International Journal of Literature and Arts
2014; 2(6): 258-265
Published online January 05, 2015 (http://www.sciencepublishinggroup.com/j/ijla)
doi: 10.11648/j.ijla.20140206.15
ISSN: 2331-0553 (Print); ISSN: 2331-057X (Online)

Evolutionary jazz improvisation - jazz harmony and solo improvisation created by means of evolutionary algorithms with automatic fitness

Kjell Bäckman

University West, Trollhättan, Sweden

Email address:
kjell.backman@hv.se

To cite this article:
Kjell Bäckman. Evolutionary Jazz Improvisation - Jazz Harmony and Solo Improvisation Created by Means of Evolutionary Algorithms with Automatic Fitness. *International Journal of Literature and Arts.* Vol. 2, No. 6, 2014, pp. 258-265. doi: 10.11648/j.ijla.20140206.15

Abstract: This paper describes the Evolutionary Jazz Improvisation (EJI) system for creation of jazz harmony and jazz solo improvisations based on evolutionary algorithms with automatic evaluation. The evaluation function is implemented using score calculation based on specific events recognized in the chord progression and solo line, and alignment to different optimal intensity increase and relaxation curves. 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. The results have been evaluated by our live jazz group and used in jam sessions, some of which have been recorded and made available for listening at the links given later.

Keywords: Jazz, Algorithm, Automatic Evaluation, Automatic Fitness, Improvisation

1. Aims and Context

1.1. The EJI Harmony System

Jazz harmony has during the jazz history mainly been functionally based on principles of tonality derived from classical and romantic periods of the 18th and 19th centuries. Since the birth of jazz during the first two decades of the 20th century, the jazz harmony has been systematically organized around a tonal centre by fifth progressions, which means that each chord has been related to a base note and classified as minor or major, and optionally also enriched with colourization, such as:

Cm, Eb7, G13b9, A7#11

Blues and ragtime harmony mainly used simple major/minor triads at the distance of fifths. Swing music enriched the chords with sixths and ninths but the chord progressions were mainly the same. Bebop further enriched the chords with further colourizations such as b9, #9, #11, 13, b13 etc. and exchanged some chord progressions by inserting an extra subdominant parallel, e.g.

G7 – C was replaced by Dm7 – G7 – C

However, the focus was still on major/minor and fifth progressions. The main harmonic contribution from cool jazz and hardbop during the 50's was further advanced chord colourizations. A few new-thinking musicians, like Ornette Coleman, Cecil Taylor, Don Cherry and others, began at the end of the 50's to split up the harmonic foundation prevailing until then, and this development continued during the subsequent decades under stylistic classification into "modal jazz", "avant-garde", "free form" etc. Current jazz musicians have to some extent adopted this break-up tendency.

Experiments have also been made during the 60's and 70's by e.g. Herbie Hancock, Miles Davies and fusion musicians Brecker Brothers. Not to mention all experiments in the classical music domain during the entire 20th century from Schoenberg and onwards.

However, mostly you can in the "modern" jazz styles trace some remainders of the functional harmony principles and the fifth circle basis. When progressive or avant-garde musicians create compositions with new harmony, there still is a risk to get stuck in conventions dictated by routine and learnt behaviour, idiomatic properties of the instrument and the musician's physical and muscular restrictions. The computer has no such restrictions but creates harmonies controlled by the programmed algorithms. The aim is to be able to free oneself from traditional thinking and find new ways of thinking about jazz harmony.

By means of computer based evolutionary principles we introduce a function-less harmony system that somewhat changes the musical feeling in jazz compositions by excluding the relation to any base note and the major/minor concept.

In addition to alternate types of harmonies, the EJI harmony system also forms the basis for scale construction used by the jazz musician when performing a solo.

Dahlstedt [5], Dean [9], Levine [10], Manning [11], Rowe [13] and Thywissen [15] have made valuable contributions in the same area and have been sources of inspiration in the harmony area of the EJI system.

1.2. Jazz Solo Improvisation

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. A good improviser is not allowed to drop the focus and give way to meaningless cascades of notes or producing routine phrases for lack of artistic ideas. The challenge is to be able to plan the structure of the solo already from start, and then stick to the plan during the entire solo. There are some excellent examples in the history of jazz with this ability, such as John Coltrane, Miles Davis, Keith Jarrett, Bill Evans and Johnny Griffin, just to mention a few.

