

Algorithms for Collaborative Harmonization

Eyal Briman¹, Eyal Leizerovich¹ and Nimrod Talmon¹

¹Ben Gurion University of the Negev

Abstract. We consider a specific scenario of text aggregation, in the realm of musical harmonization. Musical harmonization shares similarities with text aggregation, however the language of harmony is more structured than general text. Concretely, given a set of harmonization suggestions for a given musical melody, our interest lies in devising aggregation algorithms that yield an harmonization sequence that satisfies the following two key criteria: (1) an effective representation of the collective suggestions; and (2) an harmonization that is musically coherent. We present different algorithms for the aggregation of harmonies given by a group of agents and analyze their complexities. The results indicate that the Kemeny and plurality-based algorithms are most effective in assessing representation and maintaining musical coherence.

1 Introduction

Social choice theory provides aggregation algorithms that facilitate collaborative creation of diverse outputs within agent communities [2]: e.g., single-winner elections, multi-winner elections [10], and participatory budgeting [20, 3]. No good solutions, however, exist for the aggregation of preferences regarding the collaborative creation of text documents. The Following work serves as a milestone towards this process of collaborative text writing. We consider a musical melody and a population of agents, each of which is suggesting a different harmonic sequence (equivalently, an harmonization) for the melody; and our aim is to aggregate those suggestions to come to a single harmonization. Our pursuit is to discover algorithms that strike a balance between respecting the agent community's preferences and crafting aggregated sequence of harmonies that are likely to appear according to prior knowledge of chord sequences translated to a 2-gram.

To this end, we first model the problem of collaborative harmonization and then explore various specially-crafted aggregation algorithms in pursuit of this goal. We then report on computer-based simulations performed on real-world data and generated data.¹

Paper Structure After reviewing related work, we provide musical preliminaries 2). This groundwork will facilitate the formal delineation of the social choice framework for collaborative harmonization in Section 3. Proceeding from there, we delineate various approaches and algorithms addressing the problem in Section 4 and conduct a comprehensive analysis of the computational aspects associated with each approach in Section 5. Concluding the substantive

portions of the paper, we execute simulations to evaluate the algorithms and approaches in Section 6, presenting and scrutinizing the results in detail.

Related Work We mention related work from both the social choice literature and chord sequence generation. First, we mention the work on general aggregation in metric spaces [5, 23], that also includes suggestions on how to perform text aggregation. Another model that is relevant is that of *multiple attribute list aggregation* [4], in which a sequence of elements (not necessarily text characters) are the output of the social choice instance. We also wish to mention judgment aggregation [11], which corresponds to a very general social choice setting and for which some of our algorithms are related, and get inspiration from the general work on aggregating preferences under constraints in formatted ways [13] Next we mention works on decision-theoretic planning techniques into automatic harmony generation and chords progression generation based on stochastic processes [14, 8, 22]. Our work makes assumptions for simplicity needs that differ from the classic chord progression problems, as we are interested in aggregation of chords preferences of agents, with respecting the probability of a chord progression to appear (based on pre-trained 2-gram model), but with no explicit use of the melody it self– we assume the user is responsible of giving a valid harmony to a melody, and we offer an aggregation method.

2 Musical Preliminaries

While the focus of the paper is rather on its social choice aspects, it is nevertheless essential to provide a foundational understanding of the musical elements we will be working with. Music, like other complex systems, can be viewed mathematically. At its core, music is composed of three fundamental components: *rhythm*, *melody* and *harmony*. Below, to make the preliminaries accessible also to readers without a background in music theory, we break these components using a more abstract, mathematical framework [12, 17].

Rhythm Rhythm in music is the organized arrangement of sound and silence within time. A melody consists of beats, grouped into sections called bars or measures, similar to paragraphs in writing. Each bar, determined by a time signature, contains a specific number of beats, indicating the beat count and the type of note representing one beat. This rhythmic structure forms the pulse of a song, guiding musicians and engaging listeners with its rhythmic pattern.

Melody The melody, in its simplest form, can be seen as a sequence of notes played one after the other. Each note can be represented using an "alphabet", which consists of a finite set of symbols. These symbols are include letters such as C, D, E, and so on (there is, indeed, a different, equivalent "alphabet" in which the symbols are Do,

¹ In this work, we present an aggregation method for harmonies without addressing the melody. We assume for simplicity needs, that the user is responsible for providing a valid harmony to accompany a melody, and we solely offer the aggregation methods.



(a) A simple harmonization of a melody.



(b) A more complex harmonization of the same melody.

Figure 1: Comparison of different harmonizations.

Re, Mi, which we, however, will not use here). These symbols represent different pitches or musical notes. E.g., consider the melody of the well-known nursery rhyme "Twinkle, Twinkle, Little Star." It can be broken down into a sequence of note symbols: C-C-G-G-A-A-G. Note how the arrangement of these symbols forms the song's melody.

Harmony As we are interested in collaborative harmonization, the harmony will be the crucial aspect of the music that we will be concentrated on. In particular, in this paper we focus our investigation on the harmony; and take a close look at its rules and grammar.

