THE *FLUXATIONS* STOCHASTIC INTERACTIVE ALGORITHMIC MUSIC ENGINE (SIAME) AND iPHONE APP

Joshua B. Mailman
Dept. of Music, Columbia University
and Steinhardt School, NYU
New York, NY, USA
jmailman@alumni.uchicago.edu

ABSTRACT

This paper reports on aspects of the *Fluxations* paradigm for interactive music generation and an iPhone app implementation of it. The paradigm combines expressive interactivity with stochastic algorithmic computer generated sound as guided by the author's musical preferences. The emphasis is on pitch-oriented (harmonic) continuity and flux, as steered through sliders and sensors. The paradigm enables the user-performer to maximize exotic but audible musical variety by spontaneously manipulating parameters within the paradigm.

1. INTRODUCTION

The *Fluxations* Stochastic Interactive Algorithmic Music Engine (SIAME) paradigm enables the user-performer to manipulate *emergent* (macro) properties of a generated stream of musical sound, properties whose significance would be felt when they change quickly or slowly, as relevant to the perception of medium- and long-range form in fully composed music. [9, 10, 11, 12, 15] By focusing on *pitch* related aspects, the motivation is to create computer generated music that engages listening skills, habits, and affective-receptive capacities pertinent to unconventional modern music for conventional acoustic instruments. The paradigm enables musical emergent properties—which normally could only be controlled through compositional planning—to be manipulated spontaneously in real-time, providing immediate feedback, thus initiating a particular interweaving between compositional and improvisational aspects of musical expressivity, called *comprovisation.* The *Fluxations* comprovisational paradigm enables the flexibility to spontaneously create sudden attention-getting changes as well as nuanced atmospheric changes to the audible properties of the stream of musical sound. Such nuanced fluctuations are intended to serve as "vectors of transmission for feeling" as described in the metaphysics of Whitehead, and recently applied to the aesthetics of media. [20, 28]

2. INTERACTIVE MUSIC MAKING

The *Fluxations* system presented here fits within a broader context of interactive music making paradigms and technologies so it is helpful to situate it relative to previous work in this area. As a music creation paradigm, *Fluxations* probably relates most closely to what Chadabe calls *interactive composing,* which "redefines composing and performing." (p.26) [3] Since 1984—and in related ways going back to the 1970s—Chadabe describes a situation in which a composer designs a specific compositional process or interface and algorithmic technology with which a player interacts to create music is a semi-spontaneous way. For instance "the computer's function in [Chadabe's] *Solo* is to compose automatically the notes of a melody, its accompaniment chords, and other aspects of the music, and to interpret the positions of a performances hands in relation to two proximity sensors." The player triggers changes through his movement, but cannot foresee exactly what chord will be played next. He moves in response to what he hears. The music making algorithm "responds to a performer in a complex, not entirely predictable way, adding information to what a performer specifies and providing cues to the performer for further actions." [3]

2.1 Random Numbers, Complexity, and Emergence

As Chadabe describes it, the response algorithms employ random number generators, but not to make random sounding music. Rather they are used to create *complexity*, which means the rules determining events (the exact causes of individual details) remain elusive. Since this situation can obtain with real human composers or improvisers as well as with computationally implemented knowledge-based artificial intelligence models of a self aware self-organizing system, it is better to call it *complexity*. [3] Or as Eigenfeldt describes it: with a performance system that employs random numbers, the "interaction [can be] 'complex,' because, although general tendencies can be predicted, details cannot." [6] Such c*omplexity* is an important aspect of the *Fluxations* paradigm presented here because it sets the precondition for its medium of expressivity: flux of emergent properties.

2.2 Computation and Improvisation

Interactive music systems relate to improvisation because they are capable of being deployed spontaneously. They also relate because of the unpredictability inherent in the way they usually create complexity, as just discussed. In regard to his *Voyager* system, George Lewis describes processes involving random numbers as providing much of the "personality" of the system, which gives it the character of an improvisation. [8, 18] Thus even the algorithmic deployment of random numbers is viewed as improvisatory. Pierre Saint-Germier argues: Even if we experience improvisation as a creative non-deterministic process, it is actually an emerging property of a complex system of algorithms (of varying transparency). This can be *simulated* by machine algorithms that create unpredictable complexity through the use of random numbers. [18]

2.3 Comprovisation

Though *Fluxations* is not a tool for improvisation *per se*, it does enable *comprovisation*, which is a partly spontaneous, but partly planned, technological mode of expression that swerves variously between composition and improvisation in the space laid out by Chadabe. Richard Dudas discusses aspects of this, which include recognizing: (a) a balance between composition and improvisation when technology is involved; (b) that an improvised performance may follow a pre-determined form or not; (c) that computer music creation is like “slow improvisation” (a term deployed by Trevor Wishart); (d) a continuum that likens the creation of a musical instruments to the composing of a score; (e) a continuum that likens the development of computer music software to the composition of a score. [4] Similarly the term *comprovisation* is deployed by Sandeep Bhagwati, organizer of the 2010 conference: *Comprovisations: Improvisation Systems in Performing Arts and Technologies* [1].

