Automatic Evaluation Methods in Evolutionary Music: An Example with Bossa Melodies

Freitas, A. R. R.[1], Guimarães, F. G.[2], and Barbosa, R. V.[3]

[1] Graduate Program in Electrical Engineering, Federal University of Minas Gerais
Av. Antônio Carlos 6627, 31270-901, Belo Horizonte, MG, Brazil
alandefreitas@gmail.com
[2] Departamento de Engenharia Elétrica, Universidade Federal de Minas Gerais
Belo Horizonte, MG 31270-901, Brazil
fredericoguimaraes@ufmg.br
[3] Departamento de Teoria Geral da Música, Universidade Federal de Minas Gerais
Belo Horizonte, MG 31270-010, Brazil
rvb@musica.ufmg.br

Abstract. Evolutionary algorithms need measures of how appropriate a solution is in order to make decisions. This is always a problem for evolving art as codifying aesthetics is a complex task. In this paper we consider the problem of evaluating melodies. The evaluation of melodies in evolutionary music is an open problem that has been tackled by many authors with interactive evaluation, fitness-free genetic algorithms and even neural networks. However, all approaches based on formal analysis of databases or formal music theory have been partial, which is something to be expected for such a complex problem. Thus, we present many metrics that can be used for evaluating melodies and their practical results when applied to a Bossa Nova database of melodies coded by the authors. Although the paper is meant to extend the cycle of possible ideas for evolutionary composers, we argue that there is still much to be developed in this field and each genre of music will always need specific measures of quality.

Keywords: Evolutionary Music, Genetic Algorithms, Evaluation of Melodies

1 Introduction

Evolutionary Algorithms and Algorithmic Composition methods need to measure how appropriate a solution is in order to make decisions. Easy ways to evaluate melodies would be comparing tunes, using only music theory or having a mentor to guide the process.

Evaluating music and art faces many challenges that we discuss in Section 2. Given the open problems for music evaluation and the methods recently proposed, we focus this paper on the definition of metrics more formally based on music theory or data extraction, as we develop the idea in Section 3.

In this context, we describe a list of metrics divided in many categories from Section 4. In parallel to those metrics from a musicology research, we also show the results of those metrics on a database of Bossa Nova melodies the authors have created. We discuss each of those metrics as Information Retrieval or Computational Musicology processes.

In our discussion of the results, in Section 5, we argue that this work should be useful for scientists intending to create algorithms for generating melodies but there will always be metrics which will be more useful for genre-specific music generation.

2 Forms of Music Evaluation

Codifying aesthetics is a complex task and the biggest problem in evolutionary composition [1]. Approaches to circumvent codifying aesthetics such as interactive evolution [2–4], fitness-free GAs [5] and neural networks [6, 7], all still present many drawbacks.

Using a human mentor usually leads to fitness bottlenecks [3, 4]. Fitness-free algorithms [5, 8] are bolder proposals but they also avoid studying the problem and oblige the genetic operators to be conservative. Most works based on Neural Networks do not have the ability to generalize beyond training sets [6, 8, 7].

Thus, an open problem is to create automatic evaluation functions [1] machine representable, capable of measuring human aesthetic properties and practically computable. They should not only define what is more likely to occur on melodies but they should also allow creativity when considering all the different aesthetic objectives to generate ideas not imagined before [9]. Computational aesthetic evaluation is a distinctly non-trivial unsolved problem [1].

3 Automatic Objective Functions

Many different metrics based on perceptions of the composer or music theory can be employed to analyze melodies in a process of algorithmic composition [10]. In this paper we present many automatic metrics and their results on a database of Bossa Nova melodies manually created by the authors.

There have been partial attempts to automate measures of fitness [1, 11] and studies on which features are most important [11]. Those include four part harmonization [12] and jazz melodies [13], for instance. The influence of the genetic operators on musical features has also been partially studied [14, 5]. Target values have also been used to measure fitness [15–17].

From an analysis over the literature, most algorithms do not examine the possible relation between all categories of metrics possible [10]. Thus, we define metrics that should be applicable to most classical, baroque or popular twentieth-century melodies and the results of their employment on a database of melodies.