An improviser often uses standard phrases and motives trained during a long time of practicing and concerting. He relies on routines built up through repeated usage of similar muscular movements well accommodated to the physical design of the instrument.

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

The EJI project aims at making the computer build up a solo based on these principles. This is done using evolutionary principles on a genome structure consisting of a raw melody line split up into small melody fragments (delta phrases) and a hierarchical structure of operators applied to the delta phrases. Initially, the raw melody line is built up according to a “rubber band” principle, where each pitch interval is constructed using energy constraints much like the tension of a stretched rubber band. After application of the operators, the delta phrases will be somewhat different, however hopefully preserving some kind of musical idea. The aim is to ensure logical development of a consistent material during the solo and thus reflect the feature of a well-planned solo.

There are others working with similar concepts. Al Biles [1] has developed a system, GenJam, which uses phrases played by the “master” soloist as basis for the computer played solo. GenJam has three types of improvisation: whole chorus, chase improvising and collective improvising. The two last types are typical for older jazz forms like New Orleans jazz and the Swing era, while the first type is more relevant to modern jazz forms from the second half of the 20’th century. In GenJam the function of listening to motives from fellow musicians has been solved by means of an Analog-To-MIDI converter device. However, GenJam does not have the grand format principle of building up a solo with focus on intensity fluctuations, which is the long-term aim of our research.

Francois Pachet [12] has developed a system where the user plays a melodic material, from which the system builds its solo according to Markov chain probability calculation. The user can at any time introduce new melodic material to which the system responds.

Dahlstedt [5][6] uses small melody fragments and combines them using operators in a recursively generative tree structure. Thywissen [15] uses generative music grammars in his GeNotator project. Dahlstedt and Thywissen apply their theories to classical music composition. In our project we try to apply similar theories to improvised jazz music. Also Manning’s [11] exploration of MIDI technologies and Dean’s [9] work on hyperimprovisation have been valuable.

Robert Rowe has in his two volumes, Interactive Music Systems and Machine Musicianship [13] some interesting features such as scales connected to certain chord types, which have been valuable for the development of this system.

2. Methodology

2.1. Artificial Evolution

The evolutionary algorithm process starts, from a basic set of parameters (genome), by creating a first random population of pictures, melodies, chord progressions or whatever. The fitness evaluation then takes place by examining the bred material (children) and selecting the best, optionally by giving each a score. The children with the highest score have the highest probability to become parents for the next generation. The breeding is done by combining the genome of two parents, optionally by applying a mutation somewhere in the genome. The mutation might imply a shift between two parameter values, or a slight modification of a genome value.

The principle of using evolutionary algorithms to develop new artistic productions was first started on a broader scale in the digital graphics area, such as forerunner Karl Sims [14]. The evolutionary algorithms principle is well accommodated to that area because when using interactive evaluation of a created generation, as described by Dawkins [7, 8], you can swiftly scan over a great number of pictures and select the best according to your personal preference. With audio material, however, the selection procedure is much slower since you will have to listen through each bred melody in a generation, one at a time. The first experiments in the music area have been made by Collin Johnson and Palle Dahlstedt [5, 6].

The fitness selection and breeding is repeated generation by generation until you arrive at a genome good enough to be used for reproduction of artworks (pictures / melodies etc.).

This process is much the same as the genetic process of creating a new species generation in nature, only that it must be sped up considerably to have a chance to be completed in proper time, which is accomplished by the support of the computer.

2.2. Automatic Evaluation

The first version of the EJI 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 for the next generation. The problem was to keep your concentration at the same high level. After listening to all melody examples in a couple of generations you got tired and felt a difficulty to be able to search for real musical quality in each melody. When listening to the 10th melody of a specific generation it was hard to remember melody no. 1, 2, 3 etc and estimate whether the 10th was better than the 3rd, for instance. During a listening session you might also be inclined to change your focus on what you actually are searching for.

That first version revealed the need for an automatic fitness function in the evolutionary algorithm. A computer based automatic evaluation function enables taking full advantage of the strengths of the evolution process in terms of a large population and a great number of generations within a manageable time.