Similarly, I describe *comprovisation* as a kind of musical creativity that relates composition and improvisation in an unprecedented fashion, one which was impossible to achieve with older technologies. *Comprovisation* is compositional in two respects: (1) it involves composing music-generating algorithms as guided by aesthetic concerns, and (2) it may involve the planned choreography of physical movements. Comprovisation is improvisational in three ways: (1) it may involve spontaneously decided physical movements; (2) planned (choreographed) movements may be spontaneously ornamented with expressive nuanced deviations; (3) the quasi-stochastic algorithm may be regarded as “improvising” since its determining of certain details cannot be predicted in advance (this component could be programmed to function according to knowledge-based improvisational rules rather than stochastically).

Comprovisation technologies, such as *Fluxations*, are not just interactive but more specifically are expressive, in that they enable a wide spectrum of smooth and abrupt changes which the performer can exert according to plan or impulse. In this respect comprovisation technologies differ from other interactive music apps for the iPhone such as Eno and Chiver’s *Bloom*, *Trope*, and *Air*. Being based on a tape-delay-loop model of musical continuity, these generative ambient music iPhone apps enable production of smooth, repetitive, soothing washes of harmonic ambience (of which *Fluxations* is also capable to some extent), but do not allow drastic sudden changes and cannot travel to remote stylistic worlds. The way *Fluxations* enables such possibilities (drastic change, smooth continuity, and stylistic diversity) will become apparent through explanation of its software design concept and the specifics of its algorithm and interactivity.

3. DESIGN CONCEPT

3.1 Stochastic Interactive Looping (SIL)

The paradigm of Stochastic Interactive Looping (SIL) does not necessarily produce music that repeats. (Actually, in this implementation the resulting music is non-repeating.)

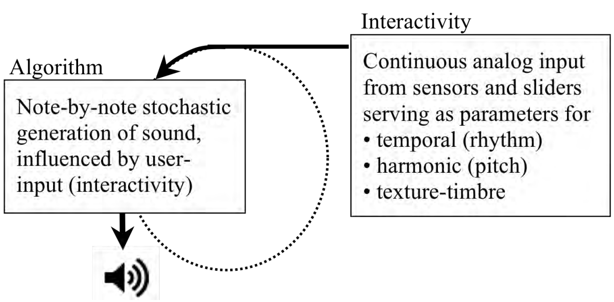

Figure 1. Stochastic Interactive Looping (SIL)

As Figure 1 shows, it simply means that the generative algorithm takes real-time user interactivity as input parameters for stochastic procedures that create each note event; then the algorithm re-executes itself, with updated input parameters. (Although it does not do so in this implementation, the algorithm could make use of its output as input, in a feedback system to create incremental processive variation in the stream of music, to be implemented in a future version of the software.)

The important point about Stochastic Interactive Looping is that it does not wait for the detection of discrete gestures from the user, but rather continuously generates a stream of pitch-based musical sound (somewhat similar to real-time granular synthesis) even when the user does nothing. [26] In this way it is unlike a standard musical instrument. Unlike with *hyperinstruments* and other kinds of sound processing interactivity, however, in the SIL paradigm, the user’s interactivity is not used to manipulate sound drawn from another source [23]. All sound is generated internally; this gives the user the most direct control possible over the macro properties emerging in the algorithmic stream of music. The stochastic nature of the algorithm ensures that all features not under user control are rendered neutral (held stable), in the sense of being indifferent to any specific tendency.

3.2 Pitch Based Continuity and Flux

Much interactive music technology (such as NIMEs) involves either a one-to-one or one-to-many mapping between discrete gestures and discrete sounds (emulating the interactivity of conventional musical instruments) or timbral manipulation of pre-recorded or real-time sampled audio, for instance using one of the Wessel/Wright metaphors for meta-mapping interaction: Scrubbing, Catch/Throw, or Dipping. [7, 22, 27] The approach of *Fluxations*, though not unrelated, is somewhat different, particularly in regard to continuity and flux as projected through the use of musical pitches aggregating into harmonic regions.

One of the premises of *Fluxations* is that nuanced shift of harmonic region has the potential to be expressively powerful, as occurs in much music for conventional instruments—whether tonal or atonal, minimalist or maximalist. Therefore this paradigm enables the user to navigate through harmonic regions as shifting emergent properties of a continuous stream of sound. Specifically it enables the user to efficiently, smoothly, and systematically alter the harmonic qualities of the generated music to quite drastic extremes, inhabiting diverse expressive realms, either over a long period of time or on the drop of a dime. (See sections 3.4 and 3.5 below.)

