

UNRAVEL:
ACOUSTIC AND ELECTRONIC RESYNTHESIS

Peter McCulloch, B.M.

Thesis Prepared for the Degree of
MASTER OF MUSIC

UNIVERSITY OF NORTH TEXAS

August 2004

APPROVED:

Joseph Rován, Major Professor

Jon Christopher Nelson, Minor Professor

Joseph Klein, Committee Member and Head of the
Music Composition Department

James C. Scott, Dean of the College of Music

Sandra L. Terrell, Dean of the Robert B. Toulouse
School of Graduate Studies

McCulloch, Peter. *UNRAVEL: Acoustic and Electronic Resynthesis*. Master of Music (Composition), August 2004, 59 pp., 30 illustrations, reference, 25 titles.

UNRAVEL, a work for alto saxophone and interactive electronics. Examines works for saxophone and electro-acoustic music. Analyzes modes of interactivity using Robert Rowe's guidelines, with sonogram, score, and programming examples. Investigates hybrid serial-parallel signal-processing networks, and their potential for timbral transformations. Explores compositional working methods, particularly as related to electro-acoustic music.

Copyright 2004

by

Peter McCulloch

TABLE OF CONTENTS

	Page
PART I: ANALYSIS	1
1. Introduction.....	2
2. History of Interactive Performance.....	4
3. History of Electro-Acoustic Works with Saxophone.....	10
4. Interactivity	13
5. Compositional Considerations	22
6. Technical Considerations.....	31
Design Philosophy	
Efficiency	
Feedback Delays	
7. Contextualization and Effects on Future Work	46
References.....	47
PART II: COMPOSITION	49
Key	50
Quarter-Tone Fingering Chart	51
Composition: UNRAVEL.....	52

PART I: ANALYSIS

Chapter 1

Introduction

UNRAVEL was composed between June 2003 and May 2004 at the Center for Experimental Music and Intermedia (CEMI) at the University of North Texas. The work was composed in collaboration with saxophonist Gregory Dewhirst, who premiered the work on April 20, 2004, at a concert for the fortieth anniversary of CEMI, with the composer operating the electronics. An additional performance followed at the North American Saxophone Alliance's biennial conference at the University of North Carolina at Greensboro on April 29, 2004.

The electronics are programmed in Max/MSP 4.1,¹ an extensible real-time digital signal-processing environment with a graphical interface. Additional third-party libraries were used to augment Max/MSP: the PeRColate 0.9 library, the Bennies 1.0 and Jimmies 1.1 libraries, and the Tap.Tools 2.0 collection.² Prerecorded samples of the saxophone and physically-modeled flute/electric-guitar hybrid were edited in Peak 4.0,³ with minimal processing.

The computer used in performance is an Apple⁴ G4 PowerBook running Macintosh Operating System 9 (MacOS9) with a clock speed of 1 GHz and 1 GB of RAM. Onboard audio jacks provide stereo input and output at a sampling rate of 44.1 kHz. The piece may be controlled either by the computer's keyboard, or by a Peavey⁵ PC1600x MIDI controller,

¹ www.cycling74.com

² www.music.columbia.edu/PeRColate; www.ircam.fr/forumnet; www.sp-intermedia.com

³ www.bias-inc.com

⁴ www.apple.com

⁵ www.peavey.com

connected to the computer by a USB MIDI interface; the controller offers additional control over volume levels in certain portions of the piece, but is not necessary for performance.

The title “*UNRAVEL*” has a dualistic meaning; the piece is based on an oblique quote from Ravel’s *Bolero*. One reading (literally “not Ravel”) describes the departure of the new work from its quotational material. The title, however, also describes the process of intervallic unraveling that defines the work’s formal structure. In order to represent this duality without the implication of preference, the title is capitalized.

Chapter 2

History of Interactive Performance

Interactive electro-acoustic music is a medium that, by necessity, relies on changing technology. The state of technology often influences the composition of a piece, not only in terms of what is capable, but also in how the composer relates to that technology. Nevertheless, the technology itself does not create musical value; Gérard Grisey's description is apt: the map is not the terrain.⁶ Interactive pieces begin with the composer, and indeed, the history of interactive music is one of composers using the technology available to create compelling music.

Advancements in technology are useful, however, in that they affect the technical complexity, and, moreover, the management of complexity possible within a system. Ultimately, however, interaction requires transparency. The interaction—from the standpoint of the audience—is perceived, rather than the technology that provides it.

The early live electronic works were based on the analog technology of the era, such as oscillators, ring modulators, impulse generators, and tape playback and delay. Though many remarkable pieces were produced using limited technology, such as John Cage's *Cartridge Music*,⁷ which utilized phonograph cartridges and contact microphones, and David Tudor's *Rainforest IV*,⁸ a performance art work premised on the filtering effects of resonating household objects, the physical constraints of the hardware ultimately limited the types and numbers of signal processes available to composers. Sometimes these constraints were quite literal, as David Behrman comments: "I needed my synthesizers to weigh only eight pounds, so they would fit

⁶ Lebaron, Anne Lebaron, Denys Bouliane, "Darmstadt, 1980," *Perspectives of New Music* 19, no. 1 (Autumn, 1980–Summer, 1981), 426.

⁷ Chadabe, Joel, *Electric Sound: The Past and Promise of Electronic Music* (Upper Saddle River, New Jersey: Prentice Hall, 1997), 81.

⁸ *Ibid.*, 98-100.

into a suitcase.”⁹ It might be argued, however, that such minimal interfaces encouraged more meaningful interactions with the electronics. That said, while the electronics could process sound, they did so on a temporally localized basis. Early electronics had no awareness of the larger context in which they operated; as a result, the composer was responsible for supplying control changes. Gordon Mumma’s highly advanced *Hornpipe* stands as a noteworthy exception. The instrument consisted of a horn fitted with custom filter circuitry; the filters, over the course of the piece, would adapt to the resonances of the performance space.¹⁰

With the developments of the modular synthesizer and analog sequencer, came greater independence. The standardization of voltage-control levels facilitated interplay between various synthesizer and sound-processing modules, allowing greater parametric freedom from the original input. Analog sequencers and low frequency oscillators increased the temporal flexibility of the system, allowing the repeated execution of parameter changes that exceeded human speed, endurance, and precision, and were increasingly uncoupled from the performer’s live input.

With improvements in the speed, miniaturization, and price of microchips, computer control of synthesizers became attainable, resulting in systems such as the Alles Synthesizer at Bell Labs,¹¹ and the SYTER system at GRM,¹² and Di Giugno’s 4A, 4B, and 4C machines at IRCAM.¹³ With analog devices, the interaction of modules was limited by hardware connections; for example, a sequencer could only control those modules for which it had outlets

⁹ Ibid., 101.

¹⁰ Ibid., 94-95.

¹¹ Ibid., 178-179.

¹² The Groupe de Recherches Musicales in Paris, now InaGRM, The Groupe de Recherches Musicales de l'Institut National de l'Audiovisuel. Roads, Curtis, *The Computer Music Tutorial* (Cambridge, MA: MIT Press, 1996), 197.

¹³ Roads, 918-919.

available.¹⁴ Software control of synthesizers removed many of these constraints, resulting in a massive increase in the number of parameters controlled.

This greater flexibility also made systems more difficult to control. By eliminating the physical interface in favor of a more precise, text-based interface, control became an increasingly abstract concept. This became even more important with the development of the 4X at IRCAM; the 4X was the first fully programmable real-time digital signal processor. The creation of higher-level control languages, such as Max, in 1987, partially addressed this problem. Though not a programming language by computer science standards, Max incorporated programming concepts such as encapsulation¹⁵ and templates.

Computer speeds now have advanced to a point where several complex audio processing tasks may be simultaneously accomplished without the need for specialized hardware. There are several robust real-time interactive computer music languages spanning a variety of computer platforms, including Max/MSP, SuperCollider, jMax, PureData and CsoundAV.¹⁶ In the seventeen years since its creation, Max has been extended to perform a variety of real-time digital signal processing tasks: pitch-tracking, score-following, and a host of other technologies allow for higher-level analysis, so that the computer may have more awareness of its musical surroundings, and, therefore, make informed decisions with some degree of autonomy. This, in turn, will improve the quality of interactions, allowing the player to play with, rather than simply into, the electronics. The notion of having a built-in “critic” has been incorporated into many

¹⁴ The use of a multiplexer would address this issue, but requires additional hardware for each multiplexed parameter.

¹⁵ Encapsulation refers to the grouping together of actions into one higher-level action. Playing a note on a piano is an encapsulation of several physical gestures, such as moving the arm into position, depressing and then releasing a key.

¹⁶ www.cycling74.com; www.audiosynth.com; www.ircam.fr/jmax; www-crca.ucsd.edu/~msp/software.html; www.csounds.com/csoundav.

MIDI-based interactive systems.¹⁷ With greater processing overhead, this degree of self-analysis is increasingly possible with live audio input.¹⁸

New music, in general, presents more problems for study than the traditional repertoire; many works remain unpublished, or, where published, prohibitively expensive, and recordings are fewer in number where available. Interactive music introduces additional challenges: a score is not a complete representation of a piece and is limited in terms of the information it can convey about the behaviors of the electronic part. For a composer, it is often useful to study another composer's electronics in order to understand how to design and implement interactions. There remain many barriers to this study, however; unsurprisingly, these are also the factors that impede the dissemination of interactive electro-acoustic music. The performance of hardware-based works requires the purchase or rental of specific sound-processor units, for which the marketplace will quickly provide replacements. Works featuring custom electronics are even more likely to become lost to history, at least with regard to performance.¹⁹

Software-based works fare slightly better, at least conceptually, though they carry intellectual property concerns. A commercial audio effects unit is the intellectual property of the commons; any owner may freely use the unit's algorithms. A custom work of programming, on the other hand, is highly personalized. Composers may invest considerable amounts of time in the construction of electronics, and having this system quickly turned into a cliché by the open market is an unattractive proposition. Many interactive pieces also contain proprietary technologies, further restricting their distribution.

¹⁷ Rowe, Robert, *Interactive Music Systems* (Cambridge, MA: MIT Press, 1993), 200-202.

¹⁸ Calculations capable of being performed at, or faster than, the sampling rate are referred to as signal-rate.

¹⁹ There is also the question of what happens to an interactive piece when, due to technological obsolescence, it is no longer interactive.

Due to these factors, many interactive composers have little exposure to live interactive performances, and even fewer opportunities for the detailed study of these works. This has the theoretical benefit of encouraging original approaches, but, more often, simply causes a duplication of efforts on low-level software tasks. Furthermore, without models for the higher-level control structures that facilitate complex, well-designed interactive compositions, many aspiring interactive composers' works lack basic mechanisms for rehearsal or in-performance error-correction.

Given the above observations, composers' conceptions of human-computer interaction may be profoundly influenced by a relatively small repertoire. Indeed, throughout the composition and programming of *UNRAVEL*, I have drawn particular inspiration from two interactive works: Jonathan Harvey's *Advaya*, for cello, electronic keyboard and electronics, and Cort Lippe's *Music for Flute and Computer*. In the case of *Advaya*, there is a published score that is very detailed and provides sufficient material for study; since the real-time signal-processing is done using standard effects, such as harmonization and reverb, study is relatively straightforward. I have been fortunate to have the opportunity to study the electronics for *Music for Flute & ISPW*.

Both pieces have extraordinarily intricate electronics parts characterized not only by the quality of their sound, but also by their restraint; processes do not continue on arbitrarily, but have articulated starting and finishing points. As Lippe describes,

The instrument/machine relationship moves constantly on a continuum between the poles of an "extended" solo and a duo. Thus, musically, the computer part is sometimes not separate from the flute part, but serves rather to "amplify" the flute in many dimensions and directions; while at the other extreme of the continuum, the computer part has its own independent musical function.²⁰

²⁰ Lippe, Cort, *Music for Flute & ISPW* (Paris: Cort Lippe, 1994), 2.

