

Aaron Yang
ISE599
19 February 2004

Determining Context-Defining Windows: Pitch Spelling using the Spiral Array
& Mapping MIDI to The Spiral Array: Disambiguating Pitch Spellings
by Chew and Chen

These papers address the problem of disambiguating enharmonic pitches. Enharmonic pitches have the same frequency but can be spelled differently depending on the context of the surrounding notes and of the whole piece. Assigning the appropriate pitch names is important for music analysis, automatic transcription, and music information retrieval. The identification of a specific pitch in MIDI is just a number so correct pitch spellings need to be assigned for the music to be analyzed. The Spiral Array, a geometric model for representing music entities like pitch and showing the relationship between them, is used to find the correct pitch spellings. Chords can be generated from convex combinations of pitches and then keys are just represented by convex combinations of chords. More stable pitches have shorter distances to each other. Important pitches in a chord have more weight in respect to the other pitches. The convex combination of pitches weighted with their durations can generate a point in space, the center of effect. The center of effect will give away the key context of the successive pitches being put through the algorithm. The pitch spelling algorithm is biased towards pitches of fewer sharps/flats. The second paper implements a cumulative center of effect, which is generated from all the sets of pitches played so far. The software used to implement it, MuSA, analyzes the MIDI file as it is being played in real time. Graphically, the distance of the center of effect to the minor and major keys are shown and also the visual of the spiral array. The Beethoven sonatas, Op. 79 and 109, were tested against the system and were shown to have a 99.93% and 95.18% percentages of correct spellings. Unfortunately, the algorithm does not deal with linear motion of the piece and cannot deal with changes to key contexts quick enough. The first paper then

adds and tries to improve the success rate by introducing the concept of the sliding window. The sliding window is more sensitive to changes in local key contexts but cannot detect sudden changes to distant keys fast enough. So a third algorithm, two-phase, is created with both the ideas of cumulative c.e. and sliding window c.e. where the algorithm can revisit previous decisions. This improves spellings at a local level and gives more sensitivity to abrupt key changes. Other variations like excluding the cumulative c.e. and dynamically changing the window size are also attempted. The two-phase algorithm is shown to work better. So it can be concluded that short term and global contexts are both important to pitch spelling.

It would be nice to see these algorithms benchmarked against other algorithms using the same pieces. Using the sliding window was a smart idea since sliding windows are very good for dealing with entities that move in time. Improving the sliding window algorithm with a dynamic algorithm didn't show a more effective result, however other algorithms to innovate the sliding window may prove to be fruitful.