Learning and Creating Novel Harmonies in Diverse Musical Idioms: An Adaptive Modular Melodic Harmonisation System

Maximos Kaliakatsos-Papakostas
Department of Music Studies,
Aristotle University of Thessaloniki,
54124, Thessaloniki, Greece.
email: maxk@mus.auth.gr

Dimos Makris
Department of Informatics Studies,
Ionian University,
49100, Corfu, Greece.
email: c12makr@ionio.gr

Costas Tsougras
Department of Music Studies,
Aristotle University of Thessaloniki,
54124, Thessaloniki, Greece.
email: tsougras@mus.auth.gr

Emilios Cambouropoulos
Department of Music Studies,
Aristotle University of Thessaloniki,
54124, Thessaloniki, Greece.
email: emilios@mus.auth.gr

Abstract

Melodic harmonisation is a sophisticated creative process that involves deep music understanding and specialised music knowledge relating to melodic structure, harmony, rhythm, texture, form. In this paper a new melodic harmonisation assistant is presented that is adaptive (learns from data), general (can cope with any tonal or non-tonal harmonic idiom) and modular (learns different aspects of harmonic structure such as chord types, chord transitions, cadences and voice-leading). This melodic harmonisation system can be used, not only to mimic given harmonic styles, but, to generate novel harmonisations for diverse melodies by exploring the harmonic possibilities provided by the implied harmonies of input melodies, or by allowing the imposition of user-defined chord constraints leading thus to new unforeseen harmonic realisations. The various components of the proposed model are explained, and, then, a number of creative harmonisations of different melodies are presented, along with an intuitive statistical analysis, to illustrate the potential of the system.

1 Introduction

Creative music systems are often criticised as not 'really' being creative *per se*; underlying this criticism is the belief that the actual human programmer is the true source of creativity. However, machine learning has made such criticisms more difficult to maintain, as a machine may acquire knowledge from data, construct a new conceptual space (new structural relations or even new elements) or explore an existing one without human intervention and, then, create new unforeseen output (Wiggins et al. 2009). Adaptability, flexibility, independence, autonomy are features associated with creativity (see key components of creativity in Jordanous (2013)); general representations and machine learning techniques allow creative systems to be open to new environments, to evolve, to transform existing or construct new concepts, to create new unexpected results.

A model of creativity has been proposed by Boden (2009) whereby a conceptual space may be explored by an agent in order to generate new creative instances (exploratory creativity) or the rules of the conceptual space are transformed changing the conceptual space itself (transformational creativity) or different conceptual spaces that share structural similarities are combined to create new blended spaces (combinational creativity). In the current study, the conceptual spaces are learned in a bottom-up fashion from data and are structured in a modular way, so as to allow (at a later stage) to combine different modules from different spaces creating thus new blended spaces. At this stage, the system is indicated to exhibit exploratory creativity, by composing harmonies that potentially excess the harmonic borders of a corpus.

This paper focuses on melodic harmonisation seen as a creative musical act. Some researchers follow the knowledge-engineering approach whereby experts encode the essential rules for harmonisation in a certain tonal style (from the Bach chorale expert system by Ebcioglu (1988) to the explicitly structured knowledge paradigm by Phon-Amnuaisuk and Wiggins (1999); Phon-Amnuaisuk et al. (2006)). In recent years, more attention has been given to probabilistic approaches that learn harmony from music data, using techniques such as Hidden Markov Models, N-grams, probabilistic grammars, inductive logic programming (Steedman (1996); Rohrmeier (2011); Conklin (2002); Scholz et al. (2009); Pérez-Sancho et al. (2009); Raphael and Stoddard (2004); Whorley et al. (2013); Dixon et al. (2010); Granroth-Wilding and Steedman (2014)). Such models automatically derive harmonic structure from training data and are thus more flexible than rule-based systems; however, they are applied usually to very narrow well-circumscribed tonal styles (e.g. Bach chorales or hymns or blues harmonies) and they generate acceptable harmonic progressions only within the corresponding learned harmony (cannot create new harmonies beyond the learned space).

This paper describes a creative melodic harmonisation system that can assist a user in generating new conventional or unconventional sequences of chords for a given melody. The creative melodic harmonisation assistant is based on a novel *General Chord Type* representation (Cambouropoulos et al. 2014) that allows the system to extract appropriate chord types from diverse harmonic idioms that comply with the traditional Western well tempered scale. For any harmonic idiom, the system learns from a set of pieces (harmonic reductions) the chord types that are relevant for the specific style, extracts probabilistic information on the most likely chord transitions (first-order transition tables), examines

2

phrase endings with a view to establishing common endings/cadences, and learns basic features of voice leading (bass movement in relation to melodic motion, chord inversions and omission/duplication of chord notes). This information constitutes a conceptual space that characterises a specific harmonic idiom and is used to create original harmonisations for new previously unseen melodies.

Apart from learning harmony from a particular harmonic style and producing new harmonisations in this style, the current paper explores other creative aspects of the proposed melodic harmonisation assistant that diverge from a learned harmonic style (not automated in this phase). Firstly, a user may assign particular chords to specific melodic notes of a given melody, thus 'forcing' the system to explore harmonic regions of the learned harmonic space that are less common (or even alien) thus giving rise to potentially unexpected harmonisations, expressed as chord sequence paths that accommodate the selected chords. Secondly, a user may choose to harmonise a melody with different potentially incompatible learned harmonic styles (e.g. traditional folk melody with tonal harmony or octatonic harmony etc); potential inconsistencies are dealt with manually at this stage (automation of such processes is under development). The ultimate goal of this research is to enable a system to create original harmonisations by combining harmonic components of different harmonic spaces; such creative blending aspects are explored in Zacharakis et al. (2015); Cambouropoulos et al. (2015) and is part of ongoing research.

The proposed melodic harmonisation assistant is original in the following ways:

1. harmony is learned in an idiom-independent manner (i.e., harmonic features are acquired via machine learning for various tonal and non-tonal systems);

2. the system allows the exploration of a learned harmonic space by user-defined intermediate chords that may lead the system outside its expected course;

3. the creative system can use existing harmonic styles to harmonise melodies with 'incompatible' harmonic outlook.

In the following sections the proposed modular probabilistic melodic harmonisation system is presented; this system is able to learn different harmonic aspects (chord types, chord progressions, cadences, voice-leading) from practically any musical idiom and can use the acquired harmonic knowledge to harmonise novel melodies in innovative ways. The next section provides a short discussion of previous approaches to melodic harmonisation and an overview of the proposed system. Then, Section 4 analyses the module for constructing chord sequences by automatically employing cadences and allowing user-defined chord constraints. The module for fixing the voicing layout of chords is presented in Section 5 and finally several examples of melodic harmonisations in diverse harmonic idioms are given, along with an intuitive statistically-based analysis in Section 6.

2 Melodic harmonisation: related work and overview of the proposed system

Among the first approaches for capturing the characteristics of harmony in automated melodic harmonisation were ones that incorporated human expert knowledge (e.g. Ebcioglu (1988)) encoded in the form of rules, leading to systems that could harmonise melodies according to explicit stylistic directives. For a review of rule–based systems the reader is referred to Pachet and Roy (2001). A similar approach to rule–based methodologies is the one followed by systems that utilize genetic algorithms (GA), like the ones briefly reviewed by Donnelly and Sheppard (2011) and, also, in Phon-Amnuaisuk and Wiggins (1999). The similarity between these two approaches is that both rely on a set of harmonic rules intended for a specific musical idiom; in the case of the GAs, the employed fitness function quantifies such rules. The advantage of rule–based systems is that they can capture the hierarchical structure of complex musical idioms, e.g., by using grammar-related structures for tonal (Rohrmeier 2011; Koops et al. 2013) or especially focussed on jazz (Granroth-Wilding and Steedman 2014) music.

However, the melodic harmonisation methodologies that utilise rule-based techniques have a major drawback when dealing with melodic harmonisation in many diverse idioms: the encoding of rules that describe even a simple musical idiom is not always a realizable task, since idioms abound in complex and often contradicting interrelations between harmonic elements. In order to overcome such shortcomings, the formulation of *probabilistic* techniques and *statistical learning* has been proposed. Probabilistic techniques can, on the one hand, be trained on any musical idiom, given a set of harmonically annotated pieces, while on the other hand they encompass the possibility to take 'unusual' decisions if necessary – e.g. if the melody's implied harmony diverges from the learned harmony. Among many proposed methodologies, Bayesian networks (Suzuki 2013) and prediction by partial matching (Whorley et al. 2013) have been utilised to construct the bass, tenor and alto voices below a given soprano voice; hidden Markov models (HMMs) for constructing chord sequences for a given melody (Raczyński et al. 2013); and probabilistic graphical models for similar chord-assignment tasks (Paiement et al. 2006).

The main drawback of probabilistic methodologies, especially HMMs, is that they do not capture larger scale dependencies between remote harmonic parts (Pachet et al. 2011). For instance, phrase endings, indicated by cadences, are very distinctive parts of higher-level harmonic structure that are not captured by methodologies that concern chord-to-chord harmonic progressions. Cadences have been studied under different contexts in the computational harmonisation literature. For instance, in Borrel-Jensen and Hjortgaard Danielsen (2010), a methodology based on cadences was utilised to evaluate the outcomes of an automatic melodic harmonisation system. The methodologies presented in Allan and Williams (2004) and Hanlon and Ledlie (2002) utilise a backwards propagation of the hidden Markov model (HMM) methodology, starting from the end and constructing the chord progression in a backwards fashion, highlighting the role of the cadence in reflecting structure. In Yi and Goldsmith (2007) a probabilistic system was presented that rewarded those chord sequences that ended with a perfect cadence, while in Yogev and Lerch (2008) a probabilistic methodology that identifies the probable positions of cadences was introduced. Special consideration of cadences has also been employed

4

in HMM-based methodologies, either by assigning an additional layer of probabilities for the final chords of sequences (Simon et al. 2008) or by fixing the ending or intermediate chords in probabilistically produced chord sequences (Kaliakatsos-Papakostas and Cambouropoulos 2014).

The architecture of the proposed system incorporates a simple statistical approach for preserving structural relations between remote harmonic parts, while at the same time diverse harmonies can be learned from data. Therefore, the merits of rule-based systems are preserved by learning and automatically employing intermediate and final cadences, leading to harmonisations that are structurally consistent. Additionally, the probabilistic nature of the incorporated algorithms allows for radically diverse harmonic idioms to be learned, while the generated harmonisations reflect the characteristics of learned idioms in terms of chord transitions and voicing layout. An additional advantage of the presented system is the fact that the output is a harmonic realisation with actual chord notes (not only chord symbols).

The presented harmonic learning system is trained independently on several harmonic aspects that are divided in two groups: chord generation and the voicing layout. Figure 1 illustrates this setting, where 'GCT generation' on the left block refers to the generation of chords symbols in the General Chord Type (GCT) representation (Cambouropoulos et al. (2014); Cambouropoulos (2015); see brief description in the next section), while the right block refers to the translation of GCT chords to actual music by assigning proper voicing layouts, converting the final output to MIDI notes. The oval blocks refer to modules that have been trained from data. The arrow leading from the 'GCT generation' to the 'GCT to MIDI pitches' block indicates the current generative process workflow of the melodic harmoniser: first, chord sequences in GCT form are produced and, afterwards, voicing layout is applied to the composed GCT sequences, providing the finalised output in MIDI format. In turn, the bass voice motion is first defined for the given GCT sequence and the given melody and, afterwards, the intermediate chord notes between bass and melody are fixed.