With many analyses of those melodies from different points of view, we can compare the results to a potential solution from our generative algorithm. Some results indicate parameters with normal distribution, such as in the distribution of pitches, which can be tested in the candidate solution with a Jarque-Bera test

[18]. Some results may show that parameters come from another distribution, such as in the distribution of rhythmic proportion, which can be compared to a candidate solution with a Two-sample Kolmogorov-Smirnov test [19]. Some parameters may only represent categorical values, which can be compared with a nominal statistical test [20]. Finally, other results indicate potential individual target values for the melodies, such as the tempo of each melody, which can be directly included in objective function values as the distance from the target.

4 Metrics and Results

In order to give a good representation of the mentioned metrics, we have created a database with 26 Bossa Nova melodies from Tom Jobim's songbook [21] and manually coded by the authors. All the data is available from the authors[4].

4.1 Tonality, Pitches, and Intervals

We first detect the key of each melody with the K-S key-finding algorithm [22], based on key profiles. As the melodies may even have key changes, it may be a simplistic approach, but all the keys detected matched the key signature in the scores and the results can give us an idea of the keys as we can see in Table 1. Thus, we transpose all songs to C (or its minor relative, Am) to make key dependent analyzes possible, such as the detection of dissonances. The correlation of the algorithm's key profiles to the pitch distribution of the pieces leads to a representation of the strength of each key, as in Figure 1(a). The correlation values are significantly higher for the C major and A minor key profiles, indicating some relevance of the method applied. The results can also be projected on a self-organizing map trained with key profiles [23], as in Figure 1(b).

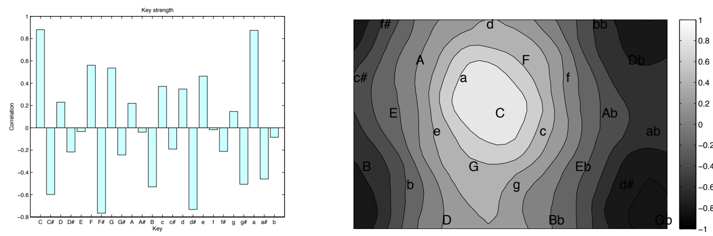
Table 1. Key Profiles

C	C#	D	D#	E	F	F#	G	G#	A	A#	B	**Total**
12%	0%	15%	0%	4%	4%	0%	4%	0%	4%	0%	4%	**46%**
c	c#	d	d#	e	f	f#	g	g#	a	a#	b	**Total**
0%	0%	15%	0%	12%	4%	12%	4%	4%	4%	0%	0%	**54%**

The pitches used in all melodies are in Figure 2(a), showing that the distribution of the notes is very normal. However, by shifting all melodies to the same key, we have a large difference of occurrence between consecutive notes, as in Figure 2(b). This is due to dissonant notes, which are strange to the main scale.

Given the 12 note classes, the modulo of a pitch number by 12 is the class of this note. The occurrence of those pitch classes gives a better idea of the scales used in the melody, as in Figure 2(c). We can see a higher occurrence of notes of the diatonic scale of C. The note variety is also different for each melody. A method of measuring pitch variety [11] is by dividing the number of

[4] `http://www.alandefreitas.com/downloads/problem-instances.php`



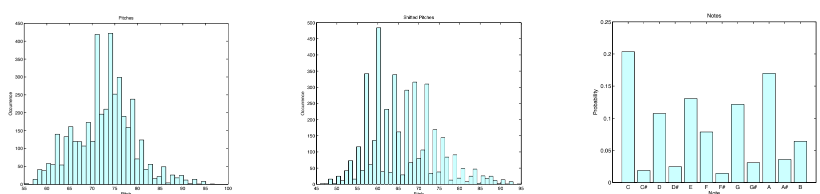
(a) Correlation (b) Results projected on a SOM

Fig. 1. Melody keys

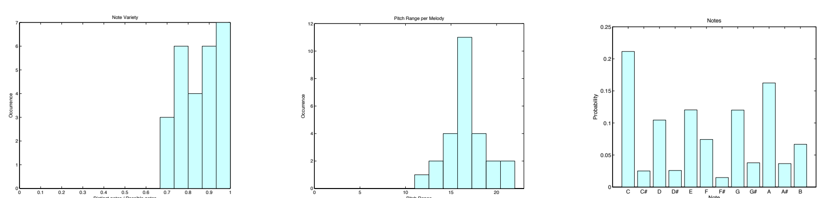
distinct notes in a melody by 12, as in Figure 2(d), which shows that the variety of pitches is very different among the melodies but all melodies use more than 70% of the possible notes. Another aspect of pitch variety is pitch range [11], or the difference between the highest and lowest pitches. As we can see in Figure 2(e), the pitch range has a more normal distribution, centered on a range of 16 pitches.

