MUSICAOG - SUPPLEMENTARY INFORMATION

Anonymous authors
Paper under double-blind review

1 DETAILED FORMULATION OF NODES AND RELATIONS IN MUSICAOG

The **MusicAOG** is expressed by a 6-tuple:

$$\mathcal{AG}_{\text{mus}} =< S, V, E, \mathcal{R}, X, \mathcal{P} > \tag{1}$$

with $S, V, E, \mathcal{R}, X, \mathcal{P}$ represents root nodes, and-or nodes, relations, production rules, attributes and the probability model respectively, as explained in the main article. The nodes are layered as structural nodes and textural nodes. Textural nodes are further classified into event nodes and metrical nodes, as explained below.

1.1 STRUCTURAL LEVEL

At the structural level, MusicAOG provides a description of the musical form. Each node at this level represents a section within a piece of music, such as verses and choruses in songs, or exposition and capitulation in sonata forms, among others. The specific names of these sections are not critical for generalization in MusicAOG. Instead, they are defined by section nodes and period nodes, denoted as $V_{\text{struct}} = V_{\text{section}} \cup V_{\text{period}}$.

At this level, nodes are arranged successively along the time dimension. This means that all elements within the defined time region must belong to the corresponding structural node, and simultaneous nodes are not allowed. However, it is possible for two structural nodes to share common child nodes due to various succession types, which will be discussed later.

A notable feature of the structural level is that structural nodes can exhibit both recursion and recurrence. Recursion implies that a structural node and its child node can have the same type. Recurrence means that the child nodes of a structural node can all be of the same type, regardless of their population:

$$\begin{aligned} \text{recursion:} \quad & A \to AB, \\ \text{recurrence:} \quad & B \to AA, \quad \text{with } A \in V_{\text{struct}}, B \in V \end{aligned} \tag{2}$$

As mentioned earlier, we differentiate between node types based on their potential attributes. In this study, we assume that human perception of music at different scales involves distinct cognitive processes. Therefore, we refer to a structural node representing a short time scale, perceived through automatic processing, as a **period**. Conversely, a structural node representing a longer time scale, perceived through controlled processing, is referred to as a **section**. The attributes of a period primarily focus on emotions, style, and other intuitive aspects, while the attributes of a section encompass more reflective elements, such as the stories and philosophies behind the music, encouraging imagination. However, the specific attributes themselves are beyond the scope of this paper and are considered as future work.

1.2 TEXTURAL LEVEL

The textural level corresponds to the concept of "texture" in music theory, referring to a specific time region where music fragments align not only temporally but also "spatially". In this level, certain period nodes can be considered as terminal nodes in the structural level. It should be noted that the term "texture" in this context should not be confused with its use in vision.

Nodes at the textural level can be explicitly based on events such as notes and phrases, forming an event tree. There are also implicit nodes that represent ideas like meters and harmony, which form

a metrical tree. Horizontal links exist between these two trees to ensure synchronization. More formally, a period or-node can generate either a smaller, regular period and-node or a "terminal period" node known as an ensemble node. The ensemble node can be further decomposed into an event and-or tree and a metrical and-or tree, which are constrained by various relations. The following equations illustrate with simplicity:

$$\begin{aligned} v_{\text{period}} &\rightarrow u_{\text{period}} | u_{\text{ensemble}}, \\ u_{\text{ensemble}} &\rightarrow (\mathcal{AG}_{\text{event}}, \mathcal{AG}_{\text{metrical}}) :: E_{\text{ens}}, \\ &\text{with } v \in V_{\text{or}}, u \in V_{\text{and}}, E_{\text{ens}} \subset E \end{aligned} \tag{3}$$

1.2.1 EVENT TREE

In the event tree, nodes can represent either a phrase or a radical, denoted as $V_{\text{event}} = V_{\text{phrase}} \cup V_{\text{radical}}$. In this paper, the term "radical" is introduced to include notes, chords, musical figures, and unpitched sounds (e.g., percussion). These elements serve as the fundamental building blocks of music, enabling the translation of symbolic music into audio. The term "note" is intentionally avoided due to the following reasons:

(i) In many cultures, the smallest indivisible musical entities are not strictly equivalent to the Western notion of a note.

