PROBABILITY OF INCOMING CHORDS AND EXPECTATION FOR CHORDS

Nart Bedin Atalay
Middle East Technical University, Ankara, Turkey

ABSTRACT

One of the theories of the perception of chords bases on neural networks. MUSACT proposes that perception of chordal structures and chordal progressions can be explained with spreading activation model. There are numerous experimental evidences that corroborate MUSACT as a model of chord perception. Besides, it was shown that exposing a neural self-organization mechanism to chord sequences results in MUSACT equivalent architecture. On the other hand, recent experimental evidences suggest that structural representation and chordal representation MUSACT may not be sufficient.

In this study, I would like to investigate whether statistical learning from environment is sufficient to extract regularities for chord progressions. A recurrent neural network is trained in a rich environment. The environment includes passing, auxiliary, and appoggiatura chords, secondary dominants and modulations.

Test results suggest that statistical learning can govern expectations for incoming chord in complex sequences.

1. INTRODUCTION

One of the computational theories of the perception of chords bases on connectionism. MUSACT (Bharucha, 1987) models the expectation for incoming chord after being listened to a chord or chord progression. Expectation for incoming chord is explained as a result of spreading activation in the network. Activation map of the network is determined by constraints encoded in connection weights. There are numerous experimental results that corroborate predictions of MUSACT (Bharucha and Stoeckig (1986), Tekman and Bharucha (1998), Tillmann, Bharucha, Bigand (2000)). It was shown that exposing a neural self-organization mechanism to chord sequences results in MUSACT equivalent architecture (Tillmann, Bharucha, Bigand, 2000).

On the other hand, recent experimental results suggest that MUSACT has inaccurate predictions for certain structural relations of chords in sequences (Horton, 2002) and for certain types of chords (Tekman and Atalay, 2002). One of the arguments against foregoing experimental results in favor of connectionism may be that the training environment of MUSACT does not include such chord structures and chord types. This is why predictions of the model and experimental results do not fit.

In this study I would like to investigate whether statistical learning from environment is sufficient to develop expectations for complicated chordal relations in chord progressions. I would like to use much more complex training sequence than those used in previous simulations. In order to investigate this question a simple recurrent neural network (Elman, 1990) is implemented. This network is trained in an environment that includes auxiliary, passing and appoggiatura chords, secondary dominants and modulations. Then, performance of the network is tested.

2. ENVIRONMENT

For our purposes, it is not necessary to use chord sequences that embody all kinds of rules expressed in a harmony book. In this study, training set includes appoggiatura, auxiliary and passing chords, secondary dominants and modulations to other keys.

Chord progressions were generated by context free re-write rules and transformational rules. I do not claim that those re-write and transformational rules are 'the rules' for describing chord progressions. They simply serve as an artificial intelligence tool for implementing appropriate prescripts in a chord sequence production program. Figure 1 displays those rules. Re-write rules are expressed in *Definite Clause Grammar* (DCG) notation. DCG rules can be read as generalized context free grammar rules (Pereira and Shieber, 1987). Chord progressions were generated in twelve major and twelve harmonic minor modes. Applying re-write and transformational rules creates chord sequences that include various relations between chords. An example sequence in Figure 2 depicts some of those relations.

One constraint for chord progressions, which is not expressed clearly in the rules of Figure 1, should be mentioned. Sixth and seventh degrees of harmonic minor scale is chromatically raised or lowered with respect to the direction of melody. If the melody is in ascending form the sixth degree is raised. If the melody is in descending form the seventh degree is lowered. These transformations become necessary for functional elaborations and for main chord progressions. Transformational rules that handle functional elaborations are implemented well. Rules that necessary for handling main chord progressions did not implemented. As a result of this, the environment may include chord progressions that violate aforementioned rules in bass notes of chords. For example, in the environment, G major six-five chord may immediately precede A flat major five-three chord.

A few notes are also necessary for modulating pieces. The first step for modulation is to select a pivot chord. Pivot chord is a chord that belongs to both host key and modulating key. To select a pivot chord, a progression in host key is created. From this sequence, a chord is selected randomly with the constraint that pivot chord cannot be an elaboration chord. Subsequently, list of tonalities that include pivot chord is created. From this list a tonality is chosen randomly. For example, a sequence in C major may modulate to one of following tonalities: C major, G major, F major, D major, B flat major, A minor, E minor, D minor, B minor, C minor, F minor. Then, the function of pivot chord is determined and progression started from that function is created.

