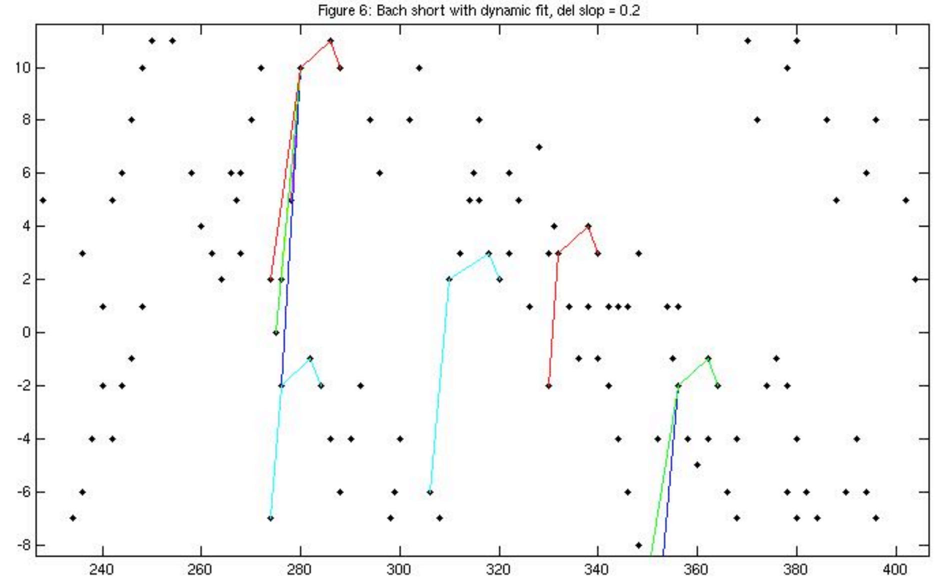
Math 470: Honours Research Project

Computational Music Analysis:

The Development and Comparison of Methods and Measures of Motive Detection

Figure 6: Bach short with dynamic fit, del slop = 0.2



By Finn Upham
110227477

Submitted April 28th, 2006

To Prof. Adrian Vetta
Office: BURN 1118

Abstract

Repetition in music is one of the means by which we understand it: a theme's return, a choruse's revival signal organisation in the work through time. Patterns just a few notes long can give a piece cohesiveness in familiar and unfamiliar genres. Our cognitive mechanisms allow us to recognize the same abstract patterns within segments of music even if, on the surface, there are significant differences. The questions of what defines the abstract patterns, and what differences qualify as perceptually acceptable ambiguities have lead to many searching for mathematical measures of similarity in melody and musical line. Music theory's historical bias towards the written score has caused great confusion amongst composers, listeners and researchers as to what transformations of musical gestures ensure or interfere with the ear's interpretation of recurrence as the music goes by. In hopes of shedding light on this quandary, this project explored different computational approaches to searching within pieces for similar segments, through which definitions of similarity were tested and compared. The music samples used were from Fugue 8 in J.S. Bach's Well Tempered Clavier, and Arnold Schoenberg's Kalvierstrück.

Hypothesis

Given a larger set of points in $\mathbb{R}^2$, called "music", and a model, M, of a few points, find subsets of the music which are "similar" to M.

Here, the music is abstracted to points in Time x Pitch space, where each point represents a note as described in the music's score, p_i being the pitch of the i^{th} note in the score, and t_i being the time it begins. The model, M, is a sequence of notes ordered in time that might otherwise be considered a musical motive, theme or piece of melody, and thus have the characteristic of $t_i \gneqq t_{i+1}$.

The definition of similarity of the subsets is a big question in field of music perception. The goal here is to test measures of difference which are representative of the ways humans hear and understand music, and thus many of the traditional means of comparing points in space are not relevant. Mapping the music into $\mathbb{R}^2$ can be misleading; it suggests the two dimensions being considered, pitch and time, are interchangeable, or at the very least comparable. Perceptually, the measurement of time and pitch are very different processes and any cognitively significant measure of deviation should treat them as such. Many systems have been suggested for comparing bits of music, a few which were tested in this research project, but the focus of research was the 'finding' algorithms. The programs created were designed to be flexible, able to test more measures than featured below.

