Bachelor Kunstmatige Intelligentie, UU

Search on a multi-dimensional interval lattice as a method for the automatic intonation of arbitrary pitch configurations

T.M. van Breugel
4148886

July 3, 2019

7.5 ECTS
Supervisor: Dr. M. van Ommen
Second reader: Dr. F. Adriaans

Universiteit Utrecht

In a world where the global cultural impact of music is evident, its study naturally attracts interest. Western music is primarily tuned in accordance with the twelve-tone equal temperament system, 12TET. A system which allows for consistent and equal intonation across keys, but systematically deviates from the theoretically ideal just intonation. An automatic and interactive intonation system can be implemented which aims to provide the most consonant intonation for any given pitch configuration using the principle of just intonation. To determine the most consonant intonations a measure called the harmonic distance is used. The output of the ConsAInance system is evaluated and compared to 12TET tuning, and a brief overview of related work is given.

Contents

1 Introduction

What does it mean for an instrument to be

in tune? The sound of a significantly out-of-tune instrument is unmistakable. A feeling of dissonance creeps in when a tuning deviates from established norms. Throughout history multiple tuning systems have been proposed. In western music the most prominent tuning system is twelve-tone equal temperament, 12-TET. This system divides the octave into twelve steps of equal size. This has the advantage of having every pitch being equally spaced apart. Essentially, this means that music can be performed in any key without re-tuning, but at the same time, each interval is slightly out of tune. Just intonation works with pure integer ratios to divide the octave. This has the advantage of being the most pure but the disadvantage of being key-dependent. This unfortunate property will be touched on in further sections. It is the reason we must employ search to properly utilize just intonation for arbitrary musical input.

In artificial intelligence we are interested in what it means to be intelligent and how to model this perceived intelligence. Implementing a system that optimizes the consonance of arbitrary pitch configurations brings us closer to understanding the concept of music as a whole. This system enables further exploration into and experimentation with music. It can be a stepping stone to automatic and tune-able intonation systems for multi-instrument synth bands and deeper insight into music itself. Further understanding a very human subject with the aid of artificial intelligence techniques like search algorithms seems to be a worthwhile effort.

In this research a custom search algorithm will be designed and implemented to power an automatic and interactive intonation system, ConsAInance. This system aims to provide a tuning based on just intonation for any set of inputs presented. The performance of this system can be analyzed by evaluating the consonance of the intonation it posits. A comparison can be made to standard 12TET intonation, answering whether an adapted shortest path algorithm is suitable for automatic and interactive intonation of arbitrary pitch configurations.

1.1 Terminology

Before continuing it is productive to provide an overview of the terms that will be used in this work. A pitch can be thought of as the perceived frequency of a note. It is the entity that determines the tuning of a note. The pitch configuration is a record keeping entity which accounts for all currently sounding pitches. A pitch configuration is more involved than simply keeping track of individual pitches or chords. Any event that adds or removes currently sounding pitches go through this. These events are MIDI events that are sent to the pitch configuration through a MIDI controller provided by ConsAInance. Intervals are the ratios that describe a partitioning point in the octave. Examples of this are the perfect fifth with a ratio of 3/2 and the major third with a ratio of 5/4. When comparing two intervals it is easier to compare the value of the intervals in cents. In 12TET, the distance between each semitone is 100 cents, meaning the whole octave is divided into 1200 cents. This is a useful measure to directly compare 12TET tunings with ConsAInance tunings.

Prime limits are limits as to the magnitude of the prime numbers used for intervals within a tuning. For example, in 5-limit tuning the largest prime used for base interval ratios is 5. In such a tuning, the aforementioned perfect fifth and major third are used as well as ratios formed by the exponentiation of these base intervals. In 7-limit tuning, the harmonic seventh will be added. This interval has a ratio of 7/4. This pattern continues for higher limits.

1.2 The Tonnetz

For automatic and interactive intonation of arbitrary pitch configurations a search algorithm will be defined. This algorithm will operate on a multi-dimensional Tonnetz-like lattice as defined below. The Tonnetz is a concept first defined by Leonhard Euler, defining the relations between pitches in a scale. Any point in this lattice defines an interval. By assigning points on this lattice to pitches, a list of exponents is assigned to this pitch, allowing it to evaluate to an interval, effectively intonating it.