Both the 'GCT generation' and the 'GCT to MIDI pitches' blocks include modules that learn from data, giving the system the ability to express the characteristics of each learned idiom on several harmonic aspects. The GCT generation block incorporates three learning modules: (a) the 'Chord types' module which learns chord types by converting the pitches of the training harmonies to GCTs and organising them into chord type categories; (b) the 'Cadence constraints' module that learns and assigns cadences to user-defined positions of phrase endings (giving an essence of high-level structure); and (c) the constraint hidden Markov Model (cHMM) (Kaliakatsos-Papakostas and Cambouropoulos 2014) that learns first-order GCT chord transitions and performs probabilistic harmonisation given the aforementioned cadence constraints as well as user-defined chord constraints. The 'GCT to MIDI pitches' block includes the following learning modules: (a) the 'Bass Voice Leading' module that defines the motion of the bass in relation to the melody; (b) the 'bass-to-melody distances' that learns statistics about the distances between the bass and the melody for each idiom; and (c) the 'Chord inversions' module that learns statistics about the inversions of the learned GCT chords. The aforementioned voice-related modules contribute to the definition of the bass voice and afterwards, a simple algorithmic process, namely the 'GCT voicing layout' module, defines the chord notes between the bass and the melody.
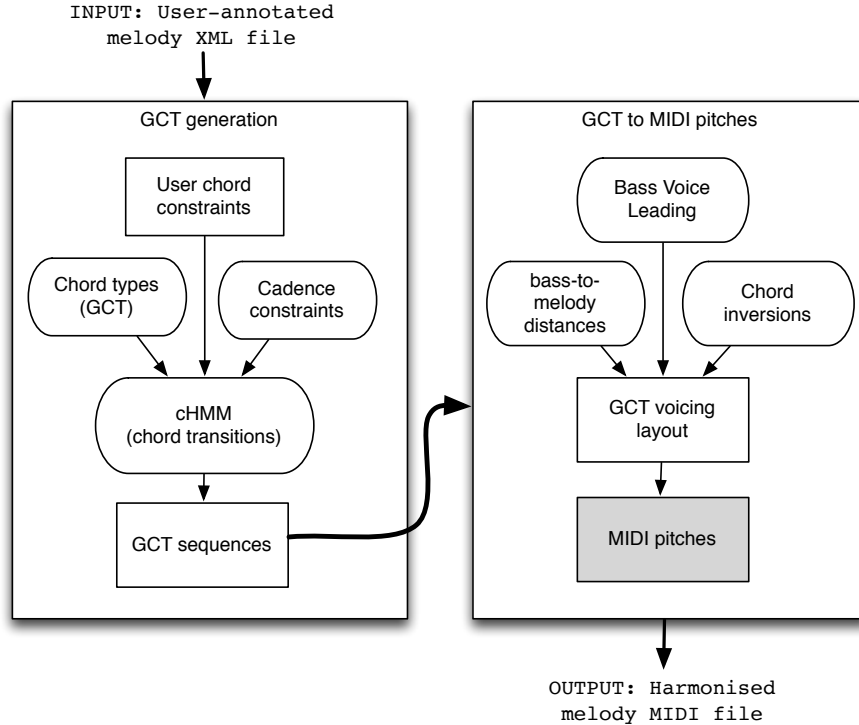
Figure 1: Overview of the Modular Melodic Harmonisation system. Oval blocks indicate modules that learn from data.

3 Chord representation and data input for training and generating

The system learns a given harmonic content given through annotated training data, while it produces new harmonisations according to guidelines provided in the melody input file. Since the processes of training and composing incorporate many diverse musical idioms, the system learns the available chord types therein (according to their root notes) based on the *General Chord Type* (GCT) (Cambouropoulos et al. 2014) representation. The training data include the notes on a level of harmonic reduction (manually annotated reductions), where only the most important harmonic notes are included, while additional layers of information are given regarding the tonality and the metric structure of each piece. Accordingly, the format of the user melody input file includes indications of several desired attributes that the resulting harmonisation should have. The chord representation scheme, the format of the training data and the user melodic input file are analysed in the following subsections.

3.1 Representation of harmony in diverse idiom with the General Chord Type encoding

The *General Chord Type* GCT provides accurate harmonic representation in the sense that it encompasses all the pitch-class-related information about chords. At the same time,

6

for every pitch class simultaneity the GCT algorithm rearranges pitch classes so that it identifies a root pitch class and a chord 'base' (which can be considered as a basic type, e.g. major or minor) and 'extension' (which give information about chord extensions, e.g. seventh or sixth, etc.), leading to chord representations that convey musical meaning for diverse music idioms. The GCT representation has common characteristics with the stack-of-thirds and the virtual-pitch-root-finding methods for tonal music, but has differences as well (see Cambouropoulos et al. (2014)). This encoding is inspired by the standard Roman numeral chord type labelling, but is more general and flexible. A recent study (Kaliakatsos-Papakostas et al. 2015) on the GCT representation indicated that it can be used both as a means to represent harmonic chords and to describe musically meaningful relations between different harmonic labels in diverse and not necessarily tonal music idioms (Cambouropoulos et al. 2014; Kaliakatsos-Papakostas et al. 2014b; Cambouropoulos 2015).

The GCT algorithm computes, for a given multi-tone simultaneity, the 'optimal' ordering of pitches such that a maximal subset of consonant intervals appears at the 'base' of the ordering (left-hand side) in the most compact form; the rest of the notes that create dissonant intervals to one or more notes of the chord 'base' form the chord 'extension'. Since a tonal centre (key) is given, the position within the given scale is automatically calculated. Input to the algorithm is the following:

- Consonance vector: a Boolean 12-dimensional vector is employed indicating the consonance of pitch-class intervals (from 0 to 11). E.g., the vector $[1, 0, 0, 1, 1, 1, 0, 1, 1, 1, 0, 0]$ means that the unison, minor and major third, perfect fourth and fifth, minor and major sixth intervals are consonant; dissonant intervals are the seconds, sevenths and the tritone; this specific vector is referred to in this article as the 'tonal consonance vector'.
- Pitch Scale Hierarchy: is given in the form of scale tones and a tonic. E.g., a D major scale is given as: $2, [0, 2, 4, 5, 7, 9, 11]$, or an A minor pentatonic scale as: $9, [0, 3, 5, 7, 10]$
- Input chord: list of pitch classes (MIDI pitch numbers modulo 12).

For instance, the tonic chord is labeled as $[0, [0, 4, 7]]$, where the first occurence of 0 denotes the root of the chord in relation with the scale's tonic and the base, $[0, 4, 7]$, denotes the maximally consonant setup of the included pitch classes. In relation to the tonal naming of chords, type $[0, 4, 7]$ is a major chord. Similarly the dominant seventh (inverted or not) is labeled as $[7, [0, 4, 7], [10]]$, where there is a third element, $[10]$, which is an extension, i.e. an existing pitch class that cannot be inserted in the maximally consonant set. The *compressed* GCT form will be sometimes used in this paper, where no intermediate brackets are used, e.g. $[7, [0, 4, 7], [10]]$ will be denoted as $[7\ 0\ 4\ 7\ 10]$. An example taken from Beethoven's *Sonata no. 14, op.27-2* (Figure 2) illustrates the application of the GCT algorithm for different consonance vectors. For the tonal vector, GCT encodes classical harmony in a straightforward manner. This way we have an encoding that is analogous to the standard Roman numeral encoding (Figure 2, 'tonal'). If the tonal context is changed to a chromatic scale context and all intervals are considered equally 'consonant', i.e., all entries in consonance vector are 1s, we get the second 'atonal' GCT analysis (Figure 2, 'atonal') which amounts to normal orders (not prime forms) in standard pc-set analysis. In pitch class set theory normal orders do not have 'roots' – however, they have transposition values (T0-T11) in relation to a reference pc (normally pc 0); the GCT for the 'atonal'

consonance vector is equivalent to the normal orders with transposition values of pc-set theory.

	i	i2	VI	IIN6	V7	i64	V7	i
c#:								
tonal	0,[0,3,7]	0,[0,3,7,10]	8,[0,4,7]	1,[0,4,7]	7,[0,4,10]	0,[0,3,7]	7,[0,4,7,10]	0,[0,3,7]
atonal	0,[0,3,7]	10,[0,2,5,9]	8,[0,4,7]	1,[0,4,7]	5,[0,2,6]	0,[0,3,7]	5,[0,2,6,9]	0,[0,3,7]

Figure 2: Beethoven, *Sonata no. 14, op.27-2* (reduction of first five measures). Top row: Roman numeral harmonic analysis; middle row: tonal GCT analysis; bottom row: atonal GCT analysis. The tonal GCT successfully encodes all chords, including the Neapolitan sixth chord (fourth chord).

An additional fundamental concern of the harmonic representation in the presented harmoniser is the grouping of chords according to their GCT representation with a methodology described in Kaliakatsos-Papakostas et al. (2015). For example, the V chord in a scale can be expressed either as $[7, [0, 4, 7]]$ or in a 'reduced' ($[7, [0, 4]]$) or an 'expanded' ($[7, [0, 4, 7, 10]]$) forms, that actually represent the same chord label. Each GCT group includes the GCT types that satisfy the following three criteria:

1. they have the same scale-degree root;
2. their GCT bases are subset-related; and
3. they both contain notes that either belong or not to the given scale (see Table 1 for details).

Regarding criterion 2, two bases B_1 and B_2 are considered subset-related if $B_1 \subseteq B_2$ or $B_2 \subseteq B_1$, e.g. $[0, 4] \subseteq [0, 4, 7]$ while $[0, 4] \not\subset [0, 3, 7]$. Criterion 3 is utilised to identify and group together chords that belong to secondary tonalities within the primary tonality of the piece. For instance, in a diatonic major context, while $c_1 = [0, [0, 4, 7]]$ and $c_2 = [0, [0, 4, 7, 10]]$ fulfil criteria 1 and 2, according to criterion 3 they are not grouped together since c_2 includes value 10, which is mapped to the non-diatonic 10 pitch class value. In a major context $[0, [0, 4, 7, 10]]$ is secondary dominant to the IV (V/IV) and is differentiated from the I major chord.

Furthermore, each group is represented by an 'exemplar' GCT type, which is the one that is more often met in the datasets under study. Some common chord groups in the major scale Bach chorales are illustrated in Table 1. This table also includes the functional naming of each group in order to assist the comparison of the derived GCT types with the standard Roman-numeral labelling. Testing this simple algorithm on sets of both major and minor Bach chorales gives a reasonable first classification of the 'raw' GCTs. Groups of GCT chords are extracted from datasets as explained in Section 3.2 and their exemplars are used to train the system.

8

Table 1: Four tonal chord groups and their exemplar GCTs. The group $[0, [0, 4, 7]]$ has been separated from the group $[0, [0, 4, 7], [10]]$, due to the non-diatonic pitch class 10 of the latter.

functional name	exemplar	Group members			
tonic	$[0, [0, 4, 7]]$	$[0, [0, 4, 7]]$	$[0, [0, 4]]$	$[0, [0, 4, 7], [11]]$	
dominant	$[7, [0, 4, 7]]$	$[7, [0, 4, 7]]$	$[7, [0, 4, 7], [10]]$	$[7, [0, 4], [10]]$	$[7, [0, 4]]$
subdominant	$[5, [0, 4, 7]]$	$[5, [0, 4, 7]]$	$[5, [0, 4]]$	$[5, [0, 4, 7], [11]]$	
V / IV	$[0, [0, 4, 7], [10]]$	$[0, [0, 4, 7], [10]]$	$[0, [0, 4], [10]]$		

3.2 Training data and harmony annotations

The development of the presented melodic harmoniser incorporates statistical learning on different harmonic levels (chord transitions, cadences and voice leading) from a data pool with 'real-world' representations of historical traditions of music creation. By employing rich multi-level structural descriptions of harmony in different idioms, the harmoniser is able to create new music that accurately reflects the characteristics of these idioms. A diverse collection of musical pieces drawn from different historic eras and from different harmonic styles has been assembled by music experts. Each idiom/style is internally as coherent as possible such that regularities of the specific harmonic space can be extracted; the collected idioms are as different as possible on all the examined harmonic levels. Additionally, the musical pieces are manually annotated such that structural harmonic features may be extracted at various hierarchic levels. Specifically, the following structural aspects are manually annotated: (a) harmonic reduction(s) of each musical work/excerpt so that structural harmonic/non-harmonic notes are explicitly marked; (b) local scale/key changes are determined so that harmonic concepts relating to modulations can be learnt; and (c) grouping structure is given so that cadential patterns at various hierarchic levels can be inferred.

