

Toward Automatic Tuning of the Piano

Joonas Tuovinen

Acoustics Lab,
Dept. Signal Processing & Acoustics,
Aalto University, Espoo, Finland
joonas.tuovinen@aalto.fi

Jamin Hu

Sibelius Academy,
University of the Arts Helsinki,
Helsinki, Finland
jamin.hu@uniarts.fi

Vesa Välimäki

Acoustics Lab,
Dept. Signal Processing & Acoustics,
Aalto University, Espoo, Finland
vesa.valimaki@aalto.fi

ABSTRACT

The tuning of a piano is a complicated and time-consuming process, which is usually left for a professional tuner. To make the process faster and non-dependent on the skills of a professional tuner, a semi-automatic piano tuning system is developed. The aim of the system is to help a non-professional person to tune a grand piano with the help of a computer and a motorized tuning machine. The system composes of an aluminum frame, a stepper motor, an Arduino processor, a microphone, and a laptop computer. The stepper motor changes the tuning of the piano strings by turning the pins connected to them whereas the aluminum frame holds the motor in place. The Arduino controls the motor. The microphone and the computer are used as a part of a closed loop control system, which is used to tune the strings automatically. The control system tunes the strings by minimizing the difference between the current and optimal fundamental frequency. The current fundamental frequency is obtained with an inharmonicity coefficient estimation algorithm, and the optimal fundamental frequency is calculated with a novel tuning process, called the Connected Reference Interval (CRI) tuning. With the CRI process, a tuning close to that of a professional tuner is achieved with a deviation of 2.5 cents (RMS) between the keys A_0 and G_5 and 8.1 cents (RMS) between $G\#_5$ and C_8 , where the tuner's results are not very consistent.

1. INTRODUCTION

Tuning a piano is known to be a complicated process, which takes a considerable amount of time and effort. To many musicians tuning all the 200 plus strings of the instrument is a daunting task, especially as doing it incorrectly may leave the instrument in even worse tune. Because of this the tuning of a piano is usually left to professional tuners.

The scale of a piano is based on the twelve-tone equal temperament scale (12-ET), which specifies the fundamental frequency of each key. The difficulty of tuning a piano comes from the fact that, because mode frequencies

of piano strings deviate from the harmonic series, in a phenomenon called inharmonicity, tuning the fundamental frequencies of the strings to follow the 12-ET scale leads the instrument to sound out of tune [1]. Instead, professional tuners use the beating effect, produced by two frequencies close to each other, to tune the instrument, as the 12-ET scale specifies beating rates for each interval [2].

To make the process of tuning a piano faster and non-dependent on the skills of a professional tuner, a semi-automatic piano tuning system is developed in this work. There have been related previous developments, e.g. [3], but an automatic piano tuning system is still not commonly used. The proposed tuning system is aimed towards tuning a grand piano with the help of a non-professional tuner. The system includes a stepper motor, an aluminum frame, an Arduino Uno [4], a microphone and a computer.

The system uses closed loop control to change the fundamental frequency of a string from the current frequency to a target frequency. The current fundamental frequency is determined by an inharmonicity coefficient estimation algorithm. The target frequency is determined by a novel tuning process, called the Connected Reference Interval (CRI) tuning process, which calculates the optimal fundamental frequency for each string based on the beating rates between the current and previously tuned strings. The change from current to target fundamental frequency is implemented with a Proportional-Integral-Derivative (PID) controller, which is discussed later in more detail.

There have been previous algorithms that are designed to find the optimal tuning for a piano, but these algorithms require all or some of the strings to be recorded before tuning [5–7], unlike the CRI tuning process, which calculates everything while the tuning is done.

The paper is structured as follows. In Section 2 the structure of the piano tuning robot, in charge of turning the pins of the piano, is described. Next, in Section 3 the control system which automatically changes the tune of a string to a desired tuning is discussed. The system needs the current fundamental frequency, discussed in Section 4, and the target fundamental frequency, described in Section 5 to tune the string. The accuracy of the CRI tuning system is also evaluated in Section 5. Finally, Section 6 includes conclusion of the project as well as discussion about the future of the system.



Figure 1. Proposed piano tuning system with the Arduino processor near the midpoint of the picture and the stepper motor on its left-hand side, attached to the topmost aluminum bar.

2. STRUCTURE OF PIANO TUNING ROBOT

In the process of tuning a piano, tuners use a lever to tune the strings of the piano. The lever is used to turn a pin which has a string wrapped around it. Turning the pin changes the tension of a string and this change in tension determines the fundamental frequency of the string according to:

$$f_0 = \frac{1}{2L} \sqrt{\frac{T}{m/L}}, \quad (1)$$

where L is the length of the string, m is the mass of the string and T is the tension of the string. The fundamental frequency of a string (along with the inharmonicity coefficient which will be discussed later) determines the mode frequencies (partials) that tuners listen to when tuning the instrument.

