AUTOMATIC FITNESS IN GENERATIVE JAZZ SOLO IMPROVISATION

Kjell Bäckman
Informatics Department
University West
Trollhättan, Sweden

ABSTRACT

Recent work by the author has revealed the need for an automatic fitness function in the evolutionary algorithm for generation of jazz solos. A computer based automatic fitness function enables greater populations and a great number of generations within a manageable time. This paper describes how the automatic fitness function is implemented using score calculation based on specific events recognized in the solo line The evaluation function is based on the author's solo analysis of 73 great masters' solo characteristics and their techniques used for build-up of jazz solos.

1 INTRODUCTION

The most important feature of a good jazz musician is to be able to keep an entire solo together as an entity, i.e. to build up the solo phrase by phrase in collaboration with the other musicians, where each phrase is a natural continuation of the previous phrase and leads further to a climax of intensity. After the climax the solo should be rounded off. A longer solo might contain several climaxes, but they should then be organized in a musically meaningful way.

This paper builds upon the experiences of the Generative Jazz Improvisation experiment of making the computer build up a solo by means of evolutionary algorithms with the genetic representation consisting of the rubber band and operator tree principles (Bäckman 2008). That project utilized a manual implementation of the fitness function, i.e. the user was obliged to listen through all sound examples in one generation and manually specify the two best to act as parents in the next generation.

This project uses a computer based function for analysing melodies based on the author's analysis of 73 prominent jazz musicians (Bäckman 2009 and 2011), giving them scores and selecting the melodies with highest score for breeding of the next generation.

The overall main purpose of this project is not to create a jazz improvising computer of its own, but merely to open your mind to new thinking, free you from old habitual paces of playing and enrich your improvisation style with new kinds of musical material.

Dahlstedt (Dahlstedt 2004) uses an automatic fitness function in his experiments in automatic composition. Since improvisation is equal to composition in real time, Dahlstedt's work has been a source of inspiration.

2 Automatic Fitness

The evolutionary algorithm process is described in (Bäckman 2008).

By using an automatic evaluation process, it is possible to take advantage of the evolution process by using huge populations and a great number of generations.

The problem of building an automatic fitness function is obvious. How can we with computer code reflect things as sensibility, intensity, musicality, tension, expectation, climax, relaxation etc., i.e. concepts used by musicians and musicologists to describe a good improvisation solo. The problem must be broken down to the level of analysis of musical components like intervals, rhythm, motif repetition, melodic curves, melodic heights etc., i.e. components more or less prevalent in the MIDI information of a melody. The analysis process must be capable of evaluating the melody components and assigning score points at different positions of the melody.

This evaluation process is based on the tools and methods jazz musicians use to build up a solo and increase the intensity during the solo, both as regards experiences from listening to and analysing world famous jazz musicians (Bäckman 2009). That analysis categorizes the techniques jazz musicians use to build up the intensity into tools and records which artists are using the tool bo build up the intensity.

The intensity is self-conserving to some extent. With this we mean that an intensity increase at a certain moment in a solo will not immediately get lost, but instead be kept to some extent for subsequent moments, however with diminishing strength.

Thus, an intensity increase will stay for some time in the listener's ear and contribute to the build up towards the climax.

3 The Evaluation Process

The evaluation is carried out by a number of analysis functions, which contribute with points that are stored per note of the melody. When all analysis functions have contributed with their points per note, the points are aggregated per bar. See (Bäckman 2009 and 2011) for a full description of the background to the score settings.

3.1 Fitness Selection and Breeding

We start with an initial population of 100 individuals. Each individual is evaluated and given a score. The individuals with the highest score, i.e. which most perfectly align to the optimal intensity figure, will have the best chances to be selected as parents for breeding (Bäckman 2008). The selection of two parents for a single child is stochastically made, based on their evaluation score.

The following sections describe the analysis functions, which contribute with the detailed scores. These functions correspond to the categorization of the techniques resulting from the solo analysis of 73 great masters (Bäckman 2009 and 2011). Furthermore, the score given by each function corresponds to the level of utilization. So if a technique, like repetition, say, is used by many musicians, it will give a high score when encountered in the solo.

3.2 Density

Density refers to metric aspects, i.e. the number of notes per time unit. However, a rapid sequence of short notes could in some situations be an efficient means of increasing the intensity, especially as a sudden cascade of short notes in an environment of long notes or rests. Therefore, density has been considered a candidate to intensity increase.

The density function goes through each sequence of 5 notes and checks whether the average note length is shorter than 48 MIDI ticks, which in this application corresponds to an eighth note. If so, a contribution of the difference between the average and 48, divided by 4, is given.

