Automatic species counterpoint composition by means of the dominance relation

Maciej Komosinski Piotr Szachewicz

Institute of Computing Science
Poznan University of Technology
Piotrowo 2, 60-965 Poznan, Poland

Abstract

This paper introduces a new method for automated composition of the first species counterpoint. The method employs the *dominance relation* – a fundamental notion in the area of multi-criteria analysis, never used so far to analyze counterpoints in the context of algorithmic composition research. The dominance relation allows for analysis of a number of evaluation criteria without making any assumptions on the importance of each criterion; this way aggregations of criteria that would lead to loss of information are avoided. Seven criteria are used in this work to evaluate counterpoints in large-scale computational experiments, and the distributions of criteria values are demonstrated for a few types of cantus firmi including popular tunes, Gregorian chants, ascending or descending musical lines, and randomly generated melodies. Mutual discordance of these criteria is also evaluated, revealing pairs of criteria that correlate and others that are conflicting.

Keywords: counterpoint; algorithmic composition; computational musicology; multi-criteria analysis; dominance relation

1 Introduction

Counterpoint is the art of composing a melodic line to some fixed melody – the *cantus firmus*, so that these two lines played simultaneously obey a set of harmonic and melodic rules. Species counterpoint, also called the *strict counterpoint*, was invented for educational purposes. The rules of this counterpoint were thoroughly codified in 1725 by Johann Joseph Fux in *Steps to Parnassus*, where he describes five species of counterpoints of increasing complexity (Fux, Mann, and Edmunds, 1965). In this paper we are concerned with the first species of counterpoint.

1.1 Algorithmic composition

Since the rules of counterpoint can be expressed formally, nowadays they can be employed in algorithms and computer programs. A research area that focuses on development of algorithms that are capable of creating musical compositions is called algorithmic composition (Papadopoulos and Wiggins, 1999; Maurer, 1999). Programs for automatic composition employ various approaches and techniques, such as fuzzy logic (Yilmaz and Telatar, 2010), expert systems (Ebcioğlu, 1990), answer set programming (Boenn et al., 2011), learning from examples (Cope, 2004), probabilistic logic (Aguilera et al., 2010) and Markov chains (Farbood and Schoner, 2001).

The final version of this paper appeared in *Journal of Mathematics and Music* 9(1):75–94, 2015.
http://dx.doi.org/10.1080/17459737.2014.935816

Automatic counterpoint composition is a popular direction of research in algorithmic composition because of the strictly defined rules that need to be obeyed. This facilitates implementations of automatic composition systems. One of the approaches to automatic composition of counterpoint is to introduce penalty (or reward) values for each broken (or satisfied) rule, and then aggregate these values using an objective function. This way the quality of each counterpoint can be evaluated as a single number. When considering penalties as positive values, the objective function should be minimized so it is often called a loss function or a cost function. It can be defined as follows:

$$\sum_{i=1}^{N} p_i \cdot n_i \tag{1}$$

where p_i represents the penalty for breaking i-th rule, and n_i is the number of times the i-th rule was broken in a particular counterpoint. The task of finding the best possible solution to the problem is therefore equivalent to finding the counterpoint which has the minimum value of the cost function (i.e., the minimum weighted sum of penalties). The search for the counterpoint that minimizes or maximizes the objective function can be done using various optimization methods such as an exhaustive search, a best-first search (Schottstaedt, 1984), or metaheuristic algorithms including genetic and evolutionary techniques (Weale and Seitzer, 2003; Acevedo, 2005; Jelonek and Komosinski, 1999; Komosinski and Krawiec, 2000; Hapke and Komosinski, 2008).

The use of an additive function is known to have several drawbacks. First of all, it assumes that one can somehow determine the p_i values denoting the importance of each rule. The literature does not explicitly specify the exact, quantitatively expressed significance of each rule – the only information that is available is that some rules are more or less important than others:

> However, to return to the above-mentioned octave, the *batutta*, I shall leave to your discretion the use or avoidance of it; it is of little importance. (Fux, Mann, and Edmunds, 1965)

This means that the choice of p_i values is to a great extent arbitrary. Another important drawback of using the additive function is that breaking one very important rule is equivalent to breaking several less important rules, cf. (Roy and Vincke, 1981). For example, if in a given counterpoint a rule with penalty $p_1 = 3$ is broken once, then such a counterpoint has the same value of the quality function as another counterpoint in which a rule with penalty $p_2 = 1$ is broken three times. In reality however, these two counterpoints may be incomparable to each other, or it may be clear that one of them sounds much better than the other. Aggregations of criteria lead to loss of information and introduce trade-offs between criteria which are often undesirable and inconsistent with human perception of the quality of counterpoints.

Aware of the weaknesses of an additive function used to determine quality of counterpoints and motivated by interesting properties of the dominance relation discussed later, we propose a novel method that uses the dominance relation in order to find the set of best counterpoints for a given cantus firmus while preserving their diversity.

1.2 Computational musicology

Currently, techniques of computer science are often employed in order to solve musicological problems; this interdisciplinary field is known as computational musicology (Byrd and Crawford, 2002; Camilleri, 1993). On the other hand, computer science has an impact on musicology, which opens up new perspectives for musicological studies (Volk, Wiering, and van Kranenburg, 2011).

Examples of works in this area include developing tools for automatic harmonic analysis (de Haas et al., 2011), analysis of chord progressions (Steedman, 1984), automatic music genre classification (Tzanetakis and Cook, 2002), or simulation of perceiving music in an artificial society of agents (Coutinho et al., 2005). Studies of counterpoint performed within computational musicology include discovering patterns in polyphonic music (Conklin and Bergeron, 2010; Utgoff and Kirlin, 2006), analysis of counterpoints using Markov chains (Farbood and Schoner, 2001), a counterpoint

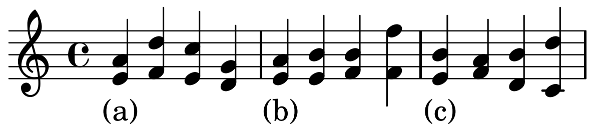

Figure 1: Examples of three types of motions: (a) direct motion, (b) oblique motion, (c) contrary motion.

grammar checker (Huang and Chew, 2005), and developing a mathematical theory of counterpoint (Mazzola et al., 2002; Junod and Mazzola, 2007).

Apart from implementing a computer program that uses the dominance relation to compose the first-species counterpoint, the other goal of this research has been to analyze the "solution space" (the full set) of counterpoints. Such analysis concerns relationships between the number of possible counterpoints and the length of the cantus firmus, comparisons between different types of cantus firmus (e.g., random or human-composed melodies), determining which evaluation criteria are easier to fulfill, or checking how often pairs of these criteria clash.

Such experiments increase our understanding of the nature of all possible counterpoints and relationships between the evaluation criteria, but also help us discover dependencies which can change the way a counterpoint is composed. This is how developments in computer science may have impact on our understanding of the art of counterpoint, and may stimulate research in musicology by providing new data.

2 Principles of the counterpoint and their implementation

In this section, a few basic definitions are introduced, and the set of counterpoint rules that has been used for all reported experiments is described. These notions are based on the counterpoint textbook by J. J. Fux et al. (Fux, Mann, and Edmunds, 1965).

2.1 Definitions

An **interval** is the distance between two notes that are played either simultaneously or one after another. Intervals can be divided into consonances, which are often described as pleasing to the ear, and dissonances, which sound harsher.

Consonances are further divided into perfect consonances (which include the unison, the perfect fifth, and the octave) and imperfect consonances (the major and minor third, the major and minor sixth). Dissonances include the major and minor second, the perfect fourth, the diminished fifth, the tritone, and the major and minor seventh. Note that nowadays the perfect fourth is also considered to be a consonance, but in the days when the counterpoint was developed, that was not the case. In this paper we follow the classical rules of the counterpoint and consider the perfect fourth to be a dissonance.

Intervals can be vertical (harmonic) if two notes are played simultaneously, or horizontal (melodic) if two notes are played one after another. Melodic intervals can be divided into skips and steps. Steps occur when a minor or a major second is used as a horizontal interval, otherwise the interval is a skip.

Motions denote the direction of melodies when one vertical interval is changed into another interval. There are three types of motions (Fig. 1):

- Direct motion – when melodies in both parts ascend or descend in the same direction.
- Oblique motion – when the melody in one part ascends or descends, and in the other part the previous note is repeated.

- Contrary motion – when one part ascends and the other part descends.

The contrary and the oblique motions are the preferred ones in the counterpoint.

A **church mode** is a set of notes from which a single counterpoint melody can be composed. Six church modes are used in the experiments reported here: Dorian, Phrygian, Lydian, Mixolydian, Aeolian and Ionian. The first five of them are considered because they were used in the first-species examples in the J. J. Fux et al. textbook (Fux, Mann, and Edmunds, 1965). The Ionian mode is included because it is typical for popular melodies which are also examined in this work. For all modes, the distance between the highest and the lowest note is an octave – the normal (perfect) ambitus.