The first step in making an automatic piano tuner was to create a structure which allows the automatic control of string tension. The proposed structure (stepper motor, aluminum frame and Arduino) is able to turn the pins of the piano with high precision (small angle) and has enough torque to turn even the tightest strings. The Arduino is able to turn the pins of the piano depending on input given by the computer with a program uploaded to it. The prototype structure can be seen in Figure 1.

3. CONTROL SYSTEM

The control system used to automatically determine the number of steps needed to change the fundamental frequency of a string from current to a desired value, is a closed loop control system. The general structure of a closed loop control system can be seen in Figure 2a. The system has a reference value as its input, and the aim of the control loop is to minimize the difference between the reference and the value measured from the output of the system with the sensor. This is accomplished with the controller, which changes the input to the process based on the difference between the reference and the measured output.

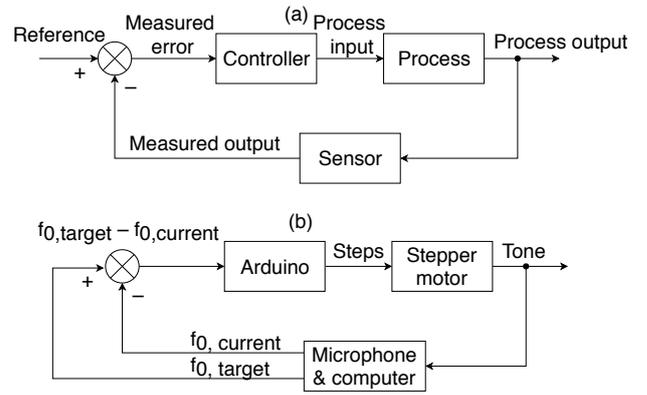


Figure 2. Block diagram of (a) a general closed loop control system (b) closed loop control system used in the piano tuning robot.

Figure 2b shows a diagram of the piano tuning system. In the piano tuning system the Arduino is used as the controller as the microprocessor has enough processing power to do this task and it controls the number of steps that the stepper motor takes. The process is the stepper motor turning a pin and thus affecting the fundamental frequency of the string attached and the process output is a tone produced by the string. To be able to measure the fundamental frequency of a string based on the tone, a microphone and computer are added to the system. In addition to working as the sensor, providing the measured output, the two components provide the reference as well, as information extracted from the current tone as well as previous tones is used to calculate the target frequency. The difference between the current and target fundamental frequency is used by the controller (Arduino) to calculate the appropriate number of steps the stepper motor should take.

A PID (proportional-integral-derivative) control scheme is used by the Arduino to control the stepper motor. PID controller calculates the error value $e(t)$ between the reference ($f_{0,target}$) and measured output ($f_{0,current}$) and applies a correction to the process based on proportional (P), integral (I) and derivative (D) terms. This control value $u(t)$ (number of steps) attempting to minimize the difference between the reference and measured output is then applied to the process.

4. FUNDAMENTAL FREQUENCY

The fundamental frequency of a string can be estimated looking at the spectrum of its tone. This is done by finding spectral peaks belonging to mode frequencies of the string. The relationship between these partials and the fundamental frequency is affected by an effect called inharmonicity. In this section, inharmonicity as well as algorithms for estimating the fundamental frequency of a string are reviewed.

4.1 Inharmonicity

The partials of an ideal string are integer multiples of its fundamental frequency (harmonics). However, real strings have stiffness, which acts as a restoring force, making the

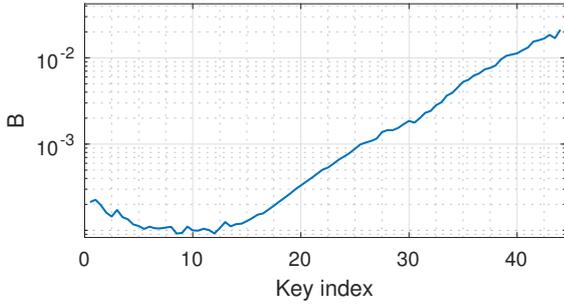


Figure 3. Inharmonicity coefficients of piano strings calculated with an inharmonicity coefficient estimation algorithm from tones recorded from a Yamaha grand piano. The strings for keys 1–7 are wrapped twice in wire, those of keys 8–26 are wrapped once, and the rest of the strings are unwrapped.

mode frequencies deviate from the harmonic series. This deviation is called inharmonicity and the effect is greater with higher modes, as they have more bends [8].

The mode frequencies of piano strings can be calculated from the following equation [9]:

$$f_k = k f_0 \sqrt{1 + B k^2}, \quad (2)$$

where f_k is the frequency of the k th mode (also known as the k th partial), f_0 is the fundamental frequency, and B is the inharmonicity coefficient of the string. The value of inharmonicity coefficient of a solid string depends on the length, tension and radius of the string according to the following equation [8]:

$$B = \frac{\pi^3 r^4 E}{8 T L}, \quad (3)$$

where r is the radius of the string, E is Young's modulus, T is tension, and L is the length of the string. The strings in the bass end are wrapped in wire to lower their fundamental frequency by increasing their linear mass. This increases their inharmonicity slightly from Equation 3, but not as much as adding the linear mass by increasing the radius of the solid string.

