SBEAT3: A Tool for Multi-part Music Composition by Simulated Breeding

Tatsuo Unemi
Soka University, Hachiōji, Tokyo, 192-8577 Japan

Abstract

This paper presents a design of support system for musical composition based on Simulated Breeding. In our system named SBEAT3, each individual in the population is a short musical section of sixteen beats including 23 parts, thirteen solos, two chords and eight percussions.The melody and rhythm are generated by a type of recursive algorithm from genetic information. By selecting favorite piece among scores displayed on the screen, the user listens to the sounds and decides which should be theparents to reproduce the offspring in the next generation. The genetic codes of children are generated through mutation and crossover. Iterating this process, the user obtains better pieces gradually. Embedding some domain specific functions, such as changing tempo and selecting tones, we can build a useful tool to make it easier for a beginner to compose his/her favorite musical pieces.

Introduction

This paper presents an application of *Simulated Breeding* (Unemi 2002b), a type of *Interactive Evolutionary Computing* (IEC) (Takagi 2001),that realize breeding the artifacts on the computer. The ordinary type of evolutionary computing consists of iteration of selection, crossover and mutation. Selection is done based on a fitness function that evaluate each individual. This framework works well if the human designer can drawan appropriate procedure to compute fitness values.However, we often suffer difficulty to figure it out explicitly bysome reasons, such as multi-objectivity, subjective criteria,dynamic environment, and so on. IEC is a promising technique to find better solutions in the domains for optimization by user's subjective criteria, of which root can be found in *Blind Watchmaker* by R. Dawkins (Dawkins 1986). Differently from ordinary methods of evolutionary computing,fitness values are not calculated automaticallyby the predefined evaluation function but are given by the userfor each individual in some manner.

Some researchers are calling this method as*Interactive Genetic Algorithm* (IGA) (Smith 1991),because it can be seen as a modified version of*Genetic Algorithm* (GA) (Goldberg 1989).Usually, the application systems have a methodfor the user to rate each individual, typically a graphical user interfaceusing a slider or a set of radio buttons.In the method named *Simulated Breeding*, the user directly picks uphis/her favorite individuals as parents for the next generation.This means the fitness values can take only one (selected) or zero (not selected).It disables stochastic selection, but has an advantage to reducethe number of user's operations to assign the fitness values.

Our project named SBEAT is to try to build a support tool for composing music. We started from a prototype of small system SBEAT1 (Unemi & Nakada 2001), of which individual includes sixteen beats of three parts, guitar, bass and drums. We extended it to eight parts by adding four more solos and percussion part (Unemi & Senda 2001). The current newest version SBEAT3 has ability to use full of sixteen channels of General MIDI (Mid 1995).

Some researches on a music application of IEC have already been tried in alternative methods. G. L. Nelson (1993) proposed some methods to generate music by emergent techniques, and built a system named Sonomorph to breed short musical scores in a similar style of Blind Watchmaker. J. A. Biles *et al* (1994) also proposed an alternative framework to breed improvisational phrases of Jazz music. His system named GenJam generates solo melody from each individual in the population in turn. The user pushes 'g' key for good and 'b' for bad to select appropriate individual. N. Tokui (2000) and M. Unehara (2001) are also independently challenging from their alternative points of view for rhythms and longer tune with a type of domain knowledge.

Our system resembles an extended version of Sonomorph (Nelson 1995), but has richer target domain and an improved graphical user interface toward more practical use as described later.

The following sections show our design of phenotype and genotype, describe the procedure to produce phenotype from genotype, present the graphical user interface to make the system more useful and practical, and then state concluding remarks.

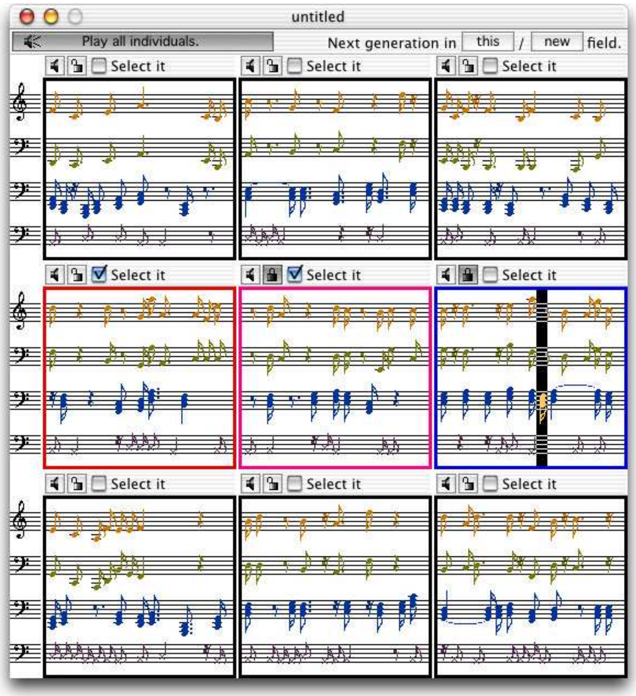


Figure 1: A field window of SBEAT3.

