Computational Linguistics and Chinese Language Processing
Vol. 15, No. 1, March 2010, pp. 1-18

Harmony Graph, a Social-Network-Like Structure, and Its Applications to Music Corpus Visualization, Distinguishing and Music Generation

Wei-An Chen*, Jihg-Hong Lin+, and Shyh-Kang Jeng#

Abstract

In this research project, we propose a model, the Harmony Graph, to decompose music into a social-network-like structure according to its harmonies. The whole Harmony Graph network represents the harmony progressions in music. The Harmony Graph is utilized to visualize, distinguish, and generate music for four prepared corpora using social network techniques. We experimented on different characteristics in social network analysis, and we found significant differences among the Harmony Graphs of the four corpora. A new measure called *Agglomeration* is created to characterize the agglomerating phenomenon that cannot be described sufficiently by existing measures. A corpus-based music composition method is also proposed in this research. By performing random-walk in a Harmony Graph, we generated new music that differs from yet reflects the style of music pieces in the corpus. With the link prediction technique, we also generated music more pleasant aurally than simply using random walks.

Keywords: Social Network Analysis, Corpus Visualization, Corpus-Based Generation

* Computer Music Laboratory (JCMG), Graduate Institute of Communication Engineering, National Taiwan University
E-mail: r98942051@ntu.edu.tw

\+ Business & Marketing Strategy Labortory, Chunghwa Telecommunication Laboratories
E-mail: johnlin@cht.com.tw

\# Department of Electrical Engineering and Graduate Institute of Communication Engineering, National Taiwan University
E-mail: skjeng@cc.ee.ntu.edu.tw

1. Introduction

Corpus is the basis of Corpus Linguistics. It is also important in Computer Music (Manaris *et al*., 2007). In the research of corpus-based generation, such as generating articles from text corpora (Stribling *et al*., 2009; Marom & Zukerman, 2005), speech synthesis based on audio corpora (Iida *et al*., 2003), and generating music from music corpora (Conklin, 2003; Polashek *et al*., 2005), one interesting topic is that the selection of corpora may lead to results with different styles. For example, an article generated from a corpus of Abraham Lincoln may reveal his style, which reads differently from one generated from a corpus of William Shakespeare. In this paper, we develop a new model, the Harmony Graph, to make music using corpus-based generation and use this model for music-corpus distinction.

The Harmony Graph is applied to organize a music corpus into a social-network-like structure, analogous to the Word Graph (Oerder & Ney, 1993) in Corpus Linguistics. Four distinct music corpora were prepared that are collections of music in different genres.

In Corpus Linguistics, there has been relevant research on text corpus visualization (Paley, 2002; Fortuna *et al*., 2005; Rohrer *et al*., 1998), which generally provides the overall concept of the corpora. Nevertheless, they cannot be used to tell corpora apart at a glance. In the area of Computer Music, distinction of music pieces into different genres using Harmony Graphs is found to be accurate and believed to be new.

In addition to visual inspection, we apply social network analysis to these Harmony Graphs. The calculated measures, namely degree distribution, average path length (APL), and clustering coefficient (CC), indicate that social network analysis is very useful for distinguishing corpora. We also devise a measure, Agglomeration, to capture the density of connections in the graph. This new measure is found to be even more helpful in distinguishing music corpora.

In corpus-based music generation, we begin by performing a random-walk in a Harmony Graph to generate music. For zero-occurrence smoothing, we apply the link prediction technique in social network methodology to add potential edges, and increase the variety of produced music. The generated music somewhat reflects the style of selected corpus according to results of subject tests. Although some relevant research regarding regeneration of music styles has been published (Dubnov *et al*., 2003; Trivino-Rodriguez & Morales-Bueno, 2001; Pachet, 2003), the Harmony Graph model stands out for being visualizable, analyzable and interpolable by social network methodology.

Sequential music pattern mining has also been applied to model music by finding important sequences sampled from a database and using the model to classify or generate music (Shan *et al*., 2002). Harmony Graph, however, is an approach quite different from pattern mining. After constructing the Harmony Graph, the music content is reduced from sequential data into a folded form, and attributes like APL, CC, and Agglomeration can be retrieved from it, based on social network analysis techniques. These attributes cannot be retrieved from pattern mining models directly.

To quantitatively evaluate the Harmony Graph's distinguishing ability, we built up a classifier according to the results of the social network analysis. Given a music input, the classifier predicts which corpus it belongs to by the social network features of its Harmony Graph. The accuracy of our experimental result is 73% out of 59 songs. As for the evaluation of corpus-based music generation, we conducted a subject test. In about 70% of the test queries, subjects agree that the generated music matches its corresponding corpus best in style among five choices. A demonstration program is also provided on the Internet for free download.

2. Experimental Setting and Model

2.1 The Four Corpora

In this study, we prepared four corpora, namely polyphony, homophony, pantonality, and atonality. Polyphony and homophony are tonal music, and the other two belong to atonal music. These corpora will be used for visualization, social network analysis, and corpus-based music generation. Table 1 shows the details.

Table 1. Four corpora used in this paper

Genre	**Composer** and Works
polyphony	**Bach** Inventions, Sinfonias, preludes, fuga
homophony	**Chopin** etude, ballade **Mendelssohn** Songs Without Words
pantonality	**Prokofiev** toccata, prelude, sonata **Shostakovitch** toccata, prelude, sonata
atonality	**Schoenberg** Klavierstücke

2.2 The Harmony Graph

Western music evolved from modal music in the Middle Ages to polyphonic music, glorified by Bach at its peak, and gradually became homophonic, which is music with melody accompanied by chords. The development of Harmony has been mature. As time progressed into the 20th century, the breakdown of tonality led to escape from harmony rules. In this research, we do not refer "harmony" as "chords" in classical Harmony. Rather, in a wider sense, we refer harmony as "the notes played simultaneously."

