Techniques to Foster Drum Machine Expressivity

Jeff A. Bilmes *
<bilmes@amt.mit.edu>
Perceptual Computing Group
MIT Media Laboratory
Massachusetts Institute of Technology
Cambridge, MA 02139

Abstract

Electronic drum machines need algorithms to help them produce "expressive-sounding" rhythmic phrases. In [Bilmes, 1992], I claim that three perceptually separate elements characterize percussive rhythm: metric content, tempo variation, and deviations (formerly called event-shifts). Herein, I demonstrate that algorithms based on this model may considerably facilitate reproduction of expressive rhythm. I describe one such algorithm which extracts the separate elements from a percussive performance. The performance is then resynthesized with varying degrees of tempo variation and deviations. Without the deviations, the performance sounds mechanical. With them it sounds rich and alive. Consequently, I claim that we should begin a concentrated study on the separate elements characterizing percussive rhythm, particularly deviations. To this effect, development has begun on a graphical drum machine program with which deviations may be explored.

1 Introduction

To err is human. Yet most users of drum machines and music sequencers strive to eliminate "errs" in musical performance. In fact, some computer musicians[1] never turn off the quantize option, destroying forever "flaws that make the performance sound sloppy." At the same time, other computer musicians complain about the mechanical quality of computer music. They call for the development of techniques which would enable computers to sound better, i.e. more "human."

There are two orthogonal criteria of performance. The first is sheer technical proficiency. Clearly, computers have long surpassed humans on this axis. The other is expressivity, something more elusive, something that gives music its emotion, its feeling, its joy and sorrow, and its humanity. Music exudes humanity; computer music exudes uniformity. This, I strive to eliminate.

*Author's current address: Computer Science Division, EECS Department, U.C. Berkeley, Berkeley, CA 94720. <bilmes@cs.berkeley.edu>

[1] I use "computer musician" to refer to anyone who uses a computer to create music, and "computer music" to refer to music created thereof.

2 Rhythm

There is no doubt that music devoid of both harmony and melody may still contain considerable expression. Percussive music is a case in point, as anyone who has truly enjoyed traditional music from Africa, India, or Central or South America knows.

Unsuitable for percussive music however, previous representations of expressive timing ([Clines, 1977], [Jaffe, 1985], [Schloss, 1985], [Repp, 1990], [Wessel *et al.*, 1991], [Anderson and Kuivila, 1991], [Anderson and Bilmes, 1991], and [Desain and Honing, 1992]) can all (with the exception of [Desain and Honing, 1992]) be reduced to tempo variation.

In [Bilmes, 1992], I introduce a new model of rhythmic expressivity. Specifically, I state that beat-based rhythm can be characterized by three components: metric structure, tempo variation, and deviations (formerly called event-shift models). In this paper, I argue that deviations are most important for percussive and non-Western music, and that they are indispensable study for any drum machine architect wishing to create an expressive sounding product.

When we listen to or perform music, we often perceive a high frequency pulse, frequently a binary, trinary, or quaternary subdivision of the

musical tactus[2]. What does it mean to perceive this pulse, or, as I will call it, *tatum*[3].

Perceiving the tatum does not necessarily imply a conscious ticking in the mind, like a clock. Often, it is an unconscious and intuitive pulse that can be brought into the foreground of one's thought when needed. Perceiving the tatum implies that the listener or performer is judging and anticipating musical events with respect to a high frequency pulse. It is a natural perception, perhaps similar to the illusory contour in the well known picture on the front cover of Marr's book [Marr, 1982].

The tatum is not always explicitly stated in a piece of music. How, then, is it implied? The tatum is the lowest level of the metric musical hierarchy. Often, it is defined by the smallest time interval between successive notes in a rhythmic phrase. For example, two sixteenth notes followed by eighth notes would probably create a sixteenth note tatum. Other times, however, the tatum is not as apparent; then, it might best be described as that time division which most highly coincides with all note onsets.