Perhaps the most influential lesson, however, came in terms of philosophy. Ultimately, the electronics are not the piece; they are simply a space in which the acoustic material may resonate.

UNRAVEL, like many works before it, attempts to make the most out of the technology available. At the component level, the electronics are not without precedent, but their combinations offer novel variations on standard signal-processing approaches. Virtuosity is required not only of the performer, but also of the computer; attention to efficiency in programming provides the overhead that allows the computer to deliver this virtuosity.

Chapter 3

History of Electro-Acoustic Works with Saxophone

[A]n instrument with an entirely new sound – powerful, far-reaching, expressive and beautiful. With its unique tonal quality, it offers the best imaginable link between the very high voices of the orchestra and the very weak ones or those with a very uneven timbre ... Uniting strength and charm, it does not drown out the one kind and cannot be drowned out by the other – it is a perfect instrument.²¹

The timbral and dynamic flexibility of the saxophone is the result of careful design and experimentation by Adolphe Sax in the nineteenth century. Nevertheless, these same characteristics have proved particularly advantageous in electro-acoustic music of the twenty-first century. The saxophone has sufficient power to be heard in loud situations, yet is also capable of playing with great subtlety; similarly, its wide variety of timbres allows the saxophone to blend with a diverse spectrum of electro-acoustic sounds.

In the twentieth century, the saxophone emerged as an important voice in both jazz and western art music. A lack of traditional repertoire has encouraged saxophonists to commission works, and participate in emerging genres of music, such as jazz and electro-acoustic music; such experimentation, in turn, has expanded the timbral possibilities of the saxophone.²² This adventurousness has paid huge dividends in the advancement of saxophone technique and timbres. Advanced saxophonists often have a range exceeding three octaves,²³ and can play at extremely fast speeds; additionally, players are familiar with a wide variety of extended techniques, including multiphonics, slaptongue, keyslaps, and fluttertongue.

²¹ Raumberger, Clause, Karl Ventzke, 'Saxophone: History and Use', Grove Music Online ed. L. Macy (Accessed 25 June 2004), <http://www.grovemusic.com>.

²² This describes not what the saxophone is physically capable of, but rather, those sounds that are considered to be acceptable saxophone sounds (e.g. multiphonics).

²³ Some players have a range of greater than four octaves.

This facility has not been lost on composers; there is a rapidly growing body of works for saxophone with electro-acoustic music.²⁴ Similarly, there are many different compositional approaches. One of the most rewarding examples comes from Jean-Claude Risset's *Voilements*, for tenor saxophone and stereo tape. In a lecture given at the University of Texas, Risset describes the work's impetus:²⁵

In the course of *Voilements*, the tape first echoes the soloist, multiplying his sound, but also altering its way of playing, warping it as a wheel which does not go round (the title alludes to a veil or a sail, but it also means "buckles" or "warps"). The equal temperament tuning is eroded by microtonal intervals or multiphonics; the tension increases, up to a point where melodic lines get twisted into loops as on a broken record. Then, as if one zoomed backwards, the relation between the soloist and the tape become more remote and peaceful: the tape becomes a distant background for the gestures of the soloist.

The result of this process is an overarching timbral transition, in both the tape and saxophone parts, from bright saxophone sounds towards darker, electronic sounds. By moving towards a common goal, the differences between the two parts are minimized, even as the saxophone is distanced from its signature timbre; however, Risset also uses synthetic sounds, in the form of inharmonic oscillator banks, to spectrally separate the two parts in the middle movement.

Gesture also plays an important role in *Voilements*. The saxophone part features a wide variety of extended techniques, most notably tongueslaps, fluttertongue, glissandi, and unpitched breath attacks. Because the tape sounds are derived from the saxophone part, they retain a certain unity of gesture; however, varying levels of timbral similarity provide contrast, though remaining gesturally subservient to the live saxophone.

Morton Subotnick's *In Two Worlds*²⁶ provides another example of the possibilities available with saxophone and computer. *In Two Worlds* is a work for alto saxophone and

²⁴ Both the North American Saxophone Alliance biennial conference and the World Saxophone Congress feature electro-acoustic concert series.

²⁵ Risset, Jean-Claude, "Computer Music: Why?" (Austin: University of Texas, France-UT Institute for Interdisciplinary Studies, January 9, 2004, www.utexas.edu/cola/france-ut/archives/risset_2.pdf), 10.

²⁶ This work has been presented in a variety of forms. I will be speaking of the recording with John Sampen.

interactive electronics. The electronics are programmed in Interactor²⁷ and control a Yamaha TX-802 FM tone module.²⁸ *In Two Worlds* and *Voilements* differ in terms of timbre; while *Voilements* makes extensive use of the saxophone, *In Two Worlds* utilizes FM timbres, which, though blending with the saxophone during the dream-like interludes, transform into a significantly more aggressive timbre in the driving rhythmic sections. Subotnick deals with this disparity by uniting the saxophone and electronics in a contrapuntal manner. The timbral differences complement the adversarial nature of the player-computer relationship.

As mentioned before, the standard of virtuosity for saxophonists is high; computers are obviously capable of feats of technical dexterity as well. Because of these characteristics, many pieces for saxophone and computer (or tape) place the saxophone and computer in an adversarial relationship. This competition generates a natural sense of drama, because it sets the work in the larger context of the “man versus technology” archetype, a formula that has been successful not only in music, but in other fields of human-computer interaction, such as chess.²⁹ Though *UNRAVEL* is not wholly characterized by this conflict, it does make use of this modality, pushing both the saxophone and computer to their respective limits.

²⁷ An early interactive programming environment developed by Subotnick and Mark Coniglio.
www.troikaranch.org/interactor.html

²⁸ www.yamaha.com

²⁹ Gary Kasparov’s February 10, 1996, loss to Deep Blue, an IBM supercomputer, attracted worldwide attention.

Chapter 4

Interactivity

This definition, by Todd Winkler, provides insight into the philosophy that underlies interactivity:³⁰

Interactivity comes from a feeling of participation, where the range of possible actions is known or intuited, and the results have significant and obvious effects, yet there is enough mystery maintained to spark curiosity and exploration.

All music utilizes a mixture of predictable and unpredictable elements to create drama.

Interactivity requires a similar balance: if interactions between the performer and computer are too obvious, the outcome becomes inevitable, and interest is lost; if interactions are consistently oblique, the feeling of participation is lost. Most interactive works utilize varying levels of oblique and obvious connections to ensure that the interaction is both dramatic and perceived, both for the audience and the performer.

In *Interactive Music Systems*, Robert Rowe sets out three ways of classifying interactive systems. As Rowe points out, these categories are by no means exhaustive or mutually exclusive, but they may be useful in discussing characteristics of an interactive music system.³¹ An interactive piece will typically contain a more specific implementation of processes than an interactive system, but Rowe's following classifications are still helpful:³²

1. Score-driven or performance-driven
2. Transformative, generative or sequenced
3. Instrument or player paradigms

³⁰ Winkler, Todd, *Composing Interactive Music: Techniques and Ideas Using Max* (Cambridge, Ma.: MIT Press, 1998), 3.

³¹ It is important to note that Rowe is writing primarily about systems, which, by definition, tend towards a greater degree of flexibility than interactive pieces.

³² Rowe, 6-7.

UNRAVEL is primarily a score-driven work; most of the electronic events in the piece are planned around the score. For example, cue 21 triggers a low, sustained sound upon the saxophone's arrival at C#4; in performance, C#4 can only trigger cue 21 once, and only after the computer operator has advanced to cue 20.



Figure 1 *UNRAVEL*, mm. 37-40

The reason for choosing a score-based, rather than performance-driven, approach lies in the differences between composition and improvisation. Composition has a significant advantage over improvisation in that it happens out of time; as such, the material may be evaluated, edited, expanded, or even discarded, and all before performance. Improvisation dictates a certain immediacy, particularly in a solo context, and this “in-time” quality often leads to practical choices in material. Complex formal structures are extremely difficult to construct in real-time for a composer, and, similarly, it is much to ask of an instrumentalist. Additionally, improvisatory pieces require a significant amount of trust in the improvisational ability of the performer, not to mention the performer’s skill in matching the style of the piece.

Since the score is fixed, the additional analytical overhead associated with performance-driven approaches, such as the usage of score-following software, was unnecessary. At present, score-following is still not entirely reliable. Instead of a single score-follower, *UNRAVEL* uses

event-specific followers, of a considerably simpler nature, to trigger actions within a cue; these triggers are repeatable in some circumstances, allowing for further flexibility in performance.

Figure 2 provides an example of this approach. The software first checks to see if the current cue is cue 26: if no, pitch values are ignored; if yes, test the pitch range. When the saxophone's pitch is between B4 and D5, turn on the harmonizer; notes outside of that range turn off the harmonizer. The advantage of this type of interactivity within a cue is that it decreases time-specificity in the execution of the cue; the computer operator, by advancing to cue 26, is only making a preparatory gesture for the saxophone. The electronics do not actually change until triggered by the saxophonist, allowing the saxophonist to time the entrance more flexibly. Additionally, this format allows the composer to compose fewer cues, yet with tighter interactions within the cues.

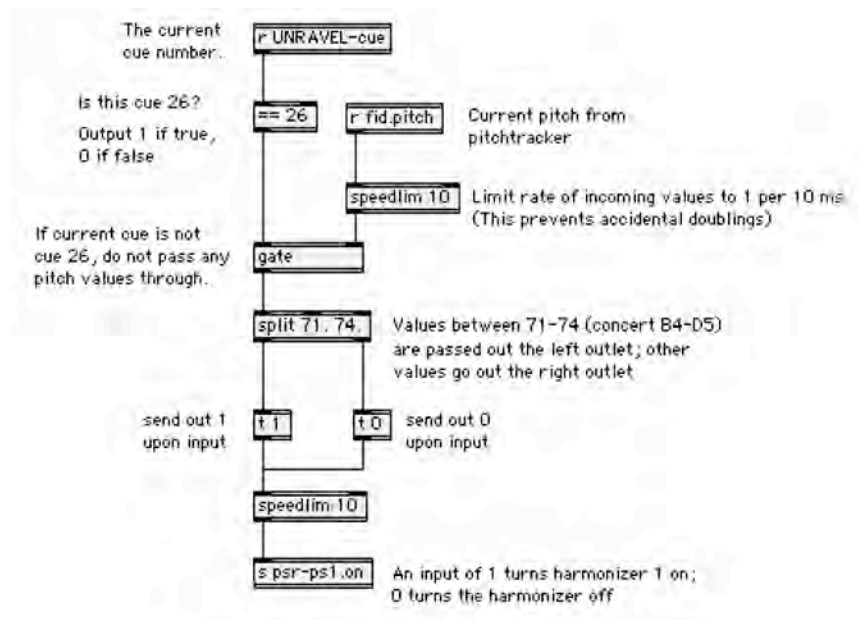


Figure 2 Event control for cue 26

In addition to control-rate triggering, there are sections of the work where the performer has direct control over the signal processes. In Figure 3, the player uses amplitude to choose

between inputs of a feedback matrix containing four interconnected harmonizers and delays. Because of the connection coefficients, soft input will produce descending, ghosted triplet figures, whereas loud input will produce rapidly ascending glissandi (Figure 4). By moving through these dynamic regions, the player can dramatically alter the directionality of the electronics.³³



