circular trajectory only during silences (low RMS), or only during loud sounds (high RMS).

B. *Gestural positioning:* gesture directly controls sound position according to dancer position using the horizontal ultra-beam sensor and with rules such as point or axis symmetry, shifting and scaling.

C. *Transfer function:* shifting the quasi-rectangular transfer function (cf. Figure 17) along the x axis allows for changing the feature ‘‘interest portion’’, whereas shifting along the y axis allows for changing the maximum motion speed. As with more complex modifications of the transfer function, this gestural control of the adaptive control mapping layer allows for interpolating between specific behaviours of the effect.

D. *Source width:* sound is spectrally split into 8 frequency bands, each one being adaptively positioned in space. If the mapping between frequency band number and position is constant, the resulting effect is a spatial opening of sound when its spectrum enriches. If this mapping varies with time, the resulting effect is more complex to explain, changing from spatializing to surround and wah-wah-like filtering.

E. *Trajectory*: Sound adaptively moves in position or speed upon an ellipse defined by:

$$\left(\frac{x}{a}\right)^2 + \left(\frac{y}{b}\right)^2 = 1. \qquad (7)$$

Gestural control warps the trajectory in various ways: shifting x, or y, or both directions, or changing one or both axes length a and b, in order to uniformly transform a circle trajectory into an ellipse trajectory and then a segment trajectory, as depicted in Figure 18.

F. *Motion arc length:* a trajectory $\mathcal{T}$ is defined as $(x,y) = (x(t),\ y(t))$, its shape being constant or

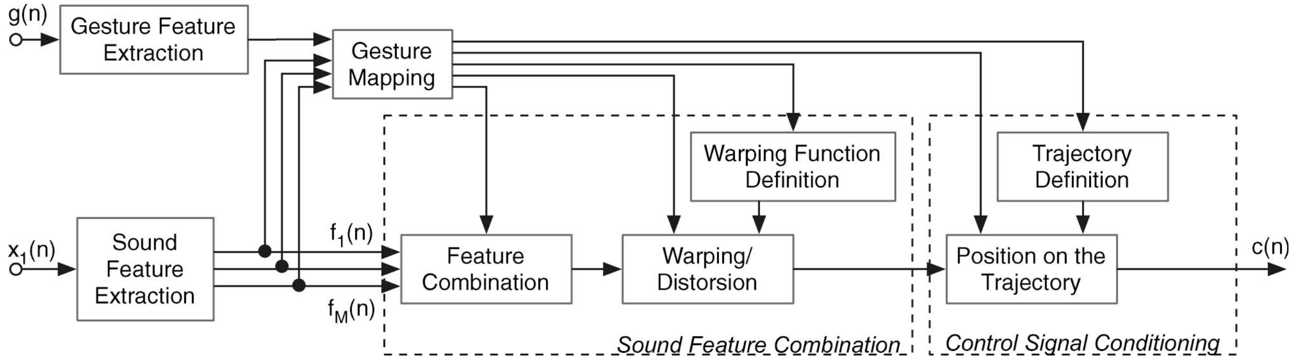
g(n)
Gesture Feature Extraction
Gesture Mapping
Warping Function Definition
Trajectory Definition
x_1(n)
Sound Feature Extraction
f_1(n)
f_M(n)
Feature Combination
Warping/ Distorsion
Position on the Trajectory
c(n)
Sound Feature Combination
Control Signal Conditioning

Fig. 16. Diagram of gesturally controlled adaptive spatialization.

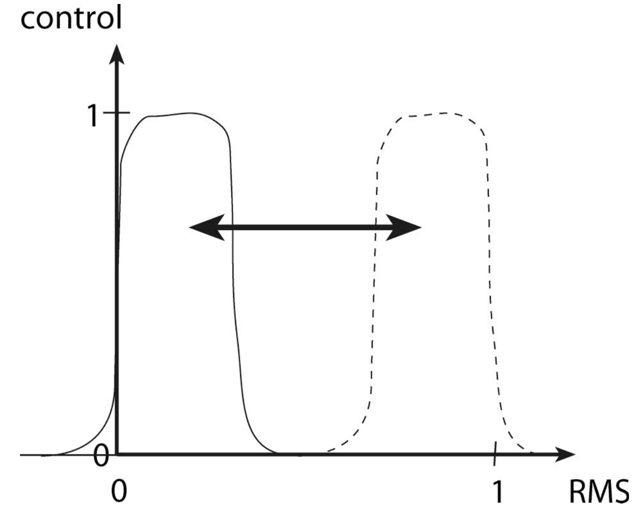

Fig. 17. Using a quasi-rectangular transfer function, the sound source moves only during silences (left) or during loud sounds (right), using the energy by RMS as adaptive control.

