Piano Tuning Method

Zuheng Kang

CONTENTS

Abstract 2
Project Location.. 2
1 Introduction 2
2 Technical Knowledge 3
2.1 Key Names 3
2.2 Key Numbers 3
2.3 Functions 3
2.4 Tuning Methodology 4
3 Piano Tuning Method 4
3.1 Traditional Method 4
3.1.1 Sampling Piano 4
3.1.2 Audio Processing 4
3.1.3 Frequency Analysis 4
3.1.4 Catchup Overtone 5
3.1.5 Inharmonicity Model 6
3.1.6 Tuning Curve Optimization Model 8
3.1.7 Temperament Model 11
3.1.8 Creating Tuning Strategy Table 12
3.2 Entropy Tuning Method 12
3.2.1 Sampling Piano & Audio Processing 13
3.2.2 Construct Spectrum 13
3.2.3 Tuning with Entropy Optimizer 13
3.2.4 Creating Tuning Strategy Table 15
3.2.5 Tune for Songs 16
4 Audio Processing & Pure Sound Tuner 17
4.1 Tuning 17
4.2 Sound Purify 17
5 Future Work 19
6 Reference 19
7 Appendix 20

ABSTRACT

Since the piano string is consider to be a stick rather than a pure ideal string, it contains stiffness and its overtone will shift in such way that make piano tuning a difficult work. In this work, two optimization algorithm for piano tuning method is presented. The traditional tuning algorithm is divided into several models that using various fitting technique model the target piano, and then convert to linear regression problem for optimization. The entropy tuning method is a trial method to tune the piano to minimize the entropy value when all key are pressed – to achieve simpler spectrum in pitch domain. In addition, a pure tuner method is invented to get rid of all inharmonic effect of piano sound.

Keyword: piano tuning, inharmonicity, entropy, audio processing

PROJECT LOCATION
Reference [2]

---

1 INTRODUCTION

Piano tuning is a difficult work since the harmonics shift that make the piano hard to tune. The tuning process will be a task to highly reduce the audible cacophonous. There are several factors we need to consider, which the rule of harmony is.

- The cacophonous created by its base frequency and audible harmonics; a good tuning will largely reduce the inharmonic effects for harmonies (the frequency domain should be simple, which the frequency peaks should merged or coincide).
- The inner music scales related pitch; the odd pitch tuning will result in the weird effect when playing music scales.

Other famous related works are:

- Tunelab (closed source; has trial version)
- Reyburn CyberTuner (closed source; no trial version)
- Entropy Piano Tuner (open source) [1]

The first two is similar, which represent the old tuning techniques, and my work mostly focus on this algorithm.

As for Entropy Piano Tuner, it represents the new way of piano tuning. It can also achieve very good result for tuning a piano, however this temperament is not regular 12-equal temperament, but a piano approximation temperament starting from 12-equal temperament, in order to largely eliminate the non-harmonious effect.

- Since the pitch in the piano does not have relatively same pitch interval, some inner scales sound weird.
- Since the piano optimize all 88 keys harmony, it values overall harmonious – some simpler chord might not sound harmonious.
- It only considers the sound which at the certain striking level of piano keys, which result in the optimization of keys are based only on sampling striking level. However, it values the average case for piano performance, thus it covers the majority situation of harmony cases.
- The accuracy cannot be too high due to large amount of calculation, it does not achieve an ideal result.

In my work, I will talk about two piano tuning methods, and one audio processing method.

- As for traditional tuning method, since it is closed source, I guessed their tuning method and create a similar solution, and will be shown in this article. Besides, I used more accurate model for inharmonicity coefficients.
- I will reproduce the result for Entropy Piano Tuning method.

- The tuning for audio and a pure sound tuner is introduced.

In this article, the first part is to introduce the technical knowledge for high level modeling algorithms. The second part is to introduce my piano modeling and tuning optimization method. Then, followed an audio processing technique. Finally, the future work will be introduced.

2 TECHNICAL KNOWLEDGE

2.1 KEY NAMES

The left most key name is defined as "A0", where "A" is the note name, 0 is the scale number. "C" is the starting point of one scale. It only allowed sharp in the note, flat is not allowed in this naming format.

A0, A#0, B0, C1, C#1, …, B1, C2, …, B7, C8

There are 88 keys for standard piano.

2.2 KEY NUMBERS

In the real world, the piano key will be labeled with numbers when the piano is open and machine part is shown off.

A0 key is labeled to be 1, and "C8" is 88.

However, in my program, "A0" key is labeled as 0 for easier calculation, which is defined as k.

2.3 FUNCTIONS

Frequency ratio to cents function:

$$\mathrm{Fr}_{\to\mathrm{c}}\left(\gamma\right)=1200\log_2\left(\gamma\right) \tag{2.1}$$

The inverse process is:

$$\mathrm{C}_{\to\mathrm{fr}}\left(c\right)=2^{\left(\frac{c}{1200}\right)} \tag{2.2}$$

Where cents is from 12 equal temperament, each half note has 100 divisions, named cents.

Frequency add cents (pitch) function:

$$\mathrm{F}_{+\mathrm{c}}\left(f,c\right)=f\cdot 2^{\left(\frac{c}{1200}\right)} \tag{2.3}$$

This function returns the frequency that added the pitch (cents) c.

The ideal frequency for the key k is:

$$\tilde{f}_k=\tilde{f}_{[A4]}\cdot 2^{\left(\frac{k-48}{12}\right)} \tag{2.4}$$

Where $\tilde{f}_{[A4]}$ is the international standard pitch for "A4", usually defined as 440Hz. Other tuning standard will replace this number, 48 is the key number for "A4".

2.4 TUNING METHODOLOGY

Since the minor tuning for each string will rarely affect its stiffness, from Equation (3.3), we assume that the B_k is the constant.

3 PIANO TUNING METHOD

3.1 TRADITIONAL METHOD

The traditional tuning method is to match the specific frequency peaks that aimed at largely eliminating the "beat" (pitch differences from two notes; for example, "A3's" second overtone matches its octave "A4", which is denoted to be 2:1). Then, use a smooth curve to optimize/minimize all the differences to achieve relatively good result.

Since the piano overtone shift (inharmonicity) has a very nice relation, it enables us to just sample very few keys and guess all the properties for all piano; then, get the tuning strategy.

3.1.1 Sampling Piano

Before tuning a piano, we need to sample a piano by recording few piano keys sound audios. This process will roughly or precisely measure the inharmonicity of piano strings (which will talk about later), such that we could model the inharmonicity for the targeted piano.

The sampling is suggested to measure keys "C1", "C2", "C3", "C4", "C5" (and probably "C6"; user could record more piano keys such as "A1" ~ "A6" for better result). Since the tuning inharmonicity curve is a smooth curve and predictable, thus it is possible to sample fewer notes. The piano key sound should be recorded in a quiet environment, which allows more accuracy for later frequency analysis. In this sampling process, we need to press the key hard in order to get higher harmonic peaks for measurement.

In my program, I use fully or almost fully sampled piano for research purposes.

3.1.2 Audio Processing

Since the real audio may contain the white space at the start or the end, and the sound length varies. I use this method to process my sampled audio:

- Normalize ($\mathrm{N}(x) = x / \max(x)$) the audio file into 1, then, find the peak volume of audio, and start from here.
- Slice these audio pieces into tiny partitions, say 0.1 second is one partition. The maximum number of each partition will be its assumed volume at this time point.
- Trim the audio at the volume start from some large number to small number – since piano sound is loud from its beginning and decay by the time. Say from 90% to 2% of the sampled sound audio.

