PITCH SPELLING ALGORITHMS

David Meredith
City University, London, UK

Background

Since the 1970s various researchers have developed *pitch spelling algorithms* that attempt to compute the pitch names (e.g. C♯4, B♭5 etc.) of events in 'piano-roll'-like music representations. Such algorithms are used in systems for computing scores from MIDI data. Also, knowing the letter-names of pitch events is indispensible in music information retrieval. For example, the repetition of a motif on a different degree of a scale (e.g., C-D-E-A repeated as E-F-G-E) might be perceptually significant even if the corresponding chromatic intervals in the patterns differ. Such matches can be found using fast, exact-matching algorithms if the pitch names are encoded in the input but not if the input is in a MIDI-like, 'piano-roll' format.

Aims

The aims of this paper are (1) to present a clear picture of the state-of-the-art in the field of pitch spelling algorithms; and (2) to introduce a new algorithm that performs consistently better than previous ones.

Method

Various datasets were compiled, each containing around 40000 notes and consisting of works in a particular genre by a particular composer, varying in style from baroque to jazz. Various pitch-spelling algorithms were run on these datasets and the results were analysed.

Results

All the algorithms correctly notated over 90% of the notes in the datasets. However, no algorithm scored 100% on any dataset. Longuet-Higgins's algorithm, developed in 1976, performed significantly better than some more recent algorithms. Overall, the author's new algorithm made the least number of mistakes.

Conclusions

No existing algorithm can compute the correct pitch name of every note in any sophisticated tonal style. The best algorithms attempt to model the way that pitch names depend on harmonic, voice-leading, rhythmic and motivic factors.

ISBN: 3-931852-66-0
ISSN: 1617-6847