The tatum provides a useful means of defining two components of beat-based rhythm, tempo variation and deviations. Tatums pass by at a certain rate, and may be measured in tatums per minute. Tempo variation may be expressed as tatum duration (in seconds) as a function of tatum number. Similarly, deviations may be expressed as deviation (in seconds) as a function of tatum number. That is, a deviation function determines the amount of time that an event metrically falling on a particular tatum should be shifted when performed. Note that tempo variation is defined per ensemble, whereas deviations are defined per performer.

A common question asked is, if we assume tempo variation is also per person, why not just use tempo variation to represent deviations in a performance? That is, is there mathematically any difference between tempo variation and deviations? The answer is no, there is no *mathematical* difference. Either can represent performance timing. There is, however, a perceptual difference.

When listening to or playing in a drum or jazz ensemble, there are times when the tempo is considered constant, even though members are playing off the beat. Notice, the concept of being "off the beat" suggests that there is deviation from some tempo generally followed by the ensemble. There is no concept, however, of individual members of the ensemble slightly adjusting their own personalized tempo. Furthermore, the tempo change needed to represent deviations in a typical performance would be at an unnaturally high frequency and high amplitude. Imagine, at each successive tatum, varying the tempo between 2000 and 354 tatums per minute (Figure 1A). Perceptually, this seems quite impossible. However it seems quite reasonable to assume that a person could, at a constant tempo of 300 tatums per minute, play every other note 15 percent of a tatum early (Figure 1B). In other words, although they might be mathematically equivalent, tempo variation and deviations are different in more important ways – they are distinct functionally and conceptually.

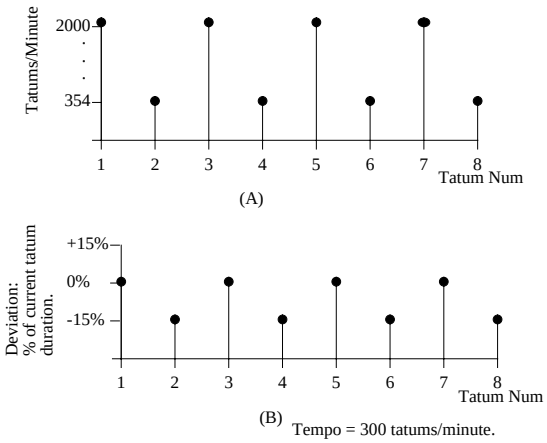

Figure 1: Equivalent Tempo Variation and Deviation Representations

The previous paragraph suggests that there must be some (per person) upper limit on tempo oscillation frequency. That is, any performance variation not accounted for by tempo variation because of its high frequency must be owing to "deviations." The following algorithm was developed with this assumption in mind.

3 Timing Extraction Algorithm

In [Bilmes, 1992], I point out the necessity of an algorithm that extracts deviations from a performance. Presented herein is one that extracts the quantized score, the tempo variation, and the deviations. The input to the algorithm is a list of attack times.

The algorithm is given complete metric knowledge of the performer. That is, it knows the time signature, the number of tatums per beat, the number of beats per measure, and where the

[2]The stroke of the hand or baton in conducting, or, often, the main quarter note beat

[3]When I asked Barry Vercoe if this concept had a term, he felicitously replied "Not until now. Call it *temporal atom*, or *tatom*." So, in honor of Art Tatum, whose tatum was faster than all others, I chose the word *tatum*.

beginning of the measure is (the answer to "where is one?"). There are two versions of the algorithm; one is primarily for percussive music, the other is slightly more general.

Version I of the algorithm requires a repetitive *reference instrument* (such as a bell, a clave, or a bass). The reference instrument is used to extract tempo and must repeatedly play a known pattern. The pattern period must be an integer multiple of the measure duration. Percussive music normally contains such an instrument, so this is not an unreasonable requirement. The algorithm produces the expressive timing of a *performance instrument* relative to the reference instrument. In an ensemble, any instrument other than the reference instrument may be considered a performance instrument.