Figure 3 *UNRAVEL*, mm. 55-57

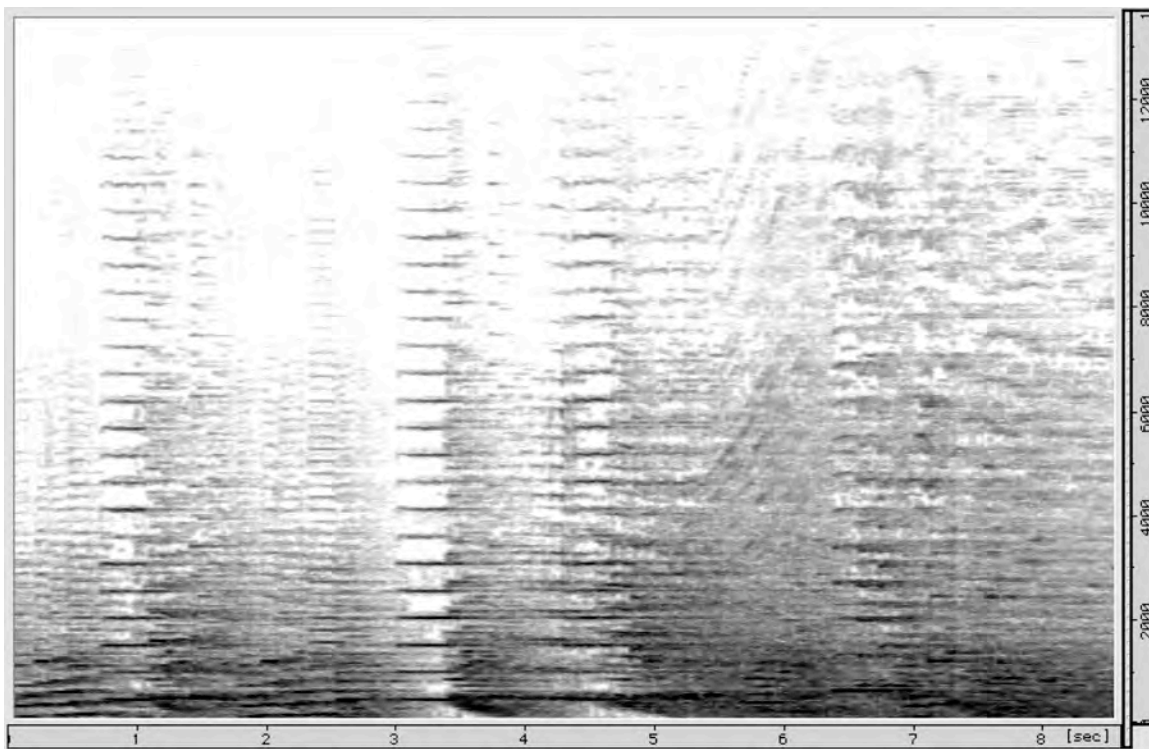


Figure 4 Sonogram of mm. 55-57 (saxophone and electronics)

³³ The design of this module is discussed in Chapter VI.

UNRAVEL uses a combination of response structures; the structure is articulated by a transition from sequenced and generative material, reacting to discrete values, such as pitch, towards transformative material, controlled by amplitude and timbre. Cue 7 provides an example of a sequenced response: at this cue, the computer simply plays four soundfiles in a preprogrammed order. A more interesting example comes at measure 27, where a set of soundfiles are automatically triggered by the arrival on E6; here, there is a delay between the activation and the playing of the soundfiles, so that they deflect the saxophone's trajectory, instead of capping it (Figure 5).

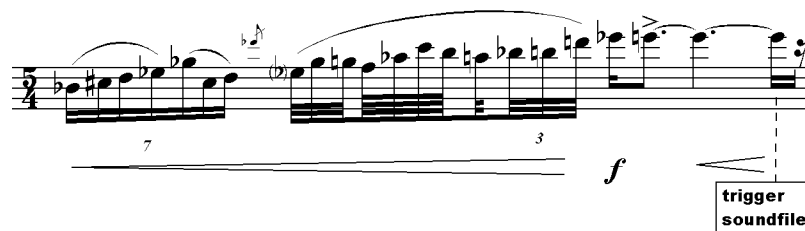


Figure 5 *UNRAVEL*, m. 27

Generative responses also appear in *UNRAVEL*, albeit less prominently. The slightly churning background sounds that appear following the long, low note in measure 39 are the result of real-time algorithmic processing of the original *Bolero* excerpt. The process is essentially a variant of phasing:³⁴ various portions of the soundfile are being looped, but the size, location, and volume of the loops are cyclically altered in a terraced fashion. A linked process examines whether the saxophone is playing a note within a particular grid of time; that value goes into one of two probability systems, one if playing, the other if silent. Each of these systems has an independent percentage-pass value (Figure 6); if a randomly picked value exceeds this threshold, the sound is muted. If it is within the threshold, the loop is unmuted. By

³⁴ Phasing in a formal, pattern-oriented sense, not DSP phase-shifting.

altering the threshold, the loops may play in one of three ways: with the player, partially with the player, or only when the player is resting—providing, in this case, an almost subconscious level of interaction.

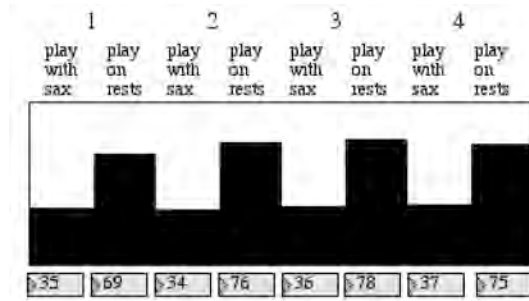


Figure 6 Interface for four cyclical soundfile players

Transformative algorithms perform the bulk of the work at the control-level as well as the signal level. *UNRAVEL* applies a variety of mapping strategies to the parameters of pitch, amplitude envelope, and spectrum, to produce interactivity.

The duo between the saxophone and physically-modeled flute is governed by pitch; the flute receives the saxophone’s pitch values at a delay of one second. Since the flute’s amplitude is linked with the saxophone’s undelayed amplitude envelope, however, the relationship is made more complicated. Additionally, the flute, in later cues of the first section, avoids playing G5 with the saxophone, as shown in Figure 7. This avoidance relationship breaks up the correlation between the two instruments, and helps to create drama.



Figure 7 *UNRAVEL*, mm. 33-34

The most important continuous parameter is amplitude envelope, which governs the majority of transformative processes. Tracking the amplitude envelope of the saxophone offers several benefits. It is easy to measure to consistently,³⁵ provided proper care is taken in setting gain levels at the input stage. Instrumentalists are aware of loudness on a continuous basis; amplitude tracking exploits this ability, offering an immediate and familiar control to the performer. The granularity of the computer's amplitude values, however, exceed that of the player, whose estimation of dynamics, sometimes thankfully, is context-sensitive.³⁶ The result is a system which offers the player some level of control, yet also has enough margin of error to provide variation between performances.

Amplitude mapping is used in a number of ways in *UNRAVEL*. In mm. 13-40, the saxophone's amplitude is mapped to the synthetic flute in several ways. First, the amplitude envelope of the saxophone is applied directly to the flute; as the saxophone gets louder, the flute follows. Amplitude directly controls the depth and frequency of the flute's vibrato; higher amplitudes correspond to higher values for both parameters. The depth of the vibrato also increases; because vibrato depth is a function of air pressure in this physically modeled flute, high depth values create the timbral effect of double and triple-tonguing. As the depth increases, the flute's timbre becomes more noisy, and less pitched. This timbral change is borne out in Figure 9, where the clearly defined harmonics of the flute become progressively noisier as the saxophone crescendos at the beginning of measure 36 (Figure 8).

³⁵ As opposed to pitch-tracking, which is best tracked in discrete values due to its perceptual nature.

³⁶ For this reason, it is generally preferable to use amplitude as a continuous control, rather than in a discrete manner, i.e. a trigger.



Figure 8 *UNRAVEL*, mm. 36-37

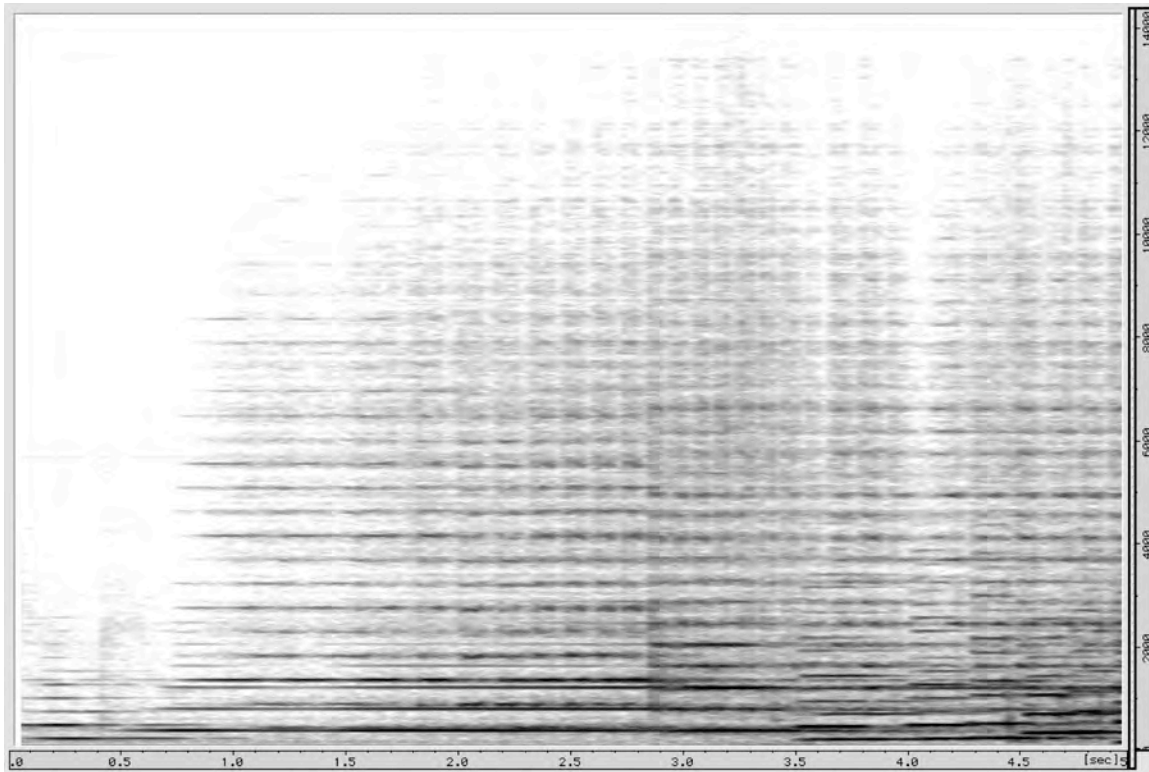


Figure 9 Sonogram of saxophone and flute, mm. 36-37

In later sections of the work, amplitude is used to control input into a bank of interconnected delays and pitch-shifters by driving a transfer function that dictates which of four pitch-shifters will receive input at any given time. This module is further detailed in Chapter 6.

The last classification that Rowe describes is that of instrument or player paradigms. An instrument paradigm uses the computer as an extension of the performer, whereas the player paradigm is focused on creating “an artificial player, a musical presence with a personality and

behavior of its own.”³⁷ *UNRAVEL* utilizes both strategies. The beginning of *UNRAVEL* is based around the player paradigm; though the flute roughly follows the contours of the saxophone line, it is timbrally independent of the saxophone. Additionally, the computer’s ability to trigger sequences of prerecorded flute soundfiles provides temporal separation from the saxophone.

As the work progresses, however, the computer becomes treated more as an extension of the saxophone, with synthesis techniques replaced by signal-processing. Unsurprisingly, the pitch-shifters, delays, and vocoder in the second half of the piece produce sounds that exhibit a stronger timbral correspondence to the saxophone. Nevertheless, due to the complexity involved with these processes, these modules produce a wide variety of sounds, some of which may be significantly removed from the original saxophone source.