Inharmonicity coefficient values of a Yamaha grand piano can be seen in Figure 3. It can be seen how the short strings in the treble end have inharmonicity coefficients close to $1e^{-2}$ and the long bass strings have inharmonicity coefficients close to $1e^{-4}$. It can be also seen how the values of inharmonicity coefficients increases towards the bass end, as the strings are wrapped in one or two layers of wire.

4.2 Inharmonicity Coefficient Estimation

Inharmonicity coefficient estimation algorithms find spectral peaks belonging to partial frequencies and make estimations for the inharmonicity coefficient and fundamental frequency based those values. These algorithms need rough estimations for the values of B and f_0 and make better ones based on the found partial frequencies. The difficulty of finding spectral peaks, belonging to partial

frequencies, comes from distinguishing partial frequencies from other spectral peaks.

There has been many algorithms tackling the issue of partials frequency estimation [10–14], but most of these algorithms suffer from high computational complexity. As the control system of the piano tuner needs the sensor (microphone and computer) to calculate the value of fundamental frequency on every iteration loop, the chosen algorithm has to be fast as well as accurate.

The Median-Adjustive Trajectories [14] (MAT) algorithm for estimating the inharmonicity coefficient best fulfills the accuracy and runtime requirements of the piano tuner. The algorithm calculates estimations for B and f_0 based on the frequencies of known partials, and finds new partials based on these estimations. The algorithm is based on the idea that if the frequencies of two partials are known, the value of B can be calculated purely based on their values.

If equation 2 is solved in terms of fundamental frequency, with partial number m :

$$f_0 = \frac{f_m}{m \sqrt{1 + B m^2}}, \quad (4)$$

and then 4 is substituted into equation 2 for partial k :

$$f_k = k f_0 \sqrt{1 + B k^2} = k \frac{f_m}{k \sqrt{1 + B m^2}} \sqrt{1 + B k^2}. \quad (5)$$

The value of B can be solved from this equation:

$$B = \frac{(f_k \frac{m}{k})^2 - f_m^2}{k^2 f_m^2 - m^2 (f_k \frac{m}{k})^2}. \quad (6)$$

This means that if the frequencies of first two partials can be found from the spectrum of the tone, an estimation for the value of B can be made. As the first two partials do not deviate very much from the harmonic series, these can be found with good initial estimations of f_0 and B . This new B estimation can then be used together with the found partial frequencies to make new estimations for f_0 with Equation 4. After that, these f_0 and B estimations can then be used to find new partial frequencies from the spectrum.

Figure 4 shows the block diagram of the MAT algorithm. The diagram shows how the initial estimates of the f_0 and B are used to find the first two partials of the tone. The original MAT algorithm suggested that first two partials should be found by looking at a window around frequencies $f_{0,init}$ and $2f_{0,init}$. A small adjustment to the algorithm is made by using the initial values of f_0 and B in Equation 2 to calculate estimates for the first two partials. By doing so a slight improvement to the accuracy of the algorithm is achieved.

After the first two partials are found, the first B estimation can be made with Equation 6. This value is stored to an array of B estimates and a median of this array is taken to make two estimation for the value of f_0 . The f_0 estimations are then stored into an array of f_0 estimations and the median values of the B and f_0 arrays are used in Equation 2 to make an estimation for the value of the third partial f_3 . A smaller window around the estimated value can be used to find the partial, as the estimate is more accurate than the

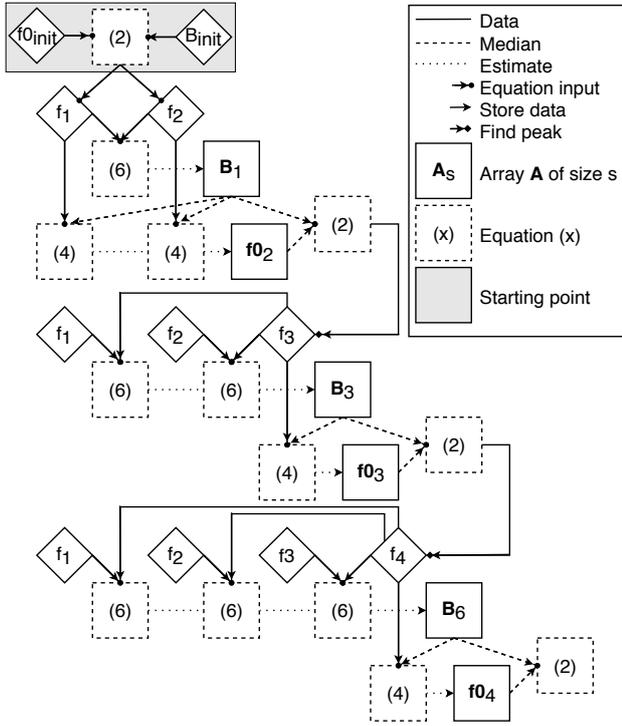


Figure 4. Block diagram of the MAT algorithm adopted from [14]

estimation for the first two partial. The process moves on making new estimations for B , f_0 and partial frequencies until the found partials have a magnitude below a specified threshold. At this point the median of the B and f_0 estimation are then used for the final estimations.