3.3 Stochastically Generated Music

The use of stochastic procedures in music is not new, since it was pioneered by Xenakis, Tenney, and others. [24, 25, 26, 29] For the present paradigm the decision to generate sounds solely within a stochastic looping algorithm is driven by specific aesthetic concerns pursued through statistical means. This is because, through analytical study of repertoire composed for conventional instruments, the author has identified various emergent properties as being interesting bearers of musical form over medium and long ranges of time. [9, 10, 11, 12, 15] Therefore the aesthetic concerns are to exploit as much as possible the expressive potential of the emergent properties. Although such properties are important facets of compositional expressivity, they are nearly impossible to manipulate improvisationally through the conventional discrete gesture - to - discrete sound mapping of musical instruments, even those making use of technological interfaces. This is because an emergent property cannot *emerge* except on the condition of sufficient *complexity*. Achieving complexity in a controlled manner with conventional instruments takes intricately planned coordination. With the conventional note-by-note interactivity of conventional instruments, sufficient complexity cannot be achieved in a controlled manner in improvisation. Such properties *can* however be manipulated spontaneously with a system such as described here. Thus this system provides to improvisation a certain kind of control usually available only to composition.

Stochastic procedures partake in this because they enable the articulation of a complex space of sounds that create a mood, for instance by projecting the tones of a rich harmony, so its smooth or abrupt shift to a new harmony is experienced as compellingly nuanced. Such are examples of emergent properties, analogous to temperature or humidity, whose flux is often felt gradually over a duration of time. (The *computational* aspects of music listening are brought to bear in this case, such that differences of intensity of emergent qualities are sensed [13].)

Employing a controlled system of stochastic procedures makes it possible for the user of the interactive system to neutralize (hold stable) several emergent properties so that the flux of another can be heard as the only trend occurring over a period time.[1] This way, emergent properties (such as harmonic region and pitch height or attack density and rhythmic volatility) that might vary together when improvised can instead be disassociated from each other as separate properties, which vary independently.

For instance, the pitch-class set can be transposed (rotated on the pitch-class clock) without increasing or decreasing the overall pitch height. In improvisation, such an maneuver would be difficult to achieve while maintaining sufficient complexity, unless the improviser had practiced multiple variations of the specific maneuver with every pitch-class set, and honed such maneuvers to great speed and facility. Without requiring such demanding preparation, the stochastic algorithm enables the user to isolate the flux of intensity of one emergent property in relation to others, to explore exotic expressive territories that have been potent in composition but which have been impossible to coordinate in improvisation.

3.4 Interactivity

In accordance with the aesthetic concerns just asserted, the kind of interactivity envisioned for *Fluxations* diverges as well. A somewhat detached approach is encouraged, to put the focus on changes to holistic aspects of the sonic experience, rather than instantaneous reaction. Nevertheless *Fluxations* is capable of sudden attention-getting sonic effects as well; long gradual subtle change as well as abrupt obvious change are within the expressive possibilities of the paradigm. These extremes are not bifurcated into separate user interface elements or software implementation categories but rather arise organically according to how drastically and quickly the user adjusts any particular continuous algorithmic input parameter.

The fact that, in this paradigm, there is no possible way to instantaneously initiate a specific sound, or effect the exact timing of a sound, can frustrate the expectations of some users. Yet one does adapt. And this is to the benefit of exploiting the expressive potential of the paradigm. The stochastic nature of the paradigm's algorithm lets the interactive control of emergent properties exert relatively more influence than the more direct approach allows.[2] For instance, if it permitted instantaneous initiation of sounds such as occurs with processing sampled or live sounds from another source or allowing the user to play discrete sound events by tapping for instance. The particularities of the *Fluxations* paradigm promote their own particular kind of interactivity, which supplements the interactivities promoted by other musical expression technologies.

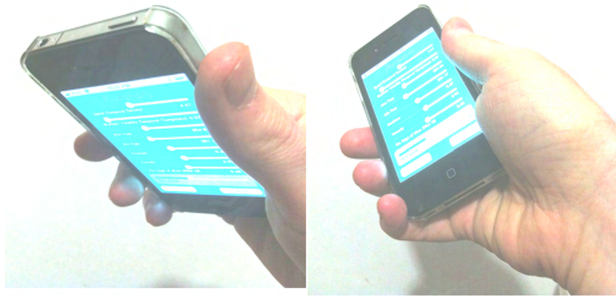

Figure 2. Input from the iPhone's accelerometer enables expressive interactivity through tilting motions

By controlling some of these expressive properties through tilting motion, as shown in Figure 2, one is encouraged to experience the resulting sonic flux as embodied phenomena, which helps make unconventional kinds of musical flux more palpable, and thus more easily heard, which is the didactic aspect of this interactivity. [15]

The parameters mapped to tilting are chosen partly to exploit the physicality of the interactive medium (wrist motions) which can be smooth or abrupt. To exploit this, the parameters mapped to tilting should seem musically significant whether changed smoothly or abruptly. (Basic features such as the pitch, loudness, or duration, of individual notes are usually too facile to project such significance.)

Also, in choosing the mappings, an attempt is made to balance expressive capability that is more direct (immediate) with that which is less direct (more nuanced). For more direct expressivity, the low-pass filter is chosen because it corresponds to the vocality of human verbal expression. The sound colors of vowels of human speech are created by the frequency filtering effect of opening and closing the mouth, which makes filtering a powerful resource for musical organization, as explained by Slawson in his book on *sound color*. [21]. Because humans perpetually effect similar timbre changes with the mouth for the purpose of verbal expression, the low-pass filter serves well as a direct vehicle of expression in an interactive music system. This is probably also why the mapping of low-pass filter to tilting is exploited so successfully in the standard *wah-wah* pedal effect for electric guitars. This also makes it relatively comfortable for electric guitarists to exploit forward-backward tilting similarly on the iPhone.

