Extending Generative Neo-Riemannian Theory for Event-based Soundtrack Production

Simon Colton and Sara Cardinale

School of Electronic Engineering and Computer Science
Queen Mary University of London, UK
{s.colton, s.cardinale}@qmul.ac.uk

Abstract. We present the GENRT music generation system specifically designed for making soundtracks to fit given media such as video clips. This is based on Neo-Riemannian Theory (NRT), an analytical approach to describing chromatic chord progressions. We describe the implementation of GENRT in terms of a generative NRT formalism, which produces suitable chord sequences to fit the timing and atmosphere requirements of the media. We provide an illustrative example using GENRT to produce a soundtrack for a clip from the film A Beautiful Mind.

Keywords: Generative Music · Neo-Riemannian Theory · Soundtracks

1 Introduction

Musical soundtracks are an essential part of many forms of media. They help create an immersive experience by supporting story lines, portraying characters, establishing the overall setting and mood, highlighting important events and adding to the aesthetic value of media. Soundtrack music has requirements not shared by music in general, which makes it an interesting and valuable target for generative music, which could lead to useful tools for composers. Our problem setting here is to be given an existing piece of media like a video clip from a film, or an audio excerpt of an actor reading for an audiobook, and to generate an appropriate soundtrack for it. The media is supplied along with a list of episodes given in terms of (a) the start and end timestamps of the episode in milliseconds (b) an overall atmosphere (e.g., calm) for the episode and (c) a change in overall atmosphere to be reflected smoothly during the episode (e.g., getting more exciting). Optionally, each episode may have a particular event at the conclusion of it, specified by the timestamp at which it needs to occur and the emotional or situational change in the media that the event represents.

An appropriate soundtrack would have to satisfy many things, but we concentrate here on it being *accurate* in terms of timings of episodes and events, *influential* in terms of musical changes which audibly support situational or emotional changes in the media and *atmospheric* in terms of the nature of the music between events. We describe the GENRT music generation system which can produce MIDI files that act as accurate, influential and atmospheric soundtracks

NRO	Major	Minor
R	$\{a, b, c+2\}$	$\{a-2, b, c\}$
P	$\{a, b-1, c\}$	$\{a, b+1, c\}$
L	$\{a-1, b, c\}$	$\{a, b, c+1\}$
N	$\{a, b+1, c+1\}$	$\{a-1, b-1, c\}$
N'	$\{a-2, b-2, c\}$	$\{a, b+2, c+2\}$
S	$\{a+1, b, c+1\}$	$\{a-1, b, c-1\}$

Compound NRO	Emotion/Situation
LP	Antagonism
L	Sorrow, loss
N	Romantic encounters
PRPR	Mortal threats, dangers
RL	Wonderment, success
NRL	Suspense and mystery
RLRL	Heroism (Lydian)
NR	Fantastical
S	Life and death

Table 1. (a) NRT Rewrite rules for major and minor chord starting points. In each case, the starting chord is $\{a, b, c\}$. (b) Association of compound Neo-Riemannian operators and emotional and/or situational scene elements (from [11]).

to a given piece of media. Underpinning GENRT is an extension of a musical theory called Neo-Riemannian Theory, described in section 2. This theory was devised for the analysis of music, but we provide a generative formalism, which can be employed to make music, as detailed in section 3. We describe GENRT in terms of this formalism in section 4, highlight the two main issues of producing appropriate chord changes for events and making the music between events suitably low-key, so the event music is not obfuscated. In section 5, we use GENRT to produce a soundtrack for a clip from the film A Beautiful Mind, and we end with conclusions and a discussion of future work.

2 Background

Neo-Riemannian Theory (NRT) comprises methods to analyse chromatic chord progressions that do not employ tonality [6]. These types of chord progressions can be difficult to analyse using conventional music theory, as they do not follow traditional models which include the use of keys and modes to understand harmonic developments. A musical cue such as a chord change can be analysed in terms of which Neo-Riemannian Operators (NROs) have been used. As per the original theory, an NRO applies a transformation to trichords consisting of the tonic, third and fifth of a major or minor scale in some permutation. Every NRO outputs a major chord given a minor one as input, and vice-versa, and each can act on any trichord. The six NROs originally identified are given in table 1(a). It is also possible to analyse a single chord change as the application of multiple NROs iteratively, which we call *compound* NROs.

Chromatic chord progressions are widely used in movie soundtracks, as the film scenes often change in rapid, fairly pronounced ways, and composers have found that chromatic chord progressions highlight these changes well. In [11], Lehman analysed film scores in depth and found certain chromatic chord changes – which he described in terms of compound NRO usage – were commonly used by composers to illustrate emotional or situational changes in the movies. For instance, a quick change of chords can be used to represent important on-screen

events such as the appearance of a character, a significant death or a disappointing failure. Lehman mapped compound NROs to emotional/situational changes in film narratives, as per table 1(b) by analysing various soundtrack cues and noting when certain operators were used to describe specific events. While not an exhaustive list of compound NROs and emotions, it provides a starting point to understand how chromaticism can represent events and emotions in soundtracks.

The original papers on NRT obfuscated matters somewhat with fairly dense and sometimes superfluous mathematics. In [4], we rationalised the operators as NROs as portrayed in table 1(a). This enabled us to describe a basic generative reading of NRT where the NROs are employed as conditional re-write rules, with the condition being whether a trichord is major or minor. We likewise rationalised the notion of compound NRO and, drawing on the work of Cohn and Lehman, stated the main tenets of generative NRT as follows:

- Chord sequences are produced from a given starting chord via the repeated application of compound NROs.
- Emotional and situational changes in the target media can be reflected in the music via the application of particular compound NROs, as per table 1(a) to produce suitable chord changes.
- The longer the compound NRO used for a chord change, the more striking it will be, audibly.