³⁷ Rowe, 8.

Chapter 5

Compositional Considerations

The composition of an interactive piece entails multiple levels of collaboration. Frequently, the composer will recruit a player to record test soundfiles based on sketches for the piece. This allows the composer to explore musical ideas with any existing electronics, and/or construct new electronics based upon the sketches. In both of these instances, there is a process of matching acoustic to electro-acoustic sounds. Similarly, in the case of pre-existing electronics, the composer may write acoustic music that will be oriented towards a particular signal process. The effect is comparable to that of a filter, where the source material (here the acoustic sketches) is tested by a filter (the electronics). If the combination is fortuitous, the tuning process may begin; the goal is to tune either the filter or the source material such that the resonance between the two is greatest—e.g., the electronics behave in a predictable, musical manner for a specific style of input,³⁸ with a minimum of control changes required on the part of the composer.

UNRAVEL is the progeny of such tuning processes. In the early stages of planning a piece for saxophone and electronics, I came across Perry Cook's blotar~ Max/MSP object,³⁹ a physically-modeled flute/electric-guitar hybrid with impressive timbral capabilities. After constructing a prototype using blotar~ as a virtual instrument, I wanted to test it with live input. In lieu of this, I used a recording of a solo excerpted from Ravel's *Bolero*, which worked well from an input standpoint; so well, in fact, that I continued using it while finalizing the design of

³⁸ There are, of course, other potential relationships, and the breadth of this style is subject to the whims of the composer.

³⁹ www.music.columbia.edu/PeRColate.

the virtual instrument. When the patch was finally functioning properly, I tested it with disjunct motivic materials, but the output lacked musical resonance.

Quotation represented a technically effective but musically undesirable solution; *Bolero* is a highly recognizable piece, and a quotation from it would be glaring. Nevertheless, the quotation approach was not entirely illogical, as the setting for a quote greatly determines the perception of the quote in the context of the piece. Richard Felciano's program note for *Prelude*,⁴⁰ for solo piano, offered an interesting slant:

The problem I set for myself in *Prelude* is that of creating a transition between my own prelude and one of Chopin (his *Prelude in E-flat minor*) which would be so smooth that the listener would be out of one and into the other and back again without realizing it.

Felciano's *Prelude* is able to effect this transition because of its construction from materials drawn from the Chopin *Prelude*; in essence, Felciano's piece exists as an analog of the Chopin, occupying many of the same parameter spaces, such as gesture, register, and rhythm. Like any analog, it is not an exact copy, yet the connection to the original is convincing. Parametric, rather than direct, quotation emerged as the answer. In order to compose a suitable replacement passage, the *Bolero* passage was analyzed, with particular attention to how its structures were parsed by the Max/MSP patch.⁴¹

At a structural level, the *Bolero* excerpt has interesting, albeit subtle, features. The melody is quite consistent with respect to intervallic displacement, and this seemed to be the most important parameter to the electronic component, which loosely tunes itself to the input pitches. Intervallically, the melody is extremely linear; there are only three intervals that exceed a minor third, and none of these exceed a tritone. Repeated pitches help to create drama; though the repeated pitches in mm. 134-136 (Figure 10) are the most obvious example, there are

⁴⁰ Felciano, Richard, *Prelude for Piano* (San Francisco: Richard Felciano, 1997).

⁴¹ This is to say, in essence, what the patch found parametrically interesting, rather than simply what I found interesting about the quote.

repetitions in the form of figurations around a pitch, such as those occurring around Bb5 in mm. 131-132.



Figure 10 Maurice Ravel, *Bolero*,⁴² mm. 131-147, concert pitch⁴³

Rhythmically, the excerpt is somewhat restrained; the rhythms serve to stretch and compress the line temporally. Subtle syncopations, particularly in conjunction with sustained notes, articulate important phrases. Melodic motion tends towards arch-type figures, such as in mm. 135-138, and mm. 139-143. Overall, the passage has a sinuous, organic quality.

In an effort to duplicate the organic sound of the original, I recorded a series of improvisations with a Disklavier⁴⁴ utilizing my analytical observations as a rough set of guidelines. After transcription, I assembled the various motivic fragments into longer lines, and added ornamentation. This was done without consulting the score for *Bolero*; the intent was to fill the parameter space of the melody, instead of trying to duplicate it.⁴⁵

⁴² Ravel, Maurice, *Bolero* (Bryn Mawr, PA: Theodore Presser, 1957), 16-18.

⁴³ Ravel scores measures 131-143 for the soprano saxophone, with the soprano saxophone taking over on the F4 in measure 143. Many orchestras, however substitute the soprano saxophone throughout for the less common soprano, as was the case in the recording used in the composition of this work.

⁴⁴ www.yamaha.com

⁴⁵ Much in the way that two waveforms with sonically identical spectra may differ in appearance due to phase differences.

Despite this attempted decorrelation, many similarities emerged between the original and the analog. These similarities are not continuous; *UNRAVEL* is neither the integral nor derivative of the *Bolero* excerpt. Both openings feature a transit of a tritone (Figure 11-12). *Bolero* moves from Bb down to E, *UNRAVEL* moves from E to Bb (concert pitch). The retrograde contour of m. 13 of *UNRAVEL* corresponds to the contour of *Bolero*, m. 131. Both lines arrive on Bb in their second measure. This note is ornamented before descending to a lower sustained note. Though the ornamentation occurs in a different order in *UNRAVEL*, the effect is, nevertheless, roughly the same. These examples exhibit similar rhythmic structures as well, with flowing groups of notes interrupted by longer, sustained notes. *UNRAVEL*, however, utilizes more syncopation.



Figure 11 *Bolero*, mm. 131-134



Figure 12 *UNRAVEL*, mm. 13-18

There are organic connections between pitch, rhythm, and contour in other areas. *Bolero*, m. 136 (Figure 13), and *UNRAVEL*, mm. 21-24 (Figure 14) are different in many ways, but aspects of the repeated note figure in the *Bolero* excerpt resonate in the *UNRAVEL* example, such as the triplet rhythm in m. 24. Many of the other connections are extremely fragmentary in nature; the *UNRAVEL* excerpt is a noisy, time-distorted amplification of the original.



Figure 13 *Bolero*, mm. 134-138



Figure 14 *UNRAVEL*, mm. 21-25

The filtering model also provides an accurate description of the compositional process. *Bolero*'s replacement resonates within roughly the same musical space as the original, albeit with some amplification. While the listener is unlikely to perceive these more oblique connections, they accomplish their designed purpose as parametric analogue for the electronics, and offer an interesting paradigm for melodic construction and elaboration.

Though there are parallels, substantive differences remain between the two works. *Bolero* is a work that relies on its orchestration to provide color for an unchanging melody; each repetition of the themes adds further weight, with the whole being greater than the sum of its parts. Consequently, evaluating a section, much less a single instrument, outside of the context of the piece is contrary to the monolithic spirit of the piece; hence, *UNRAVEL*.

The significance of intervals in this first composed section led me to use the unraveling of intervals, through expansion and contraction, as a formal structure. *UNRAVEL* has six main sections, each focusing around a different arrangement of intervals.

1. Unison to microtonality, mm. 1-12
2. Small intervals, mm. 13-40
3. Wide intervals with glissandi, mm. 41-69

4. Inharmonic interval sets, mm. 70-91
5. Combination of previous sets, mm. 92-101
6. Extremely wide intervals, mm. 102-110

Section one (mm. 1-12) deals initially with only one pitch; the saxophone establishes the C-sharp before gradually expanding first to microtonal intervals, triggering the entrance of the computer, and eventually to whole steps. The flourishes in the computer part progressively widen in the length of sweep, while preserving narrow intervallic structures; this serves to draw the saxophone out of its narrow range.

The second section is constructed primarily from intervals of a third or less, as is the pattern in the underlying Ravel excerpt. Larger intervals are used to articulate phrases, and grace notes provide variety and deflection. Wide intervals and pitch bends are the focus of the saxophone in the third section. Specific wide intervals, losing definition (?), transform into registral displacements. The electronics initially explore microtonal variations around the saxophone's pitches, before branching out to aggressive arpeggios and glissandi.

The fourth section, a saxophone solo, utilizes a combination of spectral and serial procedures. Previous sections were primarily concerned with a specific interval or type of intervals. This section was composed as a prefiguring of the electronics heard in the section that follows;⁴⁶ the Max/MSP patch utilizes a combination of frequency and pitch shifting to produce arpeggiated chords. Analysis of the output of the patch revealed chords corresponding to the Bb, F, and C of chord 12 in Figure 16; the E-natural was added to the set to break up the uniformity of the intervals. Using that chord as a destination value, a recursive process of positive

⁴⁶ Those electronics are diagrammed in Chapter 6

frequency-shifting, and negative pitch-shifting was applied⁴⁷ in OpenMusic⁴⁸ yielding a progression of twelve chords that decompressed upwards, a process similar to the overall form of the piece. To make this pitch set progression less predictable, I created an indexing function (Figure 15) to break up the linearity.

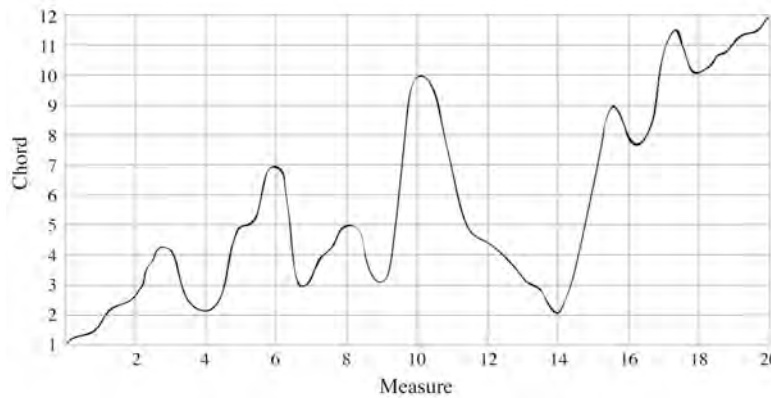


Figure 15 Transformation indexing function

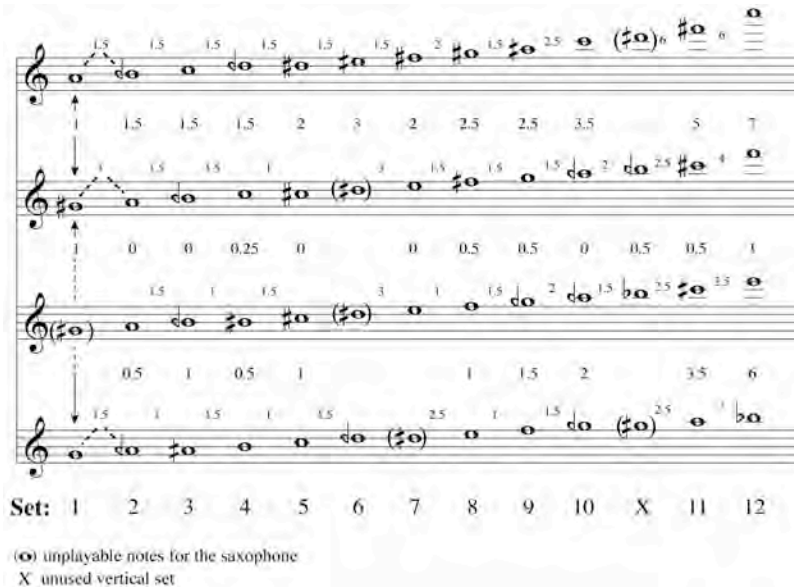


Figure 16 Pitch transformations

⁴⁷ Positive frequency-shifting causes the compression of intervals in an upward manner; negative frequency-shifting results in the downward expansion of intervals. By using a combination of frequency-shifting and pitch-shifting, I was able to produce a compressed spectrum that expanded upwards.

