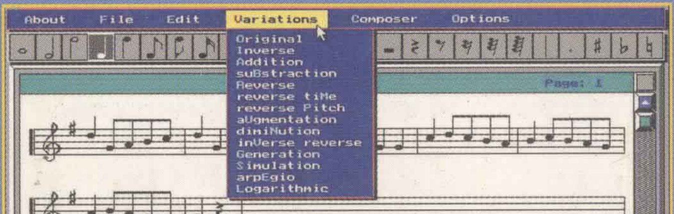
Hot Topics

LNCS 3310

Uffe Kock Wiil (Ed.)

Computer Music Modeling and Retrieval

Second International Symposium, CMMR 2004
Esbjerg, Denmark, May 2004
Revised Papers



with CD-ROM

Springer

Uffe Kock Wiil (Ed.)

Computer Music Modeling and Retrieval

Second International Symposium, CMMR 2004
Esbjerg, Denmark, May 26-29, 2004
Revised Papers

Springer

Volume Editor

Uffe Kock Wiil
Aalborg University Esbjerg
Department of Software and Media Technology
Niels Bohrs Vej 8, 6700 Esbjerg, Denmark
E-mail: ukwiil@cs.aue.auc.dk

Library of Congress Control Number: 2004118126

CR Subject Classification (1998): H.3, H.4, H.5, H.2, I.2, C.3

ISSN 0302-9743
ISBN 3-540-24458-1 Springer Berlin Heidelberg New York

Springer is a part of Springer Science+Business Media

springeronline.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Boller Mediendesign
Printed on acid-free paper SPIN: 11381006 06/3142 5 4 3 2 1 0

Lecture Notes in Computer Science 3310

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Preface

This volume contains the final proceedings for the 2004 Computer Music Modeling and Retrieval Symposium (CMMR 2004). This event was held during 26–29 May 2004 in Esbjerg, Denmark on the joint campus area of Aalborg University Esbjerg and the University of Southern Denmark, Esbjerg. CMMR is an annual event focusing on important aspects of computer music. CMMR 2004 is the second event in this series. CMMR 2003, which was held in Montpellier, France in May 2003, was a great success and attracted high-quality papers and prominent researchers from the field of computer music. The CMMR 2003 postsymposium proceedings was published by Springer in the Lecture Notes in Computer Science series (LNCS 2771). CMMR 2004 was jointly organized by Aalborg University Esbjerg in Denmark and LMA, CNRS, Marseille in France (in cooperation with ACM SIGWEB).

The use of computers in music is well established. CMMR 2004 provided a unique opportunity to meet and interact with peers concerned with the cross-influence of the technological and creative in computer music. The field of computer music is interdisciplinary by nature and closely related to a number of computer science and engineering areas such as information retrieval, programming, human computer interaction, digital libraries, hypermedia, artificial intelligence, acoustics, signal processing, etc. The event gathered many interesting people (researchers, educators, composers, performers, and others). There were many high-quality keynote and paper presentations, that fostered inspiring discussions. I hope that you find the work presented in these proceedings as interesting and exciting as I have.

First of all, I would like to thank the Program Chair Richard Kronland-Martinet for the very fruitful cooperation during the organization of this second event in the CMMR series. I would also like to thank my colleagues Laura Hyland, Stefania Serafin, and Lars Graugaard for their help in organizing the event. Finally, this volume would not have been possible without the help of Springer, Heidelberg. In particular, I would like to thank the computer science editor, Christine Günther, and the executive editor of the LNCS series, Alfred Hofmann.

October 2004 Uffe Kock Wiil

Organization

CMMR 2004 was jointly organized by Aalborg University Esbjerg, Denmark and LMA, CNRS, Marseille, France.

Symposium Chair

Uffe Kock Wiil (Aalborg University Esbjerg, Denmark)

Local Arrangements and Publicity

Uffe Kock Wiil (Aalborg University Esbjerg, Denmark)
Laura Hyland (Aalborg University Esbjerg, Denmark)
Stefania Serafin (Aalborg University Esbjerg, Denmark)
Lars Graugaard (Aalborg University Esbjerg, Denmark)

Program Committee

Chair: Richard Kronland-Martinet (LMA, CNRS, Marseille, France)