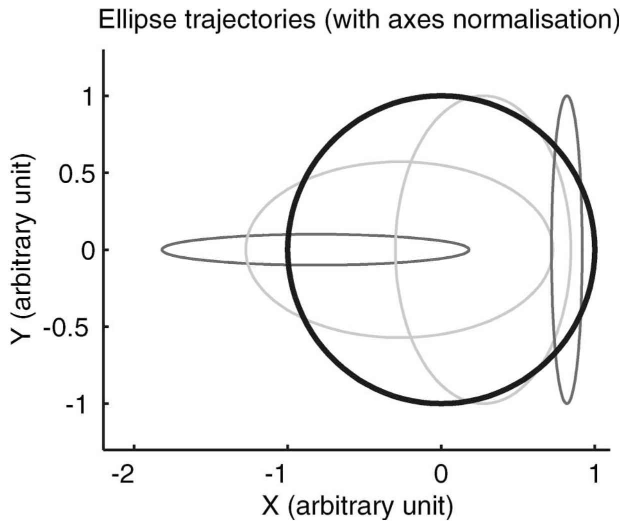

Fig. 18. Examples of ellipse trajectory warping by gestural control: ellipse axes length are changed, thus allowing for uniformly transforming a circle into a segment, using an ellipse with normalized axes.

controlled by gestures. Sound source is positioned upon a small arc of the whole trajectory, with variable length $[t^{-};t^{+}]$ depending on the sound energy. This control provides a localized and limited spatial area within which sound moves, as when a performer has a specific position on stage, and moves by small amounts when playing. With a small arc length, sound is heard as a unique local source. On the other hand, with a big arc length (half a circle, a whole circle), one sound seems multi-localized. Since adaptive sound motions are too rapid, they are not perceived as physical motion. Instead, several virtual sound sources appear. Moreover, motion is not sinusoidal, so amplitude modulation does not correspond to a ring modulation. The sound is segmented into several auditive streams. For example, if RMS is mapped to motion, parts of the sound with the same RMS have the same position. Another example: if centroid is mapped to motion, sound time segments with similar centroid values have the same position. Then, the fact that two dimensions of the sound—either localization and energy, or localization and spectrum via centroid—are very coherent helps to segregate the sounds (Bregman, 1990).

4.4 Prosody change

The prosody of a spoken voice is defined by its intonation (pitch and loudness) and its time duration. Modifying prosody then requires subtle modulations of pitch, loudness and time unfolding (Verfaille, 2003; Verfaille et al., 2006).

An intonation modification effect can be designed by combining pitch-shifting and gain applied to a specific fundamental frequency decomposition. Adaptive time-scaling and intonation modification are basic components of a prosody modification effect. The fundamental frequency $F_{0,\mathrm{in}}(m)$ is decomposed into a micro-intonation $\Delta F_{0,\mathrm{in}}(m)$ and a macro-intonation $\overline{F^{loc}_{0,\mathrm{in}}}$:

$$F_{0,\mathrm{in}}(m) = \overline{F^{loc}_{0,\mathrm{in}}} + \Delta F_{0,\mathrm{in}}(m), \tag{8}$$

which can then be arbitrarily recombined (Arfib & Verfaille, 2003), e.g. using $\overline{F_{0,\mathrm{in}}}$ the mean of $F_{0,\mathrm{in}}$ over the whole signal length:

$$F_{0,\mathrm{out}}(m) = \gamma\overline{F_{0,\mathrm{in}}} + \alpha\left(\overline{F^{loc}_{0,\mathrm{in}}} - \overline{F_{0,\mathrm{in}}}\right) + \beta\Delta F_{0,\mathrm{in}}(m). \tag{9}$$

The mapping layer between sound features and intonation change provides a one-to-one feature transformation into coefficients of Equation (9). Provided a formant analysis, the two first formant frequencies can control α and β after scaling. The mapping layer between sound features and time-scaling uses scaling and warping functions to transform a linearly varying combination of features into an exponentially varying time-scaling ratio (Verfaille & Arfib, 2001).

Gestural control is provided on:

- pitch-shifting: mean $\overline{F_{0,\mathrm{in}}}$ and $F_{0,\mathrm{out}}$ variation range;
- time-scaling ratio variation range;
- gain variation range;

and adaptive control on:

- pitch-shifting ratio: pitch can be flattened, enhanced or inversed, at a micro or macro level;
- time-scaling ratio around 1, e.g. in [0.8,1.25]; voice can be made slower when louder, faster when louder, slower when silent, faster when silent, etc.

By controlling intonation change, adaptive time-scaling and adaptive intensity change, this combination of effects allows for prosody changes. With adaptive control, pitch contour can be replaced by another feature or feature set contour. Loudness contour can be modified to emphasize word, syllables, consonants or vowels. Time folding can be modified in a meaningful way, changing the emotion behind the words: voice becomes more stressed (faster when silent), or relaxed (slower when silent). This first step in a prosody change effect has a major drawback: it does not give clues for prosody change with meaning, that is to say the user cannot specify a given emotion that the system would use. To be able to do so, further work is needed.

4.5 Adaptive equalizer

This effect is an equalizer with a slow time-varying equalizing curve derived from a vector feature (Verfaille et al., 2006). Depending on the mapping used, it provides not only equalizing, but also spectral panning. Based on the FFT, the adaptive equalizer uses a vector feature. In order to avoid timbre modulation due to amplitude modulation of each frequency bin when multiplying by the envelope transfer function, a time-smoothing of the feature vector is needed. The vector feature has the same length as the FFT. It is scaled to the [0,1] range (or $[-1,1]$ if one wants to allow for phase opposition), and under-sampled in time in order to avoid timbre modulation. The refresh rate allows one to control the under-sampling ratio, or interpolation time between two different filter FFT.

