Music complexity measures predicting the listening experience

Søren Tjagvad Madsen

Austrian Research Institute for Artificial Intelligence Vienna, Austria <u>soren.madsen@ofai.at</u>

Gerhard Widmer

Department of Computational Perception Johannes Kepler University, Linz, Austria gerhard.widmer@jku.at

ABSTRACT

This paper examines the assumption that we continuously while listening tend to focus on the most complex (least repetitive) voice, experiencing this as foreground. We present a computational model calculating the level of attraction a voice in a score is likely to require at a given time. The model is based on a music information complexity measure. Calculating the complexity in each voice over a short time window, the model predicts the most complex voice to be the most interesting to listen to. The capability of the model is evaluated in terms of melody prediction. With promising results the predicted notes are compared to melody annotated scores. We discuss how to measure music complexity of pitch and rhythm, and examine which factors are the most important in the perception of music.

Keywords

Music complexity measures, melody note prediction.

INTRODUCTION

Music can be seen as a carrier of information. In this paper we present ideas for calculating the amount of information present in different parts in a composition at a given time. The information measures will be calculated from the structural core of music alone: the score – not considering timbre and performance aspects of the music.

Proceedings of the 9th International Conference on Music Perception & Cognition (ICMPC9). ©2006 The Society for Music Perception & Cognition (SMPC) and European Society for the Cognitive Sciences of Music (ESCOM). Copyright of the content of an individual paper is held by the primary (first-named) author of that paper. All rights reserved. No paper from this proceedings may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information retrieval systems, without permission in writing from the paper's primary author. No other part of this proceedings may be reproduced or transmitted in any form or by any information retrieval system, without permission in writing from the paper's primary author. No other part of this proceedings may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information retrieval system, with permission in writing from SMPC and ESCOM.

CALCULATING COMPLEXITY

The music information measures are intended to calculate whether a voice in a short period is a carrier of continuity or change. Continuity should lead to a low complexity rate and change will lead to a high complexity value.

A voice introducing new material will potentially become interesting to listen to. However if the new material is constantly repeated, we will pay less and less attention to it – become habituated or accustomed to the stimulus. Less attention is required from the listener and the voice will fall into the background (Snyder 2000, p. 208).

The model we present here will try to balance novelty and repetition by using entropy as information measure.

Sliding Window

The algorithm operates by in turn examining a small subset of the notes in the score. A fixed size (with respect to duration) window is slid from left to right over the score and the notes in the window are considered as present in that window. Different window sizes have been examined.

The window is advanced to where the next 'change' happens (so no windows will contain exactly the same set of notes). The next window will begin at the first occurring of the following two possibilities:

- 1. onset of next note after current window
- 2. right after next ending note in current window

In each window, we calculate a complexity value for each voice present. The most complex voice is expected to be the one preferred to listen to in this time period. The complexity measures and the melody note prediction method are explained below.

Entropy

Shannon's entropy (Shannon 1948) is a measure of randomness or uncertainty in a signal. If the predictability is high, the entropy is low, and if the predictability is low, the entropy is high. Let

$$X = \{x_1, x_2, ..., x_n\}$$

and

$$p(x) = Pr(X = x)$$

then the entropy H(X) is defined as:

$$H(X) = -\sum_{x \in X} p(x) \log_2 p(x)$$

X could for example be the set of MIDI pitch numbers and p(x) would then be the probability or frequency of a certain pitch. In the case that only one type of event (one pitch) is present in the window, that event is highly predictable or not surprising at all, and the entropy is 0 since the probability of this pitch p(x) is 1 (and $log_2(1)$ is zero). If however different events are present the entropy will grow. Entropy is maximised when the probability distribution is uniform – when no events are repeated.