Members: Jens Arnspang (Aalborg University Esbjerg, Denmark)
Philippe Depalle (McGill University, Montreal, Canada)
Barry Eaglestone (University of Sheffield, UK)
Anders Friberg (Royal Institute of Technology, Sweden)
Goffredo Haus (University of Milan, Italy)
David L. Hicks (Aalborg University Esbjerg, Denmark)
Henkjan Honing (University of Amsterdam, The Netherlands)
Kristoffer Jensen (University of Copenhagen, Denmark)
Matti Karjalainen (Helsinki University of Technology, Finland)
Henrik Legind Larsen (Aalborg University Esbjerg, Denmark)
Brian Mayoh (University of Aarhus, Denmark)
Jocelyne Nanard (LIRMM, Montpellier, France)
Marc Nanard (LIRMM, Montpellier, France)
Peter J. Nürnberg (Aalborg University Esbjerg, Denmark)
François Pachet (Sony Research Lab, France)
Violaine Prince (LIRMM, Montpellier, France)
Esben Skovenborg (TC Electronic, Denmark)
Julius Orion Smith III (Stanford University, USA)
Leonello Tarabella (CNR, Pisa, Italy)
Daniel Teruggi (INA, Paris, France)
Hugues Vinet (IRCAM, Paris, France)
Gerhard Widmer (University of Vienna, Austria)
Sølvi Ystad (LMA, CNRS, Marseille, France)

Music Selection

Chair: Lars Graugaard (Aalborg University Esbjerg, Denmark)

Sponsoring Institutions

Aalborg University Esbjerg, Denmark
University of Southern Denmark, Esbjerg, Denmark
Vestjysk Musikkonservatorium (Esbjerg Music Conservatory), Denmark
Seminariets Uddannelsesfond, Esbjerg, Denmark
LMA, CNRS, Marseille, France

Table of Contents

Pitch, Melody Detection

Rhythm, Tempo, Beat

Music Generation, Knowledge

Music Performance, Rendering, Interface

Music Scores, Synchronization

Synthesis, Timbre, Musical Playing

Music Representation, Retrieval

Music Analysis

Separating Voices in Polyphonic Music: A Contig Mapping Approach

Elaine Chew and Xiaodan Wu

University of Southern California,
Viterbi School of Engineering, Integrated Media Systems Center,
Epstein Department of Industrial and Systems Engineering,
3715 McClintock Avenue GER240 MC:0193,
Los Angeles, California, USA
{echew, xiaodanw}@usc.edu

Abstract. Voice separation is a critical component of music information retrieval, music analysis and automated transcription systems. We present a contig mapping approach to voice separation based on perceptual principles. The algorithm runs in $O(n^2)$ time, uses only pitch height and event boundaries, and requires no user-defined parameters. The method segments a piece into contigs according to voice count, then reconnects fragments in adjacent contigs using a shortest distance strategy. The order of connection is by distance from maximal voice contigs, where the voice ordering is known. This contig-mapping algorithm has been implemented in VoSA, a Java-based voice separation analyzer software. The algorithm performed well when applied to J. S. Bach's *Two- and Three-Part Inventions* and the forty-eight Fugues from the *Well-Tempered Clavier*. We report an overall average fragment consistency of 99.75%, correct fragment connection rate of 94.50% and average voice consistency of 88.98%, metrics which we propose to measure voice separation performance.

1 Introduction

This paper presents an algorithm that separates voices in polyphonic music using basic principles of music perception and proposes metrics for evaluating the correctness of the machine-generated solutions. Creating music with multiple voices that are relatively independent is a compositional technique that results in auditory pleasure and has been practised for centuries in western music. This has led to a library of compositional rules that facilitate auditory streaming and the perception of multiple voices dating as far back as Palestrina (1526-1594) and as recently as Huron (2001, see [7]). In this paper, we use knowledge of the perceptual principles of auditory streaming to create an $O(n^2)$ contig mapping algorithm for separating polyphonic pieces into their component voices.

Distinct from audio source separation, voice separation is the determining of perceptible parts or voices from multiple concurrently sounding streams of music. The multiple streams can originate from the same source and also be of the

U.K. Wiil (Ed.): CMMR 2004, LNCS 3310, pp. 1–20, 2004.

same timbre. The contig mapping approach described in this paper considers only pitch height and event boundaries, ignoring information on timbre and sound source. Prior researchers (such as [8], [11] and [2]) have not reported any significant testing on large corpora because of the lack of methods for quantitative evaluation of voice separation results. We propose three metrics for quantifying the goodness of voice separation results and test the contig mapping algorithm on Johann Sebastian Bach's 15 *Two-Part Inventions*, 15 *Three-Part Inventions* and 48 Fugues from the *Well-Tempered Clavier.*

Computationally viable and robust methods for voice separation are critical to machine processing of music. Separating music into its component voices is necessary for notating music in separate staffs according to voice or instrument, or in the same staff with stems up or down depending on voice [8]. Another application related to music transcription is pitch spelling, the assignment of letter names to numeric representations for pitches or pitch classes (see for example, [3], [4] and [10]). The spelling of any given pitch is based on its tonal context as well as accepted voice leading principles. Voice separation is a precursor to incorporating voice leading spelling rules to any pitch spelling method.

