Automatic Jazz Accompaniment Computation : An Open Advice-Based Approach.

Mira Balaban and Stanislav Irish
Ben-Gurion University of the Negev
POB 653 84105 Beer-Sheva, Israel.

mira@cs.bgu.ac.il irish@cs.bgu.ac.il

Abstract

Music systems are based on always partial, sometimes default, knowledge. In order to meet this essential character, we need *open systems* that can constantly adapt their music assumptions. Ideally, the system should provide the naive end-user with the ability to manipulate the music knowledge on a user-understandable level. This paper presents a computer tool, entitled JOINME, that composes a jazz accompaniment consisting of drums, bass and piano lines. JOINME is an advice propagation system, that separates the music knowledge from the actual accompaniment computation. The system supports changes/revisions of its musical knowledge on the user interface level. Its open architecture enables a close, smooth collaboration with jazz experts.

1 Introduction

Music systems, generally, are based on always partial, sometimes default, knowledge. In order to meet this essential character, we need *open systems* that can constantly adapt their music assumptions. Ideally, the system should provide the naive end-user with the ability to manipulate the music knowledge on a user-understandable level. The parameters control, as in MAX and DMIX, provides this service on the performance level. Nevertheless, controlling the music knowledge is more complicated.

Automatic accompaniment in jazz is an attractive target for computer tools due to its relatively organized structure and content. Several commercial and research tools exist : Cybernetic Composer by Charles Ames [C. Ames, 1992] , NeurSwing by Denis Baggi [Baggi, 1992], Band-In-A-Box by PGmusic, etc. The commercial tools, by their nature, are “black box” tools that provide specific services to their users. The user has no impact on the music knowledge that determines the operation of the system.

In this paper we describe **JOINME** (*Jazz-Oriented Music Environment*) system, a jazz accompaniment tool with an open architecture. JOINME

is an advice propagation system, which novelty lies in employing no global algorithm for accompaniment computation. The accompaniment consists of three instrument lines for drums, bass and piano and is constructed on the basis of an input chord progression. In addition, the system also supports real-time parameter changes, via its graphic user interface and MIDI foot switch.

The music knowledge is structured as a classified collection of local advices, and the computation of the accompaniment is driven by their propagation. Local advices have patterns based on their kinds, and they can be manipulated by the user. In the current stage of the system, the advice patterns are still written in C++ (which is the implementation language of JOINME). A future, markable step will involve the design of a user-level language that can support these patterns. In the planned user language advices will be expressed, solely, in terms of music parameters, properties, operations and relations.

2 JOINME Architecture

The architecture of JOINME is shown in Figure 1.

2.1 Input Data and Parser

The input for the system is a chord sequence, describing the harmony and melodic structure of the song to be played, along with its tempo and meter. These data are read from the input file by the *Parser*. The parser interprets the input consulting the *Music Knowledge Base*.

2.2 Music Knowledge Base

JOINME constructs a jazz accompaniment as a result of interaction between its Music Knowledge Base and input data. Hence, the the knowledge base has ultimate impact on the system design and operation.

JOINME benefits from the Object-Oriented approach, using natural hierarchy and inheritance for

Figure 1: JOINME architecture.

Figure 2: The Structure of Rhythmic Point.

the class construction of music concepts. For example, a scale is identified by its interval structure and tonic, and a chord - by its root and inner interval structure. These common features can be organized into base classes, while all other music concepts of the same type are derived from them. The books by J. Mehegan [Mehegan, 1984] were taken as the main source for the music knowledge of JOINME.

The Music Knowledge Base logically consists of two components : *Tonal Music Harmony* and *Jazz Knowledge Base*. The first component describes general tonal music concepts, as follows :

- Notes and Intervals. These are abstracted into an arithmetic system of *primitive elements* that capture numerical computations with diatonic and chromatic properties of notes and intervals. Following [Balaban, 1983], the primitive

elements form an additive, commutative (algebraic) group. Computations such as *note + interval*, *note1 - note2*, *interval1 + interval2* are performed by the primitive element abstraction.