Using these tenets as a foundation, we significantly generalise and extend this formalism in section 3 below, to enable generative NRT to become the basis of a working implementation of the GENRT music generation system.

2.1 Related Work

The generation of chord progressions has been much investigated for automatic production of musical harmony, and Wiggins [15] looked at the notion of intentionality in this respect, which is important for the generation of chord progressions to accompany scenes from media such as films. Bernardes et. al. [2] implemented the D'accord harmony generation system which worked over a perceptually motivated tonal interval space. While not using NRT, Monteith et al. [13] generated music to induce targeted emotions, using statistical techniques such as Hidden Markov Models, and applied this in [12] to produce affective music to accompany the audio of fairy tales being read aloud.

Neo-Riemannian Theory for music generation has been explored in as Chew and Chuan [5], where a style-specific accompaniment system was proposed. The system utilises NRT to represent transitions between neighbouring chords. The authors make use of roman numeral musical analysis to describe the chord progression. This is an analysis technique usually used in tonal music, as the roman numerals describe tonal relationships between chords. Given this observation, this work seems to be aimed at the generation of diatonic, tonal chord progressions which respect functional harmony. This is a limitation of this approach, as NRT is used to analyse chromatic progressions which are not restricted to

functional harmony rules. Similar limitations can be observed in other systems and approaches, such Amram et al. [1]. Here, the authors implement a generative chord-based composition tool. Similarly to [5], the authors implement NRT in a tonal, diatonic way, therefore not using the theory to its full potential.

Other emotion-based systems include Mind Music [7], which focuses on designing music based on a character's emotions using mood nodes and a spreading activation model, creating adaptive audio for game characters. Hutchings et al. [9] further developed these ideas, proposing an adaptive music system that combines a multi-agent system (MAS) and cognitive models of emotion. The agents focused on harmony, melody and rhythm to develop compositions.

3 Extended Generative NRT

In order to produce more diverse music and to have more options for controlling the generative process, we have substantially extended and adapted our generative NRT approach as described below. Our starting point was the observation that each of the existing NROs took a trichord of tonic, third and fifth notes from either a major or minor scale, and produced a new trichord which is also a major or minor tonic, third or fifth. Each of the original NROs preserves at least one note of the original chord and no note in the output chord is more than a tone away from a note in the original chord. To describe the extended generative NRT implementation in the next section, we use a formalism which extends trichords and NROs. We start by generalising trichords past the limitation of just major and minor chords, as follows:

• [A1] A *note* is an integer from the midi pitch range $\{0, \ldots, 127\}$. For instance, middle C on a piano keyboard is note number 60.

• [A2] The *note class* for a given note, N, denoted $nc(N)$, is calculated as:
$nc(N) = \{k : 0 \leq k \leq 127 \text{ and } k \bmod 12 = N \bmod 12\}$
e.g., the note class for middle C (midi note 60) is $\{0, 12, 24, 36, \ldots, 120\}$.

• [A3] A *general trichord*, $T = [t_1, t_2, t_3]$, is an ordered triple of three distinct notes from strictly different note classes where no two notes are less than a whole-tone apart, i.e., $\forall\ i, j\ |t_i - t_j] >= 2$. We say T is *sorted* if $t_1 < t_2 < t_3$.

• [A4] A general trichord $T = [t_1, t_2, t_3]$ is a *major trichord* if some permutation, $[a, b, c]$, of T is such that $b - a = 4$ and $c - b = 3$. Likewise, T is a *minor trichord* if there is a permutation $[a, b, c]$ such that $b - a = 3$ and $c - b = 4$.

Given this foundation, we generalise Neo-Riemannian operators from the originals to work with the expanded definition of trichords.

• [B1] A *General Neo-Riemannian Operator (GNRO)* is an ordered triple of integers $[n_1, n_2, n_3]$ such that $\forall\, i\ (-2 \leq n_i \leq 2)$ and $0 \in \{n_1, n_2, n_3\}$.
• [B2] An application of a GNRO, $G = [n_1, n_2, n_3]$, to a general trichord, $T = [t_1, t_2, t_3]$, denoted $G(T)$, involves adding in place the three values of G to the three values of T, i.e., $G(T) = [t_1 + n_1, t_2 + n_2, t_3 + n_3]$.

• [B3] A *Compound GNRO* (CGNRO), $[G_1, G_2, \ldots, G_k]$, of *length* k, is a non-empty ordered list of k GNROs.
• [B4] An application of a CGNRO, $C = [G_1, G_2, \ldots, G_k]$, to a general trichord, T, denoted $C(T)$ involves the iterative application of each G_i to the previous result, i.e., $C(T) = G_k(\ldots G_2(G_1(T))\ldots)$.

The original set of NROs given in table 1(a) had the advantage that, given a major or minor chord as input, the output would also be major or minor. Unfortunately, that is not true of GNROs, i.e., given a general trichord as input, the output might not be a general trichord as per definition A4. Hence we specify the admissibility of a GNRO for a given chord below. Also, as we discuss in the following sections, to produce acceptable music to accompany a given video clip, requires the ability to produce situational chord changes at exactly the right dramatic moment in the target video. This is enabled by the use of the original NROs of table 1(b), re-interpreted as CGNROs.