2.2 Rules of the first species counterpoint

The first species counterpoint, also called *note against note*, is comprised of two voices having notes of equal length. The precise duration of these notes is insignificant, so we assume the whole notes. In the computational experiments, the following set of rules is used while composing a counterpoint to a given cantus firmus:

1. The only vertical intervals that can be used in the first species counterpoint are consonances – no vertical dissonances are allowed.
2. The first and the last vertical interval should be a perfect consonance.
3. The mode used for the counterpoint part must be the same as used for the cantus firmus part, except that it is transposed up or down by an octave.
4. Vertical perfect consonances should never be reached by a direct motion; moreover, direct motion should be used as rarely as possible.
5. The next to last vertical interval in the counterpoint must be a major sixth if the cantus firmus is in the lower part, otherwise it should be a minor third.
6. A single voice in a counterpoint should never jump by a tritone or by an interval larger than the perfect fifth (excluding the octave and the minor sixth, the latter interval can only be used if the voice ascends).
7. Imperfect consonances should be used as often as possible.
8. The same note should not be repeated too often.
9. A step is preferred to a skip.
10. It is inappropriate if two skips occur in a row and each is in the same direction.

The guidelines enumerated above are just a subset of all rules that can be used for the composition of the first species counterpoint. The choice of this particular subset is based on literature suggestions stating which rules are more important than others (Fux, Mann, and Edmunds, 1965), as well as on our arbitrary decisions. This choice provides a good starting point to perform multi-criteria analyses. The approach presented in this article could also be employed for other possible selections of criteria.

In order to simplify the problem, a one-octave distance is enforced between the octave intended for the cantus firmus part and the octave for the counterpoint part. This means that if the ambitus of the cantus firmus is around D2-D3, then the counterpoint would be composed within the D4-D5 octave – and vice versa. Thanks to that, the two parts never intertwine; intertwining would mean that some counterpoint notes would be above the cantus firmus, and some of them would be below.

The reason for using this simplification is that overlapping melodies could have an excessive influence on the results obtained. In this paper, different types of melodies are analyzed, one of which consists of successive ascending notes. For cantus firmi of this melody type, the phenomenon described above would be very common, as the ambitus of successive notes exceeds one octave if the number of notes is larger than 8. Thus, when comparing the characteristics of this cantus firmus melody group with the other groups, the obtained differences could result from the fact that the two melodies overlap and not from the innate characteristics of the groups of cantus firmi. As the latter is to be examined in this work, the intertwining of the two melodies was prohibited.

2.3 Implementation of counterpoint rules and the dominance relation

Some of the rules enumerated above are implemented at the stage of generating counterpoints, while others are implemented at the stage of evaluating counterpoints. The algorithm for automatic composition consists of three modules:

1. Generator – generates all possible counterpoints for the given cantus firmus. The generated counterpoints have to meet two rules: all notes must be selected from the given mode, and all vertical intervals must be consonances. Additionally, the first and the last interval in the counterpoint have to be perfect consonances.
2. Evaluator – evaluates each generated counterpoint using the following set of criteria. Each criterion determines the number of times a specific situation occurs in the counterpoint.
   (a) DirectMotion – direct motion.
   (b) NoteRepetition – a repeated note.
   (c) ImperfectConsonances – a vertical imperfect consonance.
   (d) NumberOfSkips – a skip (an interval larger than major second).
   (e) TwoSkipsInTheSameDirection – two skips one after the other and both in the same direction.
   (f) PerfectConsonancesByDirectMotion – a vertical perfect consonance is reached by a direct motion.
   (g) ForbiddenJumps – skips by a tritone or larger than a perfect fifth (except for minor sixth, which is allowable in the upward direction).
3. Search Engine – finds all interesting (i.e., non-dominated) counterpoints.

After the evaluation of every generated counterpoint using each of the criteria described above, the values of these criteria may be aggregated or not, depending on the goal. In this work we avoid aggregation and thus avoid introducing trade-offs between these criteria. Since they are often conflicting, we use the multi-criteria dominance relation to identify counterpoints that are better than others in any aspect, and to reflect a human goal of finding *interesting* counterpoints. Formally, we assume that a counterpoint c_1 is better than c_2 (i.e., *dominates* it), if c_1 is not worse (which means it is better or equally good) than c_2 on all criteria, and c_1 is strictly better than c_2 on at least one of these criteria. A counterpoint c that is not dominated by any other counterpoint in the set is called a *non-dominated counterpoint*, an *efficient counterpoint*, or a *Pareto optimal counterpoint* (Ehrgott, 2006; Doumpos and Grigoroudis, 2013; Komosinski et al., 2014).

Note an interesting characteristic of the dominance relation defined above: introducing more criteria will probably increase the number of non-dominated counterpoints as long as the criteria are in conflict (i.e., when comparing two counterpoints, one counterpoint is better according to some criterion and worse according to some other criterion). However, adding new criteria may also reduce the set of non-dominated counterpoints, as new criteria may provide additional information that will differentiate the quality of previously identical counterpoints. In any case, the dominance relation allows for reducing the set of counterpoints and thus obtain a smaller set of interesting ones without the need of introducing strong, arbitrary assumptions (like imposing some order or hierarchy of criteria) and aggregation models (like a weighted sum of criteria values).

3 Computational experiments

The algorithm described above was tested on a dataset consisting of 9 different types of cantus firmus melodies:

- **p** – popular tunes (for example "Twinkle, Twinkle, Little Star", "Pop! Goes the Weasel", or "Happy Birthday").
- **e** – cantus firmi from the exercises taken from the J.J. Fux's textbook (Fux, Mann, and Edmunds, 1965).
- **cf** – melodies taken from an online database of Gregorian chants (Koláček et al., 2009).
- **regs** – cantus firmi consisting of a set of ascending or descending adjacent notes from the given mode:
  - **regsup** – consisting of adjacent ascending notes from the given mode, each a second (major or minor) higher than the previous one, starting from the root note,
  - **regsdown** – consisting of adjacent descending notes from the given mode, each a second lower than the previous one, starting from the root note.
- **regc** – cantus firmi consisting of a set of ascending or descending chromatic notes:
  - **regcup** – consisting of adjacent ascending notes, each a minor second higher than the previous one, starting from the root note,
  - **regcdown** – consisting of adjacent descending notes, each a minor second lower than the previous one, starting from the root note.
- **rand** – randomly generated melodies of two types:
  - **randc** – melodies using randomly selected notes from the chromatic scale,
  - **rands** – melodies using randomly selected notes from the mode for which the cantus firmus was selected.

Note that the counterpoint rules were originally designed to be used with melodies of types **e** and **cf**. However, we still employ them for all the other melody groups in order to check if the characteristics of the results obtained differ between the groups, e.g. if the results for the melodies in the **p** group are different than the results for the **cf** and **e** groups.

Each type listed above formed a group which consisted of 6 cantus firmus melodies except for the **cf** group, which consisted of 12 melodies. Each melody within a given group was in a different mode (as for the **cf** group, each two melodies are in the same mode), except for the **p** group which only consisted of cantus firmi in the Ionian mode.

Most of the original melodies from the **p**, **e**, **cf** groups consisted of 11 notes, and some of the longer melodies were adapted to that length. Since we wanted to study the influence of the length of cantus firmus on the number of counterpoints, truncated versions of the melodies were also considered; a shortened version (to the length of n notes) of a melody consisted of n initial notes of the original melody.

To sum up, the full set processed in computational experiments consisted of 60 cantus firmus melodies in 9 different lengths, which gives a total number of 540 cantus firmus melodies. For these melodies, counterpoints were composed above and below of the given cantus firmus. This gives a total number of 1080 of counterpoint compositional tasks. In this paper, the notion of *counterpoint compositional tasks* will be used to denote a task of composing a counterpoint in a given mode to a given cantus firmus melody of a given length, in the given position (above or below) relatively to the cantus firmus. Note that for a single cantus firmus melody, multiple, differing counterpoints are generated, and each of them is evaluated according to the aforementioned criteria.