The algorithm first computes a tempo function using the reference instrument. The tempo function is then transformed into a tatum duration function – tatum duration as a function of tatum number. The tatum function determines a normalized metric grid; i.e. a time grid spaced so that grid markers determine the time points of each tatum. The metric grid is then used to judge the performance instrument. For each performance instrument attack, the deviation is its distance to the nearest grid marker.

Let L be the number of tatums per measure, R be the number of reference instrument attacks per measure, $x[n]$ be the n^{th} reference instrument attack time, $y[n]$ be the n^{th} performance instrument attack time, and let $z[n] = x[n \times R]$ be our estimate of the starting time of the n^{th} measure (if reference instrument attacks do not fall on the measure starting points, we interpolate, add entries to $x[n]$, and pretend that it does). $y[0]$ must lie past the first measure's starting point.

For $n = 0 \ldots R-1$, we compute

$$P[n] = \frac{1}{M} \sum_{m=0}^{M-2} \frac{x[mR+n+1] - x[mR+n]}{z[m+1] - z[m]},$$

where M is the number of measures in the performance. $P[n]$ is the average measure fraction of the time duration between reference instrument attacks n and $n+1$. If the performer is playing very uniformly (i.e., nearly quantized), and we have the score for the pattern, $P[n]$ may be obtained directly from the score rather than from the attack times.

Next, a rough tempo function

$$T'[n] = \frac{x[n+1] - x[n]}{P[n \bmod R]},$$

is computed. $T'[n]$ provides an estimate, at time $x[n]$, of the measure duration. At any one point in time, the reference instrument informs the ensemble what the tempo is. The performance instrument, depending on whether it is dominant in the ensemble (such as a lead drum), controls when the tempo speeds up and slows down. That is, we say the reference instrument *defines* the tempo, and the performance instrument *controls* the tempo. Therefore, when obtaining the timing of a dominant performance instrument, we look slightly ahead, and compute

$$T[n] = \frac{1}{C+1} \sum_{k=n}^{n+C} T'[k],$$

where C is a parameter determining how far into the future we should look. C depends on the performance instrument, and could equal zero; accordingly, $T[i]$ may or may not be an anticipatory measure duration estimate.

Creating a continuous time function, we next linearly interpolate[4]

$$D(t) = D[n] + (D[n+1] - D[n]) \times \frac{t - x[n]}{x[n+1] - x[n]},$$

where

$$n = \{n : x[n] \leq t < x[n+1]\}.$$

It follows that $D(t)$ is an estimate, at time t, of the measure duration.

Clearly, $D(t)$ increases as tempo decreases, and $1/D(t)$ decreases as the tempo decreases. We want the tempo-normalized time points. Therefore, for each measure, we find the time points that divide the area under $1/D(t)$ into L equal area regions. The time points provide a tatum time function, a function that gives the time point for each tatum.

So, for each measure m, $0 \leq m < M-1$, and each tatum i in measure m, $1 \leq i < L$, we find $b_L[mL+i]$ where

$$z[m] \leq b_L[mL+i] < z[m+1],$$

and

$$\frac{\int_{z[m]}^{b_L[mL+i]} 1/D(t)dt}{\int_{z[m]}^{z[m+1]} 1/D(t)dt} = i/L.$$

The array $b_L[n]$ is the L-tatum per measure time location of tatum n.

We next compute the first order forward difference by linear convolution

$$d'[n] = b_L[n] * (\delta[n+1] - \delta[n]),$$

where $\delta[n]$ is the unit sample sequence[5].

[4] Higher order interpolation schemes could be used, but they probably would not significantly alter the final outcome.

[5] $\delta[0] = 1, \delta[n \neq 0] = 0$

To filter out high frequency variation, we convolve again, and compute

$$d[n] = d'[n] * h[n],$$

where $h[n]$ is either an FIR low-pass filter with a desired stop-band, or a Savitzky-Golay smoothing filter [Press *et al.*, 1992]. This step removes high frequency variation in $d'[n]$. Thus, the array $d[n]$ is our estimate of the duration of the n^{th} tatum. We next recover the tatum positions from $d[n]$ by convolving with the unit step sequence $u[n]$ [6]

$$b[n] = (d[n-1] + b_L[0]\delta[n]) * u[n].$$

The array $b[n]$ then gives us the time position of tatum n.

