Musiplectics: Computational Assessment of the Complexity of Music Scores

Ethan G. Holder

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Masters of Science
in
Computer Science

Eli Tilevich, Chair
Amy Gillick
R. Ben Knapp

April 23, 2015
Blacksburg, Virginia

Keywords: Music Scores; Music Complexity Assessment; Novel Computing Domains; MusicXML

Musiplectics: Computational Assessment of the Complexity of Music Scores

Ethan G. Holder

(ABSTRACT)

In the Western classical tradition, musicians play music from notated sheet music, called a score. When playing music from a score, a musician translates its visual symbols into sequences of instrument-specific physical motions. Hence, a music score's overall complexity represents a sum of the cognitive and mechanical acuity required for its performance. For a given instrument, different notes, intervals, articulations, dynamics, key signatures, and tempo represent dissimilar levels of difficulty, which vary depending on the performer's proficiency. Individual musicians embrace this tenet, but may disagree about the degrees of difficulty.

This paper introduces *musiplectics*[1], a systematic and objective approach to computational assessment of the complexity of a music score for any instrument. Musiplectics defines computing paradigms for automatically and accurately calculating the complexity of playing a music score on a given instrument. The core concept codifies a two-phase process. First, music experts rank the relative difficulty of individual musical components (e.g., notes, intervals, dynamics, etc.) for different playing proficiencies and instruments. Second, a computing engine automatically applies this ranking to music scores and calculates their respective complexity. As a proof of concept of musiplectics, we present an automated, Web-based application called Musical Complexity Scoring (MCS) for music educators and performers. Musiplectics can engender the creation of practical computing tools for objective and expeditious assessment of a music score's suitability for the abilities of intended performers.

This thesis is based on research submitted for publication at ONWARD'15.

That this work received support from a Virginia Tech ICAT SEED grant.

[1]musiplectics = music + plectics, Greek for the study of complexity

Dedication

Acknowledgments

This thesis would not have been possible without the generous help of many people. First and foremost, I would like to thank my research advisor, Dr. Eli Tilevich, for taking a chance on me as a student, guiding me through the publication process, and supporting me throughout my studies. I owe a great deal of thanks to Dr. Cliff Shaffer, a dedicated professor and reference, who has encouraged me to get involved in research in only my first semester as an undergraduate. I must also thank the numerous professionals that advocated the research process as both an undergraduate and graduate student, including Mr. Mack McGhee, Mrs. Andrea Bailey, and Mr. Adrian Rodriguez.

I generously thank my fellow researchers, Eeshan Shah and Mohammed Davoodi, without whom much of this research would not have been possible. I must express significant thanks to all of the other various professors and advisors who have worked with me over my years at Virginia Tech. Among all of these, I would be remiss if I did not expressly thank Mrs. Terry Arthur. Your guidance from the very first time I met you, before I even began classes, all the way through until my end as an undergraduate has been outstanding in every way.

Last but not least, I must thank all of my family and friends who pushed me to achieve as much as I have. I especially thank my mother, Laurel Holder, and father, Mark Holder, for instilling a love of learning in me from a very young age and for constantly encouraging me even when I had stumbled along my way. I owe it all to you, and I can never thank you enough.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Which piano concerto is more difficult: Rachmaninoffs Second or Third? A newly appointed band director wonders if this new orchestral score is appropriate for a high school band, given that the clarinet and bassoon sections are quite advanced, while the flute and oboe sections are more novice. Music educators working on pedagogical guidelines for K-12 students are trying to decide whether a given piece belongs in the N or N+1 curricular level. A publisher wonders which audience to target when marketing new works, while the publisher's customers face great uncertainty when determining whether unfamiliar music matches their playing ability. Performers, band directors, educators, and publishers encounter these non-trivial questions throughout their professional careers.

Unfortunately, determining the relative complexity of music is a non-trivial cognitive task. Additionally, methods in the current state of the art depend solely on individual opinions, a process influenced by personal biases and lacking common criteria. In other words, the only way to answer these questions in a viable way is to carefully analyze music scores by hand, a tedious, error-prone, and time-consuming process. The stakeholders at hand would rather spend their precious time on more creative pursuits.

Can computing help decode these persistent and challenging questions? Is it possible to provide such technology in a ubiquitous and user-friendly way, accessible to any interested musician? To answer these questions, this paper presents musiplectics, a new computational paradigm, that systematically evaluates the relative difficulty of music scores, thus benefiting educators and performers. Two insights provide a foundation behind musiplectics. First, certain notes and other musical components, including intervals, dynamics, and articulations, are harder to play than the others. Second, automated computer processing can transform a prohibitively tedious, error-prone, and subjective process into a practical and pragmatic solution if exposed via an intuitive user interface. Hence, musiplectics fuses commonly accepted music tenets and novel computing paradigms, to objectively answer the questions above.

Figure 1.1: The process of decomposing code and music to extrapolate a conclusion.