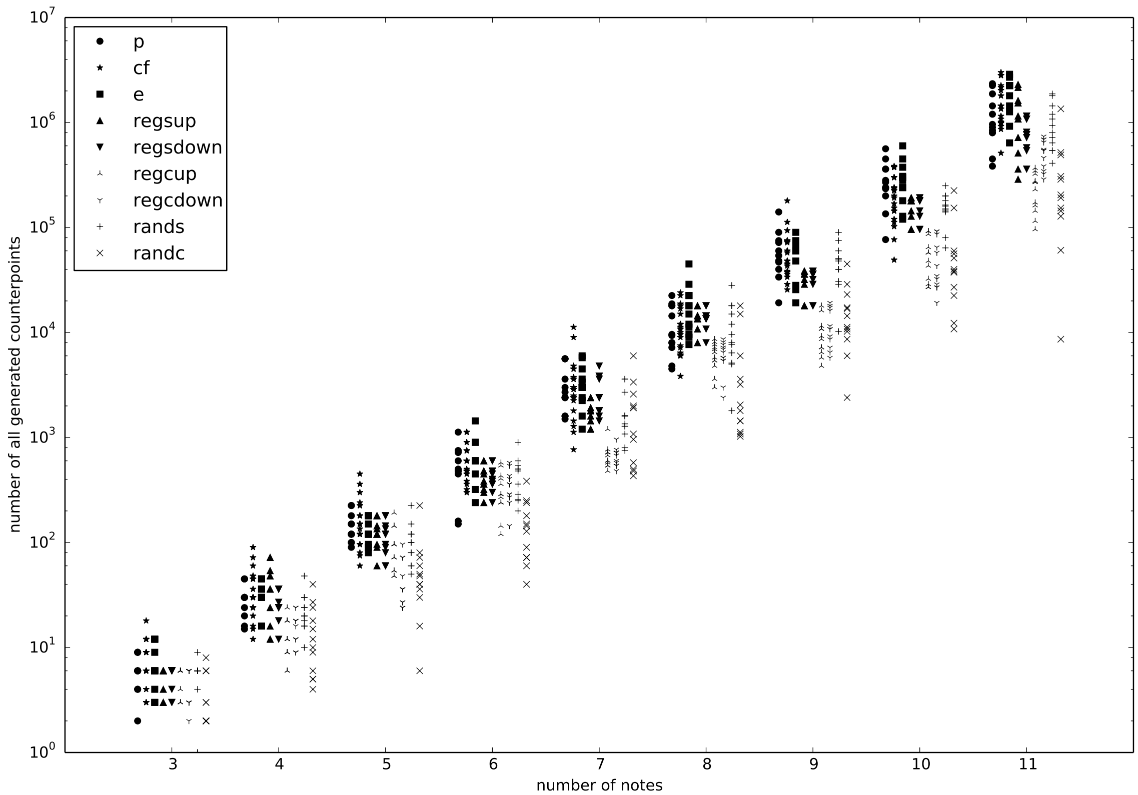

Figure 2: The number of all generated counterpoints for different cantus firmus groups and different lengths.

3.1 The number of generated and non-dominated counterpoints

The total number of counterpoints generated for each cantus firmus is shown in Fig. 2. Each mark in the plot designates the number of generated counterpoints for a given counterpoint compositional task. As illustrated, the number of generated counterpoints grows exponentially as the length of the cantus firmus increases (note that the vertical scale is logarithmic). For melodies only 11 notes long, the number of possible counterpoints is around 100,000–3,000,000.

The plot demonstrates some differences between the groups. First of all, more counterpoints are generated for groups **p**, **cf**, **e**, **regsup/down**, and **rands**, than for groups **regcup**, **regcdown**, and **randc**. The reason for this is that the cantus firmi from the first set consist only of notes taken from the given mode, while the melodies from the second set use the chromatic scale (which uses notes outside of the mode). For each note of the cantus firmus, the generator finds all possible counterpoints that consist of consonances that belong to the given mode. If a particular cantus firmus note is outside of the mode, the unison and the octave are also outside of this mode. What is more, in such a case it is less probable that the perfect fifth consonance (that could also be used) is in the mode. Therefore, the number of counterpoints for the groups that consist of chromatic notes is decreased at the generation stage.

Another interesting observation is that the variance of the **regsup**, **regsdown**, **regcup**, and **regcdown** groups is smaller than the variance of the other groups. This is because all cantus firmi in each of these groups consist of the same notes in a similar order, i.e. the melodies are very similar to each other, which in turn results in smaller variances visible in the plot.

To verify these observations, a statistical procedure was employed: the independent samples

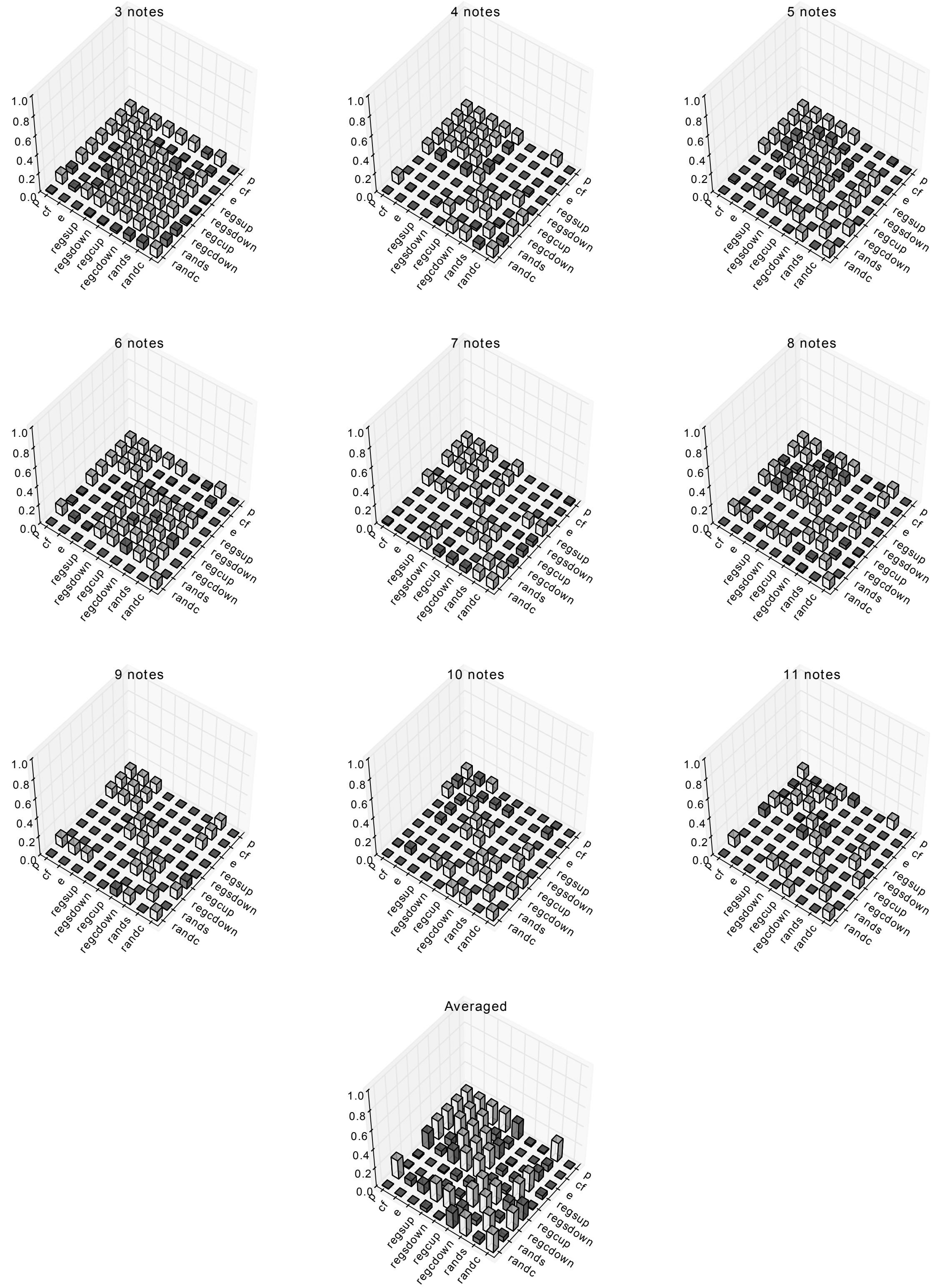

Figure 3: The results of a statistical test (p-values are shown) used to compare if the mean numbers of all generated counterpoints for different groups of cantus firmi are statistically different. To increase readability, a cut-off level is introduced for high p-values: 0.1 for individual charts, 0.2 for the "Averaged" chart. The bars below the cut-off level are darker.