• [C1] A CGNRO, C, is *admissible* for a given general trichord, T, if and only if the resulting triple of notes, $C(T)$, is itself a general trichord, i.e., with three notes in distinct note classes and each pair of notes at least a whole tone apart.
• [C2] A CGNRO, C, is *m-admissible* for a given general trichord T if and only if $C(T)$ is either a major or a minor trichord as per definition A4.
• [C3] An ordered pair of two admissible CGNROs, $[C_1, C_2]$, produces a *situational chord sequence* from a given general trichord T, if C_1 is m-admissible, and C_2 is from table 1(b). In effect, C_1 makes sure that the general trichord input to C_2 is a major or minor trichord, so the emotional effect of using C_2 is correct.

Importantly, the generative process also requires producing music in between the dramatic moments which doesn't include any striking chord changes that might obfuscate the dramatic chord change when it happens. The following definitions help describe how we try to achieve this:

• [D1] A *major key signature* for the note N is the list of note classes:
$[nc(N), nc(N+2), nc(N+4), nc(N+5), nc(N+7), nc(N+9), nc(N+10)]$.
A *minor key signature* for the note N is the list of note classes:
$[nc(N), nc(N+2), nc(N+3), nc(N+5), nc(N+7), nc(N+8), nc(N+10)]$
• [D2] A general trichord, $T = [t_1, t_2, t_3]$, is in the (major or minor) key signature, K, of given note N if $\forall\ t_i \in T$, $nc(t_i) \in K$.
• [D3] Given a *focal note*, N, a general trichord, $T = [t_1, t_2, t_3]$, can be *focal mapped* to N by calculating $T' = [t'_1, t'_2, t'_3]$ such that $\forall i,\ (t'_i \in \{1, \ldots, 127\}$ and $\nexists k\ \in \{1, \ldots, 127\}$ where $nc(k) = nc(t'_i)\ \wedge\ |k - N| < |t'_i - N|)$. Informally, the mapping takes each t_i to the note in the same note class closest to N.

Soundtracks usually comprise more than a chord sequence, and GENRT affords the production of music with 8 tracks: three for the notes of a trichord, three for percussion, and one each for a melody and bassline. Described in more detail below, the trichord sequence is dictated by the application of CGNROs, starting from a given general trichord. It can be given a rhythm so that the chord is played multiple times during a (given) timespan. The melody and bassline can

len	num	admissible			m-admissible			identity			different		
		min	av	max	min	av	max	min	av	max	min	av	max
1	60	16	31	51	2	9	12	0	1	2	15	31	51
2	3600	1242	1697	1920	212	405	552	60	72	95	102	142	195
3	216000	77151	94470	104535	15594	21880	25146	1248	1805	2955	220	314	412

Table 2. Number of admissible, m-admissible and identity CGNROs of length 1, 2 and 3 over all general trichords with notes in one octave. Number of different chords realised. Values have been rounded to the nearest integer.

mirror the trichord to selectively play a sequence of notes from the chord at different octaves, using a given rhythm. To describe this, we use the following definitions:

• [E1] A musical *episode* of a given duration (in ms), d, spans a chord sequence of N general trichords $[c_1, \ldots, c_N]$, with the duration (in ms) of chord c being denoted $dur(c)$, such that $\sum_{i=1}^{N} dur(c_i) = d$.
• [E2] For a given general trichord, T, and given a number of beats b that $dur(T)$ is to be split into, a *rhythm specification*, denoted $rh(T)$, for T is a set of pairs $(t/b, d/b)$ which each dictates that the chord should be played on beat t, and last for d beats.
• [E3] For a given general trichord $T = [t_1, t_2, t_3]$ and given the number of beats, b, that $dur(T)$ is to be split into, the *melody specification* for T, denoted $mel(T)$ is a set of triples $(t_i, t/b, d/b)$ where each $t_i \in C$. Each triple dictates that while chord T is playing, melody note $t_i + 12$ should start at beat t and last duration b. The *bassline specification* for T is the same as a melody specification, but specifies that notes $t_i - 24$ are to be played.
• [E4] A *duration fraction signature* is a list of fractions $[\frac{1}{p_1}, \ldots, \frac{1}{p_k}]$ such that each p_i is an integer and $\forall i,\ p_i < p_{i+1}$.
• [E5] Given a duration of d ms, a sequence of notes $N = [n_1, \ldots, n_j]$ to be played over d, and a duration fraction signature $F = [\frac{1}{p_1}, \ldots, \frac{1}{p_k}]$, we say that N *adheres* to F if the start time, s_i, of each n_i can be expressed as $s_i = d * \frac{f_i}{p_i}$ for some $p_i \in \{p_1, \ldots, p_k\}$ and integer, f_i, such that $f_i \leq p_i$ and $f_i \geq 0$.
• [E6] If a sequence of notes, S, with starting times $s_1, \ldots, s_n$ adheres to duration fraction signature $F = [\frac{1}{p_1}, \ldots, \frac{1}{p_k}]$, we say the *level of adherence* of S is the smallest $x \in \{1, \ldots, k\}$ for which $\forall\, i\ s_i = \frac{f_i}{p_x}$ for some integer f_i. As an example, suppose $F = [\frac{1}{1}, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}]$ and a three note melody, M, is to be played over duration d, with notes starting at $d * \frac{0}{4}, d * \frac{2}{4}$ and $d * \frac{3}{4}$. Here, M adheres to F at level 3, as $\frac{1}{4}$ is the third fraction in F.

