Paper given at the Eighth International Research Conference on Fibonacci Numbers and Their Applications, RIT, June 22-26, 1998.
Paper appeared in *Applications of Fibonacci Numbers, Volume 8*, pp. 61-73, 1999, published by Kluwer Academic Publishers.

Composing with Sequences: ...but is it art?

John A. Biles
Information Technology Department
Rochester Institute of Technology

Abstract

Algorithmic composition is not a new idea. Noteworthy composers from Mozart to Cage have experimented with algorithmic techniques, and the current musical landscape is dotted with composers and performers who use algorithms to generate a wide variety of musical parameters in compositional and real-time interactive situations. This paper focuses on the use of sequences, specifically Fibonacci sequences, as compositional devices, and tries to draw a distinction between simply auralizing a sequence and composing a piece of music based on a sequence. At the heart of this distinction are notions of form and structure, and the role of the composer. To help illustrate these compositional issues, we describe the development of a modest two-part invention, called PGA-1, which is based on a Fibonacci sequence and was realized using software developed by the author. We discuss compositional alternatives and explain the decisions made in composing PGA-1.

Introduction

Algorithmic composition dates to around 1026, when Guido d'Arezzo used vowels in the text of a choral piece to determine the pitches in the melody. In the 1400's Guillaume Dufay experimented with "formal processes" and even composed a piece using the golden mean. Mozart, whose parlor game *Musikalisches Würfelspiele* allows players to compose minuets with the aid of a pair of dice, is a slightly more recent example of a composer who was at least intrigued by the idea of algorithmic composition. In the twentieth century, serial composers have openly employed algorithmic techniques, and many composers, for example Cage and Xenakis, have composed aleatoric or "chance" music using random number generators [1]. Part of the appeal of algorithmic techniques is the possibility that the composer can focus on the essence of a piece, its form and deep structure, while the algorithm takes care of surface details like the actual notes. Clearly not all composers have such a carefree attitude about choosing "the actual notes," but the use of algorithms as a composer's assistant is gaining acceptance in many musical circles.

Algorithmic techniques can be categorized in several dimensions. One important dimension is stochastic vs. deterministic. Mozart's dice music is an example of a stochastic technique; a random process controls some aspect of the selection of melodic material, and the repeated invocation of the algorithm will yield different results. Deterministic techniques, on the other hand, yield the same sequence of events on repeated invocations, assuming the parameters controlling the process are the same. The use of Fibonacci sequences falls into the deterministic category [3].

Another dimension of algorithmic techniques is how comprehensively the algorithm is used. Some composers use algorithmic techniques only for generating low-level details, like pitches and/or lengths of specific notes, within tight constraints that they set. These composers determine the deep structure of the piece, make most of the larger-scale decisions, and use the algorithm only to generate the surface structure. Other composers may use algorithms that generate sequences that exhibit their own deep structure, which can allow the algorithm a more comprehensive role in creating the composition. The composer still makes plenty of compositional decisions, but these decisions are more collaborative in nature and often serve to emphasize the deep structure inherent in the sequence produced by the algorithm.

Fibonacci sequences can be used comprehensively due to the important property of self-similarity. Many composers in the last 20 years have pointed to the self-similarity of fractals as a justification for using fractal generators to compose music [4]. The contention is that the scale invariance of embedded structures commonly found in fractal sequences provides an "automatic" deep structure, which can serve to provide the underlying form for a composition based on such a sequence. This certainly is an appealing notion because music is widely recognized to be hierarchical in nature, and the recursive nature of scale-invariant structures certainly provides a hierarchy, at least in theory.

