EIMAS 2010

Ant Colony Optimization and Evolutionary Algorithms Applied to Jazz Solo Improvisation

Kjell Bäckman

West University, Trollhättan, Sweden

Abstract: This paper describes an experiment of using a variant of the TSP (Traveling Salesman Problem) of ACO (Ant Colony Optimization) and automatic fitness in the evolutionary algorithm to create jazz improvisation solos. It is a sub-project of our overall EJI (Evolutionary Jazz Improvisation) project, where we try to explore the inner nature of jazz music and model jazz improvisation solos and jazz harmony in the computer by means of evolutionary algorithms, swarm theory, chaos theory, neural networks, memetics and other kinds of heuristics.

Key words: jazz, improvisation, evolution, ACO, TSP.

Introduction

Biological theories, like evolution processes, swarm theory, neural networks etc., have during decades been used for the production of artworks, especially in the graphics art area by such forerunners as Sims (Sims, 1991), but also to some extent in the music area (Pachet, 2000), (Thywissen, 1996), (Dahlstedt, 2004). Most of these efforts have been made in the classical music domain, but some are also in the jazz area (Biles, 1994). However, the focus has mostly been on note level (pitch and note length), and very little on how to really build up a solo in a broader sense.

In the EJI project we take a deeper step into the area of jazz improvisation and try to model the way of building up a solo, raise and keep the musical intensity and make a solo musically meaningful. Our assumption is that biological processes could provide valuable contributions to this effort. However, the overall goal is not to create a self-playing jazz improvisation program, but rather to explore new ways of improvisation and free one’s own thinking from old habits and learnt routines and behavior, and consequently improve and renew our own musical quality in our jazz musician profession.

The experiment presented in this paper uses swarm intelligence, and especially the TSP (Travelling Salesman Problem) to build up an improvised jazz solo. The created solo is then used as a starting point in the evolutionary algorithm, where the automatic fitness function plays an essential part. Compared to manual fitness, the automatic fitness enables greater populations and a large number of generations within a manageable time.

The solos produced have been incorporated into tunes with harmonies created by other sub-projects of the EJI project. These results are available in the supplied sound examples. The results have also been evaluated and recorded by our live jazz group consisting of professional jazz musicians. Sound examples are provided.

Tim Blackwell and Peter Bentley (Blackwell, 2002) have written a program that mimics insect swarming to "fly around" the sequence of notes that the musician is playing, and improvise a related tune of its own. They believe that improvised music is self-organising in the same way as swarms of insects and flocks of birds.

Dahlstedt 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.

Background

Ants living in a colony walk around looking for food. They deposit pheromone, a chemical substance, along their walk, to guide other ants to the food. After a huge number of walks the shortest paths leading to the food will contain most pheromone. This is the ant colony (ACO) principle. The TSP problem is based on the ACO principle, and is characterized by searching for the shortest way for a salesman to travel around a set of cities, where a specific city should be visited once and only once, and the salesman has to return to the original city. In our experiment each city is represented by a specific note of the improvised solo within a certain pitch range. However, due to the particular characteristics of a jazz solo, a specific city (note) is allowed to be visited more than once during a solo, and every city (note) is not obliged to be visited. Neither is the solo forced to return to the original note.

The available space does not admit a full description of ACO and TSP. The reader can refer to (Michalewicz, 2000) for details about swarm theory, and to (Bonabeau, 1999) and (Kennedy, 2001) for details about the TSP algorithm. A brief conceptual overview is given in this paper. The focus of this paper is how the TSP algorithm is applied to jazz solo improvisation.

To avoid an endless repetition of the same note, a tabu list has been implemented with a parameter controlled length according to the algorithm proposed by Kennedy. When a new note is to be selected, the tabu list is checked to not contain that note. If not recently visited, the note is selected for the solo and also inserted at the end of the tabu list. When the tabu list has been filled up, the earliest added note is discarded from the list, which means that the same note can be selected when a certain number of notes have been played, equal to the length of the tabu list.

Each salesman’s tour represents the MIDI pitches of one solo. However, the MIDI representation also requires length and volume for each note, and these are picked from the rubber band principle described in another publication (Bäckman, 2010) by the author. This information together constitutes pitch, length and volume per note for an entire solo.