3.1 The Distribution of Admissible CGNROs

In order to guide the implementation of the GENRT system, we checked the distributions of admissible CGNROs of lengths 1, 2 and 3, as reported in table 2. With an exhaustive search, we determined that there are 112 general trichords

as per definition A4, with notes all between middle C and the B note above it. There are also 60 CGNROs of length 1, hence $60^2 = 3600$ of length 2 and $60^3 = 216,000$ of length 3. We applied each of the CGNROs of length 1 to each of the 112 general trichords and recorded the number of CGNROs that are admissible and the number which are m-admissible for each trichord. We did the same for length 2 and 3 CGNROs and recorded the minimum, average and maximum numbers of admissible and m-admissible CGNROs in table 2. In addition, we determined how many applications of a CGNRO to a trichord were identity mappings resulting in no overall change in the three notes of the chord. Finally, we recorded how many distinct chords resulted in the application of the entire set of CGNROs to each trichord, again recording minimum, average and maximum values of this in table 2.

We see that when CGNROs of length 1 are used, there are an average of 31 which can be used for a general trichord and this number greatly increases as the length of the CGNRO increases. However, there are only an average of 9 m-admissible CGNROs of length 1, and for some chords, this can be as low as 2. While this guarantees that every general trichord can be turned into a major or minor chord with a CGNRO of length 1, there is not much diversity. We also see that there is a lot of redundancy in the set of CGNROs. For instance, while there are 216,000 possible CGNROs of length three to apply to a given general trichord, they result in only 314 different chords, on average, and 1,805 of the CGNROs are the identity mapping, again on average.

4 The GENRT Music Generation System

Two initial requirements for generative soundtrack production are that musical changes happen at precise times and for tempo changes to increase and decrease tension in episodes. We have implemented in the GENRT system the following approach for generating music which fits a given video clip by ensuring that certain CGNROs are employed at given times which match emotional or situational changes in the video. The user can also specify ways in which the music is *dampened* between the dramatic changes, so that the chord changes the CGNROs impose have more effect, i.e., are not obfuscated by the surrounding music.

4.1 Event-Reflecting Musical Episodes

As per the definitions of section 3, each soundtrack is composed of a series of episodes which each consist of a sequence of general trichords, played on one instrument. The soundtracks also have a melody and bassline played on (optionally) different instruments, and three percussion instruments adding rhythm and texture. The user specifies the number of episodes that the composition will have. Each episode is specified to either have a particular duration or to end with a particular situational chord sequence (as above), with the final chord change happening at a precise number of milliseconds since the start of the composition. The user also specifies a target for the duration of the first chord (in ms) and a

target for the duration of the last chord in each episode, along with the notes of the starting chord (or specifies that it can be a random general trichord). If the first and last chords of an episode differ in duration, then this indicates a gradual change in tempo during the episode, to be smoothly realised by GENRT.

Given these specifications, GENRT calculates the durations for a chord sequence as follows. Suppose the start chord duration target is s and the end chord duration target is e, with the final chord happening at time t to coincide with a situational/emotional change in the video. For a smooth linear transition of chord durations from s to e, GENRT determines the number of chords in the sequence, n, so that the number of milliseconds changes from one chord to the next is r and the final chord occurs at t. We note the following equations hold:

$$(i)\ t = \sum_{i=0}^{n-2}(s + ir) \qquad (ii)\ e = s + (n-1)r$$

The first equation states that the total duration of the first n-1 chords needs to equal the timestamp, t, when the final chord is to be played; and the second expresses the final chord duration, e, in terms of the start chord duration, s, and the variables n and r. Expanding (i) using the well-known formula for the summation of the first i integers, we see that:

$$(iii)\ t = (n-1)s + r\left(\sum_{i=1}^{n-2} i\right) = (n-1)s + r\left(\frac{(n-1)(n-2)}{2}\right)$$

We can also arrange (ii) to show that: (iv) $r = \frac{(e-s)}{(n-1)}$, and substitute this version of r into (iii), to get:

$$t = (n-1)s + \frac{(e-s)}{(n-1)}\left(\frac{(n-1)(n-2)}{2}\right) = (n-1)s + \left(\frac{(e-s)(n-2)}{2}\right)$$

enabling us to solve for n, giving:

$$n = \frac{2(t+e)}{(s+e)}$$

We can calculate r using n in equation (iv). Note that in practice, we need to take the floor, $\lfloor n \rfloor$, of the calculation for n above, as it is the number of chords in the episode, hence an integer. This introduces a discrepancy that means the final chord doesn't happen at the correct time t. To correct this, a value of x milliseconds is added to the duration of each of the chords. x is calculated by using $\lfloor n \rfloor$ in equation (iv) to calculate r, which is in turn used along with $\lfloor n \rfloor$ in equation (iii) to calculate a value t'. The difference between user-given t and t' is divided by $n-1$ to give x. This ensures that the timing of the final chord is at exactly time t, but means that the start and end chord durations are often not exactly as targetted by the user, but are usually very close. Note also that,

if the user only supplies a duration, d, for the episode and not a timing for the final chord, GENRT calculates n and r in a similar way using d instead of t.

