A GPR Based System for Real Time Segmentation and Analysis of Performed Blues

Bruce Pennycook,[*1] Carlos Guedes[#2]

[*1]Butler School of Music, Dept. of Radio-TV-Film, The University of Texas at Austin, Austin, TX USA
[#2]Faculty of Engineering, The University of Porto, Porto, Portugal
bpennycook@mail.utexas.edu.edu,carlosguedes@mac.com

ABSTRACT

A system to automatically extract musical phrases from performed blues is presented. This system is based on three sections: an audio transcriber, a segmenter based on Grouping Preference Rules and a set of secdonary analyses that extract pitch, tonality and contour information from each segment in near real-time. The output of the system was compared to manually segmented blues excerpts. It was shown that segmentation success was depenedent in part on musical styles and the selection of relative weights for the collection of rule descisions.

I. INTRODUCION

The research presented herein is one component of a large, multi-year research initiative funded under the University of Texas at Austin – Portugal digital media initiative. The larger project is entitled "Kinetic Control Driven Adaptive and Dynamic Music Systems". The over-arching concept of the project is to create a mobile device application that can a) listen to monophonic performed music b) create similar melodic materials within the context of generative rhythmic and harmonic templates. The research used as a starting point Lerdahl and Jackendoff's (1983) seminal work on grouping preference rules, the real time implementation of a GPR based techology (Pennycook and Stammen, 1994; Stammen and Pennycook, 1994). Certain elements from the Local Boundary Detection mode (Cambouropoulos, 1996) such as style considerations and extending the look-back, look-forward short term memory to eight notes influenced this study. Juhász (2009) demonatrates that a large corpus of melodic materials can be segmented through self-organizing maps however these methods were not applicable to realtime performance.

II. OBJECTIVES

The overall objective of the research is to capture performed music and then generate similar output. Previous efforts by, most notably, David Cope who has published extensively on algorithmic composition (Cope, 2000) and music pattern detection (Cope, 1998), Françis Pachet (2003) and Tristan Jehan (2005) have demonstrated that algorithmic generative technologies can be applied to acquired musical materials to produce similar musical output. Nevertheless, our more modest pursuit, predicated on the analysis of monophonic melodies, may contribute some useful insights into the general problem of generating new musical materials that resemble the input. Specifically, the objectives of the Musical Phrase Segmentation (MPS) system presented here are to capture performed melodies and create running, real-time segmentation and segment feature analyses from note event data. The segments are musical phrases of varying lengths that provide a wealth of information for the subsequent generative procedures. These generative procedures and software will be presented elsewhere.

III. PROCESSES

A. Note Event Representation

While we fully recognize the many advances in audio signal feature analysis using MIR technologies, note event information provides the kind of information required for the music phrase analysis procedures central to this research. Specifically, segmentation of monophonic melodic materials from audio or MIDI instrument sources using as a starting point the well-known Grouping Preference Rules (GPR) presented in 1985 by Lerdahl and Jackendoff (1983).

B. Analysis Components

The MPS system is a form of melodic analysis that utilizes the following information extracted directly from the source signals. The system first extracts pitches (0-127), durations (including delta times) in floating point and dynamic levels (0-127). Next a dynamic beat tracker extracts a running tempo value (averaged at the end of the run) and a representation is constructed which includes: pitch #, velocity, duration, onset time, offset time, total running duration and the interval from the prior note event. The accumulation of all this data is retained in a long-term memory as flat text file. The accumulating data is also sent to a short-term memory system in eight note-event packets. (From hereon we will use the term event to mean pitch-time-amplitude MIDI like data representations).

1) N8 Analyser

The central technology of this system is called the N8 Analyser. As shown in Figure 1, the phrase extractor requires three past events and four future events.

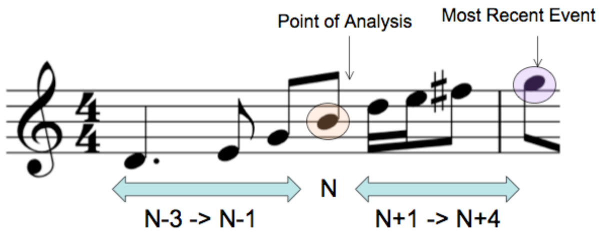

Figure 1. short term memory of N8 analyser

This memory structure achieves two goals: first it provides a sufficiently long set of past and future events to evaluate the likelihood of a segment and second, it minimizes the output lag time to eight events. It is important to recall here that

Lerdahl and, later, Pennycook and Stammen [2] used fewer prior and future events to assess the phrase boundaries. Through extensive experimentation with a variety of musical styles MPS has, for now, settled on N8 as the optimal compromise between analysis window size and latency.