In this sense, we build a graph from music accordingly, which is named Harmony Graph. A node of a Harmony Graph is a harmony represented by a string of note names, *e.g*. "D F #A". The octave information is suppressed, which means that both C1 and C2 are regarded as the same, and are notated as C. The links of a Harmony Graph represent note changes, that is, the progression of harmonies. Notice that, for simplicity of explanation, Harmony Graph here does not contain any temporal information, such as beat and rhythm, but only the progression of harmonies, namely pairs of harmonies that are temporally neighbored.

In addition, we create a "null" node, at which the music starts and ends. The music starts from null to the first harmony, and ends from the last harmony to null. Null also represents rests, where no notes occur.

Harmonies are encoded as a 12-bit binary number, corresponding to the twelve tones in an octave. For example, "000000000001" represents C, "001000000010" represents "A C#", and so on. There are 4096 possible combinations of all harmonies. Hence, each harmony can also be represented by an integer from 0 to 4095, including the null node “000000000000”.

The weight of each link represents the number of times that the same progression happens. For example, the more harmony A to harmony B occurs in a piece of music, the higher the weight of the link AB will be.

We use MIDI as raw data format. To simplify the problem, we consider only the onset time, offset time, and the pitch position of each MIDI event.

Figure 1 is an example of how to build a simple Harmony Graph. Three steps are required to construct the Harmony Graph of one music piece:

Step 1. Extracting notes.

Scan the sheet music along the time line and record notes happening at the same time as a harmony. As soon as a note combination changes, a new harmony is generated and recorded.

Step 2. Suppressing octave information.

Suppress the octave information of the harmonies obtained in Step 1, and merge the notes with the same note name. For example, the first harmony (C3 E3 G3 C4) becomes (C E G), because C3 and C4 are both C, just in different octaves.

Step 3. Constructing the graph.

Connect the harmonies in Step 2 according to their sequential order. Then, link the null node to the first harmony, and link the last harmony back to the null node. Furthermore, rest notes in the music piece are treated as the null node. After connecting the harmonies and the null node, a Harmony Graph is accomplished.

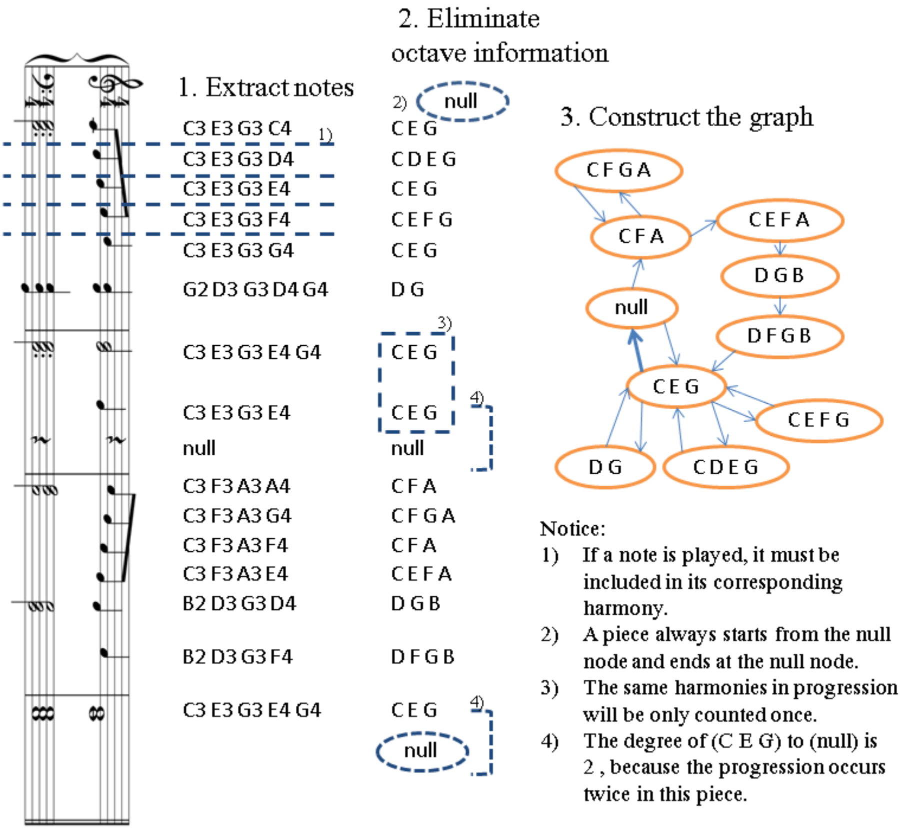

Figure 1. A simple example of constructing a Harmony Graph

3. Results

3.1 Corpus Visualization

Graphviz (Ellson *et al*., 2002) is applied to visualize a Harmony Graph. We have found that its built-in fdp engine is especially suitable for drawing graphs, because the higher-degree nodes will be placed closer to the center and the lower-degree nodes closer to the boundary. This makes it easier to observe the characteristics of the graph.