Unfortunately, theory doesn't always make it into practice. The problem with much fractal music is that the self-similar deep structure of the sequence often does not translate readily to a musically interesting deep structure. The structures that are so obvious when looking at a plot of a fractal sequence often go unnoticed when the sequence is heard. This is likely due to the temporal nature of music. An image can be viewed all at once, in its entirety. Viewers of an image can shift their focus from "the big picture" to the tiniest of details whenever and as often as they choose. A piece of music, on the other hand, must be listened to from beginning to end, at the appropriate tempo, without lingering over one portion or ignoring another. Certainly the human perceptual system allows the listener to focus attention on different levels from specific sounds to the overall "texture" of the piece; however, we simply cannot hear a piece of music "all at once" the way we can view an image, and we cannot skip around to arbitrary sections of a piece during its performance.

This means that the deep structure of music must unfold linearly over time, which puts pressure on the composer to make the form of a piece apparent. Furthermore, the composer must present the form of the piece while walking the fine line between expectation and surprise. Listeners tend to form mental models of a piece as they hear it. This model sets up harmonic, rhythmic and melodic expectations of what will happen in the piece. Tapping one's foot to the beat of a tune manifests a common rhythmic expectation. Hearing a V chord resolve to a I chord meets a harmonic expectation. Hearing the melody end on the tonic note meets a melodic expectation.

Surprise happens when expectations are not met. Shifting rhythms, dissonant chords, and angular melodies are examples of surprise. The composer must meet enough expectations to make the piece accessible to the audience, while providing enough surprises to make it interesting. A piece with too few surprises is usually termed "boring," while one with too many surprises is usually greeted by comments like, "That's not music--it's noise!". Clearly different listeners bring different musical experiences, attitudes, and knowledge to a given listening experience, and they can form very different mental models while listening to a given piece. In an effort to help listeners form expectations that can be met, the author sometimes provide the audience with a graphical representation of the underlying sequence for a piece. This image serves a function similar to that of the score that a veteran symphony patron often brings to a concert.

The role of the composer, then, is multifaceted, even when composing with Fibonacci sequences. The composer must convey enough of the deep structure of the sequence, meet enough expectations, spring enough surprises, and in general engage enough listeners to make the piece a success. To explore the algorithmic composer's role, we will describe the development of a modest piece, called PGA-1, which is based on a Fibonacci sequence. The development will proceed from a definition of the sequence, its visualization as a graph, and a description of the properties that make it an appealing choice as a compositional device, to the choices made by the composer in evolving the piece from a simple auralization of the sequence to a (hopefully) successful composition.

The conference presentation for this paper includes sound samples that obviously cannot be included in the medium of a written paper. However, those sound samples will be made available on the Internet from the author's home page, http://www.it.rit.edu/~jab/.

The Sequence

The primary sequence used for PGA-1 was suggested by Peter G. Anderson, hence the rather unimaginative use of his initials in the piece's title. The sequence has been referred to as the Fibonacci partition function, v_j, which counts the number of ways a non-negative integer j can be represented as a sum of distinct Fibonacci numbers [2]. The sequence is defined as the coefficients:

$$v_0, v_1, \ldots, v_{F_{N+2}-2}$$

such that

$$\sum_{j=0}^{F_{N+2}-2} v_j x^j = \prod_{k=2}^{N} (1 + x^{F_k})$$

where F_n is a Fibonacci number and

$$F_{N+2} - 2 = F_2 + F_3 + \ldots + F_N$$

is an easily derived identity. A plot of this sequence over its first 1291 members appears in Figure 1.

This sequence has several interesting properties that make it appealing as a compositional device. First, it is self-similar. As can be seen in Figure 1, there is a structure, bounded by 1's, that is repeated and elaborated as the sequence progresses. Incidentally, the portion of the sequence used for PGA-1 ends at the midpoint of the 14th elaboration of this structure.