Gestural control is provided on:

- the mean, min and max values of the filter curve (mean given by y and width of the interval by x of a joystick);
- interpolation time using the twist of the joystick;
- interpolation type (complete, i.e. with no attack, or uncomplete, i.e. with attacks) with a slider, using the mouse;

whereas adaptive control only concerns the filter curve, given by features such as the wave form or the spectral envelope.

When applying the adaptive equalizer to a stereophonic sound, the user can apply various strategies. A first strategy consists in using the same control signal for both channels, e.g. computed from features of both channels. A second strategy consists in using two control signals provided by features from only one channel, for instance from the processed channel or from the opposite channel. A third strategy that can be considered as a specific preset—or even another effect—consists in using "constant power" complementary curves. This provides adaptive spectral panning. When applying the adaptive equalizer to a pink noise and extracting features from a musical sound (external adaptive equalizer), one obtains a cross-synthesis with controls such as the interpolation time and the smoothness of attacks. This was used as an instrument under the name *Noisonic* to create electronic sounds used in *Flute Salad*, a piece for tape, flute and adaptive effects presented by the first author during concerts of the *Tutti Quanti Computing Orchestra* (Verfaille, 2003).

The gesturally controlled adaptive equalizer allows for timbre modulation, filtering with or without discontinuities, spectral envelope and time amplitude envelope shaping of musical sounds and noises. The spectral panning allows for a kind of stream segregation in timbre (Verfaille, 2003; Verfaille et al., 2006). The control over the time amplitude envelope is given by the interpolation time and can be very accurate. Conversely, the control upon the spectral envelope is indirect, since one does not control the shape itself, but the bounds and the mean value. The spectral envelope is adaptively imposed to the user, but he/she controls modulation intensity and speed.

4.6 Adaptive spectral tremolo

Spectral tremoli consists of applying a tremolo to each frequency bin of the STFT, with a frequency $f_{trem}[k]$ depending on the frequency bin k. When adaptively controlled, frequency is, for example, given by the spectral envelope. Noting $E[m,k] = 1 + \mathrm{d[m]} \quad \sin 2\pi f_{trem}[m,k]$ the spectral amplitude modulation, we derive the output STFT as:

$$F_y = E \cdot F_x. \tag{10}$$

Two vector sound features can be mapped to amplitude and frequency of tremoli frequency bins. Spectral envelope is a curve that suits well to such an adaptive control. Then, an interesting mapping consists in transforming this envelope into a monotonous curve by cumulative sum. Once normalized in [0,1], it can be mapped to the frequency bin frequencies (tremoli rates). The spectral envelope directly mapped in [0,1] gives the amplitude for each frequency bin (tremoli depth).

Gestures can control the bound fitting sublayer applied to the rates, whereas sound features adaptively control depths and rates of spectral tremoli. That way, the user can globally accelerate or deccelerate the tremoli and phasing aspect of the effect, or force tremoli to be in or out of phase.

An example of preset is given with tremoli rates in [0,4] Hz and sinusoidal amplitude modulation (Verfaille, 2003). Other amplitude modulation curves were used, such as sine in dB scale, or inversely log (sin()) in linear scale. To provide a better phasing effect, the tremoli depth has to be kept to their maximal value. To provide a better tremolo effect, the tremoli rates have to be controlled by a constant value for all frequency bins. Since each frequency bin has a different tremolo, two perceptual effects co-exist: a tremolo when tremoli are in phase and a phasing effect when out of phase. Then, depending on the control curve used, the effect oscillates between phasing and tremolo, depending on the adaptive control curve shape and slope (see Figure 19).

5. Musical implications of our mapping model

We develop implications of the proposed mapping structure, using the theoretical information given in Section 2 and 3, and the examples of gestural and adaptive control given in Section 4.

5.1 A clear separation between sound features and gestures

From the thorough observation of usual DAFx control and mapping structure, we defined the two layer mapping structure proposed in Section 3.1. This structure generalizes the structure of any effect, whatever its control is: with only adaptive control when removing the gestural control level, with only gestural control when removing the adaptive control, or with both controls. This clear separation between the mapping of sound features and gestural control helps the user to identify the mapping components. Such a modular aspect simplifies the mapping design and offers a high level of control, as shown with the four musical examples (Section 4).