An example of the required types of information from a music piece for training the system are illustrated in Figure 3; annotated music files include: a) the original musical data the actual musical surface and b) expert annotations that are provided as manually entered analytical information about the contents. At the lowest level of the *musical surface* (denoted as ms_0), which is the actual notes of a musical piece and the lowest level of representation that has musical significance (Jackendoff 1987), a custom text-based encoding is used. *Expert annotations* in a music piece describe musicological aspects that refer to specific analytic concepts (e.g., the use of harmonic symbols to describe note simultaneities, modulations, phrasing etc.). Specifically, the expert annotations are given in musical form and include time-span reduction of the original content (ms_1) and annotations concerning *tonality* (and tonality changes) and *grouping* information.

On the chord transitions level the system is trained according to the chord progressions on the harmonic reduction (ms_1), with chords being encoded in the General Chord Type (GCT) (Cambouropoulos et al. 2014) representation. Since the GCT requires tonality information, the GCT forms of the extracted chords are computed by using the tonality annotations. Annotations of grouping indicate the positions of cadences, where the system learns the final pairs of chords before any group ending. Even though a cadential form may incorporate more or fewer than two chords, considering the last two chords of a phrase as

a cadence was decided as a rational compromise.

The indicators of the *tonality* – and the tonality changes – include accidentals in chordal form, with all the included notes indicating an octave of the utilised scale (lowest note is the tonality's fundamental), while the time instance of a tonality activation/change is defined by the indication's onset. Additionally, it has to be noted that at least one tonality indicator at the beginning of the piece is required otherwise the tonality annotations of the piece are considered absent (repetitions of the same indicator are ignored). The *grouping* part contains annotations about melodically coherent temporal regions of the music surface. At the beginning of each phrase, a group identifier is placed indicating the level of the phrase hierarchy. One note on any line indicates the lowest level groupings (e.g. phrase); two notes on the lowest two lines indicate an immediately higher-level of grouping (e.g. related phrases in a row); three notes indicate even higher level of grouping and so on. The cadences of each grouping level, i.e. the final pair of chords at the end of each grouping part, are learned separately.

Figure 3: Annotated file containing original song transcription (ms_0), time-span reduction of the original content (ms_1), as well as *tonality* and *grouping* information.

The dataset consists of over 430 manually annotated musicXML documents categorised in 7 categories and various subcategories. The separation of pieces in sets primarily focuses on genre categorisation, while subcategories are created within genres that present notable differences in their harmonic structure. The diversity in harmonic features among different sets and subcategories allows the inclusion of a wider spectrum of options, enabling the melodic harmoniser to produce harmonisations with strong references to diverse idioms. On the other hand, there is intra-idiom consistency in each subcategory of pieces, which is expressed by 'patterns' in harmonic features that are characteristic to this subcategory, in a sense that these features are often encountered in several pieces within this idiom.

The dataset comprises seven broad categories of musical idioms, further divided into sub-categories, and presented in the following list[1]:

---

[1]Categories 4, 5 and 6 may seem to overlap, but they are essentially different: category 4 includes harmonisations of initially monophonic folk melodies made by art music composers of European National Schools, category 5 comprises 20th-century original compositions (not based on folk songs) and category 6 contains

1. Modal harmonisation in the Middle Ages (11th – 14th centuries): includes subcate-gorys of medieval pieces in the Organum and Fauxbourdon styles.

2. Modal harmonisation in the Renaissance (15th – 17th centuries): includes modal music from the 16th – 17th centuries along with modal chorales.

3. Tonal harmonisation (17th – 19th centuries): includes a set of the Bach Chorales, the Kostka-Payne corpus[2] and tonal harmonisation sets from the 18th – 19th centuries.

4. Harmonisation in National Schools (19th – 20th centuries): includes 19th – 20th century harmonisation of folk songs from Norway, Hungary and Greece.

5. Harmonisation in the 20th century: includes harmonisations of Debussy, Hindemith, Whitacre, Stravinsky and Schnittke among others.

6. Harmonisation in folk traditions: includes Tango (classical and nuevo styles), Epirus polyphonic songs and Rebetiko songs.

7. Harmonisation in 20th-century popular music and jazz: includes mainstream jazz, pieces from Bill Evans and a collections of songs from The Beatles.

For the harmonisation examples incorporated in the present paper, a subset of eight harmonic idioms was used from the dataset, presented in the following list:

1. The 15th-century Fauxbourdon style, based on parallel $\frac{6}{3}$ chords.

2. The homophonic tonal harmonic idiom of J. S. Bach chorales.

3. The Kostka-Payne corpus, describing mainstream tonal harmony of the 18th/19th-centuries (Kostka and Payne 2004).

4. Edvard Grieg's 19th-century chromatic harmonic idiom, as expressed in his folk songs harmonisations op. 17 & 63.

5. The Epirus polyphonic singing style, based on the minor pentatonic scale (Lolis 2006; Kaliakatsos-Papakostas et al. 2014b).

6. Yannis Constantinidis's 20th-century modal style, as encountered in his '44 Greek miniatures for piano' (Tsougras 2010).

7. Paul Hindemith's 20th-century harmonic idiom, as expressed in his 'Six Chansons'.

8. Mainstream jazz harmony, as encountered in selected jazz standards (tonal or modal) from the *Real Book*.

---

original harmonisations embedded in the folk idioms.

[2]This dataset consists of the 46 excerpts that are longer than 8 measures from the workbook accompanying Kostka and Payne's theory textbook Tonal Harmony, 3rd edition (Kostka and Payne 2004) and is available in machine readable format at `http://theory.esm.rochester.edu/temperley/kp-stats/index.html`.

3.3 Melodic input

After the system is trained, it is able to harmonise a given melody. Figure 4 illustrates an instance of the input protocol for the system, which includes the melody to be harmonised and information regarding some harmonic attributes that are not inferred by the system at this stage. The input melody, in this stage, is manually annotated as to harmonic rhythm, harmonically important notes, key and phrase structure. The file that produced this figure is used as input for harmonising the example in Figure 7 (b). Initially, the user provides the positions where chords should occur (harmonic rhythm), as well as the important notes (harmonic notes) that should be considered with higher weight when selecting chords for each segment. If the user provides no information for these attributes, the system produces default harmonic rhythm and important note selection schemes that might lead to 'unwanted' harmonic results. Additionally, the user has the freedom to choose specific chords at desired locations (constraint chords), forcing the system creatively to produce chord sequences that comply with the user-provided constraints, therefore allowing the user to 'manually' increase the interestingness of the produced output. Finally, the user should accompany the melody with higher level harmonic information concerning the tonality or tonalities of the piece, as well as with its phrase grouping boundaries. Tonality is indicated by a cluster of all notes included in the scale, with the lowest note indicating the tonality's tonic. Grouping is annotated by arbitrary notes at the metric position where grouping changes occur, while the number of notes in these positions indicate the grouping level of the phrase.

Figure 4: Example of a user melodic input file. This melodic part is from L. v. Beethoven, (b. 1-8), second movement in A♭ major of the Piano Sonata no. 8. This input file is used for the harmonised example in Figure 7 (b).

Why is tonality among the features that are specified by the user along with the melody? Although the local tonality of a melody could be automatically deduced algorithmically (Chai 2005; Kaliakatsos-Papakostas et al. 2013), manual annotation of tonality and changes has been decided for the following reasons:

1. Utilisation of non-standard (major/minor) tonalities: The collected dataset include

pieces that do not conform to the standard Western music tonalities, e.g., there are pentatonic or octatonic modes. Additionally, the user is allowed to specify any desirable tonality, which will lead the system to select the proper set of chords to harmonise the given melody.

2. Accuracy in tonality-change boundaries: Algorithms that perform melodic segmentation according to tonality (Chai 2005; Kaliakatsos-Papakostas et al. 2013) are not able to identify the exact location of tonality boundaries. For the presented melodic harmoniser, it is important that the tonality (and phrase level) change locations stay aligned with the melody segments that a human user indicates.

3. The ability to insert 'subtle' tonalities: The user is able to introduce tonality changes in places where an algorithm might not identify any change. This ability introduces additional agility and potential of variety to the system.

In the training data, tonality changes are treated differently in different idioms, while, additionally, some idioms do not include (or include very specific) modulations between certain – neighbouring in the circle-of-fifths – tonalities. Since modulations are dependent on the melody, and a user input melody might incorporate arbitrary modulations, it is clear that no learning strategy on every idiom could cover the entire spectrum of modulations that are possible for input melodies. For instance, in the idiom of modal music there are no modulations, since entire pieces are composed in a single mode. Therefore, it would be impossible to harmonise a melody that incorporates modulations using the harmony of a modal idiom, since no training paradigms would be available for such a task. For the purposes of the 'idiom independent learning' that is required for the presented system, modulations are not tackled: a cadence in the first tonality is assigned before the modulation occurs and the material after the modulation is treated as a new phrase in the new tonality.

4 Chord progressions, intermediate constraints and cadences

The core of the generative process is the production of GCT chord progressions with a probabilistic methodology that is a simple extension of the hidden Markov model (HMM) method that allows the inclusion of fixed 'anchor' chords. Harmonisation with fixed anchor chords is considered a crucial component of the presented work, since it enables the prior definition of important chords in intermediate positions of the melody to be harmonised. Two types of important chords (or pairs of chords in the case of cadences) are considered: (a) intermediate or final cadences at positions where phrases end and (b) user-defined fixed chords that the system is forced to use. For the pieces used to train the system, the format of which is described in Section 3.2, annotations about phrase boundaries are also included. During training, the final pair of chords (penultimate and final chord) in each phrase is independently stored in the cadence module of the system, wherein the probabilities of intermediate and final cadences are calculated. In addition to the indicated positions of phrase endings, the user is also able to assign specific desired chords at any desired position, directly allowing the involvement of the user's creativity in the harmonisation process. Both the phrase ending positions and the user-defined chords are included in the

directions provided by the user in the melody input file, as described in Section 3.3. These chords act as 'anchors' that remain fixed for the constrained HMM (cHMM) (Kaliakatsos-Papakostas and Cambouropoulos 2014) algorithm that selects *'proper'* chord sequences connecting the intermediate parts between the fixed chords, under the conditions introduced by the melodic material to be harmonised. The results presented in Kaliakatsos-Papakostas and Cambouropoulos (2014) indicate that cHMMs produce harmonisations that are potentially completely different to the ones produced by HMMs, depending on the imposed constraints.