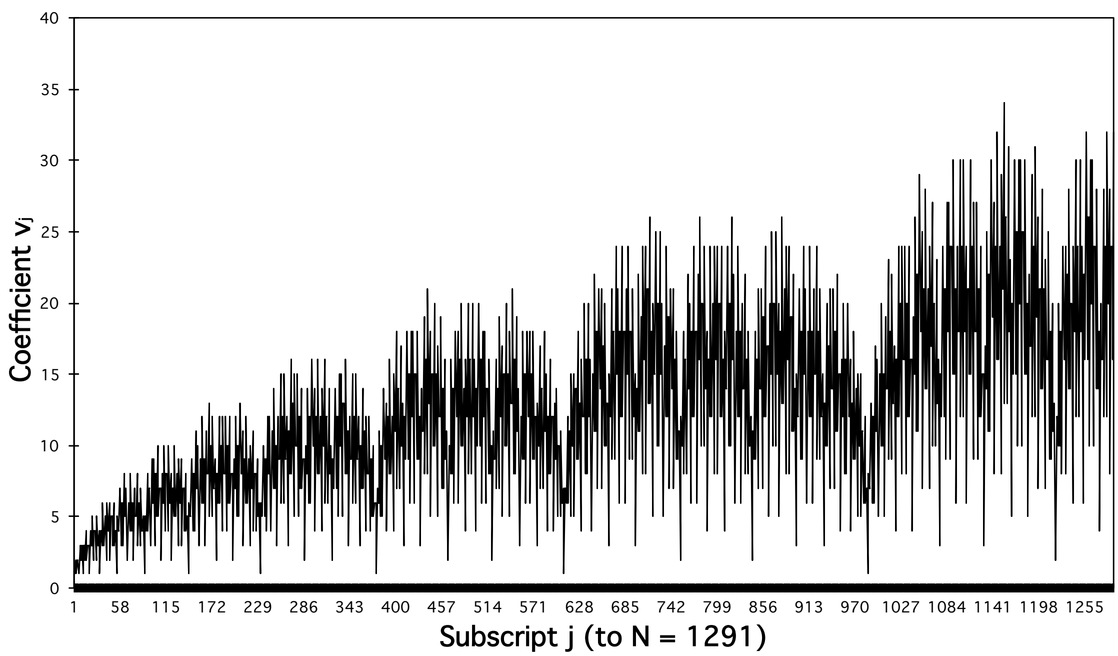

Figure 1. Visualization of Primary Sequence Used for PGA-1

Another interesting property of the sequence is that the repeated structure is symmetric about its midpoint, a fact that is distorted somewhat by the inadequate resolution of Figure 1. Symmetric structures result in a compositional technique called retrograde motion, where a melodic motif is developed by playing it in reverse. The elaborated structure in our sequence divides itself into three sub-structures, the middle one being symmetric (a palindrome) and the right one being the retrograde of the left one. This can be seen clearly in the last complete elaboration, which happens to be the 13th elaboration and runs from element 609 to 986.

The lengths of successive elaborations of the repeated structure are also interesting. The beginning of the sequence, with bars alternating above and below the first six elaborations, which include the first 33 elements, is:

1 1 2 1 2 2 1 3 2 2 3 1 3 3 2 4 2 3 3 1 4 3 3 5 2 4 4 2 5 3 3 4 1

The endpoints of the elaborations are shared 1's. If each elaboration counts only one of its two endpoints, the lengths of the successive elaborations form the Fibonacci sequence. Within each elaboration, the lengths of the substructures are similarly related. The three largest sub-structures in each elaboration, which also share endpoints (1-2, 2-2, 2-1), have Fibonacci lengths if only one endpoint is counted. Furthermore, each substructure can be broken down successively in the same way to the limit of resolution (substructure lengths of 2).

Finally, one can see each elaboration in every successor elaboration, only at a larger scale. For example, the sixth elaboration above is clearly visible as the "low points" of the 13th elaboration (the last complete one) as seen in Figure 1, with the 5's and the outer 4's doubled. Each elaboration, then, retains all the previous elaborations and adds detail with higher values. If the numbers are mapped to pitches, this means that the new details will be heard as higher notes and the established structures as lower notes. Clearly, there is a great deal of structure in this sequence to exploit.