Acoustically, harmony complements the melody by introducing a layer of complexity to music. It focuses on the simultaneous sounding of multiple notes, forming so-called chords; the notes of these chords are played in parallel (simultaneously) to the notes of the main melody. Continuing our mathematical framework, chords can be considered as combinations of note symbols. In our "musical alphabet", a chord like "C-E-G" consists of symbols representing individual notes played together (in addition to the main melody). The structure within these chord sequences arises from the specific combination of notes and their temporal arrangement.

The relationship between harmony and melody is fundamental to music. Melodies are often played over a backdrop of harmonies, creating rich and emotionally expressive compositions. The choice of chords and their sequence can significantly influence the mood and feel of a musical piece, much like how words and their arrangement in a language can convey meaning and tone. For example, playing a melody over a *C major* chord (containing C, E, and G) can result in a different emotional tone than playing the same melody over an *A minor* chord (containing C, E, and A).

Example 1. *Sub-figure 1a and sub-figure 1b present the same melody on their upper staves, accompanied by different harmony symbols like Fmaj7, G7, Cm7, and E7. However, the bottom staves in both sub-figures depict the differing harmonic interpretations through the specific note collections associated with the harmony symbols shown on the upper staves.*

2.1 A Basic Grammar of Harmony

Our main motivation for studying the collaborative harmonization problem is that harmony can be naturally represented as text (where

<p>CMaj7, Cm7, CmMaj7, C7, CdimMaj7, Cdim7, Cm7b5, Cm6, C+7, C+maj7,</p> <p>DbMaj7, Dbm7, DbmMaj7, Db7, DbdimMaj7, Dbdim7, Dbm7b5, Dbm6, Db+7, Db+maj7,</p> <p>DMaj7, Dm7, DmMaj7, D7, DdimMaj7, Ddim7, Dm7b5, Dm6, D+7, D+maj7,</p> <p>Ebmaj7, Ebm7, EbmMaj7, Eb7, EbdimMaj7, Ebdim7, Ebm7b5, Ebm6, Eb+7, Eb+maj7,</p> <p>Emaj7, Em7, EmMaj7, E7, EdimMaj7, Edim7, Em7b5, Em6, E+7, E+maj7,</p> <p>FMaj7, Fm7, FmMaj7, F7, FdimMaj7, Fdim7, Fm7b5, Fm6, F+7, F+maj7,</p> <p>Gbmaj7, Gbm7, GbmMaj7, Gb7, GbdimMaj7, Gbdim7, Gbm7b5, Gbm6, Gb+7, Gb+maj7,</p> <p>GMaj7, Gm7, GmMaj7, G7, GdimMaj7, Gdim7, Gm7b5, Gm6, G+7, G+maj7,</p> <p>AbMaj7, Abm7, AbmMaj7, Ab7, AbdimMaj7, Abdim7, Abm7b5, Abm6, Ab+7, Ab+maj7,</p> <p>AMaj7, Am7, AmMaj7, A7, AdimMaj7, Adim7, Am7b5, Am6, A+7, A+maj7,</p> <p>Bbmaj7, Bbm7, BbmMaj7, Bb7, BbdimMaj7, Bbdim7, Bbm7b5, Bbm6, Bb+7, Bb+maj7,</p> <p>BMaj7, Bm7, BmMaj7, B7, BdimMaj7, Bdim7, Bm7b5, Bm6, B+7, B+maj7.</p>

Figure 2: The alphabet of chords used in this study.

each chords corresponds to its character) and, more importantly, it has some general structured grammar that is more basic than general text documents. In what follows we provide a simplified view of the grammar of harmony that is useful for our purposes.

In this paper, we consider a finite set of chords we address as "alphabet of chords" as shown in Figure 2 (this is not a complete list of all chords used in western music, though, but it is sufficient for the purposes of our study). Correspondingly, an harmonic sequence is simply a sequence of characters from that alphabet of chords.

Chord Similarity – Spatial Grammatic Aspect Considering the chords shown in Figure 2, it is important to realized that, acoustically, certain chords are more similar to others. Essentially, this is somehow similar to the fact that certain **words** are more similar to other words (e.g., bicycle and bike are quite similar to each other, while apple and spaceship are perhaps more semantically distant).

Correspondingly, it is useful to consider a distance metric to quantify chord similarity. Below we discuss two options.

- **Jaccard Distance:** Recalling that each chord (internally) contains few notes, this metric measures dissimilarity by comparing the intersection size between the notes of two chords (treated as sets of 4 notes) to the size of their union. It accounts for overlapping similar-pitch notes, providing a similarity measure that ranges from 0 (no similarity) to 1 (identical chords).

Concretely, the Jaccard distance between two chords A and B is

$$d_{\text{Jaccard}}(A, B) = 1 - \frac{|A \cap B|}{|A \cup B|}, \quad (1)$$

where $|A \cap B|$ denotes the number of common notes between chords A and B , and $|A \cup B|$ represents the total number of unique notes in both chords.

Example 2. *In Figure 3, we can see that CMaj7 (C E G B) and FMaj7 (F A C E) have two common notes, namely C and E, and 6 unique notes - B, C, E, F, G, A. So their Jaccard distance is $\frac{2}{3}$.*

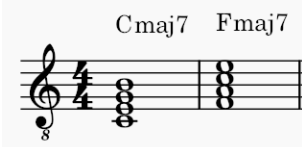


Figure 3: Two Chords: CMaj7 and FMaj7.