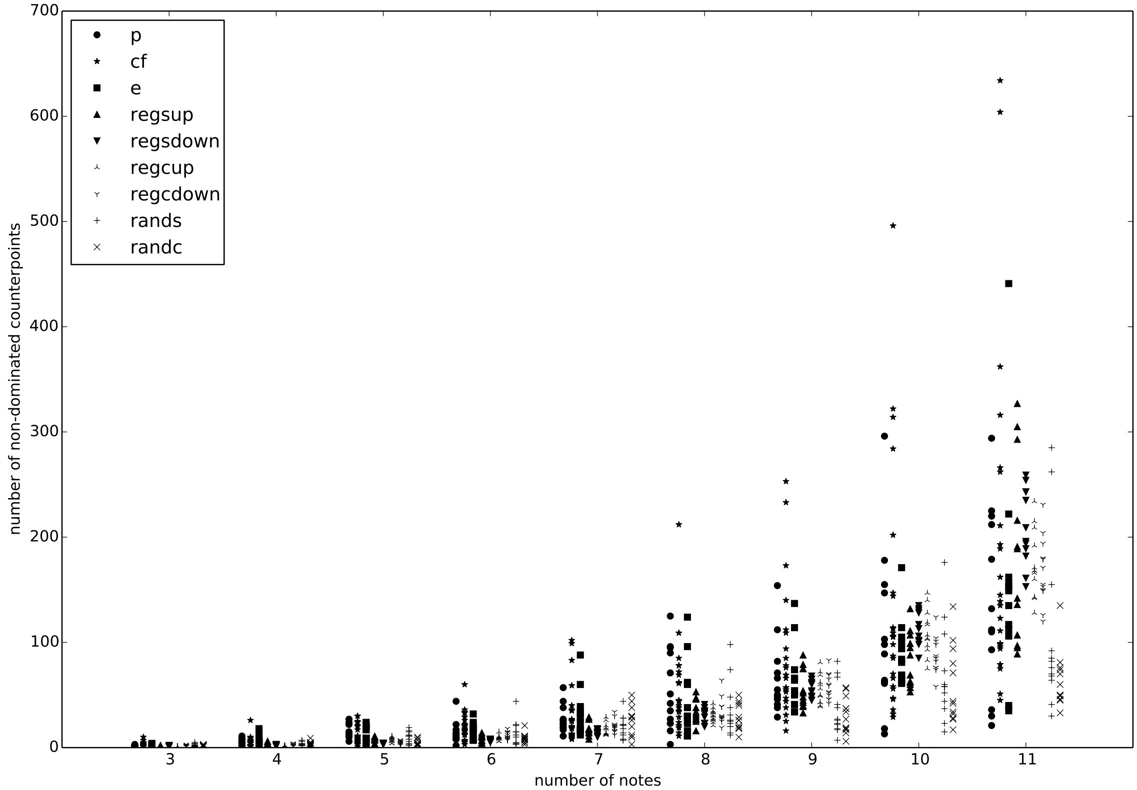

Figure 4: The number of non-dominated counterpoints for different cantus firmus groups and different lengths.

t-test was used to compare the mean number of counterpoints in each pair of groups (Fig. 3). If the samples did not comply with the assumptions of the t-test, i.e. the data were not normally distributed (which was tested using the Shapiro test, $p_{threshold} = 0.05$) or the difference in size of the groups was larger than 150%, the Mann-Whitney U test was used. The Levene's test was employed to check the homogeneity of variances of the groups.

Due to the complex nature of the statistical procedure, the individual p-values shown on the charts do not have to be directly comparable. Similarly, the averaged p-values shown in the bottom plot do not have a strict statistical interpretation, but they demonstrate prevailing patterns present in individual plots for different numbers of notes. Where averaged p-value is nearly zero, it must be low for all lengths of cantus firmi, which means that the difference between the number of generated counterpoints for such a pair of cantus firmi groups was statistically significant. The plots with averaged p-values confirm the observation that the mean number of generated counterpoints for groups using chromatic notes is generally smaller than for the other groups.

Another observation is that the difference of generated counterpoints between the **cf** and the **rands** group is statistically significant. Fig. 2 indicates the direction of this difference: there are more generated counterpoints for group **cf** than for **rands**. It seems that the **cf** group has some characteristics that allows for easier generation of counterpoints; this may be caused by the stepwise nature of the melodies. Discovering the true nature of the **cf** group and describing it formally is an interesting direction of further research.

For each cantus firmus, all pairs of generated counterpoints were compared using the multi-criteria dominance relation described in Sect. 2.3 in order to find subsets of non-dominated counterpoints. This allowed for the reduction of the number of counterpoints as shown in Fig. 4. For short

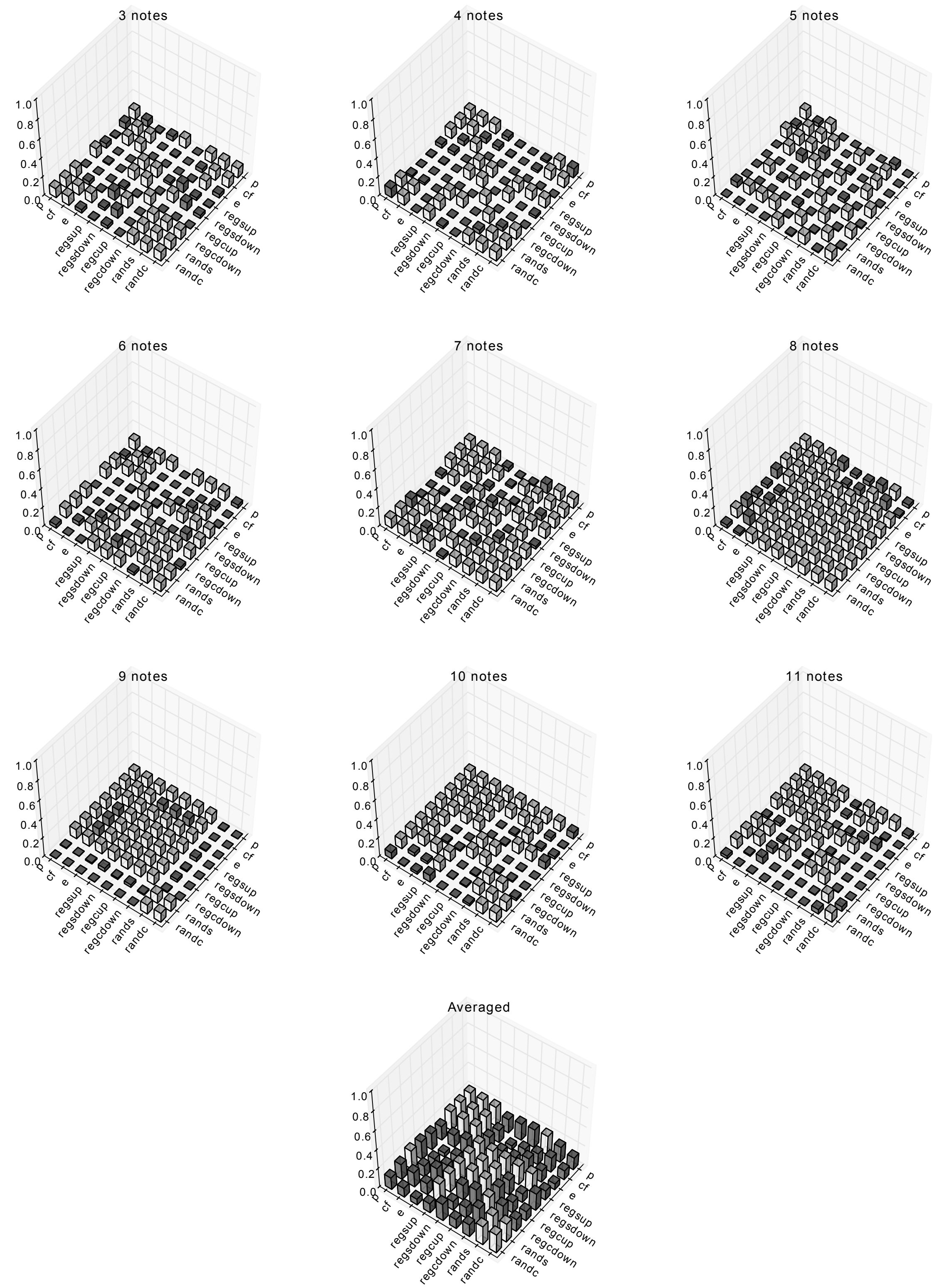

Figure 5: The results of a statistical test (p-values are shown) used to compare if the mean numbers of all non-dominated counterpoints for different groups of cantus firmi are statistically different. To increase readability, a cut-off level is introduced for high p-values: 0.1 for individual charts, 0.2 for the "Averaged" chart. The bars below the cut-off level are darker.

cantus firmi, the set of non-dominated counterpoints is small enough to be analyzed by a human expert, but for longer cantus firmi, the dominance relation used with the considered criteria may be too weak to significantly reduce the set, as the trend is exponential. While it may be tempting to try to reduce the set of counterpoints and find the single "best" counterpoint – or a small number of them – it is natural that longer melodies allow for a larger number of non-dominated counterpoints; diverse flavours of counterpoints are found in the non-dominated set, and the aggregation of criteria can only be used if one needs to introduce a strict ordering among the counterpoints. However, one has to be aware that the outcome of this ordering (i.e., the "best" counterpoint) will depend on the details and the (arbitrary) parameters of the procedure that aggregates the criteria.