For less direct, more nuanced, expressivity, pitch-class transposition is chosen. This is because pitch-class transposition enables systematic and reliable expression of mood-shift-like behavior, either sudden or smooth. It can be used expressively in that it behaves in a consistent fashion, which musical experts know explicitly and which most music listeners feel implicitly. Pitch-class transposition projects

1 This is the same statistical principle that motivates the use of randomization in biomedical, psychological, and sociological studies: to eliminate the effect of a variable, randomize it.

2 The interactive control is significant because it makes the ontological assertion of these emergent properties more concrete. In other words, it enables kinesthetic (embodied) experience to sharpen and extend the listener phenomenology that identifies these properties as musically significant. Thus it contributes to the *cybernetic phenomenological* impact on music ontology. [14]

mood-like changes somewhat analogous to changes of key or chord. Such change is underused in interactive music systems. (Certain mapping possibilities were eliminated for pragmatic reasons: since the sensitivity of the iPhone's accelerometer well exceeds the steadiness of one's hand—especially when operating other controls—it proves cumbersome to assign tilting to pulse speed; with such a mapping it is difficult to maintain a steady pulse.)

3.5 Variety

One of the goals is to promote expressive power through variety of musical style—not variety understood as an array of crisp categories, but rather as an inclusive multidimensional continuous space of possibilities. Thus, depending entirely on how the user manipulates the interface, the music generated by the algorithm ranges from a soothing ambient style to a wild angular avant garde style. Likewise its temporal character ranges from relaxed flowing seemingly without pulse to a taut rhythmically propulsive style. Such diversity of "style" is not accessed through different style category options, but rather arises organically from continuous change of algorithm input parameters.

Another aspect of variety is indirect; it is that certain parameters that might ordinarily be associated with each other are instead disassociated (see 3.3 above), so they can be manipulated independently of each other. For instance *transposition* is implemented in a pitch-class (circular) space so it is manipulated independently of high-verses-low pitch range, which can therefore serve as independent agents of expression.[3]

Figure 3. iPhone user interface (visible portion)

4. ALGORITHM AND INTERACTIVITY

The visual portion of the interface of the iPhone implementation is shown in Figure 3. Although this generic iPhone interface looks superficially like a graphic equalizer, it is instead a series of sliders that permit smooth control of input parameters that act as seeds and bounds in the stochastic music generating algorithm. It is a design decision to put the visual focus on the quantitative parameters since they are meant to serve as vehicles of expression in this paradigm. This also focuses the user's attention on the dynamism of the sound rather than on visuals.

Under most circumstances, change of any of these parameters is relatively easy to hear because there is no external audio input source whose features might otherwise dominate over the user's manipulations (as would occur with an RJDJ app for instance). The *Fluxations* parameters listed in Table 1 are grouped according to their role in the stochastic algorithm and labeled according to their UI controls. Most are controlled by sliders; two are controlled through continuous accelerometer readings. (A future version of the software may allow the mappings to be customized, so users can adapt the setup to their own needs and preferences.)

	Parameter	*UI control*
Timing	Speed (attack density)	slider
	Rhythmic volatility	slider
Pitch	Pitch-class (tone chroma) transposition	lateral tilt (accelerometer)
	Pitch space	slider
	Maximum	slider
	Minimum	slider
Texture-Timbre	Hardness (marimba loudness)	slider
	Viscosity (relative duration of additive synthesized sound)	slider
	Low-pass filter	forward tilt (accelerometer)

Table 1. Algorithmic Parameters and their UI controls

Any number of these parameters can be manipulated at the same time, for instance by simultaneously using two or more fingers on separate sliders or by tilting the iPhone while moving its sliders. One can also tilt at an angle affecting both lateral and longitudinal axes at the same time. Any combination of such moves is possible. This is one of the expressive advantages of the software: it permits sudden drastic change through the same controls that enable smooth change.

4.1 Pitch (Harmony)

The pitch collection (harmonic) model generates pitch-class sets by the cycle of 5ths, using the M_7 operation (multiply by 7 mod 12). [5,17] For example taking a set of consecutive numbers {0,1,2} and multiplying each by 5 produces pitch collections rich in perfect 4ths and 5ths, in this case the "hollow," "Baba O'riley," "sus 4" trichord {C, G, D}; expanding the source set to {0,1,2,3,4} and multiplying by 5 produces the pentatonic scale collection {C,G,D,A,C}, as shown in Figure 4, and expanding to 12 produces the full chromatic. Thus by varying the source set size (called *Pitch Space* in the UI) it is possible for the user to efficiently, smoothly, and systematically alter the harmonic qualities of the generated music to quite drastic extremes. Figure 4 illustrates how the various pitch spaces are generated by increasingly larger cycles of P4/P5 intervals.