3.1.3 Frequency Analysis

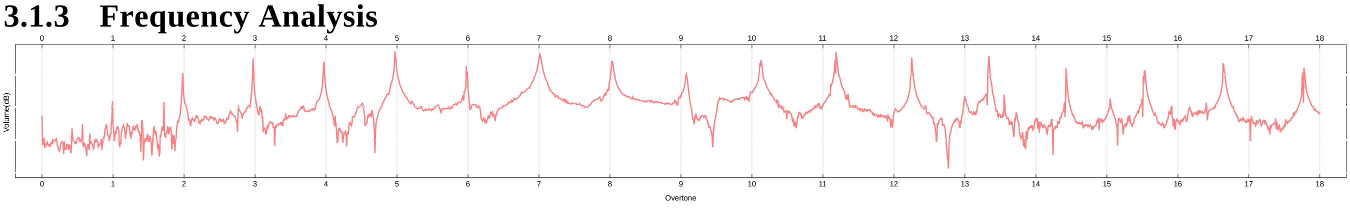

Figure 3-1 "A#0" Key (at Upright Piano Samples) Overtone Plot; Volume at Logarithm Scale

Then, put this audio samples into fourier analysis (FFT algorithm). Then we get the function $\mathrm{G}_k(f) = \left\| \mathrm{FFT}(\mathrm{S}_k(t)) \right\|_2$ where $\mathrm{S}_k(t)$ is the audio function, and $\mathrm{G}_k(f)$ is the frequency domain function, k is piano key number, f is the frequency variable, $\|\cdot\|_2$ is the 2-norm of complex numbers. In our work, the frequency domain is converted to the ratio to its ideal fundamental frequency, thus we can see the Figure 3-1, the peaks will always almost lies in the grid by dividing its ideal frequency.

From Figure 3-1, we can see that the higher overtone (right hand side peaks with larger numbers) shifts higher.

It is a problem to capture all these peaks numbers, since some are not clear: the fundamental frequency (at 1), and some has multiple peaks: at 15 ~ 16.

In my work, I use the frequency *Catchup Method* to get octave values for all these peaks.

3.1.4 Catchup Overtone

From the charactors of these peaks, there are several charactors will be considered:

- From left to right, the gap between two peaks are increasing gradually.
- The largest value of this plot is probably some peak of overtone
- The valid peak should be nearly larger than fundamental frequency position: at 1.
- The peak may be broken into several peaks, we need centralize the targeted position.

From this charactoristics, the *Catchup Method* could be built:

- Analyze the frequency samples which roughly larger than 1 (my program is starting from 0.8), get the peak frequency $f_{k,peak}$ at key number k, and overtone number $peak$.
- Comparing with ideal frequency $\tilde{f}_k$. We can then assume that it is $n = \mathrm{round}\left(f_{k,peak} / \tilde{f}_k\right)$ harmonics. Then, we can know its guessed fundamental frequency is $\hat{f}_k = f_{k,peak} / n$. Then, this should be the step size for catchup method.
- The catchup method is forward (goes to the right), and the backward (goes to the left). If we are in the forward operation, the next guessed target frequency is $\hat{f}_{k,peak+1} = f_{k,peak} + f_k'$, where f_k' is the assumed gap between two peak at this position. In the first try, we set this number to $f_k' = \hat{f}_k$, and this number will be increasing for more right harmonics. Then, we get the around data (in a relatively small area) for guessed target frequency $\hat{f}_{k,peak+1} \pm \delta$. We can find its maximum number these data to be the frequency candidate $\hat{f}_{k,peak+1}^{candidate}$, then we get the data of smaller surround area $\hat{f}_{k,peak+1}^{candidate} \pm \delta'$ where $\delta' << \delta$. Then, we calculate the weighted average for this smaller area, and the result is the actual frequency of this peak $f_{k,peak+1} = \int_{\hat{f}-\delta'}^{\hat{f}+\delta'} \omega \cdot \mathrm{G}(\omega)\, \mathrm{d}\omega$, where ω is proportional to frequency. Then, the assumed gap between two peak at this step is updated to be $f_k' = f_{k,peak+1} - f_{k,peak}$.
- Iterate this method for forward catchup to get all higher frequencies.
- If the highest peak is not fundamental frequency, we will perform the backward catchup. Since there are less peaks and the overtone shift will be far less than the right, the assumed targeted gap between two peaks is set to be the assumed fundamental frequency $\hat{f}_k$.

From this method, we can get a overtone (frequency) list for the key k. Which is:

$$k \to \left\{ f_{k,1}, f_{k,2}, \ldots \right\} \tag{3.1}$$

3.1.5 Inharmonicity Model

From Figure 3-1, we can see that the overtone will shift higher and higher as the frequency goes higher. This effect is caused by the stiffness of an object, its natural frequency will follow a certain pattern.

From reference [1], we assume that the piano string is a bar with two fixed ends, which approximately follows the partial differential eqution:

$$\ddot{y} \propto -y'' - \varepsilon y'''' \tag{3.2}$$

Where y is the special position of piano string (bar model). The prime is the derivative to spatial domain, and dots is the derivative to time domain.

Then, use the modal analysis and solved the natural frequencies for this string are:

$$f_{k,n} \propto n \cdot f_{k,1}\sqrt{1+B_k \cdot n^2} \Rightarrow f_{k,n} = A_k \cdot n \cdot f_{k,1}\sqrt{1+B_k \cdot n^2} \tag{3.3}$$

Here we have two unknown variables A_k and B_k.

Then, we use this function to fit all frequency results at Equation (3.1). The parameter A_k is set since not all fundamental frequency is guessing perfectly. We can ignore this number by making sure the fundamental frequency always target on 1, and focus only on B_k.

Then, we can get inharmonicity parameter list $\{\{k, B_k\}\}$.

From my observation, the logarithm of this number has some beautiful properties with the data $\{\{k, \ln(s \cdot B_k)\}\}$, where s is a scaling parameter (I set to 10000).

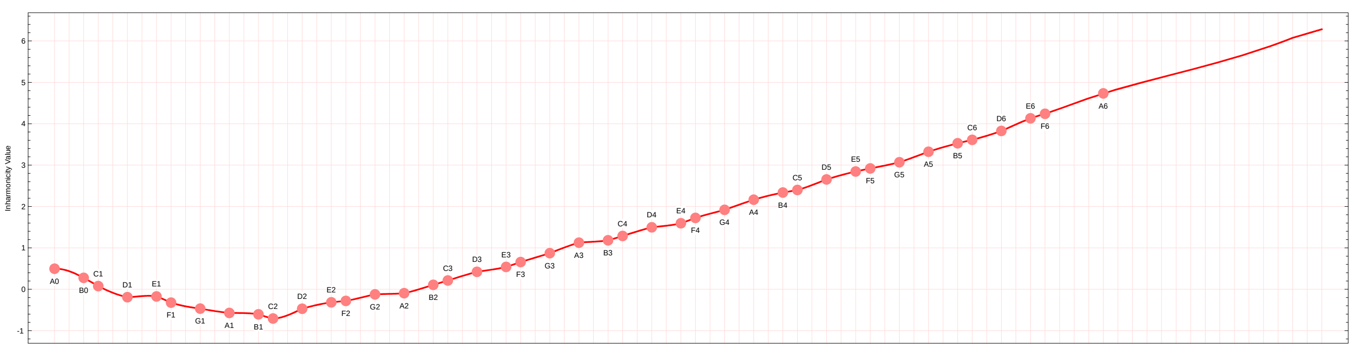

Figure 3-2 Inharmonicity Plot of Grand Piano IH(*k*)