2. GPR Detectors

The segment is determined by a linear sum of weights assigned to a set of GPR detectors. The detectors used in MPS include: rest, attack point, duration, dynamic level and multiple intervallic conditions. Each is shown in the following Figures 2 through 5.

Figure 2. rest detector

Figure 3. change of duration detector

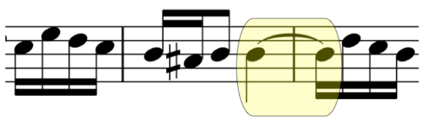

Figure 4. attack point detector

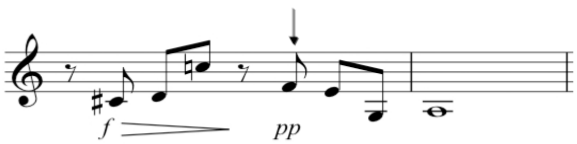

Figure 5. Change of dynamics detector

The interval detector is slightly more complex in that there are two levels of operation. Isolated large intervallic leaps receive a higher base-weight than direction change and change in intervallic motion. The direction change and intervallic motion detectors are particularly valuable if the music has long streams of similar durations (certain 18th C music for example) and, like all detectors may or may not contribute to a segment decision. In our system, the weights change dynamically depending on how many interval types are detected and the second and third interval detectors are only activated during a run of similar durations. All three interval detectors are shown in Figure 6.

The N8 analyzer executes the detector tests at the end of each event (note) to decide if a new segmentation point can be determined. The Analyzer assumes that the most recent event is actually four notes in the future, with the current note in focus, "N," being four notes back. In other words, by looking only 4 notes into the future, we can glean a lot of information about the events surrounding "n" and the test is only 4 notes behind the live performer. The weights are calculated for each event and are accumulated over time. Eventually, any one test will meet the maximum weight threshold, but with enough simultaneous accumulated weights, a threshold can be met at any time.

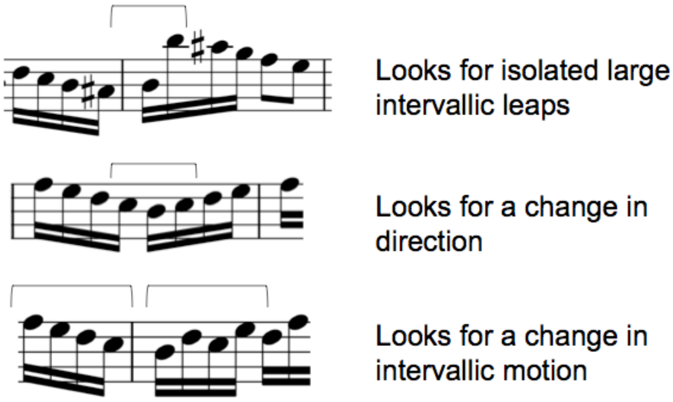

Figure 6. interval detectors

IV. Methodology

A series of blues solos were performed by Mário Santos, a Portuguese saxophone player. The player was asked to create a series of blues solos that increased in complexity from simple (linear, non-chromatic) to highly chromatic with angular intervallic motion. The recordings were then organized by complexity and processed by the software as follows:

- Single voice monophonic audio file
- Transcription to midi-like internal format using the Max/MSP fiddle pitch detector and a novel beat tracker developed by George Sioros[Sioros, 2010]
- N8 GPR Segmentation → Segment Event File
- Secondary Analysis → Detailed Segment Analysis File

The segment event file is an abbreviated form that serves as input to another max program to play and compare individual segments. The detailed segment analysis file contains results from all the secondary analysis tests for each segment including intervals, pitch classes, estimated tonality, estimated root and contour number from the dynamic time warp sub-system. These functions will be described below.

For comparison, the source audio files were loaded into Peak Pro and a set of markers were entered at aurally determined segment boundaries. These manually determined segment times were then used to control the playback of the audio file within a segment playing program designed to let a user compare the output of the system to the manually marked files. Segments may be auditioned individually in three ways: event data (played as MIDI), the corresponding audio segment derived from the timings of the event data and a comparison audio file with segments selected aurally. Ideally the event list MIDI and audio segments should closely correspond to the hand made segments.

A. Manipulating the GPR Weights

During the development of the system, it was determined that the success of the analysis was largely style dependent (Pennycook et al, 2011). That is, weight settings suitable for a solo passage from the Bach Flute Sonata in E major, BWV 1035 or an excerpt from Anton Webern's Op.1 (clarinet solo), were not the same. Through empirical testing a set of weights best suited to jazz and blues were selected and used in the initial test runs for this study. Figure 7 illustrates the user interface for setting a number of critical parameters including the GPR weights, ornament duration minimum (to remove grace notes and ornaments), minimum and maximum segment note count and a rest toggle that removes rests from the analysis process.