[3] That new fluctuating properties can project form or expression is a significant aspect of technological music, in that it forges new resources to articulate narrative, as I discuss in regard to Lucier's *Crossings* (1982) and other works. [16]

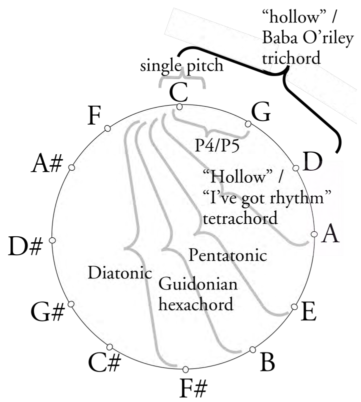

Figure 4. Hollower to fuller harmonic) spaces generated by the circle of 5ths (*Pitch Space* ranges from 1 to 12, plus a 13th which designates all microtonal pitches.)

Tilting the iPhone laterally, as in Figure 5, has the effect of *transposing* to different harmonic regions, represented as a pools of pitches (shown in Figure 6) from which the stochastic algorithm chooses. The farther the tilt, the more remote the harmonic region. Tilting by 15° transposes the pitch selection pool (harmonic region) {C,G,D} to {G,D,A} —a somewhat subtle rather than drastic change. Further tilting, such as by 90°, produces a more drastic change in harmonic regions: {C, G, D} becomes {F#,C#,G#}. A variety of degrees of harmonic change are thus enabled, accessible in predictable ways, thus making change of harmony available as a vehicle of expression.

Figure 5. Lateral tilting rotates the pitch-class pool on the circle of 5ths. It also alters the hue of the background color.

Each pitch in the continuous stream of sound is chosen individually according to the following model, which combines stochastic procedures with interactive input. Pitch-class transposition is derived through this equation:

$$T = \frac{mod_{180}(Tilt°)}{15} \quad (1)$$

Thus, just as the pitch selection pools (harmonic regions) are derived from the cycle of 5ths, so too is the transposition; it is a rotation on the circle shown in Figure 4. The M_7 (multiply by 7 mod 12) operation is used.

$$pc = integer(mod_{12}(rand(0, PitchSpace\text{-}1) \times 7 + T) \quad (2)$$

The register of the pitch is chosen independently of pitch-class; it is chosen randomly from within the bounds set by the user.

$$register = rand(min, max) \quad (3)$$

The computation of pitch combines pc with register.

$$pitch = (register \times 12) + pc \quad (4)$$

The effect of tilting is best shown in Figure 6. Because the pitch pools (harmonic regions) and the transposition (by tilting) are based on the same interval (P4/P5), the navigation is efficient, smooth, and systematic. When *Pitch Space* is set to 3, tilting by only 15° holds two of three pitch-classes invariant; tilting by 90° produces a pool of entirely different pitch-classes; tilting by 180° returns to the original pitch class pool.

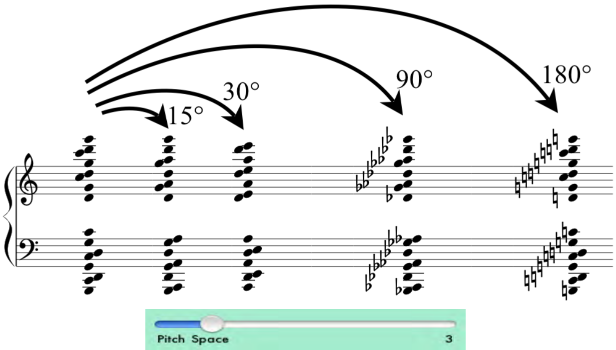

Figure 6. With *Pitch Space* set to 3, these are some of the pools of pitches (harmonic regions) from which the stochastic algorithm chooses. Different degrees of rotation transpose such that pitch pools holding more or fewer pitch-classes (tone chroma) invariant are used, thus creating varying degrees of continuity to be used expressively.

The effect of tilting varies significantly according to the *Pitch Space* chosen, because of their intervallic properties. Generally the larger the *Pitch Space,* the more harmonic continuity there is per each degree of tilt. Tilting with a *Pitch Space* of 1 (a single pitch-class) produces an entirely new pitch class to the exclusion of the original one (switching from C to G). Yet on the other extreme, with a *Pitch Space* of 12 even a 90° tilt produces no change whatsoever since all pitch-classes on the circle (the full chromatic) are already present. The interactivity model is most interesting with intermediate *Pitch Spaces* such as 3, shown in Figure 6, because lesser tilts swap out some but