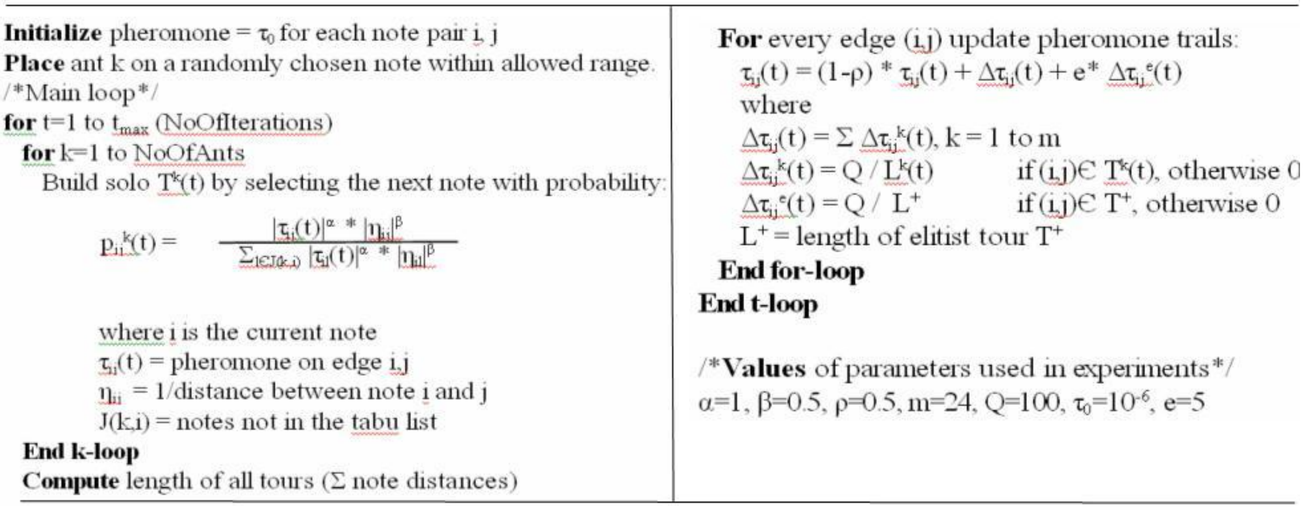
The evolutionary algorithm process in this project starts from an initial population resulting from the TSP process described above. The genetic representation (genome) is explained in (Bäckman, 2010) by the author. The fitness function then takes place by evaluating each individual and giving each a score value. The evaluation is made automatically by the computer program. The individuals with the highest score will most likely be parents for the next generation. The breeding is done by crossover of the genomes of two parents, optionally by applying a mutation somewhere in the genome.

The fitness selection and breeding is repeated generation by generation until we arrive at a genome good enough to be used for reproduction of a specific jazz solo.

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

The TSP Algorithm

The TSP algorithm as applied to the jazz solo production is given by Pict. 1. For a detailed description of the TSP algorithm, refer to (Kennedy, 2001).



```
Initialize pheromone = $\tau_0$ for each note pair i, j
Place ant k on a randomly chosen note within allowed range.
/*Main loop*/
for t=1 to $t_{max}$ (NoOfIterations)
  for k=1 to NoOfAnts
    Build solo $T^k(t)$ by selecting the next note with probability:

        $p_{il}^k(t) = \frac{|\tau_{il}(t)|^\alpha * |\eta_{il}|^\beta}{\Sigma_{l \in J(k,i)} |\tau_{il}(t)|^\alpha * |\eta_{il}|^\beta}$

        where i is the current note
        $\tau_{il}(t)$ = pheromone on edge i,j
        $\eta_{ii}$ = 1/distance between note i and j
        J(k,i) = notes not in the tabu list
  End k-loop
  Compute length of all tours ($\Sigma$ note distances)
  For every edge (i,j) update pheromone trails:
    $\tau_{ij}(t) = (1-\rho) * \tau_{ij}(t) + \Delta\tau_{ij}(t) + e* \Delta\tau_{ij}^e(t)$
    where
    $\Delta\tau_{ij}(t) = \Sigma \Delta\tau_{ij}^k(t)$, k = 1 to m
    $\Delta\tau_{ij}^k(t) = Q / L^k(t)$          if (i,j)∈ $T^k(t)$, otherwise 0
    $\Delta\tau_{ij}^e(t) = Q / L^+$             if (i,j)∈ $T^+$, otherwise 0
    $L^+$ = length of elitist tour $T^+$
  End for-loop
End t-loop

/*Values of parameters used in experiments*/
$\alpha=1, \beta=0.5, \rho=0.5, m=24, Q=100, \tau_0=10^{-6}, e=5$
```

Picture 1. The high-level TSP algorithm as applied to jazz solo creation.

The parameter values given in Pict. 1 have been used in this experiment. We have also experimented with other values, however without any audible improvement. A discussion of the meaning of the parameters is given below.

The α and β parameters can for instance be used for balancing the impact of distances between notes against the pheromone trail. A small α value will premier small distances between notes, where α=0 will have the effect of minimizing the note intervals. If β=0, only pheromone trail amplification is at work.