ISBN 3-931852-67-9
ISSN 1617-6847

Lastly, appropriate transformation rule is applied. This rule erases chords in modulating progression come after the pivot chord in host tonality and attaches the sequence starts with pivot chord in new tonality.

3. NEURAL NETWORK

A simple recurrent neural network with back-propagation through time learning algorithm was implemented. Being statistical learning machines, recurrent networks are considered as an appropriate architecture for learning regularities in time (Elman, 1990) and back-propagation algorithm tries to find relevant aspects of input history to predict the incoming event. (Mozer, 1994).

Figure 3 shows layers of the network. Input layer is the notes layer and output layer is the chords layer. Nodes in the notes layer represent 12 chromatic pitches. Nodes in the chords layer represent 60 chords (12 major, 12 minor, 12 diminished, 12 augmented and 12 dominant seventh chords). Representations of notes and chords are local. By the way, notes layer activation can be seen as distributed representation of chords. As a consequence, activation in the chords layer can be calculated with a transformation function that maps distributed activation of chords in the notes layer to probability distribution in the chords layer. Bridle (1990) proposed such function (in (Mozer, 1994)) (please refer to Figure 4). After transformation, activation of a chord unit can be interpreted as a probability of incoming that chord. This principle was successfully used for composition of music pieces (Mozer, 1994).

4. TEST RESULTS

The network performance was tested with sequences that not appeared in the training set. Tests were aimed to understand how well the network discovered specific relations, which are implicit in the training set. First, the network's ability to recognize secondary dominants is tested. Chord sequence 1 was constructed for this purpose. (1) [C minor, F minor, C minor, A flat major, B natural diminished]. B natural diminished chord in C minor tonality can the secondary dominant of the secondary dominant of F minor chord or the dominant of the sequence. Figure 5 shows the activation pattern of highly activated chords in output layer after the first sequence was given. Activation pattern corroborates the claim that the network learned secondary dominants. Second, network's ability to follow passing elaboration is tested. Chord sequence (2) was used for this purpose. (2) [C minor, F minor, C minor, A flat major, F minor, G major, F sharp dim]. Figure 6 shows highly activated nodes after the sequence. G major is the most expected chord. It so seems that the network recognizes passing sequences. Lastly, the network's ability to recognize modulation was tested. (3) [C minor, F minor, C minor, A flat major, F minor, B diminished]. (4) [C minor, F minor, C minor, A flat major, F minor, G diminished]. Tonalities in chord sequences (3) and (4) are ambiguous till the last chord appears. Those chords determine number of flats for continuing sequence. Figure 7 and Figure 8 shows values of highly activated chords after sequence (3) and (4) respectively. Activations suggest that expectation of the network was mostly shaped by secondary dominant principle. Besides, as a result of transformation function, activations were also shaped by number of shared notes between chords.

5. CONCLUSION

It is for sure that detailed tests are still necessary to reach accurate conclusions about the results of this simulation. However, initial test results suggest that statistical learning can be sufficient to extract detailed relations between chords in sequences.

6. REFERENCES

1. Bharucha, J. J. (1987). MUSACT: A connectionist model of musical harmony. In *Ninth Annual Conference of the Cognitive Science Society,* p. 508-517, Hillslade. NJ: Erlbaum.

2. Bharucha, J. J. (1987) and Stoeckig, K. (1986). Reaction time and musical expectancy: Priming of Chords. *Journal of Experimental Psychology: Human Perception and Performance,* 12:403-410.

3. Bridle, J. (1990). Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation of parameters. In D. S. Touretzky (Ed.), *Advances in neural information processing systems 2 (p. 211-217).* San Mateo, CA: Morgan Kaufmann.

4. Elman, J.L. (1990). Finding structure in time. *Cognitive Science*, 14:179-211.

5. Horton, T. (2002). Some data that falsify spreading-activation accounts of global context effects in tonal music. In C. Stevens, D. Burnham, G. McPherson, E. Schubert, and J. Renwick, editors, *Proceedings of the 7th International Conference on Music Cognition and Perception.*

6. Mozer, M.C. (1994). Neural network music composition by prediction: Exploring the benefits of psychophysical constraints and multiscale processing. Connection Science, 6, 247-280.