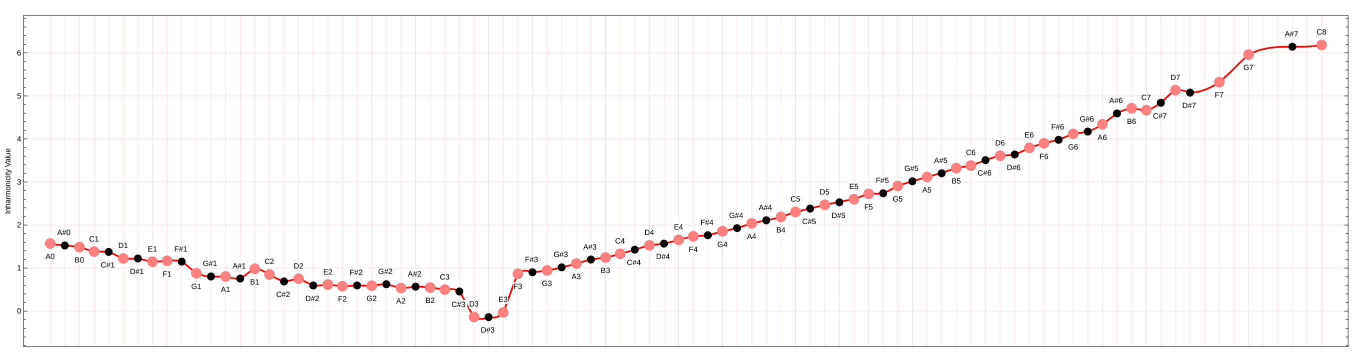

Figure 3-3 Inharmonicity Plot of Upright Piano IH(*k*)

From Figure 3-2 and Figure 3-3, we can clearly see the line is divided into 2 parts.

Figure 3-4 Grand Piano String Arrangement

Figure 3-5 Upright Piano String Arrangement

From Figure 3-4 and Figure 3-5, we can clearly see that the string is divided into two parts, the steel string and copper string (may be covered by silver for highly expensive pianos). The upright piano has more copper strings since the steel string cannot goes longer, and the string will become thicker to make the string vibrate slower. From spring vibration formula:

$$\omega = \sqrt{\frac{K}{m}} \tag{3.4}$$

Where ω is proportional to frequency, m is the mass of spring, K is the stiffness of spring.

When m increases, K increase a little bit, ω decreases, then frequency decrease.

Since the piano cannot growing longer, it become thick and more like a stick rather than an ideal string. For higher notes strings, it is too short, and the thickness become relatively larger comparing to its length, thus it is more likely to be a bar.

Thus, from the plot, we can see the inharmonicity increases at two ends, and break at the position of separation of two kinds of strings.

Since grand concert piano is longer, and can have more steel strings, less copper strings, thus the break will become more left side.

The figure of inharmonicity plot also tell us that two separate line are almost linear. In my model, I used the valid sampled points are modeled with interpolation function, and two edges are modeled with linear function, and it is method is shown below.

- We get several samples from one line, and fit in a linear form.
- Get its slope, and build a line which pass the right end point (since I will not wish to have a break for the interpolation function), and add some samples for edges situation to sample pool.
- Similar to the left hand side.
- We use interpolation for these samples of sample pool – "left hand side + samples + right hand side", which is our final model for inharmonicity model function $\mathrm{IH}(k)$.

$$\mathrm{IH}(k) = \ln(s \cdot B_k) \tag{3.5}$$

Thus, we can have the modeled parameter B_k with:

$$B_k = \frac{\mathrm{e}^{\mathrm{IH}(k)}}{s} \tag{3.6}$$

Then, the frequencies $\tau(k,n)$ will be:

$$\tau(k,n) = f_{k,1} \cdot n \cdot \sqrt{\frac{1 + B_k \cdot n^2}{1 + B_k}} \tag{3.7}$$

Where $f_{k,1}$ is currently unknown but it will be eliminated, since it is in frequency ratio form. In this equation, we divide a term $\sqrt{1+B_k}$ to make sure the fundamental frequency is $f_{k,1}$.

3.1.6 Tuning Curve Optimization Model

Similar to Tunelab ®, I set the tuning optimization method to separate the lower tones (bass) and higher tones (tenor) into two tuning target optimization method, the separation point k_0 is "C#4/D4". And the default tuning method for bass is to set 6:3. Since 6/3=2 (a/b), this frequency ratio is $\gamma = a/b$, and its corresponding pitch range is $\mathrm{Fr}_{\to c}(\gamma)$ which is 1200, and 1200 is an octave, it means the tone say "A0"s 6th harmonics will largely match its octave's "A1"s 3rd harmonics.

Here pitch is defined by cents.

The error function ε_k is defined as:

$$\begin{aligned}\varepsilon_k &= \mathrm{Fr}_{\to \mathrm{c}}\left(\frac{\tau(k,a)}{\tau(k+Fr_{\to c}(a/b),b)}\right)\\ &= \mathrm{Fr}_{\to \mathrm{c}}\left(\sqrt{\frac{\left(1+B_k\cdot a^2\right)\cdot\left(1+B_{k+Fr_{\to c}(a/b)}\right)}{\left(1+B_{k+Fr_{\to c}(a/b)}\cdot b^2\right)\cdot\left(1+B_k\right)}}\cdot\frac{a}{b}\cdot\left(\frac{f_{k,1}}{f_{k+Fr_{\to c}(a/b),1}}\right)\right)\\ &= \mathrm{Fr}_{\to \mathrm{c}}\left(\sqrt{\frac{\left(1+B_k\cdot a^2\right)\cdot\left(1+B_{k+Fr_{\to c}(a/b)}\right)}{\left(1+B_{k+Fr_{\to c}(a/b)}\cdot b^2\right)\cdot\left(1+B_k\right)}}\right)\end{aligned} \tag{3.8}$$

We can do this for all bass strings.

For tenor strings, the default tuning method is set to 4:1 (c/d). But this time we count the higher note as the target to calculate.

$$\varepsilon_k = \mathrm{Fr}_{\to \mathrm{c}}\left(\sqrt{\frac{\left(1+B_{k-Fr_{\to c}(c/d)}\cdot c^2\right)\cdot\left(1+B_k\right)}{\left(1+B_k\cdot d^2\right)\cdot\left(1+B_{k-Fr_{\to c}(c/d)}\right)}}\right) \tag{3.9}$$

The combined expression is:

$$\mathrm{E}(k)=\begin{cases}\mathrm{Fr}_{\to \mathrm{c}}\left(\sqrt{\frac{\left(1+B_k\cdot a^2\right)\cdot\left(1+B_{k+Fr_{\to c}(a/b)}\right)}{\left(1+B_{k+Fr_{\to c}(a/b)}\cdot b^2\right)\cdot\left(1+B_k\right)}}\right) & k\le k_0\\ \mathrm{Fr}_{\to \mathrm{c}}\left(\sqrt{\frac{\left(1+B_{k-Fr_{\to c}(c/d)}\cdot c^2\right)\cdot\left(1+B_k\right)}{\left(1+B_k\cdot d^2\right)\cdot\left(1+B_{k-Fr_{\to c}(c/d)}\right)}}\right) & k>k_0\end{cases} \tag{3.10}$$

From this equation, we can see $\mathrm{E}(k)$ is only a value for calculation at given k .

From this point, we need a function to largely eliminate these errors. The piano tuning curve $\mathrm{C}(k)$ is introduced, it represent the deviation of the actual tuning pitch to the ideal 12-equal temperament pitch.

The optimizer deviation function $\mathrm{D}(k)$ is:

$$\mathrm{D}(k)=\mathrm{C}(k)-\mathrm{E}(k) \tag{3.11}$$

The cost function $\mathrm{J}(k)$ for optimization is:

$$\mathrm{J}(k)=\sum_k\left(\mathrm{D}(k)\right)^2 \tag{3.12}$$

Which minimize the square error of these functions.