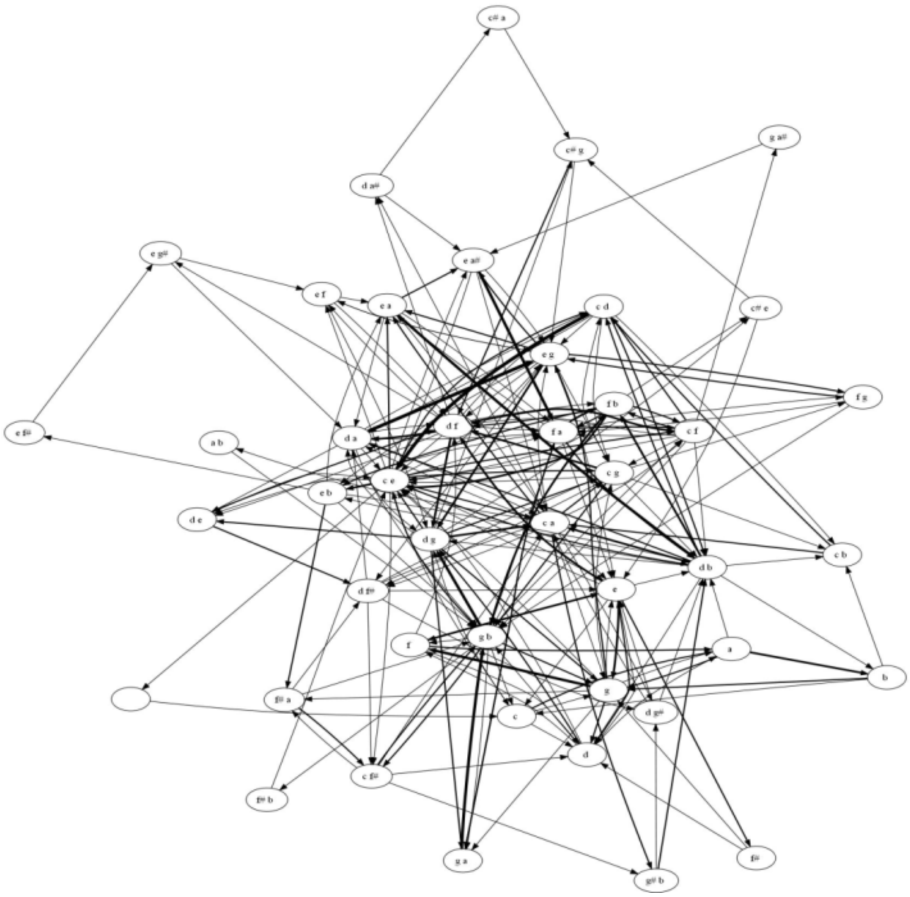

Figure 2. Visualized Harmony Graph of Bach’s Invention No. 1, a polyphonic piece

Figures 2 to 5 show some representative outcomes of our corpora from four genre of music.

Figure 2 is derived from Bach's “Invention” No. 1, and is a representative Harmony Graph for polyphonic music. We find:

The number of notes in each single node is at most two, because “Invention” is two-part polyphonic music, like a dialogue between two melody lines. Therefore, there are a maximum of two notes at the same time. The upper bound of node number is 79 for two-part polyphonic music, since

$$\mathrm{C}_2^{12} + \mathrm{C}_1^{12} + 1 = 79. \tag{1}$$

Near the center of the picture, nodes are connected with each other in a very complicated way. We call this agglomeration, which will be discussed further in Section 3.2.4.

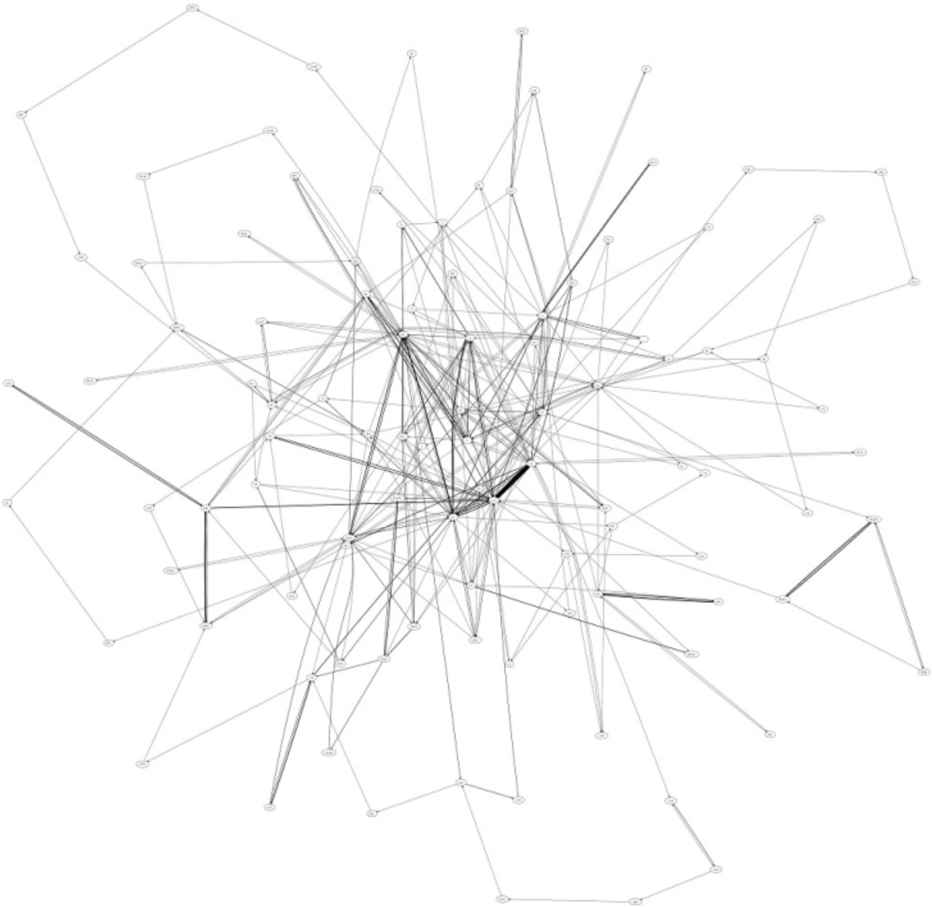

Figure 3. Visualized Harmony Graph of Mendelssohn's Song Without Words, Op. 19-2, a homophonic piece

Close to the border of the picture, a small number of nodes have only one incoming link and one outgoing link. This means that these harmonies are used only once in the whole masterpiece. These harmonies tend to be special ones used by the composer.