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, polyphony, 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 found components and assigning points at different positions of the melody. To be able to do this, a comprehensive analysis of 73 great masters from the jazz history and their improvisation styles has been made. The results have been published in another paper [not yet officially published].

2.3. Automatic Harmony Evolution

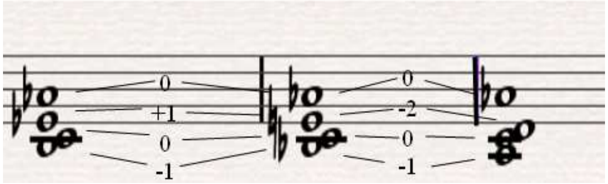

Fig 1. *Chord genome.*

For the automatic harmony evolution, a number of parameters control the overall behaviour of the genetic evolution process:

- Number of notes per chord
- Number of notes to change from chord to chord. A higher value gives abrupt chord changes, while a lower value gives a more homogenous chord sequence.
- Maximum number of half-note steps to be allowed when a voice moves from chord to chord. Also in this case, a greater tolerance gives more abrupt chord changes.

A genome consists of the absolute MIDI pitches for the initial chord. The pitches are randomly created within a specific pitch range around middle C. For each chord change the genome holds the number of half-note steps per note (fig. 1).

In this case the genome will be:

59 60 63 68 -1 0 +1 0 -1 0 -2 0 …

From the beginning, an initial population of 100 individuals (chord sequences) is created.

Each individual is then evaluated. The evaluation of a chord sequence is based on the principles that small chromatic steps from chord to chord have the strongest emotionally pushing character, and that upward intervals tend to increase the intensity at most. Therefore, such characteristics will be favoured (table 1).

Table 1. *Voice step scoring.*

Type of interval	Contribution
Upward minor second	3
Upward major second	1
Downward minor second	2

The internal structure of each chord is also evaluated, where intervals like small seconds and quarters are premiered, since they tend to avoid tonal centres, while intervals like thirds and fifths are avoided for the same reason. However, within one chord, only one small second is allowed to avoid cluster chords. The same applies to quarters. Table 2 shows the contribution figures from the internal chord analysis.

Table 2. *Internal chord interval scoring**.***

Type of interval	Contribution
Minor second	3
Major second	2
Fourth	1

The contribution total for the entire chord sequence is saved for each individual of the population. The selection of parents for the next generation is stochastically made, i.e. the individuals with the highest score will most likely be subject to parenthood for the next generation.

On breeding, the crossover is performed by combining different sections of two parents’ genomes similar to the process of combining DNA for species. This means that the figures (e.g. -1, 0, 1, 0) are taken from one of the parents from the beginning of the genome, up to the randomly selected break point, from where the remainder is taken from the other parent. Then, a mutation is made by amending a few values one step up or down, so -1 might be -2 or 0, etc.

When a child has been created in this way, it is evaluated as described above. If the child's score exceeds that of the worst individual, it will replace that individual, which is discarded. If the created child is worse than the worst individual of the population, the child will be discarded. Thus, according to the elitism principle, a created child, if kept, will always improve the quality of the entire population.

In this experiment 10000 iterations are used in each run. Maybe a larger number of iterations could result in a still better solution, but by experimentation we have found 10000 iterations enough.

That said about harmony evolution, let us turn to solo evolution.

2.4. The Improvised Solo Genome

A jazz solo in this project consists of the melodic raw material and a hierarchical structure of operators. The melodic raw material is split into small portions, delta phrases, to allow processing at a lower level. Each operator of the operator structure processes a single delta phrase. For instance, one operator can add a note to the delta phrase, another operator can transpose the delta phrase a stipulated interval, and still another operator can invert a delta phrase. After processing of the delta phrases by the operators the result will be a modified melody. The operators are applied hierarchically, i.e., one operator is applied to the whole melodic material, then one to each half, etc.

Fig. 2 shows how the raw material is split into delta phrases and then processed by the operators of the operator tree.

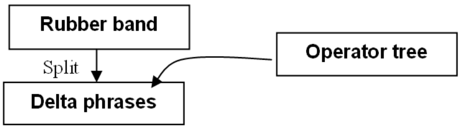

Fig 2. *Delta phrases are created by splitting the rubber band. The operator tree then processes the delta phrases.*

The creation of the raw material and the application of the operators to the delta phrases are described in the subsequent sections.