In 5-limit tuning, the largest prime factor used for interval ratios is five. When constructing the harmonic space for such a tuning we span a lattice. A lattice for 5-limit tuning will have two dimensions, one for both primes used by the tuning, three and five. The prime number 2 is left implicit. In musical context it is a doubling which is equivalent to an octave. It is explicitly utilized when determining the intervals for points on the lattice. The points on the lattice are utilized as exponents to primes to calculate intervals. For example, given a point P where $P = (1, 0)$ we can evaluate P to be equal to the interval 3/2. Since each axis in the lattice has its associated prime, we can exponentiate this prime by the value on the respective axis. An example lattice for 5-limit tuning can be found in figure 1. Figure 2 is related to figure 1 and visualizes pitch classes in the Tonnetz, the concept of pitch classes is explained in section 4.1. Any interval can be multiplied by another interval to create a compound interval. The lattice will contain all possible pitches and is thus the space containing all possible intervals for a given prime limit. The musical intervals relating to the primes are defined in table 1.

For this work octave equivalence is assumed. This entails that for any pitch that enters the pitch configuration will inherit the tuning of any pitch with the same pitch class that is already sounding. Musically, any note that is struck will be tuned the same across all octaves.

The ratios in the Tonnetz are generated by the exponentiation of the ratios associated to primes with prime exponents. The resulting ratios are then multiplied by either 2/1 or 1/2 to make them fit between the unison and the doubling. In ConsAInance, a pitch is defined by its prime exponents. These exponents can thus be used to determine their intervals as exemplified in table 2.

Several approaches can be taken to mathematically define the consonance

Prime	Musical interval
3	3/2
5	5/4
7	7/4
11	11/8

Table 1: Musical intervals related to prime numbers

Prime	Musical interval	Exponent	Result
3	3/2	-2	16/9
5	5/4	3	125/64
7	7/4	-1	8/7
11	11/8	0	1/1

Table 2: Musical intervals and their results after exponentiation and being made to fit within the octave

of a system[Ryan, 2016]. The exact consonance measure used will be defined further along in this work. The alternatives provide the same behaviour for our search algorithm, and will thus not be looked at in detail.

Starting off with a brief overview of related work, this work continues on to provide insight into the automatic intonation system, ConsAInance. This is followed by an introduction to automatic intonation and the search algorithm used to achieve it. Lastly, the results will be presented followed by a brief discussion and directions for future work.

2 Related work

In "adaptive tuning utilizing gradient descent to maximize the consonance of pitches" Sethares [Sethares, 1994] utilizes gradient descent on pitch frequencies to maximize consonance. In this work we utilize discrete intervals and a search algorithm with fixed step sizes to try and achieve the same.

In his work pertaining to his Micromælodeon tuning model, Sabat [Sabat, 2008]

36/25	9/5	9/8	45/32	225/128
48/25	6/5	3/2	15/8	75/64
32/25	8/5	1/1	5/4	25/16
128/75	16/15	4/3	5/3	25/24
256/225	64/45	16/9	10/9	25/18

Figure 1: The unison and its direct neighborhood in a Tonnetz-like 5-limit lattice

6	10	2	6	10
11	3	7	11	3
4	8	0	4	8
9	1	5	9	1
2	6	10	2	6

Figure 2: Pitch classes corresponding to the intervals in figure 1

references Tenney and the harmonic distance measure, though a different interpretation of it is used in this work. It operates in the continuous frequency domain and needs "appropriately chosen reference pitches and parameters". ConsAInance produces its own reference pitches, as demonstrated in the subsection on consonance evaluation.

Denckla [Denckla, 1995] presents a theory of tuning, along with the dynamic intonation software Helm. Helm utilizes a score follower program to tune annotated scores through MIDI. ConsAInance utilizes purely interactive input, working without the aid of annotations or scores.

The spelling of pitches can also be interesting. Honingh presents a pitch spelling model based on compactness in the Tonnetz [Honingh, 2009]. This work utilizes the Tonnetz to determine how pitches should be spelled instead of how they should be tuned. The Tonnetz being a useful aid for pitch spelling naturally extends to it being able to be used for the intonation of pitches, which is what ConsAInance uses it for. When a pitch can be spelled as the note that is currently being pressed, this pitch is a candidate for being the correct pitch to assign to that note.