- **Tonal Distance:** Besides the simple note-counting process that underpins the Jaccard distance, there are other psychoacoustic features that affect chord similarity. The tonal distance relates to the *acoustic-functional relationship* between chords: it assesses the harmonic function of chords, rather than focusing solely on the pitch content [16, 15].

As the concept of *acoustic-functional relationship* between chords is challenging to model directly, we have chosen the Jaccard distance as our metric for this work although it is a less common "chord-distance" in practice, as we are interested in the aggregation process itself. We speak about this decision more in section 8 and recommend investigating other distance metrics for future work.

Chord Progression – Temporal Grammatical Aspect Besides the similarity (and interchangeability) of chords, there is also importance to chord progression – how chords in a sequence relate to each other. Essentially, this is somehow similar to the fact that certain **words** are more likely to come after other words (e.g., the word *dog* is somehow likely to come after the word *barking* but perhaps less likely to come after the word *fruitful*; this concept is known as the n-gram model in natural language processing [6]).

Concretely, in this paper, for the calculation of transition probabilities used in the context of chord progressions, we took a data-based approach. In particular, we considered a data set consisting of 1,410 Jazz songs (iRealPro – <https://www.irealpro.com/main-playlists>). To extract the chord symbols from this data set, we utilized the <https://github.com/pianosnake/ireal-readerireal-reader> tool. After acquiring the chord symbols, our next step involved converting each chord into a simplified representation using the alphabet shown in Figure 2. For the purpose of aggregation, we filtered 1015 songs from the data set (in particular, those that are precisely 32 bars in length, each encompassing a maximum of two chords per bar; technically, when encountering instances where only a single chord was present in a bar, we replicated it to ensure uniformity and consistency throughout the data set). Then, we used the adjusted data set to compute the probabilities of two chords being successive in an harmonic sequence.

As a result, we have a complete, directed graph with weighted arcs that represent the probability of one chord appearing after another, defined as $G = (V, w)$. In this context, each vertex $v \in V$ corresponds to a chord from "chords alphabet" as shown in Figure 2. The set of edges, E encompasses all possible arcs, where the weight $w_{(u,v)}$ on the arc $(u, v) \in E$ signifies the degree of smoothness associated with transitioning from chord u to chord v , with $0 \leq w_{(u,v)} \leq 1$.

3 Formal Model

We describe our formal model and discuss how to evaluate the quality of different winning harmonizations.

A Formal Model of Collaborative Harmonization With respect to a certain "harmonic alphabet A (containing the set of m possible chords), an *instance* of our model contains the following ingredients:

- A given *size* k of the harmonic sequence to produce.

- A set $V = \{v_1, \dots, v_n\}$ of n agents; each agent suggests its ideal harmonic sequence, denoted by v_i , where $v_i \in A^k$.

It is thus convenient to denote an instance of the model by (k, V) . Given an instance (k, V) of the model, a *solution* W corresponds to an aggregated harmonic sequence; formally, $W \in A^k$.

An *aggregation method* (i.e., a voting rule) for the setting of collaborative harmonization takes an instance (k, V) as its input and outputs a solution W .

Example 3. Figure 4 is an example of an input and an output of our algorithms; the example contains 5 agents (depicted on the left) and a possible aggregated harmonic sequence (depicted on the right).

Remark 4. Note that, perhaps surprisingly, the melody itself is not part of instances of our model, as we address only the harmonization; however, refer to Section Conclusions to a more elaborate discussion regarding this point.

Solution Quality What is missing from the section above is a discussion on how to evaluate the quality of solutions (i.e., aggregated harmonic sequences) of instances of our model. Informally speaking, we are looking for aggregation methods whose output strike a balance between these two aspects: (1) first, a solution shall respect the suggestions of the agents – i.e., the agents ideal harmonic sequences shall be taken into account by the aggregation algorithm; and (2) second, a solution should be musically appealing on its own.

There are several ways to formally capture these two desires. Below we describe our approach, which builds on (1) chord similarity for the first aspect; and (2) chord progression probabilities for the second aspect.

3.1 Agent Satisfaction

Our approach at the consideration of the correspondence between the ideal harmonic sequences of the agents and a possible winning harmonization is to define *agent satisfaction* (this follows the utility-based approach in social choice, such as used, e.g., for participatory budgeting [21]). Concretely, consider some agent v_i and a possible winning harmonization W ; intuitively, the more similar v_i is to W (with respect to the metric used for chord similarity) the more satisfied v_i shall be. Formally, we define as follows:

- Let d be the Jaccard distance (refer to Section Similarity).
- We then define, for agent i voting as v_i and a possible winning harmonization W , the satisfaction of agent i from W to be (assume that $|v_i| = |W| = k$):

$$sat(i, W) := \sum_{j \in [k]} d(v_i[j], W[j]) .$$

Using the concept of agent satisfaction, one may, e.g., consider the goal of maximizing the social welfare (i.e., maximizing the sum of agent satisfaction $\sum_{i \in [n]} sat(i, W)$). In this paper we largely take this utilitarian approach.