Rubber Band Principle

The creation of the raw material uses a similar technique as the one used in the mid-point displacement algorithm for landscape generation. It is created by originating from start and end pitches, then dividing the interval recursively. The middle pitch is stored for each interval division. It is represented as a deviation from the mean between the start and end pitch. Thus, the representation is a binary tree of deviation figures from the mean line between two interval end points.

The raw material is created as follows. First we calculate a random start pitch and an end pitch (fig. 3).

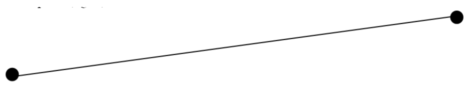

Fig 3. *The start and end points of the rubber band.*

Then we generate a random pitch in the middle of the time span, which is allowed to deviate from the mean pitch line by a maximum pitch span (fig. 4) specified by a control parameter.

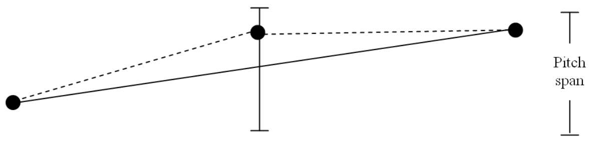

Fig 4. *The middle point is created within the allowed span.*

Then we split the interval into two equally long intervals and repeat the process of generating a new note in the middle of each time interval (fig. 5).

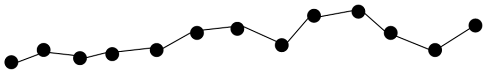

Fig 5. *The complete rubber band.*

For each recursive subdivision, the maximum allowed pitch span is reduced by a certain factor (control parameter). The deviation is represented as a factor of the maximum allowed pitch span for each individual interval.

The pitches can be accommodated to a given chord progression, described in the EJI Harmony System section above.

So far we have talked about the rubber band principle in connection with pitches. But the rubber band principle is also applied to note lengths and volumes. For instance, a start note length and an end note length are generated. The middle note length of the melody interval is selected with a deviation from the mean length. The deviation must be within the allowed length span, which also is reduced by a certain factor each time the interval is divided. This has the effect that a short series of notes will have about the same length, however by modifying the length span factor, this can be adjusted to achieve sudden burst outs of short notes.

The note lengths are not quantized to the standard rhythmical values of whole notes, half notes, quarter notes, eighths, triplets etc. The lengths can have any MIDI ticks value. The reason for this is to not being tied up to traditional musical thinking concerning rhythm, but to concentrate on melody shapes and intensity fluctuations. This will however provide a free-rhythmic feeling separated from the beat. As an option, when applying the melodies to a jam session situation, we have created a function for accommodation of the rhythms to standard values of quarter notes, eighths, sixteenths, triplets etc.

For dynamics, the rubber band principle is applied similarly; a start note volume and an end note volume are generated. The middle note volume of the melody interval is selected with a deviation from the mean volume. The deviation must be within the allowed volume span, which also is reduced by a certain factor each time the interval is divided. By modifying the volume span factor you can achieve more or less smooth volume shapes.

By combining the rubber band principles for pitch, length and volume we achieve pitch shapes, length shapes and volume shapes operating independently of each other.

The technical representation of the raw material is MIDI pitch, length and volume for the start note and end note. The contour is represented as a tree structure of relative values of pitch, duration and volume (relative to the mean of the end points of the current interval), which, when applied recursively, will recreate the exact contour.

Thus, the genome representation for the raw material rubber band melody is:

- Maximum allowed span from the beginning for pitch, length and volume- Factors for reduction of the allowed span for pitch, length and volume - Start note: pitch, length, volume
- End note: pitch, length, volume- For each interval division (2, 4, 8, 16, etc.): Middle note – deviation from the mean: pitch, length, volume, stored as a factor of the maximum allowed deviation span for the interval in question.

Delta Phrases

The raw material, after creation, is split into delta phrases. A delta phrase is a series of notes with pitch, length and volume. The number of notes per delta phrase is given by the number of notes in the rubber band divided by the number of delta phrases (global parameters). An example with n delta phrases, ∂Ph_0 - ∂Ph_{n-1} , is shown by fig. 6.