⁴⁸ OpenMusic is a visual, non-realtime music composition programming language.
www.ircam.fr/equipes/repmus/OpenMusic/

The source patch also produced the distinctive rhythmic figures seen in m. 10 of Figure 17. Similar to the process used in the creation of Figure 16, I created, in essence, a backstory for the rhythmic figure. The three-line staff used in Figure 17 represents contours for use with the pitch material of Figure 16; the inversions of these contours are also available. The contours are in groups of two or three pitches to allow for pitch selection within the chord, resulting in greater melodic flexibility when the transfer function is moving slowly.

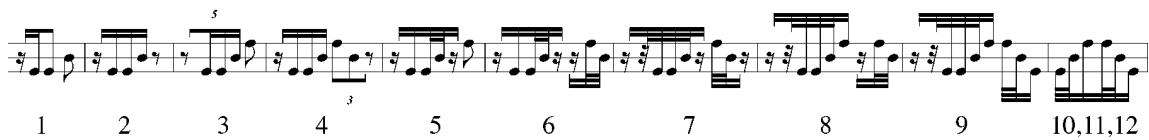


Figure 17 Rhythmic transformations

After writing out the results, it became clear that the rhythmic progression was too uniform; therefore, some notes, particularly at the beginning of the passage, were converted to rests. As one final layer, grace note outbursts (bearing resemblance to the flute's flourishes from m. 7) were sporadically interjected, as in m. 73 of Figure 18.



Figure 18 UNRAVEL, mm. 70-73

Though spectral and serial procedures were used in the construction of mm. 70-91, the passage does not carry the formal implications of either system. Rather, these techniques are used as a novel means of generating material and effecting an organic transition from the acoustic sounds to the electronic sounds.

The reentrance of the electronics in the cadenza marks a departure from the rigidity of this system; the chord table from the previous section is now read horizontally, creating ascending lines that mimic the ascending feedback processes in the computer part. These lines similarly progress towards the pitches in chord 12 of Figure 16. This cadenza section is a reinterpretation of materials heard in previous sections. The saxophone's ascent to the top of the altissimo range (Figure 19), and subsequent "one-upmanship" by the computer, fractures the saxophone's register, leaving the saxophone, in a sense, crippled.

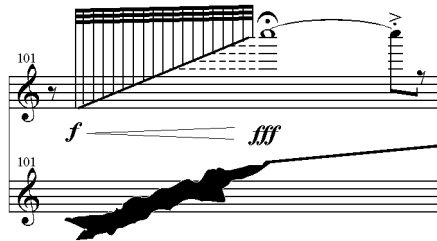


Figure 19 *UNRAVEL*, m. 101

The final section of the piece represents the saxophone exploded in register and broken in timbre. In a futile effort to compete with the hyper-altissimo range of the computer, the saxophonist utilizes alternate techniques (a whistle-tone, using teeth on the reed) to stretch the instrument's range higher. Keyslaps further represent the saxophone's disembodiment from its characteristic sound. The Eb, E, and F cell in mm. 107 and 109 offer the prospect of reconciliation, but this is never realized.

Chapter 6

Technical Considerations

Design Philosophy

The design philosophy utilized for the electronics in *UNRAVEL* is the synthesis of several years of experience with Max/MSP. Though not codified in the development process, these basic ideas guided the construction and configuration of the electronic modules:

1. Simple interactions can produce complex, organic results.
2. A module processes different inputs in different ways.
3. Operational complexity is kept to a minimum through the usage of presets.
4. A module is optimized for efficiency at the signal rate, independent of the control rate where possible.

The first principle is important in that it encourages dynamic, rather than static, interactions; these prove more rewarding to the listener as well as the performer. It has the added benefit of encouraging specificity in programming; modules are built to suit a specific musical context, instead of a general case. Principle two is a reflection of the way music works: if a performer plays at a different dynamic, articulation, or tempo, it affects how the music is perceived. Logically, it follows that the computer should similarly have some degree of context-sensitivity in its processing.

Preset systems are used in order to allow the composer to focus on larger-scale constructs. They represent higher-level data structures, making it easier to create, as well as notate, processes that change over time. The final principle ensures that the piece will run in a reliable manner in concert situations.

Efficiency

The separation of signal and control rates dates back to the earliest days of computer music; with mainframe time at a premium, reducing information was a way of improving efficiency. Lacking the speed necessary for real-time signal-processing, computers were first incorporated as control mechanisms for synthesizers, offering greater flexibility than their analog counterparts. For instance, Max, which later became Max/MSP,⁴⁹ was initially developed by Miller Puckette to control IRCAM's 4X⁵⁰ machine for Philippe Manoury's *Jupiter*.⁵¹ With faster chips, real-time signal-rate processing became possible. A number of languages now offer this possibility, most notably Max/MSP, jMax, SuperCollider, Csound, and PD.⁵² Though differing in substance, all adhere to some type of rate separation to improve efficiency.

Efficiency is important in real-time systems for two primary reasons: processor usage, and architecture. Processor usage is a huge factor in the design of real-time systems: when overhead is exceeded, audio production grinds to a halt. To make a signal network more efficient, control-rate variables are favored over signal-rate where substitution is possible. This, however, tends to alter the conceptual architecture of the patch in that it reinforces the static character of the affected variables. If a control-rate variable, for instance, gain (Figure 20, left), can only be seamlessly changed when the input to the network is muted, the flexibility of the network will be diminished. The solution to this problem is a set of objects that take input at the control-rate, but interpolate between values at the signal rate (Figure 20, center). While these objects address the problem of interpolation, they negate the processing gains made by using the control-rate. To address this problem, the programmers of MSP 2.0 created highly efficient

⁴⁹ MSP is the signal-processing extension of Max.

⁵⁰ di Giugno et al.

⁵¹ Chadabe, 183-184.

⁵² www.audiosynth.com; www.csounds.com; www-crca.ucsd.edu/~msp/software.html

objects that accept both control and signal rate variables (Figure 20, right). By reducing the number of function calls needed—one object does the job of two from the previous example—and taking advantage of vector-processing optimizations, less processing overhead is required.

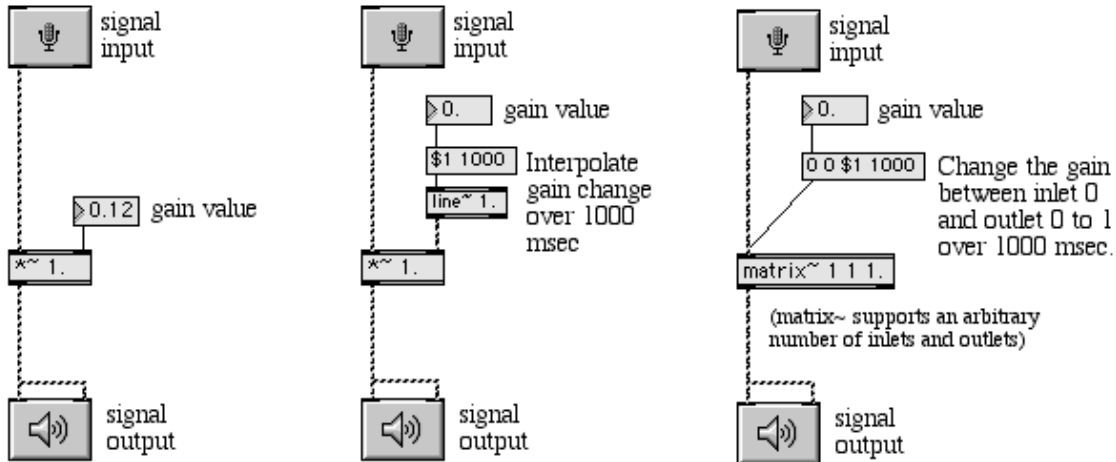


Figure 20 Three signal networks

These objects have had a significant impact in the design of the electronics for *UNRAVEL*. *UNRAVEL* utilizes over 1000 real-time signal calls, and would not run on existing machines without these optimizations. Many of these objects are used in a non-standard manner; the `matrix~` object, which is intended as a matrix mixer, is used in many different configurations due to its highly optimized code.

Figure 21 illustrates one non-standard usage of a `matrix~`: audio-rate preset morphing. The coefficient weights in the diagram are audio signals between 0 and 1. By inversely varying the two coefficient weights, a morph between the two sets of parameters is effected; the rate of change vastly exceeds that possible at the control rate, and incurs minimal CPU overhead. This system is utilized in the vocoder which appears at the end of *UNRAVEL*. The vocoder module

has over 300 parameters; control rate interpolation of these parameters proved to be prohibitively expensive. The matrix system, by contrast, is extremely reliable, as well as nimble.

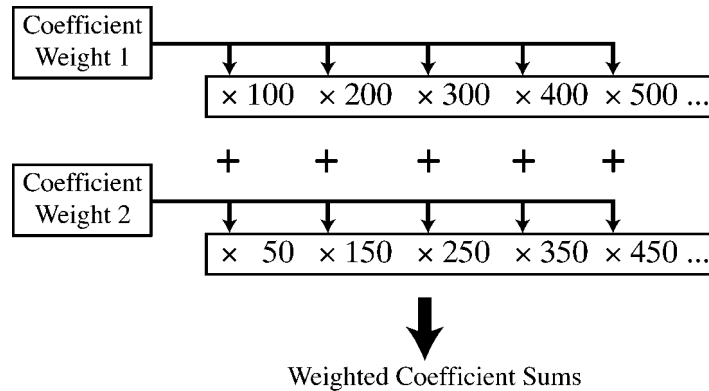


Figure 21 Matrix system for signal-rate preset interpolation

Feedback Delays

Programming an interactive piece is always a balancing act between processing efficiency and sonic interest. In the past, composers have often used delay, particularly with feedback, to produce thick textures with minimal processing strain. While it has been used effectively in pieces such as Larry Austin's *BluesAx*,⁵³ it has also been frequently abused. To avoid this, I examined the strengths and weaknesses of feedback delay, and developed two alternative implementations.

Feedback delay is extremely efficient from a CPU standpoint. Though less of a concern now, this was very important in the beginning days of real-time digital signal processing, or DSP.⁵⁴ Feedback delay also can create lingering harmonic and rhythmic resonances,⁵⁵ and allows the composer easy access to the timbres of the input instrument. Due to the cyclic nature of the

⁵³ Austin, Larry, *BluesAx*. Denton, TX: Larry Austin, 1996, digital audio disc.

⁵⁴ In fact, it is interesting to note that Max's development as a control system to the 4X had to do with a desire for greater signal processing capabilities. Winkler, 16-17.

⁵⁵ Jean-Claude Risset's *Sud* provides a good example of the usage of delay to create harmonic resonances.

process, the mingling of the current input with the past input can also yield interesting timbral transformations.

There is a significant downside to this cyclic construct, however. In acoustic music, repetition may be used to create or relieve tension. By repeating a figure in a dramatic passage, or greatly prolonging a note, another form of repetition, a composer sets up an expectation of change, creating tension. Repetition at the end of a phrase, however, decreases musical information, signaling denouement. As a technique, repetition is effective in performance due to a combination of performance practice and human error. Performers are generally trained to alter a repeated phrase slightly, in order to make it sound more natural. Even if the performer were to attempt to replicate the previous phrase, it would be sonically different, due to the immense number of variables involved in playing an instrument.⁵⁶ Analog tape delays shared the second characteristic; the magnetic tape did not capture high frequencies as well as low frequencies, resulting in copies of the input sound that lacked the fidelity of the original, producing an incremental diminution of higher frequencies. Digital feedback delay in its simplest configuration, however, does not share this attribute, creating an output that is an exact copy of the input. While there is a great tradition of canonic writing in Western music, this practice requires significant foresight and craft. Though the use of multiple taps decreases the temporal regularity of the signal, and thus the perception of repetition, several issues remain unaddressed.