Also the ρ parameter impacts the pheromone trail by controlling the pheromone decay for each iteration. A value just below 1 means a rapid decay, while a value just above 0 preserves the earlier pheromone replenishment.

The Q parameter controls the amount of pheromone replenishment; the greater value, the larger quantity of pheromone replenishment. It should be balanced against the initial pheromone τ_0.

The e parameter controls the number of elitist ants and has a similar effect as the Q parameter.

The tabu list length can be shortened to allow more frequent repetitions of a single note, and vice versa. In this experiment we have used the length of 2, to allow for rather frequent repetitions of each note.

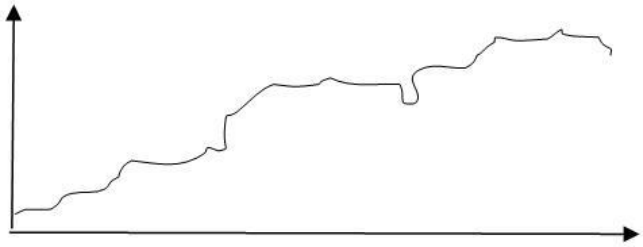
The distance between adjacent cities (notes) in our experiment has been set to 1, i.e. we utilize the complete set of notes within each octave, which implies a kind of twelve-note approach.

The Evaluation Process

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

The evaluation functions examine jazz solo features revealed by the author's study of 73 great masters in jazz history These functions contribute with the detailed scores and correspond to the categorization of the techniques resulting from the solo analysis of the 73 great masters, to be published in a close future. Furthermore, the score value 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. The space available in this article does not admit a closer discussion of each evaluation function, but they comprise techniques as sound, density, repetition, sequences, phrasing, polyrhythm, chromatics, rest utilization, doubling of tempo, swing and many more.

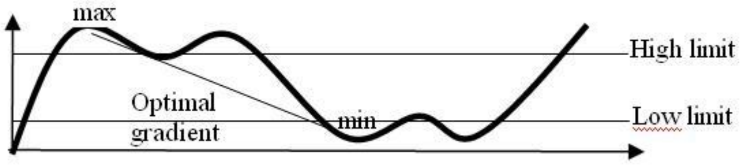
Since the intensity provided by a melody fragment tends to stay in the listener's ear for some time, the score value per bar will be preserved to some extent; 50% of the intensity score value for one bar are added to the score value of the next bar. Thus each bar will contribute to the score value of the next bar with half of its value. The score value per bar will have a graphic representation something like in Pict. 2.



Picture 2. Calculated score value per bar during the solo, with 50% accumulation from the previous bar.

Having calculated the score value per bar, it is time to compare it to an optimal curve, reflecting some kind of “perfect” intensity fluctuation model. The aim is to make the solo intensity level align to the optimal curve as closely as possible.

There may be several intensity maximum and minimum points, which can occur anywhere in the solo and in any sequence. We classify a point above a specific limit as a climax (max point). The limit is specified as a percentage of the overall maximum point. The percentage is a control parameter. 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. This percentage is also a control parameter. 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 the 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 Pict. 3.



Picture 3. Several maxima and minima in sequence.

The system rewards great differences between max and min points. The greater differences, the greater score value will be assigned. The value added to the accumulated score value 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 from the accumulated score value. 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 alignment to the optimal gradients, will render the highest score value.

Fitness Selection and Breeding

Having evaluated the individuals of the initial population, the ones with the highest score value, i.e. which most perfectly align to the optimal intensity curve, will have the best chances to be selected as parents for breeding. The selection of two parents for a single child is made stochastically, based on their evaluation score, implying that the best parents will create the greatest 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 specified 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.

Breeding is carried out in the following way. Two parents are selected according to the description above, a child genome is created by crossover, and point mutations are performed on a probability basis. The mutations will imply a shift between two genome values, or a slight modification of a genome value. Then the child is evaluated and, if it has a better score value than the worst one, it will replace the worst individual, which is discarded from the population.

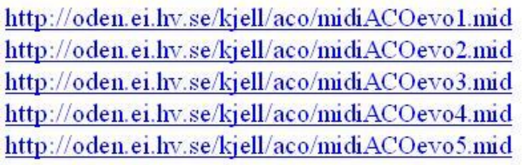
The fitness selection and breeding is repeated a number of times, specified by a control parameter. We have experimented with 10 000. It is also possible to perform the repeated selection and breeding until the total score is high enough, also specified by a control parameter.