Many applications in music information retrieval require the matching of *monophonic* queries to *polyphonic*[1] (or *homophonic*) databases, for example, query by humming applications. While other approaches to matching single line queries to multi-line records exist (see for example [9]), one approach made possible by voice separation is to first separate each piece into its component voices prior to matching the melodic query to now single-line records. Hence, a robust voice separation algorithm will vastly improve the hit rate of matching melodic queries to polyphonic databases. Another computational problem relevant to music information retrieval is the automatic detection and categorization of music by meter. Metric structure is most obvious in the lower voices and methods for meter induction can be improved by voice separation techniques.

The final example of a voice separation application is that of expressive performance. One of the main tasks of the performer or conductor is to determine the main melody or motif in any musical segment. The notes in the segment to be highlighted is often played louder or even a little before the others that are notated simultaneously in the score [6]. At other times, different voices are sounded at different volume levels to produce a foreground and background effect. Hence, machine models for voice separation are also essential to knowledge-based approaches to generating expressive performances.

As shown above, voice separation is a valuable tool in music information retrieval, automated transcription and computer analysis of music. One of the

[1] In traditional music literature, there exists a clear distinction between *polyphony* and *homophony*. Polyphonic music is multi-voice music where the voices exhibit independence relative to one another. Homophonic music, although also consisting of multiple voices, has one primary lead voice while other voices act as accompaniment to the main melody. In contrast, *heterophonic* music (less well defined) is music with one primary melody, and all accompanying voices embelishing with variants of the main theme.

easiest approaches to voice separation is to split voices according to some set of non-overlapping pitch ranges. According to [8], this is the method adopted by most commercial sequencer software packages. Needless to say, this method of separating voices can produce highly inaccurate and unsightly (in the case of automatic transcription) results. Various researchers have proposed ways to improve on this primitive approach.

In [11], Temperley proposed a preference rule approach to voice separation, incorporating the following rules for assigning voices to piano-roll representation of music: 1. avoid large leaps in any one stream; 2. minimize the number of streams; 3. minimize long breaks in streams; 4. avoid having more than one stream occupy a single square; and, 5. maintain a single top voice. Rules 1 through 4 were tested on four of Bach's fugues. Rule 5 was found to be necessary for handling classical music; rules 1 through 5 were tested on a few classical string quartets. The errors were analyzed in terms of the number of breaks, missed or incorrect collisions and misleads. Another rule-based approach was briefly described by Cambouropoulos in [2]. This method segments the input into beats then, within each beat, connects all onsets into streams by selecting the shortest path. The crossing of streams is disallowed and the number of streams is set to be equal to the number of notes in the largest chord.

In [8], Kilian and Hoos proposed a local optimization approach to voice separation. The piece was first partitioned into slices which can contain parts that overlap (in time) with other slices. Within each slice, the notes are then separated into voices by minimizing a cost function, which assigns penalty values for undesirable features such as, overlapping notes and large pitch intervals. One flexible feature of the Kilian and Hoos model is the ability to assign entire chords to one single voice. (The cost function penalizes chord tones that are spread too far apart.) The penalty values can be adjusted by the user to achieve different tradeoffs between the features. Their algorithm was tested on selected Bach *Chorales* and Chopin *Valses*, and Bartok's *Mikrokosmos*, and was found to be sensitive to the choice of penalty function parameters. For the purpose of automated transcription, the user can change the parameter values until a satisfactory result is achieved.

Like Temperley, our goal is to produce a correct analysis rather than an appropriate one for transcription, as is the case for Kilian and Hoos. In this paper, we propose three metrics to measure the correctness of a voice separation solution. They are: the average fragment consistency, the correct fragment connection rate and the average voice consistency. These metrics allow the algorithm's results to be quantified objectively. Unlike Kilian and Hoos' local optimization approach, our method does not allow synchronous notes to be part of the same voice. On the other hand, the contig mapping approach exhibits high fragment consistency, the grouping of notes from the same voice into the same fragments.