Figure 3 is the Harmony Graph of "Song Without Words", Op. 19-2 by Mendelssohn, and is a representative for homophonic music. It has a larger scale with more nodes than Fig. 2 has. And the phenomenon of agglomeration is also obvious. In the periphery, however, there are more lower-degree nodes. This may stand for the more freedom of harmony usage, compared with polyphonic music. Furthermore, unlike in Figure 2, we can find a thick link in Fig. 3. Similar links are also found in other graphs. We infer that this thick link is the outbound of the tonal center.

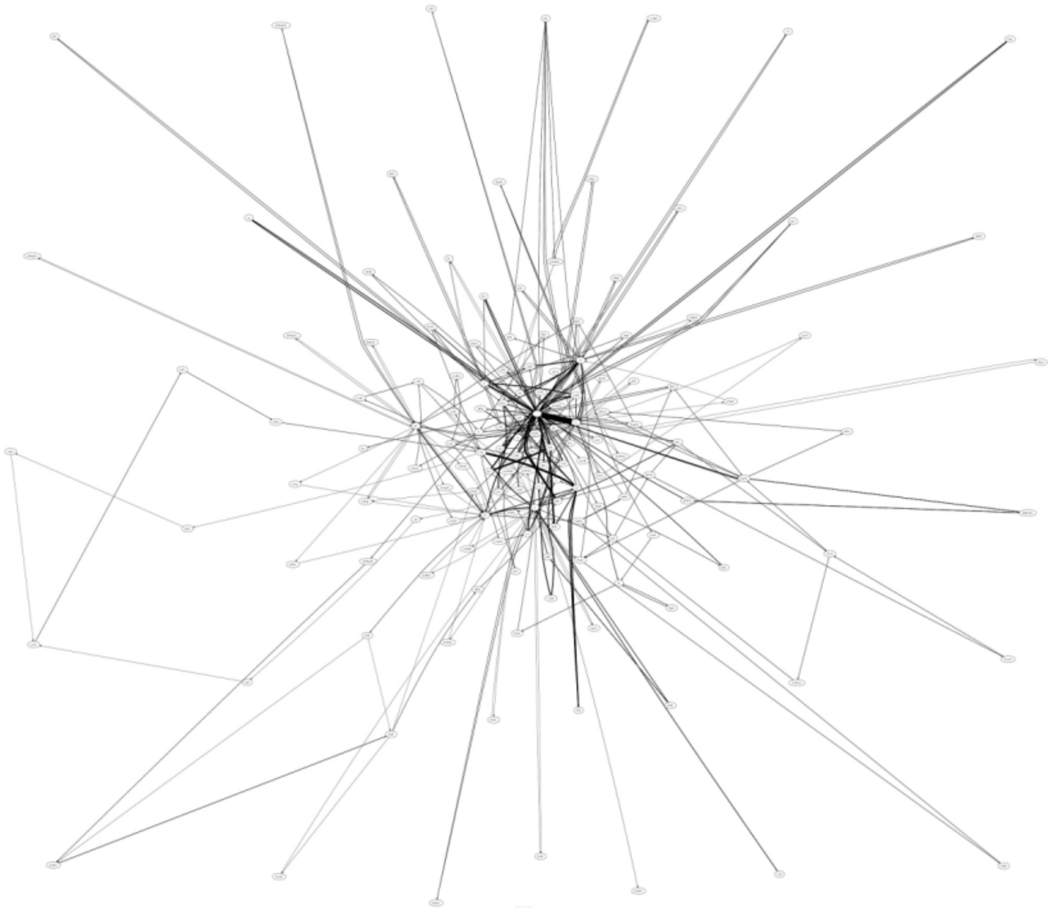

Figure 4. Visualized Harmony Graph of Prokofiev's Sonata, Op. 14, Movement 2, a pantonal piece

Figure 4 is the picture of Sonata, Op. 14, Movement 2, by Prokofiev, which is a pantonal piece. Apparently, the number of peripheral nodes is much more than in the previous two figures. This is an indication of atonal music and the increased freedom in usage of harmonies. The agglomeration is less obvious, which means that the treatment of harmonies is less confined than traditional music. We also can see thick links near the center. After inspection of Prokofiev's graphs in general, we find that the thick links are mostly linked to the null node, which may be an indication that Prokofiev treats the piano as a percussive instrument.

Figure 5 shows Schoenberg's "Klavierstücke", Op. 19-5, which is atonal piece. Compared with the previous ones, there is almost no agglomeration, which means a more distant relationship among harmonies.