5.2 Interpolation

Interpolation is an important feature of mapping in sound synthesis (Goudeseune, 2002), providing ways to navigate in the sound database. Interpolation has also been used to define ways of manipulating control curves and presets (Allouis, 1984; Terrugi, 2001). This navigation between presets is an important feature of creative

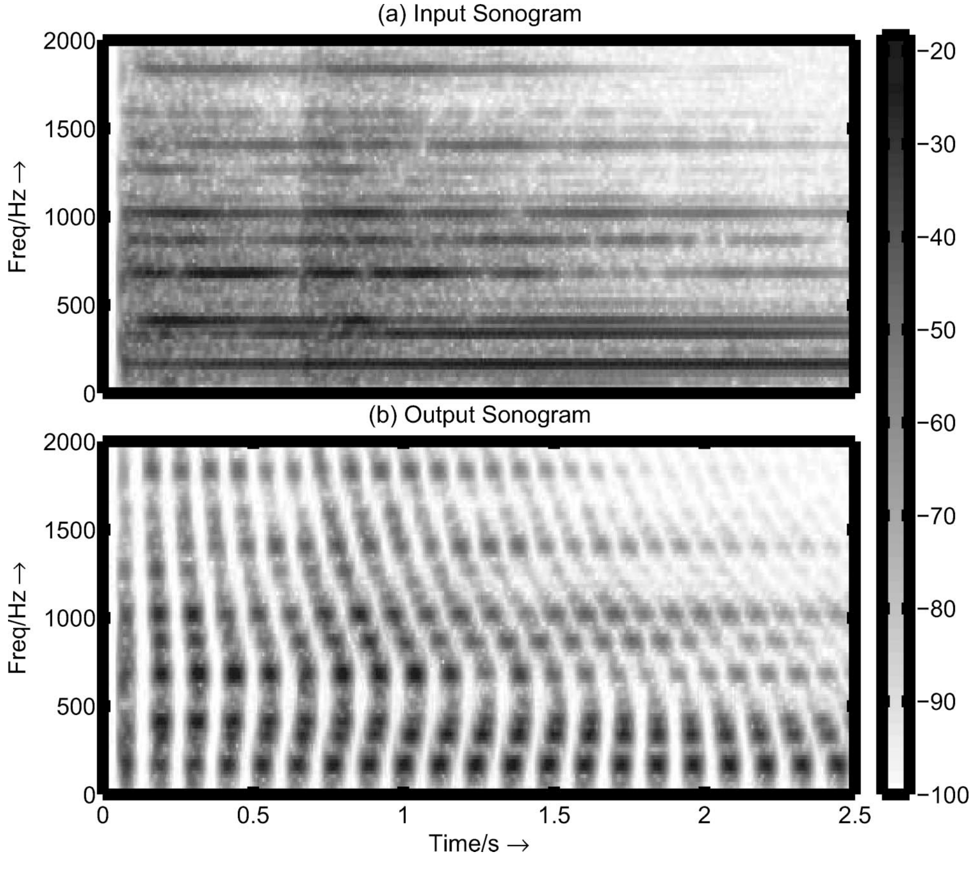

Fig. 19. Sonogram of a bell sound (from Varèse's "Poème Électronique"): a) before and b) after adaptive spectral tremolo: each frequency bin has its own tremolo frequency, thus inducing a tremolo effect when tremoli are in phase, and a phasing effect when tremoli are out of phase.

audio effects (Favreau, 2001), and has also been used for adaptive effects, e.g. from "keep the transient" to "keep the stable part of the sound" preset with an adaptive granular delay in Section 3.3. This can be generalized to gestural and adaptive control by using the proposed hierarchy: the gestural mapping controls the sound feature mapping (see Figure 12), thus permitting one to navigate between effects as well as between presets of an effect. An example of navigation between DAFx types (chorus, flanger and phaser) was given in Section 2.2. Another example in Section 4.6 uses the adaptive spectral tremolo to interpolate between flanger and tremolo.

5.3 The choice of sound features

When adding an adaptive control to an effect, one must choose sound features that will drive the effect. We claim that this choice is another musical aspect of the mapping, and therefore depends on musical goals. For instance, time-scaling instrumental sounds in order to slow down high pitch and speed up low pitch notes requires a pitch tracking algorithm (Verfaille & Arfib, 2001). Part of the sound feature choice is however constrained by the feature type. Indeed, sound features are either short-term or long-term features; therefore they may have different and well identified roles in the proposed mapping structure. By comparing sound feature types to gesture types, we can propose some hints for easier design of adaptive effect mappings. Short-term features, for instance energy, instantaneous pitch or loudness, voiciness, and spectral centroid provide a continuous adaptive control with a high control rate. We consider that these features affect the effect controls as modification gestures. They are useful as inputs, as represented by left horizontal arrows in Figures 13 and 15. Long-term features, for instance vibrato, roughness, duration, note pitch or loudness, are computed after signal segmentation. Often used for content-based transformations (Amatriain et al., 2003), they provide a sequential adaptive control with low rate. We consider that those features affect the effect controls as selection gestures. They are useful as mapping controls, as depicted by upper vertical arrows in Figures 13 and 15.

5.4 Perspectives

From the mapping strategies developed in the present work, further works could be carried out. First, there is no systematic investigation of mapping efficiency in the context of DAFx yet, whereas such studies exist in the context of sound synthesis (Hunt & Wanderley, 2002). Comparing users' ability to manipulate various types of adaptive and gestural control in order to perform a musical task will help to define other efficient mappings, and to highlight "playable" configurations of DAFx, sound feature and gestural controller.