Here I use polynomial for easier calculation:

$$\mathrm{C}(x) = \sum_{i=1}^{n} \chi_i \cdot x^i \tag{3.13}$$

Since $\mathrm{C}(x)$ will pass the fix point, which is "A4" pitch at 440Hz frequency at pitch deviation of 0, thus i is from 1 and $x = k - k_{[A4]}$, where $k_{[A4]}$ is the key number (index) at "A4", which is 48.

Thus, $\mathrm{J}(k)$ is the second order multi-variable polynomial function, which is very easy to minimize by linear regression method to calculate the fitting parameter $\{\chi_i\}$, and rebuild the functions.

Then, we can bring $\{\chi_i\}$ to the $\mathrm{D}(k)$ function to calculate its deviations.

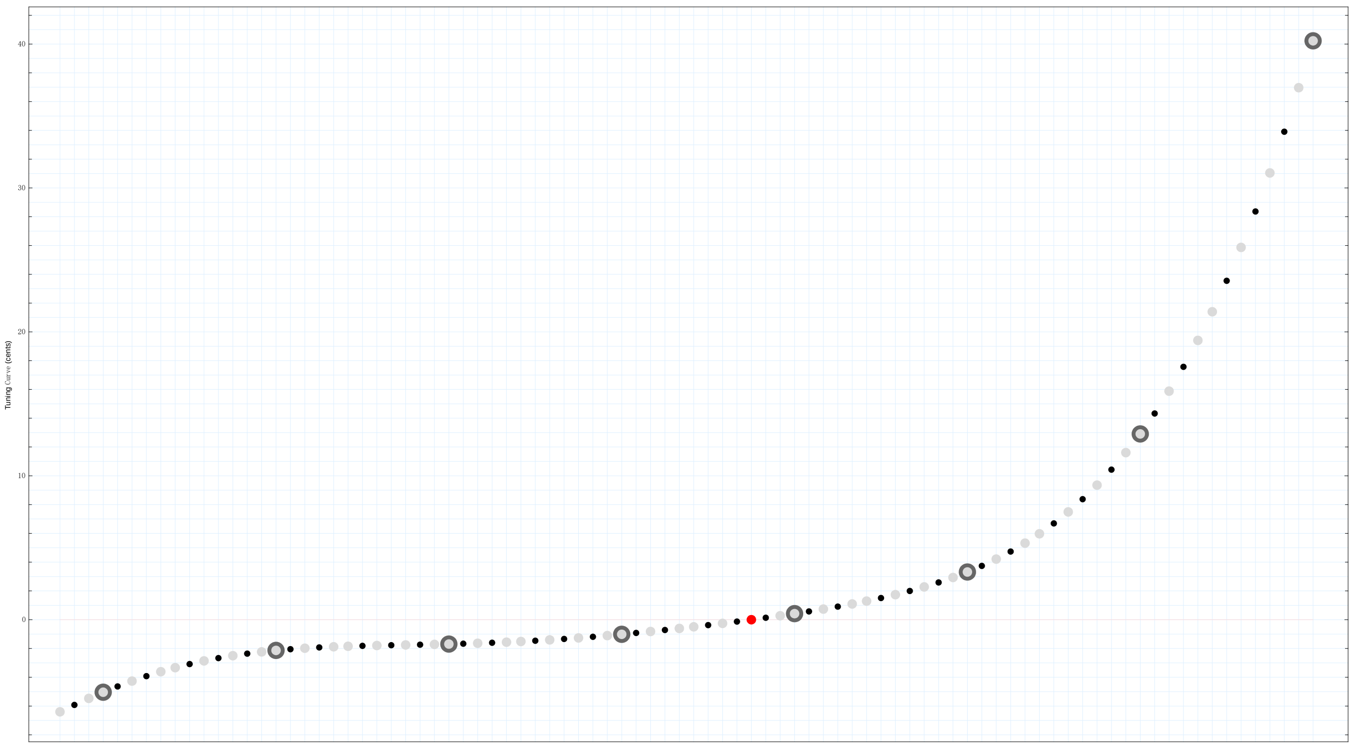

Figure 3-6 C(*k*) for Grand Piano

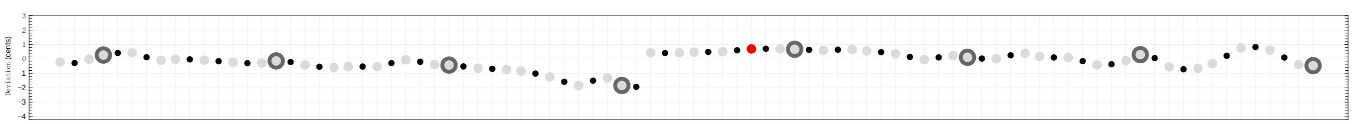

Figure 3-7 D(*k*) for Grand Piano

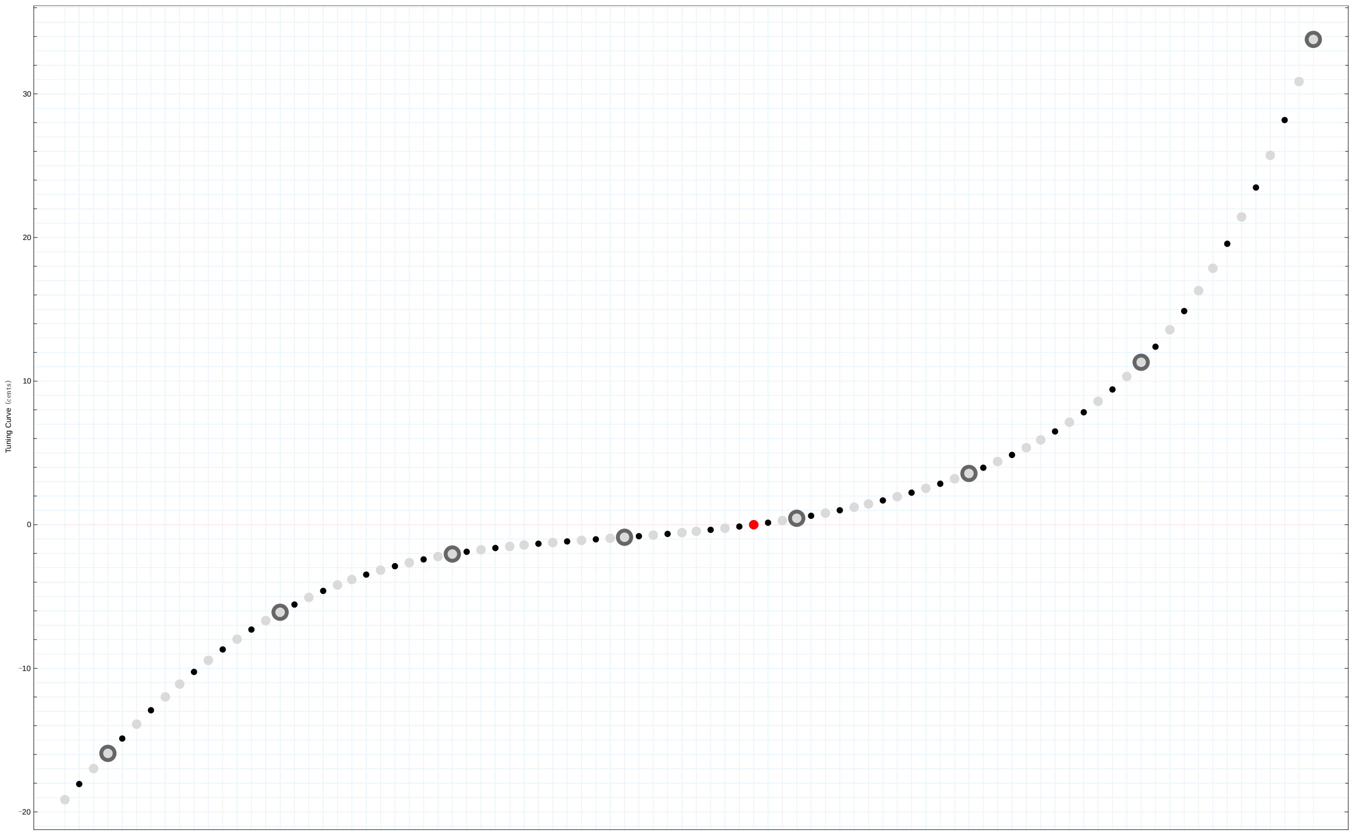

Figure 3-8 C(*k*) for Upright Piano

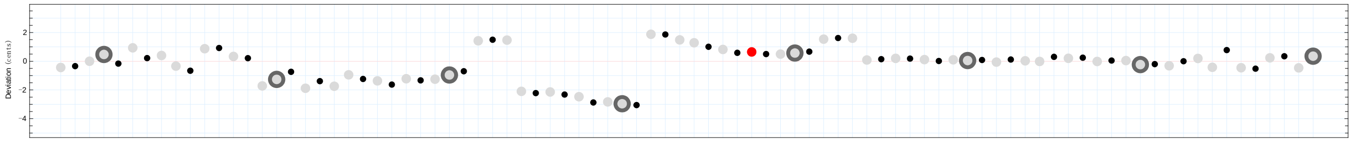

Figure 3-9 D(*k*) for Upright Piano

The result of two piano is shown above. Horizontal axis is the key number, and the vertical axis the pitch interval with its ideal frequencies represented by cents.

From this tuning method, we can see that the bass tuning will consider the deviations from the tenor part, and vice versa. The effect is inner related. Thus this tuning method is theoretically to optimize almost the whole piano keys tuning.

3.1.7 Temperament Model

With the development of music, various temperament appears and create unique flavor of music. The temperament model is using the pitch deviation tables of different temperament (the unit is cent). We can then create the non-12 equal temperament tuning strategy. The temperament function is defined to be $\mathrm{T}(k)$.

The tuning table such as "Bach - Bradley Lehman" is:

C	C#	D	D#	E	F	F#	G	G#	A	A#	B
5.87	3.91	1.96	3.91	-1.96	7.82	1.96	3.91	3.81	0	3.91	0

Table 3-1 Table for "Bach - Bradley Lehman" Temperament

Where A note will always be 0 since A is the reference frequency and will always keep to 440 Hz (if is standard situation).

This table shows the situation of "C" major.

The other major tuning will follow the rotation of table. For example: if tuning "D" major, the "D" will rotate to current "D" →"C" place, which is rotating left 2 times. However, we will make sure "A" note will always be 0, then, we can subtract the number at "B" → "A" to make it possible.

Then, add these pitch errors to all the notes of tuning, the modified tuning curve is:

$$\mathrm{C}'(k)=\mathrm{C}(k)+\mathrm{T}(k) \tag{3.14}$$

3.1.8 Creating Tuning Strategy Table

The final tuning strategy $\tau(k,n)$ (unit: Hz) is:

$$f_{k,1}=F_{+c}\left(\tilde{f}_k,\mathrm{C}'(k)\right) \tag{3.15}$$

$$\begin{aligned}\tau(k,n)&=f_{k,1}\cdot n\cdot\sqrt{\frac{1+B_k\cdot n^2}{1+B_k}}f\\&=\mathrm{F}_{+\mathrm{c}}\left(\tilde{f}_k,\mathrm{C}'(k)\right)\cdot n\cdot\sqrt{\frac{1+B_k\cdot n^2}{1+B_k}}\\&=\mathrm{F}_{+\mathrm{c}}\left(\tilde{f}_k,\mathrm{C}'(k)\right)\cdot n\cdot\sqrt{\frac{s+\mathrm{e}^{\mathrm{IH}(k)}\cdot n^2}{s+\mathrm{e}^{\mathrm{IH}(k)}}}\end{aligned} \tag{3.16}$$

From Equation (3.16), we can see only $\mathrm{C}(\cdot)$ and $\mathrm{IH}(\cdot)$ function is modeled function, other function are basic mathematics functions.

From the modeling, we can get a strategy of piano tuning, then we can convert this strategy into a tuning table, which shows all the frequency of fundamental and its harmonics frequencies, and corresponding deviation to ideal frequencies represented by cents.

The grand and upright piano tuning strategy is shown in Figure 7-1 and Figure 7-2.

The red font is the frequencies recommended for the devices to tune.

3.2 ENTROPY TUNING METHOD