A variant of this sequence was also used in developing PGA-1. This secondary sequence is defined identically to the primary one except that F_1 is also included, which means there are two ones to work with in creating the partitions. In other words, k in the power series starts at 1 instead of 2, and j in the summation runs from 0 to the sum of F_1 to F_N, not F_2 to F_N. A plot of this sequence appears in Figure 2. The secondary sequence shares the major structural properties of the primary sequence, but its elaborations do not descend all the way back to 1 at each endpoint, and its details ascend to higher numbers (note that the vertical scale is 0-60 for Figure 2 versus 0-35 for Figure 1). The secondary sequence was used in PGA-1 to provide counterpoint (a counter-melody) midway through the piece, and to provide velocity information for the primary sequence throughout the piece. Its use will be detailed in the discussion of "The Piece" below.

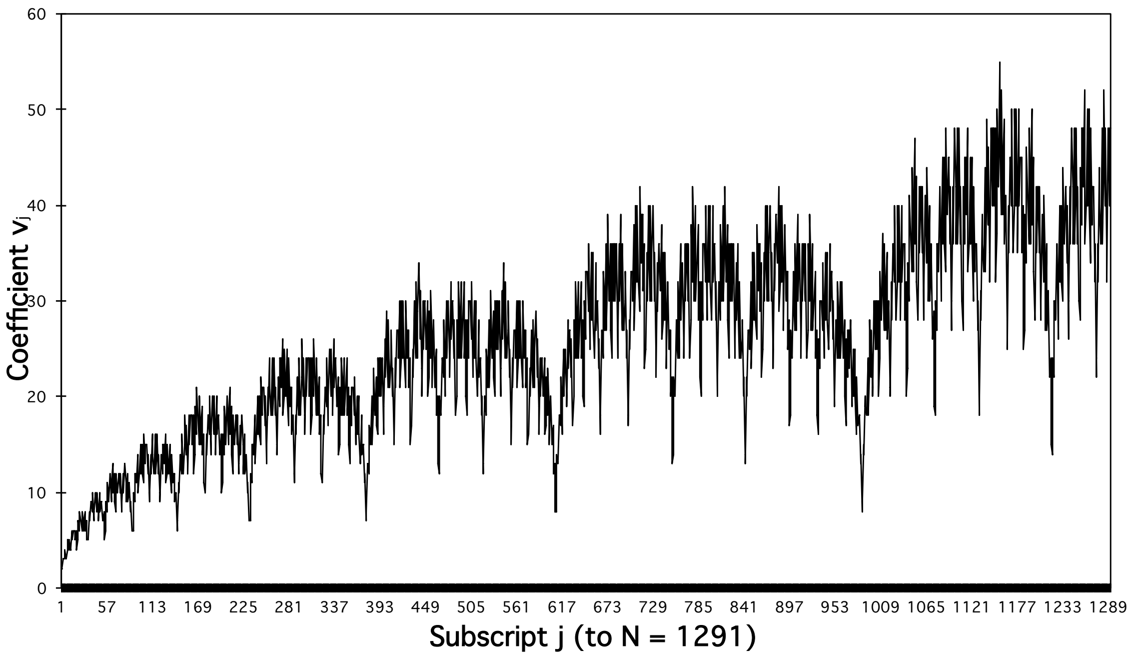

Figure 2. Visualization of Secondary Sequence used for PGA-1

One goal of the algorithmic composer is to make mathematical self-similarity musically meaningful. It is seldom musically satisfying to simply auralize a sequence, but that is a good place to start exploring the compositional potential of a sequence. Such explorations require tools for hearing a sequence. The primary tool used for PGA-1 was a program written by the author, originally for “playing” sequences of stock quotations. Hence, the program is called “Dow” by the author and has been used to generate a series of experiments he shamelessly calls “stock arrangements.” Since the Dow program has not been documented elsewhere, the next section describes the its capabilities and limitations to set the stage for explaining its use in realizing PGA-1.