- Scales : General Scale, Major and Minor Scales. Methods that account for the relationships between notes and intervals are provided.
- Chords : General Chord, Major and Minor Triads. Methods that account for the relationships between chords and scales are provided.

The jazz component is *derived* from the tonal music component, using its general concepts, such as scale and chord to construct specialized jazz scales and chords. It includes the following jazz music concepts :

- Scales : Dorian Mode, Phrygian Mode, Lydian Mode, Mixolydian Mode, Locrian Mode, Diminished Scale.
- Chords : Major Seventh Chord, Minor Seventh Chord, Dominant Chord, Diminished Chord, Half-diminished Chord.
- Rhythmic Points - A rhythmic point is a slice of time in a musical score : it has a score position, duration and a value. Finer distinctions between rhythmic points depend on the score position, melodic functionality and instrument. The value can be either a note or a chord. The common structure and the information content of rhythmic points is captured within a class *RhythmicPoint* that includes, in addition, the harmonic context, the possible values and advices. There are advices that determine the possible values (Domain Advices) and there are advices that constraint allowed values (Composition Advices); see Figure 2.
- Rhythmic Structures - A rhythmic structure is a sequence of rhythmic points. The rhythmic structures of the instruments differ in the kind of rhythmic points they use. JOINME uses a fixed set of rhythmic structures, which are called rhythmic patterns. There are rhythmic patterns for Drums, Bass and Piano. Drums use predefined music pieces (pitch + duration) called *music patterns*.

The rhythmic and music patterns are associated with weights (natural numbers in the range [0, 100]) that describe their relative “popularity” within an instrument line.

2.3 JOINME Acoustic Interface

The acoustic interface includes a MIDI Manager, MIDI Interface (MIDI Sequencer), Synthesizer Keyboard, MIDI Foot Switch and Audio Stereo Equipment. The MIDI interface in JOINME is based on the “raw” version of SGI (Silicon Graphics) MIDI library and uses SGI MIDI manager to play time-stamped MIDI events on the keyboard. The MIDI sequencer allows for a higher-level manipulation of MIDI events and a MIDI file option provides portability of the music results making their future processing possible.

3 Computation

Automatic jazz accompaniment tools that we are aware of cannot support partiality and inconsistency of available knowledge in music. These tools have their composing algorithms programmed once and for all; all their music knowledge is “buried inside” them,

being inaccessible from the outside, and unrecognizable as a separate, logical unit. Consequently their knowledge is not extendible or modifiable. JOINME is designed to overcome these problems by providing a user with an open architecture framework.

3.1 The Accompaniment Ontology

The accompaniment consists of three instrument lines for drums, bass and piano. The instrument line classes are *DrumsLine*, *BassLine* and *PianoLine*. An instrument line object has a rhythmic structure, which can be viewed as a measure organization of rhythmic points. An instrument line classifies the rhythmic points by their harmonic context and score position, and triggers the computation of their values. The computation itself has three stages :

1. The domain advices of rhythmic points yield the possible values for each point.
2. The composition advices are triggered and the computation proceeds by self propagation.
3. The final result is then translated into a MIDI track.

It should be emphasized that all music knowledge, used in the accompaniment computation, is folded within the various rhythmic point advices. The three stages serve only as a framework of the computation.

A rhythmic structure attribute of an instrument line for Bass and Piano is an object of the *Rhythmic Structure* class. It is constructed by concatenation of rhythmic patterns from the Jazz KB. The construction is directed by their relative weights. The rest of the computation is completed by the process of advice propagation, which is initiated after the construction of the rhythmic structure.

3.2 Advices

The advices are attached to the rhythmic points. They are implemented as methods (member functions) of various rhythmic point classes. There are two types of advices : *Domain* and *Composition Advices*.