Both Temperley's preference rule approach as well as Kilian and Hoos' local optimization approach can protentially incur prohibitive computational costs if all possible solutions were enumerated and evaluated. Temperley utilized dynamic programming while Kilian and Hoos used a heuristically-guided stochastic local search procedure to avoid the exponential computational cost of exhaustive enumeration. In contrast, the contig mapping approach has an $O(n^2)$ performance and does not require approximation methods to compute a solution.

Distinct from previous approaches, our method hinges on one important feature of polyphonic music that has been ignored by other researchers. Because voices tend not to cross, when all voices are present, one can be certain of the voice ordering and assignment. We use these maximal voice segments as pillars of certainty out of which each voice connects to other members of its stream. This method requires no pre-assigned parameters or rule definitions. The perceptual rules are incorporated into the mathematical model and the algorithm has a guaranteed worst case performance of $O(n^2)$.

Section 2 describes the perceptual principles and the concepts underlying the contig mapping approach, and introduces the contig mapping algorithm. Section 3 presents additional details of the computer implementation of the algorithm and describes the VoSA (Voice Separation Analyzer) software. Section 4 presents our evaluation techniques and computational results. Finally, Section 5 outlines our conclusions and future work.

2 The Contig Mapping Approach

This section presents the contig mapping approach and its underlying perceptual principles. Section 2.1 outlines the auditory perceptual principles relevant to our approach, and Section 2.2 extracts from the principles and rules the assumptions underlying the contig mapping algorithm. Section 2.3 describes the contig mapping algorithm, including the segmentation procedure and the fragment connection policy.

2.1 Perceptual Principles for Voice Leading

In this section, we highlight the perceptual principles that are relevant to the contig mapping approach. Because the goal of the rules of voice leading is to create two or more concurrent yet distinct parts or voices, the same rules result in optimal auditory streaming. In [7], Huron reviews the perceptual principles for the organizing of auditory stimuli into streams and derives the rules of voice leading from these principles and empirical evidence.

The first is the pitch proximity principle. In the review, Huron reports that Bregman and his colleagues have gathered strong evidence for the pre-eminence of pitch proximity over trajectory in stream organization [1]. He argues that "the coherence of an auditory stream is maintained by close pitch proximity in successive tones within the stream," and that this principle holds true in the music across different cultures. Thus, in determining the connections between notes that are perceived to be from the same stream, proximity should be the guiding principle.

The second is the stream crossing principle. Humans have great difficulty in tracking streams of sounds that cross with respect to pitch. Huron reports the results of Deutsch [5] who showed that concurrent ascending and descending streams of the same timbre are perceived to switch directions at the point of crossing[2] as shown in the diagram on the right in Figure 1. Hence, a guiding principle in connecting notes in the same stream is that the streams should not cross.

Fig. 1. Possible interpretations of crossing streams.

These perceptual principles lead to numerous traditional and non-traditional rules for writing polyphonic music with perceptibly distinct parts. The ones relevant related to the pitch proximity principle are (following Huron's numbering system):

[D6.] Avoid Unisons Rule. *Avoid shared pitches between voices.*
D10. Common Tone Rule. *Pitch-classes common to successive sonorities are best retained as a single pitch that remains in the same voice.*
D11. Conjunct Movement Rule. *If a voice cannot retain the same pitch, it should preferably move by step.*
C3. Avoid Leaps Rule. *Avoid wide pitch leaps.*
D13. Nearest Chordal Tone Rule. *Parts should connect to the nearest chordal tone in the next sonority.*
[D18.] Oblique Approach to Fused Intervals Rule. *When approaching unisons, octaves, or fifths, it is best to retain the same pitch in one of the voices.*
[D19.] Avoid Disjunct Approach to Fused Intervals Rule. *If it is not possible to approach unisons, octaves and fifths by retaining the same pitch, step motion should be used.*

while D6, D14 and D15 are encapsulated in the stream crossing principle:

[D6.] Avoid Unisons Rule. *Avoid shared pitches between voices.*
D14. Part-Crossing Rule. *Avoid the crossing of parts with respect to pitch.*

[2] A simple and informal experiment conducted on March 4th in a class of 14 students showed that this result held true even when the ascending and descending streams were played using the rhythm of the Christmas carol "Joy to the World," where the opening melody is essentially a descending scale embellished with temporal variation. This perceptual principle is so strong that it overrode the perception of the well-known melody.