For longer melodies, generating all counterpoints and looking for non-dominated ones will not be practical; in such cases multiple-criteria evaluation may be integrated with a heuristic or a meta-heuristic approach. The optimization perspective includes a possibility to guide the search for interesting counterpoints using either a single selected parameter as their characteristics, or using multiple criteria simultaneously. The efficiency of heuristic search algorithms like multiple random start local search, tabu search, simulated annealing, particle swarm, or evolutionary techniques (Talbi, 2009; Gendreau and Potvin, 2010; Coello, Lamont, and van Veldhuizen, 2007) will be increased with the appropriate definition of neighbourhood and a smooth fitness landscape, so fitness-distance analyses will be highly recommended (Merz and Freisleben, 1999; Hoos and Stützle, 2005; Vanneschi et al., 2003).

The differences in the number of non-dominated counterpoints were also compared using the previously described statistical procedure. As shown in Fig. 5, the statistical significance of these differences is lower. However, one can still see that the human-composed cantus firmi from the **e** group tend to have more non-dominated counterpoints than the randomly generated cantus firmi from group **randc**.

We also investigated whether the number of possible counterpoints depends on the position of the cantus firmus in the counterpoint compositional task – i.e., whether the counterpoint is built above or below the cantus firmus. It turned out that for groups **cf** and **e**, there were slightly more counterpoints generated above than below the cantus firmus. These minor differences are even less pronounced when comparing the number of non-dominated counterpoints for both groups. A similar analysis was performed to examine whether different church modes influence the number of generated or non-dominated counterpoints. No clear differences were noticed – in this regard, the difficulty of composing a counterpoint does not seem to depend on the mode that is used.

3.2 Imposing constraints on the most important properties of counterpoints

Not all of the criteria in our set are of equal importance. Most of them, like DirectMotion or ImperfectConsonances, are only suggestions stating that counterpoints with less direct motion or more imperfect consonances are preferable. However, two rules in the set – PerfectConsonancesByDirectMotion and ForbiddenJumps – are more important than the others and should never be broken. In order to improve the quality of non-dominated counterpoints, we filtered out the counterpoints that break these two rules before employing the dominance relation.

Three situations were considered:

- **7 criteria** – no filtering was done, thus all the criteria are used to determine the set of non-dominated counterpoints;
- **6 criteria** – all counterpoints that break the ForbiddenJumps rule are filtered out;
- **5 criteria** – all counterpoints that break the ForbiddenJumps rule or the PerfectConsonancesByDirectMotion rule are filtered out.

Fig. 6 shows the number of non-dominated counterpoints in these three situations. When no filtering is done, the number of non-dominated counterpoints is the highest, and imposing constraints

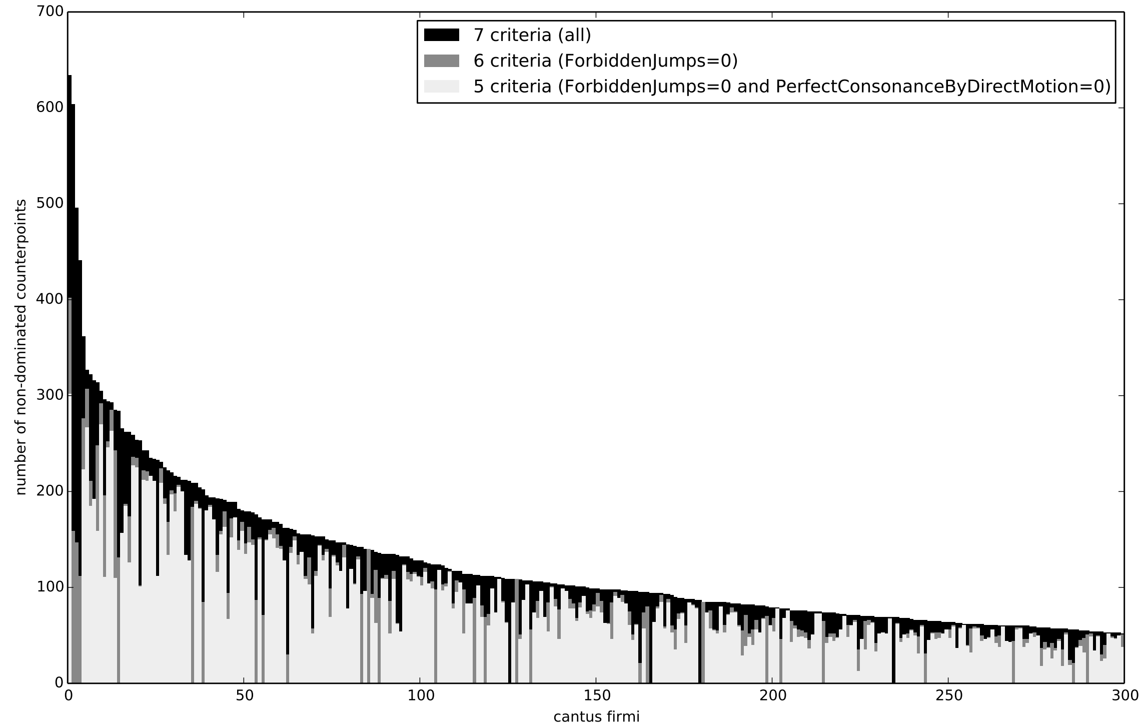

Figure 6: The number of non-dominated counterpoints after filtering out the counterpoints that break the ForbiddenJumps rule or both ForbiddenJumps and PerfectConsonancesByDirectMotion rules. The values are shown for 300 (out of 1080) cantus firmi that generated the highest number of non-dominated counterpoints when no filtering was done. Each column contains three bars standing one in front of another: the black bar at the back shows the number of counterpoints when no filtering is done, the grey bar in the middle shows the number of counterpoints when the ones that break the ForbiddenJumps rule are filtered out, and the light grey bar in front shows the number of counterpoints when counterpoints that break at least one rule are filtered out. Since the bars overlap vertically, the bar lengths that are visible correspond to the loss of counterpoints when a constraint is introduced.

reduces this set. For most cantus firmi, the filtering decreases the number of non-dominated counterpoints only slightly. However, there exist some melodies for which filtering out the counterpoints that break one or both criteria decreases the number of non-dominated counterpoints to zero. Note that if the number of non-dominated counterpoints is zero, it must mean that all counterpoints were filtered out by the one of the filters. In other words, for some cantus firmi it is impossible to compose a counterpoint that would obey the two most important rules.

As an example, let us consider the situation shown in Fig. 7. The cantus firmus used was a Dorian mode melody from the **e** group. The compositional task in this example was to build a counterpoint above this melody using the Dorian mode and notes D4-D5. The part in the violin key shows one of the non-dominated counterpoints found using our algorithm. Without filtering, the number of non-dominated counterpoints for this cantus firmus was equal to 153. When the ForbiddenJumps filter was used, the number of counterpoints dropped to 117, and after using both the ForbiddenJumps and the PerfectConsonancesByDirectMotion filters, the number of non-dominated counterpoints decreased to 112. As can be seen in Fig. 7, the characteristics of changes in the number of non-dominated counterpoints is generally similar to the example described above: for most of the cantus firmi, using filtering only slightly decreases the number of non-dominated

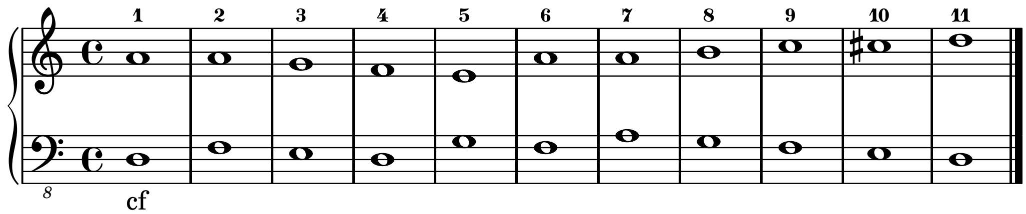

Figure 7: One of the 153 non-dominated counterpoints composed to the Dorian mode cantus firmus from the **e** group. Note that this solution would be forbidden by Fux because of the chromatic half step, C to C#. However, as our algorithm did not include this rule, the rule was not satisfied in this counterpoint.