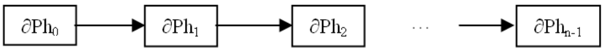

Fig 6. *Division of the rubber band into delta phrases*

Operator Tree

The purpose of organizing operators hierarchically into an operator tree is to allow each delta phrase to be processed hierarchically by a series of operators. Fig. 7 shows the structure of an operator tree.

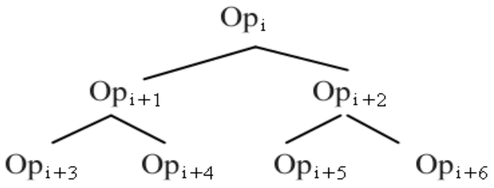

Fig 7. *Structure of an operator tree.*

The operator at the top level is applied to all delta phrases. The operators at level 2 are applied to half of the delta phrases each. The 4 operators at level 3 are applied to ¼ of the delta phrases. The division by 2 for each level is continued until there is one single operator for each delta phrase. The effect of this is that each delta phrase is processed by a series of operators from top to bottom of the operator tree, one operator per level. Two adjacent delta phrases will be processed by the same series of operators, except for the last operator, which is different. Two distant delta phrases will be processed by two different sets of operators. The effect will be a process introducing conformity over the whole solo and as the recursive process branches out, variation is introduced between sections.

Table 3. *Operator options.*

Unity, no change.
Transposition of a random single note by a random number of half notes
Transposition of the entire delta phrase by a random number of half notes
Addition of a note at a random position of the delta phrase, random within the allowed span
Removal of a randomly selected note in the middle of the delta phrase
Augmentation of each interval in the delta phrase
Diminuation of each interval in the delta phrase
Retrograde, the delta phrase is reversed
Inversion, the pitches of the delta phrase are mirrored around its average pitch
Rhythm modification, flip the length of a note around the length of a quarter note
Volume modification, flip the volume of each note around the center
Insertion of a rest of random length at the end of the delta phrase
Repetition of part of the delta phrase. A delta phrase is divided into three segments, and one of them is repeated

Each operator modifies a delta phrase in one particular way. The operator type options are given in table 3.

The composition of operators in the operator tree is randomly created based on probability percents (control parameters) per each type of operator in table 3.

To incur more operators processing to a delta phrase than accomplished by one series of operators from top to bottom of the operator tree, a delta phrase is allowed with a certain probability of recursion. If an operator is selected for recursion, all operators in the corresponding sub-tree will get the same number of recursions (fig. 8). The frequency of doing this is controlled by a parameter.

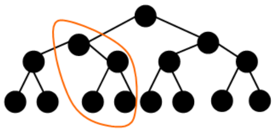

Fig 8. *Recursion of a sub-tree.*

To allow for a richer variation, each node of the operator tree can be equipped with up to 10 operators, which means that several operator options from table 1 can be performed to the corresponding set of delta phrases. If recursion is in effect for the node, all additional operator options will be performed the specified number of times.

The genome representation of an operator is:

- Operator type (table 3)
- No of steps to transpose
- Note number for transposition, rhythm adjustment, addition, removal
- New deviation factors for addition (pitch, length, volume)

- Additional operators (a list of up to 10 additional operators per node)
- Recursion (number of times to repeat the operator sub-tree)
- Insertion of rests (length)
- Repetition (first, middle or last part)

2.5. The Solo Evaluation Process

The evaluation of an improvised solo is carried out by a number of analysis functions, which contribute with a score value stored per note of the melody. When all analysis functions have contributed with their score values per note, the points are aggregated per bar. At the end of the evaluation, the aggregated points per bar reflect the intensity fluctuations of the melody.

Since the intensity provided by a melody fragment tends to stay in the listener’s ear for some time, the score values per bar will be preserved to some extent; 50% of the score value for one bar is added to the score value of the next bar. The score values per bar will have a graphic representation something like in fig. 9.

