A SINUSOIDAL SYNTHESIS ALGORITHM FOR GENERATING TRANSITIONS BETWEEN NOTES

Bryan Holloway and Lippold Haken
University of Illinois, CERL Sound Group
103 S. Mathews / Urbana, Illinois 61801
E-mail: hollo@cerl.uiuc.edu

ABSTRACT

Producing realistic transitions between notes is an open problem in real-time sinusoidal synthesis. It is possible to analyze a large number of different transitions (e.g. tongued or untongued, with or without bow change, varying intervals, dynamics, and pitch ranges) but a musician controlling the synthesis might play a transition that hasn’t yet been analyzed. We are developing algorithms to manipulate pre-ordained transitions to produce the new transitions required by a musician. We use an extended McAulay-Quastler (MQ) technique to analyze transitions between several parts of notes on a violin. These transitions are presented by our algorithm, which can produce modified versions of the transitions as required by the performer.

1. INTRODUCTION

A transition between two notes consists of the time between when the first note begins to amplit and the second note has reached a sustained level (for non-purposive instruments) or a decay (for percussive instruments.) We present a real-time synthesis algorithm for generating transitions from scratch. A musician can then play a keyboard where real-time synthesis occurs not just during the notes, but also between them.

Sfra предложил и синтезировал музыкальные переходы, используя дискретное время-среднее преобразование Фурье (DSTFT) (Sfra 1987). We are pursuing further investigation using an extended McAulay-Quastler Analysis/Synthesis technique. Unlike the DSTFT, this technique doesn’t require fundamental pitch tracking, and one can analyze non-harmonic sounds (Quastler and McAulay 1985).

2. TRANSITIONS

1.1 Three Transition Types

After looking at MQ analyses of several transitions, we classified three types of behavior of tracks (partial or sinusoidal components) in a transition, as shown in Figure 1. The first type is where tracks overlap for the duration of the transition, requiring two separate oscillators for sinusoidal additive synthesis. The second type is where MQ analysis connects the track from the first note to the track of the second. In this case, only one oscillator is needed for synthesis. The third type is where the first track ends while the second one begins at the same instant in time. This type also only requires one oscillator. One would think that this kind of frequency discontinuity results in discontinuity during resynthesis, but it has been shown in previous work that if the first note has dropped substantially in amplitude, there is no audible distortion during resynthesis (Sfra 1987).

Figure 1: MQ Analyser produces three different types of transitions. Here frequency is represented vertically, and time is represented horizontally.
2.2 MQ Transition Analysis

One characteristic of the MQ Analysis/Synthesis technique is the number of parameters that can be adjusted to produce a satisfactory result. The Capture Range is one such parameter that controls how much a frequency track can deviate from its current value (Quatieri and McAuley 1985). All three analyzed transition types can be produced by altering this parameter although the first two types are definitely predominant.

2.3 Background Analysis/Synthesis Work

Strawn used the DSTFT to analyze the same transition twice; once centered on the fundamental of the first note, and again centered on the fundamental of the second note. At the synthesis stage, he then crossfaded between the first analysis and the second, producing a realistic transition. Due to the magnitude of data provided by the DSTFT, Strawn went on to create line-segment approximations of the amplitude and frequency envelopes. Here he used Type 3 transitions where the first note ends at precisely the same instant of time the second note begins. He showed that straight vertical line segments between the two tracks did not produce distortion during resynthesis, providing that the releasing partial was at a low amplitude (Strawn 1987).

3. TRANSITION ANALYSIS USING MQ

3.1 Analysis of a Violin Transition

Unlike the DSTFT, only one analysis is necessary when using the MQ technique. In Figure 2, we see a transition of an ascending whole-step played on a violin. From the graph it is apparent that the MQ analysis indeed generates three different types of transitions. An enlargement of the lower harmonics is given in Figure 3.

![Graph showing violin transition analysis](image)

Frame length: 2.902 ms
Frames: 1 - 975
Capture Range: 2.00%

Analysis length: 2.829s
Frequency Range: 0 - 11025 Hz

Figure 2: MQ Analysis of an ascending violin transition from G (444 Hz) to C# (554 Hz)
Figure 3. The lower harmonics of the ascending whole-step violin transition. As the harmonic number increases, corresponding harmonics drift apart in frequency. Eventually the 8th harmonic of the first note is connected with the 7th harmonic of the second.

3.2 MQ Artifacts

Examining our three-dimensional graphs (amplitude represented by shades of gray), it is evident that connected tracks during transitions appear to be a misleading artifact of MQ analysis. MQ can effectively analyze and resynthesize a transition with an interval of a whole-step. Given the tight parameters, it will track the fundamental such that tracks connect. As a track’s harmonic number increases, the frequency difference between the harmonics of the first and second note increases such that MQ no longer connects the two tracks. Instead, we get overlapping tracks of Type 1. Eventually the tracks drift to the point where MQ connects the nth harmonic of the first note with the (n-1)th harmonic of the second note. Although this isn’t necessarily incorrect, one would expect a proper analysis to connect the harmonic of the first note with the corresponding harmonic of the second. Resynthesis using MQ showed no perceivable distortion due to this “corruption” of the data. We conclude that overlapping tracks (Type 1 transitions) is the appropriate model for representing transitions. This supports Struew’s extending synthesis approach.

4. TRANSITION SYNTHESIS

4.1 Previous Work

Struew was able to successfully analyze and resynthesize transitions using the DSTFT. Using MQ Analysis and Synthesis, we want to analyze a number of recordings of transitions and extract the necessary components to generate new transitions (new intervals and note-lengths) using information from these analyses.

4.2 LemurEdit

To facilitate the process of extracting transition components, we wrote “LemurEdit,” a graphical editing tool for the Apple Macintosh. The program allows the user to read analysis data created by “Lemur,” an extended MQ
Analysis/Synthesis technique (Fitz, Walker, and Haken 1992). We use LemurEdit to then extract the necessary components and output a new set of data following the Lemur data format. The components are then resynthesized using Lemur's MQ synthesis routine. The result is a "library" of transition components for synthesizing new transitions.

4.3 Altering Transition Intervals

Earlier we came to the conclusion that transitions can be successfully modeled as overlapping tracks. Using LemurEdit, we manually identify and extract the release and attack between successive notes for different intervals and different timbres. Our initial experiments use a variety of violin transitions including high and low notes, with and without bow change. Preliminary work involves changing the pitch of one of the notes thus creating a transition with a new interval.

4.4 Future Work

At this writing we've been able to successfully generate new intervals from our library of transition components. Now we want to alter note lengths of transitions. To properly handle the real-time synthesis, we dynamically select the appropriate release from the transition library, and then synthesize it, together with an attack (also from the library) that most closely matches the desired new note. When connecting transition components, we must consider both the amplitude and frequency of the next note. Scaling is done to make sure the appropriate tracks connect.

5. REFERENCES