With this method the estimation for the current value of f_0 is gotten. These f_0 and B estimates will also be used in the CRI tuning process, as they can be used in Equation 2 to represent the partial frequencies of the string, which are used to calculate beat rates between piano strings.

5. CRI TUNING PROCESS

The CRI tuning process determines a target frequency for every piano string. The process specifies the order of tuning so that as many intervals as possible can be used to calculate the target fundamental frequency. The keys of the piano are connected to one, two or three strings, and all the strings connected to the same key are called a string unison. The tuning process specifies that a single string from each string unison is to be tuned at first, using single strings from other unisons as a reference, and after that, the rest of the strings in the same unison are tuned using the tuned string as a reference. The other strings in the unison are tuned to have approximately a 1.5 cent difference to the reference as that maximizes the decay time of the combined strings [15]. From here on, the tuning process talks only about the single string of every string unison, which is tuned using single strings of other unisons as reference.

Strings are tuned one by one starting with a reference tone (A_4) which is tuned to a reference frequency, and after that, the rest of this strings are tuned in the following order: ref-

erence octave (F_3 to F_4), tones above the reference octave ($F\#_4$ to C_8) and tones below the reference octave (E_3 to A_0).

The target fundamental frequency for each string is found by optimizing the beating rates between the string that is currently tuned and all the strings that have been already tuned and are a certain interval away from that string.

5.1 Beats

When a tone contains two frequencies that are close to each other, the frequencies cause periodic changes in the amplitude of the tone. These amplitude modulations are called beats and the frequency of these modulations can be calculated from equation [16]:

$$f_B = |f_2 - f_1| = \Delta f, \quad (7)$$

where f_1 and f_2 are the two frequencies close to each other. Equation 7 applies only until a certain point. As the two frequencies get further away from each other the frequency of beats gets faster at first, until unpleasant roughness between the two frequencies emerges. From this roughness two distinct tones can be heard after Δf exceeds the limit of frequency discrimination and after Δf surpasses the critical band, the roughness disappears and only two distinct frequencies can be heard [16].

5.2 Scale of the Piano

The tuning of a piano is based on the equal temperament scale, which makes all the steps in the scale equal. This means that the ratio between fundamental frequencies of subsequent tones in the scale should be the same. More specifically the scale is a twelve-tone equal temperament scale (12-ET) which in addition to having equal steps specifies the ratio between an octave to be 2:1 (f_0 of the lower tone is two times the f_0 of the higher one) and divides each octave into twelve steps. The ratio of 2:1 and 12 equal steps leads a single step in the scale to have a ratio of $\sqrt[12]{2}:1$, as $12 \sqrt[12]{2} = 2$.

The distance between two tones in a scale is called an interval. Musical scales are usually designed so that some partials of two harmonic tones having a certain interval line up to produce the minimum level of roughness between tones. This is achieved by designing fundamental frequencies of the intervals to have certain frequency ratios, as harmonic overtones are integer multiples of the fundamental. For example, if the fundamental frequencies of two tones have frequency ratio of 2:1, the $2k$ partials of the lower tone match with the k partials of the higher one ($k = 1, 2, 3, \dots$). The names of these intervals and ratios of their fundamental frequencies are listed in the first two columns of Table 1.

The way the 12-ET scale is designed leads all other interval ratios except for the octave to deviate. The amount of deviation per interval can be seen in the third column of Table 1. This deviation is measured in cents, which is a logarithmic unit, expressed as:

$$Deviation = 1200 \log_2 (b/a), \quad (8)$$

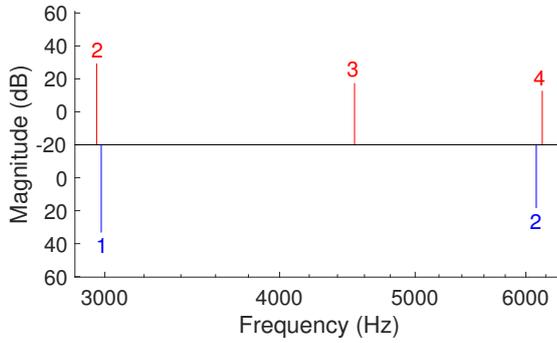


Figure 5. The first two matching partials in the octave between Bb_7 and Bb_8 .

where the deviation is positive if b is greater than a . The Distance column in Table 1 shows the number of steps (semitones) is between each interval in the 12-ET scale.