3.2 Musical Coherence

Besides corresponding to the ideal harmonic sequences of the agents, a solution of a collaborative harmonization instance shall also be musically coherent. Our approach towards the assessment of the musical quality of a potential winning harmonization builds on the 2-gram approach described in Section Musical Preliminaries. While, indeed,

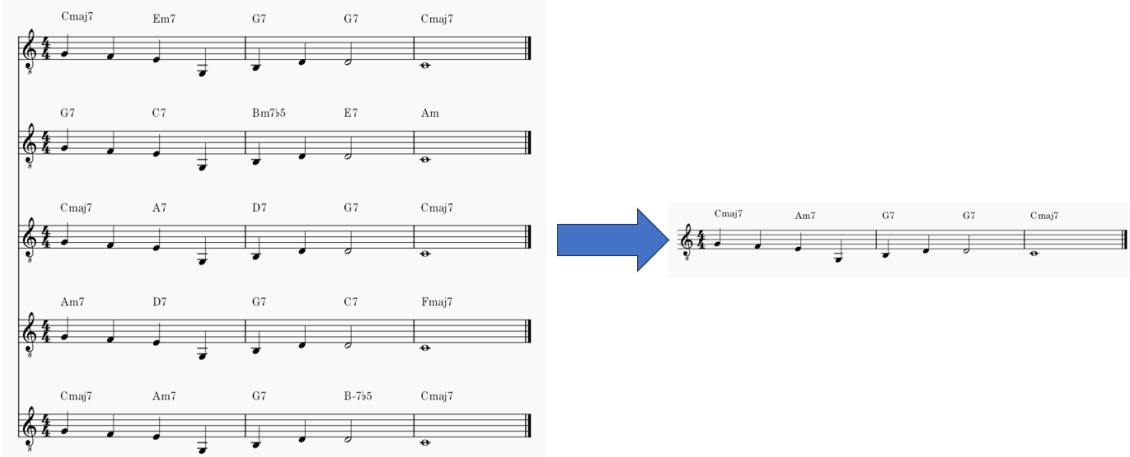


Figure 4: An example of an instance to the collaborative harmonization problem (on the left) with one possible aggregated harmonic sequence (on the right).

musical coherence is more involved than our simple 2-gram approach (such as using an n-gram), our approach nevertheless captures the basics of chord progression: acoustically, a smoother transition sequence between chords contributes to a smoother “flow”.

Remark 5. While it makes sense to consider Musical Coherence of the whole harmonization as a whole, a natural simplification is to consider pairwise transitions.

Formally, we define Musical Coherence as follows:

- We start by the *chord transition graph* G , as described in Section Musical Preliminaries.
- Next, we define the *Musical Coherence* of a potential winning harmonization, denoted as W . It is calculated as the product of consecutive smoothness values represented by the weights of G , denoted by w . Formally, the Musical Coherence of W can be expressed as:

$$sat(W) := \sum_{j \in [k-1]} \log(w(W[j], W[j+1])) .$$

This mathematical framework allows us to quantitatively evaluate the musical quality of a harmonization by considering the smoothness of transitions between chords in the context of the defined chord transition graph.

3.3 Example and Discussion

Consider the following collaborative harmonization scenario involving a sequence of 4 chords and a community of 3 agents. The ideal harmonic sequence of the three agents are as follows:

- Agent 1: Cmaj7, Dm7, Db7, Cmaj7.
- Agent 2: Am7, Dm7, E7, Am7.
- Agent 3: Cmaj7, Fmaj7, G7, Am7.

Suppose we have a potential solution, as follows

1. Cmaj7, Dm7, E7, Am7.

Chord	Notes
Cmaj7	C,E,G,B
Am7	A,C,E,G
Dm7	D,F,A,C
Db7	Db,F,Ab,B
Fmaj7	F,A,C,E
E7	E,Ab,B,D
G7	G,B,D,F

Table 1: Chords and Notes they Contain.

Agent satisfaction (based on Jaccard distance of common notes of two chords) is calculated using table 1 as follows:

$$\text{Agent 1: } (0 + 0 + \frac{2}{3} + 0.4) = 1.0667$$

$$\text{Agent 2: } (0.4 + 0 + 0 + 0) = 0.4$$

$$\text{Agent 3: } (0 + 0.4 + \frac{2}{3} + 0) = 1.0667$$

Thus the total satisfaction is 2.53334.

4 Approaches and Algorithms

In this section, we develop and discuss algorithms for solving instances of collaborative harmonization. For convenience, we will use the following notation:

- The preferences of the agents are denoted using a matrix \mathcal{B} of size $n \times k$ (for n agents and k being the length of the harmonic sequence). Each element $b_{i,j}$ in this matrix represents the chord selected by agent i in position j .
- Given such an instance, a solution is denoted by $W \in A^k$.

Plurality We first consider the adaptation of the Plurality rule to our setting. In this simple aggregation method, we consider each chord-position independently; and, seek to find the most popular chord for every position of the k chords. Procedurally, Plurality works by calculating a score $M(W)$ for each chord $W[j]$ as follows:

$$M(W) = \sum_{i=1}^n \sum_{j=1}^k \mathbb{1}(b_{i,j} = W[j]) . \quad (2)$$

In this equation, $b_{i,j}$ represents the chord selected by agent i in position j of matrix B , and \mathbb{I} is the indicator function. And, the output of the Plurality Algorithm is the chord W that maximizes $M(W)$, represented as $\arg \max_W (M(W))$.