Materials

Within the MatLab programing environment, most of the testing was done on a G4 Powerbook, 1.33 Ghz. Two pieces were used as test subjects: J. S. Bach's Fugue VIII, in D# minor, from the first volume of the Well Tempered clavier and Arnold Schoenberg's “Klavierstruck”. For the sake of simplicity, only excerpts were taken from these. The fuga.m file encodes the information from the first 16 measures and the last 12 of the Bach Fugue. Klav.m describes the first 68 bars of Schoenberg's piece. The music was thus reduced to a matrix of points such that each row was a note, ordered first chronologically, second by pitch, with the first column being the time onset in reasonable musical time unites and the second being the pitch measured linearly in semitones for the origin, middle C. Other information such as voicing/timber and dynamics were encoded, but not used for this project. The last column of the music matrix was the index of the note, or row number, which was useful for retrieving note specific information after subsets had been selected. The models for each piece were found in reference to the scores as motives that did return within the excerpts. For the Bach, the first four notes of the incipite were used, being the beginning of the fugue subject, and in the Schoenberg, a four note pattern from the first measure was chosen.

Methods

The problem of finding relevant bits of music was divided into three parts: segmentation, extraction and comparison. The first, segmentation, was common to all the methods applied, the extraction had two forms, and the comparisons were testing definitions of similarly – in this project, three were attempted.

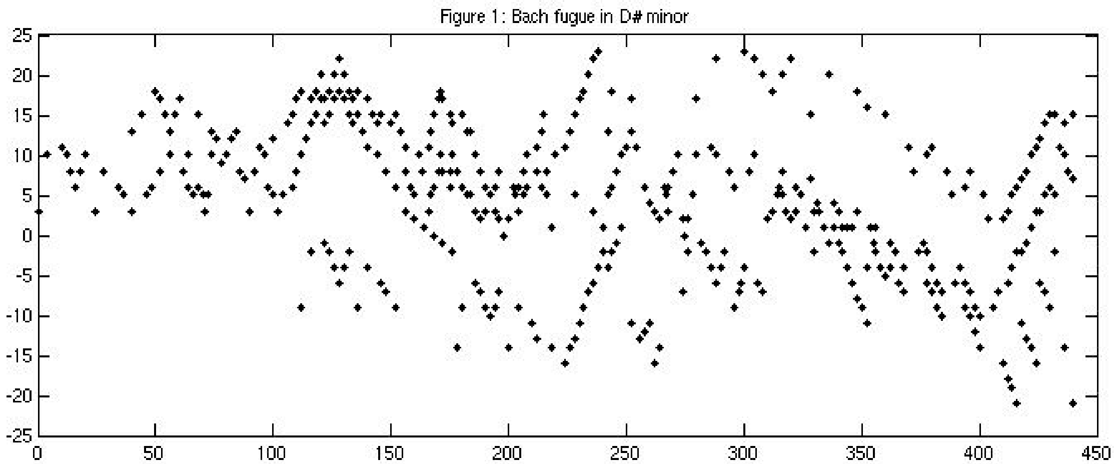


Figure 1: Abstraction of Bach's fugue in D# minor, mm 1-16, and mm 75-87.

The segmentation process clumped the music in to segments of length Δt, each segment a matrix entry of a cell defined by it's starting note, α_i, and all notes from the music to follow it within time Δt. Thus every note but the last few began one of these clumps of music. The search algorithm could thus run exhaustively through the music, looking for similarity within a reasonably sized number of notes.

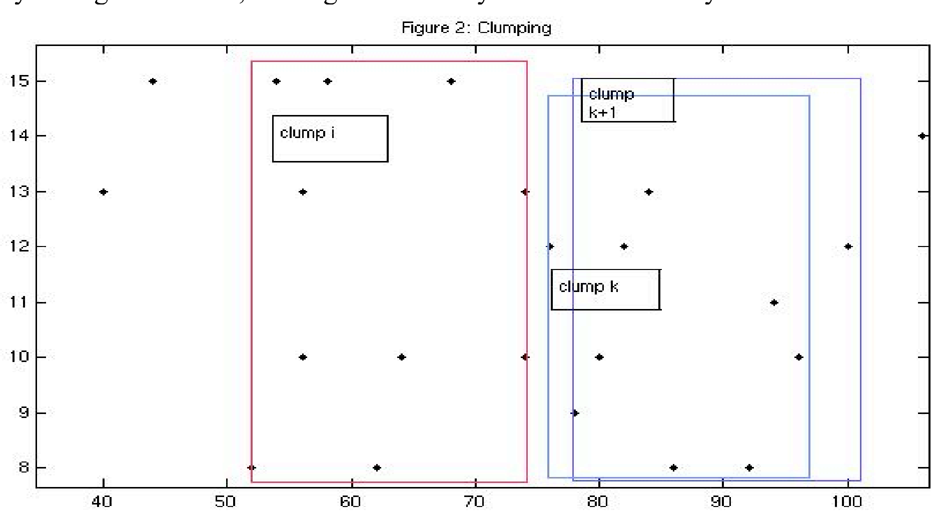


Figure 2: Examples of segments or clumps in which the model might be found.