4.1 Probabilistic generation of chord progressions with intermediate constraints

The proposed harmoniser uses the cHMM (Kaliakatsos-Papakostas and Cambouropoulos 2014) algorithm for generating chord progressions. The aim of this algorithm is to preserve the merits of probabilistic harmonisation, i.e., ability to train on different idioms and flexibility in generation, while allowing prior determination of intermediate chords (also named as *checkpoints* in the literature; see Chuan and Chew (2007)). Such constraints in the context of Markov chains (with no demands imposed by observations) are also know as 'unary' constraints (Pachet et al. 2011), however the cHMM algorithm works under the assumption sequences of states (chords) are composed given a set of observations (melody). Allowing fixed intermediate chords introduces two advantages for the presented harmoniser: (a) the preservation of higher level harmonic structure by the imposition of intermediate and final cadences and (b) the interactivity with the user by allowing any desired chord to be placed at any position. In the case of the cadences, the intermediate chords that comprise the cadence are specified by a probabilistic algorithmic process described later, that captures statistics about cadence occurrences either in intermediate phrase endings or at the end of the piece, allowing the learning of music structure on a higher hierarchical level. Direct human intervention on selecting desired chord constraints in the cHMM algorithm allows the presented harmoniser to function as a melodic harmonisation assistant that allows its user to specify a harmonic 'spinal chord' of anchor chords that are afterwards connected by chord sequences that give stylistic reference to a learned idiom.

The cHMM methodology divides the problem of finding intermediate constraints (i.e. fixed chords specified by the user or by the cadence module) into several consecutive problems of finding boundary constraints, i.e. fixed beginning and ending chords. Table 2 illustrates this process, where the intermediate chord constraints (I_j) are preserved while new chords (C_i^j) are generated, given the melody notes (m_i). The problem of assigning intermediate chord constraints is transformed into the problem of finding consecutive beginning and ending chords for each intermediate segment. In Simon et al. (2008), the HMM variation that was presented included an additional layer of probability distributions for beginning and ending chords for harmonising a part. In the cHMM methodology, used in the presented harmoniser, the probability values in the distributions for beginning and ending chords in each intermediate segment are actually binary: the chord that is selected as constraint has probability value 1, while all the others have 0.

During the cHMM training phase, an initial set of music phrases is considered which provides the system with the required statistical background, constituting the training set.

Table 2: The melody notes (m_i) need to be harmonised according to the intermediate chord constraints (I_j). The cHMM algorithm breaks the problem into two boundary constraints problems and composes the most probable sequence of chords (C_i^j) according to the observed melody, transition probabilities and given constraints.

	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8
melody notes	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8
intermediate constraints	I_1				I_2			I_3
			C_1^1	C_2^1	C_3^1	C_1^2	C_2^2	

$$\underbrace{\qquad\qquad}_{\text{boundary constraints } 1} \quad \underbrace{\qquad\qquad}_{\text{boundary constraints } 2}$$

$$\underbrace{\qquad\qquad\qquad\qquad}_{\text{cHMM}}$$

For the reminder of this section, the set of possible states–chords will be denoted by $\mathcal{S}$, while the letters C and c will be used for denoting chords. The set of all possible notes (playing the role of 'observations' in the HMM methodology) will be denoted as $\mathcal{Y}$, while Y and y denote melody notes. There are four factors the the cHMM algorithm needs to generate a chord sequence, given a melody. Four factors are induced by the statistics from the training set.

1. The probability that each state (chord) is a beginning chord. This distribution is computed by examining the beginning chord for each phrase in the dataset and is denoted as $\pi(C_1 = c)$, $c \in \mathcal{S}$.

2. The probability that each state (chord) is an ending chord. This distribution is computed by examining the ending chord for each phrase in the dataset and is denoted as $\tau(C_T = c)$, $c \in \mathcal{S}$.

3. The probability that each state follows another state, denoted as $P(C_i = c_i | C_{i-1} = c_{i-1})$, $c_i, c_{i-1} \in \mathcal{S}$. One additional 'pseudo-distribution' is included, except from the beginning and ending chords and transition probabilities learned from data.

4. A vector assigning 'pseudo-probability' values to chords that include the melody's important notes for each chord segment, denoted by $P(C_i = c_i | Y_i = y_i)$. As discussed in further detail in Section 3.3, a chord might be harmonising a phrase segment that includes more than one melody notes, while the user is able to select which among the melody notes are important. For assigning a proper chord over a melody segment, the harmoniser tries to find chords that include as many of the important notes as possible. Thereby, for each melody segment to be harmonised by a chord, each chord is assigned with a 'pseudo-probability' value according to how many of the segment's important notes it includes. Therefore, for a melody segment, chords that include more important melody notes are more probable.

The overall probability for selecting a chord in a segment of T chords is computed by

$$P(C_i = c_i | Y_i = y_i) = P_\pi \, P_\mu \, P_\tau, \tag{1}$$

where

$$P_\pi = \pi(C_1 = c_1)\ P(C_1 = c_1|Y_1 = y_1), \tag{2}$$

$$P_\mu = \prod_{i=2}^{T} P(C_i = c_i|C_{i-1} = c_{i-1})$$
$$P(C_i = c_i|Y_i = y_i), \tag{3}$$
$$P_\tau = \tau(C_T = c_T)\ P(C_T = c_T|Y_T = y_T). \tag{4}$$

The generated sequence of chords is statistically optimal, in a sense that it presents a maximal combination for the probabilities in all the counterparts (P_π, P_μ and P_τ), typically through the Viterbi Forney (1973) algorithm. The probabilities in P_π promote some chords as better solutions to begin the path of chords: the ones that are more often used in the beginning of pieces in the dataset. Similarly, the probabilities in P_τ advance solutions that are more often met as concluding chords. However, if the beginning and/or ending chord is a constrained chord, the P_π and/or P_τ distributions respectively become 'binary', promoting only the chord that has been selected as constraint (with probability value 1). Specifically, if the beginning and ending chords are selected to be α_1 and α_T respectively, the new probabilities that substitute the ones expressed by Equations 2 and 4 are the respective following:

$$P'_\pi = \begin{cases} 1, & \text{if } C_1 = \alpha_1 \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

$$P'_\tau = \begin{cases} 1, & \text{if } C_T = \alpha_T \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

By allowing the imposition of final or intermediate chord constraints, the system is allowed to explore new harmonic paths that are suboptimal, but potentially more interesting. The relations between statistical optimality and musical interestingness is an interesting subject of research, that is left for future work.

4.2 Learning and assigning intermediate and final cadences

The limited memory according to the order of the Markov-based methods Pachet et al. (2011) does not allow them to consider longer time dependencies, a fact that is necessary for reflecting hierarchical structure of harmony. The intermediate chord constraints, as well as allowing direct user intervention in the generative process, offer the possibility to assign harmonic information in distant events, by employing intermediate and final cadences according to the phrase boundaries indicated by the user in the melodic input. Statistics for these cadences are learned during the training process (see Section 3.2), where expert annotated files including annotations for phrase endings are given as training material to the system.

Cadences are considered to be the final two chords of a phrase; during the cadence training process the two final chords in every phrase of every piece in the training data are captured. Statistics for unique cadences/pairs of chords are collected for two types of cadences:

1. *Final* cadences that are taken from the end of each piece's final phrase and

2. *Intermediate* cadences that are taken from the ending of every non-final phrase in each piece.

The final cadences collected from a set of 31 Bach chorales, which is a well-know idiom, are demonstrated in Table 3, along with the number of times they have been used. The set of final cadences collected from this set of Bach chorales reveals the specificity of cadential patterns in this idiom, including only variations of the perfect (and the use of the Tierce de Picardie for the minor). The number of different intermediate cadences is not overwhelmingly large: except for the perfect and half cadences, there are also some occurrences of the plagal and deceptive cadences along with some isolated cadential schemes that appear rarely.

Table 3: Number of occurrences of cadences induced from a set of Bach Chorales.

	Final cadences		
Penultimate chord	Major scale final chord $[0\ 0\ 4\ 7]$	Minor scale final chord $[0\ 0\ 3\ 7]$	$[0\ 0\ 4\ 7]$
$[7\ 0\ 4\ 7]$	8	1	2
$[7\ 0\ 4\ 7\ 10]$	13	1	6

After collecting the statistics about cadences from all idioms, the system, before employing the cHMM algorithm, assigns cadences as fixed chords to the locations indicated by user input (see Section 3.3). The cadence to be imported is selected based on three criteria: (a) whether it is a final or an intermediate cadence; (b) the cadence likelihood (how often it occurs in the training pieces); and (c) how well it fits with the melody notes that are harmonised by the cadence's chords. Specifically, for an intermediate or a final cadence, cadence likelihoods are taken from the probability distributions of each cadence in an idiom, i.e., how many times a cadence occurred over the total number of cadences. The appropriateness of a cadence according to the melody notes that the cadence's chords harmonise, is measured for each chord separately with the same method used in the cHMM algorithm, however, if a cadence chord lacks at least one important melody note in the segment it harmonises, then the cadence is disregarded as a whole (pair of chords). If for a given phrase ending no complete cadence (as a pair of chords) is found, then only the final chord is determined. If the utilisation of solely the final chord fails again, then no constraint is assigned for the cHMM. The motivation behind this cadence rejection mechanism was the reflection of the notion that the selected harmonisation idiom does not include a phrase closure toolset for the notes of the melody in the location that the user indicated a phrase ending – or at the end of the piece.

5   Bass voice leading and voicing layout of chords

Experimental evaluation of methodologies that utilise statistical machine learning techniques demonstrated that an efficient way to harmonise a melody is to add the bass line first (Whorley et al. 2013). This conclusion was made through the information theoretic measure cross-entropy, when the soprano, alto, tenor and bass voice were pairwise

compared regarding their statistical relations. The proposed harmoniser uses a modular methodology for determining the bass voice leading presented in Makris et al. (2015b), which includes independently trained modules that function on the previously defined GCT chords that constitute the harmonisation. These modules include (a) a hidden Markov model (HMM) deciding for the bass contour (hidden states), given the melody contour (observations), (b) distributions on the distance between the bass and the melody voice and (c) statistics regarding the inversions of the chords in the given chord sequence. The generation of chords (in GCT form) is performed by the cadence and cHMM probabilistic modules thus the selection of the proper voice layout scenarios for each GCT chord depends on the chords' inversion probabilities. After the bass voice is defined, the voicing layout of the internal chord notes is fixed.

5.1 Defining Bass Voice Motion

For constructing the bass voice leading, it is assumed that the bass voice is both a melody itself and it also depends on the piece's melody, a fact that motivates the utilisation of HMM. The primary module for defining bass motion plays the role of the hidden states under the first-order Markov assumption for bass contour (a bass motion depends on its previous one), in combination with the observations of the melody's contour (a bass motion depends on the underlying melody motion). Both the bass and the melody voice steps are represented by abstract notions that describe general quantitative information on pitch direction, also called 'direction descriptors'. In Makris et al. (2015a) several scenarios for voice contour refinement were examined, providing different levels of accuracy for describing the bass motion in different datasets. The proposed harmoniser the melody and bass note changes are divided in seven steps, as demonstrated in Table 4. The selected scenario of seven steps is based on the assumption that the perfect fourth is a small leap while the perfect fifth is a big leap.

refinement	short name	range (semitones)
steady voice	`st_v`	$x = 0$
step up	`s_up`	$1 \leqslant x \leqslant 2$
step down	`s_down`	$-2 \leqslant x \leqslant -1$
small leap up	`sl_up`	$3 \leqslant x \leqslant 5$
small leap down	`sl_down`	$-5 \leqslant x \leqslant -3$
big leap up	`bl_up`	$5 < x$
big leap down	`bl_down`	$x < -5$

Table 4: The pitch step and direction refinement scale considered for the development of the utilised bass voice leading system.