Other useful metrics are the beginning and ending pitches, and the note distribution weighted by duration. For our database, this measure did not represent much difference, as we can see in Figure 2(f).

Perhaps, more important than the pitches themselves are the intervals between them. Figure 3(a) shows the intervals present in our melodies. In accordance with theoretical models [24], intervals of small size are more common than large ones. Figure gives a good representation of the interval sizes used in the

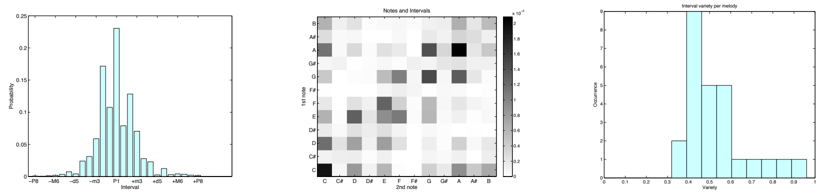


(a) Pitches (b) Shifted Pitches (c) Notes



(d) Note Variety (e) Pitch Range (f) Weighted Notes

Fig. 2. Pitches



(a) Intervals (b) Notes & Intervals (c) Interval Variety

Fig. 3. Intervals

melodies. In fact, it is a common practice to penalize very large intervals in the evaluation of the solutions [13]. However, this can be only applicable to some genres of music and an approach based on a better analysis is recommended. If we combine the information of the notes to the intervals, we are going to find out that the probability of the next note depends on the current note, as shown in Figure 3(b). Also similarly to the notes, we can analyze the interval variety for each song, as in Figure 3(c).

Contour refers to the movements being performed by the melodies. There are many sorts of contour [25] and the direction of those movements may be easier to remember than the movements themselves [26]. An easy way to analyze contour is to measure how many intervals are ascenders or descenders, and the stability in relation to direction. Table 2 shows the values of ascenders and descenders in general or in relation to the last interval. The values in bold represent the contour stability, which is a criterion that has also been used in evolutionary algorithms [11], and represented for each song in Figure 4(a). Another simple form of controlling contour is through the average contour direction [13, 11].

Table 2. Contour

	Ascendent	Unison	Descendent
After an ascendent	**30.17%**	18.78%	51.03%
After an unison	19.92%	**55.24%**	24.82%
After a descendent	45.36%	14.57%	**40.06%**
In general	33.52%	26.95%	39.52%

By analyzing pitches and tonality together, we can have an idea of the dissonances used in the songs. From Figure 2(c) we can see that it would be more than reasonable to analyze dissonance in terms of the proportion of notes that do not belong to the diatonic scale. Thus, the probability of a dissonant note is 30.53%, but Figure 4(b), which represents the occurrence of dissonance divided by the number of possible dissonant notes, shows how this value can vary considerably.

Attraction of dissonant notes to tonally stable notes happens to 55.25% of the dissonances. However, this measure may overlap with the measure of second order notes, as shown in Figure 3(b).

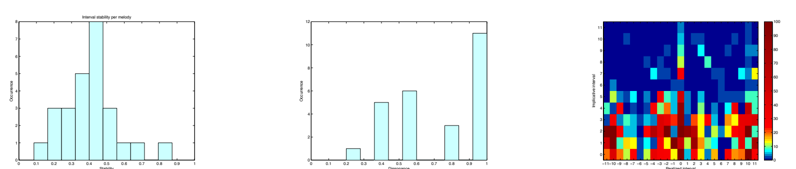
Narmour's Implication-Realization Model [27] is a study on melodic expectancy based on many principles that consider expectation of the listener after a given interval. With a quantification of the principles in model [28, 29], we can either penalize melodies that disrespect the principles or measure how much the melodies follow the model.

As the model can be context-specific or inefficient to consider tonally stable intervals [30], we can also use the interval values in the melodies to infer our own model of expectancy which would be specific for our goal. Figure 4(c) shows such a model, where the rows represent implicative intervals and columns represent realized intervals. The model confirms the expectation of small intervals. Melodic attraction should also be considered by this model of expectation as we have different responses for different pitches [31]. One way of doing that would be to infer 12 different models according to the current note.