In "Fundamental Principles of Just Intonation and Microtonal Composition" [Nicholson, 2018] Nicholson demonstrates the harmonic distance measure utilized in this work and provides the theoretical groundwork necessary to reason about just intonation and its applications.

Villegas and Cohen define the Golden Ear algorithm[Villegas, 2010]. A concept of roughness based on the fundamental frequencies of sounding notes is used to "produce chords containing intervals closer to pure intervals". Their aim is to get close to the ideal, while in this work we tune ideally using just intonation. Besides, ConsAInance is liberated from dealing with the concept of the frequency of sound. This is achieved by abstracting intonation to work with intervals and ratios instead of direct frequency analysis.

The approach taken in this paper involves the Tonnetz, and the usefulness of utilizing search algorithms on the Tonnetz for interactive and automatic intonation. As seen, several related works have tried their hand at automatic intonation, though none have implemented it exactly as this work. Polanksy however suggests the method used in this work [Polansky, 1987]. In his section on future work a suggestion is made regarding a tuning system utilizing a distance function in harmonic space. The distance function utilized for the ConsAInance system is the harmonic distance measure by Tenney as defined by Nicolson [Nicholson, 2018], but adapted slightly. The harmonic space is the

Tonnetz. The ConsAInance system is the implementation of this concept.

3 ConsAInance - System overview

ConsAInance is an automatic, interactive tuning system for midi enabled digital pianos or analogues thereof. It tunes incoming notes according to the principles of just intonation. This means we try tuning all notes according to singular or compound prime intervals. In effect, intervals such as 3/2, a perfect fifth, 5/4, a major third or 15/8, their compound interval, corresponding to a major seventh. Finding the interval that maximizes consonance for a pitch given the other pitches that are sounding is defined in the section on search on the Tonnetz.

To enable interaction with the tuning system, a MIDI controller was implemented. This controller takes inputs and presents them to a pitch configuration. The pitch configuration in turn seeks to provide a pitch for the given input. The MIDI controller only accepts direct MIDI input via MIDI enabled peripherals like digital pianos.

Figure 3 provides an overview of the structure of the ConsAInance system, while figure 4 demonstrates the execution flow of the system.

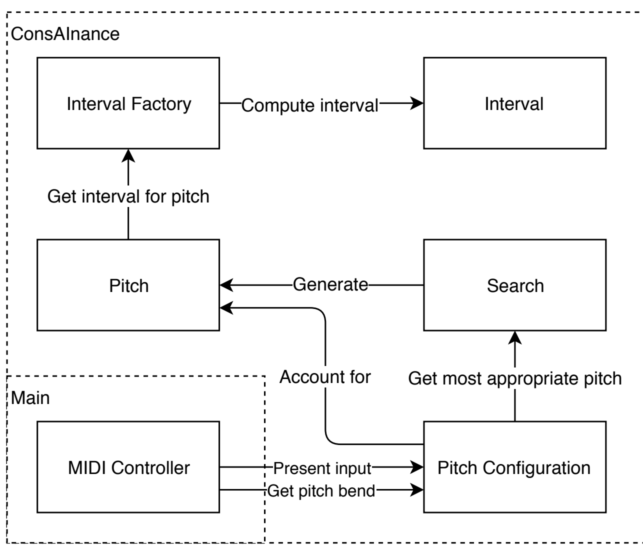

Figure 3: Basic overview of the structure of the ConsAInance system

4 Automatic intonation

When given an input, an automatic intonation system devises for that input the best intonation possible according to its internal rules. In the case of ConsAInance the input consists of MIDI messages provides to the system through