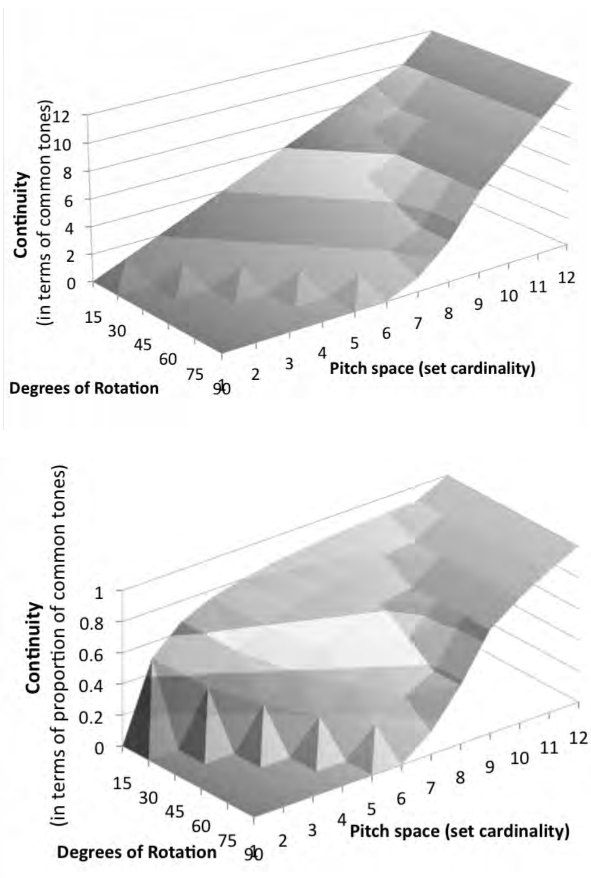

Figure 7. The degree of harmonic continuity per each degree of tilt increases with the size of the *Pitch Space*.

all not pitch-classes, thus allowing the user to manipulate degrees of continuity in the sounding stream. In this way, the smoothness of physical motion (lateral tilting) translates to smoothness in the flux of harmony, to an extent that is itself controlled by the *Pitch Space* parameter, which indirectly but systematically influences the number of pitch-classes held invariant (common tones) when transposition occurs.

In terms of number of common tones (top) and proportion of common tones (bottom), Figure 7 graphs the relation between *degree of rotation* and *Pitch Space* in regard to *harmonic continuity*.

Outside the model shown, an additional expressive resource is offered when *Pitch Space* is set to 13. This triggers a new mode by which pitches are chosen outside the 12-based equal tempered system; instead totally random frequencies are chosen (though still within the minimum and maximum range set by the user).

The temporal responsiveness of the tilting is immediate to within the constraints of the looping paradigm, which means the responsiveness depends on the speed (temporal density) setting. The loop duration is set to a default of four notes (four pulses). This means that for 90% of the pulse speed range, the transposition tilting responds in less than one second. Therefore, provided the pulse speed is not extremely slow, and provided the pitch space is hollow enough (has relatively few pitch-classes), the user can suddenly move to a completely different harmonic area and back, just by two quick twists of the wrist.

4.2 Timing, Scheduling, Rhythm, Rests

The rhythmic strategy is to enable the user to control the stream's *attack density* (*temporal density)* and the *rhythmic volatility* independently of each other (as shown in Figure 8.) This is done by defining an underlying pulse grid that could be adjusted according to either user-controlled changes of *attack density (temporal density)* or *rhythmic volatility* (*temporal clumpiness)* while holding the other stochastic parameter constant.

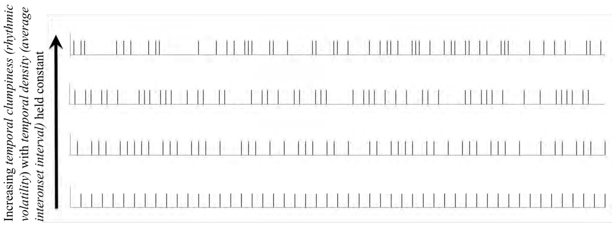

Figure 8. Increasing *rhythmic volatility* with *attack density* held constant

The scheduled timing of each note event (and thus the resulting rhythm) is determined as follows, such that *rhythmic volatility* and *attack density* are controlled independently of each other.

$$IOI = interonset\ interval \tag{5}$$

$$Density_{Pulse} = \frac{1}{IOI_{Pulse}} \tag{6}$$

$$Density_{Attack} = \frac{1}{Avg(IOI_{Pulse})} = Density_{Pulse} \times Prob(Attack) \tag{7}$$

$$Volatility_{Attack} \approx \frac{1}{Prob(Attack)} \tag{8}$$

$$Adjusted\_IOI_{Pulse} \approx \frac{IOI_{Pulse}}{1 + Volatility_{Attack}} \tag{9}$$

The stochastic nature of the rhythm is realized by presenting an absence of attack on some pulses whenever the user adjusts the rhythmic volatility to a value greater than zero. The probability of an attack on a given pulse is inversely proportional to the rhythmic volatility as adjusted by the user. At slower tempos (lower attack density settings) this creates an irregular energetic syncopated rhythm; at faster tempos (higher attack density settings) this creates a frenetic erratic series of sounds, a virtually uncontrolled chaos—though the effect depends a lot on how the harmonic and timbral-textural parameters are set at any given time. When the texture is sufficiently fluid, increasing *rhythmic volatility* creates a preponderance of rests.

4.3 Texture and Timbre

Viscosity varies such that durations may be much shorter than interonset intervals or much longer.[4] Thus the range from staccatos separated by rests on the one hand to a sustained blend, a pile-up of overlapping notes, on the other, can be achieved, as in Figure 9. The staccatos with rests (bottom) are more fluid (not viscous) whereas the blended texture (top) is less fluid (more viscous). The continuum this varies on is manipulated by a continuous slider in the iPhone app implementation.