(ii) Sound resources often include samples that encompass not only pure notes in musicological terms but also various sound effects that can be classified as radicals.

(iii) Humans sometimes perceive a small group of notes as a cohesive whole. This is supported by the existence of notations such as ornaments and tremolos in musical scores, which serve to condense the representation.

(iv) In contemporary natural language processing (NLP), computer scientists primarily deal with words rather than individual letters. Similarly, in the field of AI music, researchers should explore the notion of radicals rather than focusing solely on notes.

In addition to notes and chords, musical figures encompass a variety of elements, including scales, broken chords, turns, mordents, repeats, trills, and more. These elements collectively form a set of radicals, which can be considered as a music dictionary. The representation of this music dictionary is defined as follows:

$$\begin{aligned} \nu_{\text{radical}} \in & \Delta_{\text{mus}} \\ = & \{(\Phi_i(t_{\text{on}}, t_{\text{dur}}; \mathbf{p}_i, \tau_i, \alpha_i), \beta_i) \\ & : t \in \Lambda_i(t_{\text{on}}, t_{\text{dur}}) \subset \Lambda\} \end{aligned} \tag{4}$$

In this representation, the onset time t_{on} and duration t_{dur} jointly specify the time domain Λ_i within the music piece. The pitch of a radical, or pitches if it represents a chord, is defined by the variable $\mathbf{p}_i$. If the radical corresponds to a percussion sound, the pitch can be null. For musical figures composed of multiple notes, one or several pitch centers exist, which serve as reduced representations of the figure. The timbre category of the sound is denoted by the variable τ_i, representing characteristics similar to instruments. Additionally, the variable α_i captures other attributes, including articulations. By employing the musical function Φ_i with these variables, a radical can be produced. Each radical possesses a set of address variables that establish connections to other elements within the music piece. These address variables dictate, for instance, how the current radical relates to the subsequent radical. Collectively, these variables contain sufficient information to generate MIDI data and, ultimately, render it into audio form.

A phrase is constituted by one or several radicals, accompanied by the relations established between them. These phrases embody distinctive musical entities and can encompass various forms, such as melodic fragments (referred to as motifs in musicology), accompaniment patterns, supportive sound effects, or background chords. It is important to note that both phrases and radicals can exhibit recursive and recurrent characteristics as well. In essence, they correspond to the grouping structure proposed in Generative Theory of Tonal Music (GTTM), but provide flexibility that remains faithful to human perception.

1.2.2 METRICAL TREE

The event tree formed by phrases and radicals alone is insufficient to constitute a complete music period, as time is a crucial element in music. Taking inspiration from the metrical structure proposed in GTTM, we introduce a metrical tree as a representation of the implicit meter in music, providing a generalized framework for time signatures.

$$\begin{aligned} &x_i(u_i) = (b, l_{\text{ref}}, s, h), \\ &\forall x_i \in X_{\text{metrical}}, u_i \in V_{\text{metrical}}, t(u_i) \in \Lambda_i \subset \Lambda \end{aligned} \tag{5}$$

Nodes within the metrical tree represent musical time intervals (distinct from physical time). Each node in the metrical tree represents a specific time interval and its corresponding beat hierarchy. Beginning from the root of the metrical tree, which possesses the "strongest" beat, child nodes are generated with gradually decreasing beat strength, along with nodes representing one or more weak beats. Polyrhythms can arise when a strong beat produces two strong beats as offspring, along with different numbers of weaker beats. Additionally, child nodes of a metrical node can have different durations, enabling the representation of polymeter. The beat value, denoting the stress of the beat within a node, is consistently assigned a value of 1 ($b(u) = 1$), where $u \in V_{\text{metrical}}$ represents the metrical node. This value indicates the relative strength of the beat in relation to its parent node. To facilitate comparisons, an additional variable known as the level variable, l_{ref}, signifies the reference level of beats. The level variable can exhibit flexibility and accommodate multiple possible values. For example, in Western music, the downbeat at the beginning of a measure may have $l_{\text{ref}} = 1$, while nodes at the beat level may have $l_{\text{ref}} = 0$.