Entropy tuning method is not to model the exact value of frequencies or pitches, it simulates the condition that simultaneously press down all piano keys, and uses entropy method as cost function to largely merge the peaks at pitch domain to create more sharp and simple sound for piano, which optimize the piano sound. The method is extremely simple, however, it is really computational intensive.

Why simulate pressing all keys? We need to know the philosophy of piano in behind. To deal with all kinds of complicated situations, let us assume several cases. Whether the chord is harmonious is to check the transient pitch domain. In other word, several notes at certain short time period will contact with each other, and we need to make sure this sound is harmonious. However, the contact cases of notes at all time for all songs are too complicated, and the key pressing level varies all the time. What if assuming that all notes has equal probability to contact, and the key pressing level when playing each small pieces of music in average is the same – some pieces are loud, some are small but they usually approximately on the same level when playing the piano. As for the key pressing level which change the sound quality, we suggest the sample sound will be played in medium level.

3.2.1 Sampling Piano & Audio Processing

In entropy piano tuning method, sampling every piano key is necessary. Other requirement is similar to traditional method. The audio processing is also similar to traditional method.

3.2.2 Construct Spectrum

Since human ear is sensitive to the pitch ("pitch" is equivalent to the logarithm of frequency component for approximation: ignore non-linear effect of ear structures) within the hearing range (20Hz ~ 10000Hz is reasonable for optimizing algorithm). Thus, the model should be built by putting equal significance to the pitch scale. Traditionally, the pitch is represented as music note. If we evaluate the "pitch" content/data by equally sampling from the pitch scale of spectrum, it put the equal importance to the pitch scale – logarithm scale of frequencies. In my experiment, I put 0.1 cent as the precision.

Then, we have the converted the spectrum into pitch domain $\mathrm{I}(\kappa)$, to resample the data with the key number:

$$\mathrm{I}(\kappa) = \left\| \mathrm{G}\left(f_\kappa \right) \right\|^\beta \Big|_{\kappa \to 12 \cdot \log_2\left(\frac{f_\kappa}{\tilde{f}_{[A0]}} \right), \beta \to 2} \tag{3.17}$$

Where for each key k we will have 1000 samples in total, each sample pitch denote as κ. Namely, each sample will represent 0.1 cent. Since the audio is also the limited samples, I use the interpolation function to resample the data.

In this model, I use the square of spectrum $\beta = 2$. The reason is that: although human ear sensitive to the sound pressure level is based on logarithm of magnitude of sound, unit could be decibel (dB), however human ear also has the auditory mask, which mask small peaks around it, thus we should value more on major peaks, and ignore minor one. From the paper [1], and my trial and error, the square is actually achieve very ideal result. I also tried other numbers for β, when $\beta = 1$, the sound is messy at all; $\beta = 2$ is perfect; β is larger, the simpler sound will hear more harmonious, however the complicated chord may not hear well since the algorithm may value more on merging major peaks of spectrum and ignore the little ones. If people need to play more simple chord songs, they may try larger numbers of β, if need to play more messy types songs like Impressionist or Jazz, I suggest they will use smaller β. On average, 2 is great number for β.