Phenotype

An individual, the unit of breeding target, is a barof sixteen beats including 23 parts, thirteen solos, two chords and eight percussions. Figure 1 shows an example of field window of SBEAT3, quite same with SBEAT2. We named it *field* inspiring from a field of experimental farm for breeding agricultural products. Each of field window corresponds to a population and includes nine sub windows each of which corresponds to an individual in the population.

The score of each solos is a sequence of single notes. The score of a chord part is a sequence of combination of three notes. The score of a percussion part is a sequence of timbre. The timbre for each solos and chord parts are chosen by the user from more than 128 types of instruments programmed in MIDI.

Genotype

Figure 2 shows the structure of genotype, that consists of three types of chromosomes for melody, rhythm and velocity. Each chromosome is a two dimensional array of 16 by 23 elements, for beats and parts. The melody chromosome is for calculating pitches for each note by a development procedure described later. The values of each element for rhythm are interpreted in similar manner of GenJam as follows:

1. it continues the previous note or rest, if the most significant bit is one;
2. it rests, if the left most three bits are 011_2; and

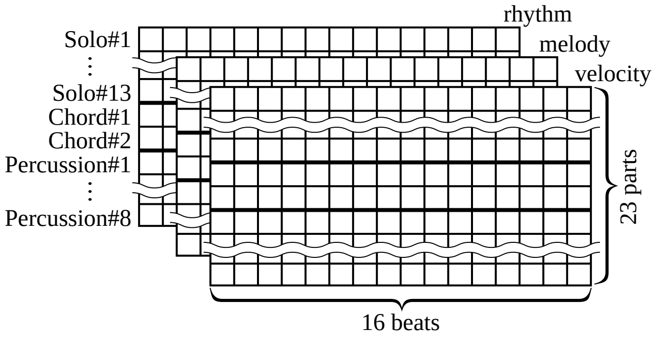


Figure 2: Structure of genotype.

3. it plays a new note, otherwise.

We employ mutation by bit flip in a constant possibility 5% for each bit in genotype. Relatively high mutation rate is useful when the population size is small. Crossover operation is done in a style of one point crossover, where it cuts a chromosome at randomly selected position of element boundary into two parts and exchanges them. Mutation is applied when the user selects only one individual as a parent to move to the next generation. Crossover is applied when more than one parents are selected.

Morphology

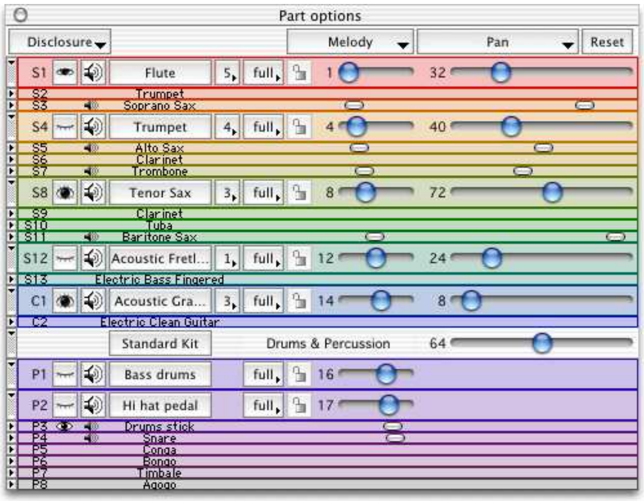
Morphology, the process to develop phenotype from genotype, plays an important role to generate feasible candidates in the initial population usually generated from random gene.

We employ the same type of recursive algorithm as SBEAT1 and 2 to guarantee some degree of similarity between parents and children in terms of human's feeling. Data in the chromosome for melody in the first chord part are used as parameter values for the algorithm. (See (Unemi & Senda 2001) for a detail.) Each integer for pitches is mapped onto a note in major or minor tonic scale in default, to generate a melody acceptable by a beginner. We call this melody line of sixteen beats *basic melody*.

In the chord parts, three notes, the note from basic melody, three steps above it, and three steps below it, are played in the same beat. The actual pitches for solo parts are calculated by adding a value from data in melody chromosome. The value can be -3, 0 or 3 steps to keep an acceptable harmony. The pitch data of the beats that the rhythm chromosome indicates rest or continuation are ignored.