The level of refinement for representing the bass and melody voice movement give us the number of states and observations. According to the HMM methodology, the training process incorporates the extraction of statistics about the probabilities that a certain state (bass direction descriptor) follows another state, given the current observation element (melody direction descriptor), independently of the chord labels. These statistics are extracted from the training pieces of each idiom and incorporate four aspects:

1. The probability for each bass motion to begin the sequence.

18

2. The probability for each bass motion to end the sequence.
3. The probability that each bass motion follows another (transition probabilities).
4. The probability of a bass motion to be present, given a melody step.

The sequence of states that is generated by an HMM system, is produced according to the maximum probability described by the product of the aforementioned statistics, given a sequence of melody contour observations. The extracted probabilities for each possible next bass motion are stored in a vector of probabilities $\vec{p_m}$, which is afterwards utilised in the product of probabilities from all modules in Equation 7.

The bass voice motion provides abstract information about the motion of the bass, however, assigning actual pitches for a given set of chords requires additional information. Additionally, it might be the case that the best bass motion selected from the HMM module does not match other criteria concerning the chords that have already been selected by the cHMM, or the limits of permitted bass note pitch height. What if the best bass motion cannot be implemented for a chord, because it requires a rather improbably inversion of this chord (e.g., a V in second inversion)? What if the best bass motion drives the bass voice too high and close to the melody or too low? In order to assign a bass voice to a chord, additional information are required in the voice layout modules of the presented methodology, namely about *inversions* and the *melody-to-bass distance*. The inversions of a chord play an important role in determining how eligible is each chord's pitch class to be a bass note, while the melody-to-bass distance captures statistics about the pitch height region that the bass voice is allowed to move according to the melody.

All the *inversions* of a chord are obtained by assigning each of its pitch classes as a bass note. For instance, the chord with pitch classes $[0, 4, 7]$ has three inversions, with each one having a bass note that corresponds to a different pitch class. The voicing layout module of the harmonic learning system regarding chord inversions, is trained through extracting relevant information from every (GCT) chord every piece from each music idiom. For mapping pitch-class-related inversion information directly to GCT chords, a GCT chord is considered in the form $g = [r, \ \vec{t}]$, where $\vec{t}$ is the vector describing the type of the chord, i.e. its GCT base and extension in one array. For instance, the V chord in a key is expressed as $g = [7, \ [0, 4, 7, 10]]$ in the GCT representation, where 4 denotes the major third and 7 the perfect fifth and 10 the minor seventh. In this context, the GCT type is a set of integers, $\vec{t} = [t_1, \ t_2, \ \dots \ , \ t_n]$, where n is the number of type elements, that can be directly mapped to relative pitch classes (PCs). The statistics concerning chord inversion are expressed as the probability (p_i) that each type element in g is the bass note of the chord, or

$$p_i = (v_1, \ v_2, \ \dots \ , \ v_n),$$

where v_i, $i \in \{1, 2, \dots, n\}$, is the probability that the element t_i is the bass note. Table 5 demonstrates the extracted statistics for inversions for the most often met chords of the major mode Bach chorales. Therein it can be observed that the these chords are more often met in root position, while they are rarely played in the second inversion (fifth as bass note). Therefore, by integrating the inversion probabilities (p_i) within the voice layout modules as described in Equation 7, for instance the second inversion of the $[7, [0, 4, 7]]$ chord would be avoided when harmonising the style of the Bach chorales, since the probability related to its fifth being the bass note is zero.

19

GCT chord	relative PC	inversions
$[0, [0, 4, 7]]$	$[0, 4, 7]$	$[0.74,\ 0.23,\ 0.02]$
$[7, [0, 4, 7]]$	$[7, 11, 2]$	$[0.78,\ 0.22,\ 0.00]$
$[5, [0, 4, 7]]$	$[5, 9, 0]$	$[0.65,\ 0.34,\ 0.01]$

Table 5: Probabilities for chord inversion (p_i) in the three most frequently used chords in the major-mode Chorales of Bach.

An additional important aspect of voice layout concerns the absolute range of chords in the chord sequences of an idiom, i.e. the absolute difference between the bass voice and the melody. Different idioms encompass different constraints and characteristics concerning this voicing layout aspect, according to several factors, e.g., the utilised instruments' ranges. For capturing the distances between melody and bass pitch height in an idiom, interval-related information is extracted as approximate indicators about the expected pitch height of the bass voice through histograms of all melody-to-bass intervals found in the idiom's training pieces. Since exact intervals are scale-sensitive, e.g. different scales potentially produce different distributions of melody-to-bass intervals, the 'expected' bass pitch height is approximated by a normal distribution (denoted by p_{h_x}) that is adjusted to fit the distribution of the melody-to-bass intervals observed in the dataset.

For defining the pitch value of the bass in every step, the probabilities gathered from all the modules described hitherto are combined into a single value, computed as the product of all the probabilities from all the incorporated modules. To this end, for each GCT chord (C) in the sequence composed by the cHMM and cadence modules, every possible scenario of chord inversions and bass note pitch height, denoted by an index x, is generated. For each scenario (x), the product ($b_x(C)$) of all the modules discussed so far is computed, i.e. the bass motion ($p_{m_x}(C)$), the inversions ($p_{i_x}(C)$) and melody-to-bass interval $p_{h_x}(C)$:

$$b_x(C) = p_{m_x}(C)\ p_{i_x}(C)\ p_{h_x}(C). \tag{7}$$

Therefore, the best scenario (x_{best}) for the bass voice of chord C is found by: $x_{\text{best}} = \arg\max_x(b_x(C))$.

It has to be noted that the bass note motion probability ($p_{m_x}(C)$) of all examined inversions and pitch heights is obtained by the HMM module and takes a value given by the vector $\vec{p_m}$ according to the bass step it leads to. Therefore, the HMM probabilities are not utilised to compute the best sequence of all bass motions throughout the harmonisation, i.e., using the Viterbi algorithm. Contrarily, for the bass motion that is currently examined, all seven probabilities are calculated and stored in $\vec{p_m}$, while all possible pitch heights of the current chord (indexed by x) are assigned with a probability value accordingly. It should also be noted that the exact pitch height of the first bass in the first chord is calculated without information from the bass motion module ($p_{m_x}(C)$) since there is no motion in the bass before that.

An additional adjustment concerning the melody has to be made to avoid 'abnormal' for the idiom bass fluctuations in the selection of the optimal bass pitch height that are caused by potential large skips in the melody. For instance, a given melody may at some point move suddenly to very high pitches and then return to where it previously was. The effect of the melody-to-bass distribution would be to 'drag' the bass notes and make them

follow the melody, producing a bass motion that sounds unnatural to most tested idioms. To this end, the melody line is 'smoothened' with a moving average of 10 positions, i.e., every pitch height in the melody is substituted by the mean value of its 10 previous pitch heights (or fewer than 10, for melody notes before the 10th).

5.2   Defining the chord notes between the bass and the melody

Obtaining the best scenario for bass voice leading determines the exact pitch value of the bass voice for each GCT chord according to the bass motion HMM, inversions of the given GCT chord and the distance between the bass voice and the melody. Depending on the number of notes in each GCT, the voicing layout, i.e. exact pitches for all chord notes, for each chord is defined. To our knowledge, no study exists that focuses on examining the position of inner voices in a generated chord. To this end, a simple statistical model is proposed that utilises a generic tree data structure to find the best combination of the intermediate voices for every chord according to some simple criteria. Our proposed methodology summarises as follow:

1. Find all the possible combinations of the intermediate notes and store them in a generic tree structure.

2. Calculate the cost for every combination and select the best.

The total cost of every combination, in turn, is based on a weighted combination three cost criteria:

C1 Proximity to a *pitch-attractor*: The combination that best matches this criterion is the one that incorporates inner voice pitch values that are closest to a certain pitch value, named the *pitch-attractor*. The pitch-attractor value is set to a fixed ratio between the bass and the lowest melody note in the block of each chord.[3]

C2 Evenness of *neighbouring notes distances*: Evenness in inner voices of a chord is measured by calculating the standard deviation of their pairwise distances.

C3 Inner voice *movement distances between chords*: The inner voice movement between the previous and the current chord is calculated as the mean value of distances between the highest and the lowest inner voices. The best chord according to this criterion is the one with highest and lowest intermediate note pitches that are closest to the respective ones of the previous chord.

After thorough examination of the results in many simulations, the weight of the cost criteria are respectively: 0.5, 0.2 and 0.3. The voicing layout that is selected is the one that achieves the lowest total score in the weighted cost combination value.

---

[3]In the current version of the harmoniser the attractor is placed a 1/3 distance between melody and bass from the melody note. Additionally, for avoiding the 'dragging' effect of sudden melodic changes, the moving average version of the melody is used.

For example, consider that the GCT chord currently examined is $[2\ 0\ 3\ 7\ 10]$ with pitch classes $[0, 2, 5, 9]$ (D minor seventh), while the previous chord was the GCT chord $[0\ 0\ 4\ 7]$ (C major). Consider also that the MIDI pitches of the chord that comes before the aforementioned one are $[48, 55, 64]$, where the melody note is not considered, i.e. 55 and 64 are the inner notes of this chord, while for the D minor seventh the bass note value calculated by Equation 7 is 50 and the current melody note is 76. There are many possibilities for arranging the current chord's (D minor seventh) inner notes. To this end, the generic tree structure illustrated in Figure 5 is generated that represents all the voicing layout possibilities. All possible voicing layouts are taken by the tree interpretation by descending each branch from the root and they are then evaluated according to the three aforementioned criteria, the results of which are shown in Table 6.
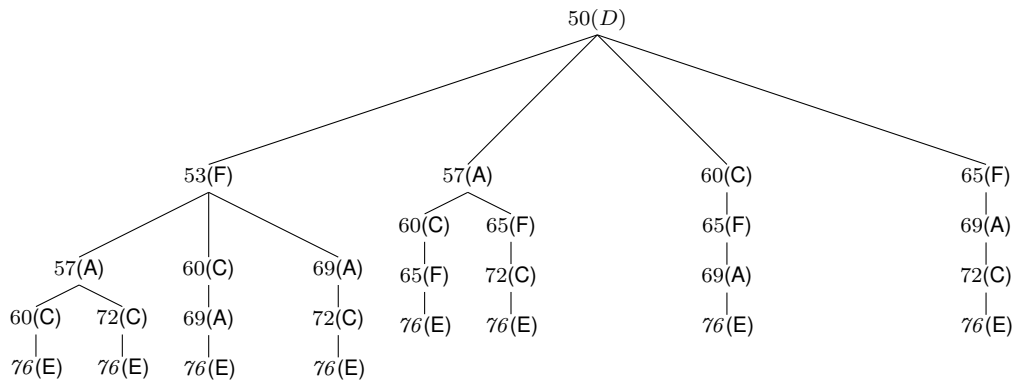
Figure 5: Tree representing all possible voicing layout scenarios for a D minor seventh with bass note 50 harmonising the melody note 76. The melody note is shown in italics since it is not actually part of the chord; it is used to demarcate the upper pitch limit of the chord's inner pitches.

Table 6: Evaluating the voicing layout scenarios for a D minor seventh with bass note 50 harmonising the melody note 76, following the chord $[48, 55, 64]$ (C major). The selected voicing layout is the one with the lowest total weighted score, shown in bold.