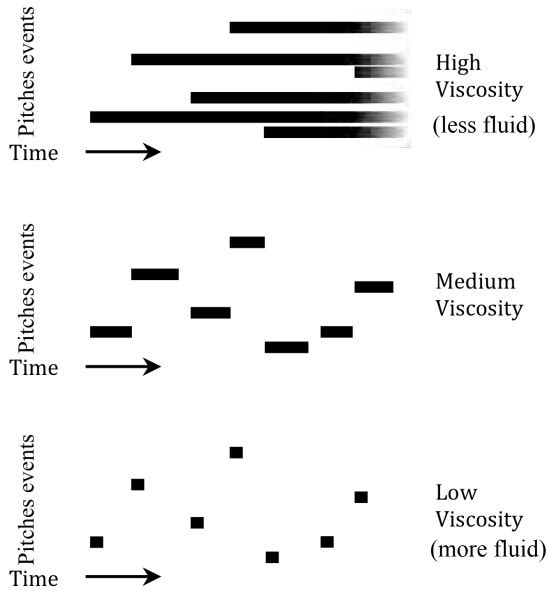

Figure 9. Varying degrees of *viscosity* influence the texture significantly. (Note that all three examples have the same *attack density* and *rhythmic volatility*.)

Timbre is varied in two ways, one pertaining to synthesis and the other from filtering. The default timbre is based on a custom designed stochastic additive synthesis model involving exclusively sine waves at randomly higher harmonic frequencies. It creates a soft flute-like sound. This timbre is supplemented by a marimba timbre controlled by a slider labeled *Hardness*. Since the marimba is staccato by default, in order to maximize timbral variety, the actual heard viscosity is

[4] *Viscosity* is an emergent property of a passage of note-based music, relating to texture. Previously I have defined it computationally (statistically). [10, 12, 15] For purposes here it can be understood as comparable to average grain duration in granular synthesis.

scaled inversely by the *Hardness*, so that when *Hardness* is at a maximum, viscosity is minimized; though the reverse is not the case.

A low-pass filter (affecting both the marimba and additive synthesis sounds) is controlled by forward-backward tilting, as read by the iPhone's built-in accelerometer. The filter is fully open when the iPhone is held nearly upright (70°) and completely closed when tilted 20° beyond the horizontal plane. Additionally, the background color of the interface darkens in proportion to the tilting; thus the darkening of the sound color is matched visually. From the user's perspective, the interactivity of the low-pass filter is just like an electric guitar *wah-wah* pedal, a device recognized for its expressivity. (The *Q* of the filter—the suddenness or sharpness of the filter cutoff edge—is controlled by a slider called *The Edge of Wow*.)

5. SOFTWARE IMPLEMENTATION

The sound synthesis engine is RTcmix. [19] Each loop is realized as an execution of an RTcmix *score file*, which is a script designating the scheduling of sonic events. The script, its looping, and its interactivity were prototyped with the rtcmix~ object for Cycling '74's Max/MSP, using sliders for parameter control. [2] Then a slightly adapted version of the same RTcmix score script was used in an Apple Xcode project with iRTcmix, which is the iPhone implementation of RTcmix, developed by Brad Garton and Damon Holzborn.

The additively synthesized sound employs RTcmix's WAVETABLE function and the marimba sound employs a physical model based synthesis function MMODALBAR. The low-pass filter is implemented with RTcmix's *pfield* facility, which permits the filter to be changed in real-time even within each loop, after score script scheduling, thus instantly altering the sound.

The raw accelerometer data of the iPhone is low-pass filtered and transformed by the arctangent function in order to serve as control parameters for pitch-class transposition (lateral tilt: x-axis) and the *pfield* enabled audio frequency low-pass filter (forward-backward tilt: y-axis). The low-pass filtering of the raw accelerometer data is what enables the orientation (rather than sudden movements) of the iPhone to serve as a control parameter thus realizing the interactivity philosophy advanced by this paradigm as explained in sections 2 and 3 above.

6. CONCLUSION

Fluxations is both an approach to music interactivity and a technological paradigm for realizing this approach through Stochastic Interactive Looping (SIL). It enables the user-improviser to smoothly, efficiently, and systematically manipulate (in this case through an iPhone interface) the intensities of exotic emergent properties, which usually can only be isolated through painstaking composition. These pertain to the perception of mood shifts heard over medium- and long-range form in composed music. The paradigm facilitates *comprovisations,* which range quite drastically in style. It also nicely supplements other kinds of NIMEs, which offer more direct control of individual sounds.

The author is preparing a commercial release (on Apple's App Store) of the *Fluxations* iPhone app. The author is also already engaged in additional interfaces for the *Fluxations* paradigm involving whole body movements (in collaboration with Sofia Paraskeva). There are in principle an infinite variety of emergent properties of music [9]. More of them (such as *durational diversity, vertical cellularity (homophony)*, and *pitch-permeation*) are gradually being added to this technology.

Acknowledgments

The author thanks Brad Garton and Damon Holzborn for their tutelage and for example code on which aspects of the iPhone implementation are based.