There are also major problems in the control of feedback delay. Once a sound has entered the delay loop, it is practically impossible to excise the offending sound without clearing the delay buffer's contents. Undesirable incidental input, in the form of instrumental squeaks or

⁵⁶ This is what Rowe terms the "jitter" of the performer system.

clicks, will be greatly magnified by recurrences. Without a means for filtering the input, this problem is inevitable. There are similar vulnerabilities in the feedback stage: the same high levels of gain that produce sustain also make the chain highly vulnerable to excessive feedback, which would produce undesirable clipping. Additionally, changing the gain value alters the signal in the delay permanently. For instance, there is no way to create a delay that, for all continuous input, echoes three increasingly loud copies, and then decays. Recognizing these problems, I implemented two different constructions of feedback delay. The first is a feedback matrix network, which combines four harmonizers and four delays; the second is what I term a “variable serial network”.

Traditionally, most signal processing networks may be categorized as being either parallel or serial. Parallel networks allow for a varying blend of several independent processes (Error! Reference source not found.); unfortunately, the richness of the output is limited by the complexity of its modules. Serial networks are less independent, in that every process in the chain depends upon the output of its predecessor (Figure 23); nevertheless, they have deeper transformational capabilities than parallel networks.⁵⁷ Matrix networks, which incorporate characteristics of both serial and parallel networks, offer rich signal-processing capabilities.

⁵⁷

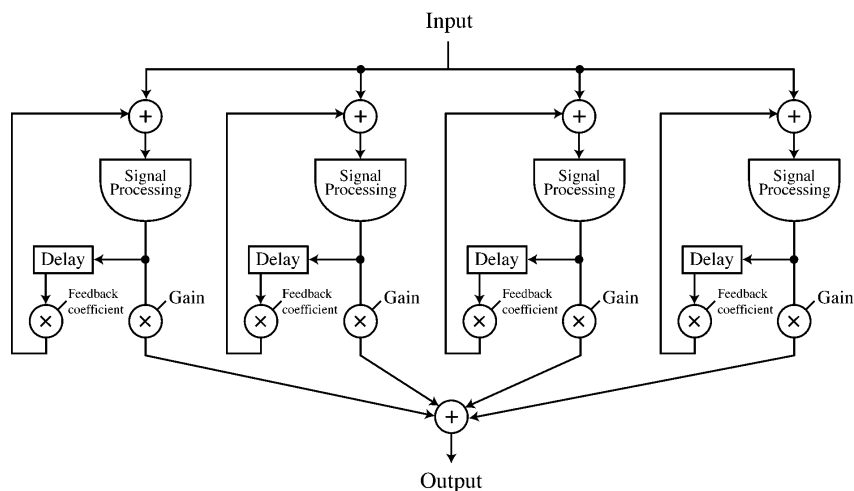


Figure 22 Parallel network

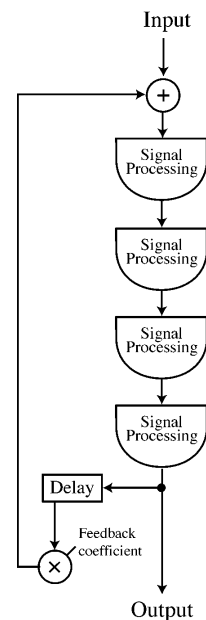


Figure 23 Serial network

UNRAVEL uses four harmonizers⁵⁸ and four delays that are interconnected by a mixing matrix. With a few exceptions,⁵⁹ the output of any module may be routed (with variable gain) to the input of any module, as well as to the stereo outputs. This allows for a variety of configurations, such as a harmonizer and delay routed together in a feedback loop; this example would produce a glissando, for very short delay times, or a scale with longer delay times. The uncoupling of the output of a module from the stereo output of the network has many implications. In the previous example, output might be heard before, or after, the harmonizer or delay modules; this offers four different viewpoints on the same process.

This increased flexibility has drawbacks, however. Matrices are particularly prone to feedback problems; to remedy this, specialized limiters are employed at the output of every module. Excessive feedback often happens rapidly, but may also occur as the result of a gradual

⁵⁸ Also referred to as pitchshifters.

⁵⁹ Harmonizers are not capable of direct feedback, i.e., connecting the output straight to the input.

subsonic accumulation;⁶⁰ consequently, it is important that the limiters be able to deal with both contingencies. A hybrid amplitude-following system was developed that tracks three different amplitude values: peak amplitude for the last 100 milliseconds, peak amplitude for the last 2000 milliseconds, and the continuous amplitude envelope of the input; this envelope smoothes attacks over 100 milliseconds, and decays over a span of ten seconds. This allows the system to react quickly to sudden spikes in volume, yet have enough memory of previous values to avoid “pumping”.⁶¹

The limiter uses the maximum of these three values as its reference amplitude level. If this level is below a specific threshold value, the original input signal is passed through unchanged. If, however, the signal exceeds the threshold, the limiter makes a decision as to what kind of feedback it is seeing, and then acts accordingly. For extreme situations, the gain is halved; when the incoming amplitude levels remain below a threshold value for three seconds, the gain returns to ninety-five percent of its original level.⁶² Less severe instances of feedback trigger an eighty-five percent reduction in the incoming signal to eighty-five percent of the input level; if this proves sufficient, the gain is restored slowly to its full value.⁶³ Though complex in development, this system of limiters is simple to operate, and highly fault-tolerant.

Amplitude levels control the saxophone’s input into the network. Each of the four inputs into the network has an amplitude envelope transfer function, as shown in Figure 26. This function determines how much of the saxophone’s signal is routed into any given input,

⁶⁰ In addition to the limiters, filters with dc-blocking coefficients were used to combat the accumulation of these artifacts.

⁶¹ Pumping is a form of self-oscillation of volume levels as the result of overly short attack and release times.

⁶² While ninety-five percent does not seem like a significant difference, this is bound to be reflected in more than just one node of the chain (i.e., if three members of a chain move to ninety-five percent, the reduction is more significant).

⁶³ Unlike in extreme cases of feedback, the amplitude propagation across the network will tend to happen at a relatively slow rate. As a result, it is more likely that the amplitude reduction will happen at one or two nodes, instead of the entire network, producing sustain.

dependent on the amplitude envelope of the saxophone. For example, the network in the upper-left of Figure 26 is active at soft and loud dynamics, and inactive at medium dynamics. The signal chain in the upper-right, however, only passes input when the saxophone is playing at a medium dynamic. Because these transfer functions are continuous, the saxophone, via crescendo or decrescendo, may cross-fade smoothly between signal processes. This attribute is exploited to considerable effect at cue 31 (Figure 24): loud input creates a descending glissando; input at a moderate volume is pitch-shifted upwards. Soft input is delayed 350 milliseconds, pitch-shifted up, then transformed into a descending glissando.