Kemeny Rule The Kemeny rule [1] is a well-known aggregation method that is applicable to many social choice settings. In the standard model of ordinal-based social choice, Kemeny proceeds by assigning a score to each possible social welfare function (i.e., a linear order of the available candidates) on the distance from that permutation to all other individual permutations (i.e., votes); and selecting the one for which this sum of (swap) distances is the smallest. We apply this concept to our setting. This is accomplished by using the Jaccard distance function described in Section Musical Preliminaries. Correspondingly, in our adaptation of Kemeny, at each position within the chord sequence $W \in A^k$, we wish to select a chord $W[j]$ for position j in a way that minimizes the cumulative distance from all individual chords $b_{i,j}$. Formally, we define the Kemeny optimization quantity as follows:

$$K(W) = \left(\sum_{i=1}^n \sum_{j=1}^k d(b_{i,j}, W[j]) \right). \quad (3)$$

Here, the function d quantifies the dissimilarity between two chords. In this paper, we consider d as the Jaccard distance, and so $d(b_{i,j}, W[j]) \in [0, 1]$ for all i, j . The output of the algorithm is $\arg \min_W (K(W))$.

PAV Next, we are interested in proportionality [19]; and, to this end, we adapt the PAV voting rule to our setting. Concretely, we consider the *proportional Borda count* rule: this rule utilizes the harmonic series to assign scores to candidates or preferences based on their rankings, aiming for proportional representation according to their ranking [9]. In our context, the objective is to create a chord sequence that effectively represents the diverse preferences of the agents, proportionally. To accomplish this, we employ an objective functions that is largely similar to the proportional Borda count, albeit where the Borda score is replaced by our Jaccard metric. Formally, the objective function is defined as follows:

$$P(W) = \left(\sum_{i=1}^n \sum_{j=1}^k \frac{1}{j} \cdot (s(U(B_i, W))[j]) \right). \quad (4)$$

In this equation:

- U is a utility function that takes two k -sized vectors and returns a k -sized vector of utilities. We set $U(B_i, W)[j] = 1 - d(b_{i,j}, W[j])$.
- B_i represents a k -sized vector of chords, representing the i th row in matrix B .
- s denotes a sorting function that takes a k -sized vector and arranges it in descending order.

The output of the algorithm is $\arg \max_W (P(W))$. As this problem is naturally NP-Hard (see Section Computational Complexity), for its computation (in our computer-based simulations), we utilized a heuristic approach a local search algorithm of simulated annealing.

Clustered-Kemeny We introduce Clustered-Kemeny, an algorithm based on the Kemeny voting rule, adapted for the natural division of musical pieces into sections. The goal is to identify clusters of individuals with similar chord preferences, enabling the partitioning of the musical piece into sections and matching each cluster to its most representative section.

Assuming a given partition of the harmonic sequence's length k into $x \leq n$ continuous sections, we formulate a linear program for optimizing voter clustering and section matching. The partition, represented by vector Z , must satisfy specific conditions.

The problem is divided into two nested sub-problems: (1) Given a partition Z , find the optimal clustering of agents into sections; and (2) Find the optimal Z from all possible partitions into at most X sections, returning the optimal agent clustering. The objective is to maximize total satisfaction by minimizing the distance of selected chords and their respective sections within the clustering. Formally:

$$\min_{Z \in \text{partitions}} \min_{a_i \in Z, W} \sum_{z \in Z} \sum_{i=1}^n \sum_{j=1}^k P(j, z, Z) \cdot Q(i, z, Z) \cdot (d(b_{i,j}, W[j]))$$

Where: (1) $Q(i, z, Z)$ indicates whether agent i is in section z (1 if true, otherwise a specified value less than 1); (2) $P(j, z, Z)$ is 1 if position j belongs to section z , 0 otherwise; and (3) For the last section: $P(j, x, Z)$ is 1 if $Z[x] \leq j \leq k$, 0 otherwise. These collectively define the optimization problem for the Clustered-Kemeny algorithm.

4.1 2-Gram-Based Approaches

Note that, above, we have only relied on the Jaccard distance for quantifying the chord similarity (between the ideal harmonic sequence of each agent and some proposed solution). Next we consider also the musical coherence (or harmonic flow), corresponding to the 2-gram approach described in Section Musical Coherence.

In particular, the following approaches build upon the previous methodologies and share the same input as their predecessors; but they aim to maximize a new objective function, denoted as G , which is defined as follows using transition probabilities p :

$$G(W) = - \sum_{i=0}^{k-1} (\log(p(W[i], W[i+1]))) . \quad (5)$$

Here, the transition probabilities p quantify the likelihood of transitioning from one chord, chord_i , to another chord, chord_{i+1} , within a sequence. These probabilities are derived from observed frequencies of such transitions in our data set as described in Section Musical Coherence.

These 2-Grams Based Algorithms expand upon the previous approaches, considering transition probabilities to enhance predictive accuracy while accommodating the collaborative nature of chord sequences. Below we formally describe the details of taking into account such transition probabilities in the computation of the specific aggregation algorithms described above.