Since for each key sound, the first peak of spectrum should start from its fundamental frequency, thus, we will set it 0 to ignore these noise.

3.2.3 Tuning with Entropy Optimizer

The tuning process from programming point of view is to move left or right of array $\mathrm{I}(\cdot)$ as minor tuning process with $+c$ cent shift.

$$\mathrm{I}_k\left(\kappa - c \right) = \left\| \mathrm{G}\left(f_{\kappa - c} \right) \right\|^\beta \tag{3.18}$$

The entropy function is defined as:

$$\mathrm{Entropy}(x) = -x \cdot \log(x) \tag{3.19}$$

Entropy for a function is defined as:

$$\begin{aligned}\mathrm{Entropy}\left(\phi(x)\right) &= \int_{-\infty}^{+\infty}\left(-\phi(x)\cdot\log\left(\phi(x)\right)\right)\mathrm{d}\,x \\ &= \sum_{x}\left(-\phi(x)\cdot\log\left(\phi(x)\right)\right)\end{aligned} \tag{3.20}$$

Where $\phi(\cdot)$ is the density function:

$$\begin{aligned}1 &= \int_{-\infty}^{+\infty}\phi(x)\,\mathrm{d}\,x \\ &= \sum_{x}\phi(x)\end{aligned} \tag{3.21}$$

3.2.3.1 How to calculate entropy value for optimizer.

Since the algorithm optimize the case that all sound volume is equal, however the sampling time are different, we will make a standard case to simulate all keys are pressed in an equal key pressing level. In my program, I use density function $\overline{\mathrm{I}}_k(\kappa)$ to simulate the equal key pressing level for each piano key sound in pitch domain:

$$\overline{\mathrm{I}}_k(\kappa) = \frac{\mathrm{I}_k(\kappa)}{\sum_{\kappa}\left(\mathrm{I}_k(\kappa)\right)} \tag{3.22}$$

When press all piano keys, the total volume $\mathrm{V}(\kappa)$ for each key pitch shift $+c_k$ cents for tuning is:

$$\mathrm{V}(\kappa) = \sum_{k}\left(\overline{\mathrm{I}}_k(\kappa - c_k)\right) \tag{3.23}$$

The density function for this function is:

$$\overline{\mathrm{V}}(\kappa) = \frac{\mathrm{V}(\kappa)}{\sum_{\kappa}\left(\mathrm{V}(\kappa)\right)} \tag{3.24}$$

Then, the cost function value J (entropy value for function $\overline{\mathrm{V}}(\kappa)$) is:

$$\mathrm{J} = \sum_{\kappa}\left(-\overline{\mathrm{V}}(\kappa)\cdot\log\left(\overline{\mathrm{V}}(\kappa)\right)\right) \tag{3.25}$$

3.2.3.2 Steps to calculate tuning strategy

In my program, there are several steps to dig out the good strategy for tuning.

Step 1: Calculate the traditional tuning strategy which is simpler version of Traditional Tuning strategy, to be the initial starting point for entropy minimizer to begin. In this algorithm, no inharmonicity model is built, but just use the captured frequency to optimize.

Step 2: Randomly change tuning for one key for c_k cents, and check its entropy value. If entropy value is smaller than last time, we keep this tuning strategy, otherwise, drop. Where the changing pitch is defined as a random number between 0 to some small number p. We will try both side of tuning by adding and subtracting the pitches. The "A4" key never change since it is standard pitch.

Step 3: We do "step 2" experiment for all keys and all directions as one round of experiment. Each time we count the times of successfully tuning, until we cannot find a round with no improvement.

Step 4: We stop the algorithm with the test for p precision. Then we shrink the p and more accurate spectrum data (more data), and calculate "Step 2" and "Step 3"

Step 5: Calculate tuning strategy and get report.

In this process, "Step 1" is because the algorithm has many local minimums; although some local minimum can achieve similar simple and sharp harmony, it perform badly in simpler harmonies, such as an octave. A traditional tuning method can roughly optimize major overtones, the best result for entropy minimizer should be around the traditional tuning strategy.

In "Step 2", although there should be more improvement during this step, however from probability point of view, when it stops, the result is good enough for this precision. It could also use the parallel algorithm. In my program, I modeled several CPUs (not GPU program this time: GPU should calculate array sum much faster) with one shared memory to modify the result altogether. Although all CPUs will affect the overall result, however, if we can understand it will stop at the point that several CPUs could not find improvement, the effect are the same.

In "Step 4", my program uses 3 round with 1, 0.5 and 0.2 cent boundaries as step size for entropy minimizers. Since there are many local minimums, and we need to achieve a smooth tuning strategy for not creating weird music scale sound, we cannot set the step size to be really large. Thus, 1 cent boundary is a good point to start. The, next two round is accurate tuning, the accuracy will be increased to 0.1 cent, which is desirable.

In "Step 5", the frequency peaks frequencies $f_{k,n}$ are captured also by "catchup method", but without weighted average.

3.2.4 Creating Tuning Strategy Table

The method to get the frequencies components for each key sound is simple:

$$\tau'(k,n) = f_{k,n} \cdot \mathrm{C}_{\to \mathrm{fr}}(c_k) \tag{3.26}$$

However, this process is problematic. Since the whole process is based on pitch shift with certain precision, the "A4" standard frequency will not be the fix number. Here we need to eliminate this tuning error by introducing a correction factor $\varepsilon_{[A4]}$:

$$\varepsilon_{[A4]} = \frac{\tau'([A4],1)}{\tilde{f}_{[A4]}} \tag{3.27}$$

Thus, the tuning strategy $\tau(k,n)$ is modified to be:

$$\tau(k,n) = f_{k,n} \cdot \mathrm{C}_{\to \mathrm{fr}}(c_k) \cdot \varepsilon_{[A4]} \tag{3.28}$$

To build the tuning curve, the pitch deviation to the ideal frequency function $\mathrm{C}(k)$ is shown:

$$\mathrm{C}(k) = \mathrm{Fr}_{\to c}\left(\frac{\tau(k,n)}{\tilde{f}_k}\right) \tag{3.29}$$

The tuning strategy is shown in Figure 7-3.

The tuning curve is shown in Figure 3-10, the spectrum of optimized result is shown in Figure 3-11:

