Available online at www.sciencedirect.com

ScienceDirect

Artificial Intelligence 170 (2006) 1218–1221

Artificial Intelligence

www.elsevier.com/locate/artint

Book review

David Cope, Computer Models of Musical Creativity, MIT Press, 2005.

Roger B. Dannenberg

School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, USA

Available online 1 November 2006

Music and computers are natural partners. For centuries, music theory has been closely aligned with (and an important source of inspiration for) science and mathematics. Music compositions, however, are too large and complex to model mathematically, at least in practice. For the most part, science and mathematics diverged from music composition and developed independently—but then computers came along. As early as 1955, Lejaren Hiller and Leonard Isaacson composed the "Illiac Suite", one of the first pieces of music to be written by computer. In spite of this promising beginning, Computer Music as a field has been dominated by studies of sound synthesis and control, leaving the composition up to humans. However, the idea of automated composition has attracted a steady stream of composers and researchers over the years.

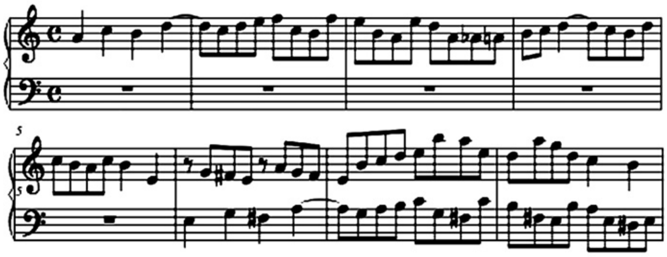
One branch of this work on automated music composition focuses on "algorithmic composition", where the emphasis is on abstract sequence and pattern generation that happens to make interesting music. So far, this approach has had the greatest impact on music, but most computer scientists and at least some composers find this disappointing. With powerful computers and AI techniques, it seems that we should be able to build sophisticated models of the music composition process. This requires not only sophisticated representations of music and music structure, but also a model and approach to musical creativity. For many musicians *creativity* lies at the core of music making and is therefore the most human of skills.

Computer Models of Musical Creativity describes its author's many-faceted approach to music composition by computer, emphasizing the nature of creativity, how it can be modeled, and how these models are applied. I find this book especially interesting because David Cope is one of the few composer/researchers to attempt to model traditional musical styles of composers such as Bach, Mozart, and Beethoven. (See the figure.) While the musical quality of his computer-generated works has been the subject of much debate, the fact that there is any debate at all is quite an accomplishment. Just as computers and AI have raised many difficult questions about the nature of intelligence, Cope's work and this book raise questions about "creativity"—where does it come from?, can computers be creative?, and is creativity different from intelligence? While most work in computer generated music ultimately relies on human creativity or possibly serendipitous discovery filtered through human perception and selection, Cope's work is much more concerned with the construction of creative processes.

Part I of the book covers definitions, background, and current models of musical creativity. This is not an exhaustive review, but it covers a wide range of interesting topics relating to machine creativity including the difference between music and language, definitions of creativity, originality vs. creativity, and tests for creativity. Cope introduces the reader to many theories on creativity, ranging from Freud, who viewed creativity as a "defense mechanism and an unconscious protection against unpleasant thoughts" (p. 40), to Chris Thorton's recent proposal that creativity is a form of "run-away learning" in which objectivity is relaxed. Cope rejects the notion that creativity results in entirely original ideas or work; thus, "originality" is different from "creativity". Instead, Cope defines creativity as "The initialization of connections between two or more multifaceted things, ideas, or phenomena hitherto not otherwise

E-mail address: rbd@cs.cmu.edu (R.B. Dannenberg).

doi:10.1016/j.artint.2006.10.004



Beginning of a fugue in the style of Bach from Cope's *The Well-Programmed Clavier* (2002).

considered actively connected". This implies that creativity works in some pre-existing context and recognizes or exploits new relationships.

Much of the book is devoted to the clarification and elaboration of this definition. Typically, creative work extends or reinterprets what came before, and this idea forms a basis for machine implementations of creativity. Cope believes creativity is distinct from intelligence, and he says that his earlier work in replicating the styles of various composers does *not* require creativity in spite of the fact that many reviewers have credited this work as creative. In this book, Cope introduces programs that use a combination of new techniques to model creativity.

Part II begins an exploration of various aspects of music composition programs. As one might expect, complex creative behavior arises through the combination and interaction of many processes rather than any one "magic" technique. An important foundation for much of Cope's work is "recombinance", a method for decomposing existing music into fragments that can be reassembled into new works. The basic idea is related to Markov models, restricting note-to-note transitions to ones that some human composer has actually written. This approach maintains local context so that musical voice leadings are at least locally valid. If the musical materials are carefully selected, say, from a small set of Mozart sonatas, a new Mozart-like piece will be generated. This basic idea is refined by preventing the fragmentation of "musical signatures", which represent important indicators of style, and by incorporating structural or contextual information so that generated music will have a stronger sense of direction.

Another ingredient of Cope's work is *allusion*, ranging from common patterns to outright quotation. Cope describes a program, *Sorcerer*, that discovers allusions in music. In one example, Sorcerer uncovers similarity between themes from Beethoven's Ninth Symphony and Bach's "Jesu, Joy of Man's Desiring". Sorcerer works by coding music as n-grams of pitch intervals and searching for commonalities (both exact matches and near-misses) among pieces. Other representations, including harmony and rhythm, can be substituted to compare music along non-melodic dimensions.