The Tool

The Dow program reads one or more sequences of numbers from data files and plays each sequence on a separate channel of a MIDI synthesizer. The MIDI (Musical Instrument Digital Interface) standard has emerged as an accessible format for storing and playing back musical information. Note level information is represented by note-on and note-off events. A note-on event contains a pitch (note on a standard keyboard) and a velocity (how hard the key is struck). A note-off event is analogous to releasing a key. The duration of a note is the latency between a note-on event for a given key and the succeeding note-off event. This latency is controlled by the computer that is driving the synthesizer.

The Dow program reads two data elements for each note-level event, a pitch and a velocity. These values are stored in arrays, and the maximum and minimum pitch and velocity are noted. These maxima and minima are used to scale the pitch and velocity data to ranges set by the composer. The scaled pitch data points are mapped to actual notes by using them as an indices into a table of 49 pitches comprising a few octaves of a musical scale selected by the composer. For example, the table for a chromatic scale, which uses all 12 tones in each of four octaves, stores a different pitch in each location in the table. Major and minor scales, which use only six or seven notes per octave, store the same pitch in consecutive table locations so that the resulting scale still covers the same four-octave range as the chromatic scale.

If consecutive pitch data points happen to map to the same actual pitch, they will be treated as one note that is held through the duration of those events, which provides some degree of rhythm. The chromatic scale, then, will sound rhythmically "busier" than a major or minor scale because there are more pitches to choose from and there is less chance that consecutive data elements will map to the same pitch.

A piece can consist of up to 16 tracks, each of which is driven by a separate sequence, allowing for counterpoint when multiple sequences are used. Each sequence is played on a separate MIDI channel, and each channel can have its own settings for a host of parameters like tempo, timbre (instrument), loudness, pan (stereo location), key signature, register (octave), and musical scale. The musical scales are particularly important because they control the piece's tonality, or lack thereof. Some of the available scales are traditional ones, like major, minor, whole tone, and chromatic. Other scales are what the author calls progressive, meaning that the key signature for the scale progresses as the scale ascends into higher octaves. We have developed three progressive scales, major, minor and pentatonic, to use with long-term sequences that "start low and end high," and they provide a subtle progression (probably too subtle) among related keys over the course of the piece.

For example, the progressive major scale is based on a hexatonic major scale (six notes per octave, skipping the fourth scale degree). Each half octave of the scale is a sequence of three notes, each a major second apart. The half octaves are separated by a minor third, which occurs where the missing fourth would be. This proceeds for nearly four octaves. The C progressive major scale, then, would be:

C D E   G A B   D E F#   A B C#   E F# G#   B C# D#   F# G# A#

with extra space separating the seven half octaves. Notice that any consecutive pair of half octaves forms a hexatonic major scale, and that 11 of the set of 12 possible pitch classes appear in the entire scale, although in any given octave, only 6 pitches appear. This scale is very consonant, partly because dissonant intervals (like the minor second, tri-tone, augmented fifth, and flatted ninth) do not occur. This means that random notes played simultaneously will sound "nice" to most ears.

An important capability of the Dow program is the ability to (re)set any parameters at arbitrary times in the piece. This allows the composer to synchronize changes in tempo, tonality, instruments, etc., with structural elements of the sequence, and it provides a rich set of tools with which the composer can cross the bridge from auralization to composition. The remainder of the paper will discuss the compositional implications of various parameters and describe the choices made for PGA-1.

The Piece

The following is a portion of the description of PGA-1 written for the program of a computer music recital at which the piece was played:

> PGA-1 begins with a very slow, sparse statement of the simplest elaboration of its form in the low register, emanating from a distance at the far left. The form repeats several times, each time somewhat longer, faster and more elaborate, as it approaches the listener. When the sound reaches front and center, the form is stated in a major tonality. It then splits into two counter-melodies, which separate gradually to the left and right as the tempo continues to accelerate and the tonality evolves through minor and pentatonic scales. This continues until the resulting textures become atonal and somewhat tense. Finally, the tension is released by returning to a major tonality for the final elaboration of the form.