This deviation leads the intervals to have specific beating rates, which tuners use to tune the instrument. The reason why the fundamental frequencies of the 12-ET scale cannot be used to tune the instrument is because piano strings are inharmonic, and thus the spacing of fundamental frequencies specified by the scale do not produce wanted beat rates. Instead the spacing is slightly wider as the partials deviate upward from the harmonic series. This leads the tuning of the piano to be "stretched", meaning that when compared to the 12-ET scale, a tuning performed by a professional tuner is slightly higher in the treble and lower in the bass. Also, as the inharmonicity is different with each string, it is impossible for all the partials that are integer multiples of the frequency ratios to have the same beating rate. Because of this, piano tuners listen to all beats and tune the strings in such a way that none of the prominent beats deviate too much from the desired beating rate.

An example of this can be seen in Figure 5 which shows the partial frequencies and magnitudes of the octave between Bb_7 and Bb_8 . According to Table 1 the beating rate and thus the difference between the two sets of frequencies (2:1 and 4:2) should be zero, but this is not possible as matching either of the two sets would leave the other one to have even more deviation. Instead the tuner has made a compromise. The second partial of Bb_7 is tuned

Interval	Ratio	Deviation	Distance
Octave	2:1	0	12
Perfect fifth	3:2	-1.96	7
Perfect fourth	4:3	+1.96	5
Major sixth	5:3	+15.64	9
Major third	5:4	+13.69	4

Table 1. Several intervals. The ratio tells which partials of the lower tone is closest to the partial of the higher tone (partial of lower tone : partial of higher tone). The deviation tells how much deviation there is between these partials according to the 12-ET scale. The distance is the number of semitones between the two tones.

slightly below the frequency of the first partial of Bb_8 and the fourth partial of Bb_7 is tuned slightly above the second partial of Bb_8 .

Electronic tuners that use partial frequencies of octaves to tune a piano match only a pair of partials. For example, a 2:1 method of tuning octaves matches only the second partial of the lower tone with the first partial of the higher tone and a 4:2 method of tuning octaves uses only the fourth and the second partial [6].

5.3 Calculating Target Frequencies

As the 12-ET scale specifies beating rates for each interval, and those beating rates should be calculated as the difference between the inharmonic partials of piano tones, the values of f_0 and B obtained with the MAT algorithm can be used to find the fundamental frequency that provides said beat rates. The beating rate in cents between two frequencies within a certain interval can be calculated from equation:

$$1200 \log_2 \left(\frac{f_{l,n+m}}{f_{k,n}} \right), \quad (9)$$

where c is the difference in cents, $f_{k,n}$ is the k th partial of the n th tone and $f_{l,n+m}$ is the l th partial of the tone which has a m semitone difference (interval) from n . The values of c , k , l and m for the first matching partials of each interval can be seen in Table 2 (for other matching partials integer multiples of k and l are used and all other variables stay the same).

Equation 9 can be rewritten in terms of f_0 and B by using Equation 2 to calculate partial frequencies:

$$1200 \log_2 \frac{l f_{0,n+m} \sqrt{1 + B_{n+m} l^2}}{k f_{0,n} \sqrt{1 + B_n k^2}} - c = 0. \quad (10)$$

When Equation 10 is used to calculate the sum of multiple intervals with the same tone n , the following equation is obtained:

$$\sum_{i=0}^{N-1} \left[\log_2 \left(\frac{l_i f_{0,n+m_i} \sqrt{1 + B_{n+m_i} l_i^2}}{k_i f_{0,n} \sqrt{1 + B_n k_i^2}} \right) - C_i \right] = 0, \quad (11)$$

where N is the number of intervals used for the tuning and k_i , l_i , m_i , and c_i are the values k , l , m , and c , respectively,

Interval	c	k	l	m
Octave (up)	0	2	1	+12
Perfect fifth (up)	-1.96	3	2	+7
Perfect fourth (up)	+1.96	4	3	+5
Major sixth (up)	+15.64	5	3	+9
Major third (up)	+13.69	5	4	+4
Octave (down)	0	1	2	-12
Perfect fifth (down)	+1.96	2	3	-7
Perfect fourth (down)	-1.96	3	4	-5
Major sixth (down)	-15.64	3	5	-9
Major third (down)	-13.69	4	5	-4

Table 2. Values of c , k , l and m for several intervals.

for a specific interval and C_i equals

$$C_i = 2^{c_i/1200}. \quad (12)$$

An estimation for the value of $f_{0,n}$ can be made by solving it from Equation 11 with the assumption that the inharmonicity coefficient of the string does not change during the tuning. This estimate is fairly accurate as the change in tension changes f_0 much more than B . Other coefficients in the equation are known, as $n + m$ is the index of a previously tuned string with known values of f_0 and B . It should be noted that the intervals used for this tuning are a design choice and that some intervals will get a tuning closer to that of a human tuner, as human tuners use only specific intervals to tune the instrument [2].