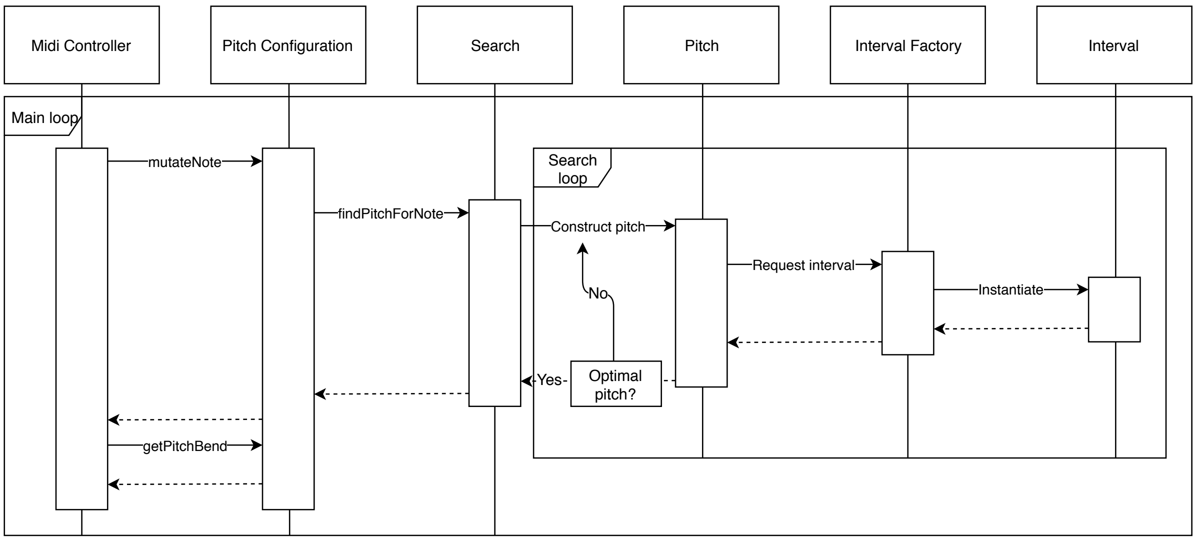

Figure 4: Basic overview of the execution flow in the ConsAInance system

a MIDI controller. Any incoming input is given a pitch that is tuned to be maximally consonant to the pitches already sounding. In 12TET all distances between semitones are equal.

To actually re-tune a note related to a pitch ConsAInance utilizes pitch bending on MIDI channels. To determine the magnitude of the pitch bend a logarithmic interpolation is required between the pitch class tuning according to 12TET an the tuning according to the system. The exact implementation of this concept is left as an exercise to the reader.

4.1 Pitch classes

Any pitch can be assigned a pitch class by evaluating its interval and fitting it to a class. A pitch class is an element of ordering. This ordering goes by perceived frequency and is analogous to note numbers. In ConsAInance there are twelve pitch classes, numbered 0 through 11. No interest is taken in note spelling and therefore a specification beyond simple numbering is not necessary. In musical context, the distance between neighbouring pitch classes is a single semitone, effectively specifying twelve distinct pitches per octave. This fits with the standard key layout on a piano. In the base case, class 0 is equal to the musical note C.

To determine if a pitch is a candidate for tuning, the pitch class must be evaluated against what note has been input. Leveraging the existing 12TET note classes, we can bootstrap a classification function to get a pitch class for arbitrary pitches. A note class in 12TET is centered on $2^{X/12}$, where $X \in \{0, 1, \cdots, 12\}$ for a system with 12 possible notes, these are equivalent to note classes. Going up and down a quarter tone from this value determines the lower and upper bounds of the pitch class in 12TET. A pitch has a corresponding interval which, when evaluated, results in a value between 1.0 and 2.0. This resulting value can be checked against the bounds of 12TET pitch classes to assign a class to the pitch in question. Equation (1) demonstrates the evaluation of a pitch, where e is the list of exponents contained in the pitch. The resulting value must then be divided or multiplied by two to produce a ratio between one and two, the unison and the doubling.

$$\text{iv(e)} = \prod_{i=0}^{n} p_i^{e_i} \tag{1}$$

4.2 Consonance evaluation

To be able to rank pitches, each pitch is evaluated for its harmonic distance. Multiple alternatives exist in literature, these have different growth rates but share the same main characteristics as far as our search is concerned. Examples of such functions and their evaluation are presented by Ryan in his work on mathematical harmony analysis [Ryan, 2016]. They are all measures of the complexity of an interval, thus any interval that is deemed more complex than another will score worse. The complexity of the interval rises along with the size of its prime exponents. The harmonic distance measure requires two lists of exponents, one provided by the pitch itself and a second provides by the pitch configuration as reference, the median exponents. The harmonic distance is defined in equation (2), where e_i equals the ith prime exponent of the pitch currently under review and r_i equals the ith prime exponent of the reference pitch. This is as defined by Nicolson in his fourth equation [Nicholson, 2018], but adapted to include a reference.