Figure 24 *UNRAVEL*, mm. 58-60

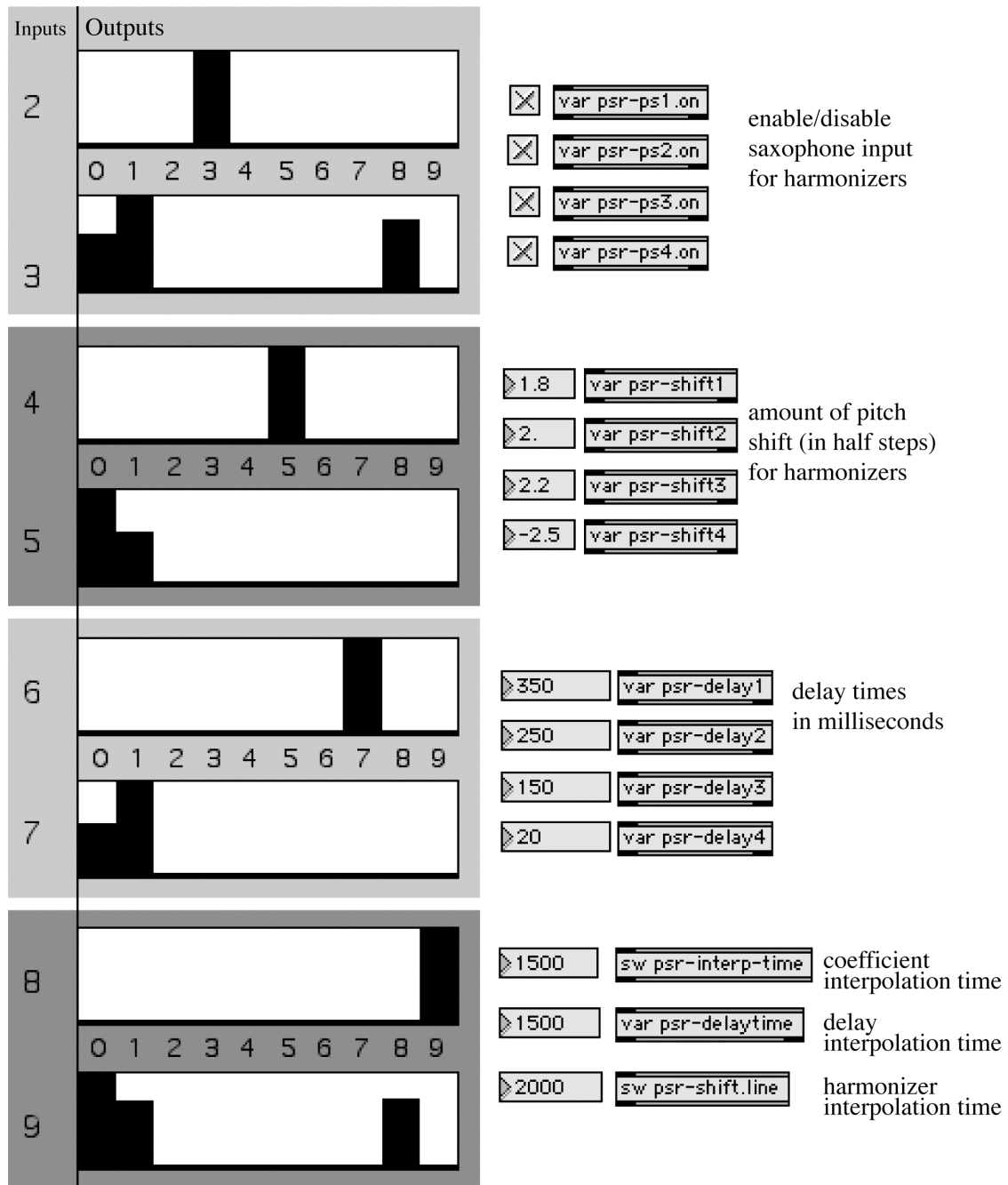


Figure 25 Interface for feedback matrix, mm. 58-60

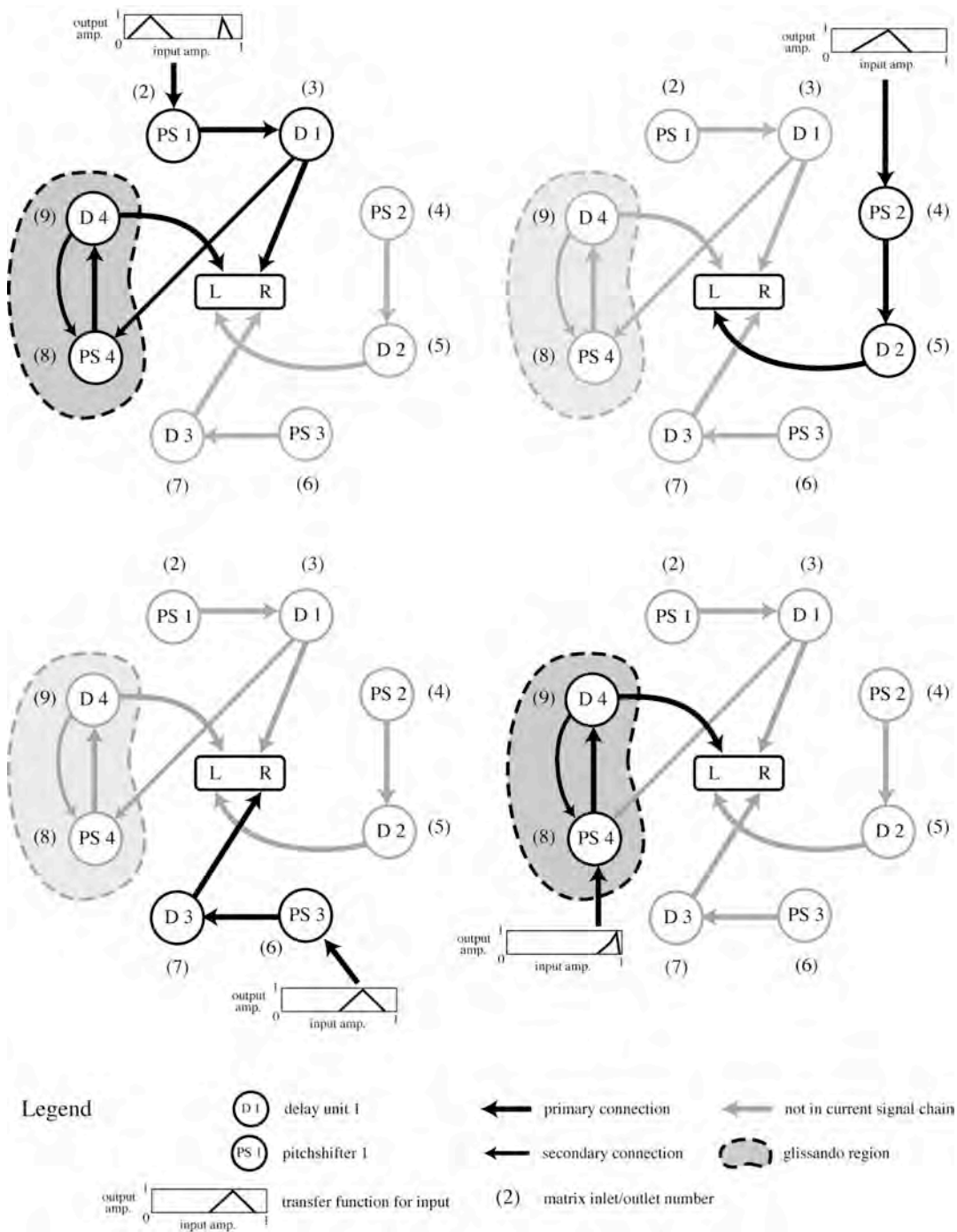


Figure 26 Diagrams of signal routings, mm. 58-60

The other type of feedback delay network used in *UNRAVEL* appears in the cadenza. This network uses a variable serial feedback connection, and has several differences from standard feedback delay implementations. Most feedback delays produce a dilation of time by repeating the input, and they do this in a continuous manner, without discrimination of input. In order to produce a stable system, the feedback coefficient must be between -1 and 1 .

This module, on the other hand, was designed with a deliberate emphasis on instability. Input is handled in a discrete, rather than continuous, manner: only notes that meet amplitude or pitch guidelines are passed into the loop. Where normal delay units are static in reaction, this module is highly gestural. When input is triggered, the delay line⁶⁴ ramps from three seconds to twenty milliseconds over a course of eight seconds; additionally, the feedback coefficient ramps from 1 to 1.2 over the course of seven seconds. From here, the input passes to the variable processing section.

The amplitude envelope of the input controls the speed of an oscillator that distributes the incoming sound to four different processes: pitch-shifting, frequency-shifting, comb filtering, and allpass filtering with delay; this signal chain is diagrammed in Figure 29. Higher volumes produce faster oscillator speeds, which disperse the sound rapidly through the network, producing dramatic timbral transformations. As the feedback coefficient increases, and the delay time decreases, the sound quickly transforms from the harmonic tones of the saxophone into an aggressive enharmonic cloud.

⁶⁴ Actually, two windowed delay lines are used to avoid the changes in pitch normally associated with signal rate interpolation of delay time values.

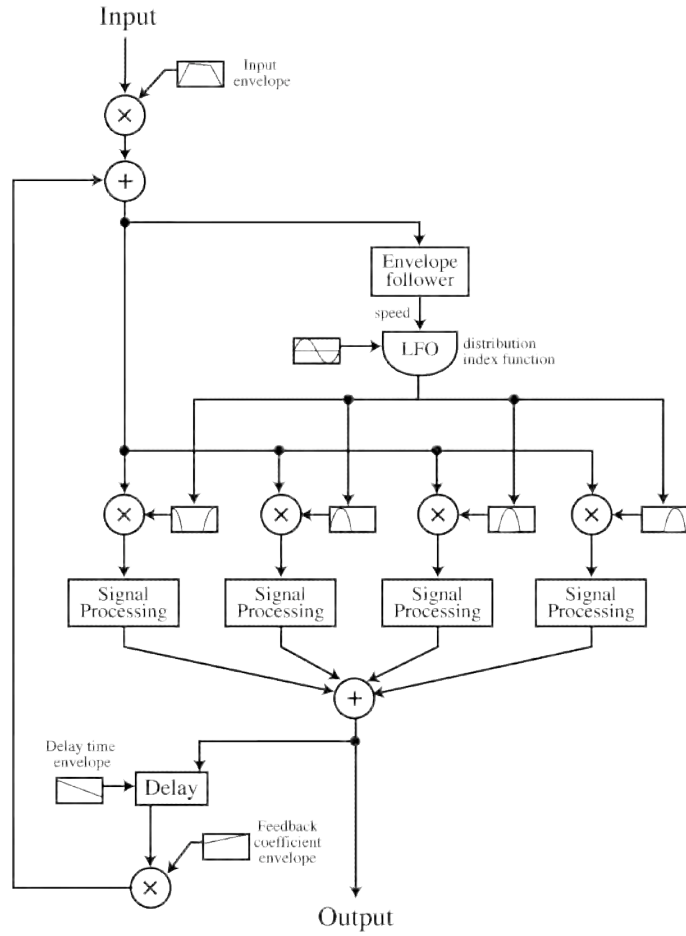


Figure 27 Block diagram of variable serial feedback network

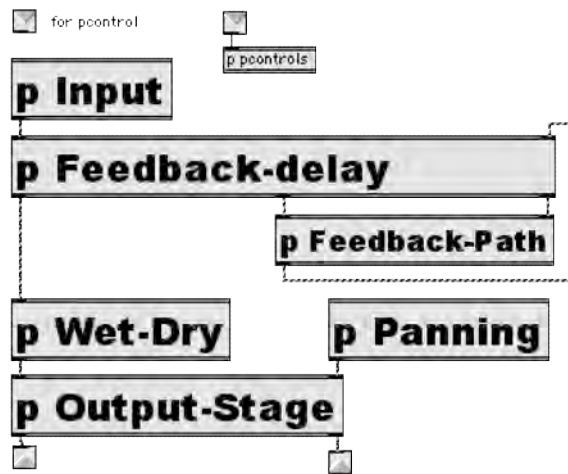


Figure 28 Accelerating delay with variable feedback

Feedback Paths

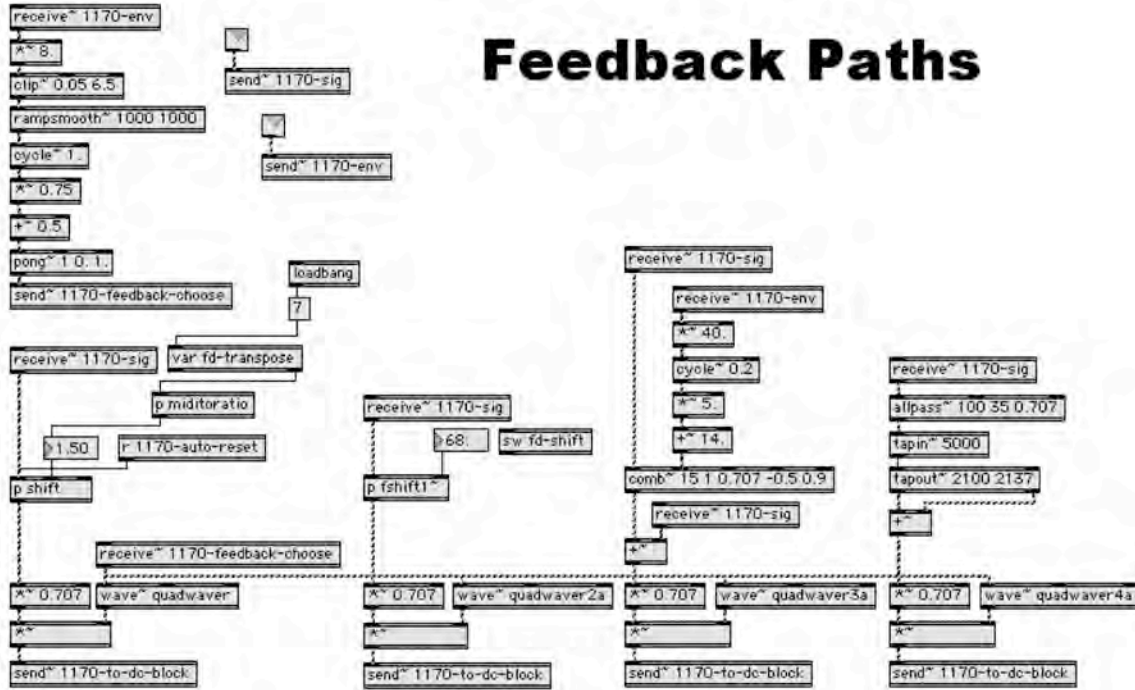


Figure 29 Subpatcher “Feedback Path” from Figure 28

The result of this is a single-minded gesture that moves relentlessly towards system overload. There are, however, processes for keeping this in check: if the amplitude envelope of the reinjected sound exceeds a certain threshold, the feedback coefficient is quickly reduced to zero, producing a cutoff, and clearing the contents of the delay line. Surprisingly, though the delay time and feedback coefficient move in a linear fashion, the rhythm produced by the feedback does not progress linearly; the variable processing stage in the feedback alters the spectrum of the incoming sound, which disperses harmonic and rhythmic resonances, resulting in a rhythmic structure that is directional, but not necessarily linear. This is visible in Figure 30, where the dotted lines indicate the perceived rhythm of the feedback delay, which increases in density over seven seconds.