Tempo (s) serves as another attribute assigned to each metrical node. While tempos on different nodes can vary, they must adhere to consistency principles by considering the tempo of the parent node and the overall tempo averaged over child nodes.

Mounting evidence suggests that harmony is akin to a metrical concept rather than being solely defined by individual musical events. Consequently, we consider harmony (h) as an attribute associated with metrical nodes. We adopt the harmonic reduction scheme proposed in GTTM but provide a more generalized encoding method. For further details, please refer to the Supplementary Materials. It is important to distinguish between "harmony" and "chord." Harmony is defined within the metrical tree and represents the broader concept of a tonal structure long time. On the other hand, "chord" is defined within a specific musical event node and refers to the simultaneous sounding of multiple pitches. This distinction is crucial to understanding the relationship between the metrical and event trees.

1.3 RELATIONS

Relations or edges play a significant role in musical composition as they provide a dependency grammar and contribute to the organization of musical elements. Temporally, the succession relations (E_{succ}) establish connections between fragments on both the structural and textural levels (in event trees and metrical trees), forming graphs. These relations govern the progression of musical elements and provide additional information regarding the continuity of separate voices and parts. Drawing inspiration from GTTM, the succession relations can be used to describe how one fragment succeeds another, encompassing attachment, tying, breathing, overlapping, and eclipsing successions. To regulate these successions, we introduce three variables:

$$E_{\text{succ}} = (s, t; \gamma, \rho, \sigma) : s, t \in V \tag{6}$$

Here, s and t represent the nodes connected by the edge. ρ defines the "temporal distance" between the fragments. If $\rho = 0$, they are simply attached. If $\rho > 0$, a rest of duration determined by the value of ρ occurs between them. If $\rho < 0$, the nodes overlap, with γ indicating the direction of the eclipse. The variable σ describes the smoothness of the transition between two successive nodes, representing how seamlessly the musical fragments flow into each other. The extreme case is the tied succession.

Vertical intervals between radicals can be described using E_{diad} and can be inferred from surrounding nodes connected by E_{succ} and E_{syn}. The synchronization relations (E_{syn}) act as pointers that bridge metrical nodes and radicals, specifying which musical time interval a node belongs to.

In many musical compositions, the restatement of musical elements enhances memorability. This is represented by the variation relations (E_{var}). It is important to note that, unlike traditional music theory, variation relations can encompass not only variations but also repetitions and sequences. The variation relations also possess attributes, although the further details are beyond the scope of this paper.