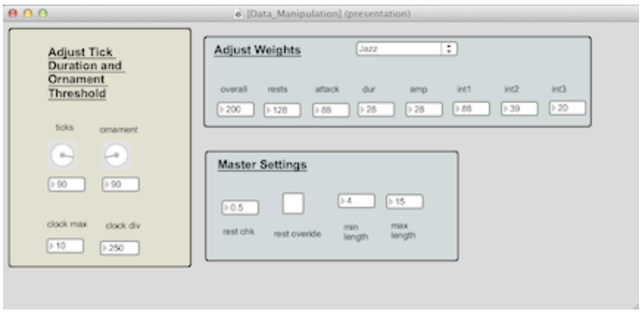

Figure 7. User Interface for Weights

The different styles were used to determine optimal weight sets. For example the weights for the Bach flute sonata, Charlie Parker and Webern (solo clarinet fragment) are shown in the Table I.

Parameter	Bach	Parker	Webern
Overall sum threshold	256	200	150
Rests	200	128	128
Attack	100	88	88
Duration	28	28	88
Amplitude	28	28	28
Interval (heavy)	88	88	100
Interval (medium)	29	39	50
Interval (light)	10	20	32

Table I: Weight Assignments by Style

These values are entirely arbitrary having been determined through trial and error. However they do support an important observation of the study and that is: the presence of rests, interval leaps or direction change, changes in duration, attack strength and relative beat position (not shown in this table) significantly affects the determination of segmentation points and that these values are style dependent.

The following Table II illustrates the differences in the segmentation of the opening passages of the Bach flute excerpt: one with weight settings for Bach and the other for Parker.

Bach Flute Sonata E major m. 1-12

The values after the word Event are: note number, amplitude (0..127), duration in ms and running total duration in ms.

Bach settings	Parker jazz settings
MPS 1	MPS 1
Seg_length 20	Seg_length 16
Event 71 70 341 341	Event 71 70 341 341
Event 76 100 682 1023	Event 76 100 682 1023
Event 76 100 682 1705	Event 76 100 682 1705
Event 76 100 170 1875	Event 76 100 170 1875
Event 80 100 171 2046	Event 80 100 171 2046
Event 78 100 170 2216	Event 78 100 170 2216
Event 76 100 171 2387	Event 76 100 171 2387
Event 75 100 170 2557	Event 75 100 170 2557
Event 73 100 171 2728	Event 73 100 171 2728
Event 71 100 170 2898	Event 71 100 170 2898
Event 69 100 171 3069	Event 69 100 171 3069
Event 71 100 170 3239	Event 71 100 170 3239
Event 76 100 170 3409	Event 76 100 170 3409
Event 75 100 171 3580	Event 75 100 171 3580
Event 73 100 170 3750	Event 73 100 170 3750
Event 71 100 341 4091	MPS 2
Event 69 100 341 4432	Seg_length 4
Event 69 73 341 4773	Event 71 100 341 4091
Event 68 100 682 5455	Event 69 100 341 4432
MPS 2	Event 69 73 341 4773
Seg_length 20	Event 68 100 682 5455
Event 71 70 341 5796	MPS 3
Event 76 70 170 5966	Seg_length 13
Event 75 70 171 6137	Event 71 70 341 5796
Event 76 70 341 6478	Event 76 70 170 5966
Event 76 77 852 7330	Event 75 70 171 6137
Event 80 70 170 7500	Event 76 70 341 6478

Table II: output of the N8 Segmenter with Bach Flute Sonata using different sets of weights.

This table shows that the weights that worked best for the Charlie Parker tune, produced more, shorter segments than the weights determined to be most suitable for the Bach. The challenge here is to determine which is “correct”. (Interestingly, the total number of notes in MPS Segment 1 for the “bach” weights equals the sum of MPS Segment 1 and Segment 2 in the “parker” weights although greater deviations appear later in the excerpt.) An argument could be made that the first phrase does indeed end after the four eighth notes in mm. 8 and 9 (up to the rest). However the GPR system with the “parker” weights determined that the change in duration from a set of sixteenth notes to eighths generated a segment boundary resulting from a combination of the overall lower threshold (sum of weights) such that the change of duration detector pushed the total over 200.