For each performance instrument attack $y[n]$, we find the closest tatum. The distance from $y[n]$ to the closest tatum is the attack's deviation. That is

$$\text{devs}[n] = y[n] - b[j]$$

and

$$\text{quants}[n] = j,$$

where

$$j = \underset{j}{\text{argmin}} \mid y[n] - b[j] \mid,$$

The array devs$[n]$ is the deviation function and quants$[n]$ is an array of tatum numbers. Therefore, the quantized score is given (in tatums) by quants$[n]$, the tempo variation by $b[n]$, and the deviations by devs$[n]$. A positive deviation means the attack occurred after the tatum, and a negative one means it occurred before.

Version II of this algorithm does not require the reference instrument to play repetitively and does allow the reference and performance instrument to be the same. It does, however, require the complete score or quantized representation. If the score is known, however, and the goal is to obtain the tempo variation and deviations, it may be used directly.

Here is the main difference from version I. $P[n]$ is not computed from the performance. Rather, $P[n]$ is obtained directly from the score. That is, $P[n]$ becomes the measure fraction of the time duration between reference instrument attacks n and $n+1$. The measure fraction is computed using the score. So an eight note in $\frac{4}{4}$ time would produce a value of $1/8$, and a quarter note in $\frac{7}{8}$ time a value of $2/7$. The starting time of each measure, $z[n]$, is computed from $x[n]$ according to the score. If, for a particular measure, no $x[n]$ falls at the beginning, we interpolate, and create an estimation of the measure starting time. The only other differences are the following:

$$T'[n] = \frac{x[n+1] - x[n]}{P[n]},$$

[6] $u[n < 0] = 0, u[n \geq 0] = 1$

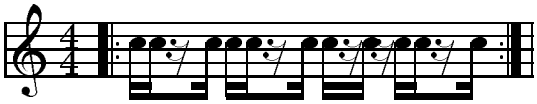

Figure 2: Guagua Pattern

The tempo variation $b[n]$ and the deviations devs$[n]$ are computed as in version I. In this case, however, the reference instrument $x[n]$ and the performance instrument $y[n]$ need not be different. Version II essentially computes the tempo variation as the low-pass filtered performance variation, and uses the high frequency performance variation as the deviations. The trick is to find the desired stop-band frequency, something which largely depends on the musical style.

4 Results

Version I of the timing extraction algorithm was applied on a performance given by Los Muñequitos de Matanzas, the extraordinary drum ensemble from Matanzas, Cuba. Timing data was obtained from the following performance instruments: the quinto (high), the segundo (middle), and the tumbao (low) drum. A new attack detection algorithm, using only high frequency energy to determine the attack, is defined in [Bilmes, 1993] and was used here. The reference instrument was the guagua, a thick bamboo cylinder, about 4" diameter, hit with two sticks. An approximation to the reference instrument pattern may be seen in Figure 2. What follows are the results of the segundo only.

The algorithm was run with $C = 3$ and $h[n] = u[n] - u[n-5]$. Therefore, $h[n]$ is just a rectangular moving window average. $b[n]$ is plotted in Figure 3. Although it looks as if there is considerable high frequency, the abscissa scale informs us otherwise. Figure 4 shows the DFT magnitude of $b[n]$. The abscissa is in normalized frequency units, where 0.5 corresponds to the Nyquist frequency. However, the frequency units are in cycles per tatum, not in Hz. The DFT magnitude is not plotted for $f > 0.1$ (10 tatums per cycle) since there is no significant energy. Also, notice the peaks at 0.0620 and 0.0630, corresponding to 16.13 and 15.9 tatums per cycle respectively. It is probably more than coincidental that 16 is both the number of tatums per measure and a large component in tempo variation.