This musical overview is intended for an audience about to hear the piece. For a compositional overview of the piece, we refer to Table 1, which summarizes some of the parameters manipulated and serves as a score of sorts. We will discuss the compositional role of each parameter in the table and explain the settings used in PGA-1.

Elab	Leng	Tempo	Time	Scale	Pan	Reverb	Instruments
1	1	10	3	Chromatic	0	120	HarpVox
2	2	20	3	Chromatic	0	120	HarpVox
3	3	25	3.6	Chromatic	6	110	HarpVox
4	5	30	5	Chromatic	12	100	HarpVox
5	8	40	6	Chromatic	18	90	HarpVox
6	13	50	7.8	Chromatic	24	80	HarpVox
7	21	65	9.7	Chromatic	32	70	HarpVox
8	34	80	12.8	Chromatic	40	60	HarpVox
9	55	105	15.7	Prog Major	56	40	HarpVox
10	89	115	23.2	Prog Major	40, 80	30, 30	HarpVox, Harp
11	144	130	33.2	Prog Minor	32, 96	20, 20	Harp, NylonHarp
12a	144	140	30.9	Prog Penta	8, 120	10, 10	WarmAtmos, Harp
12b	89	150	17.8	Prog Major	8, 120	10, 10	PizzacatoStrings
13a	144	160	27	Prog Minor	8, 120	10, 10	Kalimba, Harp
13b	233	170	41.1	Chromatic	8, 120	10, 10	HarpVox
14	306	180	51	Prog Major	8, 120	10, 10	HarpVox

Table 1. Parameter Map for PGA-1

Each row of Table 1 represents a section of PGA-1. The sections were defined to coincide with the major structures of the sequence as discussed above. We have identified the sections by elaboration numbers (the Elab column in Table 1). The Leng column gives the length of each elaboration as the number of elements including only one endpoint. By the 12th elaboration, the sections become lengthy enough to warrant splitting them, giving us sections 12a, 12b, 13a and 13b. The splits were made between the second and third subsections in the 12th elaboration and between the first and second subsections in the 13th elaboration. Section 14 is only the first half of that elaboration and was not split further.

The Tempo column shows the gradual acceleration from a very slow 10 beats per minute (BPM) to a moderately fast 180 BPM, where a beat is defined by the Dow program to be two events (two eighth notes). In general, tempi below about 15 BPM (30 events per minute) are so slow that the listener cannot perceive much in the way of melodic content; each note is usually perceived in isolation rather than as part of a phrase. At tempi from about 30 to maybe 300 BPM, the events tend to be perceived as melodic fragments or phrases made up of distinct notes. This is the range of tempi that we tend to associate with traditional music. Tempi beyond about 300 BPM begin to be perceived as textures, with individual notes no longer distinguishable.

The gradual acceleration in PGA-1 was designed so that the length of time for each elaboration was increased at roughly half the rate at which the number of notes for each elaboration increased. This relationship holds for roughly the first half of the sections, after which the acceleration is more gradual. As the Time column in Table 1 shows, the result is a gradual increase in the amount of time for each section up to the 12th elaboration, where the elaborations split into two sections. After numerous experiments that with different tempi, the tempo of 170 BPM was deemed aesthetically pleasing for listening to the 13th elaboration, so the tempi for the intervening sections were scaled to arrive at that point smoothly.