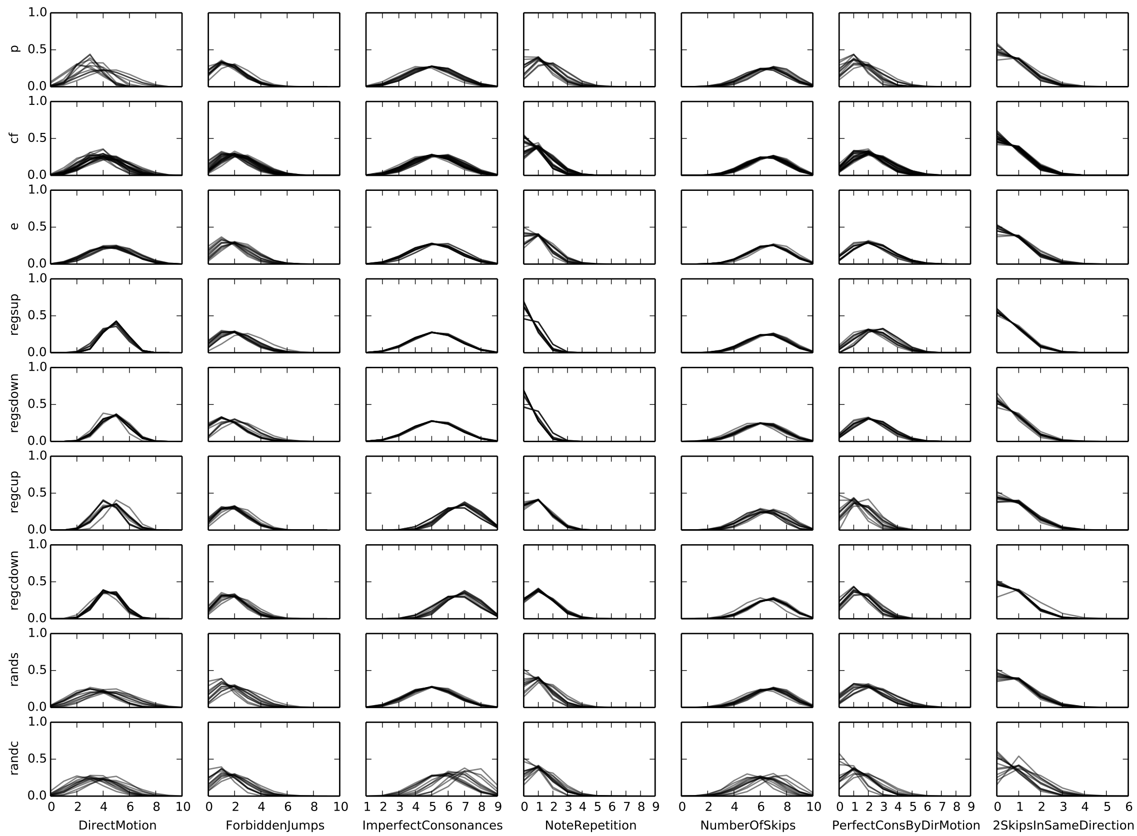

Figure 8: Distributions of criteria values for different types of cantus firmi that are 11 notes long.

counterpoints.

This experiment demonstrates that filtering out the counterpoints that break the Forbidden-Jumps and the PerfectConsonancesByDirectMotion rules may be used as a way to decrease the number of non-dominated counterpoints. However, in rare cases, such filtering can remove all the counterpoints.

3.3 Distribution of criteria values

Fig. 8 presents distributions of criteria values for different groups of cantus firmi. As one can see, shapes of these distributions vary between different criteria and between different groups. The minimized criteria with mostly small values in their distributions may be considered "easy", because most counterpoints do not break such criteria too often – examples are ForbiddenJumps, NoteRepetition, PerfectConsonancesByDirectMotion, and TwoSkipsInTheSameDirection. On the other hand, if a distribution contains large values and very few (or none) of the counterpoints have low values – as in the case of DirectMotion and NumberOfSkips – it means that it is difficult to minimize such a criterion. The ImperfectConsonances is the only maximized criterion in the considered set, so its distributions need to be interpreted differently – higher values are preferred. Analogously, this criterion may be considered "easy", as it tends to have larger values. However, large values in the DirectMotion and the NumberOfSkips criteria distributions should not be considered as the main difficulties for counterpoint composition: their importance relative to the other criteria is lower. These two rules can be broken occasionally and there are other much more important criteria in our criteria set, such as ForbiddenJumps and the PerfectConsonacesByDirectMotion.

An interesting aspect of this analysis is revealing differences between distributions for different melody groups. For example, the distributions of the ImperfectConsonances criterion tend to have its mean much higher (and its variance smaller) for the **regcup**, **regcdown**, and **randc** cantus firmus groups. This is caused by the reduced number of consonances available for these groups (the reasons for this were described in Sect. 3.1). The reduced number of consonances in these groups is also visible for the PerfectConsonancesByDirectMotion criterion.

Another clearly visible difference is the smaller variance in the DirectMotion distributions for groups **regsup**, **regsdown**, **regcup**, and **regcdown**. The smaller variance for these groups means that it is more difficult to create a counterpoint with the same (or the opposite) movements of melodies for the cantus firmi from the aforementioned groups. This results in part from the fact that if the melody of the cantus firmus goes continuously in the same direction, the melody of the counterpoint has to preserve this direction, but at the same time all the intervals between the counterpoint and the cantus firmus need to be consonances. What is more, the range of notes in the counterpoint is limited to one octave. These constraints force the counterpoint to change the direction of its movement, thus preventing it from having the same direction of movement (e.g., always up) as the cantus firmus.

In some plots, two distinct patterns of the counterpoints are visible – in particular, this is demonstrated by the **regsup** group of cantus firmi. Let us consider the NoteRepetition criterion for this group and **regsup** cantus firmus in Aeolian church mode built from the following notes: A2, B2, C3, D3, E3, F3, G3, A3, B3, C3, D4. If we want to compose a counterpoint above the cantus firmus, then the next-to-last note should be a major sixth (A) and the last note a perfect consonance (A or D). An alternative situation occurs in the Lydian church mode – in that case, the next-to-last note needs to be F#. This note is raised, which means that it will never be equal to the last counterpoint note for this cantus firmus. The two distinct patterns correspond to the two types of cantus firmi: those for which the next-to-last note needs to be raised and those for which it should not be raised.

The characteristics of the distributions for cantus firmi shorter than 11 notes are similar to the ones described above.

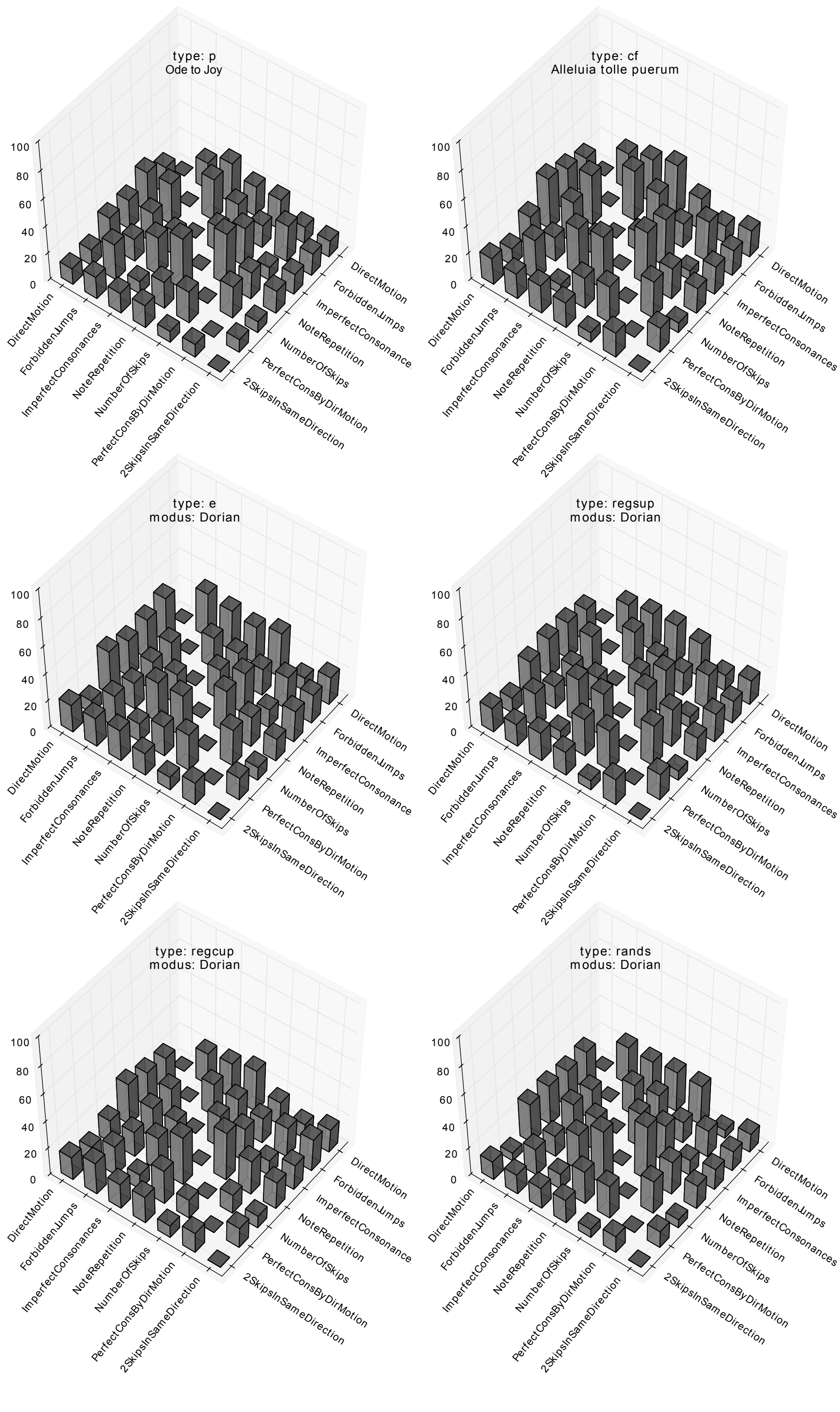

Figure 9: Discordance of pairs of criteria (vertical axis in %) for selected counterpoints from different groups.