Figure 5 shows a plot of the deviations for this performance. In this form it is hard to see any structure. Although the deviation array is essentially an unevenly sampled signal, spectral analysis is still possible. The Lomb normalized peri-

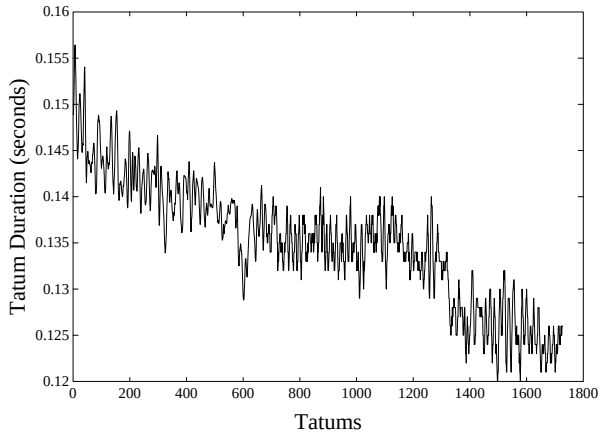

Figure 3: Muñequitos tempo track

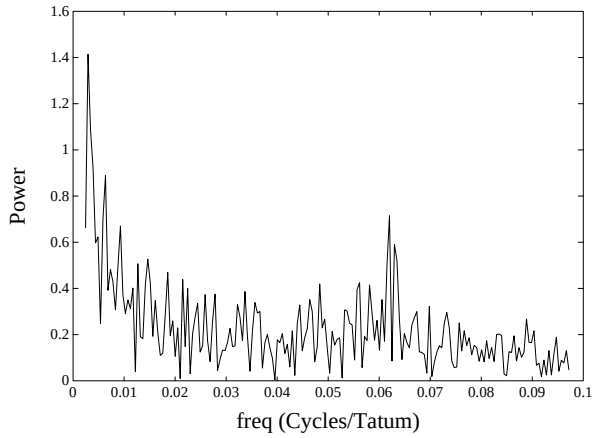

Figure 4: DFT magnitude of the tempo, $b[n]$

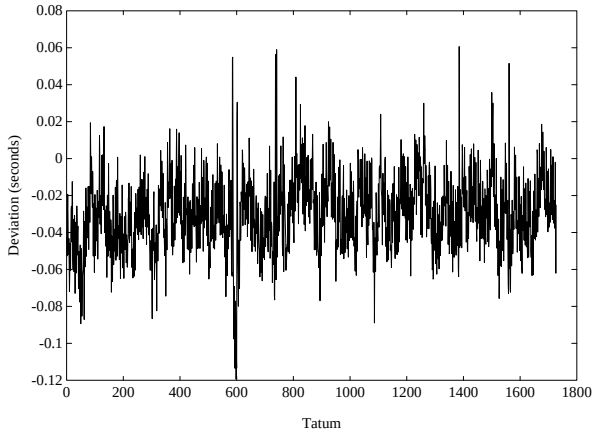

Figure 5: Segundo deviations

odogram [Press *et al.*, 1992] is a magnitude spectral analysis technique specifically designed for unevenly sampled signals. It is commonly applied to astrophysical data where regular sampling is not possible. Hoping to uncover some structure, I developed a short-time version of this algorithm, and applied it to the deviations.

Figure 6 shows the short-time Lomb normalized periodogram for the segundo deviations in Figure 5. The front axis is, again, in normalized frequency units (cycles per tatum) where 0.5 is the Nyquist frequency. The window size is 32 tatums, and the overlap is 24 tatums. Notice the strong peak at 0.25 cycles per tatum, implying considerable deviation periodicity near 4 tatums per cycle. The segundo performance, in fact, largely consists of a repeating 4 tatum phrase. For larger window sizes (order 100), this peak significantly narrows centered right on 0.25, and other small peaks appear at 0.125, 0.166, 0.333. Clearly, this confirms that structure does exist in the deviations.

The performance was resynthesized[7] by triggering select samples of the original performance. I developed an automatic note classification algorithm to obtain the drum stroke types which completed the score [Bilmes, 1993]. The various resynthesis examples follow:

1. Direct – by triggering the samples at the appropriate time.
2. Quantized – using a constant tempo equal to the overall average.
3. Quantized – using $b[n]$ as the tempo
4. Quantized – with devs$[n]$ added to the n^{th} attack.