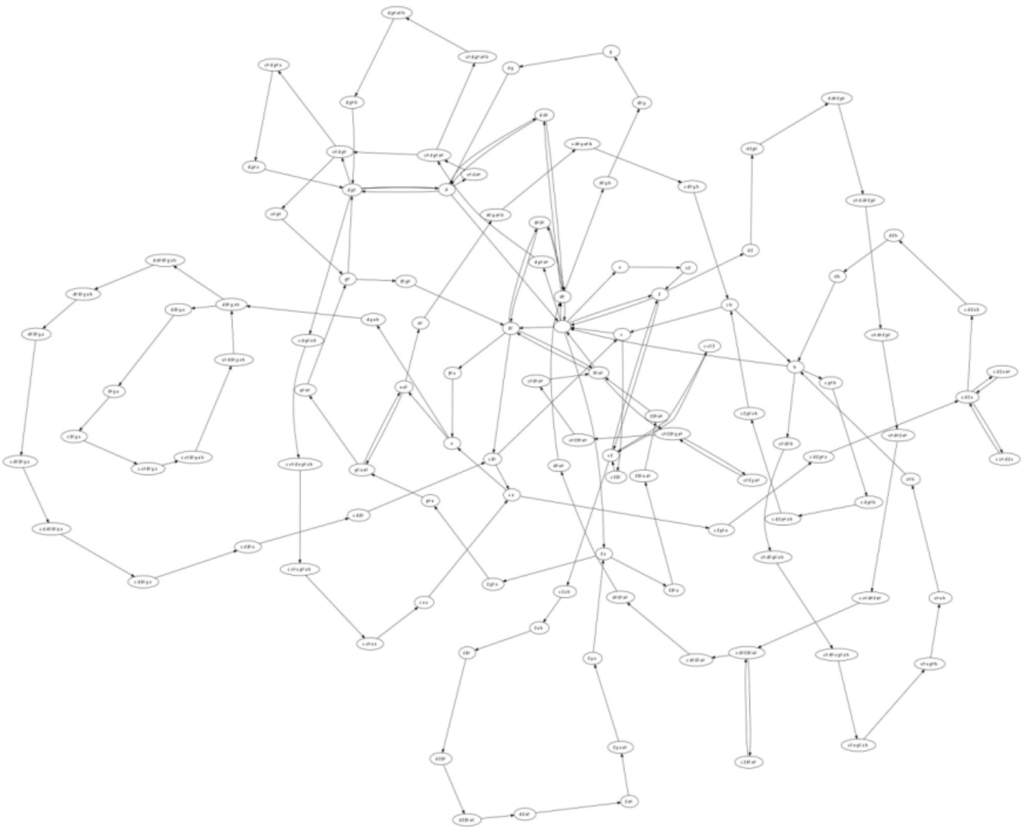

Figure 5. Visualized Harmony Graph of Schoenberg's Klavierstücke, Op. 19-5, an atonal piece

These phenomena reveal that Schoenberg's composition method deviates completely from the norms of traditional Harmony.

3.2 Social Network Analysis

In this section, we apply social network analysis techniques to examine the Harmony Graphs of the four corpora. Their degree distribution, average path length, and clustering coefficient are discussed in the following three subsections, respectively. Then, we introduce a newly proposed measure, *Agglomeration*, to describe the agglomeration phenomenon.

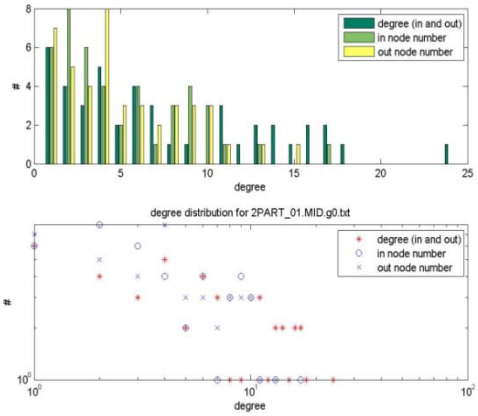

Figure 6. Degree distribution of two-part polyphonic piece

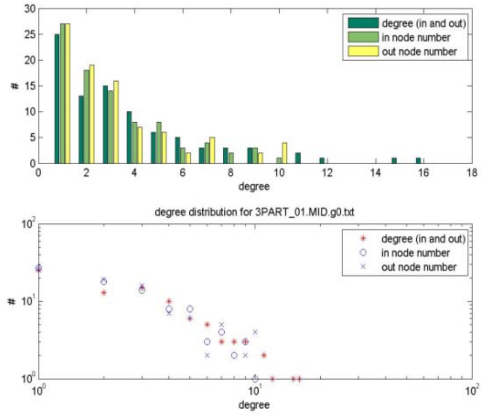

Figure 7. Degree distribution of three-part polyphonic piece

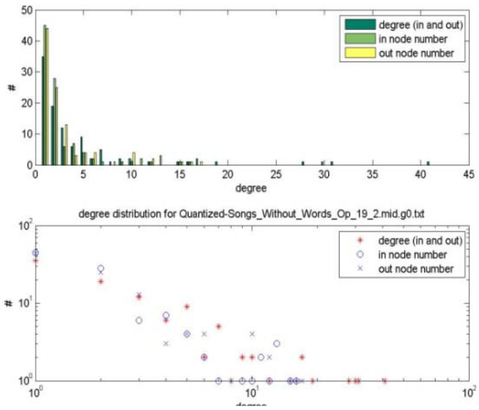

Figure 8. Degree distribution of homophonic piece

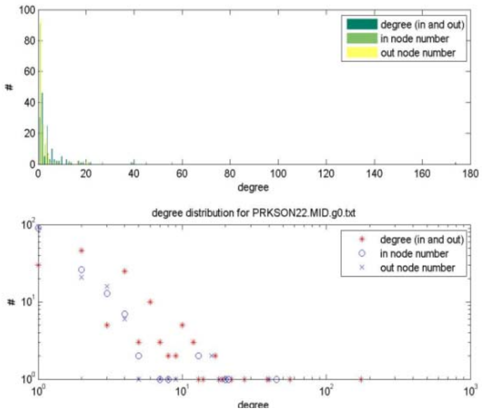

Figure 9. Degree distribution of pantonal piece

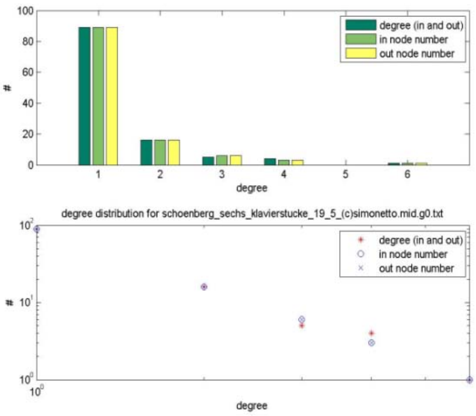

Figure 10. Degree distribution of atonal piece

3.2.1 Degree Distribution

Figure 6 to Figure 9 are the degree distributions of the four Harmony Graphs corresponding to Fig. 2 to Fig. 5, respectively. In each figure, the upper bar chart shows the degree distribution of weighted degree, unweighted incoming degree, and unweighted outgoing degrees in linear scale. As previously mentioned, the weighting is the count of the occurrences of edges. The lower scattered chart shows the same data in logarithmic scale to examine if it fits the Power Law. From those results, we see that all of them follow the Power Law except Fig. 6, Bach's two-part “Inventions”. At first glance, the reason might be that there are not enough nodes, since the Harmony Graphs of the three-part “Sinfonia”, which all follow the Power Law, have more nodes. Nevertheless, there are also very few nodes in Fig. 10, which still meets the Power Law. The same result applies in all of the other masterpieces of this genre.