4.2 Rhythm, Patterns, and Phrases

The first feature that determines the rhythm is the duration of notes. Figure 5(a) shows a second order analysis of the proportion of note durations. From 36 possible values of duration present in the melodies, the histogram is based on the durations shorter than 4 beats and longer than 1/4 of a beat [29]. The patterns show a tendency of repetition in the duration of following notes. Another interesting pattern is that the first notes in a melody, shown in Figure 5(b), tend to have shorter duration than the last notes, shown in Figure 5(c). The rhythmic proportion in each melody is the duration of the longest note divided by the duration of the shortest note, as shown in Figure 5(d). Similarly to what we did to the pitches, we can also calculate the duration variety in the melodies, as in Figure 5(e). Part of the rhythmic analysis is not only the duration of the pitches but also how much silence we have in the melodies. In Figure 5(f) we have the amount of silence (as at most 2 beats without notes) per melody. In some cases, even more than 10% of the melody may be silent.

We have mentioned the duration of the notes but another important information is when the notes are played. A hierarchical grid of note locations may exist in the expectation of Western melodies [32]. For instance, the note positions in the musical measure are represented in Figure 5(g). Also the first notes (and last notes) may use different positions, as the example in Figure 5(h). In fact, only 6 values of note position are used for the first notes while 16 values are used for



(a) Contour Stability (b) Dissonance (c) Melodic Expectation

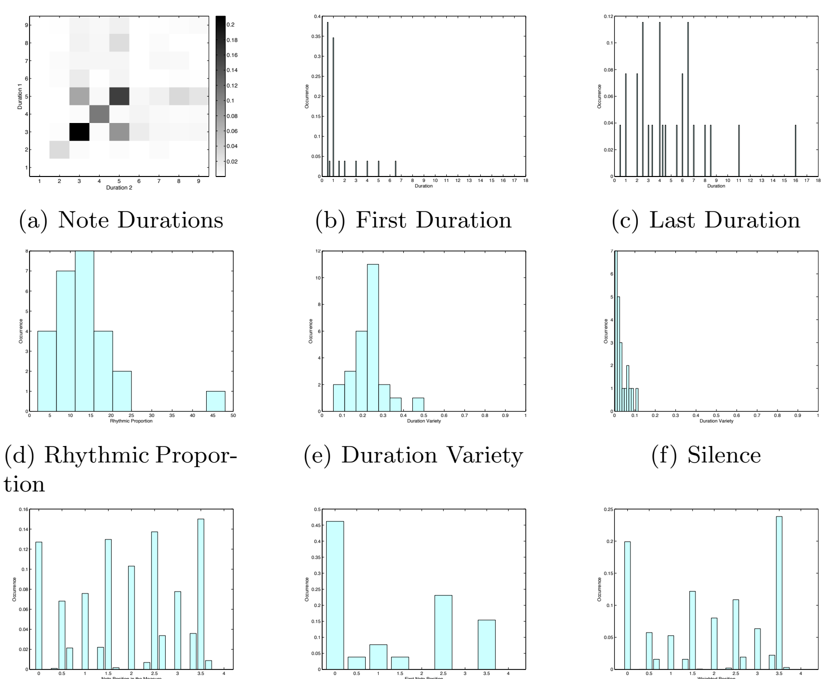
Fig. 4. Expectation and Contour

all notes. The note positions can also be weighted by the duration of those notes as it alters how listeners perceive those notes [33]. Figure 5(i) shows the relation between those two components.

Once we have information related to pitches and rhythm we can find patterns in the melodies. By autocorrelating a melody with a delayed copy of itself [34], we can identify patterns in a melody. The correlation values go from 0 to 1, and the correlation value is always 1 at point 0, when we compare a melody shape with a copy of itself, as shown in Figure 6(a). The three areas represent the maximum, mean, and minimum correlation. Similarly to the contour shape, we can apply the same technique to only pitches or duration values.

Another way of looking at the patterns is to identify the number of patterns of a specific size in a melody. We can analyze that in Figure 6(b), where each row represents a melody, each column represents a pattern size and the colors represent the amount of that pattern. Short patterns are naturally more common as longer and rare patterns may represent the repetition of phases in the melody. The same metric can be applied to notes or duration values.