The use of episodic music production is vital for the application of GENRT to soundtrack generation. By enabling the user to specify the end chord duration for one episode to be the start chord duration for the next episode, they smoothly merge during the soundtrack, allowing tempo changes to reflect longer-term atmospheric changes. This supplements the reflection of dramatic or situational changes in the video with CGNRO-generated chord changes. The user can also specify start and end volumes for each of the instruments (chords, melody, bassline and percussion) in each episode. The user specifies which MIDI instrument each track has and we use the Fluidsynth sound font (fluidsynth.org) for these. Instruments are specified on a per-episode basis, and the user can also specify a number of chords in advance of the episode change that the change of instruments begins. GENRT then uses two tracks to interpolate the volume of the latter episode's instruments from zero to one during this period, likewise interpolating the former episode's instrument volume from one to zero.

4.2 Chord Sequence Generation

Once the number and duration of chords in an episode have been calculated, GENRT generates a random sequence of CGNROs with which to generate a chord sequence. Starting with the user-given chord, e.g., middle C, E, G of C major, the first CGNRO is chosen randomly from those which are admissible for this general trichord. Applying this CGNRO produces a new chord, and the process iterates until near the end of the episode. The user can specify that a single dramatic/situational event happens per episode. To do so, they supply the timestamp in milliseconds from the start of the soundtrack that this happens at and a keyword from the list of table 1(b): [Antagonism, Sorrow, Romance, Threat, Wonderment, Mystery, Heroism, Fantasy, LifeAndDeath].

GENRT ensures that the corresponding CGNRO from table 1 is applied to generate the final chord of the episode via an appropriate situational chord change (definition C3). Given that the audible power of the chord change has only been considered for major or minor chords, GENRT must ensure that the penultimate chord to which this CGNRO is applied is either major or minor. To do this, it chooses the penultimate CGNRO randomly from the list of m-admissible ones available to apply to the anti-penultimate chord. We note from table 2 that there will always be at least two m-admissible CGNROs to choose from. The user is able to provide a different rhythm specification (definition E2) for the chords of each episode, so that over the chord duration, the notes of the chord are played rhythmically, if desired.

4.3 Bass, Melody and Percussion Generation

Once a chord sequence has been generated using random CGNROs, GENRT generates a bassline and melody to fit the chords. These processes are currently fairly simplistic and we plan to improve upon them in future work. For the bass,

the user provides a bassline specification as defined in E3, for a given trichord, that dictates which of the three notes of the chord are played and at what time during the chord duration, at two octaves below the chord note. The user can also specify how staccato the notes are by providing a cutoff proportion so that each note is curtailed before its full duration.

For melodies, the user can also provide a melody specification as per E3 which will generate note sequences for a chord an octave above its notes. Alternatively, they can specify that GENRT itself produces a voice-leading melody. It does this in two stages over all the episodes of the composition. Firstly, for the first chord of the first episode, it chooses a random permutation of the chord's three notes, so that the notes an octave above become the *backbone* of the melody for that chord. For each subsequent chord, GENRT chooses a permutation of the chord's notes for which the first note is as close to the last note of the melody for the previous chord as possible.

In the second stage, passing notes are added between every pair of backbone notes (b_1, b_2) for each chord's melody. Assuming that $b_1 < b_2$, starting at $b_1 + 2$, passing notes which are a whole-tone away from the previous one are added until $b_2 - 2$ is reached. If the final backbone note of a chord's melody is the same as the first of the next chord's melody, the former is removed. If not, then more passing notes are added to join the end and start notes accordingly. As before, the user can specify a cutoff proportion, so that the melody sounds more staccato if required. Three different percussion instruments can be specified and utilised by the user providing rhythm specifications for each, as per definition E2.

4.4 Dampening the Music between Events

In initial tests with the extended generative NRT approach, we found that it was often hard to hear the emotional/situational chord changes because there were many similar chord changes in advance which somewhat obfuscated things. One way to address this is to make the intermediate music less striking, and we implemented some techniques in GENRT that users can experiment with for this purpose, as follows.

Firstly, the minimum and maximum length of the CGNROS that are employed in generating the chord sequence can be set. As the CGNROs get longer, the further the output chord notes are from the input ones, and the more audibly striking the chord sequence is. Hence, employing only CGNROs of length 1 or 2 is advised for subdued intermediate music. The user can also specify a major or minor key signature as a *fixed key* for the chords in the episode. As per definition D2, a general trichord is in key K if the note class of each of its notes is in K. GENRT can then choose only CGNROs where the output chord is in the fixed key. We have found that this significantly dampens the music. Note that the emotional/situation CGRNO specified in an event is exempt from this rule, which makes the chord it produces more impactful. In addition, when GENRT produces voice-leading melodies, it can be told to ensure that all the passing notes are in the chord sequence's fixed key, if there is one.

Sometimes the random nature of CGNRO selection that GENRT employs can produce pitch drift, in that the chords go consistently up (or down) in pitch, dragging the whole composition with them. While this is something that users may want to add by design, it can be striking and create music which is inappropriate for an episode. Hence, we added the option for GENRT to map each generated chord to a user-given focal note, as per definition D3. This ensures that all chords stay in roughly the same pitch range.