The extraction methods were the crux of the project, and the two methods, fixed form and interval dynamic, had their respective advantages and limitations.

The fixed form approach defines the mapping of a line segment onto the model, to be translated through the piece to collect points that are close enough to the line. This type of model abstraction was used by Anna Lubiw and Luke Tanur for a similar searching process, using the duration of notes to form step functions. In this exercise, I used the idea that music gives the impression of being continuous in it's motion, and thus interpolated the points with a smooth curve: the nth degree polynomial. As is seen in figure 3, polynomial interpolation is not the best way to approximate the curve we might imagine from a nice musical line, but the damage is minimized with a small model.

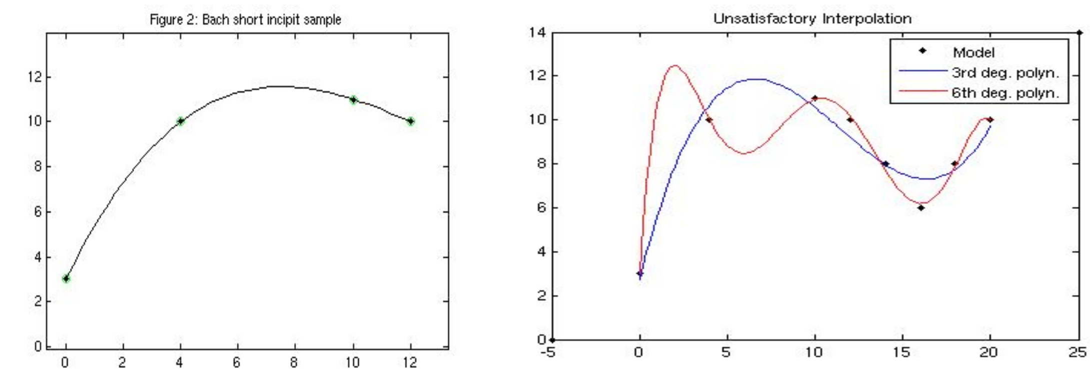


Figure 3: Examples of good and bad polynomial matching on the Bach incipite. The first has a good fit with a third degree polynomial creating a smooth curve. The only problem is the location of the maximum, which is between notes. The second is a longer excerpt of the same incipite. The blue curve has a general shape similar to is suggested by the notes, though being only a 3^{rd} degree polynomial, it does not fit the points closely. The red curve has a tighter fit but is much too bumpy with the local maximum between the first two notes, instead of at the third.

The interpolation algorithm used was native to MatLab: the polyfit function was quick and efficient, producing the coefficients of the polynomials that best fit the points given the degree of the polynomial specified. Once the line segment in pitch time space was defined, it was successively translated to beginning from every note in the score. Since each note is the α note of a clump, the algorithm collected all notes in that clump within some ϵ of the line. That extraction would be saved as a similar subset if the number of notes was close to or a greater than the number of notes in the model.

The second method of extraction defined the relations between successive notes in the pattern. Differences in pitch, in time, and the slope of the line from note i to i+1 were recorded in an array.

Now, the clumps to be searched have been defined by Δt, but there is another relevant time interval: the distance between successive notes. For the interval dynamic extraction process, we define δt to be twice the longest distance between successive notes in the model. This δt serves to cut down on the useless computation by limiting the number of comparisons each note must check within each clump to only those that might be musically useful. Thus, within each clump, relations are measured for all non simultaneous notes that are within δt of each other. Once all the relevant pairings are defined, they are compared to the successive pairs in the model, calculating how much they differ from the first to (n-1)st interval. The resulting three dimensional array is of the "cost" to use the i-jth pairing for the kth interval in the sequence. To find the succession of notes with the lowest total difference, a n-1 layered matrix is formed, each layer counting the lowest cost for that interval to be used thus far in the algorithm. In more detail: it is assumed that the α note is the first in the sequence. In the first layer of the cost matrix, the measured differences of α to its potential successors are entered, In the second layer, all intervals whose starting point is a successor of α enter a sum of that $\underline{\alpha i}$ first interval cost and the $\underline{ij}$ second interval cost. Following that, each succeeding note has entries in the next matrix layer, adding the kth interval cost to the minimum cost of the path leading to it. Thus, for every $a_{i,j,k}$ of the cost matrix its value is the minimum cost partial sum for the ith and jth notes in the clump to form the kth interval of the sequence. The best path can be found from the minimum entry in the last layer of the cost matrix and traced back to α. From here, that path can be accepted as a similar piece of music if the cost is below some specified ϵ.