User interface

Design of user interface is one of important features to make an IEC application successful.



S1 Flute 5 full 1 32
S2 Trumpet
S3 Soprano Sax
S4 Trumpet 4 full 4 40
S5 Alto Sax
S6 Clarinet
S7 Trombone
S8 Tenor Sax 3 full 8 72
S9 Clarinet
S10 Tuba
S11 Baritone Sax
S12 Acoustic Fretl... 1 full 12 24
S13 Electric Bass Fingered
C1 Acoustic Gra... 3 full 14 8
C2 Electric Clean Guitar
Standard Kit Drums & Percussion 64
P1 Bass drums full 16
P2 Hi hat pedal full 17
P3 Drums stick
P4 Snare
P5 Conga
P6 Bongo
P7 Timbale
P8 Agogo

Figure 3: Part option dialog of SBEAT3.

Breeding in the fields

Unavoidable issue in any auditory domains is that it takes some length of duration for the user to certify the phenotype by listening to the sound of individual. It is easy to compare the features of twenty individuals in the domain of graphics art, such as (Unemi 2002a). However, in the case of music it is impossible to test more than one individuals at same time.

In our project, we restrict the individual to a short phrase, and make the population size simultaneously displayed on the screen to be only nine, as similarly to Sonomorph. As shown in Figure 1 before, each of field window has a button to play all of individuals in the population sequentially, and each sub window has a button to play only the individual repeatedly. Visualization in the popular style of music score is helpful to catch up the characteristic of phenotype before listing.

Multi-field user interface

To improve the diversity of genotype, we employ a multi-field user interface (Unemi 1998). One of the issues in evolutionary computing is how it can avoid premature convergence to find a feasible solution. As know in breeding of living organisms, migration between populations separately evolved for several generations is often useful to obtain a new improved offspring. The fitness landscape of evolution of complex systems is usually multi modal, that is, there are a number of locally optimal peaks. A small size of population is easily trapped there because it has less diversity enough to find another peak.

By the reason described before, the population size of SBEAT is only nine. The operation of migration provides effective help for the user to cope with this drawback, by *copy & paste* and *drag & drop* operations for individuals between any different field windows.

Part options

Figure 3 shows dialog windows for setting parameters for each part by the user. As described before, the user can assign any instrument in General MIDI for solo and chord parts. For eight percussion parts, we prepared fourteen sets of timbre, bass drums, tom tom, snare, hi hat, ride cymbals, crash cymbals, conga, bongo, timbale, agogo, whistle, guillo, and others.

The sub window for individual includes only four parts to be displayed because of the restriction of screen space and human's ability of viewing. The user can select which parts are shown in the field. The user can also select which parts are played. We has been extended our system by increasing the number of parts, but of course it is not always need to use all of them.

The user is also allowed to set up some types of controls for each MIDI channel, such as pan, volume, reverb, chorus, and celeste. The range of pitches is also able to changed for each part by octaves.

Correspondence between parts in genotype and score can also be alternated. The parts sharing the same part in genotype of rhythm are played synchronously. The parts sharing the same part in genotype of both melody and rhythm are played in unison.

Player options

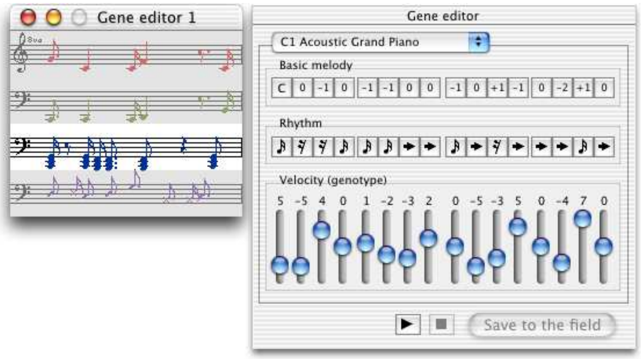
Another dialog window called *player option* is to set up the parameters applied to whole of parts. The user can indicate whether an individual is played at once or repeatedly, can set the tempo in the range from 20 to 180 quarter notes per minute, and can change the scale and key note. The key note is determined from the gene in default, but the user fix it by operating this dialog. In addition to the scales of eight notes in an octave, the user can select twelve pitches of half a step for one octave to add a flavor of avant-garde.

Protection of gene

Music has complex structure, but it is well organized, that is, any piece of ordinary music can be divided into smaller pieces and can be described by combination of common features such as rhythm, melody, expression, timbre and so on. By this reason, we sometimes want to fix some parts or some features of bred pieces and to try alternative candidates that varies the rest of parts or features. To realize it, the partoption dialog shown before includes a menu to choose the type of chromosome and of toggle buttons to indicate protection of each part. The protected parts of genotype are modified neither through mutation, crossover, nor reseting.