1.1.1 Music features

We are going to calculate entropy of 'features' extracted from notes in monophonic melodies. We will use features related to pitch and duration of the notes. A lot of features are possible: MIDI pitch number, MIDI interval, pitch contour, pitch class, note duration, inter onset interval etc. We have used the following three measures in the model presented here:

- 1. Pitch class (C): count the occurrences of different pitch classes present
- 2. MIDI Interval (I): count the occurrences of each interval present
- 3. Note duration (D): count the number of note duration classes present (a duration is given it's own class if it is not within 10% of an existing class)

With each measure we extract events from a given sequence of notes, and calculate entropy from the frequencies of these events ($H_{\rm C}$, $H_{\rm I}$, $H_{\rm D}$).

 $H_{\rm C}$ and $H_{\rm I}$ are thought to capture opposite cases. $H_{\rm C}$ will result in high entropy when calculated on notes in a scale while $H_{\rm I}$ will result in low entropy. $H_{\rm I}$ will result in relatively high entropy on an arpeggio chord and $H_{\rm C}$ will produce a lower value.

So far rhythm and pitch are treated separately. We have also included a measure weighting the three defined measures above H_{CID} :

$$H_{CID} = \frac{1}{4}(H_{C} + H_{I}) + \frac{1}{2}H_{D}$$

Entropy is also defined for a pair of random variables with joint distribution:

$$H(X,Y) = -\sum_{x \in X} \sum_{y \in Y} p(x,y) \log_2[p(x,y)]$$

We will test two joint entropy measures: Pitch class in relation to duration ($H_{C,D}$) and interval in relation to duration ($H_{I,D}$). These are expected to be more specific discriminators.

The model is not using information related to any performance aspect of the score although an actual performance of the music piece might influence the listeners experience of the piece. What we try to measure is solely the information present in the voices in the score, although a performance of the piece might be useful in the melody prediction task used for evaluation.

Although the entropy measures in principle can be applied to any set of notes, we shall for practical reasons apply these measures only to monophonic sequences of music. First of all, the MIDI interval measure, measuring jumps between notes in a melody, is not well defined for chords. Furthermore, the sheer number of notes can make the entropy grow. This could reduce the entropy measure to become a measure of event density.

In the music used in our experiments, it does happen that a voice is playing more notes simultaneously (e.g. a violin). In these cases, only the top voice is used in our entropy calculation.

Complexity via compression

The entropy function is a purely statistical measure of occurrences of events. No relationships between events will be measured. For example the events abcabcabc and abcbcacab will result in the same entropy value. However if we were to remember the first string we would probably think of something like three occurrences of the substring abc – we infer a structure. The entropy function just counts the number of occurrences of each letter so no structure is taken into account. According to Snyder, we perceive music in the most structured way possible (Snyder 2002).

To account for this, complexity measures based on compression could be a possibility. Music that can be compressed a great deal (lossless) can then be considered less complex than music that cannot be compressed. Methods exist that substitute recurring patterns with a new event, and stores the description of that pattern only once, e.g. runlength encoding or LZW compression based on the ideas of (Lempel and Ziv 1977). However since these algorithms are not really effective for compressing such small things that we are dealing with (but rather digital images), we have not yet dug further into this field.

Predicting prominent notes

Given the set of windows described above and an entropy value for each voice present in each window we are now ready to predict the notes expected to belong to the most prominent voice. We consider in turn the notes in the interval between two consecutive windows. The notes belong to voices present in this period. We want to mark the notes in this period that come from the most complex voice. The complexity value of each voice from all windows the notes are present in, are averaged and the notes of the winning voice are marked as the most prominent in this period.

Thus the predicted notes may from period to period belong to different voices – the role as the most prominent voice may change quickly. We are not just predicting an entire voice to be the most interesting.

Similarly to calculating entropy of non-monophonic voices: when a predicted voice is non-monophonic (more notes onsetting at the same time in the same voice), only the top notes are marked.

We have developed a graphical interface allowing the user to listen to the music while watching the MIDI files in a piano roll layout. The predicted notes are emphasized, and the prominence curves of all voice can be observed.

EXPERIMENTS

To perform experiments we need music, which is composed for different parts, and encoded in such a way that each part is separately available. Furthermore, since our complexity measure assumes monophonic music, each voice in the piece should be close to monophonic. This lays some restrictions on the experiments we are able to do. A typical piano sonata, lacking the explicit voice annotation (or being one non-monophonic part), will not be an appropriate choice.