The comparison section of the programs were the similarity or difference measures used by the extraction algorithms. Since the goal of this project was to create an environment in which different measures could be tested, those used for testing were kept simple, but more elaborate measures could be tried in another trial. For the fixed form algorithm, the measure of closeness was only difference in pitch, specifically,

$$\Phi(n_i) = | p_i - f(t_i) |$$

f(x) being the polynomial extracted from the model and translated to fit on α. While more traditional measure of euclidean distance were considered, the relationship between time and pitch is not as simple as that suggested by this geometric form of abstraction of music, and a finding a good way of measuring the impact of shifts in each direction will require further study. This comparison process allowed for all but the first note to be a little bit off, and the extraction doesn't take measure of the accumulated variance

For the interval dynamic extraction algorithm, two measures were tested: that of pitch difference difference, or pitch interval difference, and slop difference, in radians. With the dynamic approach, the differences are cumulative, thus ϵ caps the total.

Results

All processes used managed to pick out the model in the music and the perfect translation of the model found in the Bach fugue, which can be taken as a good sign, but besides that they diverged significantly. The results of the applications of the algorithms can be seen below in graphic form and found in the first and second appendices. Each subset is a colour path through the notes collected. Instead of interpolating or presenting the translated fixed form, the lines simply link the successive points to expose the range of shapes each algorithm selected. To facilitate comparison, the parameters were chosen to output similar numbers of subsets. The model used for the Bach fugue was the beginning of the fugue subject, and can be seen in the first graph of figure 3.

Figure 4 presents the selections made by the polynomial fixed form on the Bach fugue excerpts. The first four notes, the incipite, were correctly identified as the model, and the entry of the third voice was also correctly identified as the same shape in translation. But consider the last two subsets, in blue and green: the green could not be interpolated to have the same curve as the model because it would obviously have second oder critical points, and the blue has a repeated note at it's peak, which is musically completely different from the small steps of the model.

Figure 5 shows the dynamic interval output using differences in pitch as a measure. This kind of measure has been historically popular. More than the rhythm or accompanying harmony, pitches have been used to describe melody in the western music tradition. This measure worked remarkably well in finding similar shapes, but that success seems to depend on the style of the piece. Since the fugue subject is used through the piece in a way that preserves the note patterns shape in time, most of the pitch exclusive information is enough to establish the similarity – measuring the patterns in time would be redundant. But this is not true for all musical genres, as demonstrated by the second example.

Figure 6 is the dynamic method using the slop difference measure. This process is also successful in finding similar shapes, and is particularly good at picking up ambiguates in the music. For example, in the time interval (250,300) there are four overlapping selections starting on different notes but sharing the last three. Depending on how the piece was played, it is possible that one of these would be heard over the rest as a reference to the model, but the choice of which is up to the artist. There were a few questionable selections in this trial as well, for example, around t=200, that first interval is much too