When GENRT produces a voice-leading melody $M = [m_1, \ldots, m_n]$ over a chord, by default it plays each note of the sequence at equal intervals over the duration of the chord, d. Often, this clashes with the static, user-specified rhythms that the chords, bassline and percussion instruments play. That is, the number of notes in M means there are polymetric rhythms at play, for instance, five melody notes played at the same time as four chord repetitions. This can be quite striking, so we added the option for users to suppress this by specifying a duration fraction signature $F = [\frac{1}{p_1}, \ldots, \frac{1}{p_k}]$ as per definition E4. Then GENRT takes the generated melody for a chord and produces a list of start times $[s_1, \ldots, s_n]$ with an optimally high level of adherence to F, as per definition E6. To do this, it finds the largest fraction $\frac{1}{p_x} \in F$ which is smaller than or equal to $\frac{1}{n}$ and sets each s_i initially to $\frac{i-1}{p_x}$, with each note initially having duration $d * \frac{1}{p_x}$ over d.

This can mean that the total duration of the notes is less than d, and melodies end early for a chord, which is noticeable and undesirable. Hence GENRT randomly chooses a note to extend the duration of, and this is done by adding $\frac{1}{p_x}$. Accordingly, GENRT increases the starting points of the notes following the extended one. It does this until the total duration of the notes is d. Noting that passing notes in a melody can often clash with the chord being played at the same time, we further refined the process so that only the backbone notes of the melody (which don't clash) were extended. This quantization of the note starting points means that a duration fraction signature can be specified which fits with the static rhythms of the chords, bassline and percussion, e.g,. if they only start at moments which are multiples of $\frac{1}{8}$ of d, then a duration fraction signature of $F = [\frac{1}{1}, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}]$ will ensure that the melody notes all do likewise. Optimal adherence level to F (definition E6) means that the melody notes will have the simplest rhythm.

Trying these optional specifications for music generation, we found that we could hear the dampening of music reasonably well. However, to be more concrete and confident in this, we performed some experiments to quantify the benefits in this respect. As per the previous experiments, we exhaustively applied all CGNROs up to length 3 on all 112 general trichords with notes between middle C and the B above. However, in a second session, we stipulated that only input chords where all notes in the fixed key of C major were considered, and the result was only recorded if the output was likewise in the key of C major. This simulates the usage of a fixed key to dampen chord sequences. In a third session, we instead mapped all the input chords and output chords to focal note 66. To measure the dampening effect in these sessions, we used two measures:

session	len	admissible			m-admissible			chord dist			focal dist		
		min	av	max	min	av	max	min	av	max	min	av	max
standard	1	16	31	51	2	9	12	0.00	0.78	1.33	1.33	3.11	6.00
	2	1242	1697	1920	212	405	552	0.00	1.00	2.67	1.33	3.22	7.33
	3	77151	94470	104535	15594	21881	25146	0.00	1.17	4.00	1.33	3.33	8.67
fixed	1	5	9	13	0	3	5	0.00	0.74	1.33	1.67	3.14	4.67
key	2	337	451	518	36	118	178	0.00	0.78	2.33	1.67	3.25	6.33
	3	18002	21961	24272	3546	5723	6813	0.00	0.99	4.00	1.67	3.39	7.67
focal	1	16	31	51	2	9	12	0.00	0.88	3.00	1.33	3.00	4.67
mapped	2	1242	1697	1920	212	405	552	0.00	1.07	4.00	1.33	3.00	4.67
	3	77151	94470	104535	15594	21880	25146	0.00	1.16	5.00	1.33	3.00	4.67

Table 3. Number of admissible and m-admissible CGNROs, chord distances and focal note distances for standard, fixed key and focal mapping sessions.

• *Chord distance* measures the distance of a chord, O, output from a CGNRO from the input chord, I. To do this, for each note of O, it determines the closest note (or notes) in I, and calculates the absolute distance to it. An average over the three input notes of the input general trichord gives an overall distance. The intuition here (as highlighted by analytical NRT) is that the further a chord is away from the previous one, the more striking the chord change is.

• *Focal point distance* measures the distance of the output chord from a given focal note N. It does this by averaging the individual absolute distances of each note from N. The intuition here is that a series of chords with pitches occurring in a narrow range will be less striking than a series within a larger pitch range.

The number of admissible and m-admissible CGNROs of lengths 1, 2, and 3 for the exhaustive standard session, fixed-key session and focal-note session are given in table 3. Note that the standard session is the same as in table 2 above, for easy comparison. We see that, as expected, the mapping to a focal note doesn't alter the number of admissible CGNROs. However, if the chords of an episode are constrained to those in a particular key, then the average number of admissible and m-admissible CGNROs available for a chord drops considerably, and it may be sensible to allow longer length CGNROs to maintain a level of diversity in the output. Table 3 highlights a procedural problem, i.e., that if the user employs a fixed key and only CGNROs of length 1 (for high dampening), then for some chords, the number of m-admissible CGNROs is zero, hence it would not be possible to produce a major or minor chord ready for a CGNRO from table 1, to reflect an emotional or situational change. To fix this when it happens, we allow GENRT to temporarily increase the CGNRO length to 2.

The distance metrics are also given for the three sessions in table 3. We see that, as expected, average chord distance increases along with CGNRO length allowed, from 0.78 (length 1 in the standard session) to 1.00 (length 2) and 1.17 (length 3). There are similar increases in the other sessions. Hence, users can control the amount of dampening through experimentation with CGNRO

lengths. Fixing a key lowers the average chord distance to 0.74, 0.78 and 0.99 for CGNRO lengths 1, 2, and 3 respectively, which provides audible dampening.