Since the model presented here is not taking any performance aspect of the music into account but only the musical surface (score), we will use MIDI files generated from the MuseData format (<u>http://www.musedata.org</u>). The duration of the notes in these files are nicely quantised. The model will however run on all types of MIDI files, but performed music tends to bias the rhythm complexity.

Two pieces of music were chosen to be annotated and used in the experiment:

- 1. Haydn, F.J.: String quartet op. 54, No. 2, 1st movement
- 2. Mozart, W.A.: Symphony No 40 in G minor (KV 550), 1st movement

This is not an awful lot of data, but since the music will have to be annotated manually, this will have to do for our initial experiments.

Annotating melody notes

We will test our model by means of melody prediction. The assumption is that the most prominent voice and the melody will often coincide. We believe that this is the best and maybe the only way to test our model.

Asking a listener to point out the voice that he right away would be following while listening to a piece of music, would result in a very subjective judgment, since the listener is able to choose what to listen to. Listening to a piece more than once is also apt to affect your listening experience. In stead we asked our annotator to mark what she considers as being the melody. Due to the conventional listening perception, people mostly seem to agree about the melody of a piece much more that they agree on what they are actually focusing on while listening. Focus is often intuitive, and it is therefore difficult for people to declare. The melody is then the criteria that we have used for testing.

In pieces where we assume that the melody is the most complex voice present, we expect our model will perform well. Such pieces can be used for evaluation. But in a fugue, where the voices are approximately equally complex, the evaluation method will fail. Assuming that the model works well, what we really are measuring is the degree of the melody being the most complex voice. In the experiments section we shall see, that the melody is indeed complex in the pieces we have chosen for evaluation.

The task of defining the concept of melody was given to a musicologist and made explicit through the annotation of notes in the test pieces. The annotator was given a piano roll representation of the music with 'clickable' notes and simple playback options for listening to the MIDI sound. The annotator, who was not aware about the purpose of this task, was instructed to mark notes she would consider as 'melody'.

The test pieces turned out to have quite distinguishable melodic lines. The annotator solved the task by marking the notes that she was humming during listening to the pieces. Some immediate differences between melody and our complexity based melody prediction model were made clear after talking to the annotator:

Our model assumes that there constantly is one most complex voice present, but a melody was not found to be present at all times. Furthermore, melodic lines sometimes have been marked as overlapping, which the model doesn't allow.

The annotator chose not to mark thematic lines that clearly were a (melodic) response in another voice to the melody rather than a continuation of the melody line. Our model might mark both the melody as well as the response.

Another source of error is the situations where the melodic line consists of long sustained tones while the accompanying notes are doing all the action. The model will erroneously predict an accompanying voice. (In these situations it is also difficult to tell what a listener is actually listening to).

Despite of the differences between melody and most complex voice, the annotated notes were stored as ground truth for the evaluation. Table 1 shows some information about the test data.

	Haydn	Mozart
Number of melody notes	2555	2295
Total number of notes	5832	13428
Melody note percentage	43.8 %	17.1 %
Number of voices	4	10
Duration	11.30 min	7 min

Table 1. The test data

Evaluation method

We can now measure how well the predicted notes correspond to the annotated melody in the score. We express this in terms of recall (*R*) and precision (*P*) values (van Rijsbergen 1979). Recall is the number of correctly predicted notes (true positives, TP) divided by the total number of notes in the melody. Precision is TP divided by the total number of notes predicted (TP + FP (false positives)). The F-measure combines recall and precision into one value (an a of 0.5 used throughout this paper giving equal weight to precision and recall):

$$F(R,P) = 1 - \frac{1}{a\frac{1}{P} + (1-a)\frac{1}{R}}, 0 \le a \le 1$$

A high rate of correct predicted notes will result in high values of recall, precision and F-measure (close to 1.0).