When $f_{0,n}$ is solved from Equation 11, the following equation is obtained:

$$f_{0,n} = \left(\frac{\prod_{i=0}^{N-1} A_i}{\prod_{i=0}^{N-1} 2^{c_i/1200}} \right)^{1/N}, \quad (13)$$

where

$$A_i = \frac{l_i f_{0,n+m_i} \sqrt{1 + B_{n+m_i} l^2}}{k_i \sqrt{1 + B_n k_i^2}}. \quad (14)$$

The fundamental frequency of piano strings can be computed using Equation 13 by comparing multiple intervals. However, as the equation uses one set of partials per interval, the process does not take higher partials into consideration.

5.4 Weights

To take all the audible beats into consideration in a similar way as an human tuner does, weights can be added to Equation 13. The weight of a set of partials producing beating within an interval is calculated by taking the maximum loudness of the beating effect as well as masking into consideration. Masking is a phenomenon in which soft sounds cannot be heard because of loud ones occurring at the same time, or in other words, louder sounds mask softer sounds. When weights are added to equation 13, the following form is obtained:

$$f_{0,n} = \left(\frac{\prod_{i=0}^{N-1} A_i w_i}{\prod_{i=0}^{N-1} 2^{c_i w_i/1200}} \right)^{1/\sum_{i=0}^{N-1} w_i}, \quad (15)$$

where w_i is the weight of a specific interval.

The weights are distributed in a way that the sum of the weights for each interval is one, so the weight of each interval is the same. The weights are calculated with the following steps:

1. Find the magnitude of partial frequencies: The magnitudes of partial frequencies can be found and stored by modifying the MAT algorithm to do so.
2. Apply A-weighting: The A-weighting is applied to approximate the frequency-dependent sensitivity of human hearing.

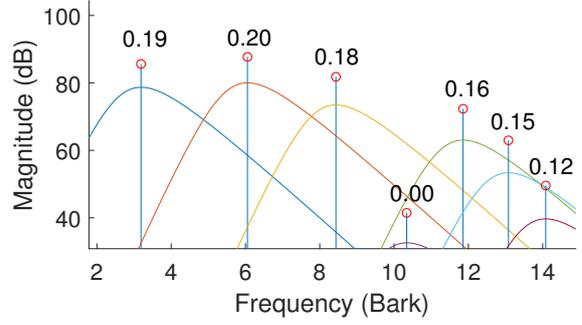


Figure 6. Weights (stems) of the beats between E_3 and E_4 (octave) and the corresponding masking thresholds.

3. Masking: An approximation of masking can be calculated by using a spreading function (SF). A popular SF proposed by Schroeder is used, as it is independent of the masking SPL, which is unknown [17]. The SF is shifted slightly lower depending on the tonality of the masker [18].
4. Weights: After all partials under the masking threshold have been taken out of consideration, the weights for each set of partials are calculated as

$$w_i = \frac{M_i}{\sum_{n=0}^{N-1} M_n}, \quad (16)$$

where w_i is the weight of i th matching partial, M_i is the maximum magnitude of the beats produced by the partials, and N is the total number of matching partials over the masking threshold.

Figure 6 shows the weights of the octave between E_3 and E_4 . It can be seen that the sum of these weights is one and that all beats under the threshold have a weight of zero.

5.5 Accuracy

The accuracy of the CRI tuning process was estimated by comparing it to a tuning performed by a professional piano tuner. The deviation (in cents) between the first partials of each tuning was used for the comparison. Single string recordings of all the 88 keys of a Yamaha grand piano were made the next day after tuning. The tuning accomplished by the CRI tuning process was emulated by resampling the recorded tones.

The accuracy was evaluated without weights, using the first matching partials for each interval, and with weights. The appropriate kind of distribution of weights for all partials of each interval could not be achieved yet, and too much weight was given to higher partials. This led to excessive amount of stretching, much more than that of the tuner. Because of this, only the first fifteen partials were considered for the algorithm with weights.

Both algorithms (with and without weights) use the same values and intervals for the reference tone and the reference octave. A_4 is used as the reference tone and the first partial of this tone was tuned to match the first partial of A_4 tuned by the tuner. This way the tunings could be compared. The