The average distance of a chord from a focal note also increases with CGNRO length, but, as we can see in table 3, the focal mapping routine minimises this. We note, however, that focal mapping considerably increases the maximum distance of an output chord from the input chord, because the mapping can occasionally move chords a lot. We see that the average chord distance in the focal mapped session was similar to that in the standard session, but the maximum chord distances are higher, hence there is a higher likelihood that a striking chord change could happen in an episode with focal mapping. Users should therefore employ focal mapping carefully and monitor the chord distances. In future, we intend for GENRT to use distance measures when selecting CGNROs.

5 An Illustrative Example

GENRT is a python program which produces music in MIDI form. It handles timings by turning chord durations into MIDI ticks, and we chose 960 ticks per second, so that millisecond timings can be passed on to the MIDI with high accuracy. The system is in development, and is not sophisticated enough yet to produce particularly beautiful or innovative music. However, returning to the problem setting in section 1, our purpose here is only to show that (a) GENRT produces accurate soundtrack music for a video clip, in terms of the timing of episodes and events which match the media exactly (b) users have satisfactory levels of control over the atmosphere of the music in the episodes, and GENRT can smoothly change the mood over the period of an episode (or multiple episodes) and (c) particular emotional or situational events in the media are highlighted in audible ways with the generated soundtracks.

To test GENRT, we asked a trained musician (second author) with a background in composing soundtracks to create event-based music for a clip of the movie A Beautiful Mind [8]. This is a biographical drama that narrates the story of the mathematical genius John Nash. After making an incredible discovery, the protagonist is diagnosed with paranoid schizophrenia, and the story follows Nash on a journey of self-discovery, balancing mental health and his job as a cryptographer. The soundtrack is tasked with representing the workings of Nash's incredible mind, and its struggles. The composer of the movie's soundtrack, James Horner, represents Nash's genius with a series of beautiful and fast-moving chord progressions throughout the soundtrack. Music theorist Frank Lehman wrote about the representation of genius in A Beautiful Mind's soundtrack [10] and used NRT to analyse its chord progressions. Horner's chord progressions use chromatic triadicism, especially the L, R and S NROs (see table 1(a)), hence are difficult to accurately analyse using traditional music theory.

In the 2 minute, 26 second film clip we chose, John Nash's antagonist and his group of friends are playing Go, when they encounter Nash. The protagonist is challenged to play a game of Go against his adversary while the group of friends watches. Nash is sure he is going to win, and the game seems to suggest that

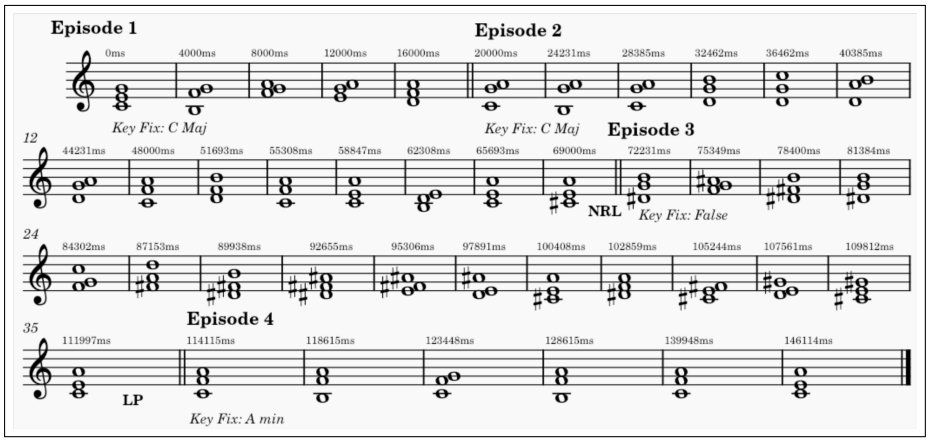

Fig. 1. GENRT chord sequence for the film clip from A Beautiful Mind, with chord start times, episodes, and the user-selected CGNRO for two transitions.

too, but to his surprise, he loses. There is a situational change in the clip to be highlighted in the soundtrack, at 69s, when the game starts, and an emotional change at 112s, when Nash loses and is antagonised by his opponent.

The musician employing GENRT used four different episodes for the soundtrack generation, each with different specifications to reflect the on-screen events and emotions. The first episode of 20s generates the music for the preamble to the scene. For this episode, the musician specified the starting chord of C major and further employed a fixed key of C Major for the chord sequence. This forced the music to be somewhat dampened, as the scene during this period is just a preparation for the game. There was no event set for this episode. The volume faded in slowly during this episode, setting an appropriate atmosphere and introducing the soundtrack. To strengthen the emphasis on the introduction setting the scene, the chord progression moves quite slowly at 4000ms per chord.

The second episode covers the period leading up to the game starting. To ensure continuity between the first and second episodes, the start chord duration was set to 4000ms, and increases to 3000ms during the episode, to raise the tension until the game starts. The key fix on C Major is continued, so the music is dampened and the change to the next episode, which generates the chord change for when the characters play the game, is audible. The event of this episode happens when the game starts, at 69,000ms, and the musician chose the mystery CGNRO (see table 1(b)) to reflect intrigue at the start of the game.

The third episode generates the music played during the Go game. The key fix functionality is turned off here so that the tool generates music which uses triadic chromaticism, using CGNROs up to length 3, which adds some disorienting tension during the game. As the game progresses, the music gets faster until the event of the episode at 112,000ms. This is when Nash loses the game, and is represented by the antagonism CGNRO. In the fourth episode, the musician

emphasised Nash's loss by turning key fix functionality back on and instantly reducing the tempo to much more slow-moving chords. As the scene comes to an end, the volume fades out to zero. The musician maintained consistency across episodes, with strings for chords, violin for the melody and cello for the bass. As per the original soundtrack, percussion was turned off. Each episode had chords fixed to focal note 48, to keep the music in a suitable pitch range.