Results

We performed prediction experiments with four different window sizes (1-4 seconds) and with the five different entropy measures described above. Table 2 shows recall, precision and F-measure values from all experiments with the two evaluation pieces. For each experiment, the highest Fmeasure value has been emphasized.

The string quartet turned out to be the less complicated of the two pieces. This is not a surprise - it is easier to discriminate between 4 than 10 voices.

Overall the joint pitch class and duration measure (H_{CD}) was found to have the greatest predictive power. Pitch class seems to be the single most important measure in the string quartet. In total, the joint measures perform better than the measures based on a single feature.

We can conclude that there do exist a correlation between melody and complexity in both pieces. The precision value of 0.602 in the best symphony experiment with a resulting F-measure of 0.514 (window size of 3 seconds) tells us that 60.2 % of the predicted notes in the symphony were also annotated as melody notes.

Table 2. Recall, precision, and F-measure

		Haydn						Mozart					
		H _C	$H_{\rm I}$	$H_{\rm D}$	$H_{\rm CID}$	$H_{\rm C,D}$	$H_{\rm I,D}$	H _C	$H_{\rm I}$	$H_{\rm D}$	$H_{\rm CID}$	$H_{\rm C,D}$	$H_{\mathrm{I,D}}$
1000 ms	R	.83	.80	.68	.85	.87	.80	.36	.31	.32	.43	.47	.32
	Р	.78	.73	.80	.82	.81	.74	.41	.31	.52	.53	.54	.32
	F	.81	.76	.73	.83	.84	.77	.38	.31	.40	.47	.51	.32
2000 ms	R	.83	.80	.64	.82	.87	.81	.35	.33	.27	.40	.45	.37
	Ρ	.81	.75	.81	.83	.87	.76	.42	.36	.52	.55	.58	.41
	F	.82	.77	.71	.82	.87	.78	.38	.35	.36	.46	.51	.39
3000 ms	R	.83	.79	.63	.83	.84	.80	.33	.28	.25	.37	.45	.36
	Р	.83	.75	.81	.86	.87	.78	.40	.31	.50	.53	.60	.42
	F	.83	.77	.71	.85	.85	.79	.36	.29	.33	.43	.51	.38
4000 ms	R	.84	.76	.59	.78	.80	.79	.34	.24	.19	.37	.44	.33
	Ρ	.84	.74	.78	.85	.86	.78	.40	.28	.41	.59	.61	.41
	F	.84	.75	.67	.82	.83	.79	.37	.26	.26	.45	.51	.36

In the string quartet, after 287 seconds one voice is alternating between a single note and notes from a descending scale, making the voice very attractive (lots of different notes and intervals) while the real melody above is playing less different notes, but has a more varied rhythm. We took a closer look at this passage. Setting the window size to 2 seconds, the measures H_D , H_{CID} , and $H_{C,D}$ solves the problem successfully whereas H_C , H_I , and $H_{I,D}$ does not. The measures based on intervals are naturally mistaking in this case, and the measure based solely on pitch class is also. Again, the joint measure $H_{C,D}$ proves to be robust.

CONCLUSION

Computational measures of music complexity were presented. The evaluation of the model showed promising results, although the amount of test data was small. We intend to continue our work by refining the complexity measures. New complexity measures based on compression techniques are also to be examined.

ACKNOWLEDGMENTS

This research was supported by the Viennese Science and Technology Fund (WWTF, project CI010). The Austrian Research Institute for AI acknowledges basic financial support from the Austrian Federal Ministries of Education, Science and Culture and of Transport, Innovation and Technology.

REFERENCES

Snyder, B. (2000). *Music and Memory: An Introduction*. MIT Press.

van Rijsbergen, C. J. (1979). *Information Retrieval*. London, Butterworth.

Shannon C. E., (1948). A mathematical Theory of Communication. *The Bell System Technical Journal, Vol.* 27, 379-423, 623-656.

Jacob Ziv and Abraham Lempel. A Universal Algorithm for Sequential Data Compression, *IEEE Transactions on Information Theory, Vol IT-23*, No 3, May 1977, 337-343.