7. Pereira, F.C.N. and Shieber, S.M. (1987). *Prolog and Natural Language Analysis.* CSLI: Standford.

8. Tekman, H.G. and Atalay, N.B. (2002). Is there a chord priming beyond associative networks? In C. Stevens, D. Burnham, G. McPherson, E. Schubert, and J. Renwick, editors, *Proceedings of the 7th International Conference on Music Cognition and Perception.*

9. Tillmann, B., Bharucha, J.J., Bigand, E. (2000). Implicit Learning of Tonality: A Self-Organizing approach. *Psychological Review,* 885-913.

ISBN 3-931852-67-9
ISSN 1617-6847

Transformational Rules:

1) Place auxilary chord to the right of the main chord, duplicate the main chord and place dublication to the right of the auxiliary chord.

2) Place passing chord to the right of the main chord, duplicate the main chord and place dublication to the right of the passing chord.

3) Place appogiature chord to the left of the main chord.

4) In Modulating_Piece, erase chords after the pivot in host tonality, erase chords before the pivot chord in new tonality, and add remaining chords in new tonality to the chords in host tonality as a child of appropriate mother.

5) In Functional_Elaborations, raise the sixth degree if the melody is in ascending form, lower the seventh degree if the melody is in descending form.

Re-write Rules:

Piece → NonModulating Piece | Modulating Piece
NonModulating_Piece → Progression (tonality)
Modulating_Piece → Progression (tonality) Modulating_Progression Progression (new_tonality)
Modulating_Progression → Progression (host_tonality) Pivot_Chord_and_New_Tonality Progression (new_tonality)
Pivot_Chord_and_New_Tonality (please refer to the article for detailed explanation of this procedure)
Progression (tonality) → Tonic(start) SubDominant Dominant Tonic(end)
Tonic(start) → [Adjunct (tonic)] Chord (tonic) [Adjunct (tonic)]
Tonic(end) → [Adjunct (tonic)] Chord (tonic)
SubDominant → [Adjunct (subdominant)] Chord (subdominant) [Adjunct (subdominant)]
Dominant → [Adjunct (dominant)] Chord (dominant) [Adjunct (dominant)]
Chord (tonic) → I_53 Elaboration | I_63 Elaboration
Chord (subdominant) → II_53 Elaboration | II_63 Elaboration | IV_53 Elaboration | IV_63 Elaboration
Chord (dominant) → V_53 Elaboration | V_63 Elaboration | V_7 Elaboration | V_65 Elaboration | V_43 Elaboration | V_2 Elaboration
Adjunct (tonic, major_keys) → III_53 Elaboration | III_63 Elaboration | VI_53 Elaboration | VI_63 Elaboration
Adjunct (tonic, minor_keys) → VI_53 Elaboration | VI_63 Elaboration
Adjunct (subdominant, major_and_minor_keys) → VI_53 Elaboration | VI_63 Elaboration
Adjunct (dominant, major_keys) → III_53 Elaboration | III_63 Elaboration
Elaboration → Secondary_Dominant | Functional_Elaborations | null
Secondary_Dominant → Secondary_Dominant Chord (dominant of dominant) | null
Functional_Elaborations → Appogiature_63_Chord | Passing_63_Chord | Appogiature_64_Chord | Auxilary_64_Chord | Passing_64_Chord | Auxilary_43_Chord | Passing_43_Chord
Appogiature_63_Chord (tonic) → appogiature_chord
Appogiature_63_Chord (dominant) → appogiature_chord
Passing_63_Chord (any_degree) → passing_chord
Appogiature_64_Chord (any_degree) → appogiature_chord
Auxilary_64_Chord (any_degree) → auxiliary_chord
Passing_64_Chord (any_degree) → passing_chord
Auxilary_43_Chord (any_degree) → auxiliary_chord
Passing_43_Chord (any_degree) → passing_chord

Figure 1: DCG rules and transformational rules for chord production. Square brackets refer mutually exclusive optional choices. Dash lines refer mutually exclusive choices.

ISBN 3-931852-67-9
ISSN 1617-6847

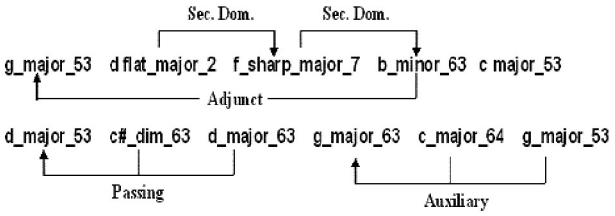

Figure 2: An example sequence that shows relations between chords.