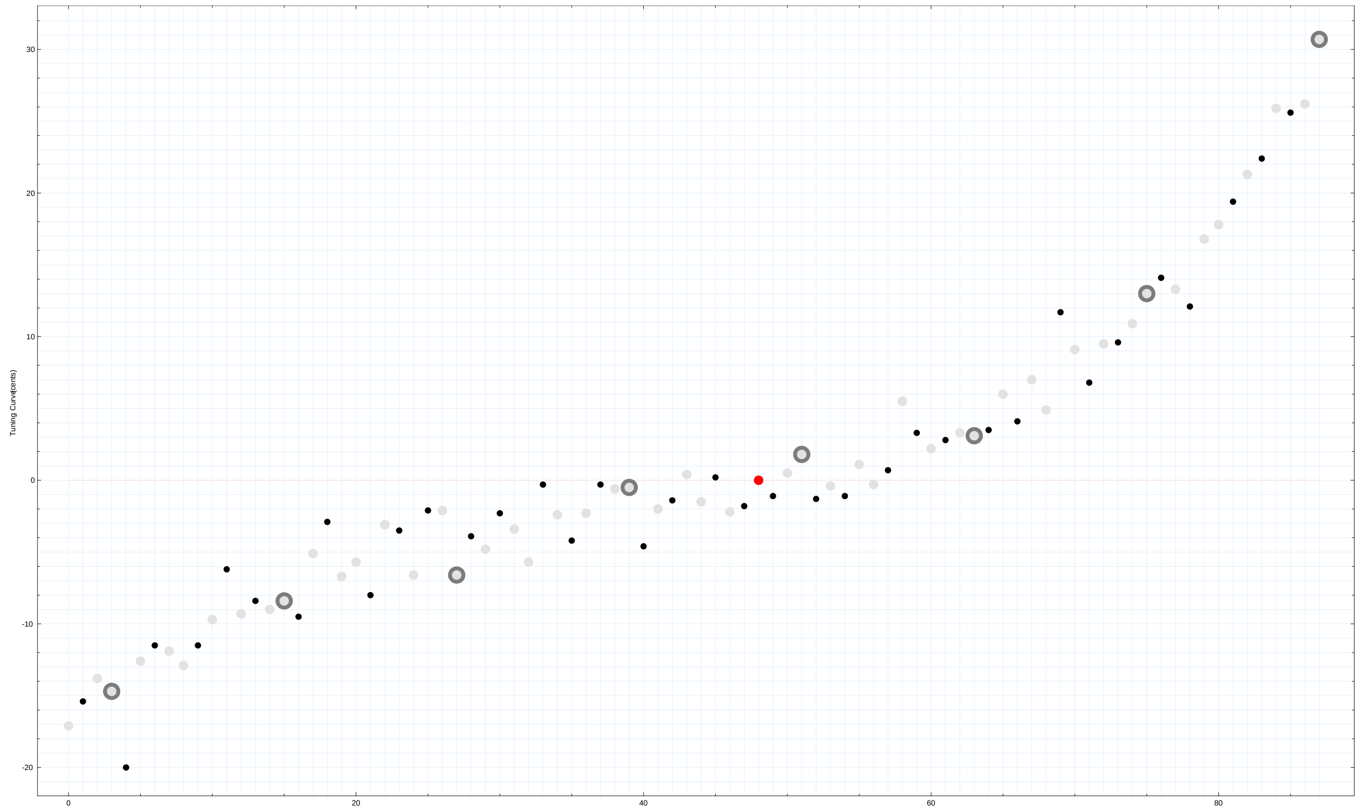

Figure 3-10 Tuning Curve for Upright Piano Optimized by Entropy Minimizer

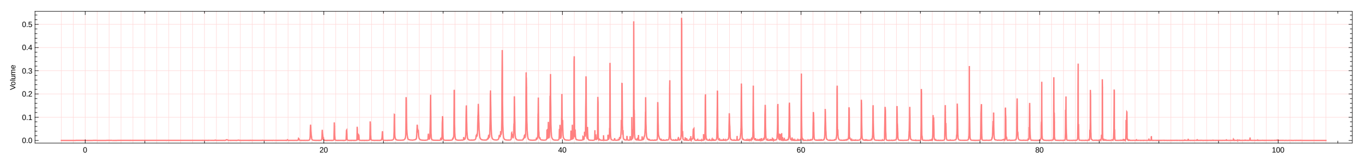

Figure 3-11 Spectrum for Optimized Result

From Figure 3-11, we could see the spectrum are largely merged. From sound quality point of view, the harmony will sound sharp and clear.

3.2.5 Tune for Songs

In the real world, some of the piano keys are not been used, especially for the simpler tonal music. Since I have mentioned the previous entropy minimizer is not quite suitable for simpler harmony music due to some of the simple harmony like octave sometimes will not sound perfect, we should ignore the keys that have not been used. Thus, I add another coefficient for the entropy minimizer.

We will put the bias Bias_k that will ignore the key k which is not been used.

$$\mathrm{Bias}_k = \begin{cases} 1 & k \in used \\ \varepsilon_{\mathrm{Bias}} & k \notin used \end{cases} \tag{3.30}$$

Where $\varepsilon_{\mathrm{Bias}}$ is a very small number – to make sure the key which is not used could be tuned by the entropy minimizer. If the bias for one key is 0, there are no spectrum for entropy minimizer for this key, and algorithm will stop tuning for this key. However, if we put a very small number as weight on this key, it still can tuned to a correct place – it just tuned, but does not affect the tuning for other keys.

Then, we will put the bias on the entropy minimizer algorithm and modify the Equation (3.25):

$$J = \sum_{\kappa} \left(-\mathrm{Bias}_{\kappa} \cdot \overline{V}(\kappa) \cdot \log\left(\overline{V}(\kappa)\right) \right) \tag{3.31}$$

Then, we use the method above to minimize this entropy function, and get the tuning strategy.

From the example of a tonal music from Mozart (Figure 3-12), we could see only the middle range and several low range keys are used.

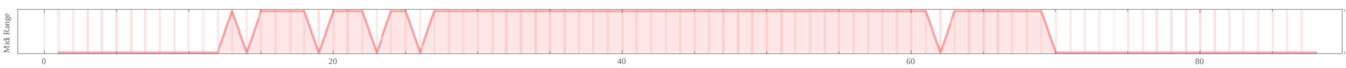

Figure 3-12 Song Key Used Cases

The optimized spectrum is shown in Figure 3-13.

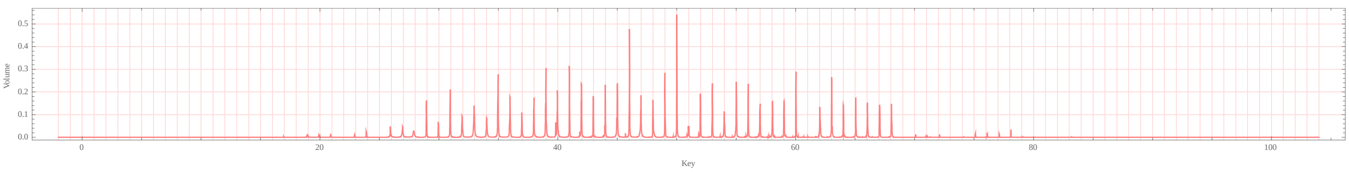

Figure 3-13 Optimized Spectrum

From this example, we can see and hear, the sound will be more optimized whenever in simple and the complicated harmonious.

4 AUDIO PROCESSING & PURE SOUND TUNER

4.1 TUNING

Tuning process for an audio is to create samples for virtual instrument so that we can hear the tuning result before tuning process to make a decision whether to adopt or drop this tuning strategy.

The sound function $S(t)$ tunes in order to add pitch c cents:

$$S_{+c}(t) = S\left(t \cdot 2^{\left(\frac{c}{1200}\right)} \right) \tag{4.1}$$

The $S(t)$ function is modeled as interpolation function.

4.2 SOUND PURIFY

This audio processing technique is invented by myself. It removes the inharmonic effect of piano sound.

Since the inharmonicity model has been built, it is possible to use audio processing technique to shrink the harmonics in order to remove the inharmonicity.

If the key k sound with the inharmonicity coefficient $\mathrm{IH}(k)$ and tuned to the fundamental frequency to be the frequency (ideal frequency) $\tilde{f}_k$; the f_k is the fundamental frequency.