Voicing layout	C1 score	C2 score	C3 score	Total score
$[50, 53, 57, 60]$	6.667	0.707	6	5.275
$[50, 53, 57, 72]$	6.667	7.778	10	7.889
$[50, 53, 60, 69]$	4.667	1.414	7	4.716
$[50, 53, 69, 72]$	4.667	9.192	10	7.172
$[\mathbf{50, 57, 60, 65}]$	2.000	1.414	3	$\mathbf{2.182}$
$[50, 57, 65, 72]$	2.000	0.707	10	4.141
$[50, 60, 65, 69]$	0.000	0.707	10	3.141
$[50, 65, 69, 72]$	0.000	0.707	18	5.541

6 Experimental results

Evaluating computational or even human creativity is a difficult task, especially when the assessment of aesthetic quality is also involved. Furthermore, there is not a well-established and commonly accepted definition of creativity, as many authors approach it from different

perspectives (e.g. Boden (2004); Wiggins (2006); for a comprehensive discussion see Jordanous (2013), chapter 3). The creative and structural characteristics of the system are examined through presenting examples on different harmonisation tasks as well as through statistical measures of similarities in harmonisations of melodies with different learned harmonies. The melodic harmonisation examples concern five melodies as well as different structural harmonisation attributes, e.g. intermediate phrase boundaries and user-selected chord constraints. These examples demonstrate the system's potential and indicate the integrity of harmonisations that, in some cases, reach human expert-standards with minor adjustments.

The statistical experimental process (presented in Section 6.2) examines the similarity between system-generated harmonisations of (11) different melodies and original training harmonisations. This process reveals that the harmonisations produced by the system when trained on an idiom may diverge from that idiom, depending on how its harmonic characteristics align with the structural properties and implied harmony of input melodies.

6.1 Example Harmonisations

Five diverse short melodies were chosen, three from classical music (baroque, classical and romantic periods), one from pop music and one folk song:

1. J. S. Bach: The fugue theme from the Well-Tempered Clavier I, fugue no. 8, transposed in D minor. The melody is a 3-bar phrase that concludes with a perfect cadence in D minor.

2. L. v. Beethoven: The melodic theme (b. 1-8) from the second movement in A♭ major of the Piano Sonata no. 8. The melody comprises two 4-bar phrases (half cadence – full cadence) that form an 8-bar period.

3. The Beatles: The first melodic phrase of the song 'Michelle', transposed in C minor. It is a 6-bar phrase, ending with a half cadence to the dominant.

4. Greek folk song: 'Tou Kitsou ē mana', taken from Yannis Constantinidis's collection '44 miniatures for piano' (no. 27). The melody is in A Dorian mode and comprises two phrases (4-bar and 7-bar) of which the second consists of two sub-phrases (3-bar and 4-bar).

5. Gabriel Fauré: The first three phrases (b. 2-21 without the repetitions) of the *Sicilienne* for cello and piano (op. 78). The melody is mainly in the Dorian mode; the first two phrases form an eight-bar period (half cadence-full cadence), while the third phrase exhibits tonal/modal mixture.

Eight different musical idioms (see Section 3.2) were used for the harmonisation of the above five melodies, but for reasons of space only a small selection of the most interesting 13 harmonisations is presented. The system produced raw midi files that were processed by humans using the Finale 2014 musical notation software[4]. The process involved the following: correction of musical notation issues and enharmonic spellings of

---

[4]https://www.finalemusic.com

pitches, separation of the bass line in a different layer or staff, preservation of constant number of active voices in the musical texture through the use of octave doublings, manual arrangement of the inner voices for smoother voice-leading where needed, and analysis of harmonic progressions through the use of Latin Roman numeral notation in cases of tonal harmonisation. The pitch content of the chords was always kept intact, and the bass line was manually altered in very few cases (indicated by * in the scores) in order to avoid stylistic inconsistencies or achieve better voice-leading.

Three selected harmonisations of the Bach fugue are illustrated in Figure 6. The first harmonisation was based on the Kostka-Payne corpus (classical/romantic tonal harmony), which is compatible with the style of the melody, and is characterised by frequent use of the dominant and a chromatically embellished full cadence prepared by two chords with pre-dominant function: ii^6_5 and vii^7_o of V. The second harmonisation uses the Epirus polyphonic singing style and is almost consistently based on the D minor pentatonic scale (D, F, G, A, C) with the E of the last bar being the only exception. The chords are mildly dissonant verticalisations of the pentatonic set instead of the D minor triad, which – typically in this idiom – was avoided, and there is also a constant drone of the pitch center in the lower voice. The third harmonisation was made in the Hindemith style and exhibits free mildly dissonant chords, mostly free verticalisations of diatonic sets, except from the cadence which is tonal (V^2 - I^6). Interestingly, pitches not included in the scale of the melody are inserted for the creation of idiomatic harmony, such as B, F♯ and C♯.

(a) Bach melody harmonised in the Kostka-Payne style.

(b) Bach melody harmonised in the Epirus style.

(c) Bach melody harmonised in the Hindemith style.

Figure 6: Bach Melody harmonised in several diverse styles: (a) Kostka-Payne, (b) Epirus songs and (c) Hindemith.

For the theme by Beethoven the three harmonisations illustrated in Figure 7 were se-

lected. The first one (without chord constraints) was based on the Kostka-Payne idiom and is quite close to Beethoven's own style, particularly in the second phrase, which incorporates progressions in the circle of fifths and a full tonal cadence. However, the proposed harmony of the first phrase was considered static due to an insistent use of the V^7-I progression, so a second harmonisation based on the same idiom was attempted, albeit with two chord constraints in the first phrase (indicated by rectangular frames in the score). The result is substantially different, and the harmonic effect caused by the injected chords expel the tonic chord completely from the first phrase and create interesting chromatic tonicisations and a half-cadence in the phrase's end. The theme's third harmonisation used the highly chromatic Grieg idiom and rendered even more daring and interesting chromatic chords, such as the altered dominants with lowered 5^{ths} (b. 2 and 4, French-type augmented 6th chords), the borrowed vii^{o7}/V with the tonic pedal note in the 3rd voice (b. 3), the tonal mixture chords $\flat$VI and $\flat$III (b. 5 and 6), of which the $\flat$VI is doubly altered ($5\flat = B\flat$ and $5\sharp = C$), and the German-type augmented 6th chord preparing the ii^7 chord (b. 6 and 7).

For the Beatles melodic phrase two harmonisations were selected (see Figure 8), both without any chord constraints. The first harmonisation followed the Bach chorale idiom and rendered typical diatonic or chromatic tonal progressions leading to an anticipated half cadence to the dominant. The second harmonisation was based on Yannis Constantinidis's 20th-century modal idiom, and featured almost exclusively freely used major triads with major 7th and minor triads with minor 7th. In this rendering interesting parallel harmony elements are observed ($A\flat^{maj7}$-Gm^7-Fm^7-$E\flat m$), while the half cadence is avoided and substituted by a III chord with major 7th. Two bass notes were manually changed (indicated by *) in order to create a complete stepwise descent from A$\flat$ to C in the bass line.

Three selected harmonisations of the Greek folk song are illustrated in Figure 9. The first was based on the Fauxbourdon medieval idiom, characterised mainly by parallel $\frac{6}{3}$ chords and cadences to open 8th-5th sonorities. The system proposed suitable chordal content (major or minor triads, open 5^{ths} and one diminished triad as penultimate cadential chord), but the bass line had to be manually changed six times (annotated with * in the score) in order to achieve stylistic compatibility. The second harmonisation is based on Bach chorales. The result is tonal functional harmony, with diatonic and chromatic elements (tonicisations) and with tonal cadences at the end of the phrases and sub-phrases. The proposed bass line was left intact, in spite of the awkward augmented 2nd in the first bar. The last harmonisation is based on Hindemith's harmonic idiom, and is characterised by free use of chromaticism, mildly dissonant sonorities stemming from diatonic sets and more stable sonorities (major or minor triads) at the end of the phrases (a notably interesting progression is the transition from Em^7 to Gm at b. 6-7).

Finally, two harmonisations of the Sicilienne melody are illustrated in Figure 10. The first was based on the jazz harmonic idiom, characterised mainly by the free use of 7th chords and other extended/chromatic chords. The proposed harmony is a mixture of tonal and modal jazz harmony, with free chromatic or diatonic modal chords encountered during the unfolding of the melody and more tonal/functional progressions at the cadences. The second harmonisation was based on Hindemith's neotonal, mildly dissonant, non-functional harmony. The free chromaticism employed produced interesting enharmonic phenomena (e.g. at b. 9 and 11).

(a) Beethoven melody harmonised in the Kostka-Payne style without the user-defined chord constraints.

(b) Beethoven melody harmonised in the Kostka-Payne style with user-defined chord constraints.

(c) Beethoven melody harmonised in the Grieg style.

Figure 7: Beethoven melody harmonised in the Kostka-Payne style (a) without and (b) with user-defined contraints and (c) in the style of Grieg.

Overall, the thirteen harmonisations of the five chosen melodies produced by the system with some unobtrusive human manipulation incorporated a wide spectrum of musical idioms – with a range of over eight centuries – and demonstrated the flexibility and creative potential of the proposed harmonisation system.

6.2 Statistical similarities between original harmonies and new melodic harmonisations

The system is trained on several statistical aspect of a specific idiom and it uses the learned material in input melodies to produce novel harmonisations. How similar are the produced harmonisations in relation to the original training harmonisations of an idiom? In other words, is the system only able to mimic the training harmonisations or it is possible that 'divergent' harmonisations can be produced? This question is addressed by examining the

(a) The melody of Michelle by the Beatles harmonised in the style of Bach Chorales.

(b) The melody of Michelle by the Beatles harmonised in the style of Constantinidis.

Figure 8: The melody of Michelle by the Beatles harmonised in the style of: (a) Bach Chorales and (b) Constantinidis.

(a) Traditional melody harmonised in the style of fauxbourdon.

(b) Traditional melody harmonised in the Bach Chorale style.

(c) Traditional melody harmonised in the style of Hindemith.

Figure 9: Traditional melody harmonised in the style of: (a) fauxbourdon, (b) Bach Chorales and (c) Hindemith.

statistical similarities between original harmonies of idioms and harmonisations produced by the system for several melodies. The melodies used for producing harmonisations for

(a) The Sicilienne melody harmonised in the style of Hindemith.

(b) The Sicilienne melody harmonised in jazz style.

Figure 10: The Sicilienne melody harmonised in the style of: (a) Hindemith: and (b) jazz.

this study include the five of the ones presented previously in the examples (one major and four minor melodies), with the addition of five major mode and one minor mode melodies, to compile a total set of six major and five minor melodies. The set of major melodies includes melodies from Haydn, Mozart, Beethoven, Jobim and two traditional ones, while the selected minor melodies are by Bach, Michelle by the Beatles, Sicilienne by Fouré and two traditional melodies.

The statistical similarity of harmonies in this experimental process is based on the transitions of GCT chords. Voicing layout elements are disregarded for this study since their complex statistical interdependence makes it hard to construct a unique statistical model that can be used for statistical similarity. Instead, this study examines similarities of GCT chord transition probabilities in original pieces (used for training the system) and novel

28

harmonisations. The examination concerns one idiom at a time, $\mathcal{I}$, where the available training harmonies (pieces with more than 4 chord transitions) are considered to form a set $\mathcal{T_I}$, while the harmonies produced by the system for new melodies form the set $\mathcal{M_I}$. Each harmonic piece in both sets is represented by its first-order Markov transition matrix, which represents its GCT chord transition probability distribution.

The distance between two transition probability distributions is quantified by the Hellinger distance (Gibbs and Su 2002), which is a distance metric for two distributions. Using this metric a pairwise distance matrix is constructed for both the original $\mathcal{T_I}$ and the generated $\mathcal{M_I}$ harmonic pieces for each idiom ($\mathcal{I}$). This matrix is mapped afterwards into a two-dimensional space using multidimensional scaling (MDS), in order to obtain a Euclidean approximation of the space of GCT chord transition distributions based on their pairwise distances. Two major and two minor-mode examples of the two-dimensional spaces produced by this process are presented in Figure 11, where the sets $\mathcal{T_I}$ (grey ×s) and $\mathcal{M_I}$ (red circles) for the Bach chorales and the Kostka-Payne sets are illustrated.