2 Representability on musical notations under different cultures

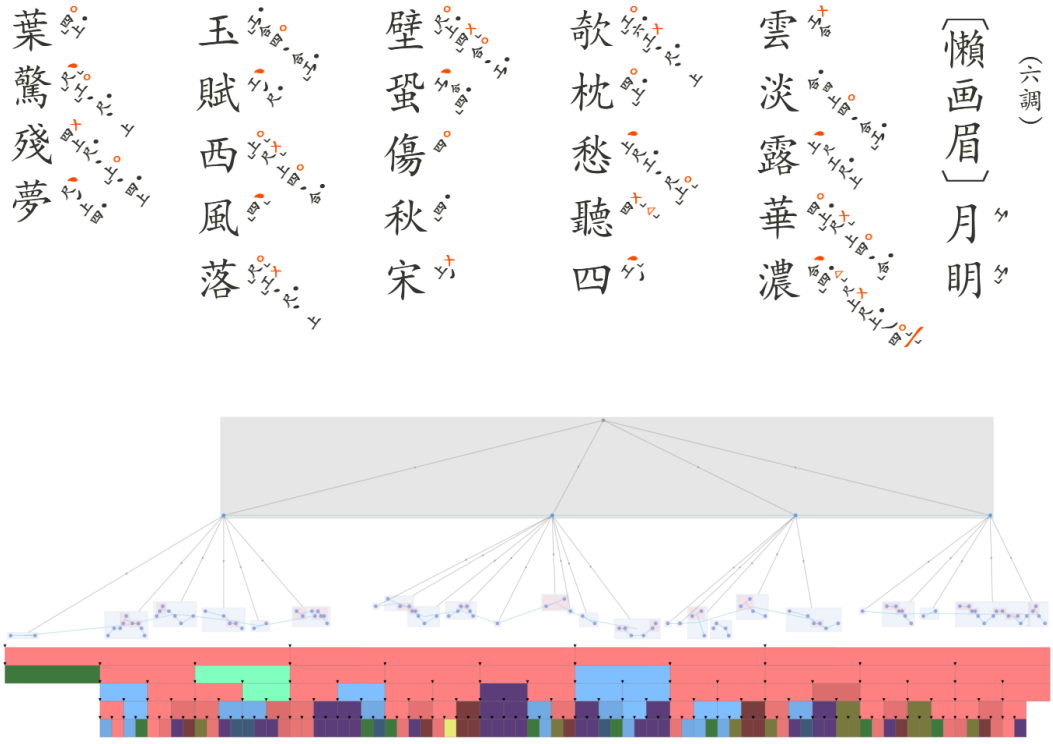

Figure 1: Gonchepu representation (top) and its representation in MusicAOG (bottom)

We have undertaken the transcription of a gongchepu [1] — a traditional Chinese musical notation — into the MusicAOG framework as shown in Figure 1. In this representation, solfège delineated by diverse qiangge is characterized as radicals, while qikou serves as a delineator for phrases, and banyan is integrated into the metrical tree structure. Despite the fact that gongchepu encapsulates a marginally less granular musical detail compared to the Western staff notation, the MusicAOG framework successfully amalgamates both under a unified representation. This facilitates a systematic comparison and potentially paves the way for parsing and style transfers between musics of varied cultural origins.

Numerous symbolic music representations, including gongchepu and staff notation, manifest inherent ambiguities. Rather than being a limitation, this ambiguity fortifies music's expressive prowess Strayer (2013). The disparities in ambiguity across these representations imply an inherent hierarchical structure in musical comprehension. With the hierarchical integration in our MusicAOG, we strive to amalgamate distinct musical notations, concurrently preserving their intrinsic expressive ambiguities.

[1] https://gongchepu.net/reader/384/

2.1 DESCRIPTORS EXPLAINATION

1. **countChildrenDurOfPhrase**: count frequencies of durations of child nodes produced by a phrase.
2. **countSuccessiveMidiHinterval**: count the horizontal intervals, in semitones, between successive notes.
3. **countLastBeatSolfegeToCadence**: count the degree of pitch of notes appeared in the last beat of a period.
4. **countDiadOctDiff**: count the difference in octave of notes connected by diad relations.
5. **countInstrumentOctave**: count the octaves of notes for each instruments.
6. **countDegreeToDegree**: measure how a degree of pitch of a note is succeeded by successive note's pitch degree.
7. **countNumChildrenOfPhrase**: count number of child nodes produced by a phrase.
8. **countDiadPitchInterval**: count the pitch interval between notes connected by diad relations.
9. **countDiadNoteDurationDiff**: count the difference in duration of notes connected by diad relations.
10. **countSolfege**: count sofa of notes within that key provided by the parent period nodes.
11. **countBeatOfNotesOfPhrase**: count the beat where the notes onsets are on.
12. **countSuccessiveNoteDurDiff**: count the duration differences between successive notes.
13. **countDiadOnNote**: count how many diad relations is on notes.

2.2 AN EXAMPLE OF BEST GENERATED MUSIC PIECE

See Figure 2.

REFERENCES

Hope Strayer. From neumes to notes: The evolution of music notation. *Musical Offerings*, 4:1–14, 2013. ISSN 23308206. doi: 10.15385/jmo.2013.4.1.1.