Therefore, we speculate that the Harmony Graph follows the Power Law in normal circumstances, but for two-part polyphonic music such as “Invention”, the Power-Law effect is weaker due to strong tonality and node scarcity. This conjecture requires further in-depth investigation.

3.2.2 Average Path Length

In Section 3.1 we mentioned that there exist "long bridges" in the Harmony Graphs of 12-tone serial works. This can be best described in terms of the average path length (APL). Actually, in our experiments, we find that APL is the most significant characteristic to distinguish musical styles.

For two-part polyphonic music, APLs are normally under 3 due to fewer nodes and a high degree of agglomeration. As the music becomes more complex, for three-part polyphony and homophonic music, APL is slightly larger, between about 3 to 4. For non-tonal music, which was composed by numerous and various techniques, the corresponding APL has the most deviation, varying from 2 to 7. For Twelve-tone series works, all of the APLs are larger than 6.

3.2.3 Clustering Coefficient

In traditional social network analysis, the Clustering Coefficient (CC) is mostly relevant to characterize the aforementioned agglomeration phenomenon. For the most agglomerated Harmony Graph in our experiments, the two-part polyphonic music, the CC is about 0.3 to 0.4. For the other types of music, the CC is relatively smaller, about 0.001 to 0.1. Generally, CC alone is insufficient to distinguish the corpora, but when used in conjunction with other measures, the results are useful.

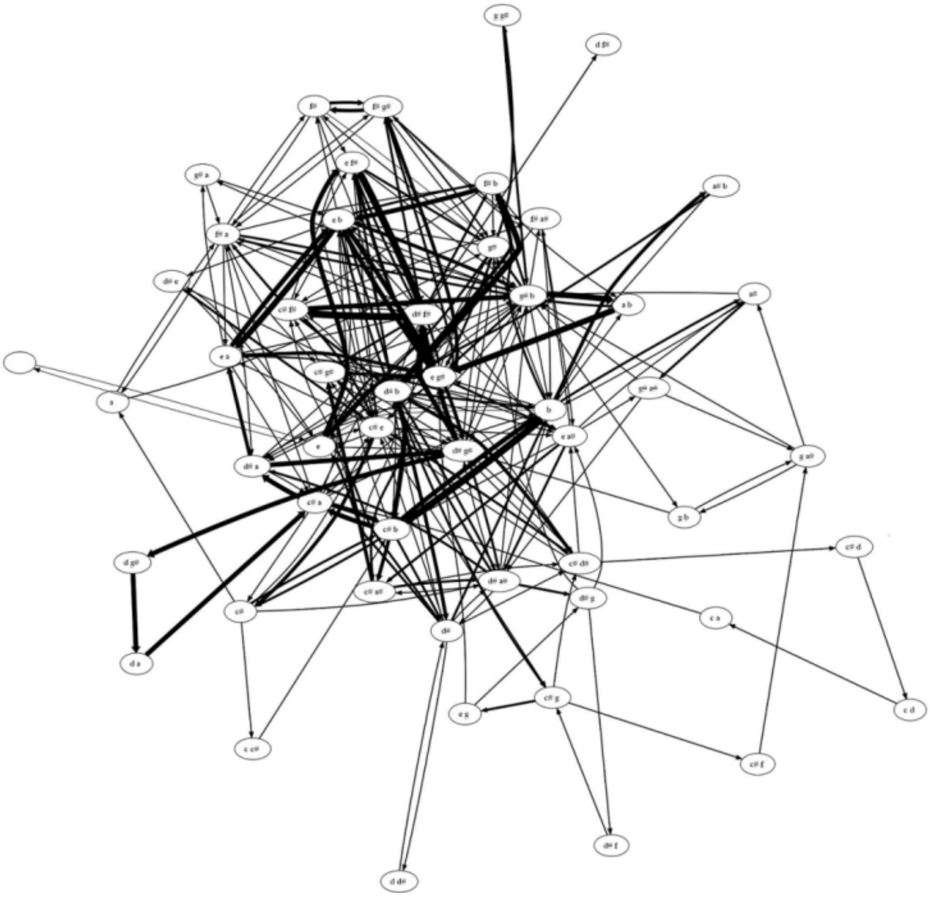

Figure 11. Visualization of Harmony Graph of Inventia 6.

In some cases, the CC does not confirm with agglomeration. For example, the Harmony Graph of “Invention” 6 in Fig. 11 has CC = 0.003, which is relatively small, but we can see that the nodes are strongly bonded with each other.

Since CC is calculated through the number of triangles, which is not necessarily related to bonding, a more reliable measure for explaining this phenomenon is needed.

3.2.4 Agglomeration

After studying Figure 2 to Figure 11, we found that the agglomeration phenomenon occurs when high-degree nodes link together, in contrast to the conditions for a large CC, which is due to large number of triangles formed by clusters of links in the graph. We thus propose an *Agglomeration* measure (*agg*):

$$agg = \frac{\sum_{i,j \in G} D(i)D(j)\delta_{ij}}{[\sum_{i \in G} D(i)]^2}, \tag{2}$$

where δ_{ij} denotes the adjacent status between nodes i and j. If nodes i and j are adjacent to each other, $\delta_{ij} = 1$, otherwise $\delta_{ij} = 0$. Notation $D(x)$ represents the degree of node x. By design, if high-degree nodes connect with each other, the corresponding *agg* value will be large.