The choice of scale(s) is a critical decision in hearing a sequence. In the initial auralization of a sequence, we usually use the four-octave chromatic scale described above, which maps the elements of the sequence to a range of 49 different notes, the maximum level of detail available in the Dow program. One problem with using the same scale for an entire piece, though, is that there usually will be no changes in tonality. A chromatic scale is, by definition, atonal (no tonal center), and while this is the most "accurate" mapping of a sequence to pitches, many listeners regard the result as less accessible because harmonic and melodic expectations are difficult to form. Listening successfully to atonal music takes practice, and many listeners aren't willing to make that effort.

Nonetheless, we used the four-octave chromatic scale for the first eight elaborations in PGA-1. This insured that the individual elements of the sequence would be heard as distinct notes and that the only held notes would be repeated numbers in the sequence and not different numbers that "rounded" to the same pitch. By the ninth elaboration, the sequence becomes interesting enough to serve as a viable melody, and we make a transition to the previously described progressive major scale. This releases some of the tension built up by the slowly accelerating atonal material of the first eight elaborations. Since the progressive major scale has only 21 different pitches for the 49 table elements, the number of held notes increases as well, making the result less active rhythmically.

In the 10th elaboration, the secondary sequence joins the fray to provide counterpoint (a counter melody). The tonality stays the same in order to make the new sequence's entrance more subtle. In the sections derived from the 11th, 12th and 13th elaborations, the tonality shifts to related keys, using progressive minor and pentatonic scales, culminating in a return to the chromatic scale for the last part of the 13th elaboration. This atonal section is fairly active, which induces a degree of tension. That tension is released in the beginning of the 14th elaboration by once again returning to the progressive major scale for the remainder of the piece. The piece finally ends on a high note, so to speak, in the center of the 14th elaboration.

As a brief illustration of how the elements of the sequence map to pitches, Figure 3 shows four "measures" from elaboration 13b, specifically j = 897-928. This section was chosen because the author finds it the most compelling moment in the piece. The elements from the primary and secondary sequences are listed in groups of eight to correspond to the eight eighth notes in each measure. The bold elements form palindromes, with the italic bold elements denoting the center points of the palindromes.

Secondary	17	18	27	24	33	30	27	36
Sequence	**27**	**33**	**39**	**30**	**36**	**27**	**24**	**36**
(upper stave)	**30**	***36***	***36***	**30**	**36**	**24**	**27**	**36**
	30	**39**	**33**	**27**	39	27	30	33

Primary	3	15	**12**	**12**	**21**	**9**	**18**	**18**
Sequence	**9**	**24**	**15**	**15**	**21**	**6**	**18**	**18**
(lower stave)	**12**	***24***	**12**	**18**	**18**	**6**	**21**	**15**
	15	**24**	**9**	**18**	**18**	**9**	**21**	**12**

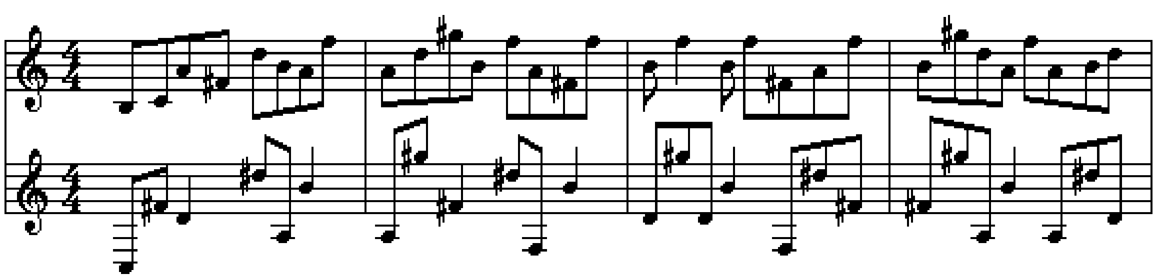

Figure 3. Elements 897-928 with secondary sequence on upper stave, primary on lower

Three other MIDI parameters, velocity (loudness), pan (left-right stereo location), and reverb (echo) are manipulated in the piece, the first algorithmically, and the other two directly by the composer. The velocity for each event was determined by using one sequence as the velocity settings for the other. In other words, the velocities for the primary sequence were derived from the secondary sequence, and the velocities for the secondary sequence were derived from the primary sequence. This served to accentuate the elaboration/section boundaries and provided a gradual increase in loudness over the course of the entire piece. In addition, since the two sequences are not totally aligned at an element-to-element level, this provides subtle changes in loudness that are not totally correlated with changes in pitch. In other words, it sounded nice!