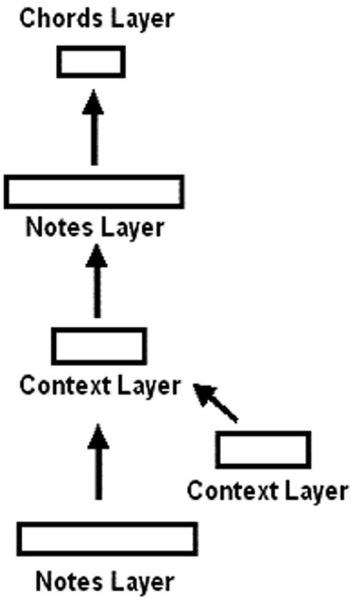

Figure 3: Architecture of the network.

$$ch = \frac{\bar{e}^{d_i}}{\sum_j \bar{e}^{d_j}}$$

Figure 4: Transformation function that maps notes layer to chords layer.

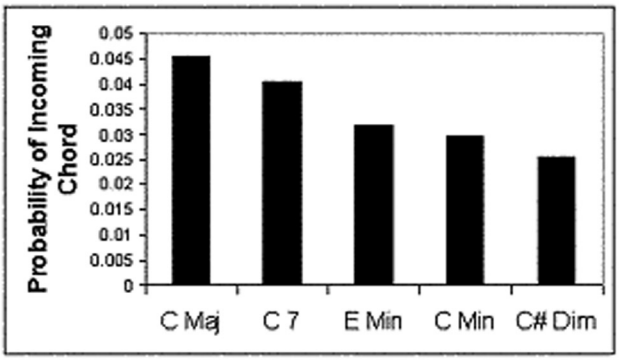

Figure 5: Activation values of highly activated chords after sequence (1) is given to the network.

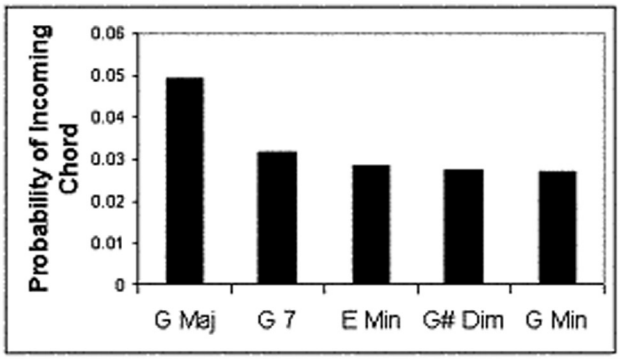

Figure 6: Activation values of highly activated chords after sequence (2 is given to the network.

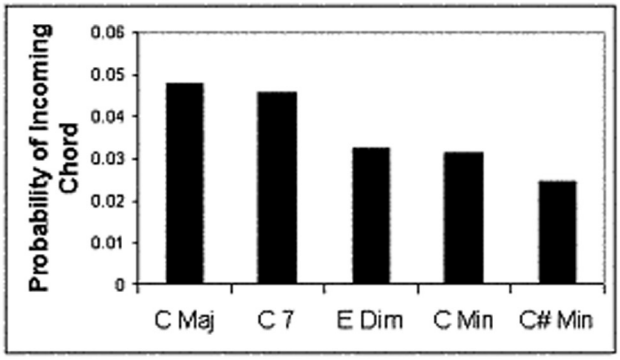

Figure 7 Activation values of highly activated chords after sequence (3 is given to the network.

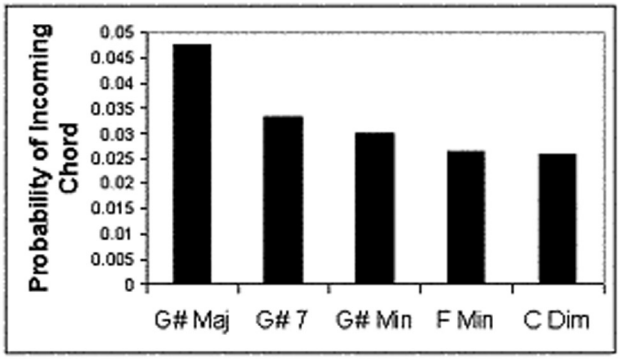

Figure 8 Activation values of highly activated chords after sequence (4 is given to the network.

ISBN 3-931852-67-9
ISSN 1617-6847