[7] Using Csound.

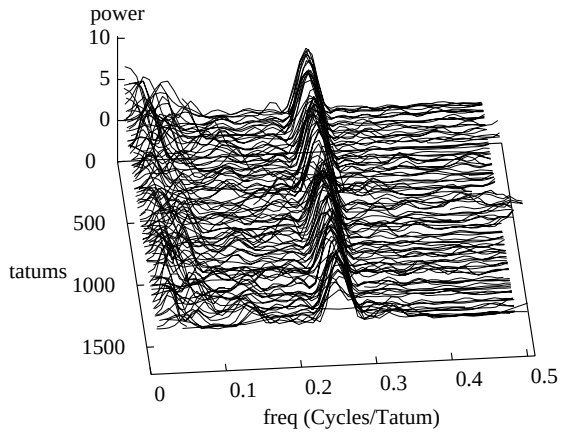

Figure 6: Lomb Normalized Periodogram: Segundo Deviations

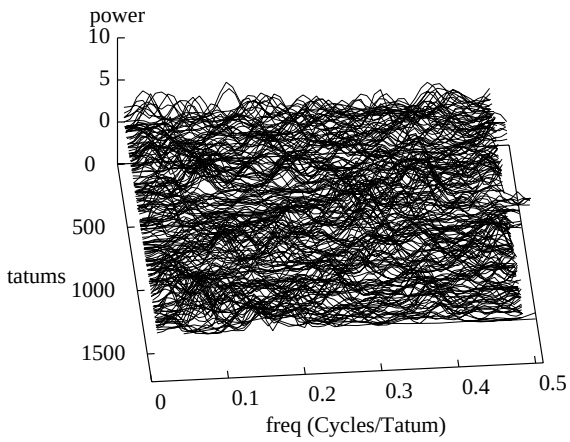

Figure 7: Lomb Normalized Periodogram: Random Gaussian Deviations

5. Quantized – with random Gaussian deviations added to each attack time. The Gaussian process had the same mean and variance as the devs[n] array.

6. Quantized – with per-tatum random Gaussian deviations added to each attack time. Here, there were 16 independent Gaussian processes, each with a different mean and variance. The mean and variance for the i^{th} process was the same as the mean and variance of devs[n mod i].

Most people who listen to these examples say that number 4 sounds most like the original, observing that only 4 contains the "feel" of the original performance. In addition, numbers 5 and 6 are considered, in general, to sound "sloppy" and "random." Accordingly, Figure 7, showing a periodogram for the deviations in resynthesis 5, confirms that there is lack of structure. As expected, resynthesis 2 sounds mechanical. Unexpectedly, even resynthesis 3 sounds mechanical. In general, without the correct deviations, the performance sounds colorless and cold – with them it sounds rich and alive.

Consequently, I propose that, in addition to the ongoing studies of tempo variation, we begin a concentrated study on performance deviations. Combining both tempo variation and deviations could eventually produce the full effect of rhythmic expressivity.

5 Experimental Tatum Deviation Program

Drum machines and music sequencers should start providing advanced facilities for experimenting with deviations. While waiting for this to occur, we[8] have developed a deviation experimentation program called `xited` (pronounced "excited" for eXperimental Interactive Tatum-Editor of Deviations, see Figure 8). Currently, `xited` runs on SGI IRIS Indigo workstations.

The program consists of a control panel, and any number of pattern windows. The main window controls global tempo in units of normal-tatums per minute, starting and stopping, and other miscellany.

The pattern windows determine the score. Each pattern window consists of a rectangular grid of toggle buttons (of any size), an additional row of sliders, and a duration value. A pattern window's grid represents a repeatedly played percussive phrase. The rows correspond to drum samples or voices and the columns correspond to pattern-tatums. If a toggle is set for row i and column j, then voice i will be triggered during pattern-tatum j. Each column also has a corresponding deviation slider. The slider for pattern-tatum j determines, in percentage of pattern-tatum, the amount of time to shift all voices set to play on that pattern-tatum.