We can divide melodies into musical phrases. Figure 6(c) shows the number of phrases per melody according to a rule-based approach [35]. There are also approaches based on probability [36]. The size of those phrases can also be



(a) Note Durations (b) First Duration (c) Last Duration

(d) Rhythmic Proportion (e) Duration Variety (f) Silence

(g) Note Position (h) First Position (i) Weighted Position

Fig. 5. Rhythm

analyzed and with those values it is possible to also study the value of the parameters for each musical phrase as well as the relation between neighbor phrases in relation to pitch and rhythm.

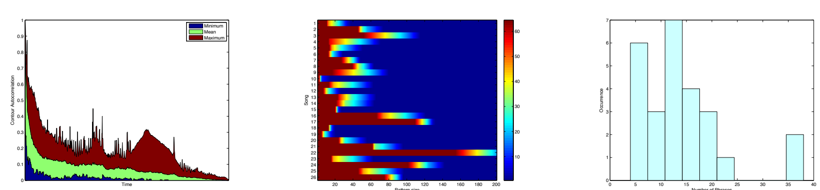
5 Discussion and Future Work

All the metrics presented here can lead to different models according to the specific genres of music. Those models, in some cases, can even lead to problems which are simple to solve in polynomial time. In that case, evolutionary computation could be even unsuitable for composing. On the other hand, with all the information to be considered when generating compositions, it is unlikely to exist a good model of composition which is too trivial.

Although there are many other metrics that could be considered in the evaluation of melodies, such as contour shapes or rhythmic variation, the authors do not have the pretension to formulate all of them as it would not be feasible. However, by studying at least some of the most important metrics in relation to each category of analysis, this paper can surely give some background to scientists with intention to be evolutionary composers. Natural extensions of the ideas presented here would be to apply all the metrics on melody phrases separately and to filtrate which metrics are most important. It would be also important to perform second-order analysis on the melodies to look for potential relations between the metrics.

Once we are able to generate melodies that follow patterns of a studied database, another issue is also the diversity and originality of the solutions generated by the algorithm, as we do not want the algorithm to either return always the same "best" melody [5] or to ignore the originality needed in masterpieces [37]. Once we have considered those issues, we can focus on applying the statistical methods mentioned in Section 3 to get more formal objective values.

In regard to evolutionary computation, an important issue in the future will also be how to put all those metrics together into one or many objective functions and which genetic operators will be appropriate for those functions. So far, the formalized evaluation metrics for evolutionary music have only been partial and this paper should expand the ideas considered by evolutionary composers on their work. However, as music is a very contextual form art, specific metrics will always need to be created for specific genres of music.



(a) Autocorrelation (b) Patterns (c) Phrases

Fig. 6. Patterns

Acknowledgements

This work has been supported by the Brazilian agencies CAPES, CNPq, and FAPEMIG and the Marie Curie International Research Staff Exchange Scheme Fellowship within the 7th European Community Framework Programme.

References

1. Galanter, P.: The problem with evolutionary art is... Applications of Evolutionary Computation (2010) 321–330
2. McCormack, J.: Open problems in evolutionary music and art. Applications of Evolutionary Computing (2005) 428–436
3. Todd, P., Werner, G.: Frankensteinian methods for evolutionary music. Musical networks: parallel distributed perception and performace (1999) 313
4. Biles, J.: Genjam: A genetic algorithm for generating jazz solos. In: Proceedings of the International Computer Music Conference, International Computer Music Association (1994) 131–131
5. Freitas, A., Guimarães, F.: Originality and diversity in the artificial evolution of melodies. In: Proceedings of the 13th annual conference on Genetic and evolutionary computation, ACM (2011) 419–426
6. Biles, J., Anderson, P., Loggi, L.: Neural network fitness functions for a musical iga. In: International ICSC Symposium on Intelligent Industrial Automation (IIA'96) and Soft Computing (SOCO'96), International Computing Sciences Conferences (ICSC) (1996) B39–44
7. Phon-Amnuaisuk, S., Law, E., Kuan, H.: Evolving music generation with som-fitness genetic programming. Applications of Evolutionary Computing (2007) 557–566
8. Biles, J.: Evolutionary computation for musical tasks. Evolutionary Computer Music (2) (2007) 28–51
9. Freitas, A., Guimaraes, F.: Melody harmonization in evolutionary music using multiobjective genetic algorithms. In: Proceedings of the Sound and Music Computing Conference (SMC). (2011)
10. Freitas, A., Guimaraes, F., Barbosa, R.: Ideas in automatic evaluation methods for melodies in algorithmic composition. In: Proceedings of the Sound and Music Computing Conference (SMC). (2012)
11. Towsey, M., Brown, A., Wright, S., Diederich, J.: Towards melodic extension using genetic algorithms. Educational Technology & Society **4**(2) (2001) 54–65
12. McIntyre, R.: Bach in a box: The evolution of four part baroque harmony using the genetic algorithm. In: Evolutionary Computation, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the First IEEE Conference on, IEEE (1994) 852–857
13. Papadopoulos, G., Wiggins, G.: A genetic algorithm for the generation of jazz melodies. Proceedings of STeP **98** (1998)
14. Phon-Amnuaisuk, S., Wiggins, G.: The four-part harmonisation problem: a comparison between genetic algorithms and a rule-based system. In: Proceedings of the AISB99 Symposium on Musical Creativity. (1999) 28–34
15. McDermott, J., Griffith, N., O'Neill, M.: Toward user-directed evolution of sound synthesis parameters. Applications of Evolutionary Computing (2005) 517–526