Second, only explicit mappings were used. Using implicit mapping such as neural networks, one could derive a control parameter from sound features and an automation target curve. This would allow one to train the neural network not only to re-create the first seconds of an automation curve from a set of sound features, but also to use adaptive control as a way to expand automation control in time after training.

A third interesting area is the mapping for complex systems using several DAFx. In the examples discussed in this paper, only one effect was controlled at once, or at least only one technique that could lead to a small number of effects. In the same way as digital musical instruments are controlled by several parameters and modify several synthesis parameters, a spatializer offers controls on distance, position, motion, and reverberation: a many-to-many mapping for its gestural and adaptive control then would require more than just explicit feature combination and control signal conditioning.

Fourth, the output mapping leading to visual feedback, representation of sound features and input mapping is a great concern in DAFx control (Todoroff, 2002). We proposed some specific GUI in Verfaille (2003), because manipulating control signals as a curve makes more sense when a visual feedback helps one to appreciate how warping functions and combination modifies the control. Moreover, when mapping gets complex, the user's ability to manipulate the mapping depends on his/her knowledge about DAFx, techniques used and perception. Using presets and interpolation between presets is a successful method (Favreau, 2001). Another idea is to offer various accesses to mapping, according to the user's expertise: beginner (simple and efficient presets), intermediate (more subtle presets, with more mapping capabilities) and expert (any possible control). That kind of GUI allows for efficient access to control and configuration of presets. Then, any user could jump into the mapping system, learn from the presets, and slowly gain a deeper understanding of what can be done.

6. Conclusion

Adaptive effects may be difficult to control as far as lots of input and control parameters are given. A solution consists in developing specific mapping strategies. Using mapping techniques dedicated to sound synthesis, we proposed a mapping structure for easily controlling them, and also gave musical examples. This mapping structure is explicit and has two layers, namely feature combination and control signal conditioning, each made of sublayers. It offers a high level of control, by combining two mapping levels: the adaptive control level manipulated by the gestural control level. Interesting interactions with adaptively and gesturally controlled

DAFx includes interpolation between adaptive effects and between presets. The clear separation of the adaptive control level and the gestural control level provides a great modularity that clarifies mapping component roles. The proposed mapping strategy can not only be applied to define mapping in an adaptive scheme, but also to encompass any type of control of DAFx: LFO, gestural, automated, adaptive, algorithmic. Such a general framework provides a creative environment to electro-acoustic music composers, performers and sound engineers.

Acknowledgments

This work was supported by the *Laboratoire de Mécanique et d'Acoustique* of the *Centre National de la Recherche Scientifique* (LMA-CNRS, France), the *Région Provence Alpes-Côte-d'Azur* (PACA, France), and the *Fond Québécois de la Recherche sur la Nature et les Technologies* (FQRNT, Québec, Canada). Many thanks to Daniel Arfib for fruitful discussions.

References

Allouis, J.-F. (1984). *Logiciels pour le système temps réel syter*. Paper presented at the International Computer Music Conference (ICMC'84), Paris.

Amatriain, X., Bonada, J., Loscos, A., Arcos, J.L. & Verfaille, V. (2003). Content-based transformations. *Journal of New Music Research*, *32*(1), 95–114.

Arfib, D., Couturier, J.-M. & Kessous, L. (2002a). *Gestural strategies for specific filtering processes*. Paper presented at the International Conference on Digital Audio Effects (DAFx-02) (pp. 1–6), Hamburg, Germany.

Arfib, D., Couturier, J.-M., Kessous, L. & Verfaille, V. (2002b). Strategies of mapping between gesture parameters and synthesis model parameters using perceptual spaces, *Organised Sound* *7*(2), 135–152.

Arfib, D., Keiler, F. & Zölzer, U. (2002c). In U. Zoelzer (Ed.), *DAFX – Digital Audio Effects*. New York: J. Wiley & Sons, Chapter Source-Filter Processing, pp. 299–372.

Arfib, D. & Verfaille, V. (2003). *Driving pitch-shifting and time-scaling algorithms with adaptive and gestural techniques*. Paper presented at the International Conference on Digital Audio Effects (DAFx-03) (pp. 106–111), London, England.

Bailey, N.J., Purvis, A., Bowler, I.W. & Manning, P.D. (1993). Applications of the phase vocoder in the control of real-time electronic musical instruments. *Interface 22*, 259–275.

Beauchamp, J.W. (1982). Synthesis by spectral amplitude and "brightness" matching of analyzed musical instrument tones. *Journal of the Audio Engineering Society* *30*(6), 396–406.

Bevilacqua, F., Müller, R. & Schnell, N. (2005). *MnM: a Max/MSP mapping toolbox*. Paper presented at International Conference on New Interfaces for Musical Expression (pp. 85–88), Vancouver.

Bregman, A. (1990). *Auditory Scene Analysis*. Cambridge, MA: MIT Press.

Burtner, M. (2002). The metasaxophone: concept, implementation, and mapping strategies for a new computer music instrument. *Organised Sound*, *7*(2), 201–214.

Cadoz, C. & Wanderley, M.M. (2000). Gesture—music. In M.M. Wanderley & M. Battier (Eds), *Trends in Gestural Control of Music*. Paris (Centre Pompidou): IRCAM.