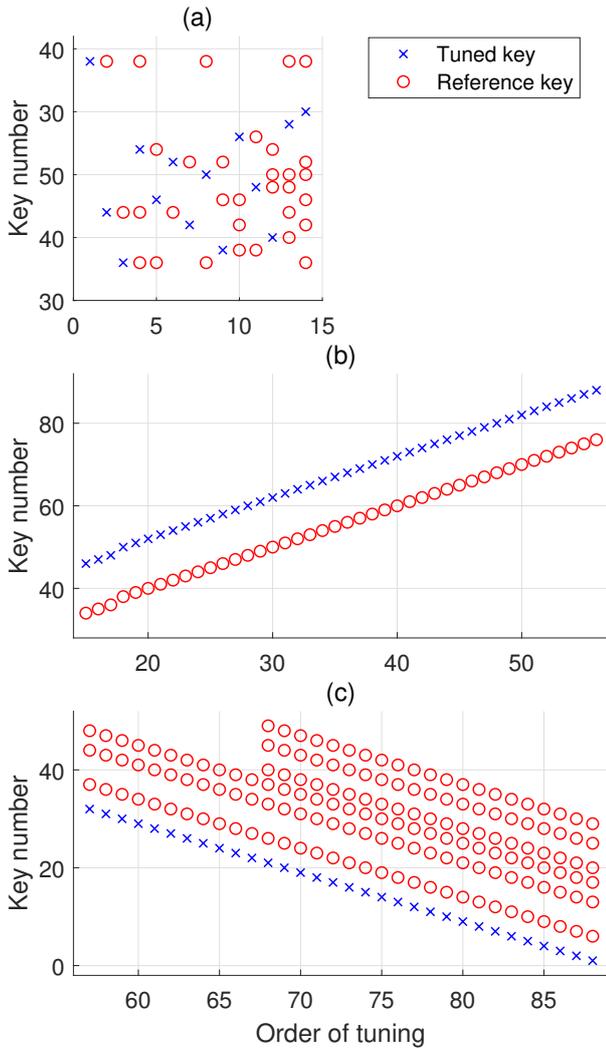


Figure 7. Order of tuning and reference intervals for the process without weights. Each cross (\times) represents the key that is tuned while the circles (o) are the reference keys.

reference octave is tuned according to the “Defebaugh F-F” temperament, which is a tuning scheme commonly used by piano tuners [2].

For the algorithm without weights, matching the following intervals gave the best result:

- $F\#_4(46)$ to $C_8(88)$: Octaves
- $E_3(32)$ to $F\#_2(22)$: Octaves, fifths, tenths.
- $F_2(21)$ to $A_0(1)$: Octaves, fifths, tenths, seventeenths, double octaves, and double octaves and a third.

Figure 7 shows the order of tuning and the intervals used for the algorithm without weights. The crosses show the key that is being tuned whereas the circles above and below it are the keys that are used as a reference. For the algorithm with weights, using an octave and a double octave gave the best result.

Figure 8 shows the tuning curve produced by the algorithm without weights. Its deviation from the tuning conducted by the professional tuner is presented in Table 3. The tuning done by the algorithm with weights can be seen

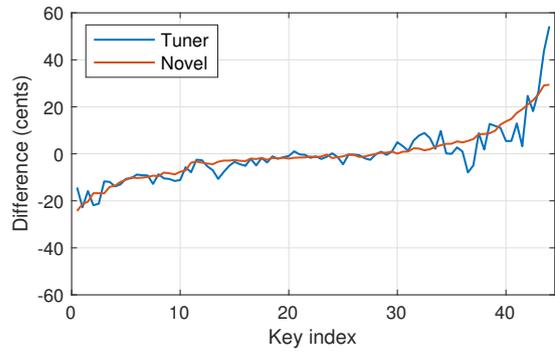


Figure 8. Deviation of the professional tuner and the CRI process (without weights) from 12-ET.

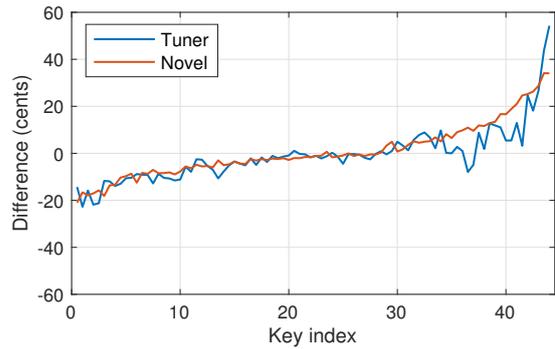


Figure 9. Deviation of the professional tuner and the CRI process with weights calculated from 15 first partials.

in Figure 9, and the corresponding deviations from the professional tuner’s result are presented in Table 4.

It can be seen that the algorithm without weights and optimized intervals gave a better accuracy with the overall deviation of 5.1 cents (RMS) than the one using weights, which had an overall deviation of 6.12 cents (RMS). It can also be seen that the deviation mostly happens in the treble end. This is most likely because these tones have a high degree of inharmonicity as well as a very short decay time, which make it harder for the tuner to hear and count beats.

6. CONCLUSIONS

In this paper a semi-automatic tuning system aimed toward tuning a grand piano with the help of a non-professional tuner was presented. The system uses a stepper motor attached to an aluminum frame to turn the tuning pins of the piano. The stepper motor is controlled by an Arduino pro-

Keys	RMS deviation (cents)
A_0 to E_3	3.2
Reference octave	1.3
$F\#_3$ to C_8	6.3
All	5.1

Table 3. Average deviation between the professional tuner and the CRI process without weights.

cessor, which is a part of a closed loop control system, to automatically adjust the tension of the strings. The control system also includes a microphone and a computer, which are used to measure the fundamental frequency and the inharmonicity coefficient of the piano strings as well as the magnitudes of partial frequencies from the tone of the string. The frequency values are obtained using an inharmonicity coefficient estimation algorithm called MAT and are used to calculate the difference between the current and the target fundamental frequency.