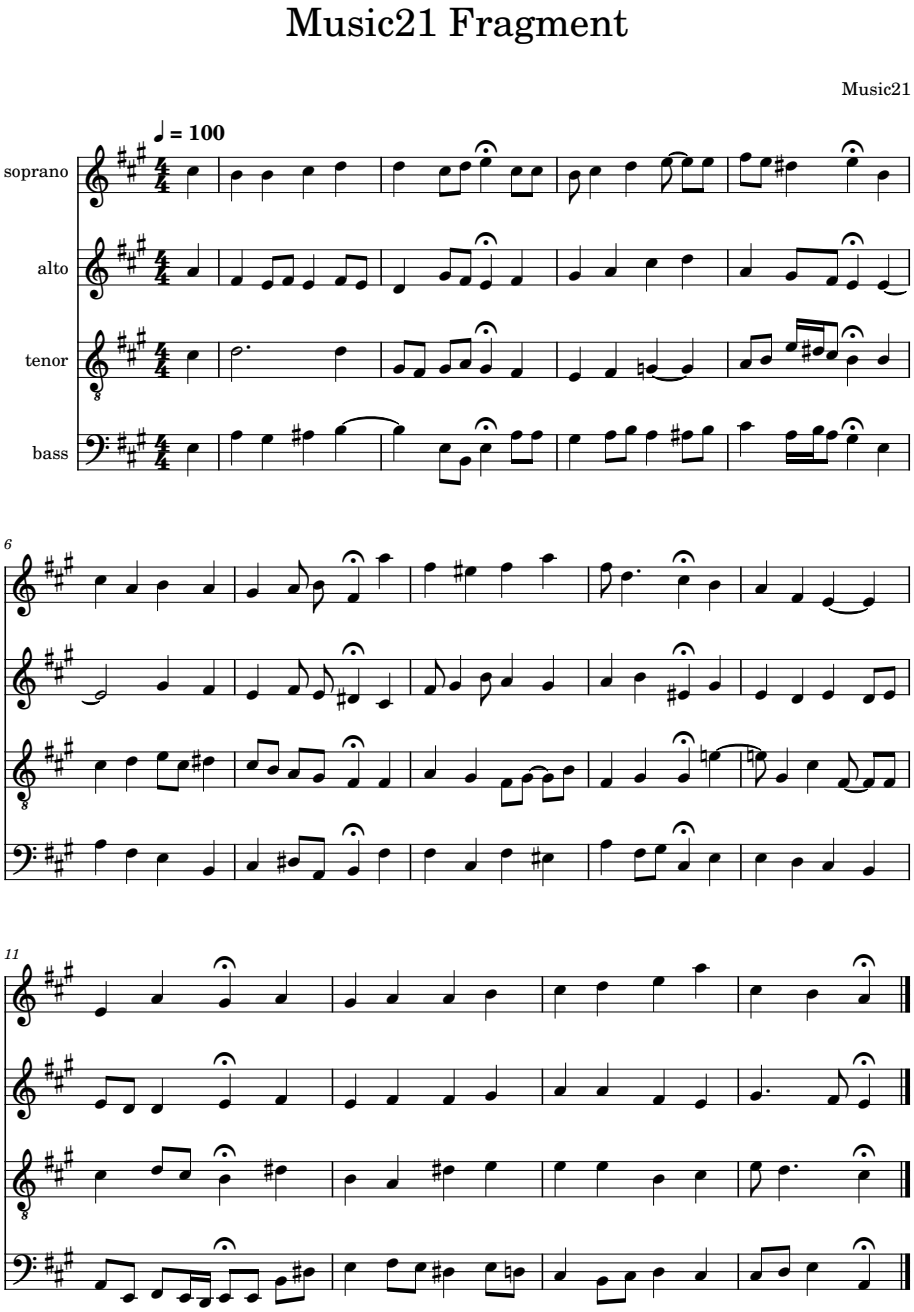

Figure 2: Example score sampled on a MusicAOG