$$\text{HD}(e, r) = \sum_{i=0}^{n} |e_i - r_i| \log_2(p_i) \tag{2}$$

For pitch configurations with any sounding pitches the median pitch can be calculated to be used as the reference. The median pitch is the pitch containing the median prime exponents for the current pitch configuration. In the trivial case of the pitch configuration being empty the pitch representation of the unison is returned. Effectively, this results in no tuning adjustments for any note that is struck in isolation.

When input is processed the order in which notes are pressed is significant. This is an artifact of the nature of the search algorithm. It always needs an initial reference, the first pitch entered into the pitch configuration becomes the initial reference, the unison. For any additional pitches entering the pitch configuration the median pitch is calculated to use as the reference. This median is not bound to be an actual pitch. In the case of an even number of pitches sounding, the median will act like any regular median and take the average of the two nearest exponents. The median is an appropriate reference for the optimization of the harmonic distance of pitches. [Nicolas, 2012]. It effectively moves the global minimum in the harmonic distance space. This is demonstrated in the section on search on the Tonnetz. Any pitch with non-integer exponents can not trivially be resolved to an interval. Luckily such conversion is not necessary since such pitches can only occur as median pitches. They are used solely as reference for the calculation of harmonic distance, where only the exponents contained in the pitch matter and the exact interval is not considered.

5 Search on the Tonnetz

```
Data: Pitch configuration, note class of new pitch
Result: Optimal pitch for note class given the pitch configuration
if no pitches currently in pitch configuration then
  | return unison pitch;
end
initialize front, currentBest, medianExponents;
while front not empty do
  | currentPitch ← pop front;
  | if note class of currentPitch == noteClass && HD of currentPitch
  |   <HD of currentBest then
  |   | currentBest ← currentPitch;
  | end
  | add all not-visited children of currentPitch to front;
  | if for all pitches ∈ front : nopossibleimprovementfound then
  |   | return currentBest;
  | end
end
```

Algorithm 1: Optimal pitch search

There are several ways in which the implemented search method deviates from the standard form. Firstly, no single explicit goal state can be defined. If it could be the search would be redundant. There do however exist theoretically infinite implicit goal states, these are all the pitches on the Tonnetz that have the correct pitch class. Secondly, no care is given for the actual path traversed. Only the most optimal goal state is desired.

The Tonnetz is a relatively straightforward search space. Due to the nature of the harmonic distance measure, there exists only one local minimum. The initial pitch for search starts with all prime exponents set to zero. The pitch can be expanded by generating new pitches with +1 and -1 offsets for all exponents as visualized in figure 5. Given that the harmonic distance utilizes the absolute value of the exponents, any increase of absolute magnitude of an exponent will result in an increase in the harmonic distance. The median offsets the global minimum. In effect the harmonic distance behaves no differently from standard city block distance, though the latter is not inspired by music theory and is therefore arguably less interesting in the context of ConsAInance.

Due to the earlier described property of the harmonic distance measure the search algorithm can be terminated when the best candidate pitch on the search front is worse than the currently optimal pitch. Best-first ordering is used for the search front and the termination criterium is checked after expanding the current pitch.

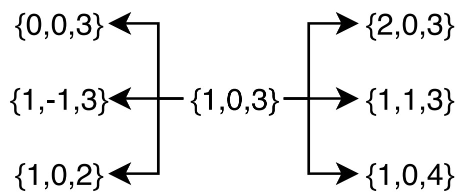

Figure 5: The expansion of a specific pitch

In ConsAInance only new input is tuned. Any pitches already sounding in the pitch configuration are fixed until the last key corresponding to its pitch class is released. This creates the aforementioned order dependence. This mostly does not pose a problem, as demonstrated in the results section. However, this does entail large intervals can be created by playing complex chords, holding the note with the largest interval and continuing to play. While these larger interval configurations stray further from 12TET tuning, they are still maximally consonant. These configurations do carry the risk of subjectively sounding more tense as found to be noticeable in subjective testing.