Equation (2) can also be rewritten as:

$$agg = \sum_{i,j \in G} \frac{D(i)}{m} \frac{D(j)}{m} \delta_{ij}, \tag{3}$$

where m is the total degree of the graph. From (3), we can see that *agg* is also the probability showing the likelihood that two randomly-chosen nodes are adjacent. We can verify that, if the high-degree nodes are linked with each other, the probability that an adjacent node pair is selected is higher. Note that the range of this measure is from 1 for a complete graph down to 0 for a completely isolated graph.

In our experiments, we find *agg* is more suitable than CC to describe agglomeration. For instance, “Invention” 6, an especially agglomerative case, has an *agg* of 0.25, which is noticeably higher than the average *agg* of all “Inventions”. On the other hand, its CC is 0.003, which is far below the average CC of all “Inventions”.

Generally speaking, *agg* represents the degree of relation between harmonies. The *agg* of the Harmony Graphs we studied varies from 0.05 to 0.4. For a genre with strong harmony relations such as tonal music, *agg* tends to be large, and *vice-versa*. Nevertheless, we should not take *agg* as a measure of the degree of tonality, because non-tonal music might also have some strong harmony relations, such as modal music.

3.3 Corpus-Based Music Generation

3.3.1 By Random Walk

In a Harmony Graph each node represents a harmony; therefore, one directed edge binds two harmonies, and can be treated as a harmony progression. If we walk randomly in the Harmony Graph, the resultant harmony progression can produce music. We call this Graph Music.

For music generation, the build-up of the Harmony Graph is slightly extended. We not only need to save the count of occurrences of the harmony progressions as the weighting of edges, but also the durations. Thus, each edge is additionally tagged with a duration, such as a quarter note or a sixteenth, according to the learned data. Then, during the random walk, the random walker can pick among edges of different durations. So, the duration of the random chord progression is also randomly picked, and the produced music is rhythmic.

One feature of Graph Music is that it can reproduce similar music styles. We constructed a demonstration program that is harnessed with different Harmony Graphs built from masterpieces of Bach, Mendelssohn, Chopin, Brahms, Prokofiev, Shostakovich, and Schoenberg. If we switch among different composers, we can hear the style of the generated Graph Music changing accordingly, because the Harmony Graph has the effect of shuffling the corpus evenly, while reserving the most important information about the styles. Thus, the produced music sounds novel yet familiar.

3.3.2 By Link Prediction

The preliminary version of Graph Music has a drawback. During the random walk, if the degree of the current node is 1, there is only one choice for the next node. It is very likely that the next node also has degree of 1 if the portion of the corresponding original music in the corpus is quite unique, thereby trapping the random walker. The longer the path with such nodes, the more the produced music sounds like just a copy of the original music. It is analogous to the zero-occurrence problem in Corpus Linguistics. Here, we utilize the "link prediction" technique in social network for improvement.

Link prediction estimates the probability of connection for two unconnected nodes. When our random walker departs from one node, we make it choose some other unconnected nodes as extra candidates according to their link prediction probability. The estimated probability that two harmonies are linked is derived from their similarity. We believe similar harmonies have better continuity.

For two harmonies, A and B, we define the similarity as the number of their common notes divided by the number of notes in each harmony:

$$\mathrm{similarity}(\mathrm{A},\mathrm{B}) = \frac{|A \cap B|}{|A||B|}. \tag{4}$$

Note that the result will range between 0 and 1, inclusively. Then, we define the link prediction probability that an edge connecting from node S to node T exists in (5).

$$\mathrm{prob}\big((\mathrm{S},\mathrm{T}) \in \mathrm{E}\big) = \max\{\max_{(\mathrm{S},\mathrm{i})\in\mathrm{E}} \mathrm{similarity}(\mathrm{i},\mathrm{T}), \max_{(\mathrm{j},\mathrm{T})\in\mathrm{E}} \mathrm{similarity}(\mathrm{S},\mathrm{j})\}. \tag{5}$$

Here, E denotes the set of edges. By (5), we first find the outbound node of S with the highest similarity to T. We also find the inbound node of T with the highest similarity to S. Then, we pick the larger similarity value between the two as the link prediction probability. The logic is “Since S connects to a node similar to T, it is likely that S also connects to T.” or “Since a node similar to S connects to T, it is likely that S also connects to T”. Since we added soft links to harmonies with good continuity, the new music demonstrated more variety without abrupt changes.

4. Evaluation

4.1 Corpus Distinguishing

The qualitative discussion in Section 3.2 gives us some insight about different corpora. So, in this section, we use the four attributes discussed in Section 3.2 to perform supervised learning to verify how well we can differentiate between different corpora. The classifier we use here is SVM. The music entries and their corresponding categories are shown in Table 2. They were MIDI files mainly collected from the websites Classical MIDI Connection and kunstderfuge.com. Using the toolkit LIBSVM (Chang & Lin, 2001), with experimental settings cost equals 4, and gamma equals 1/70, the accuracy out of 59 entries in a 5-fold cross validation is 73%, which shows pretty good performance of this new model in classification.

Table 2. Five categories used in SVM test.

Genre	**Composer** and Works
2-part polyphony	**Bach** Inventions
3-part polyphony	**Bach** Sinfonias,
homophony	**Mendelssohn** Songs Without Words
pantonality	**Prokofiev** toccatas, preludes, sonatas
atonality	**Schoenberg** Klavierstücke

4.2 Corpus-Based Music Generation

The evaluation of the produced music to see if it follows a specific style is very subjective. Therefore, we provide a downloadable demonstration program for readers to rate it in person[1].

Note that users can also test on their own corpora by adding distinct folders of MIDI files. See the included instruction file for more details.