D15. Pitch Overlapping Rule. *Avoid "overlapped" parts in which a pitch in an ostensibly lower voice is higher than the subsequent pitch in an ostensibly higher voice.*

2.2 The Assumptions and Underlying Concept

For the purpose of the contig mapping algorithm, we translate the rules and perceptual principles detailed in Section 2.1 to the following assumptions:

1. By definition, each voice can only sound at most one note at any given time.
2. All the voices will sound synchronously at some time (we use this as a baseline count of the total number of voices present in the piece.)
3. *Pitch Proximity:* intervals are minimized between successive notes in the same stream or voice.
4. *Stream Crossing:* voices tend not to cross.

The contig mapping approach derives its method directly from these assumptions. Assumptions 1, 2 and 4 imply that, at certain segments of time, all voices will sound synchronously in a well-bahaved manner. In these segments, which we call *maximal voice contigs*, we can be certain of the voice assignments for each note. Based on assumptions 3 and 4, we can use distance minimizing procedures to connect voices between segments. The maximal voice contigs seed the connection process: they act as the pillars out of which voice assignments grow at each iteration of our procedure.

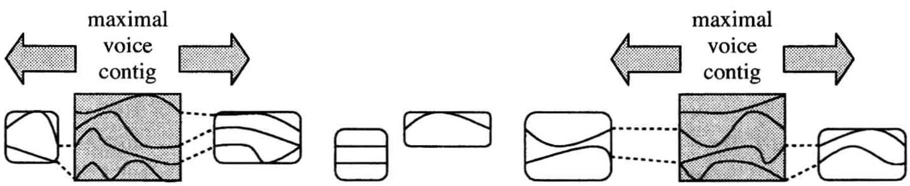


Fig. 2. Minimum distance voice connections grow out from the maximal voice contigs

2.3 The Algorithm

We have outlined the principles and concept behind our contig mapping approach in the previous sections. In this section, we shall provide the algorithmic details for its systematic implementation, including the procedures for segmentation and connection.

Before embarking on a description of the algorithm, we first introduce the terminology used in this section. A *note* is a musical entity with pitch and duration properties. A *fragment* is a sequence of successive notes that belong to the same voice. A *contig*[3] is a collection of overlapping fragments such that the overlap depth (number of fragments present) at any time is constant. A *maximal voice contig* is a contig with the maximum number of voices present. Examples of a fragment, contig and maximal voice contig are shown in Figure 4, which corresponds to bars 24 and 25 of Bach's Three-Part Invention (Sinfonia) No. 13 (shown in Figure 3.) In this case, both the first and last contigs are maximal voice contigs.

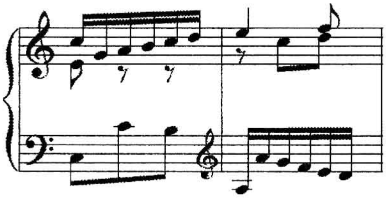


Fig. 3. Measures 24 and 25 of Bach's Three-Part Invention No.13.

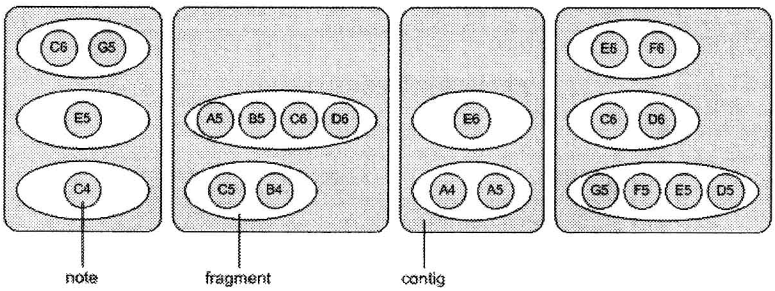


Fig. 4. Terminology

Segmentation Procedure The piece is segmented according to voice count. The segmentation procedure is best illustrated by example. The final outcome is a segmentation of the original piece into contigs such that the voice count remains constant within the contig. We return to the Bach Three-Part Invention example shown in Figure 3. Figure 5(a) shows a piano roll representation of the same

[3] The term *contig* is borrowed from the field of computational biology where, in DNA sequencing, the shotgun sequencing method utilizes computer algorithms to connect ordered sets of overlapping clones of DNA fragments in order to determine the DNA sequence.