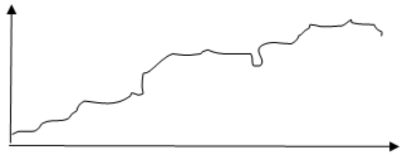

Fig 9. *Calculated points per bar during the solo, with 50% accumulation from the previous bar.*

Having calculated the score value per bar, the score value pattern is compared to an optimal curve. The idea is that the solo might contain several climaxes and several relaxation points without any predestined position. We classify a point above a specific limit as a climax (max point). The limit is specified as a percentage (control parameter) of the overall maximum point. We have experimented with 90%, but other values could be used. A relaxation point is a point below a certain limit, specified in the same way as a percentage of the overall maximum point. We have experimented with 10%. There might be several max points in sequence, and several min points in sequence. The highest of the climaxes in sequence is classified as the max point, and the lowest of the relaxations is classified as the min point. We measure the optimal gradient between each max point and the subsequent min point, and the optimal gradient between each min point and the subsequent max point. An example is shown in fig 10.

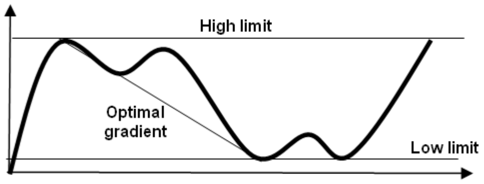

Fig 10. *Flexible optimum method.*

The system rewards great differences between max and min points. The greater differences, the higher score. The score value to be assigned is calculated as the sum of the differences between each max and min point, which then is divided by the number of max/min points with the aim of avoiding too many max/min points.

The deviations from the optimal gradient per bar are subtracted. The aim is to get as little total deviation as possible from the optimal gradient.

To summarize, the greatest differences between climax and relaxation, and the closest connection to the optimal gradients, will be the best result.

Fitness Selection and Breeding

An initial population of 100 individuals is used, and 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 [2]. The selection of two parents for a single child is stochastically made, based on their evaluation score, implying that the best parents will create the largest number of children. There is a lower score limit that must be exceeded by any parent to allow for participation in the stochastic selection. The limit is controlled by a parameter. The limit is also increased along with the evolution process, since the entire population will grow better and better. The limit is at each generation set to 90% of the best score.

After breeding by crossover, followed by some mutations on a probability basis [4], the child is evaluated and, if it has a better score than the worst one, it will replace the worst individual, which is discarded from the population.

The fitness selection and breeding is repeated a number of times, specified by a control parameter. We have experimented with 10000, but any number can be used. However, 10000 have been regarded enough, since very little score improvement has been measured after 10000 iterations. It is also possible to perform the repeated selection and breeding until the total score value has reached a limit value (a control parameter).

The analysis functions contributing with the detailed score values are described in detail in another document written by the author. They correspond to the categorization of the techniques resulting from the author’s solo analysis of 73 great masters. Furthermore, the score given by each function corresponds to the level of utilization. E.g. if a technique, like repetition, say, is used by many musicians, it will give a high score when encountered in the solo. Below is a summary of the analysis functions:

- Density
- Rhythm
- Motif repetition
- Sequence
- Chromatics
- Melodic heights
- Rests
- Polyphony
- Counterpoint

- Ornamentation
- Cascades of rapid notes
- Mix of extremely high and low notes
- Stubborn twisting
- Rhythmic effects
- Motives from other tunes
- Double tempo
- Rubato
- Hard swing
- Polyrhythmic effects

3. Results

3.1. Harmony

Chord progressions created by the evolutionary algorithm described above provide the feeling of a continuous progress towards new heights without arriving at any rest points, which is the case with traditional functional harmony, where some chords have a striving character to dissolve into tonics. Compare the chord sequence of a tune like ‘Autumn Leaves’:

Am7 D7 Gmaj7 Cmaj7
F#m7b5 H7b9 Em7 Em7

There is an intermediate rest point at the chord Cmaj7 and then a final rest point at Em7. These rest points provide a relaxation at various positions of the tune, providing a periodic feeling. Such relaxation points are not found in tunes with the new kind of harmony. Our conclusion is that this new kind of harmony has an on-going forward-striving feature not prevalent in mainstream jazz harmony.

Compared to manual evaluation, where you have to listen to each generated individual, one at a time, the automatic evaluation takes full advantage of the evolution theory by utilizing large populations and a huge number of generations. Furthermore, the evaluation criteria are kept strictly constant i.e. we do not change focus on the objectives of the evolution process. Of course there are also drawbacks with automatic evaluation. It is difficult, if at all possible, to make the computer evaluate abstract concepts such as musicality, tension, expectation, climax, relaxation etc. Anyway, with the automatic evaluation we obtain results that might not otherwise have been discovered.