Integrating bars into longer score

To support music composition, it is necessary to be able to collect bred bars into longer sequence as the result tune. A score window is to integrate bars copied from



Basic melody: C 0 -1 0 -1 -1 0 0 -1 0 +1 -1 0 -2 +1 0

Velocity (genotype): 5 -5 4 0 1 -2 -3 2 0 -5 -3 5 0 -4 7 0

Figure 4: Genome editor of SBEAT3.

any field window. As the same manner of migration, the user can copy any individual in a field window into any bar in a score window by *copy & paste* and *drag & drop* operation. The user can edit the sequence of bars by *cut, copy & paste* and *drag & drop* operation in a score window, and can modify the options for each bar using the part option dialog and the player option dialog.

Direct editing of genotype

Breeding is redundant when the user knows how the genotype should be modified to obtain his/her favorite phenotype. Figure 4 shows windows called genome editor that allows the user to directly edit genotype of an individual. It is possible to input a known melody using the genome editor. The current implementation does not allow to represent any type of melody in genotype, but this type of *reverse* morphology is useful to help user's task to make a variation of known tune.

Conclusion

A design of tool for music composition using Simulated Breeding was described above. We have improved it in the aspect of the number of parts from three in SBEAT1, eight in SBEAT2, and then 23 in SBEAT3. These tools are useful for a beginner to compose his/her favorite and original music. And also it can be helpful for the user, who already has the knowledge of music, to get inspiration for a new melody.

Of course, there remains many works we should do in future, such as;

1. combination with breeding tools for another features, timbre, effects, chord progression, and so on,
2. more facilities to collaborate with other music tools,
3. facilities to use this tool as real time music performance, and so on.

SBEAT2 and 3 run on MacOS 9 and X. The executable binaries and some sample tunes in SMF and MP3 format can be downloaded from:

http://www.intlab.soka.ac.jp/˜unemi/sbeat/ .

We hope many persons enjoy them.

Acknowledgment

The author thanks Eiichi Nakada and Manabu Senda who worked in the SBEAT project as students.

References

Biles, J. A. 1994. Genjam: A genetic algorithm for generating jazz solos. In *ICMC '94. International Computer Music Conference*, 131–137.

Dawkins, R. 1986. *Blind Watchmaker.* Essex, UK: Longman.

Goldberg, D. E. 1989. *Genetic algorithms in search, optimization and machine learning.* Addison-Wesley.

Midi Manufactures Association, La Habra, CA. 1995. *The Complete MIDI 1.0 Detailed Specification.*

Nelson, G. L. 1993. Sonomorphs: An application of genetic algorithms to growth and development of musical organisms. In *Proceedings of the Fourth Biennial Art & Technology Symposium*, 155–169.

Nelson, G. L. 1995. Further adventures of the sonomorphs. In *Proceedings of the Fifth Biennial Art & Technology Symposium*, 51–64.

Smith, J. R. 1991. Designing biomorphs with an interactive genetic algorithm. In *Proceedings of the Fourth International Conference on Genetic Algorithms*, 535–538.

Takagi, H. 2001. Interactive evolutionary computation: Fusion of the capacities of ec optimization and human evaluation. *Proceedings of the IEEE* 89(9):1275–1296.

Tokui, N., and Iba, H. 2000. Music composition with interactive evolutionary computation. In *GA2000. Proceedings of the thirrd International Conference on Generative Art.*

Unehara, M., and Onisawa, T. 2001. Composition of music using human evaluation. In *Proceedings of 2001 IEEE International Conference on Fuzzy Systems.*

Unemi, T., and Nakada, E. 2001. A tool for composing short music pieces by means of breeding. In *SMC '2001. Proceedings of the IEEE Conference on Systems, Man and Cybernetics*, 3458–3463.

Unemi, T., and Senda, M. 2001. A new musical tool for composition and play based on simulated breeding. In Dorin, A., ed., *Proceedings of Second Iteration*, 100–109.

Unemi, T. 1998. A design of multi-field user interface for simulated breeding. In *Proceedings of the third Asian Fuzzy Systems Symposium*, 489–494.

Unemi, T. 2002a. Sbart 2.4: An IEC tool for creating two-dimensional images, movies and collages. *Leonardo* 35(2):53, 71–73.

Unemi, T. 2002b. Simulated breeding – a framework of breeding artifacts on the computer. *Kybernetes* 31:(to appear).