The chord sequence and timings for the Beautiful Mind clip accompaniment are given in figure 1. We see there 41 chords and inspection of the start times of these shows that (a) the episode durations, situational change at 69s [chord 19] and emotional change at 112s [chord 35] were accurate to within a few milliseconds, and (b) the increase in tempo was smooth. On listening to the soundtrack, the music was more calm leading up to chord 19, more turbulent during the third episode and calmer towards the end of the soundtrack, as desired by the musician. The mystery CGNRO – reflecting the Go game starting – successfully interrupted the tonal chord progressions from the previous two episodes at chord 19, making the start of the game aurally noticeable. Chord change 35 (antagonism) was less prominent but still noticeable. Overall, the musician was satisfied with the accuracy, atmosphere and influence of the soundtrack.

6 Conclusions and Future Work

We have presented the GENRT system for automatically producing soundtracks for media such as video clips, and shown that it has promise for producing accurate, atmospheric and influential music. There are many next steps for this implementation, including: exposing more parameters for users to control the generation with; increasing the sophistication of melody generation and adding counterpoint production; and stronger dampening of the music between events. We also plan to combine this symbolic approach with deep learning so that a pre-trained neural model can be used to select musical sequences in terms of which fits best to a distribution, hopefully increasing the quality of the output. We may employ evolutionary and other search techniques to alter episode specifications (rather than musical choices), as part of this. Soundtracks often include musical *leitmotifs*, [3] [14], which consistently portray a character, and we intend to experiment with the generation and deployment of these in GENRT.

Once GENRT is somewhat more sophisticated, we will undertake a first round of user studies with novice composers, to assess its utility and future directions for its functionality. We also plan to carry out experiments on parameter tuning and checking correlation with other musical metrics in the output.

To describe and implement GENRT, we extended generative Neo-Riemannian Theory in various ways, including generalising the trichords which can be generated for a chord sequence. This required a number of adjustments in terms of the operation of the CGNROs. We also introduced and implemented ways in which the application of CGNROs to chord sequence generation could be dampened so that intermediate music was more in the background. This was in order to avoid obfuscating important chord changes which reflect events in the media.

As our requirements for GENRT increase, we will continue to develop the formalism to capture how generative NRT can be used to produce music. In particular, we intend to capture the interplay of chord sequences and leitmotifs with the formalism. We plan to involve GENRT in some multi-modal projects, where generative AI techniques are used to produce text and videos, with GENRT providing accompanying soundtracks. We are also interested in producing interesting and novel stand-alone music using GENRT. Our aim is to produce music that has direction and purpose, as with the best human-made compositions.

Acknowledgments

This work has been funded by the UKRI as part of the "UKRI Centre for Doctoral Training in AI and Music", under grant *EP/S022694/1*. We would like to thank the anonymous reviewers for providing helpful feedback.

References

1. Amram, M., Fisher, E., Gul, S., Vishne, U.: A transformational modified Markov process for chord-based algorithmic composition. Mathematical and Computational Applications **25(3):43** (2020)
2. Bernardes, G., Cocharro, D., Guedes, C., Davies, M.: Harmony generation driven by a perceptually motivated tonal interval space. Computers in Entertainment **14(2)** (2016)
3. Bribitzer-Stull, M.: Understanding the Leitmotif: From Wagner to Hollywood Film Music. Cambridge University Press (2015)
4. Cardinale, S., Colton, S.: Neo-riemannian theory for generative film and videogame music. In: Proc. of the Int. Conference on Computational Creativity (2022)
5. Chuan, C. H., Chew, E.: Generating and evaluating musical harmonizations that emulate style. Computer Music Journal **35**, 64–82 (2011)
6. Cohn, R.: Neo-Riemannian operations, parsimonious trichords, and their Tonnetz representations. Journal of Music Theory **41**(1), 1–66 (1997)
7. Eladhari, M., Nieuwdorp, R., Fridenfalk, M.: The soundtrack of your mind: Mind music - adaptive audio for game characters. In: Proc. ACM SIGCHI International Conference on Advances in Computer Entertainment Technology (2006).
8. Howard, R.: A Beautiful Mind, Universal Pictures (2002)
9. Hutchings, P. E., McCormack, J.: Adaptive music composition for games. IEEE Transactions on Games **12**(3), 270–280 (2020)
10. Lehman, F.: Transformational Analysis and the Representation of Genius in Film Music. Music Theory Spectrum **35**(1), 1–22 (2013)
11. Lehman, F.: Film music and Neo-Riemannian Theory. Oxford Handbook (2014)
12. Monteith, K., Francisco, V., Martinez, T., Gervás, P., Ventura, D.: Automatic generation of emotionally-targeted soundtracks. In: Proceedings of the International Conference on Computational Creativity (2011)
13. Monteith, K., Martinez, T., Ventura, D.: Automatic generation of music for inducing emotive response. In: Proceedings of the International Conference on Computational Creativity (2010)
14. Whittall, A.: Leitmotif. Oxford University Press (2001)
15. Wiggins, G.A.: Automated generation of musical harmony: What's missing. In: Proceedings of the International Joint Conference on Artificial Intelligence (1999)