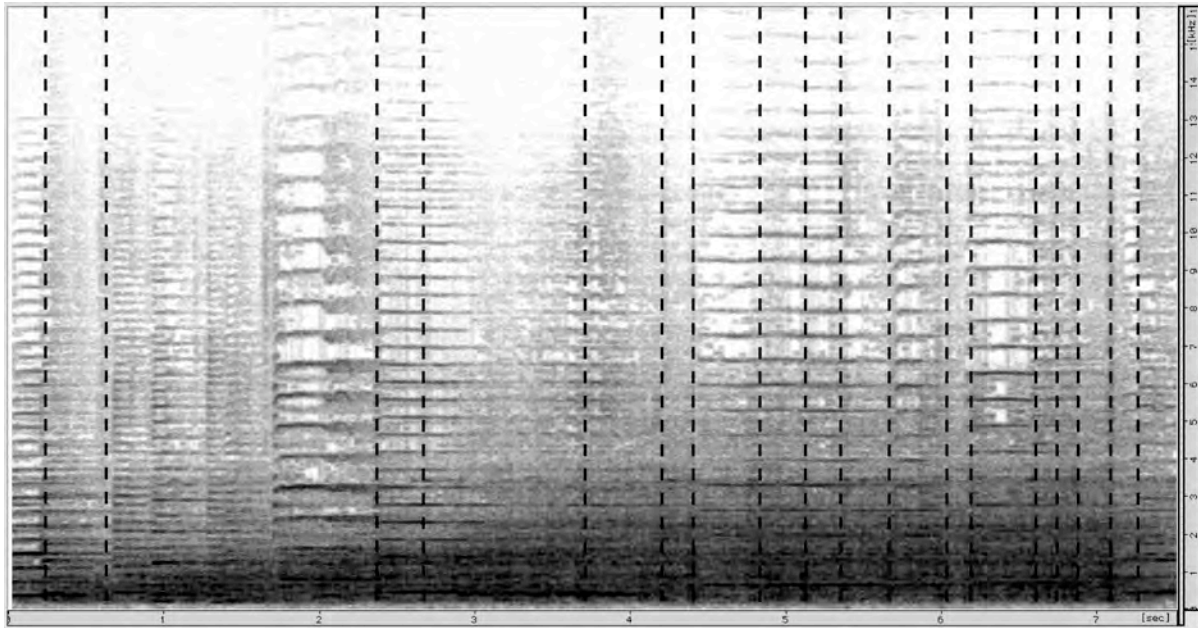


Figure 30 Sonogram of saxophone and computer, m. 99

This feedback system may be visualized as a rubber band; as the delay time decreases, and the feedback coefficient increases, the band is stretched tighter, creating tension, until that stored energy is released in a burst. New input into this system causes a momentary relaxation, before the band is again tensioned. The delay time and feedback coefficients reset to their starting positions, and begin ramping towards their respective destination values, without clearing the audio in the feedback loop. Musically, the result is a system that seems perpetually headed towards chaos, only to have its course averted at the last instant; its iteration of this cycle further increases the tension, inevitably leading to an explosive release. This property makes this device highly effective in the cadenza section, and matches the aggressive timbre of the saxophone in mm. 95-101.

Chapter 7

Contextualization and Effects on Future Work

In the rapidly changing genre of interactive music, *UNRAVEL* represents new methods for working with old materials. Innovative compositional approaches—such as parametric quotation, and the usage of electronic components in determining acoustic gestures—combine with advanced signal-processing techniques to create a work that challenges the saxophonist as well as the computer. This successful incorporation of diverse working methods suggests many avenues for compositional exploration, and new ways of bridging the gap between acoustic instruments and digital technology.

UNRAVEL has proven to be a watershed event for me, in terms of the way that I design and construct my electronics; many of the programming ideas that emerged out of necessity have also proven their worth at the conceptual level. More than simply offering additional speed improvements, these restructurings offer access to information and processes in ways that are both practical and musically significant. At the present time, some of the biggest gains remain just below the surface, awaiting future exploration.

While it is difficult to speculate about the future of interactive electro-acoustic music due to continual advances in technology, it is hoped that *UNRAVEL* will ultimately stand on its own merits, providing challenges for future saxophonists, if not future computers.

References

- AudioSculpt, www.ircam.fr, Internet; accessed 21 June 2004.
- Austin, Larry, *BluesAx*. Denton, TX: Larry Austin, 1996, digital audio disc.
- Chadabe, Joel, *Electric Sound: The Past and Promise of Electronic Music*. Upper Saddle River, NJ: Prentice Hall, 1997.
- Chopin, Frederic, Prelude in Eb minor, *Preludes for Solo Piano*. Boca Raton, FL: Masters Music, 2001.
- CsoundAV, www.csounds.com, Internet; accessed 21 June 2004.
- Felciano, Richard, *Prelude for Piano*. San Francisco: Richard Felciano, 1997.
- Harvey, Jonathan, *Advaya*. London: Faber Music, 2001.
- jMax, www.ircam.fr, Internet; accessed 21 June 2004.
- Lebaron, Anne, Denys Bouliane, "Darmstadt, 1980." *Perspectives of New Music*, 19 no. 1 (Autumn, 1980 – Summer, 1981): 426-441.
- Lippe, Cort, *Music for Flute & ISPW*. Paris: Cort Lippe, 1994, digital manuscript.
- Max/MSP, www.cycling74.com, Internet; accessed 21 June 2004.
- Mumma, Gordon, "Hornpipe," *Live-Electronic-Music*. Tzadik TZ-7074, digital sound disc.
- OpenMusic, www.ircam.fr, Internet; accessed 21 June 2004.
- PeRColate, www.music.columbia.edu/PeRColate, Internet; accessed 21 June 2004.
- PureData, www-crca.ucsd.edu/~msp/software.html, Internet; accessed 21 June 2004.
- Raumberger, Clause, Karl Ventzke, "Saxophone: History and Use." Grove Music Online, ed. L. Macy [book on-line], available from <http://www.grovemusic.com>, Internet; accessed 25 June 2004.
- Ravel, Maurice, *Bolero*. Bryn Mawr, PA: Theodore Presser, 1957.
- Risset, Jean-Claude, Computer Music: Why?" Austin: University of Texas, France-UT Institute for Interdisciplinary Studies [document online], 9 January 2004, available from www.utexas.edu/cola/france-ut/archives/risset_2.pdf, accessed 28 June 2004, p. 10.

_____. “Suds,” *Selections*. Mainz, Germany: Wergo, 1988, digital sound disc.

_____. “Voilements”, *Sax-Computers*, Daniel Kientzy. INA-GRM C2000, digital sound disc.

Roads, Curtis, *The Computer Music Tutorial*. Cambridge, MA: MIT Press, 1996.

Rowe, Robert, *Interactive Music Systems* Cambridge, MA: MIT Press, 1993.

Subotnick, Morton, “In Two Worlds,” *The Contemporary Saxophone*, John Sampen. Acton, MA: Neuma, 1992, digital sound disc.

SuperCollider, www.audiosynth.com, Internet; accessed 21 June 2004.

Winkler, Todd, *Composing Interactive Music: Techniques and Ideas Using Max* (Cambridge, Ma.: MIT Press, 1998).

Woodwind Fingering Guide, www.wfg.woodwind.org, Internet; accessed 12 October 2003

peter mcculloch

UNRAVEL

**alto saxophone
+
interactive electronics**

peter mcculloch

UNRAVEL

alto saxophone
interactive electronics +

Key:

⑧ Cues in circles are advanced by the computer operator

□ 9 Cue in squares are advanced automatically

△ Play the highest note possible
using teeth on the reed.

n niente attack or release

| subito

All accidentals are octave specific and affect only those notes which they immediately precede, except in the case of repeated notes.

Microtones are notated in quarter-sharps (\sharp) and quarter-flats (\flat).

n.b. Cues 1-3 are executed before the performance begins.

Examples are notated
at saxophone pitch.

Fingerings are transcribed from
the Woodwind Fingering Guide,
www.wfg.woodwind.org.

The first system consists of a single treble clef staff with six measures. The notes are: G4, A4, B4, C5, B4, A4. Below the staff are two rows of fingering diagrams. The first row shows fingerings for the right hand (RH) and the second row shows fingerings for the left hand (LH). The diagrams use solid black circles for fingers to be pressed and open circles for fingers to be lifted. Some diagrams include a small box with a letter 'H' inside, likely representing the harmonic key.

The second system consists of a single treble clef staff with six measures. The notes are: G4, A4, B4, C5, B4, A4. Below the staff are two rows of fingering diagrams. The first row shows fingerings for the right hand (RH) and the second row shows fingerings for the left hand (LH). The diagrams use solid black circles for fingers to be pressed and open circles for fingers to be lifted. Some diagrams include a small box with a letter 'H' inside, likely representing the harmonic key.

UNRAVEL

Peter McCulloch

Alto Saxophone in E_b

Computer (Concert Pitch)

1 Pre-initialization 2 Initialization 3 Ready, Audio On

3 10" 8" high alternate keys high alternate keys low alternate keys

pp < > *p* < > *p* < >

senza vibrato, extremely even, white tone (use open C#)

4 Soundfile

7 high alternate keys low alternate keys slow fast *mp* *mf*

5 Soundfile 6 Soundfile 7 Soundfile

13 $\bullet = 60$ 5 *pp* *mp* *p* 3 *pp* 5 *mp* *p* *pp* *p*

8 Prep. cue 9 Soundfile 10 Prep. cue 11 Flute in 12 Flute out

21 7 *mp* 3 7 *mf* 3 5

13 C4, Bb4 trigger soundfiles during cue 13 Flute in 14 Prep. cue

25

15 Flute out

16

17 C4, Bb4, and G5 trigger soundfiles during cues 17-18
Flute plays

(Soundfiles)

28

18 Ab, A, Bb, and C# turn flute on,
all other pitchclasses turn flute off

33

19 Bb turns flute off,
all other pitchclasses turn flute on

37

20 Prep. cue

21

22 23-26 Flute out, soft loops in
background, harmonizers on

(cues 23-26 advance automatically)

45 *flt.* *ord.* *subtones* *ord.* *flt.* *ord.* *subtones*

p *mp* *mf* *p* *pp* *mp* *mp* *pp*

3 3 3 3

27 Harmonizer preset 2 28 Harmonizer preset 3

51 *ord.* *flt. ord.*

mf *mp* *mp*

5 3 5 3

29 Harmonizer preset 4

56 *f* *mp* *p* *mf*

30 Harmonizer preset 4 (aggressive glissandi) 31 Harmonizer preset 6

5 3 3 3 3 3

60 *p* *f* *mf* *f*

3 3 5 6 7 7 7 6

64 *ff* *ff* **39** begin feedback

68 *pp* *mf* *p* *mf* *p* **40** electronics out **41** hit

74 *sfz* *pp* *mp* *f* *p*

77 *f* *mp* *sfz* *f* *mp* *mf* **42** hit

Cadenza
start slow, build gradually

92

p ————— *mf* | *p* ————— *mp* | *p* ————— *mf* | *mp* ————— *f*

46 Accelerating feedback delay,
C#5 triggers input

93

simile

5

47 Accelerating feedback delay,
G3 and lower pitches trigger input

94

mp < > *p* ————— *mf*

5 5 6 3 5 6 6 3 5

95

f < > *mf* < > *ff*

flt. ord. flt.

99 ord.

mp *mf* *f* *ff* *rit.* *sfz* *sfz* *sfz* *a tempo*

Forceful, with even rhythm

48 Stop input to feedback delay

101

ff *mp* *fff* *pp*

very slow, out of time

10" + ~8" ~3"

teeth on reed, as soft and steady as possible

49 Harmonizers in

50 Clear electronics, de-bus saxophone from L/R mix
(pause until saxophonist cues)

51 **52** Vocoder In
(cue 52 advances automatically)

105

6 6 6 6 5 3 3 6 6 3

Quietly finger the approximate indicated pitch to create a rustling effect on the whistle-tone, without producing a keyslap.

p quiet, pitched keyslaps

53 Vocoder preset 9

106

mp *p*

ord., senza vibrato

54 Vocoder preset 10

108

Finger the approximate indicated pitches to create a rustling effect without producing a keyslap.

~5" ord., senza vibrato ~2" ~3" ~7" ~2" ~2"

quiet, pitched keyslaps 5 5 3 *pp* *mp* *p* quiet, pitched keyslaps *ppp* quiet, pitched keyslaps

55 Vocoder preset 11

56 Vocoder preset 12

57

Fade out over 15 seconds