B. Test Data – Blues solos

The test data in this study was a series of tenor saxophone solos based on 12-bar blues improvisations. The reason for the focus on improvised soloing is that the detected fragments will eventually be utilized in the “Gimme ‘da Blues” project. For this study six phrases of increasing intervallic and rhythmic complexity were recorded and edited into audio files of approximately 25” each. The files names were given as suggestions to the soloist: head (the blues tune), simple, simple with leaps, lyrical, angular and complex.

2) Results

The first example is an analysis of the melody of Thelonius Monk’s “Straight, No Chaser” (performed by Portuguese tenor sax player, Mário Santos). This piece is a kind of “blues riff” in itself. Figure 8 shows a comparison of manual markers and N8 Segmenter markers placed in Peak Pro. The weights for the analysis were adjusted to reduce the impact of rests along with increased emphasis of attack, amplitude and interval direction change. Overall, the N8 Segmenter tended toward longer, more complete phrases rather than short bursty “riffs”. Adjusting the weights such at that attack, amplitude and interval were emphasized over rest weights, the segments were shorter and closer to the manual segmentation values.

Figure 8: Manual and Derived Segments show – Peak Pro markers

The upper image shows the manually placed segment markers and the lower image the segment start points from the N8 segmenter. Regions separated by rests predictably show clear correspondence. The N8 markers indicate that leaps (GPR 3, register change) and abrupt changes of duration (GPR 2 Proximiy) were the strongest contributing factors in this example.

C. Audio vs MIDI

The accuracy of the results with monophonic audio file (or live performance) input was compared to the results with MIDI sequence data using the same musical materials. These tests were not entirely successful due to errors in the audio to midi transcription process. Anyone who has tried to play music into a sequencer or notation program knows that even with midi keyboard data, the rhythmic values are approximations determined by tempo, beat position and accuracy of the performer. All such programs have rhythmic quantizers to help clean up resulting data. But quantizing the data is far more useful for capturing data for a score or creating beat-based music and not particularly helpful for capturing the nuanced rhythms of improvised performance.

For example, the data from the real-time transcription of the sax improvisation file, “simple.aiff, produces a nearly perfect match when listening concurrently to both the data and the original audio. Using the timing data from the analysis file, a sequence of start time-end time duples are sent to an audio file player in sync with the playback of the detected data as midi output. Pitch nuances such as glides and scoops are lost during the analysis. However, the segment boundaries, pitches and rhythms closely match. There is no reason not to retain sub-note level pitch details and that is something that will be added at later date.

D. Secondary Analyses

The N8 segmenter produces two kinds of output files. The files in the previous examples are called “abbreviated output” as they contain only the data needed to play and verify segmentation point results. The full analysis file contains a set of secondary analyses of the data on a segment-by-segment basis. Table III shows the beginning segments of a full analysis file. In this example, it is the complete analysis of an audio file of the Thelonius Monk tune, *Straight, No Chaser* as performed by the tenor sax soloist. Segment weights were set to the “parker” values

The data in this list is constructed at the end of each segment boundary. The note lists contain several additional pieces of information compared to the abbreviated lists:

- Pitch, amplitude, event duration, note on time, note duration, interval, beat, beat position
- In this case, event and note durations are the same. Interval is the change from the last note event, beat is the estimated number of beats based on the beat tracker and beat position is an offset in milliseconds.
- After the note lists the data includes:
- Seg_ticks – how many ticks have elapsed in this segment (note that ticks are not necessarily milliseconds and can be set in the user interface)
- Seg_GPR indicates which GPR rules were activated and exceeded the threshold. In this case the Rest,

Attack_Point and Interval_a rules invoked the segment.

- Hi_Lo: upper and lower bounds of the note range in the segment
- Root: this is an estimate of the root of the chord that a collection or chord of these pitches would produce. It is based on a modification of the Parncutt root finder. In this case, the algorithm determined that this is a chord on “2” and is probably a D minor seventh.
- Tonality: this is an estimate of the tonality of the segment. In this case, C which is consistent with the Root estimation of Dmin being “2”.
- Pitch: a list of the accumulated totals of each pitch class value
- Interval: a list of the accumulated total of each interval class
- Template: 2 (see below)
- Weighted_p: histogram of the pitch class totals
- Weighted_i: histogram of the interval class totals