A pattern window also contains a duration in units of normal-tatums. Therefore, different patterns may have different numbers of pattern-

[8] Jeff Foley, an undergraduate working with me at MIT, has been the primary implementor of this program

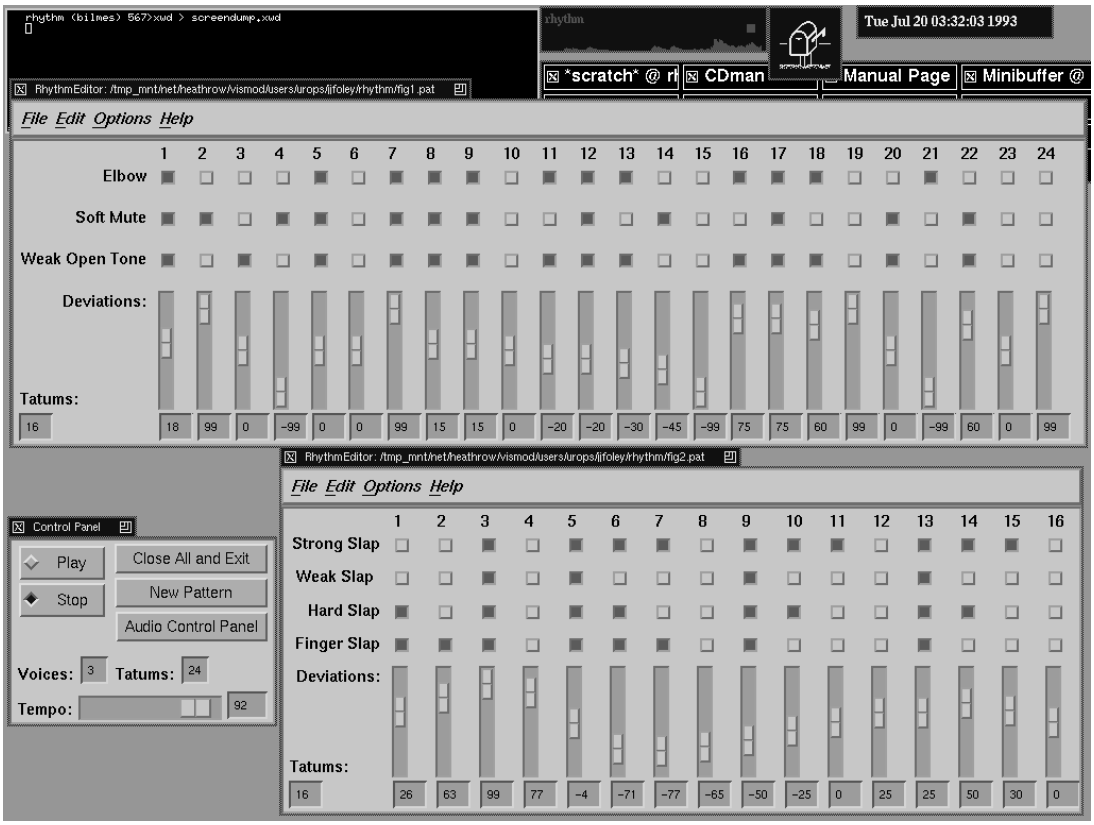

Figure 8: Graphical Deviation Program `xited` in Action

tatums, with their absolute durations the same. This may be used to express poly-rhythms and multi-tatum ethnic music. For example, in Figure 8, the top pattern has a duration of 16 normal-tatums and contains 24 pattern-tatums. The bottom pattern has a duration of 16 normal-tatums and contains 16 pattern-tatums. This example encodes, in a sense, the feeling of multiple concurrent tatums that is heard in African or Afro-Cuban music.

Each pattern window maintains a counter. When the `<PLAY>` button is pressed, the counters cycle through their patterns modulo their pattern-tatum length. When the counter reaches a particular tatum, any voices scheduled for that tatum are appropriately shifted and triggered. During playback, deviations, toggles, and pattern durations may all be adjusted.