3.4 Discordance of criteria used to evaluate counterpoints

Given a set of counterpoints each one evaluated according to multiple criteria, it is possible to estimate how often each pair of these criteria disagrees, i.e., indicates opposite preference for a pair of counterpoints (one criterion indicates that counterpoint c_1 is better than c_2, while the other criterion indicates that c_2 is better than c_1). To estimate discordance of each pair of criteria for a given counterpoint, all $n(n-1)/2$ pairs of n counterpoints are compared and opposite preferences are counted. Note that the situation where one criterion does not distinguish between two counterpoints and the other criterion prefers one counterpoint over the other is not considered a conflict.

Results shown in Fig. 9 demonstrate that discordance is rather high which explains why the number of non-dominated counterpoints is also high. One of the reasons for this is the fact that the criteria are multi-valued, which provides more occasions for conflicts that increase the number of non-dominated counterpoints. High discordance confirms that all the criteria are useful and each of them provides unique information about the quality of counterpoints; no pair of criteria is in total agreement.

The pairs of criteria for which the discordance is the lowest are NumberOfSkips and TwoSkipsInTheSameDirection, ImperfectConsonances and PerfectConsonancesByDirectMotion, and DirectMotion and PerfectConsonancesByDirectMotion. The first pair is concordant because if a counterpoint has a low number of skips, it also has a low number of skips in the same direction, and the direction of preference for these two criteria is the same. On the other hand, if a counterpoint has a low number of consonances or the direct motion is used very rarely, then the PerfectConsonancesByDirectMotion criterion has a low value, which explains the concordance of the latter two pairs of criteria.

The remaining pairs of criteria have higher discordances – an example of particularly high disagreement is NoteRepetition and NumberOfSkips. In this case, such a high discordance results from situations when there are no consonances within a step away from the previous note, and the only solutions for selecting the next counterpoint note is either by repeating the previous note or by jumping to a distant note. Each of these two solutions is preferred by a different criterion, hence their high discordance.

3.5 Examples of non-dominated counterpoints

Fig. 10 presents two non-dominated counterpoints composed by the algorithm to the same cantus firmus melody taken from the J. J. Fux's textbook (Fux, Mann, and Edmunds, 1965) and one example counterpoint presented in the book. The counterpoint rules described earlier are fulfilled: these counterpoints consist only of consonances, the first and the last intervals are perfect consonances, the next-to-last interval is a major sixth, and mainly steps are used. However, these counterpoints are not perfect considering all the criteria: the main problem with the first counterpoint is the direct motion between notes 2 and 4 and the large perfect fifth skip between notes 7 and 8. The second counterpoint uses the direct motion even more often (three times – between notes 2 and 3, 4 and 5, and 5 and 6); on the other hand, it does not use any skips. The third counterpoint has the same values as the first counterpoint for all criteria except for the DirectMotion. This means that the third counterpoint was dominated by the first one. Table 1 summarizes criteria values for all three counterpoints.

An important aspect of the first two counterpoints is that according to the multi-criteria analysis approach presented in this work, they are incomparable, because each of them has its advantages and disadvantages relative to the other. The second counterpoint uses direct motion and repeated notes more often, which makes the first counterpoint better according to the DirectMotion criterion. On the other hand, the first counterpoint contains a big interval jump between notes 7 and 8, which makes the second counterpoint better regarding the criterion of jumps, as it does not contain any jumps.

If the criteria values were aggregated (for example by employing the popular weighted sum formula), one of these counterpoints could be lost, because the value of its quality function would

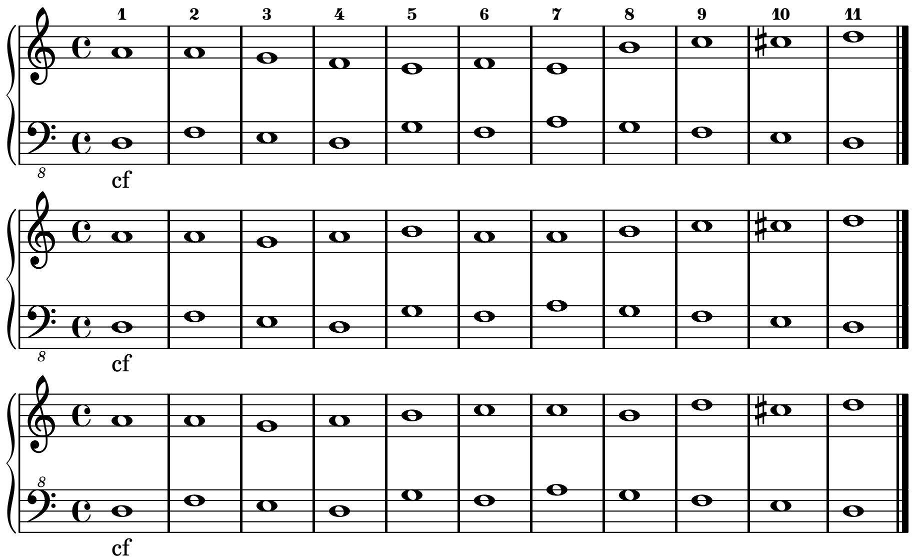

Figure 10: Three sample counterpoints composed above a cantus firmus in the Dorian mode. The first and the second counterpoints are non-dominated and were composed by our algorithm, while the third one comes from the *Steps to Parnassus* book (Fux, Mann, and Edmunds, 1965).

be inferior. However, it is not obvious which of the two counterpoints is better and how should the criteria be aggregated, therefore avoiding aggregations and leaving the final decision to a human is a more justified approach. In the multiple-criteria decision analysis (MCDA) perspective, decisions made by the decision maker regarding their preference of one counterpoint over the other would provide important clues (the *preferential information*) on the importance of criteria to this particular decision maker, and on their own subjective feeling as to which counterpoints are more interesting than the others. This information could be automatically extracted from sample choices made by a decision maker, and then employed to sort the full set of non-dominated counterpoints in agreement with preferences learned from those expressed by a human. Such an approach would be far more appropriate than using a weighted sum with arbitrarily adjusted weights. MCDA is therefore an interesting and promising paradigm to be used in algorithmic composition, and its area of application is not limited to counterpoint composition.

DirectMotion	ForbiddenJumps	ImperfectConsonances	NoteRepetition	NumberOfSkips	PerfectConsByDirMotion	2SkipsInSameDirection
2	0	7	2	1	0	0
3	0	5	2	0	0	0
4	0	7	2	1	0	0

Table 1: Evaluation criteria: the number of times a specific situation occurs in counterpoints from Fig. 10.

4 Conclusions and further work

As demonstrated by the experiments reported in this article, the relation of multi-criteria dominance provides valuable information and a useful perspective in evaluation of counterpoints. It is a general tool that can be employed in the algorithmic composition research, allowing to model the composition process without making any assumptions about the relative importance of each evaluation criterion. This work presents results of an algorithm that is capable of composing the first species counterpoint to the given cantus firmus using the dominance relation; the outcome of running the algorithm is a set of best, but mutually incomparable, counterpoints for the particular cantus firmus.

The numerical results obtained from the computational experiments allowed to quantitatively examine the "solution space" of all possible counterpoints. Comparing counterpoints generated for different types of cantus firmi revealed differences between the groups. The number of generated counterpoints for cantus firmi which used chromatic notes was smaller than for the other groups: if the notes that are outside of the mode are used in the cantus firmus, it is more difficult to find consonances to use in counterpoints. This reduces the number of possible counterpoints, because only consonances are allowed in the first species counterpoint. A smaller variance of the number of counterpoints was noticed in groups **regsup**, **regsdown**, **regcup**, and **regcdown**, which comes from the similarity of cantus firmi in these groups.

Another aspect that has been analysed are the relationships between the evaluation criteria; the discordance between some criteria (e.g. NumberOfSkips and TwoSkipsInTheSameDirection) turned out to be low, while for other criteria (e.g., NoteRepetition and NumberOfSkips) it is high. The overall high discordance of criteria corresponds with a large number of non-dominated counterpoints. For some cases this may be a challenge, because for some 11-note cantus firmi, the number of non-dominated counterpoints reaches almost 700. This number makes it difficult for the human expert to analyze all these interesting counterpoints – thus methods for further reduction may be employed, some of which may be based on obtaining subjective, preferential information from a human. For most cases however, the richness of the non-dominated set is an advantage as it preserves diversity, and no counterpoint is lost that is best on at least one criterion.

