Title: Computational creativity in music through conceptual blending

Authors: Max, Ewen, Emilios, Alan (?)

Abstract

Not yet.

1 Introduction

Overview of creative systems, referencing approaches that have hitherto been used. One or two paragraphs about Boden's categorisation of creativity, preparing the ground for shortly describing combinational creativity and conceptual blending. Emphasise that this paper describes actual implementations of blending examples in music, that indicate the creativity potential of the utilised algorithms. Emilios and Max.

2 Conceptual blending: from theory to a creative generative system

General overview of Fauconier and Turner's approach and the fact that it has primarily been used for analysing creative content rather than creating new things. Goguen's category theory proposal. Max and Emilios.

2.1 A formal model for conceptual blending

The COINVENT formal model. Max.

2.2 Implementation of the formal model

Algorithmic and technical things about the "COBBLE". Ewen and Max.

3 Musical creativity through conceptual blending in practice

Overview of the examples that follow. Highlight the fact that these examples cover a wide range of music topics. Max.

3.1 Blending chord transitions for extending the creative capabilities of melodic harmonisation

Refer to the optimised optimality principles for distinguishing the good blends. Also present cadence blending through transition blending. Give a short description of extending Markov transition tables for solving problems like tonality modulations. Include the indications obtained by the subjective experiments on cadence blending. Max and Emilios. Is the MATLAB implementation enough? If yes, we can begin writing text right away.

3.2 Blending musical form: the sonata rondo example

A paragraph describing sonata rondo form and the fact that this form is indeed a conceptual blend invented by humas. Presentation of a gramar formalism for the sonata rondo form. Allan. Do we need a COBBLE implementation on that, or we can simply refer to the methodology?

3.3 Extending musical creativity through cross-domain blending

A paragraph describing that humans create metaphors and blends all the time – a fact that makes them more creative. Metaphors: people relate geometric elements with sound, a fact that allows them to descrive audio events intuitively (e.g. pitch height with geometric height – Antovic). Additionally, people concepts from alien conceptual spaces and inject them into musical settings, pushing their creativity to unknown territories (e.g. Coltrain changes). Ewen.

3.3.1 Blending group theory with chord progressions

The methodology we present here for conceptual blending exploits the power of signature morphisms, meaning input theories can be from very different domains. As an example of cross-domain blending we consider here a simple blend of chord and cadence theory, with a theory of cyclical groups from mathematics.

Cyclical Groups   In mathematics a group is characterised by a set of objects, with a binary operation obeying the following rules:

$$a$$
$$b$$
$$c$$
$$d$$

A theory of Chords

spec RELCHORD =
    SYMBOLS
then
    sort   *RelChord*

    pred *hasRelNote : RelChord × Note*
    ops
        thirdrel : RelChord →? *Modifier*;
        bassrel : RelChord → *Note*;
        seventhrel : RelChord →? *Modifier*;
        sixthrel : RelChord →? *Modifier*;
        fifthrel : RelChord →? *Modifier*

    ∀ *c : RelChord*; *n : Note*
    • *bassrel(c) = n* ⇒ *hasRelNote(c, n)*
    • *thirdrel(c) = minor* ⇒ *hasRelNote(c, 3)*
    • *thirdrel(c) = major* ⇒ *hasRelNote(c, 4)*
    • *seventhrel(c) = minor* ⇒ *hasRelNote(c, x)*
    • *seventhrel(c) = major* ⇒ *hasRelNote(c, x1)*
    • *sixthrel(c) = minor* ⇒ *hasRelNote(c, 8)*
    • *sixthrel(c) = major* ⇒ *hasRelNote(c, 9)*
    • *fifthrel(c) = perfect* ⇒ *hasRelNote(c, 7)*
    • *fifthrel(c) = diminished* ⇒ *hasRelNote(c, 6)*
    • *fifthrel(c) = augmented* ⇒ *hasRelNote(c, 8)*
end

spec SYMBOLS =

    free type
    Note
    ::= 0 | 1 | 2 | 3 | 4 | 5
    | 6 | 7 | 8 | 9 | *x* | *x1*
    free type
    Modifier
    ::= *major*
    | *minor*
    | *diminished*
    | *perfect*
    | *augmented*

spec ABSCHORD =
    RELCHORD
then
    sort   *AbsChord < RelChord*
    pred *hasAbsNote : AbsChord × Note*
    op
        root : AbsChord → *Note*
    ∀ *c : AbsChord*; *n : Note*
    • *hasRelNote(c, n)* ⇔ *hasAbsNote(c, root(c) + n)*
end

Figure 1: CASL specifications for Relative and Absolute Chords

A theory of Cadences

A theory of Progressions We model progressions as lists of Chords or Cadences.

Blending progressions with Cyclical Groups Talk about search process....

3.3.2 Blending geometry with pitch motion

Also refer to Xenakis's composition. Present parallel voice leading blend. Ewen, Max, Alan.

4 Conclusions

Emilios and Max.