Experiment Example

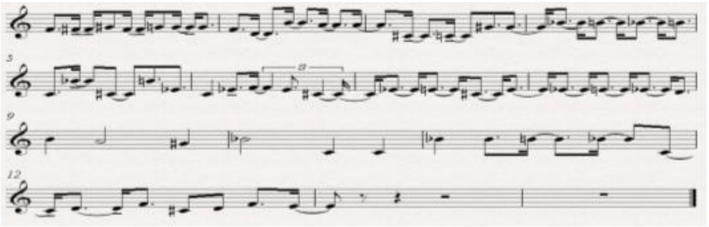
Our experiment consists of two steps; first, an initial population of improvised solos is produced by the TSP algorithm, and then the evolutionary algorithm continues to further develop the population.

In the TSP procedure, 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 3,000 iterations

In this test run, we used an allowed pitch range between 50 and 74, which approximately corresponds to two octaves around the key-hole of the piano. In the second step, in the evolutionary algorithm process, another 10 000 iterations were performed using the TSP produced solos as an initial population. Pict. 4 shows the links to the five best solos, i.e. the ones with the highest score.



http://oden.ei.hv.se/kjell/aco/midiACOevo1.mid
http://oden.ei.hv.se/kjell/aco/midiACOevo2.mid
http://oden.ei.hv.se/kjell/aco/midiACOevo3.mid
http://oden.ei.hv.se/kjell/aco/midiACOevo4.mid
http://oden.ei.hv.se/kjell/aco/midiACOevo5.mid



Picture 4. The five best solos and the first bars of the first sound example.

The solo in Pict. 4 (the first of the five ACOevo links listed above) and another of the five best solos have been incorporated into a composition created by other of the EJI subprojects, the first one played by an alto saxophone, and the other by a flugelhorn. This can be heard on the following link:

http://oden.ei.hv.se/kjell/aco/RandomLicksACOevo.mp3

In addition, the same tune has been recorded by our live jazz group and can be heard on the following link:

http://oden.ei.hv.se/kjell/aco/RandomTrio.mp3

Conclusion

Since the overall goal was not to produce a fully professional solo as played by the great jazz masters, we did not expect a master class result, especially since a good solo includes other aspects like sound, connection with fellow musicians, audience connection, body movement and other performance related characteristics, which cannot be reflected by the basic MIDI representation consisting of pitch, note length and volume, that we used in this experiment, but the resulting output contains some unorthodox features as regards phrasing, unpredictable note intervals and sudden note burst outs. We believe that the presupposition of exploring new ways of jazz improvisation, has to some extent been fulfilled.

The sounding output of the TSP algorithm has a clear atonal, twelve-note, avant-garde character, as an effect of the equidistant city (note) prerequisites. To make the output more pleasant to people in common with no or little experience of modern jazz, and to make the solo more appropriate in a standard mainstream jazz context, we will in the future experiment with the functions for accommodation of the pitches to the scales in question, and the note lengths quantized to the rhythmic beat. These functions are described in other papers (Bäckman, 2010).

References

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.

BLACKWELL, T., BENTLEY, P. (2002) *Insect Swarming Inspires Jazz Software*. Science & Technology News Service. http://www.newscientist.com/article.ns?id=dn2242 (Accessed 29 May 2010).

BONABEAU, E. et al. (1999) *Swarm Intelligence – From Natural to Artificial Systems*. Oxford University Press, New York.

BÄCKMAN (2010) *Various publications*, http://oden.ei.hv.se/kjell/eji/eji.htm

DAHLSTEDT, P. (2004) *Sounds Unheard of - Evolutionary algorithms as creative tools for the contemporary composer*, PhD thesis, Chalmers University of Technology, Gothenburg.

KENNEDY, J. et al. (2001) *Swarm Intelligence*. Academic Press, San Diego, USA.

MICHALEWICZ, Z., FOGEL, D. (2000) *How to Solve It: Modern Heuristics*. Springer-Verlag, Berlin Heidelberg, Germany.

PACHET, F. (2000) *Rhythm as Emerging Structure*. In Proceedings of ICMC, pages 316-319, Berlin, Germany. ICMA

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

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.

Thanks to my colleagues at my university, my fellow musicians in our acoustic jazz group and my supervisor Palle Dahlstedt for useful comments and support, and above all, to my big family allowing me to work with jazz playing and research.

Kjell Bäckman (born 1949 Hunnebostrand Sweden) is a teacher and PhD student in applied informatics at West University, Trollhättan, Sweden, with a profile in systems development and programming. He has been a teacher in piano and jazz improvisation at Gothenburg University, Sweden. He has also toured with professional jazz groups around the world.

Home page: http://oden.ei.hv.se/kjell

Mail address: kjell.backman@hv.se