3.3 Rhythm

The Rhythm function tests if in a series of 5 notes there is one long and several short notes, and gives contribution accordingly. A note is considered long if the number of MIDI ticks is greater than 96, i.e. longer than a quarter note. A note is considered short if the number of ticks is equal to or below 32, i.e. shorter than an eighth triplet. If the number of short notes equals 3 a contribution of 10 points is given. If the number of short notes equals 4 a contribution of 15 points is given.

3.4 Motif Repetition

The Motif Repetition function analyses the melody and tries to find a small motif that occurs also in the immediate future, maybe somewhat transposed or somewhat modified as concerns the intervals between the pitches. The two motives to be compared must have about the same corresponding intervals. The interval deviation can be maximum 2 minor seconds, and the allowed deviation is increased by 1 minor second per each third note.

If a motif repetition is discovered a contribution of 20 points is given.

3.5 Sequence

If a motif is repeated directly after the first occurrence of the motif, transposed up or down, a contribution of 6 points is given. The transposition also accepts slight pitch modifications by a major second.

3.6 Chromatics

The Chromatics function discovers any chromatic features built into the melody by checking each note, each 2nd note, each 3rd and each 4th note. If there are three equidistant chromatic notes, a contribution is given. The contribution number of points is 12 times the distance number of notes (length of the chromatic motives). For instance, if a chromatic series of notes at the distance of 3 notes is discovered, the contribution will be 12x3 = 36 points.

3.7 Heights

The Heights checks for heights in consecutive phrases and whether the heights are chromatic. If the heights are upwards chromatic, a contribution of 8 points is given, and if downwards, 5 points. The reason for this is that a sequence of upward chromatic motives provides a stronger pushing feeling than a downward movement.

3.8 Rests

Rests have the effect of creating excitation. The rest must be of some considerable length to not just function as a link between two phrases, but to work as an efficient musical component. If there is a rest longer than one bar, each extra quarter prolongation of the rest will give 5 extra points.

3.9 Polyphony

Some musicians playing instruments capable of producing polyphonic effects (piano, guitar, vibraphone etc.) use the polyphonic effect to enrich the musical output of their solos. If this occurs, a contribution of 10 points is given.

3.10 Counterpoint

If a leap bigger than a fifth is made, preceded by at least 5 notes ornamenting around the source note, a contribution of 8 points is given.

3.11 Ornamentation

If, in a series of eighth notes, the intervals do not deviate from the central note by more than a major second, a contribution of 4 points is given.

3.12 Cascade

If there are 16 notes in sequence shorter than or equal to a sixteenth note, it is classified as a cascade and gives a contribution of 4 points.

3.13 Mix of High and Low Registers

If, in a series of 16 notes, there is a difference between the highest and lowest note of more than an octave, a contribution of 4 points will be given.

3.14 Stubborn Twisting

If all pitches in a series of 16 notes are played in a narrow interval of up to a fourth, a contribution of 4 points will be given.

3.15 Rhythm Effects

If eighths are played with rhythmic hang including swing factor with a time delay of up to 12 ticks related to the beat, or if even eighths are used in a swing context (non-zero swing factor), a contribution of 10 points are given.

3.16 Motives from Other Tunes

We have decided to incorporate a function for detection of standard motives by comparing the solo to a database with standard motives. The database can easily be updated with additional motives.

The function tries to match the motives of the solo to the motives in the database. It accepts pitch deviations of up to a major second, and it also accepts any rhythm deviations, because a slightly modified motif would be regarded interesting by the listener. If a match is found, a contribution of 6 points is given.

3.17 Double Tempo

If, in a series of five notes, at least four sixteenths are found, a contribution of 5 points is given.

3.18 Rubato

If, in a series of five notes, at least three notes longer than a quarter note are found, a contribution of 8 points is given.

3.19 Hard Swing

If, in a series of 4 eighth notes, a rhythmic swing adjustment is found, by slightly prolonging the first and shortening the next, or if the blues scale is recognized in a sequence of 4 notes, a contribution of 6 points is given.

3.20 Polyrhythm

If a motif is repeated (pitch and length are the same) but the total length of the motif is not a multiple of 96 (quarter note) or equals n*96, where n is odd >=3, or if stressed notes (MIDI volume>110) are played in a context of lower volumes < 80, and the attacks are equidistant but not a multiple of 96 (quarter note), a contribution of 7 points is given.

4 Experimental Run

We have experimented with different number of iterations. At each run, the score value seems to converge after a few thousand iterations, and very little improvement is achieved after about 3 thousand iterations. In a run with 10,000 iterations, we found the score progression given in fig. 4.