Learning, inference, and *analogy* also play a role in creativity, and programs are developed that display these skills. The techniques are fairly simplistic, but it is an intriguing idea that simple means can sometimes produce impressive results. As an example of a musical analogy, Cope suggests Bach's Prelude no. 4 and Fugue no. 4 from *The Well-Tempered Clavier*. The prelude begins with a descending scale followed by a leap up to another descending scale. The Fugue begins with just one step down followed by a leap up to another step down. The analogy is that a scale down is to a step down as a leap-followed-by-scale-down is to a leap-followed-by-step-down.

One of the great challenges of any music generation is that music must make sense at many time scales, from the level of individual notes and chords up to the organization of movements within an entire symphony. Cope describes a grammar for music called SPEAC, in which note groups are labeled as S(tatement), P(reparation), E(xtension), A(ntecedent), or C(onsequent). The main use of SPEAC is to compose overall musical form and structure as opposed to conducting a random walk from one local context to the next. Another way to produce logical structure is to "lift" it from other work. While one might call this another form of analogy, Cope uses the term "influence". He introduces a program, *Serendipity*, that collects music from the Internet. The program then incorporates elements of that music into compositions. The details of how a program might narrow the scope of music considered, determine the dimensions of influence, and apply them in a composition are not given.

Part III, "An Integrated Model of Musical Creativity" explores a model of unsupervised learning called *Association Networks*. To simplify the presentation, a natural language association network is first described. An association network is a fully connected directed graph where nodes are words and connections hold weights that are updated according to a learning scheme. The learning scheme increments weights based on the input that is observed through a dialog with the user. The normal output of the network is determined by following the highest weighted connection from node to node, subject to some rules for well-formed output. However, the user can request "inductive association" in which the next-to-highest weight is followed. This simple mechanism can produce interesting and creative output, and illustrates how semantic knowledge can be captured and represented in terms of connection weights. Essentially the same network used for natural language can be applied to music data, allowing programs to learn to construct logical musical sequences and to apply sensible substitutions by association.

Cope's goal is to integrate many of the previously described techniques. In particular, the association network is used to learn from examples presented by the user, but music generation also relies on recombination, pattern matching of allusions, SPEAC, and influence on form and other aspects from other music. The description of this integration lacks detail, so it is difficult to understand how these elements actually combine in practice and how much of the integration is automated. The program is available as source code, but even the author warns that "*teaching* language and music to an association network requires an enormous commitment of time and energy".

The final chapter, "Aesthetics", is an interesting change of pace after many chapters that cover Cope's technical work. There has been quite a bit of controversy over Cope's work, and in this chapter, Cope shares some of his frustration with reviewers and critics. He also answers some of the criticism he has received. For example, computer compositions can be performed by humans playing acoustic instruments (or voice), giving the music the benefit of human interpretation and performance nuance. At least one critic ascribed the musicality of these performances to the human performers. On the other hand, when Cope removed the human element and recorded a computer-controlled piano, a different reviewer complained that the music lacked human expression.

Cope complains that people rarely listen to computer music in the same way they would listen to a human composer. "For them, computer-created music represents more of a philosophical challenge than an aesthetic experience". There is also the question of the *intention* of computer-generated music. Is it some sort of test or evaluation? Is it a threat or challenge issued to humans as in a chess competition? Is the intention relevant? To Cope, the music is just new music he has created in known musical styles with the help of a machine. He would like listeners to hear the music from that perspective, but this might be asking too much. After all, every good composer develops a distinctive style (or styles) for good reason. Who wants to compete with Mozart in his own territory? It is hard to ignore the fact that Cope's computer-composed works adopt their style from human composers.

It should be said that David Cope is not a computer scientist, and readers should not expect a traditional computer science treatment of his work. For example, Cope is often concerned with efficiency, especially in pattern matching, but never analyzes the complexity of his matching algorithms. He codes some pattern data into Lisp symbol names to take advantage of fast symbol name lookup, but fails to mention the role of hashing and the relationship to n-grams. In some cases, Cope may have minimized more technical descriptions to be accessible to a wider audience, and in most cases, computer scientists will be able to map Cope's descriptions to more familiar terminology and paradigms. All of Cope's programs and examples are written in Lisp, and some basic knowledge of Lisp is helpful, at least in a few places.

On the other hand, Cope is without question a composer and music theorist. Readers should have a background in music and a college-level understanding of harmony and counterpoint in order to study all of the algorithms presented in the book. The book is full of music notation including a number of computer compositions. I do not want to discourage readers with less musical background. The compositions can be obtained in audio form from the web, and there are also very simple music examples and some non-music examples that can be understood without a formal music education. There is a lot of interesting material here even for non-musicians, but certainly those with a musical background will be able to understand the book more completely.

Readers will learn much about Cope's approach to computer-based music composition, but they may be disappointed that they did not learn more. Often, very simple techniques are described and illustrated to introduce a concept such as SPEAC. Then, a complete composition will be presented, leaving the reader to puzzle out how the simple example could be scaled up. Cope mentions that many of his programs require careful manual selection and preparation of data. It seems clear that many of Cope's techniques are fairly fragile and sensitive to parameter choices. For example, most machine learning systems improve with more training data, but some of Cope's programs degrade when there is too much example material. It is not clear how much of the creativity of these programs can be attributed to careful human guidance. Probably, more extensive examples and more complete descriptions of typical input and output data would help the reader to better understand these programs. Readers are invited to obtain and study source code from the Web. These typically consist of 2,000 to 5,000 lines of Lisp.

In spite of these shortcomings, David Cope's book offers new insight into some of the most successful programs for composing music in the style of other composers as well as creative music generation. This work towers above most other attempts and sets a very high standard for anyone venturing into this area. Cope's book is an important guide to this body of work. The book strives to present models of creative thought in a down-to-earth manner including simple examples. Any artist working with computers is bound to find this work inspiring. Perhaps this book will also inspire Artificial Intelligence researchers to conduct further study into the nature of creativity.