MPS 1
Seg_length 7
55 80 128 70 70 55 0 77
56 84 162 360 162 1 1 53
60 88 105 465 105 4 1 79
62 90 127 592 127 2 2 15
60 85 93 685 93 -2 2 39
65 90 709 1394 280 5 3 13
Seg_ticks 1405
Seg_GPR Rest Attack_Point Interval_a x x x x
LowHigh 55 65
Root 2 2 D min7
Tonality 12 c
Pitch0 0 3 1 0 2 1 3 0 4 0 5 1 6 0 7 0 8 1 9 0 10 0 11 0
Intervals 0 0 1 1 2 3 3 0 4 1 5 1 6 0 7 0 8 0 9 0 10 0 11 0
Template 2
Weighted_p 0 3 2 1 5 1 8 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
Weighted_i 2 3 1 1 4 1 5 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Table III: Detailed Segment Analysis Output

3) Contour Templates

The value2 for “Template” refers to a contour matching process wherein each segment is passed through a Dynamic Time Warp function and matched against the nine basic contours described in Huron [4]. Figure 10 shows on the left the normalized segment shape and the left indicates the matched segment from a list of Huron’s nine possible shapes.

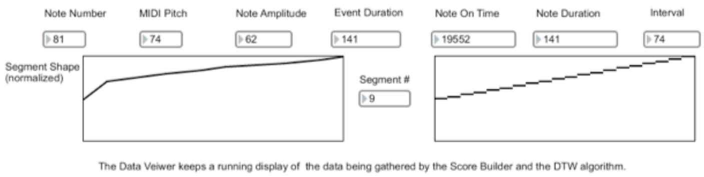

Figure 10: Section of the Max/MSP program showing input data on the left and a matched contour on the right.

V. Observations

The MPS software produces reasonable phrase_analyses of a variety of musical styles. Given that the objectives were not to develop a general theory of musical segmentation but rather to extract melodic phrase materials from arbitrary input that will be forwarded to various generative routines, it can be considered successful. Another important observation is that the 2-note tests of Lerdahl and Jackendoff work only in very limited cases and even the 4-note version developed by Pennycook and Stammen fails to find some very obvious phrase boundaries. We cannot say with certainty that our N8 Analyzer produces fewer ambiguities than other methods such as LBDM. Finally, the obvious question here is what about harmonic implications that frequently govern phrase structure in tonal music? Regrettably, no reliable polyphonic audio analysis system exists today and given our objective of audio input, a polyphonic phrase segmentation system will have to wait.

VI. CONCLUSIONS

We have created a software system based on extensions to the Grouping Preference Rules. The MPS software extracts reasonable and believable phrase boundaries in monophonic audio signals that can supply real-time date to generative operations. In addition we have produced output that is rich in analytical information about each segment including a pattern matching function with descriptive information in the form of melodic contours. Finally, the testing platform to observe output in real time of audio, musical scores and midi files

ACKNOWLEDGMENT

This work was supported the Fundação para a Ciência e a Tecnologia of Portugal and funding from the University of Texas at Austin-Portugal project.

REFERENCES

Cambouropoulus (1996). A Formal Theory for the Discovery of Local Boundaries in a Melodic Surface. *Proceeedings of the III Journees d’Informatique Musicale*, Caen, France.

Cope, David. (1998). "Signatures and Earmarks: Computer Recognition of Patterns in Music.." In *Melodic Similarity, Concepts, Procedures, and Applications.* Walter B. Hewlett and Eleanor Selfridge-Field (eds.). Cambridge, MA: MIT Press.

Cope, David. (2000). The Algorithmic Composer. A-R Editions. Madison, WI.
Jehan, Tristan. (2005) Creating Music by Listening. PhD. Thesis. MIT.
Huron, David. (1996) “The Melodic Arch in Western Folksongs.” Computing in Musicology, Vol. 10.
Juhász, Zoltan (2009) Automatic Segmentation and Comparative Study of Motives in Eleven Folk Song Collections using Self-Organizing Maps and Multidimensional Mapping. New Music Research. 38:1, 71-85.
Lerdahl, F. and R. Jackendoff. (1983) A Generative Theory of Tonal Music. MIT Press. 1983.
Pachet, François, (2003). The Continuator: Musical Interaction with Style. *New Music Research*, 32:1. 83-93.
Parncutt, R. (1988) “Revision of Terhardt’s Psychoacoustic Model of Root(s) of a Musical Chord.” *Music Perception*. 6:1.
Pennycook, B. & D. Stammen. (1994) “A model of tonal jazz improvisation”. *Proceedings of the 3rd. International Conference on Music Perception and Cognition*. ESCOM. Liege.
Pennycook, B., C. Guedes, R. Pinkston, H. Lobel. (2011, January) Kin.MPS. Paper presented at the Society for Electro-Acoustic Music in the United States. Miami, FL.
Sioros, G. (2011) kin.beattracker. Max/MSP external object. http://smc.inescporto/kinetic