16. Khalifa, Y., Foster, R.: A two-stage autonomous evolutionary music composer. Applications of Evolutionary Computing (2006) 717–721
17. Horner, A., Goldberg, D.: Genetic algorithms and computer-assisted music composition. Urbana **51**(61801) (1991) 14
18. Jarque, C., Bera, A.: A test for normality of observations and regression residuals. International Statistical Review/Revue Internationale de Statistique (1987) 163–172
19. Massey Jr, F.: The kolmogorov-smirnov test for goodness of fit. Journal of the American statistical Association (1951) 68–78
20. Bashkansky, E., Gadrich, T., Kuselman, I.: Interlaboratory comparison of test results of an ordinal or nominal binary property: analysis of variation. Accreditation and Quality Assurance: Journal for Quality, Comparability and Reliability in Chemical Measurement 1–5
21. Jobim, A., Chediak, A.: Songbook Tom Jobim. Volume 1. Irmãos Vitale (1990)
22. Krumhansl, C.: Cognitive foundations of musical pitch. Number 17. Oxford University Press, USA (2001)
23. Toiviainen, P., Krumhansl, C., et al.: Measuring and modeling real-time responses to music: The dynamics of tonality induction. PERCEPTION-LONDON- **32**(6) (2003) 741–766
24. Dowling, W., Harwood, D.: Music cognition. Volume 19986. Academic Press New York (1986)
25. Nettl, B.: Music in primitive culture. Harvard University Press (1956)
26. Dowling, W.: Scale and contour: Two components of a theory of memory for melodies. Psychological Review **85**(4) (1978) 341
27. Narmour, E.: The analysis and cognition of melodic complexity: The implication-realization model. University of Chicago Press (1992)
28. Krumhansl, C.: Music psychology and music theory: Problems and prospects. Music Theory Spectrum (1995) 53–80
29. Eerola, T., Toiviainen, P.: MIDI Toolbox: MATLAB Tools for Music Research. University of Jyväskylä, Jyväskylä, Finland (2004)
30. Krumhansl, C., Kessler, E.: Tracing the dynamic changes in perceived tonal organization in a spatial representation of musical keys. Psychological Review **89**(4) (1982) 334
31. Lerdahl, F.: Calculating tonal tension. Music Perception (1996) 319–363
32. Palmer, C., Krumhansl, C.: Mental representations for musical meter. Journal of Experimental Psychology: Human Perception and Performance; Journal of Experimental Psychology: Human Perception and Performance **16**(4) (1990) 728
33. Thompson, W.: Sensitivity to combinations of musical parameters: Pitch with duration, and pitch pattern with durational pattern. Attention, Perception, & Psychophysics **56**(3) (1994) 363–374
34. Eerola, T., Himberg, T., Toiviainen, P., Louhivuori, J.: Perceived complexity of western and african folk melodies by western and african listeners. Psychology of Music **34**(3) (2006) 337–371
35. Tenney, J., Polansky, L.: Temporal gestalt perception in music. Journal of Music Theory **24**(2) (1980) 205–241
36. Cambouropoulos, E.: Musical rhythm: A formal model for determining local boundaries, accents and metre in a melodic surface. Music, gestalt, and computing (1997) 277–293
37. Simonton, D.: Computer content analysis of melodic structure: Classical composers and their compositions. Psychology of Music **22**(1) (1994) 31–43