Figure 4: Bach short incipit polynomial fit with ep=1

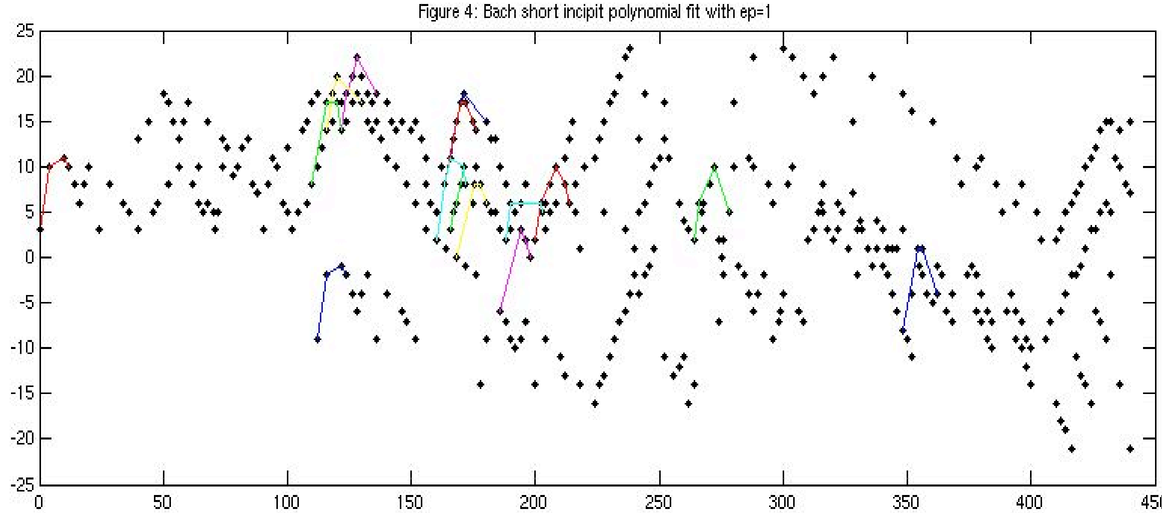


Figure 5: Bach short with dynamic fit, del pitch = 2

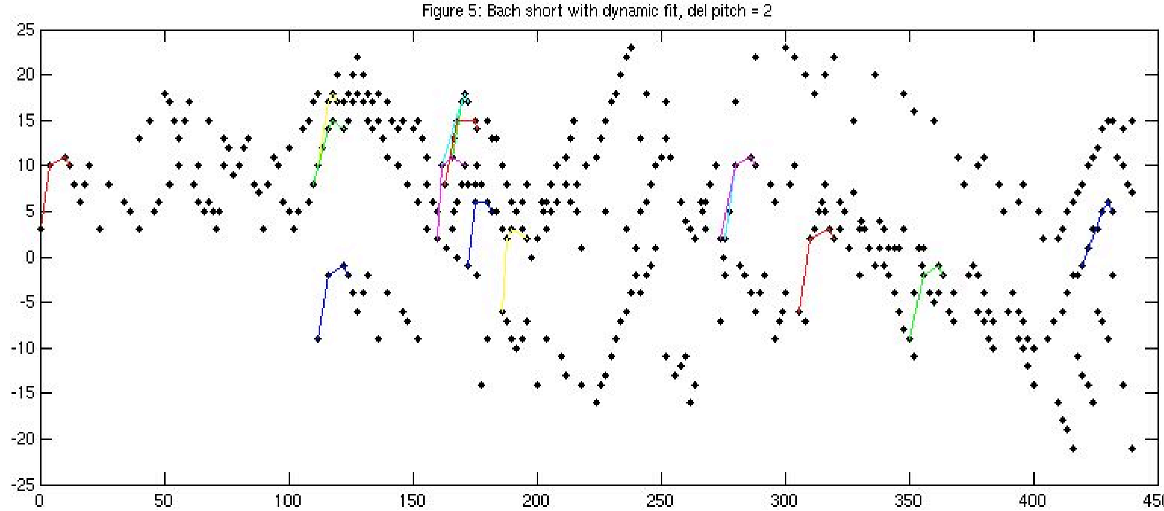
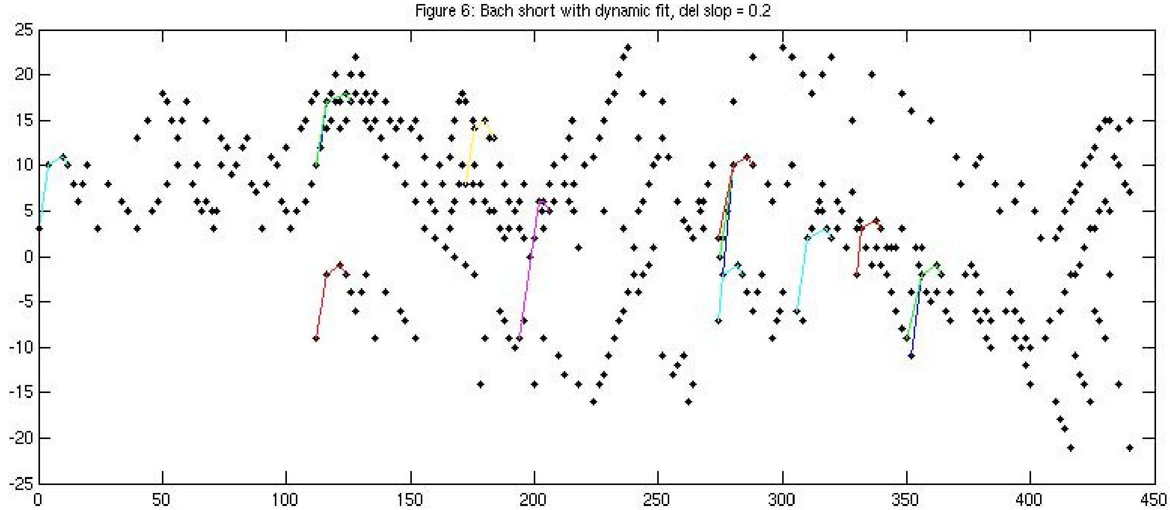


Figure 6: Bach short with dynamic fit, del slop = 0.2



big for the first note to sound like it is related to the rest of the pattern in a dense passage.