Plurality with 2-gram The objective is given by $\arg \max_W (x_M \cdot M(W) + (1 - x_M) \cdot G(W))$, where $x_M \in (0, \dots, 1)$ is a constant.

Kemeny with 2-gram The objective is determined by $\arg \min_W (x_K \cdot K(W) - (1 - x_K) \cdot G(W))$, where $x_K \in (0, \dots, 1)$ is a constant.

PAV with 2-gram The objective is determined by $\arg \min_W (x_P \cdot P(W) - (1 - x_P) \cdot G(W))$, where $x_P \in (0, \dots, 1)$ is a constant.

Clustered-Kemeny with 2-gram The objective is determined by $\arg \min_W (x_{KC} \cdot KC(W) - (1 - x_{KC}) \cdot G(W))$, where $x_{KC} \in (0, \dots, 1)$ is a constant.

Problem	Complexity
Plurality	Poly-time
Kemeny	Poly-time
Proportional	NP-hard
Plurality with 2-gram	Poly-time
Kemeny with 2-gram	Poly-time
Proportional with 2-gram	NP-hard
Clustered-Kemeny	NP-hard
Clustered-Kemeny with 2-gram	NP-hard

Table 2: Various problems of collaborative harmonization and their computational complexity.

5 Computational Complexity

We present the computational complexity of the aggregation goals defined above in Table 2. Additionally, we provide two sketches of hardness proofs. The complete proofs, along with all other missing proofs of the complexity results, are included in the supplementary material.

Theorem 1. *PAV and PAV with 2-gram are NP-hard.*

Proof. We prove the NP-hardness of the Proportional algorithm by noting that it includes Proportional Approval Voting (PAV) as a special case, where $U[i] \in \{0, 1\}$. This implies Proportional is NP-hard [18]. By setting $x_p = 1$, which reduces to the Proportional algorithm, we establish that Proportional with 2-gram is also NP-hard. \square

Theorem 2. *Clustered-Kemeny and Clustered-Kemeny with 2-gram are NP-hard.*

Proof. We reduce the k-median problem, known to be NP-Hard [7], to Clustered-Kemeny. Given n strings $S = \{s_1, s_2, \dots, s_n\}$ of length ℓ , a distance metric d , and a threshold t , the k-string median problem seeks k median strings $M = \{m_1, m_2, \dots, m_k\}$ such that:

$$\sum_{i=1}^n \min_{1 \leq j \leq k} d(s_i, m_j) \leq t.$$

We construct a Clustered-Kemeny instance by replicating each input string k times, transforming it into an agent representation (e.g., "abc" becomes "abcabcabc"). Partitions and lengths are defined as $Z = \{L, L \cdot 2, L \cdot 3, \dots, L \cdot k\}$. Setting $Q(i, z, Z) = 0$ for each agent i , we show that the k-median problem is a yes instance if and only if Clustered-Kemeny is a yes instance, establishing its NP-hardness.

For Clustered-Kemeny with 2-gram, setting $x_{KC} = 1$ aligns it with the hardness proof of Proportional with 2-gram, thus proving it NP-hard as well. \square

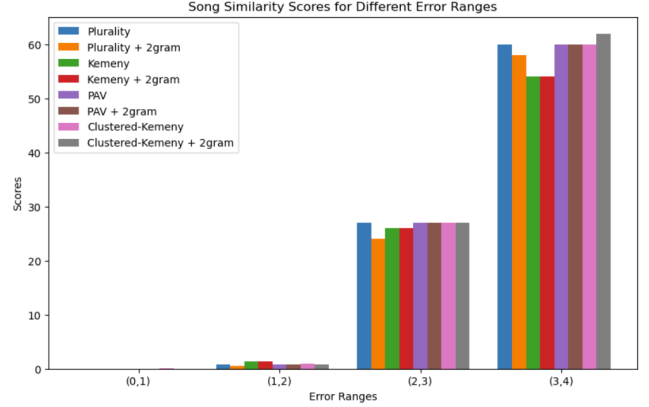
6 Computer-Based Simulations

Next, we report on computer-based simulations to evaluate the quality of the proposed algorithms. Given the NP-hard nature of the problems, we used a heuristic approach with a simulated annealing solution. The local search was initiated with the Plurality algorithm, and each search had 1000 iterations.

Due to the scarcity of relevant data for collaborative harmony composition, we adopted a semi-artificial approach. We used real harmonizations of songs and introduced random perturbations to simulate ideal harmonies by different agents.

For the dataset, we used 8, 16, and 32 agents, each contributing an ideal chord progression. Each agent's progression had 8, 16, and

Figure 5: Song Similarity vs Error Ranges for 32 Agents.



32 variations, respectively, of the original progression of songs from a processed dataset of 1015 jazz songs (see section 2.1). Variations were introduced by random chord swaps within ranges (0, 1), (1, 2), (2, 3), and (3, 4). The new chord was selected based on Jaccard distances, ensuring musical coherence.

For the 2-gram-based algorithms, we fine-tuned the weights through several rounds of testing, arriving at optimal weights: $X_M = 0.5$, $X_K = 0.9$, $X_P = 1 - 2e^{-4}$, and $X_{KC} = 0.9$.