(a) Major mode Bach chorales

(b) Major mode Kostka-Payne

(c) Minor mode Bach chorales

(d) Minor mode Kostka-Payne
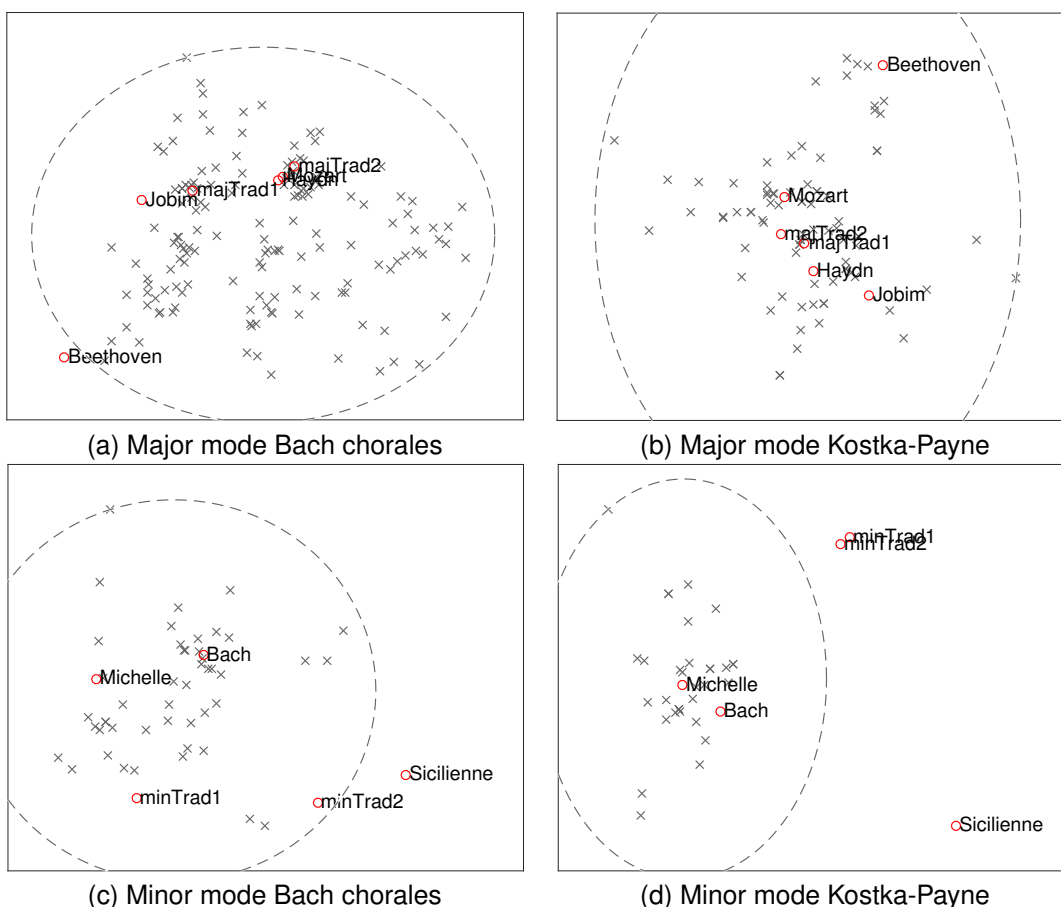
Figure 11: Examples of Bach chorales and Kostka-Payne harmonic pieces of original idiom harmonisations (illustrated with grey ×s) and new system-generated harmonisations (red circles) in the space produced by multidimensional scaling based on the Hellinger pairwise distances.

The original idiom harmonisation ($\mathcal{T_I}$), as depicted in the examples in Figure 11, are considered to form a cluster. To study the relative placement of the new harmonisations in

every idiom's cluster, the concept of *cluster radius* is used. Cluster radius is the maximum distance of all cluster members (harmonies in $\mathcal{T}_{\mathcal{I}}$) from the cluster centroid, which is the placed at the centre of mass of $\mathcal{T}_{\mathcal{I}}$. The radii of the clusters around their centroids are depicted by the dashed line ellipsoids in Figure 11, while the ellipticity is due to different axis scales. A harmonic sequence that is outside an idiom's radius, presents transitions in proportions that are not 'usual' (in a statistical sense) within the training idiom. The novel system-generated harmonisations ($\mathcal{M}_{\mathcal{I}}$) that are outside an idiom's cluster radius, are considered to constitute 'uncommon' new harmonisations that *explore* new harmonic areas in an idiom.

The radius for each cluster and the distances of new harmonies from the cluster's centroid are demonstrated in Table 7. One can notice that for some corpora there are more than one melodies that produce harmonisations outside the cluster's radius, e.g. in Constantinidis major and Grieg, Kostka-Payne (Figure 11 (d)), Hindemith and jazz minor. The Hindemith and jazz example harmonisations in Figure 10 of the Sicilienne melody, which are outside the respective clusters' radii, suggest that the general characteristics of the styles are locally preserved, even though the chord sequences as wholes are statistically 'divergent' from the idiom. On the other hand, all the Kostka-Payne (Figure 11 (c)) and jazz major new harmonisations are inside the cluster's radius. The music-theoretic reasons for such differences, or the perceptual impact of harmonisations outside or inside an idiom's radius are important subjects that should be addressed in future research.

Table 7: Cluster radius of intra-idiom harmonisations ($\mathcal{T}_{\mathcal{I}}$) and distances of system-generated harmonisations ($\mathcal{M}_{\mathcal{I}}$) in extra-idiom major and minor melodies from cluster centroid. Numbers is bold indicate the cases where new harmonisations are outside the radius.

	Major harmonies and melodic harmonisations					
	$\mathcal{T}_{\mathcal{I}}$ radius	Beethoven	Jobim	Haydn	Mozart	majTrad1	majTrad2
Fauxbourdon	0.4516	0.2249	0.3834	0.1341	0.2397	0.3393	**0.4851**
Bach Chorales	0.1430	**0.1541**	0.0796	0.0426	0.0462	0.0550	0.0560
Kostka-Payne	0.1398	0.0890	0.0539	0.0247	0.0192	0.0111	0.0190
Grieg	0.3350	0.2288	**0.4180**	0.1637	0.1708	0.1797	0.1728
Constantinides	0.1117	**0.2280**	**0.2913**	**0.1922**	**0.1473**	**0.2531**	**0.2542**
Jazz	0.3812	0.0449	0.1143	0.0674	0.0549	0.0852	0.0382

	Minor harmonies and melodic harmonisations					
	Intra-idiom	Bach	Michelle	Sicilienne	minTrad1	minTrad2
Fauxbourdon	0.4333	0.2852	0.1768	0.2538	**0.5492**	0.1894
Bach Chorales	0.2645	0.0626	0.1028	**0.3256**	0.1572	0.2438
Kostka-Payne	0.1670	0.0506	0.0052	**0.3413**	**0.2275**	**0.2155**
Grieg	0.3015	**0.1186**	0.0363	**0.3629**	0.0844	0.1656
Epirus	0.4193	0.0830	0.2099	0.3202	0.2586	**0.5280**
Constantinides	0.1497	0.1306	0.1148	**0.2892**	0.0451	0.1327
Hindemith	0.3111	0.1143	0.1530	**0.3850**	**0.3182**	0.2287
Jazz	0.1098	0.0541	0.0714	**0.2870**	**0.2181**	0.0882

Depending on the melody, the system may either produce harmonisations that are similar to the original training harmonies, or be forced to produce harmonisations that are less similar. This fact is important in two respects: on one hand the system is able to mimic hierarchically structured processes through a Markov-based process (using induced constraints), while on the other hand new harmonic paths can be explored. For instance,

harmonising the traditional or the Sicilienne melodies with the system trained with the Kostka-Payne corpus (Figure 11 (d)), forces the system to 'explore' new harmonic areas within the idiom and generate diverse novel harmonies, in contrast to the harmonisations of the Beatles and Bach melodies. The harmonies that excess an idiom's radius, on the other hand, still reflect its learned characteristics, as indicated in the example of the Bach chorale harmonisation of the minor traditional melody 2 in Figure 9 (c), even though it is placed remotely in Figure 11 (c).

Interestingly, when the system is trained with the Bach chorales and the Kostka-Payne corpus, the relative positions of composed melodic harmonisations may be different. For instance, the harmonisations produced for the Mozart and Haydn melodies when trained with the Bach chorales (Figure 11 (a)) are very similar (one is almost placed over the other), while training the system with the Kostka-Payne corpus harmonises these melodies quite differently (Figure 11 (b)) – a fact that is possible due to the probabilistic mechanics behind the cHMM methodology. Furthermore, this is also a possible outcome in the proposed system, where even similar melodies can be harmonised in completely different ways if, for instance, different cadences are automatically selected, or, potentially, different intermediate chord constraints (or cadences) are selected by the user.

7   Concluding remarks

Melodic harmonisation with automated means is a task that requires algorithms exhibiting both emergence of creativity and preservation of structure. The first approaches for automated melodic harmonisation included methodologies that were based on human-defined rules. The strength of these approaches is that the rules they incorporate preserve the hierarchical structure of harmony. Among their shortcomings, however, is the fact that different sets of rules describe different idioms and it is impossible to come up with 'one size fits all' harmonic rules for all idioms. On the other hand, methodologies that utilise statistical learning can learn specific aspects of harmony from data, a fact that enables them to learn and create harmonies in different musical idioms. The main disadvantage of probabilistic methodologies is that they work in rather 'linear' chord-to-chord manner, disregarding higher-level structural relations between remote harmonic parts. The first contribution of the proposed melodic harmonisation system is the fact that it can learn from music data from diverse idioms, while at the same time preserve relations at distant harmonic events by assigning intermediate and final cadences at locations of phrase endings. Additionally, the system output is a complete harmonic realisation with chords being described not only as labels but as note simultaneities. To this end, different harmonic learning modules are responsible for learning and composing different aspects of harmony, namely chord types, chord transitions, cadences, bass voice movement, chord inversions and melody-to-bass note distances. Furthermore, the user can choose to import any desired chord at any location of the harmonisation, 'derailing' the system from its trained harmonic course forcing it to take creative decisions and follow alternative harmonic paths.

The creative agility of the system is obvious when used to harmonise melodies in a variety of learned idioms. Therein, the implied harmony incorporated in the melody is blended with the learned harmony employed for the harmonisation, producing interesting

harmonic output. An analysis on melodic harmonisation examples, where melodies were harmonised with harmonically 'incompatible' learned idioms, indicated that some of these harmonisations were inside and some outside the cluster of the original idiom harmonisations. It was therefore demonstrated that the system not only reflects the characteristics of original harmonisations within an idiom, but also potentially invents new creative harmonic routes that at some extent constitute a blend of the harmonising idiom's and the melody's implied harmony. In other words, the system exhibits adaptivity in learning and agility in expressing learned harmonic idioms in different and potentially alien harmonic environments – as imposed by a melody's structure. Another important aspect of the system is its ability to comply with specific user preferences in harmony, expressed as chord constraints. The user is allowed to experiment by employing desired chords in any position of the melody, forcing the system to follow potentially radically different harmonic paths in order to satisfy the user-imposed constraints. The direct involvement of the user in the creativity loop, combined with the numerous potential harmonisations using different learned idioms, make the proposed system valuable not only as an autonomous creative tool, but also as a tool that enhances the creativity of the user as a composer.