Chadabe, J. (2002). The limitations of mapping as a structural descriptive in electronics instruments. Paper presented at the International Conference on New Interfaces for Musical Expression (pp. 197–201), Dublin.

Depalle, P., Garcia, G. & Rodet, X. (1995). *Reconstruction of a castrato voice: Farinelli's voice*. Paper presented at the IEEE Workshop on Applications of Digital Signal Processing to Audio and Acoustics (pp. 242–245).

Doornbusch, P. (2002). Composers' views on mapping in algorithmic composition. *Organised Sound*, *7*(2), 145–156.

Dutilleux, P. & Zölzer, U. (2002). In U. Zoelzer (Ed.), *DAFX – Digital Audio Effects*. New York: J. Wiley & Sons, Chapter Nonlinear Processing, pp. 93–135.

Egozy, E.B. (1995). Deriving musical control features from a real-time timbre analysis of the clarinet. Master's thesis, Massachusetts Institut of Technology.

Favreau, E. (2001). *Phase vocoder applications in GRM tools environment*. Paper presented at the COST-G6 Workshop on Digital Audio Effects (DAFx-01) (pp. 134–137), Limerick, Ireland.

Fels, S. (1994). *Glove Talk II: Mapping Hand Gestures to Speech Using Neural Networks*. PhD thesis, University of Toronto.

Goudeseune, C. (2002). Interpolated mappings for musical instruments. *Organised Sound*, *7*(2), 85–96.

Hunt, A. & Wanderley, M.M. (2002). Mapping Performer Parameters to Synthesis Engines. *Organised Sound*, *7*(2), 97–108.

Hunt, A., Wanderley, M.M. & Paradis, M. (2002). The importance of parameter mapping in electronic instrument design. *Organised Sound*, *7*(2), 429–440.

Jolliffe, I.T. (1986). *Principal Component Analysis*. Berlin: Springer-Verlag.

Levitin, D.J., McAdams, S. & Adams, R.L. (2002). Control parameters for musical instruments: a foundation for new mappings of gesture to sound. *Organised Sound*, *7*(2), 171–190.

Marino, G., Serra, M.-H. & Raczinski, J.-M. (1993). The UPIC system: origins and innovations. *Perspectives of New Music*, *31*(1), 258–269.

Marrin-Nakra, T. (2000). Searching for meaning in gestural data: interpretive feature extraction and signal processing for affective and expressive content. In M.M. Wanderley & M. Battier (Eds), *Trends in Gestural Control of Music*. Paris (Centre Pompidou): IRCAM.

Métois, E. (1998). *Musical gestures and audio effects processing*. Paper presented at the COST-G6 Workshop on Digital Audio Effects (DAFx-98) (pp. 249–253), Barcelona, Spain.

Myatt, T. (2002). Strategies for interaction in *construction 3*. *Organised Sound*, *7*(2), 157–170.

Pulkki, V. (1997). Virtual sound source positioning using vector base amplitude panning. *Journal of the Audio Engineering Society*, *45*(6), 456–466.

Risset, J.-C. (2002). Examples of the musical use of digital audio effects. *Journal of New Music Research*, *31*(2), 93–97.

Rochebois, T. (1997). *Méthodes d'analyse/synthèse et représentations optimales des sons musicaux basées sur la réduction de données spectrales*. PhD thesis, University of Paris XI, France.

Ryan, J. (1992). *Effort and expression*. Paper presented at the International Computer Music Conference (ICMC92) (pp. 414–416), San Francisco.

Schaeffer, P. & Reibel, G. (1998). Solfège des Objets Sonores, Seuil/INA-GRM.

Sédès, A., Courribet, B., Thiébaut, J.-B. & Verfaille, V. (2003). Chapter Visualisation de l'Espace Sonore, vers la Notion de Transduction: une Approche Intéractive Temps-Réel. *Espaces Sonores – Actes de Recherches*. Paris: CICM-EMT, pp. 125–143.

Steiner, J.-C. (2005). *[hid] toolkit: a unified framework for instrument design*. Paper presented at International Conference on New Interfaces for Musical Expression (pp. 140–143), Vancouver.

Terrugi, D. (2001). *L'interpolateur, une interface de contrôle multi-paramétrique*. Paper presented at the Journées d'Informatique Musicale (JIM'01).

Todoroff, T. (2002). In U. Zoelzer (Ed.), *DAFX – Digital Audio Effects*. New York: J. Wiley & Sons, Chapter Control of Digital Audio Effects, pp. 465–497.

Traube, C. & Smith, J.O. (2000). Estimating the plucking point on a guitar string. In *Proceedings of the COST-G6 Workshop on Digital Audio Effects (DAFx-00)*, Verona, Italy.

Verfaille, V. (2003). *Effets Audionumériques Adaptatifs: Théorie, Mise en Œuvre et Usage en Création Musicale Numérique*. PhD thesis, Université de la Méditerranée (Aix-Marseille II).

Verfaille, V. & Arfib, D. (2001). *ADAFx: adaptive digital audio effects*. Paper presented at the COST-G6 Workshop on Digital Audio Effects (DAFx-01) (pp. 10–14), Limerick, Ireland.