3.2.1 Domain Advices

Domain advices select possible values for rhythmic points. The selection depends on the type of rhythmic points. There are, essentially, two kinds of domain advices : Advices that select a range of values and advices that add a single value. The selection of the values is based on the harmonic context of a rhythmic point and its musical parameters. For example, a domain advice for Bass rhythmic points that appear on the first beat, may allow only the root of their harmonic context (at various octaves), as a set of possible values.

3.2.2 Composition Advice

Composition Advices provide constraints on rhythmic point values. A standard composition advice tests a suggested possible value and either rejects or accepts it. Typically, composition advices take additional parameters, such as the neighboring rhythmic points. The advice tests whether its constraint is satisfied by the suggested possible value with respect to already selected value(s) of the neighboring rhythmic points. Note, that a composition advice for a rhythmic point R is triggered by one of its neighboring rhythmic points upon instantiation. Hence, it is always the case that when a composition advice at R is applied, one of R 's neighbors is already instantiated (has a selected value).

3.2.3 The Process of Advice Propagation

The actual notes in an instrument line (Bass or Piano) are selected by the advice propagation process. The process operates on an already built rhythmic structure. It involves self-triggering of advices for rhythmic points. When a rhythmic point is instantiated, it triggers the neighboring RP's to select values (based on their local advices). An instantiation of an RP fails when its advices reject all possible values. In that case a backtracking process is initiated. The process terminates once all RP's are instantiated. In the current implementation the propagation proceeds from the first RP to the last.

The main advantage of the advice-based approach and the JOINME open architecture lies in the clear separation between the actual process of an instrument line construction and the music knowledge manipulation. The construction process is invariant to changes in the of domain and composition advices. The user can determine the quality of instrument lines by direct manipulation of advices. Hence, each set of advices for rhythmic points represents an accompaniment "theory" (version).

4 GUI

The Graphic User interface (GUI) of the system provides the user with a quick and convenient access to most important JOINME features. The GUI can be logically divided into the following components :

- Input of Accompaniment Parameters - The user inputs parameters such as *tempo*, *meter* and *chord progression*. This GUI component is also used to select music patterns for Drums line and rhythmic patterns for Bass and Piano lines.
- Accompaniment Construction - Initiate computation for a given input. All instrument lines can be built or recomputed at once, or alternatively, instrument lines can be computed separately.

- Accompaniment Performance - Play accompaniment back on a synthesizer keyboard. All instrument lines can be played simultaneously or individually. The accompaniment can be started or interrupted remotely, via the MIDI foot switch, attached to the synthesizer.
- Real-Time Interaction with the User - During the performance of the accompaniment, the user can control with sliders the tempo and the volume of instrument lines in real time.
- Manipulation of the Advice Set - The GUI supports access to the advice set. It provides features for advice set revision, creation a new advice set, selection of existing advice sets, and removal of old advices.
- Saving Constructed Accompaniment - An accompaniment can be saved in two different file formats : as an ASCII version of a MIDI file or as a Standard MIDI File (binary). The latter option provides portability, since the MIDI file can be further processed by an off-the-shelf MIDI application, such as a MIDI sequencer. This way the user can benefit from additional visual services, e.g. notation, real-time score scanning and performance control.

5 Conclusions

A major future step will involve the development of a *user advice language* on the basis of current advice patterns. As a first step, we intend to free the advices from C++ syntax by translation into a uniform language like Scheme. We hope that the uniformity and simplicity of Scheme will point into necessary abstractions for the development of a user-level advice language.

References

- [Baggi, 1992] D. Baggi. *Readings in Computer-Generated Music*. IEEE Computer Society Press, 1992.
- [Balaban, 1983] M. Balaban. Notes and intervals arithmetic in tonal music. Technical report, Computer Science Department, SUNYA, 1983.
- [C. Ames, 1992] C. Ames. Cybernetic Composer: An Overview. In M. Balaban, K. Ebcioglu, and O. Laske (Eds). *Understanding Music with AI : Perspectives in Music Cognition*, chapter 8. AAAI-MIT Press, 1992.
- [Mehegan, 1984] J. Mehegan. *Jazz Improvisation, vol. 1-4*. Watson-Guptill, New York, 1984.