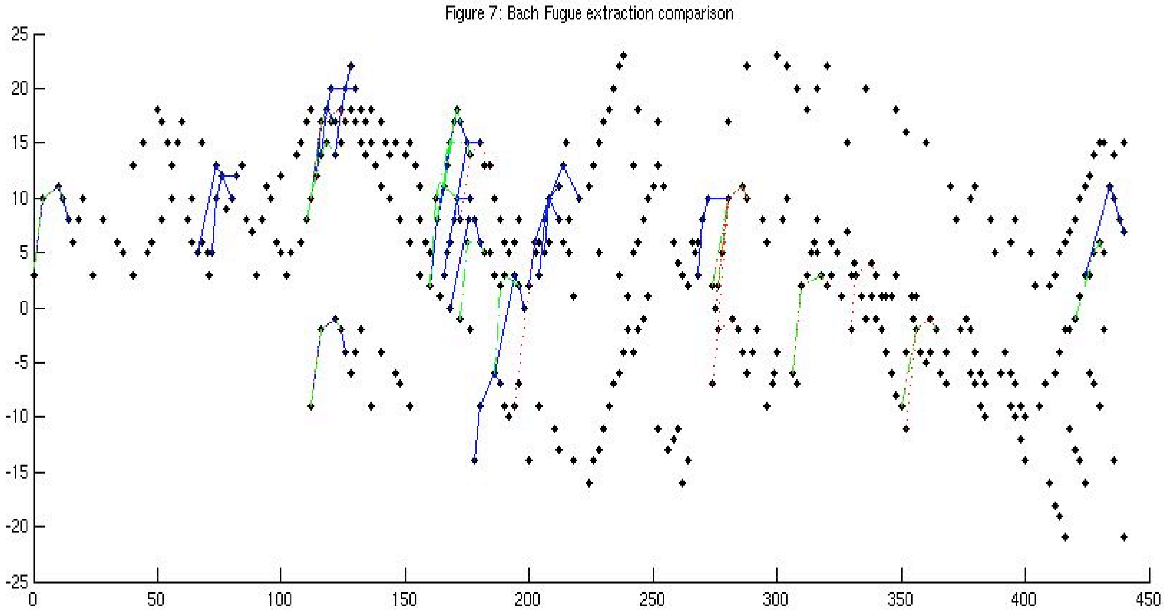
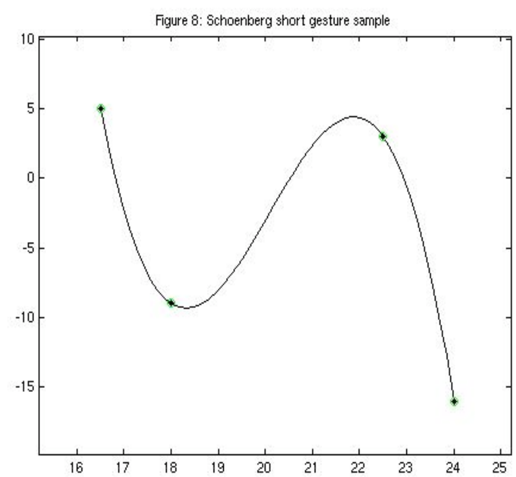


Figure 7: The collected results of trials on the Bach Fugue excerpts. In blue are the subsets chosen by the polynomial fixed form approach, the green are selected by the pitch difference dynamic algorithm, and the dotted red were found with the difference in slop measure.

Also of interest is how algorithms match up in their selections, as can be seen in figure 7. While there aren't so many clumps common to all three, the pitch difference dynamic process shared more with the fixed form and slop difference method than these did with each other. This can again be attributed to this musical style where the pitch content of melody is of primary importance.



To demonstrate this sensitivity to musical style, the other test subject was chosen from the atonal repertoire, a style of music that deliberately broke from the tonal western music tradition. Figure 8 is the 4 note model chosen from Schoenberg's piano piece “Klavierstrück”. A visual analysis of the score suggests that this shape recurres in the music, but defining a measure that can find it is not trivial. In fact, none of the measures tried were satisfactory. It should be noted that this second musical excerpt is shorter than the Bach, and the results are correspondingly fewer.