The pan and reverb parameters were manipulated to achieve the movement described in the musical overview above. In Table 1, a pan value of 0 means completely left, 127 means completely right, and 64 is in the center. A reverb value of 0 is no reverb effect (totally dry), 127 is maximum (totally wet), and 40 is a typical default value. By starting the piece far left, very wet, and relatively quiet, the sound appears to be off in the distance. By gradually shifting to the center, reducing the reverb, and getting louder, the sound seems to move closer to the listener. When the secondary sequence makes its entrance, the two parts begin relatively close together and spread farther apart. Multiple sounds that come from the same location tend to be perceived as a single voice, even if different instruments are being played. By separating the two voices somewhat gradually, they emerge as different parts, which adds to the complexity and the tension.

The identity of voices brings us to the last major parameter manipulated in PGA-1, the timbres (instruments) used to realize the sequence. Modern MIDI synthesizers have hundreds of preset instruments, ranging from traditional acoustic instruments to classic electronic sounds, with all manner of combinations in between. For example, the tone generator used in realizing PGA-1 boasts 792 different instruments from which to choose, clearly a large and varied palette of timbres. In dealing with rapid sequences of short notes, which is what the sequences for this piece map to, it is advisable to use timbres with crisp attacks and relatively rapid releases like percussive or plucked instruments. Timbres with gradual attacks and releases lead to indistinct or muddy textures, which may be desirable in some situations but would obscure individual notes. Therefore, we decided to use instruments with crisp attacks, and after experimenting with bell-like sounds, we selected a family of plucked instruments for PGA-1.

As the primary instrument, we chose an instrument called HarpVox, which is a harp sound overlayed with a subtle vocal sustain. The instrument sounds reasonable throughout the three octave range need for the piece, and the vocal sustained layer fills the space between notes and provides a gradual release, which at fast tempi, results in a chord-like effect as new notes begin while old notes release. To add variety in the later sections, other plucked instruments were used, including pizzicato strings and a kalimba (thumb piano). In the final two sections, the piece returns to the HarpVox. The trick in selecting timbres is to find instruments that complement each other when heard concurrently, while providing transitions that offer an appropriate level of surprise. The somewhat limited variation in timbres chosen for this piece fits our desired aesthetic, which is gradual, subtle changes rather than abrupt shifts.

So there you have it--an alleged piece of music derived from Fibonacci-related sequences! Hopefully, PGA-1 transcends mere auralization and succeeds as a composition for many listeners. However, a paper is not the preferred way to experience a piece of music; it must be heard. The conference presentation of this paper included a "live performance" of PGA-1, but for readers of the proceedings, a sound file can be downloaded from the author's web site at http://igm.rit.edu/~jabics/Fibo98/PGA-1.mp3. We invite comments!

References

[1] Curtis Roads. *The Computer Music Tutorial*. MIT Press, 1996.

[2] Neville Robbins. "Fibonacci Partitions." *The Fibonacci Quarterly*, Vol. 34 (1996): pp. 306-313.

[3] Manfred Schroeder. *Fractals, Chaos, Power Laws: Minutes from an Infinite Paradise*. W. H. Freeman and Company, 1991.

[4] Claus-Dieter Schulz. *The Fractal Music Project* web site. Viewed most recently in April, 1998. URL is http://www-ks.rus.uni-stuttgart.de/people/schulz/fmusic/.

AMS Classification Number:

11B39

Indexing Keywords:

Algorithmic composition
Computer music