In addition, we conducted a subject test to show that the Graph Music somehow reflected the styles of the corpora. We set up a website to allow online testing and collected 245 responses from 21 participants. For each independent test, the participant would listen to a piece of music generated from one out of the five corpora of different composers, namely Bach, Mendelssohn, Brahms, Schoenberg, and Shostakovich. Then, original masterpieces of each composer were provided for comparison. The participants just listened to these six pieces of music, without any other information such as the name of the song or the composer. The

[1] URL for Graph Music program, http://homepage.ntu.edu.tw/~d96944001/GraphMusic

participant was then asked to choose one among the five original masterpieces such that the selected music is closest to the generated piece in style. After answering this question, the participant could decide to take one more independent test or just stop.

Our theory behind the experiment is as following. If the music was unrelated to the style, the participant could answer only by random guessing, hence, the accuracy should be about 20%. On the other hand, if the accuracy is greater than random guess, it indicates that there exists some recognizable relation behind the generated music and its corresponding corpora. To study the general case, we chose the participants from friends and classmates who have no advanced music background, *i.e.*, the participants were not familiar with those composers' works.

The collected responses are shown as the confusion matrix in Table 3.

Table 3. Confusion matrix of subject test.

	Answer					
Question	**Bach**	**Mendelssohn**	**Brahms**	**Schoenberg**	**Shostakovitch**	Accuracy
Bach	49	1	3	0	1	90.74%
Mendelssohn	10	36	2	2	2	69.23%
Brahms	4	3	42	2	0	82.35%
Schoenberg	1	1	0	44	2	91.67%
Shostakovitch	3	1	0	10	36	72.00%

In the matrix, the row represents the corpora the query music is generated from, and the column represents the answers from all participants. For example, the second row shows that among the 52 Mendelssohn questions, 10 were answered to be Bach, 36 were answered to be Mendelssohn (correct), and 2 for each of the other composers, which indicates an accuracy of 69.23%. So, we can assert that the generated music somehow reflects the styles of the corpora.

In statistical hypothesis testing, for all categories, the null hypothesis "the accuracy is 20% (due to random guessing)" was rejected and the alternative hypothesis "the accuracy is more than 20%" was accepted, with all confidence more than 99.9%, assuming that the accuracies were independent random variables following the student's *t*-distribution.

5. Conclusions

A social-network-like structure, Harmony Graph, for a music corpus, and with special emphasis on corpus distinction and music generation has been proposed. We prepared four music corpora of different genres, and derived Harmony Graphs for each corpus. The experiments show that the visualization of Harmony Graph is a good way to tell corpora apart. To be quantitative, we applied social network techniques to analyze Harmony Graphs. A new measure, *Agglomeration,* was also given to assess the strength of the relations between harmonies. To show the effect of corpus distinction in corpus-based music generation, we also provided a demo program for download. A subject test was also conducted in support of that the generated music somehow reflected the styles of the corpora.

Acknowledgments

We would like to thank Prof. Shou-De Lin for his detailed introduction to the research field in the Social Network Analysis course during 2009 fall. This research can not be done without Prof. Lin's advice. We also want to thank Prof. Shyh-Kang Jeng, director of NTU computer music lab. His subtle insight into computer music and valuable discussions has inspired us and besteaded a lot. In this research we utilized Stephen Toub's MIDI library for DotNet[2], which is an excellent tool for MIDI manipulation. Special thanks to him.

References

Conklin, D. (2003). Music generation from statistical models. In *Proceedings of the AISB 2003 Symposium on Artificial Intelligence and Creativity in the Arts and Sciences,* 30-35.

Dubnov, S., Assayag, G., Lartillot, O. & Bejerano, G. (2003). Using machine-learning methods for musical style modeling. *Computer,* 73-80.

Ellson, J., Gansner, E., Koutsofios, L., North, S. & Woodhull, G. (2002). Graphviz open source graph drawing tools. In *Graph Drawing*, 594-597.

Fortuna, B., Grobelnik, M. & Mladenic, D. (2005). Visualization of text document corpus. *Special Issue: Hot Topics in European Agent Research I Guest Editors: Andrea Omicini*, *29*, 497-502.

Iida, A., Campbell, N., Higuchi, F. & Yasumura, M. (2003). A corpus-based speech synthesis system with emotion. *Speech Communication*, 40(1-2), 161-187.

Manaris, B., Roos, P., Machado, P., Krehbiel, D., Pellicoro, L. & Romero, J. (2007). A corpus-based hybrid approach to music analysis and composition. In *PROCEEDINGS OF THE NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE*, 22, 839.

[2] http://blogs.msdn.com/toub
E-mail: stoub@microsoft.com

Marom, Y. & Zukerman, I. (2005). Corpus-based generation of easy help-desk responses.

Oerder, M. & Ney, H. (1993). Word graphs: An efficient interface between continuous-speech recognition and language understanding. In *icassp,* 119-122.

Pachet, F. (2003). The continuator: Musical interaction with style. *Journal of New Music Research*, 32(3), 333-341.

Paley, W. (2002). TextArc: Showing word frequency and distribution in text. In *Poster presented at IEEE Symposium on Information Visualization*, Volume 2002.

Polashek, T., Miranda, E., Hay, J., Bauer, L. & Gunkel, D. (2005). Beyond Babble: A Text-Generation Method and Computer Program for Composing Text, Music and Poetry. *Leonardo Music Journal*, 15(1), 17-22.

Rohrer, R., Ebert, D. & Sibert, J. (1998). The shape of shakespeare: Visualizing text using implicit surfaces. In *Proceedings of the 1998 IEEE Symposium on Information Visualization,* 121-129.

Shan, M.-K., Kuo, F.-F. & Chen, M.-F. (2002). Music style mining and classification by melody. In *Multimedia and Expo, 2002. ICME '02. Proceedings. 2002 IEEE International Conference on*, 1, 97-100.

Stribling, J., Krohn, M. & Aguayo, D. (2009). Scigen-an automatic cs paper generator. *Última visita*, 4, 12.

Trivino-Rodriguez, J. & Morales-Bueno, R. (2001). Using multiattribute prediction suffix graphs to predict and generate music. *Computer Music Journal*, 25(3), 62-79.