Figure 9: Klavierstruck short gest, polyres with ep=4

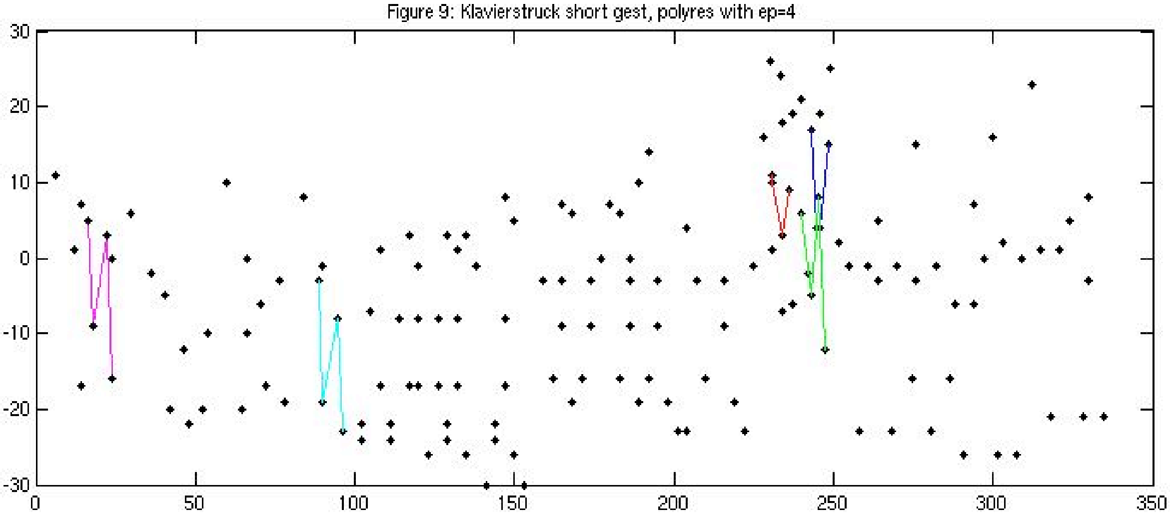


Figure 10: Klavierstruck short with dynamic fit pitch diff=4

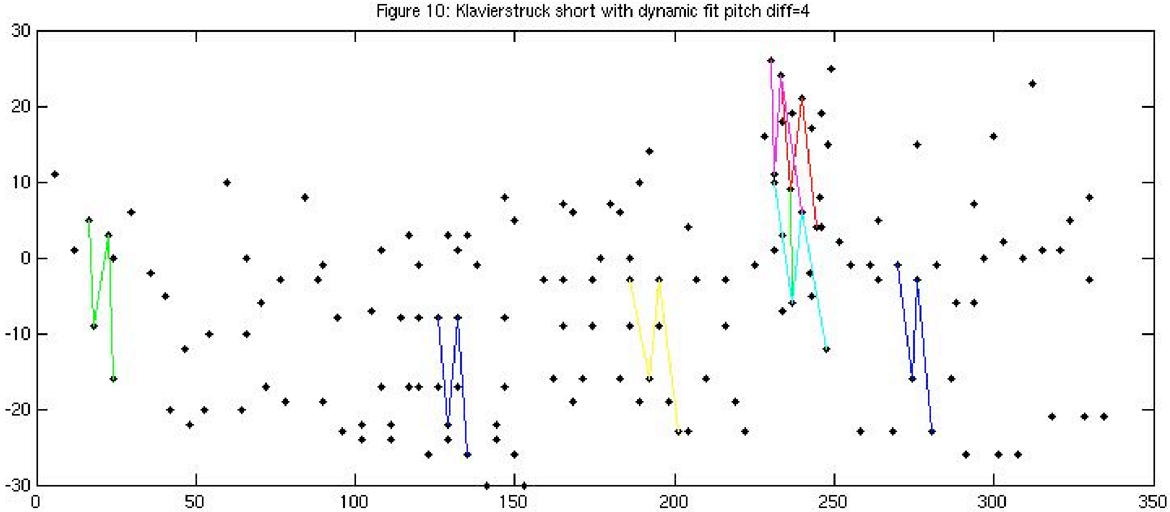
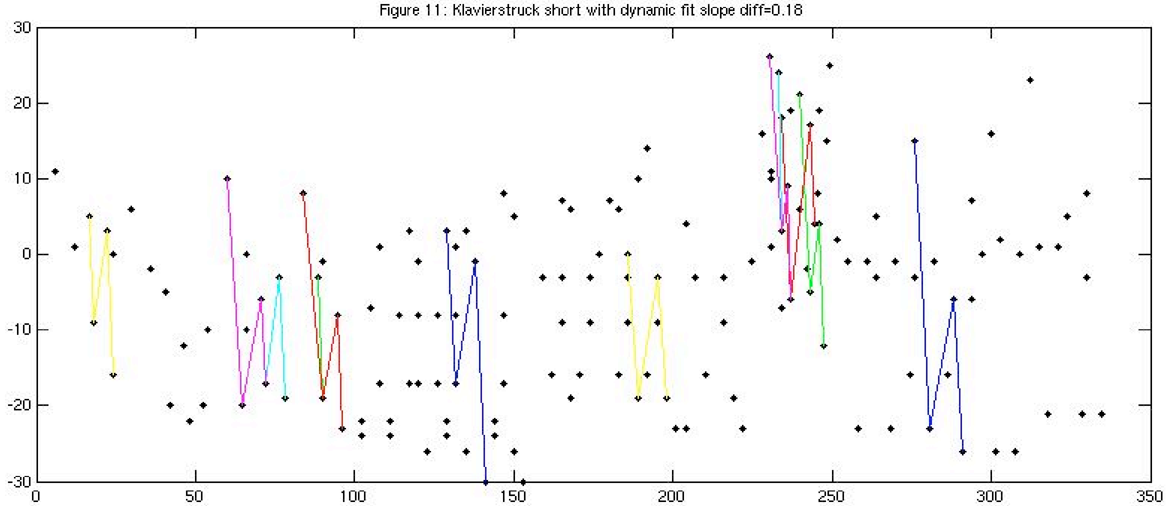