The search space can be complicated by using higher limit tunings. Higher limit tunings introduce more prime exponents which in turn increases the branching factor of the search algorithms. The intervals generated by the bigger primes in higher limit tunings are inherently more complex than intervals of lower primes. When utilizing such tunings the more complex intervals will not often be the most optimal. Though the stacking of smaller intervals can create more complex intervals than would be generated by starting with intervals with inherently higher complexity. This effect can be clearly seen in the results section as reported in table 4. It seems a prime limit of 11

The harmonic distance measure is only a first effort. It falls in line with the general consensus that pitches with equal pitch classes intervals with larger integers are more likely to be dissonant than intervals with smaller integers.

6 Results

The ConsAInance system successfully intonates arbitrary pitch configurations. It works with no perceivable added latency. On a system equiped with an AMD Ryzen 7 2700X processor and ConsAInance executing with a single thread the average time taken for an input to be tuned was between 0.4 and 1.5 milliseconds. The exact time taken depends on the tuning limit. When more primes are involved in tuning, the processing time increases due to the search algorithm having to search in more dimensions. This figure was gathered by playing a few pieces on a MIDI-enabled digital piano connected to the system. It is thus unlikely for a single player to play faster than the system can handle. It is important to note that the MIDI layer itself adds a certain amount of latency.

The relevant specifications of the system and peripherals used for testing and development are listed in appendix A.

During the authors own testing on the subjective experience of playing a piece of music through ConsAInance several things could be noticed. Firstly, due to the property that pitches hold their tunings until all notes related to these pitches are released the output can become quite messy. When playing music it is not uncommon for at least one note being held from a prior phrase when starting to play a new phrase. This held note keeps its tuning. Since all pitches are tuned relative to the median pitch, this held note influences the tuning of the new phrase. This is where melismatic tuning will prove to be an improvement. With melismatic tuning the held note of the prior phrase will be re-tuned along with the incoming notes. This, however, is for a future work to provide. Secondly, when playing both the ConsAInance retuned result and the digital pianos' own output at the same time, a clear dissonance is felt. ConsAInance tuned instruments can thus not be sounded together with 12TET tuned instruments.

6.1 ConsAInance JI versus 12TET

To evaluate the tuning behaviour of ConsAInance a few different scenarios were tested.

Firstly, the difference of tuning between 12TET en ConsAInance was evaluated. Table 3 shows the difference between 12TET tuning and ConsAInance 13-limit tuning with interval sizes measured in cents. The results show small corrections on some intervals, while larger corrections are made on others. While the smaller corrections are relatively unnoticeable, the larger corrections are definitely felt. The evaluation of the subjective experience of the produced tuning should be a part of future work. In table 4 the transition from 3-limit just intonation to 5-limit just intonation resulted in a massive decrease in interval complexity. The step up to 7-limit saw improvement for several intervals, while the step to the 13-limit did not yield any changes. In the case of 12 note-per-octave systems 11-limit tunings seem optimal, providing intervals with low complexity while having a reasonable branching factor in the search algorithm.

In table 5 we can see the intonation performance of ConsAInance when tested against, major, minor and diminished seventh chords. The results show the involved intervals to be reversed when the input order of chords is reversed. Both orders are aurally equivalent.

Distance	12TET	13-Limit JI
C - C#	100	112
C - D	200	231
C - D#	300	316
C - E	400	386
C - F	500	498
C - F#	600	551
C - G	700	702
C - G#	800	814
C - A	900	884
C - A#	1000	969
C - B	1100	1088

Table 3: 12TET interval tuning compared to 13-limit JI tuning found by ConsAInance for simple 2-note scenario's

Distance	3-Limit JI	5-Limit JI	7-Limit JI	11-Limit JI	13-Limit JI
C - C#	256/243	16/15	16/15	16/15	16/15
C - D	9/8	9/8	8/7	8/7	8/7
C - D#	32/27	6/5	6/5	6/5	6/5
C - E	81/64	5/4	5/4	5/4	5/4
C - F	4/3	4/3	4/3	4/3	4/3
C - F#	729/512	45/32	10/7	11/8	11/8
C - G	3/2	3/2	3/2	3/2	3/2
C - G#	128/81	8/5	8/5	8/5	8/5
C - A	27/16	5/3	5/3	5/3	5/3
C - A#	16/9	16/9	7/4	7/4	7/4
C - B	243/128	15/8	15/8	15/8	15/8