One of the simplest techniques that can be used to reduce the set of non-dominated counterpoints is filtering out the counterpoints which break the most important rules of counterpoint composition. This technique gave diversified results for different counterpoints – it turned out that for some cantus firmi it was impossible to compose a counterpoint that would obey the two most important rules.

Further work will focus on analyses of other counterpoint species, and on modifying the current approach so that it is capable of composing counterpoints for longer melodies. Another interesting area of research is an automated discovery of rules that were typically used by specific composers; this can be done by analyses of counterpoints they composed. Results from such analyses may have impact on the area of automatic composition, and they may also provide new knowledge in musicology.

Acknowledgement

This work has been supported by the Polish National Science Centre, grant no. N N519 441939.

References

Acevedo, Andres Garay. 2005. "Fugue composition with counterpoint melody generation using genetic algorithms." In *Computer Music Modeling and Retrieval,* 96–106. Esbjerg, Denmark: Springer.

Aguilera, Gabriel, José Luis Galán, Rafael Madrid, Antonio Manuel Martínez, Yolanda Padilla, and Pedro Rodríguez. 2010. "Automated generation of contrapuntal musical compositions using probabilistic logic in derive." *Mathematics and Computers in Simulation* 80 (6): 1200–1211.

Boenn, Georg, Martin Brain, Marina De Vos, and John Ffitch. 2011. "Automatic music composition using answer set programming." *Theory and Practice of Logic Programming* 11 (2–3): 397–427.

Byrd, Donald, and Tim Crawford. 2002. "Problems of music information retrieval in the real world." *Information processing & management* 38 (2): 249–272.

Camilleri, Lelio. 1993. "Computational Musicology. A Survey on Methodologies and Applications." *Revue Informatique et Statistique dans les Sciences humaines* 29 (1–4): 51–65.

Coello, Coello, Gary B. Lamont, and David A. van Veldhuizen. 2007. *Evolutionary Algorithms for Solving Multi-Objective Problems*. Genetic and Evolutionary Computation. Springer.

Conklin, Darrell, and Mathieu Bergeron. 2010. "Discovery of contrapuntal patterns." In *11th International Society for Music Information Retrieval Conference,* 201–206. Utrecht.

Cope, David. 2004. "A musical learning algorithm." *Computer Music Journal* 28 (3): 12–27.

Coutinho, Eduardo, Marcelo Gimenes, Joao M. Martins, and Eduardo R. Miranda. 2005. "Computational musicology: An artificial life approach." In *Portuguese Conference on Progress in Artificial Intelligence, EPIA 2005,* 85–93. IEEE.

de Haas, W. Bas, José Pedro Magalhaes, Frans Wiering, and Remco C. Veltkamp. 2011. *HarmTrace: Automatic functional harmonic analysis*. Tech. rep.. Technical Report UU-CS-2011-023, Department of Information and Computing Sciences, Utrecht University.

Doumpos, Michael, and Evangelo Grigoroudis. 2013. *Multicriteria Decision Aid and Artificial Intelligence: Links, Theory and Applications*. Chichester: Wiley.

Ebcioğlu, Kemal. 1990. "An expert system for harmonizing chorales in the style of J.S. Bach." *The Journal of Logic Programming* 8 (1): 145–185.

Ehrgott, Matthias. 2006. *Multicriteria Optimization*. Berlin: Springer.

Farbood, Morwaread Mary, and Bernd Schoner. 2001. "Analysis and synthesis of Palestrina-style counterpoint using Markov chains." In *2001 International Computer Music Conference,* Havana. September.

Fux, Johann Joseph, Alfred Mann, and John Edmunds. 1965. *The Study of Counterpoint: From Johann Joseph Fux's "Gradus Ad Parnassum"*. Vol. N277, The Norton library. New York: W. W. Norton & Company Incorporated.

Gendreau, Michel, and Jean-Yves Potvin. 2010. *Handbook of Metaheuristics*. International Series in Operations Research & Management Science. 2nd ed. Springer.

Hapke, Maciej, and Maciej Komosinski. 2008. "Evolutionary Design of Interpretable Fuzzy Controllers." *Foundations of Computing and Decision Sciences* 33 (4): 351–367.

Hoos, Holger H., and Thomas Stützle. 2005. *Stochastic Local Search: Foundations and Applications*. Amsterdam: Morgan Kaufmann.

Huang, Cheng Zhi Anna, and Elaine Chew. 2005. "Palestrina Pal: a grammar checker for music compositions in the style of Palestrina." In *Proceedings of the 5th Conference on Understanding and Creating Music,* not paged. Caserta, Italy: Seconda Universita degli Studi di Napoli.

Jelonek, Jacek, and Maciej Komosinski. 1999. "Genetic algorithms in constructive induction." In *Foundations of Intelligent Systems,* Vol. 1609 of *Lecture Notes in Computer Science* edited by Zbigniew Ras and Andrzej Skowron. 665–673. Berlin/Heidelberg: Springer. `http://dx.doi.org/10.1007/BFb0095156`.

Junod, Julien, and Guerino Mazzola. 2007. "From Fux To Ragas – Morphing Contrapuntal Worlds." In *2007 International Computer Music Conference,* Copenhagen, Denmark. August.

Koláček, Jan, et al. 2009. "Database of Melodies and Texts of Gregorian Chant." `http://www.globalchant.org/`.

Komosinski, Maciej, and Krzysztof Krawiec. 2000. "Evolutionary weighting of image features for diagnosing of CNS tumors." *Artificial Intelligence In Medicine, Special Issue on Evolutionary Computation in Medicine* 19 (1): 25–38. `http://linkinghub.elsevier.com/retrieve/pii/S0933365799000482`.

Komosinski, Maciej, Adam Kups, Dorota Leszczyńska-Jasion, and Mariusz Urbański. 2014. "Identifying efficient abductive hypotheses using multi-criteria dominance relation." *ACM Transactions on Computational Logic* 15 (4). `http://doi.acm.org/10.1145/2629669`.

Maurer, John. 1999. "A brief history of algorithmic composition." Published online: `https://ccrma.stanford.edu/~blackrse/algorithm.html`.

Mazzola, Guerino, Stefan Göller, Stefan Müller, Carlos Agon, Moreno Andreatta, Gerard Assayag, Jan Beran, et al. 2002. *The topos of music.* Birkhäuser Basel.

Merz, Peter, and Bernd Freisleben. 1999. *Fitness Landscapes and Memetic Algorithm Design.* 245–260. Amsterdam: McGraw-Hill.

Papadopoulos, George, and Geraint Wiggins. 1999. "AI methods for algorithmic composition: A survey, a critical view and future prospects." In *AISB Symposium on Musical Creativity,* 110–117. Edinburgh: SSAISB. April.

Roy, Bernard, and Philippe Vincke. 1981. "Multicriteria analysis: survey and new directions." *European Journal of Operational Research* 8 (3): 207–218.

Schottstaedt, Bill. 1984. *Automatic species counterpoint.* Tech. Rep. 19. Center for Computer Research in Music and Acoustics, Stanford University.

Steedman, Mark J. 1984. "A generative grammar for jazz chord sequences." *Music Perception* 2 (1): 52–77.

Talbi, El-Ghazali. 2009. *Metaheuristics: From Design to Implementation.* Wiley Series on Parallel and Distributed Computing. Wiley.

Tzanetakis, George, and Perry Cook. 2002. "Musical genre classification of audio signals." *IEEE transactions on Speech and Audio Processing* 10 (5): 293–302.

Utgoff, Paul E., and Phillip B. Kirlin. 2006. "Detecting motives and recurring patterns in polyphonic music." In *Proceedings of International Computer Music Conference ICMC,* Vol. 6487–494. Ann Arbor: MI: MPublishing, University of Michigan Library.

Vanneschi, Leonardo, Marco Tomassini, Philippe Collard, and Manuel Clergue. 2003. "Fitness distance correlation in structural mutation genetic programming." In *Proceedings of the 6th European Conference on Genetic Programming,* Essex, UK. EuroGP'03. 455–464. Berlin, Heidelberg: Springer-Verlag.

Volk, Anja, Frans Wiering, and Peter van Kranenburg. 2011. "Unfolding the potential of computational musicology." In *Proceedings of the 13th International Conference on Informatics and Semiotics in Organisations (ICISO),* 137–144. Leeuwarden: Fryske Akademy.

Weale, Timothy, and Jennifer Seitzer. 2003. "EVOC: A Music Generating System using Genetic Algorithms." In *International Joint Conference on Artificial Intelligence,* 1383–1384.

Yilmaz, Asim Egemen, and Ziya Telatar. 2010. "Note-against-note two-voice counterpoint by means of fuzzy logic." *Knowledge-Based Systems* 23 (3): 256–266.