Verfaille, V. & Arfib, D. (2002). *Implementation strategies for adaptive digital audio effects*. Paper presented at the International Conference on Digital Audio Effects (DAFx-02) (pp. 21–26), Hamburg, Germany.

Verfaille, V., Zölzer, U. & Arfib, D. (2006). Adaptive digital audio effects (A-DAFx): a new class of sound transformations. *IEEE Transactions on Audio, Speech and Language Processing*, *14*(5), in press.

Wanderley, M., Schnell, N. & Rovan, J.B. (1998). *Escher—modeling and performing composed instruments in real-time*. Paper presented at the IEEE International Conference on Systems, Man and Cybernetics (SMC'98) (pp. 1080–1084), San Diego.

Wanderley, M.M. (2002). Mapping strategies in real-time computer music. *Organised Sound*, *7*(2), 83–84.

Wanderley, M.M. & Depalle, P. (2000). *Gesturally controlled digital audio effects*. Paper presented at the COST-G6 Workshop on Digital Audio Effects (DAFx-00) (pp. 165–169), Verona, Italy.

Wanderley, M.M. & Depalle, P. (2004). Gestural control of sound synthesis. *Proceedings of the IEEE, Special Issue on Engineering and Music—Supervisory Control and Auditory Communication*, *92*(4).

Wanderley, M. M. & Battier, M. (Eds) (2000). *Trends in Gestural Control of Music*. Paris (Centre Pompidou): IRCAM.

Zölzer, U. (2002a). In U. Zoelzer (Ed.), *DAFX – Digital Audio Effects*. New York: J. Wiley & Sons, Chapter Introduction, pp. 1–30.

Zölzer, U. (Ed.) (2002b). *DAFX – Digital Audio Effects*. New York: J. Wiley & Sons.

Appendix A

Computation of various centroid-like features

By denoting $X(m, k)$ the short-term Fourier transform of the signal $x(n)$ with N the block or window size, m the block number, k the frequency bin and R_a the analysis step, we compute the spectral centroid from the magnitude spectrum as

$$cgs_{B,c_0} = \frac{\sum_{k=1}^{N/2}(k-1)\cdot X(m,k)}{N\left(\sum_{k=1}^{N/2} X(m,k) + c_0\right)} \qquad \text{(A1)}$$

using Beauchamp's correction (Beauchamp, 1982). The usual definition of spectral centroid is obtained for $c_0 = 0$. Depending on the c_0 value, the spectral centroid has different behaviour for low level signals (the higher c_0, the lower the centroid values of low level signals).

Another centroid-like measure is computed from the energy spectrum as

$$cgs_{FT^2} = \frac{\sum_{k=1}^{N/2}(k-1)\cdot [X(m,k)]^2}{N\sum_{k=1}^{N/2} [X(m,k)]^2}. \qquad \text{(A2)}$$

Given $h(n)$ the window and the RMS energy as

$$e(x,m) = \frac{\sqrt{\sum_{i=1}^{N} [x(mR_a - N/2 + i)\cdot h(i)]^2}}{\sqrt{\sum_{i=1}^{N} [h(i)]^2}}, \qquad \text{(A3)}$$

the low–high frequency balance is computed from the energy of the derivative as

$$b_{lhf} = \frac{e(x',m)}{e(x,m)}. \qquad \text{(A4)}$$

The zero-crossing rate is the number of times the wave form crosses 0 with a positive derivative, divided by the number of samples N of current block m.

Appendix B

Examples of warping functions

Various warping functions can be used to modify the behaviour of the control signal $c[n]$. The simplest one is the linear function

$$\mathcal{H}_1(c[n]) = ac[n] + b. \tag{B1}$$

The sine wave function, given by

$$\mathcal{H}_2(c[n]) = \frac{1 + \sin(\pi(c[n] - 0.5))}{2}, \tag{B2}$$

allows one to strengthen the proximity of the control from its bounds. We note $\mathbb{1}_a$ the indicator function (whose value is 1 if the test a is true and 0 if a is false). The truncated function given by

$$\begin{aligned}\mathcal{H}_3(c[n]) =& (t_m \mathbb{1}_{c[n]<t_m} + t_M \mathbb{1}_{c[n]>t_M} \\ &+ c[n]\mathbb{1}_{t_{m \le c[n] \le t_M}})/(t_M - t_m)\end{aligned} \tag{B3}$$

is useful either to focus on a specific range $[t_m; t_M] \in [0;1]$, or to ensure the control will not clip. The two part power function given by

$$\mathcal{H}_4(c[n]) = s_m{}^{\left(\frac{\alpha - c[n]}{\alpha}\right)} \mathbb{1}_{c[n] \le \alpha} + s_M{}^{\left(\frac{\alpha c[n]}{1-\alpha}\right)} \mathbb{1}_{c[n] > \alpha} \tag{B4}$$

is mainly used for time-scaling, since it gives the time-compression ratio $s_m < 1$ for $c[n] < \alpha$ and the time-expansion ratio $s_\mathrm{M} > 1$ for $c[n] > \alpha$. The parameter $\alpha \in [0, 1]$ divides the [0, 1] segment into two parts: the lowest $[0, \alpha]$ will be contracted, the upper $[\alpha, 1]$ will be dilated. The logarithmic function