The system is developed in the context of a wider research project, where conceptual blending (Fauconnier and Turner 2003; Goguen 2006) is studied as a generative means to creating new conceptual spaces (features and relations between them) by combining the elements of two input ones. Regarding the proposed system, learned probabilistic elements of different input idioms will be transformed in logic-related feature terms, while formal computational blending processes (Schorlemmer et al. 2014; Kaliakatsos-Papakostas et al. 2014a; Cambouropoulos et al. 2015) will create new elements and relations that creatively combine and extend the input idioms by generating new probabilistic relations between them. However, the system in its current form is still a valuable tool for potential user groups. For instance, composers are able to get a 'batch' of creative ideas on harmonisation alternatives for a given melody within a few seconds. The system is able to provide very quickly several ideas on how a melody would be harmonised under different learned conditions, enhancing the composers' creativity by providing many new ideas on the entire harmonisation or on parts of it. Additionally, the composers are able to keep some parts of the harmonisation fixed (as chord constraints) and search for alternatives in focused areas. Furthermore, the system can be used for educational purposes, indicating to students which harmonisation follows the most 'usual' harmonic paths for a given melody in diverse idioms. Students have the chance to explore creative ideas in a style-specific harmonic environment by imposing their desired chord constraints and studying the alternative harmonic routes that the system proposes in the context of a specific idiom.

Acknowledgments

References

Allan, M. and Williams, C. K. I. (2004). Harmonising chorales by probabilistic inference. In *Advances in Neural Information Processing Systems 17*, pages 25–32. MIT Press.

Boden, M. A. (2004). *The creative mind: Myths and mechanisms*. Psychology Press.

Boden, M. A. (2009). Computer Models of Creativity. *AI Magazine*, 30(3):23.

Borrel-Jensen, N. and Hjortgaard Danielsen, A. (2010). Computer-assisted music composition – a database-backed algorithmic composition system. B.S. Thesis, Department of Computer Science, University of Copenhagen, Copenhagen, Denmark.

Cambouropoulos, E. (2015). The Harmonic Musical Surface and Two Novel Chord Representation Schemes. In Meredith, D., editor, *Computational Music Analysis*, pages 31–56. Springer.

Cambouropoulos, E., Kaliakatsos-Papakostas, M., and Tsougras, C. (2014). An idiom-independent representation of chords for computational music analysis and generation. In *Proceeding of the joint 11th Sound and Music Computing Conference (SMC) and 40th International Computer Music Conference (ICMC)*, ICMC–SMC 2014.

Cambouropoulos, E., Kaliakatsos-Papakostas, M., and Tsougras, C. (2015). Structural blending of harmonic spaces: a computational approach. In *Proceedings of the 9th Triennial Conference of the European Society for the Cognitive Science of Music (ESCOM)*.

Chai, W. (2005). Automated analysis of musical structure. Thesis (Ph. D.)–Massachusetts Institute of Technology, School of Architecture and Planning, Program in Media Arts and Sciences, 2005.

Chuan, C.-H. and Chew, E. (2007). A hybrid system for automatic generation of style-specific accompaniment. In *Proceedings of the 4th International Joint Workshop on Computational Creativity*. Goldsmiths, University of London.

Conklin, D. (2002). Representation and discovery of vertical patterns in music. In *Music and artificial intelligence*, pages 32–42. Springer.

Dixon, S., Mauch, M., and Anglade, A. (2010). Probabilistic and logic-based modelling of harmony. In *Exploring Music Contents*, pages 1–19. Springer.

Donnelly, P. and Sheppard, J. (2011). Evolving four-part harmony using genetic algorithms. In *Proceedings of the 2011 International Conference on Applications of Evolutionary Computation - Volume Part II*, EvoApplications'11, pages 273–282, Berlin, Heidelberg. Springer-Verlag.

Ebcioglu, K. (1988). An expert system for harmonizing four-part chorales. *Computer Music Journal*, 12(3):43–51.

Fauconnier, G. and Turner, M. (2003). *The Way We Think: Conceptual Blending And The Mind's Hidden Complexities*. Basic Books, New York, reprint edition edition.

Forney, G.D., J. (1973). The viterbi algorithm. *Proceedings of the IEEE*, 61(3):268–278.

Gibbs, A. L. and Su, F. E. (2002). On choosing and bounding probability metrics. *International statistical review*, 70(3):419–435.

Goguen, J. (2006). Mathematical Models of Cognitive Space and Time. In Andler, D., Ogawa, Y., Okada, M., and Watanabe, S., editors, *Reasoning and Cognition*, volume 2 of *Interdisciplinary Conference Series on Reasoning Studies*. Keio University Press.

Granroth-Wilding, M. and Steedman, M. (2014). A robust parser-interpreter for jazz chord sequences. *Journal of New Music Research*, 0(0):1–20.

Hanlon, M. and Ledlie, T. (2002). Cpu bach: An automatic chorale harmonization system.

Jackendoff, R. (1987). *Consciousness and the computational mind*, volume xvi of *Explorations in cognitive science, No. 3.* The MIT Press, Cambridge, MA, US.

Jordanous, A. K. (2013). *Evaluating computational creativity: a standardised procedure for evaluating creative systems and its application*. PhD thesis, University of Sussex.

Kaliakatsos-Papakostas, M. and Cambouropoulos, E. (2014). Probabilistic harmonisation with fixed intermediate chord constraints. In *Proceeding of the joint 11th Sound and Music Computing Conference (SMC) and 40th International Computer Music Conference (ICMC)*, ICMC–SMC 2014.

Kaliakatsos-Papakostas, M., Cambouropoulos, E., Kühnberger, K.-U., Kutz, O., and Smaill, A. (2014a). Concept Invention and Music: Creating Novel Harmonies via Conceptual Blending. In *In Proceedings of the 9th Conference on Interdisciplinary Musicology (CIM2014)*, CIM2014.

Kaliakatsos-Papakostas, M., Katsiavalos, A., Tsougras, C., and Cambouropoulos, E. (2014b). Harmony in the polyphonic songs of epirus: Representation, statistical analysis and generation. In *4th International Workshop on Folk Music Analysis (FMA) 2014*.

Kaliakatsos-Papakostas, M., Zacharakis, A., Tsougras, C., and Cambouropoulos, E. (2015). Evaluating the General Chord Type representation in tonal music and organising GCT chord labels in functional chord categories. In *Proceedings of the 4th International Conference on Music Information Retrieval (ISMIR 2015)*, Malaga, Spain.

Kaliakatsos-Papakostas, M. A., Floros, A., and Vrahatis, M. N. (2013). A clustering strategy for the key segmentation of musical audio. *Computer Music Journal*, 37(1):52–69.

Koops, H. V., Magalhães, J. P., and de Haas, W. B. (2013). A functional approach to automatic melody harmonisation. In *Proceedings of the First ACM SIGPLAN Workshop on Functional Art, Music, Modeling & Design*, FARM '13, pages 47–58, New York, NY, USA. ACM.

Kostka, S. M. and Payne, D. (2004). *Instructor's Manual to Accompany Tonal Harmony, with an Introduction to Twentieth-century Music*. McGraw-Hill.

Lolis, K. (2006). *To Epirótiko Polyphonikó Tragoúdi (Epirus Polyphonic Song)*. Ioannina.

Makris, D., Kaliakatsos-Papakostas, M., and Cambouropoulos, E. (2015a). A probabilistic approach to determining bass voice leading in melodic harmonisation. In *Mathematics and Computation in Music: Proceedings of the 5th International Conference, MCM 2015*, page Accepted for publication, London, UK. Springer, Berlin.

Makris, D., Kaliakatsos-Papakostas, M., and Cambouropoulos, E. (2015b). Probabilistic modular bass voice leading in melodic harmonisation.

Pachet, F. and Roy, P. (2001). Musical harmonization with constraints: A survey. *Constraints*, 6(1):7–19.

Pachet, F., Roy, P., and Barbieri, G. (2011). Finite-length Markov processes with constraints. In *International Joint Conference on Artificial Intelligence (IJCAI) 2011*, pages 635–642.

Paiement, J.-F., Eck, D., and Bengio, S. (2006). Probabilistic melodic harmonization. In *Proceedings of the 19th International Conference on Advances in Artificial Intelligence: Canadian Society for Computational Studies of Intelligence*, AI'06, pages 218–229, Berlin, Heidelberg. Springer-Verlag.

Pérez-Sancho, C., Rizo, D., and Inesta, J. M. (2009). Genre classification using chords and stochastic language models. *Connection science*, 21(2-3):145–159.

Phon-Amnuaisuk, S., Smaill, A., and Wiggins, G. (2006). Chorale harmonization: A view from a search control perspective. *Journal of New Music Research*, 35(4):279–305.

Phon-Amnuaisuk, S. and Wiggins, G. A. (1999). The four-part harmonisation problem: A comparison between genetic algorithms and a rule–based system. In *In proceedings of the AISB'99 symposium on musical creativity*, pages 28–34. AISB.

Raczyński, S. A., Fukayama, S., and Vincent, E. (2013). Melody harmonization with interpolated probabilistic models. *Journal of New Music Research*, 42(3):223–235.

Raphael, C. and Stoddard, J. (2004). Functional harmonic analysis using probabilistic models. *Computer Music Journal*, 28(3):45–52.

Rohrmeier, M. (2011). Towards a generative syntax of tonal harmony. *Journal of Mathematics and Music*, 5(1):35–53.

Scholz, R., Vincent, E., and Bimbot, F. (2009). Robust modeling of musical chord sequences using probabilistic n-grams. In *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, pages 53–56. IEEE.

Schorlemmer, M., Smaill, A., Kühnberger, K.-U., Kutz, O., Colton, S., Cambouropoulos, E., and Pease, A. (2014). Coinvent: Towards a computational concept invention theory. In *5th International Conference on Computational Creativity (ICCC) 2014*.

Simon, I., Morris, D., and Basu, S. (2008). Mysong: Automatic accompaniment generation for vocal melodies. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, pages 725–734, New York, NY, USA. ACM.

Steedman, M. (1996). The blues and the abstract truth: Music and mental models. *Mental models in cognitive science*, pages 305–318.

Suzuki, S. (2013). Four-part harmonization using probabilistic models: Comparison of models with and without chord nodes. In *Proceedings of the Sound and Music Computing Conference (SMC)*, pages 628–633, Stockholm, Sweden. Logos Verlag Berlin.

Tsougras, C. (2010). The Application of GTTM on 20th Century Modal Music: Research Based on the Analysis of Yannis Constantinidis's "44 Greek Miniatures for Piano". *Musicae Scientiae*, 14(1 suppl):157–194.

Whorley, R. P., Wiggins, G. A., Rhodes, C., and Pearce, M. T. (2013). Multiple viewpoint systems: Time complexity and the construction of domains for complex musical viewpoints in the harmonisation problem. *Journal of New Music Research*, 42(3):237–266.

Wiggins, G. A. (2006). A preliminary framework for description, analysis and comparison of creative systems. *Knowledge-Based Systems*, 19(7):449–458.

Wiggins, G. A., Pearce, M. T., and Müllensiefen, D. (2009). Computational modeling of music cognition and musical creativity. In *The Oxford Handbook of Computer Music*. OUP USA.

Yi, L. and Goldsmith, J. (2007). Automatic generation of four-part harmony. In Laskey, K. B., Mahoney, S. M., and Goldsmith, J., editors, *BMA*, volume 268 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Yogev, N. and Lerch, A. (2008). A system for automatic audio harmonization. In *Proceedings of the 25-th VDT International Convention*.

Zacharakis, A., Kaliakatsos-Papakostas, M., and Cambouropoulos, E. (2015). Conceptual blending in music cadences: A formal model and subjective evaluation. In *Proceedings of the 4th International Conference on Music Information Retrieval (ISMIR 2015)*, Malaga, Spain.