When jamming with a jazz group on tunes with this new type of harmony, it has the effect on the soloist of continuously proceeding towards a climax never completely reached. The soloist is compelled to go on and on and on. The listener will be involved in this forward-striving feeling of wanting more all the time.

When I experimented with these ideas in a live jazz group, it turned out that the musicians had apparent difficulties in keeping chords and scales in their minds during their solos, since they had to learn completely new chords and scales. The harmony was of a kind that they could not apply their current knowledge and personal routine and not trust old learnt patterns of behaviour. Clever and experienced musicians appeared to be relative beginners, at least during the first rehearsals. Difficulties became obvious especially when playing tunes with an odd periodicity where a chord could last for 3 bars and the next chord for 2 ½ bar, etc. So the time required for rehearsal tended to grow remarkably. For example, the bassist, who normally bases his walking bass paths on a base note accentuated at the first beat of each bar and scale walking at the remaining beats, got into problems when there was no specific base note. Learning to play this new kind of music is a laborious task that requires a new way of thinking and a lot of practice and patience.

Furthermore, to find the most adequate way of playing, a lot of time was spent on discussion and reflection in the acoustic live jazz group. For instance, a great deal of cooperative work was used for accommodation of the bassist’s notes and the piano chord layout to each other.

3.2. Solo Improvisation

Comparing the manual solo evaluation 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, like in the harmony evolution, the drawback of artificial automated score calculation is by far overcome.

As previously mentioned, the overall main purpose of this project is not to create a jazz improvising computer program of its own, but merely to open your mind to new thinking, free oneself 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 obvious intensity fluctuations 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.

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

The same tune has been recorded by our live jazz group:

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

These three sound examples are also provided as separate files with this document (midievolv.mid, autofitness.mp3, Gate.mp3).

References

[1] Biles, J.A. (1994) GenJam: a genetic algorithm for generating jazz solos. Proceedings of the 1994 International Computer Music Conference. ICMA, San Fransisco, pp. 131-137.

[2] Anonymous (2008) A Generative Representation for the Evolution of Jazz Solos. In *Proceedings of EvoWorkshop Conference 2008, Napoli*.

[3] Anonymous (2008) Evolutionary Jazz Harmony, In *Proceedings of BIOMA Conference 2008, Ljubljana*

[4] Anonymous (2008) Automatic Fitness in Evolutionary Jazz Improvisation, In *Proceedings of ICMC Conference 2008, Belfast*.

[5] Dahlstedt, P. (2001) Sounds Unheard of – Evolutionary algorithms as creative tools for the contemporary composer. In *Proceedings of ICMC Conference 2001, La Habana, Cuba*.

[6] Dahlstedt, P. (2001) Creating and Exploring Huge Parameter Spaces: Interactive Evolution as a Tool for Sound Generation In Proceedings of ICMC Conference 2001, La Habana, Cuba.

[7] Dawkins, R. (1986) The Blind Watchmaker. New York W. W. Norton & Company, Inc. New York, USA.

[8] Dawkins, R. (1976) The Selfish Gene. Oxford University Press Inc., New York, USA.

[9] Dean, T. (2003) Hyperimprovisation: Computer-Interactive Sound Improvisation. A-R Editions Inc., Middleton, Wisconsin.

[10] Levine, M. (1995) The Jazz Theory Book. SHER MUSIC CO. Petaluma, CA, USA.

[11] Manning, P. (2004) *Electronic and Computer Music*. Oxford University Press, New York, USA.

[12] Pachet, F. (2002) Interacting with a Musical Learning System: The Continuator. SONY-CSL, Paris, France. http://www.csl.sony.fr/¨pachet (Accessed 2 March 2006).

[13] Rowe, R. (1993) Interactive Music Systems. The MIT Press, Cambridge, Massachusetts, USA.

[14] Sims, K. (1991) Artificial Evolution for Computer Graphics, *ACM SIGGRAPH '91 Conference Proceedings*, Las Vegas, Nevada, July 1991.

[15] Thywissen, K. (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.*