7. REFERENCES

[1] Bhagwati, S. *Comprovisations: Improvisation Systems in Performing Arts and Technologies,* Special Focus Symposium at the *22nd International Conference on Systems Research, Informatics and Cybernetics Baden-Baden* (Germany), Aug 2-6, 2010 *http://www.h-net.org/announce/show.cgi?ID=173733*

[2] Cycling '74: http://cycling74.com/

[3] Chadabe, J. "Interactive Composing: An Overview" *Computer Music Journal*, 8/1, 1984, pp. 22-27.

[4] Dudas, R. "'Comprovisation': The Various Facets of Composed Improvisation within Interactive Performance Systems. *Leonardo Music Journal* 20, 2010, pp.29-31.

[5] Eimert, H. *Lehrbuch der Zwölftontechnik*. Wiesbaden: Breitkopf & Härtel. 1950.

[6] Eigenfeldt, A. "Encoding Musical Knowledge in Software Performance Tools" *SEAMUS* 2007, Iowa State University (March 2007)

[7] Hunt, A. and Wanderley. M. 2002, "Mapping Performance Parameters to Synthesis Engines." In *Organised Sound*, 7, 2 97–108, 2002.

[8] Lewis, G. Program note for *Voyager*, 1993. http://www.ubu.com/sound.lewis.html

[9] Mailman, J.B. "Introduction to Temporal Dynamic Form Theory and Analysis" Colloquium (by invitation), Harvard University, Department of Music, April 2009

[10] Mailman. J.B. Temporal Dynamic Form in Music: Atonal, Tonal, and Other. Ph.D. Thesis, Eastman School of Music, University of Rochester, Rochester, NY, 2010

[11] Mailman, J.B. "Dynamic Form (Flux of vertical cellularity in Isaac's Quis Dabit and flux of pitch freshness)," Northeast Music Cognition Group, Yale University, May 2010.

[12] Mailman, J.B. "Emergent Flux Projecting Form in Ruth Crawford Seeger's Quartet (1931)," Analysis of Music by Women Composers Session, Committee on Status of Women, Society for Music Theory, Indianapolis, November 2010 and Seventh International Conference on Music Since 1900 / Society for Music Analysis Conference, Lancaster, U.K., July 2011.

[13] Mailman, J.B. "Seven Metaphors for (Music) Listening: DRAMaTIC" *Journal of Sonic Studies*, v.2, 2012. http://sonicstudies.org

[14] Mailman, J.B. "Cybernetic Phenomenology and Music Ontology" *Music: Cognition, Technology, Society*, Cornell University, May 11-13, 2012.

[15] Mailman, J.B. "Interactive Computer Simulation for Kinesthetic Learning to Perceive Unconventional Emergent Form-bearing Qualities in Music by Crawford Seeger, Carter, Ligeti, and Others" *Proceedings of the International Conference on Music Perception and Cognition (ICMPC) and European Society for the Cognitive Sciences of Music (ESCOM)*, Thessaloniki, Greece, July 2012.

[16] Mailman, J.B. "Agency, Determinism, Focal Time Frames, and Processive Minimalist Music" *Music and Narrative after 1900*. Michael Klein and Nicholas Reyland, eds. Bloomington, IN: Indiana University, 2012.

[17] Morris, R. *Composition with Pitch-class sets: A Theory of Compositional Design*. New Haven: Yale University Press. 1987.
[18] Saint-Germier, P. "Can computers improvise?" *ImproTech Paris-New York 2012: Improvisation & Technology*. New York University, May 17.
[19] RTcmix: http://music.columbia.edu/cmc/RTcmix/
[20] Shaviro. S. *Without Criteria: Kant, Whitehead, Deleuze, and Aesthetics*. Cambridge, MA: MIT Press. 2009.
[21] Slawson, W. *Sound Color*. Berkeley: University of California Press. 1985.
[22] Tanaka, A. "Mapping Out Instruments, Affordances, and Mobiles." Proceedings of the 2010 Conference on New Interfaces for Musical Expression (NIME 2010), Sydney, Australia.
[23] Tanaka, A. "Sensor-Based Musical Instruments and Interactive Music." In Dean, R. (Ed.) *The Oxford Handbook of Computer Music*, Oxford University Press, (2009) 233-257.
[24] Tenney, J. "Sound Generation by Means of a Digital Computer". *Journal of Music Theory* 7/1, (1963), 25-70.
[25] Tenney, J. "Computer Music Experiments, 1961-1964". *Electronic Music Reports* #1. Utrecht: Institute of Sonology: (1969). 23-60.
[26] Truax, B. "Composing with real-time granular sound" *Perspectives of New Music* 28(2), (1990), 120-34.
[27] Wessel, D. and Wright, M. "Problems and Prospects for Intimate Musical Control of Computers." In *Computer Music Journal*. 26, 3. 2002.
[28] Whitehead, A. N. *Process and Reality*. New York: The Free Press. 1929/1978.
[29] Xenakis, I.. *Formalized Music: Thought and Mathematics in Composition* (Harmonologia Series No.6). Hillsdale, NY: Pendragon Press. 1971/2001.

8. APPENDIX

To hear the author's comprovisations made with the *Fluxations* technology, and learn more about his work, visit his website: http://www.joshuabanksmailman.com