`xited` is thus a novel drum machine user interface. A similar such interface could be used by music sequencers, or eventually, by commercial drum machines. In [Bilmes, 1993], an algorithm is defined that creates a mapping between quantized musical patterns and sets of deviations. This algorithm will be eventually incorporated into `xited`. `xited` provides the ability to experiment with deviations and to determine the best sounding deviations for a drum pattern. Indeed, some very interesting rhythmic effects may be attained with `xited` by varying deviations and pattern durations.

6 Conclusion

This paper is a summary of two and one-half chapters from [Bilmes, 1993]. Therein may also be found a new drum attack detection algorithm, an automatic drum stroke classifier, and the design of a deviation learning algorithm[9].

In this paper, I have introduced the concept of tatum, have utilized the separate elements defined in [Bilmes, 1992] for rhythmic analysis, and have demonstrated the importance of deviations for representing expressivity in percussive musical phrases. Deviations play a vital role in rhythm. They should be analyzed, comprehended, and utilized. And before switching on that quantize op-

[9] This work, including audio examples, the timing extraction algorithm, the deviation experimentation program, the drum attack detection and classification algorithm, and the learning algorithm, is (or will soon be) available via anonymous ftp on amt.mit.edu:pub/bilmes-thesis, cecelia.media.mit.edu:pub/bilmes-thesis, and ftp.icsi.berkeley.edu:pub/bilmes-thesis

tion, we should remember that yes, to err is human, but to forgive divine.

References

[Bilmes, 1992] Jeff A. Bilmes. *A Model for Musical Rhythm.* Proceedings of the ICMC, San Jose CA, 1992.

[Bilmes, 1993] Jeff A. Bilmes. *Timing is of the Essence: Perceptual and Computational Techniques for Representing, Learning, and Reproducing Expressive Timing in Percussive Rhythm.* Masters Thesis 1993, Massachusetts Institute of Technology, MIT Media Laboratory, Cambridge MA, 02139.

[Desain and Honing, 1992] Peter Desain and Henkjan Honing. *Music, Mind and Machine: Studies in Computer Music, Music Cognition, and Artificial Intelligence.* Thesis Publishers, Amsterdam 1992.

[Schloss, 1985] W. Andrew Schloss. *On the Automatic Transcription of Percussive Music – From Acoustic Signal to High-Level Analysis.* Ph.D. Thesis, CCRMA, Stanford University, Stanford CA, 94305.

[Clines, 1977] Manfred Clines. *Sentics, The Touch of Emotion.* Doubleday Anchor, New York 1987.

[Press *et al.*, 1992] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C, Second Edition.* Cambridge University Press, 1992.

[Marr, 1982] David Marr. *Vision.* W.H. Freeman and Company, San Francisco 1982

[Jaffe, 1985] David Jaffe. *Ensemble Timing in Computer Music.* Computer Music Journal, **9**(4): pp.38-48, 1985.

[Anderson and Kuivila, 1991] David P. Anderson and Ron Kuivila. *Formula: A Programming Language for Expressive Computer Music.* IEEE Computer, **24**(7): pp.12-21, 1991.

[Anderson and Bilmes, 1991] David P. Anderson and Jeff Bilmes. *Concurrent Real-Time Music in C++.* USENIX C++ Conference Proceedings, Washington, D.C. April 1991.

[Anderson and Bilmes, 1992] David P. Anderson and Jeff Bilmes. *MOOD: A Concurrent C++-Based Music Language.* Proceedings of the ICMC, pp.440-441 San Jose CA. 1992.

[Repp, 1990] Bruno H. Repp. *Patterns of Expressive Timing in Performances of a Beethoven Minuet by Nineteen Famous Pianists.* J. Acoust. Soc. Am. **88**, pp. 622-641.

[Wessel *et al.*, 1991] David Wessel, David Bristow, and Zack Settel. *Control of Phrasing and Articulation in Synthesis.* Proceedings of the ICMC, Urbana, Illinois 1987.