$$\mathcal{H}_5(c[n]) = \log_{10}(\alpha + \mu c[n]) \tag{B5}$$

and the exponential function

$$\mathcal{H}_6(c[n]) = 10^{\mu\,(c[n] - \alpha)} \tag{B6}$$

are useful for conversion such as from dB to amplitude and vice-versa. The compressor function is given as a two-part function by

$$\mathcal{H}_7(c[n]) = c[n].\mathbb{1}_{c[n]<T_c} + (T_c + p.(c[n] - T_c))\mathbb{1}_{c[n] \ge T_c} \tag{B7}$$

with the slope $p < 1$ and T_c is the compressor threshold, and the expander function as

$$\mathcal{H}_8(c[n]) = c[n].\mathbb{1}_{c[n] \ge T_c} + (T_c + p.(c[n] - T_c))\mathbb{1}_{c[n]<T_c} \tag{B8}$$

with the slope $p > 1$ and T_e the expander threshold. The smoothing function given by

$$\mathcal{H}_9(c[n]) = \frac{\sum_{k=n-o}^{n+o} b(k)}{2o + 1} \mathbb{1}_{n \in [o+1; NT-o]} \tag{B9}$$

is useful to reduce the control signal bandwidth, to avoid clicks or timbre modulation artifacts when controlling amplitude modulation. Smoothing can also be applied more appropriately in the signal conditioning layer.

Appendix C

Examples of scaling functions

Examples of scaling functions, noted $\mathcal{Z}_l$, are given by the general formula:

$$\mathcal{Z}_i(d_j[n]) = \frac{d_j[n] - \mathcal{Y}_i^-[n]}{\mathcal{Y}_i^+[n] - \mathcal{Y}_i^-[n]} \tag{C1}$$

with $\mathcal{Y}_{i,j}^+[n] = \mathcal{Y}_i^+(d_j[n])$ and $\mathcal{Y}_{i,j}^-[n] = \mathcal{Y}_i^-(d_j[n])$ the new bounds. We denote:

$$d_{j,T}^-[n] = \min_{n-T<l \le n} d_j[k] \tag{C2}$$

$$d_{j,T}^+[n] = \max_{n-T<k \le n} d_j[k] \tag{C3}$$

the local maximum and minimum,

$$\mathcal{M}^+ = \max, \qquad \mathcal{M}^- = \min \tag{C4}$$

the maximum and minimum between two values, and

$$\langle d_j[n] \rangle_T = \frac{1}{T} \sum_{k=n-T+1}^{n} d_j[k] \tag{C5}$$

the local mean. The new control bounds are given by filtering two parameters in a sliding frame, and using one of the four following filters taking into account the parameter local extrema:

$$\mathcal{Y}_1^\pm[n] = d_{j,T}^\pm[n] \tag{C6}$$

or the low-pass filtered and mean values:

$$\mathcal{Y}_2^\pm[n] = \mathcal{M}^\pm\left(d_j[n], \frac{d_{j,T}^\pm[n] + \langle d_j[n] \rangle_T}{2}\right) \tag{C7}$$

or the low-pass filtered and last values:

$$\mathcal{Y}_3^\pm[n] = \mathcal{M}^\pm\left(d_j[n], \frac{d_{j,T}^\pm[n] + d_j[n-1]}{2}\right) \tag{C8}$$

or a more general function:

$$\mathcal{Y}_4^\pm[n] = \mathcal{M}^\pm\left(d_j[n], d_{j,a}^\pm + \mathcal{D}_4^\pm[n] \mp \mathcal{E}^\pm[n] + \mathcal{G}_4^\pm[n]\right) \tag{C9}$$

using the parameter derivative $\mathcal{D}$:

$$\mathcal{D}_i^{\pm}[n] = \delta^{\pm} d_j[n]\big(|\mathcal{Y}_i^{+}[n-1] - \mathcal{Y}_i^{+}[n-2]|\big), \qquad \text{(C10)}$$

an exponential curve $\mathcal{E}$:

$$\mathcal{E}^{\pm}[n] = \beta^{\pm}\left[1 - e^{\alpha\left[n-n_a^{\pm}\right]}\right] \qquad \text{(C11)}$$

and the last interval range $\mathcal{G}$ defined as:

$$\mathcal{G}_i^{\pm}[n] = \gamma^{\pm} d_j[n]\big(\mathcal{Y}_i^{+}[n-1] - \mathcal{Y}_i^{-}[n-1]\big). \qquad \text{(C12)}$$

The values of $d_{j,a}^{\pm}$ and $n_a^{\pm}$ are initialized when the curve reaches its new bounds $\mathcal{Y}_4^{\pm}[n] = d_j[n]$ as follows: $d_{j,a}^{\pm} = d_j[n]$ and $n_a^{\pm} = n$. The weights $\delta^{\pm}$, $\beta^{\pm}$ and $\gamma^{\pm}$ are used to obtain different filter strategies and then different control curves.