Evaluation metrics were based on three measures for any solution W :

- **Song Similarity Measure:** Assesses the proximity of the aggregated chord progression to the original sequence. Lower distances indicate higher adherence to the original progression:

$$\sum_{j=1}^m d(W[j], \text{original song}[j]).$$

- **Cluster Coherence:** Evaluates the coherence of chords within each 16-bar section of the aggregated chord progression. Lower cluster distances indicate smoother transitions:

$$\frac{1}{(m-16) \cdot n} \sum_{i=1}^n \sum_{j=1}^{m-16} \sum_{k=j}^{j+16} d(W[k], b_{i,k}).$$

- **Musical Coherence:** Measures the appropriateness of the aggregated chord progression within the context of the 2-gram. Higher scores indicate more harmonious sequences, as detailed in the supplementary material.

7 Results and Discussion

The full results shown in the supplementary material the average scores of all 1015 instances, each scaled by a factor of 100. These scores are associated with various parameters: a specific measurement, an algorithm used, the number of chord swaps employed to generate different variations, and the number of agents (representing the variations).

Figure 7 demonstrates a trend where an increase in the number of chord swaps (from 0 to 1) leads to a significant decline in musical coherence. This decline occurs because employing the Jaccard distance for chord swaps often generates chord progressions or variations that are less probable within the data set. Consequently, these less probable variations adversely affect the outcomes produced by the aggregation algorithms.

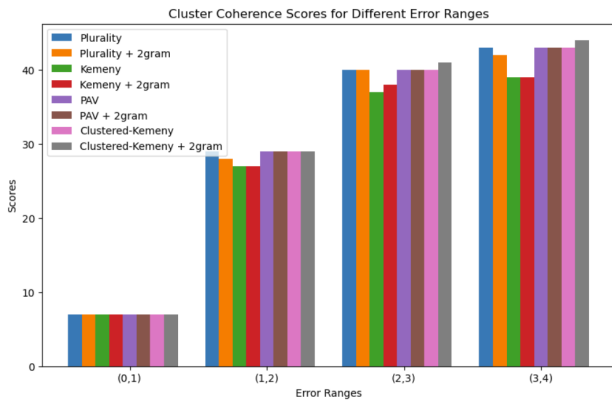


Figure 6: Cluster Coherence vs. Error Ranges for 32 Agents

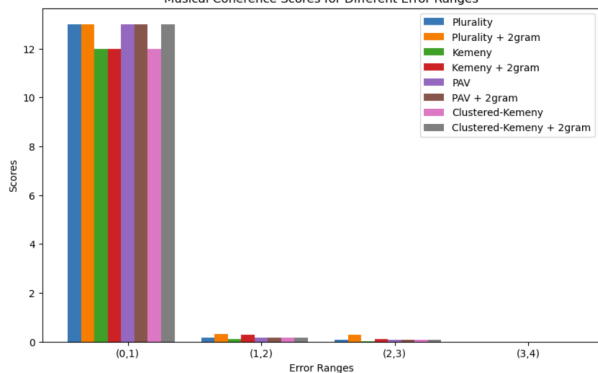


Figure 7: Musical Coherence vs. Error Ranges for 32 Agents.

Further, direct analysis of the results are given in the supplementary material and the source code can be found here <https://anonymous.4open.science/r/CollaborativeHarmonization/Probabilities%20and%20Distances.txt>.

7.1 Insights and Implications

Here are some of the main insights we derive from the results of the simulations:

- **Robustness to Errors**- Plurality-based algorithms exhibit robustness to errors, maintaining better proximity to original sequences and higher Musical Coherence across varying error ranges and agent counts.
- **Balancing Chord Adherence and Coherence**- Kemeny methods strike a balance between chord adherence and musical coherence, particularly effective when integrated with 2-gram.
- **Scalability Concerns**- Plurality-based algorithms demonstrate scalability and consistent performance across different agent counts. In contrast, Clustered-Kemeny might face challenges in larger-scale settings, potentially indicating scalability issues.
- **Clustering** - It is intriguing to note that despite the anticipated ability of Clustered-Kemeny to generate coherent and adherent sections, it appears to have fallen short in achieving this objective. One plausible explanation could be attributed to the creation of numerous small clusters, each smaller than 16 bars, coupled with only a few agents assigned to each cluster. This scenario led to the formation of multiple 16-bar clusters that lacked coherence, consequently resulting in an inadequate evaluation of the measure’s effectiveness.

7.2 Summary

Essentially, we observe that, while Kemeny + 2-gram doesn’t consistently yield the best results across all combinations of measures, number of agents, and number of errors affecting the variations of a song, it reliably positions itself in a solid middle ground. It may not excel in every scenario, but it consistently maintains a respectable performance, offering a balance among different evaluation measures when considering diverse variations and conditions.

8 Conclusions

In this study, our primary goal was to explore collaborative harmonization as a case study, drawing parallels to the structured nature of text aggregation. This comparative approach allowed us to gain insights by leveraging the structured framework inherent in harmonization, potentially informing strategies for text aggregation in the future. We established a formal model, introduced various algorithms, and found that simpler algorithms consistently performed better across all considered measures.