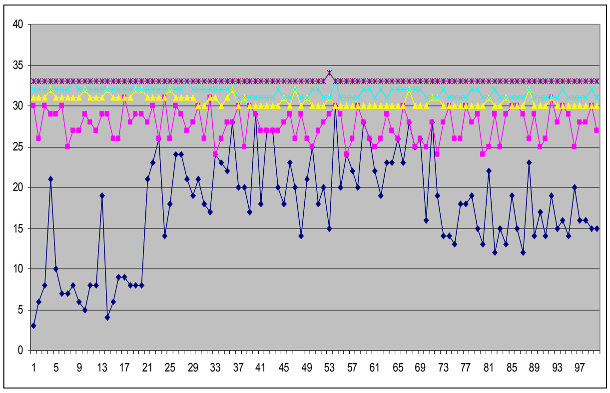

Figure 4. Score evaluation in a run with 10,000 iterations, with initial population (blue line), after 100 iterations (red), 500 (yellow), 1000 (green) and 10 000 iterations (black).

5 Results

Comparing the manual evaluation (Bäckman 2008) to the automatic evaluation, the manual method has obvious advantages by being able to select melodies of personal preference. However, by being able to use a huge population and a great number of generations, and thus taking full advantage of the evolution process, the drawback of artificial automated score calculation is by far overcome.

As mentioned in the introduction, the overall main purpose of this project is not to create a jazz improvising computer of its own, but merely to open your mind to new thinking, free you from old habitual paces of playing and enrich your improvisation style with new kinds of musical material. The automatic fitness process can be considered fulfilling this aim.

The sounding output of the generated music examples has an obvious intensity curve indicating some kind of solo grand layout often used by professional jazz musicians. The music examples definitely have an interesting feature of intensity build-up and relaxation.

The method of allowing several intensity maxima and minima and aligning the intensity fluctuations to the intermediary gradients provides interesting results, since it allows different types of solo build-up by virtue of the parameter control capability.

We have experimented with different contribution points for the different functions, and acquired a point setting corresponding to the results of the solo analysis of 73 great masters (Bäckman 2011). The score setting is the one described for the various functions.

A set of unaccompanied sound examples can be heard at this link in General MIDI format:

http://oden.ei.hv.se/kjell/autofitness .

A link to an example with computer generated drums and bass accompaniment is also available:

http://oden.ei.hv.se/kjell/autofitness/midi_evolv1.mid

An example of an orchestrated tune with virtual instruments and my own acoustic piano playing is available at this link:

http://oden.ei.hv.se/kjell/autofitness/gate.mp3

A similar tune has been recorded by our live jazz group:

http://oden.ei.hv.se/kjell/trio/Random2.mp3

And here is another version with the complete EJI Group:

http://oden.ei.hv.se/kjell/CD/RandomLicks.mp3

6 Future Work

Evolutionary algorithms can also be used to produce harmonies, drum rhythms, walking bass figures, piano and guitar accompaniment chord arrangement, and basic tune themes. This work has already been initiated with promising results, and will be published in the future.

Communication between musicians is very important in live jazz music. It would be possible to implement this according to memetics logic by letting the soloist, the drummer and the accompanying pianist “listen” to each other and reuse motives and rhythmic accents (memes). Hopefully this will render a feeling of collective improvisation, where no particular soloist is leading the others but instead a situation where all musicians have the same value and contribute to the musical result on an equality basis.

7 References

1. Dahlstedt, Palle. 2004. “Sounds Unheard of – Evolutionary algorithms as creative tools for the contemporary composer”. PhD diss, Chalmers University of Technology, Gothenburg.
2. Bäckman, Kjell. 2008. “A Generative Representation for the Evolution of Jazz Solos”. In *Proceedings of EvoWorkshop Conference 2008, Napoli*.
3. Thywissen, Kurt. 1996. “GeNotator: An environment for investigating the application of generic algorithms in computer assisted composition”. In Proceedings of International Computer Music Conference 1996 (ICMC96), pp. 274-277, Hong Kong.
4. Manning, Peter. 2004. *Electronic and Computer Music*. Oxford University Press, New York, USA.
5. Dean, Thomas. 2003. Hyperimprovisation: Computer-Interactive Sound Improvisation. A-R Editions Inc.,Middleton, Wisconsin.
6. Rowe, Robert. 1993. *Interactive Music Systems*. The MIT Press, Cambridge, Massachusetts, USA.
7. Rowe, Robert. 2001. *Machine Musicianship*. The MIT Press, Cambridge, Massachusetts, USA.
8. Bäckman, Kjell. 2009. “Analysis of Jazz Solo Build-up Techniques”. http://oden.ei.hv.se/kjell/eji/analysis.pdf
9. Bäckman, Kjell. 2011. “Evolutionary Jazz Improvisation”. http://oden.ei.hv.se/kjell/eji/EJI_General.pdf
10. Bäckman, Kjell. 2010. “Ant Colony Optimization and Evolutionary Algorithms Applied to Jazz Solo Improvisation”. In *Proceedings of EIMAS Conference 2010, Juiz de For a, Brazil* http://oden.ei.hv.se/kjell/eji/ACO_EIMAS_2010.pdf.