The target fundamental frequency is determined with a CRI tuning process using beating rates between the partials of several intervals. The process specifies the order of tuning for the strings to get the maximum number of intervals for its estimation. The process can calculate the fundamental frequency for either a specified set of partials, or for all partials, using weights. The process with and without specified partials were compared to a tuning conducted by a professional tuner. The process with specified partials was based on the first matching partials of each interval. With the optimal intervals, both processes gave great results with an RMS of 5.1 cents of deviation with specified partials and 5.7 cents without them.

Acknowledgments

This work was supported in part by NordForsk’s Nordic University Hub “NordicSMC”, project no. 86892. Jamin Hu would like to thank Mr. Kari Kääriäinen, Master Model Maker at Aalto Design Factory, for his generous advice on and fabrication of the custom parts of the aluminum frame.

7. REFERENCES

[1] D. W. Martin and W. Ward, “Subjective evaluation of musical scale temperament in pianos,” *J. Acoust. Soc. Amer.*, vol. 33, no. 5, pp. 582–585, May 1961.

[2] A. A. Reblitz, *Piano Servicing, Tuning, & Rebuilding: For the Professional, the Student, the Hobbyist*. Vestal Press, 1976.

[3] D. A. Gilmore, “Apparatus and method for self-tuning a piano,” Patent No. US 6,559,369B1, May 2003.

[4] “Arduino uno rev3,” <https://store.arduino.cc/arduino-uno-rev3>, accessed: 2019-04-04.

[5] J. Lattard, “Influence of inharmonicity on the tuning of a piano—Measurements and mathematical simulation,” *J. Acoust. Soc. Amer.*, vol. 94, no. 1, pp. 46–53, July 1993.

Keys	RMS deviation (cents)
A_0 to E_3	3.3
Reference octave	1.4
$F\#_3$ to C_8	7.6
All	5.7

Table 4. Average deviation between a professional tuner and the CRI process with weights from first 15 partials.

[6] H. Hinrichsen, “Entropy-based tuning of musical instruments,” *Revista brasileira de Ensino de Física*, vol. 34, no. 2, pp. 1–8, 2012.

[7] F. Rigaud, B. David, and L. Daudet, “A parametric model of piano tuning,” in *Proc. 14th Int. Conf. Digital Audio Effects (DAFx-11)*, Paris, France, Sept. 2011, pp. 393–399.

[8] T. D. Rossing, F. R. Moore, and P. A. Wheeler, *The Science of Sound*. Addison Wesley, 1990.

[9] H. Fletcher, E. D. Blackham, and R. Stratton, “Quality of piano tones,” *J. Acoust. Soc. Amer.*, vol. 34, no. 6, pp. 749–761, June 1962.

[10] A. Galemba and A. Askenfelt, “Measuring inharmonicity through pitch extraction,” *Journal STL-QPSR*, vol. 35, no. 1, pp. 135–144, 1994.

[11] ———, “Signal representation and estimation of spectral parameters by inharmonic comb filters with application to the piano,” *IEEE Trans. Speech Audio Process.*, vol. 7, no. 2, pp. 197–203, Mar. 1999.

[12] J. Rauhala and V. Välimäki, “F0 estimation of inharmonic piano tones using partial frequencies deviation method,” in *Proc. Int. Computer Music Conf. (ICMC-07)*, Copenhagen, Denmark, Aug. 2007, pp. 453–456.

[13] J. Rauhala, H.-M. Lehtonen, and V. Välimäki, “Fast automatic inharmonicity estimation algorithm,” *J. Acoust. Soc. Amer.*, vol. 121, no. 5, pp. EL184–EL189, May 2007.

[14] M. Hodgkinson, J. Wang, J. Timoney, and V. Lazzarini, “Handling inharmonic series with median-adjustive trajectories,” in *Proc. 12th Int. Conf. Digital Audio Effects (DAFx-09)*, Como, Italy, Sept. 2009, pp. 1–7.

[15] T. C. Hundley, H. Benioff, and D. W. Martin, “Factors contributing to the multiple rate of piano tone decay,” *J. Acoust. Soc. Amer.*, vol. 64, no. 5, pp. 1303–1309, Nov. 1978.

[16] J. G. Roederer, *The Physics and Psychophysics of Music: An Introduction*. Springer, 2001.

[17] M. R. Schroeder, B. S. Atal, and J. L. Hall, “Optimizing digital speech coders by exploiting masking properties of the human ear,” *J. Acoust. Soc. Amer.*, vol. 66, no. 6, pp. 1647–1652, Dec. 1979.

[18] Y. Lin and W. H. Abdulla, “Audio watermarking techniques,” in *Audio Watermark*. Springer, 2015, pp. 51–94.