Moving forward, we suggest several areas for future work. Firstly, the chosen Jaccard distance metric presented in this work, is less practical and does not seem to capture the entire of essence of musical harmonization and thus investigating other, more relevant distance metrics is needed. Secondly Our research focused on pairwise transitions forming 2-grams for harmony suitability. A more refined approach involving n -grams would provide a more comprehensive analysis of chord relationships beyond consecutive pairs.

Additionally, our study exclusively addressed harmonization, neglecting melody considerations. Future investigations should integrate melody into the harmonization process to enhance musical coherence and quality.

An extension of our work could involve using more common practiced distance metrics and empirical assessments with real musicians, providing valuable insights into how these algorithms are perceived and valued in practical musical contexts.

All our algorithms share a vertical nature, comparing chords in the same position. Addressing the limitation of not recognizing equivalence in non-identical chord positions is a pertinent avenue for future research to improve contextual understanding.

For future endeavors in text aggregation, recognizing and utilizing structured string or text scenarios can greatly facilitate more meaningful aggregation. Aligning the aggregation process with the inherent structure present in the text can significantly enhance the quality and relevance of aggregated outcomes.

References

- [1] A. Ali and M. Meilă. Experiments with kemeny ranking: What works when? *Mathematical Social Sciences*, 64(1):28–40, 2012.
- [2] K. J. Arrow, A. Sen, and K. Suzumura. *Handbook of social choice and welfare*, volume 2. Elsevier, 2010.
- [3] H. Aziz and N. Shah. Participatory budgeting: Models and approaches. *Pathways Between Social Science and Computational Social Science: Theories, Methods, and Interpretations*, pages 215–236, 2021.
- [4] E. Briman and N. Talmon. Multiple attribute list aggregation and an application to democratic playlist editing. In *European Conference on Multi-Agent Systems*, pages 1–16. Springer, 2023.
- [5] L. Bulteau, G. Shahaf, E. Shapiro, and N. Talmon. Aggregation over metric spaces: Proposing and voting in elections, budgeting, and legislation. *arXiv preprint arXiv:1806.06277*, 2018.
- [6] W. B. Cavnar, J. M. Trenkle, et al. N-gram-based text categorization. In *Proceedings of SDAIR '94*, volume 161175, page 14, 1994.
- [7] M. Charikar, S. Guha, É. Tardos, and D. B. Shmoys. A constant-factor approximation algorithm for the k-median problem. In *Proceedings of*

- the thirty-first annual ACM symposium on Theory of computing*, pages 1–10, 1999.
- [8] B. J. Clement. Learning harmonic progression using markov models. *EECS545 Project, University of Michigan*, 1998.
 - [9] P. Dey, N. Talmon, and O. Van Handel. Proportional representation in vote streams. *arXiv preprint arXiv:1702.08862*, 2017.
 - [10] P. Faliszewski, P. Skowron, A. Slinko, and N. Talmon. Multiwinner voting: A new challenge for social choice theory. *Trends in computational social choice*, 74(2017):27–47, 2017.
 - [11] D. Grossi and G. Pigozzi. *Judgment aggregation: a primer*. Springer Nature, 2022.
 - [12] M. Levine. *The jazz theory book*. " O'Reilly Media, Inc.", 2011.
 - [13] M. Li, Q. B. Vo, and R. Kowalczyk. Aggregating multi-valued cp-nets: a csp-based approach. *Journal of Heuristics*, 21:107–140, 2015.
 - [14] J.-F. Paiement, D. Eck, and S. Bengio. A probabilistic model for chord progressions. In *Proceedings of the Sixth International Conference on Music Information Retrieval (ISMIR)*, 2005.
 - [15] V. Persichetti. *Twentieth-Century Harmony: Creative Aspects and Practice*. W.W. Norton and Company, 1961.
 - [16] M. Rohrmeier. Towards a generative syntax of tonal harmony. *Journal of Mathematics and Music*, 5(1):35–53, 2011.
 - [17] M. Rohrmeier. The syntax of jazz harmony: Diatonic tonality, phrase structure, and form. *Music Theory and Analysis (MTA)*, 7(1):1–63, 2020.
 - [18] P. Skowron, P. Faliszewski, and J. Lang. Finding a collective set of items: From proportional multirepresentation to group recommendation. *Artificial Intelligence*, 241:191–216, 2016.
 - [19] P. Skowron, M. Lackner, M. Brill, D. Peters, and E. Elkind. Proportional rankings, 2016. URL <https://arxiv.org/abs/1612.01434>.
 - [20] P. Skowron, D. Stolicki, S. Szufa, and N. Talmon. Participatory budgeting: Data, tools, and analysis, 2023.
 - [21] N. Talmon and P. Faliszewski. A framework for approval-based budgeting methods. In *Proceedings of AAAI '19*, volume 33, pages 2181–2188, 2019.
 - [22] L. Yi and J. Goldsmith. Automatic generation of four-part harmony. *BMA*, 268, 2007.
 - [23] G. B. Zvi, E. Leizerovich, and N. Talmon. Iterative deliberation via metric aggregation. In *International Conference on Algorithmic Decision Theory*, pages 162–176. Springer, 2021.