Figure 11: Klavierstruck short with dynamic fit slope diff=0.18



Using the fixed form approach, the results, in figure 9, were mixed and relatively few sparse. One of the draw backs of this approach is finding enough close notes in a smaller time interval, and thus only represent part of the segment. There are a few subsets that should not have been accepted, since they fail to communicate the oscillatory character of the model.

As seen in figures 10 and 11, the dynamic algorithms did much better in finding segments that satisfied the bouncing shape. One interesting contrast is around t=200 where both of these figures have a yellow selection in the same range, but on close inspection, it turns out that different notes were chosen. Compared to the model, being the first marked subset in these graphs, each look fairly close to the shape, but each process lost and preserved different qualities in the model. To determine which measure is more perceptually accurate, extensive listening tests would have to be done, which I am sure someone else has undertake or will be getting to soon. In figure 11, it looks like the model managed to suggest different pattern that recurres in the music, beginning with a much larger leap downwards. Whether these are effectively similar to the model or not, it is interesting to know it is possible uncover different organization structures in the test subject.

Conclusions

While it would be very difficult to define a measure that collected all the bits of music humans might hear as similar within a piece, testing some of the criteria is possible. Looking a selection method that was more flexible in time and pitch while preserving the contour of the musical segment, the interval dynamic approach was much more successful than that of the fixed form, though it might be possible to program re-parametrisation of the curve to make it stretchable. It would also be necessary to find a better interpolation method, since polynomials do not necessarily respect the range or critical point placement suggested by the points. As for the dynamic approach, some nice additions would be the possibility of collecting more notes than are in the model. Besides testing these programs on more music, larger works and different measures, some interesting follow projects would be to include more information from the music, such as metrical weight, dynamics and timber, and developing a program to sonically output the results in order to hear the degree of success.

Bibliography

Bach, J.S., Ed. S. Ferguson, "The Well-Tempered Clavier" *MUTH~210/211 Tonal Theory & Analysis.* Montreal, Eastmans Systems, 2004.

Buteau, Chantal and Guerino Mazzola. "From Contour Similarity to Motivic Topologies." *European Society for Cognitive Sciences of Music (ESCOM)* IV, 2: 125-149

Kleinberg, Jon. *"Dynamic Programming."* Algorithm Design. Boston: Pearson/Addison-Wesley, 2006.

Lubiw, Anna and Luke Tanur. "Pattern Matching in Polyphonic Music as a Weighted Geometric Translation Problem" Proceedings 5th International Conference on Music Information Retrieval (ISMIR) (2004): 289-296.

Schoenberg, Arnold. *Klavierstück : opus 33b.* Los Angeles : Belmont Music Publishers, 1932.