Table 4: Results for tuning with different prime limits in ConsAInance for simple 2-note scenario's

		Pitch class											
Chord	Order	0	1	2	3	4	5	6	7	8	9	10	11
Major	0-4-7	1/1				5/4			3/2				
	7-4-0	4/3				5/3			1/1				
Minor	0-3-7	1/1			6/5				3/2				
	7-3-0	4/3			8/5				1/1				
Dim. 7th	0-3-6-9	1/1			6/5			7/5			128/75		
	9-6-3-0	75/64			10/7			5/3			1/1		

Table 5: Results for tuning major, minor and diminished seventh chords in left-to-right and right-to-left note order using 13-limit ConsAInance

7 Conclusion and future work

As discussed in the results section, ConsAInance successfully intonates arbitrary pitch configurations. It is interactive in the sense that it has low enough latency to be used for live playing. This entails an adapted shortest path algorithm to be suitable for automatic and interactive intonation of arbitrary pitch configurations. The interactivity falters on the previously discussed held-note intonation behaviour, which can lead to correct but subjectively unpleasant tunings. This can be solved by extending the system to allow for melismatic tuning, as posited as an option for future work.

Several options exist to expand on the ConsAInance system, the following list summarizes the options that are deemed interesting by the author.

- The system could be expanded to allow for multiple instruments to play through it at once. The current implementation supports only one instrument. For the solitary experimental bedroom musician this is fine, but one can imagine a full-blown multi-person synth band to be
- More research should be conducted into devising consonance measures more directly rooted in psychoacoustics. This can lead to more accurate and biologically plausible intonation systems.
- ConsAInance's output should be investigated for subjective value. It seems interesting to conduct research into the subjective experience of music tuned with ConsAInance in different cultures. One could imagine a culture with more exposure to microtonal music to have appreciate the output differently from western listeners accustomed to 12TET tuned music.
- The algorithm could be adjusted to allow for melismatic tuning. This allows for the re-intonation of pitches that are already sounding, leading to potential higher accuracy when tuning. This has the possibility to introduce interesting textures to music and solves the held-note realtime playing discussed in the results section.
- The system could be adapted to allow a user to exclude certain primes. One could imagine defining a tuning using only the primes 7, 11 and 23 instead of standard limit tunings. This would allow for more freedom of expression and may lead to interesting results.

References

[Denckla, 1995] Denckla, B. (1995). Dynamic intonation for synthesizer performance.

[Honingh, 2009] Honingh, A. (2009). Compactness in the euler-lattice: A parsimonious pitch spelling model. *Musicae Scientiae*, XIII(1):117–138.

[Nicholson, 2018] Nicholson, T. . S. M. (2018). Fundamental principles of just intonation and microtonal composition. *Musicae Scientiae*, XIII(1):117–138.

[Nicolas, 2012] Nicolas, A. (2012). The median minimizes the sum of absolute deviations (the L_1 norm).

[Polansky, 1987] Polansky, L. (1987). Paratactical tuning: An agenda for the use of computers in experimental intonation. *Computer Music Journal*, 11(1):61–68.

[Ryan, 2016] Ryan, D. (2016). Mathematical harmony analysis. *Computing Research Repository*, abs/1603.08904.

[Sabat, 2008] Sabat, M. (2008). An algorithm for real-time harmonic microtuning.

[Sethares, 1994] Sethares, W. (1994). Adaptive tunings for musical scales. *The Journal of the Acoustical Society of America*, 96(1):10–18.

[Villegas, 2010] Villegas, J. & Cohen, M. (2010). Roughness minimization through automatic intonation adjustments. *Journal of New Music Research*, 39:75–92.

Appendix A - Testing system specification

The testing system contained an AMD Ryzen 7 2700X processor supported by 16 gigabytes of DDR4 RAM at 3200Mhz running Ubuntu 18.04.2 LTS. ConsAInance was run on OpenJRE 1.8.0_212. Two input devices were confirmed to be working; the AKAI MPK Mini and the Casio CDP-230R.