We firstly get the FFT of the audio sample with $\Gamma_k(f)$ of complex number samples:

$$\Gamma_k(f) = \mathrm{FFT}\left(\mathrm{S}_k(t)\right) \tag{4.2}$$

Since the FFT is creating an almost symmetry data from the middle, we can extract this data into 4 parts: the real head data $\Gamma_k^{(0)}(f)$, the imaginary head data $\Gamma_k^{(1)}(f)$, the real tail reverse data $\Gamma_k^{(2)}(f)$ and the tail imaginary reverse data $\Gamma_k^{(3)}(f)$. Four of them looks similar, however it contains all the details of the sound. Since it samples the piano keys, the spectrum is pretty obvious. At its high frequencies, it is almost 0, and it is almost out of hearing range, thus if we need to compress the frequency domain, as for higher frequencies, we could regard it to be 0. For each component we write it as $\Gamma_k^{(m)}(f)$, where m is from 0 to 3 (4 cases), i is the unit imaginary number.

$$\Gamma_k(f) = \left\{\Gamma_k^{(0)}(f), \mathrm{rev}\left(\Gamma_k^{(2)}(f)\right)\right\} + \left\{\Gamma_k^{(1)}(f), \mathrm{rev}\left(\Gamma_k^{(3)}(f)\right)\right\} \cdot \mathrm{i} \tag{4.3}$$

From Equation (3.6) and Equation (3.7), we could get the compression functions, which is $\tau(k,n)$. Here the overtone is continuous, which is f / f_k, rather than n. Thus, we have the compressed frequency scaler $\overleftrightarrow{f}_k$ and its pitch component $\overleftrightarrow{\Gamma}_k^{(m)}(f)$:

$$\overleftrightarrow{f}_k = \tilde{f}_k \cdot \tau\left(k, \frac{f}{f_k}\right) \tag{4.4}$$

$$\overleftrightarrow{\Gamma}_k^{(m)}(f) = \begin{cases} \Gamma_k^{(m)}\left(\overleftrightarrow{f}_k\right) & \overleftrightarrow{f}_k \in \mathit{defined} \\ 0 & \overleftrightarrow{f}_k \notin \mathit{defined} \end{cases} \tag{4.5}$$

Where $\Gamma_k^{(m)}(f)$ and $\overleftrightarrow{\Gamma}_k^{(m)}(f)$ will be same size of samples.

Use the interpolation function to stretch, and do this for four functions; then, combine them in original way, and use inverse Fourier function to restore the audio $\overleftrightarrow{\mathrm{S}}_k(t)$.

$$\overleftrightarrow{\Gamma}_k(f) = \left\{\overleftrightarrow{\Gamma}_k^{(0)}(f), \mathrm{rev}\left(\overleftrightarrow{\Gamma}_k^{(2)}(f)\right)\right\} + \left\{\overleftrightarrow{\Gamma}_k^{(1)}(f), \mathrm{rev}\left(\overleftrightarrow{\Gamma}_k^{(3)}(f)\right)\right\} \cdot \mathrm{i} \tag{4.6}$$

$$\overleftrightarrow{\mathrm{S}}_k(t) = \mathrm{Re}\left(\mathrm{invFFT}\left(\overleftrightarrow{\Gamma}_k(f)\right)\right) \tag{4.7}$$

Where i is imaginary number, $\mathrm{invFFT}(\cdot)$ is the inverse FFT, $\mathrm{Re}(\cdot)$ is to get the real part of a number or array, $\mathrm{rev}(\cdot)$ is the reverse of an array.

Then, do this for 2 channels and create the audio as Pure Sound Tuner result.

From this function, it needs 3 data: the audio data $\mathrm{S}_k(t)$, the inharmonicity coefficient $\mathrm{IH}(k)$, and its fundamental frequency f_k (which could be captured by audio data).

5 FUTURE WORK

Over-pull tuning is implemented in some tuning apps, and I do not know its method. Since I am still lack of research on this area, I will leave it as future work to think about. I know this effect is caused by the experimental result of the percentage that the tuning pins will loosen and drop the pitch, it should have the correction coefficient for the tuner will make up the errors of this effect by over pull to tune the frequency higher than its actual one.

6 REFERENCE

[1] Hinrichsen, Haye. "Entropy-based tuning of musical instruments." Revista brasileira de Ensino de Física 34.2 (2012): 1-8.

[2] Github for Piano Tuning Project [https://github.com/RobertBoganKang/piano_tuning]

7 APPENDIX

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
A0	27.3985 -6.39?	54.8092 -6.01?	82.244 -5.38?	109.715 -4.48?	137.235 -3.34?	164.815 -1.94?	192.467 -0.29?	220.203 +1.6?	248.036 +3.75?	275.977 +6.14?	304.037 +8.78?	332.228 +11.66?	360.563 +14.77?	389.052 +18.13?	417.706 +21.72?	446.537 +25.54?
A#0	29.0357 -5.92?	58.0836 -5.56?	87.1556 -4.96?	116.264 -4.12?	145.42 -3.04?	174.637 -1.72?	203.927 -0.16?	233.3 +1.63?	262.77 +3.65?	292.347 +5.91?	322.044 +8.39?	351.873 +11.11?	381.844 +14.05?	411.97 +17.22?	442.261 +20.61?	472.729 +24.22?
B0	30.7704 -5.46?	61.5517 -5.16?	92.3548 -4.64?	123.191 -3.93?	154.07 -3.01?	185.005 -1.88?	216.004 -0.55?	247.08 +0.97?	278.243 +2.7?	309.503 +4.63?	340.872 +6.76?	372.359 +9.08?	403.976 +11.6?	435.733 +14.31?	467.64 +17.21?	499.708 +20.3?
C1	32.6082 -5.03?	65.2258 -4.78?	97.8625 -4.36?	130.528 -3.77?	163.231 -3.02?	195.981 -2.09?	228.789 -1.01?	261.663 +0.25?	294.612 +1.67?	327.646 +3.25?	360.775 +5.?	394.007 +6.91?	427.352 +8.98?	460.819 +11.22?	494.417 +13.61?	528.155 +16.16?
C#1	34.5552 -4.63?	69.1189 -4.42?	103.7 -4.06?	138.306 -3.56?	172.947 -2.91?	207.631 -2.13?	242.366 -1.2?	277.161 +0.13?	312.024 +1.08?	346.964 +2.43?	381.989 +3.92?	417.108 +5.55?	452.328 +7.32?	487.659 +9.22?	523.108 +11.26?	558.684 +13.44?
D1	36.6178 -4.26?	73.2437 -4.07?	109.886 -3.75?	146.553 -3.3?	183.252 -2.72?	219.992 -2.01?	256.781 -1.18?	293.627 +0.22?	330.539 +0.87?	367.523 +2.09?	404.588 +3.43?	441.743 +4.89?	478.994 +6.48?	516.351 +8.2?	553.821 +10.04?	591.412 +12.?
D#1	38.8028 -3.92?	77.6145 -3.72?	116.444 -3.39?	155.3 -2.93?	194.191 -2.34?	233.127 -1.61?	272.116 -0.76?	311.166 +0.22?	350.287 +1.34?	389.488 +2.58?	428.777 +3.95?	468.162 +5.45?	507.653 +7.08?	547.257 +8.84?	586.985 +10.72?	626.843 +12.73?
E1	41.1175 -3.61?	82.2444 -3.41?	123.39 -3.09?	164.563 -2.63?	205.774 -2.04?	247.031 -1.33?	288.343 -0.48?	329.721 +0.49?	371.172 +1.6?	412.707 +2.83?	454.334 +4.19?	496.063 +5.67?	537.903 +7.29?	579.862 +9.03?	621.949 +10.89?	664.175 +12.88?
F1	43.5696 -3.33?	87.1476 -3.16?	130.743 -2.88?	174.363 -2.48?	218.018 -1.98?	261.715 -1.36?	305.463 -0.63?	349.27 +0.21?	393.145 +1.16?	437.096 +2.23?	481.132 +3.4?	525.261 +4.69?	569.491 +6.08?	613.83 +7.58?	658.287 +9.19?	702.871 +10.91?
F#1	46.167 -3.08?	92.3423 -2.92?	138.534 -2.67?	184.75 -2.31?	231. -1.84?	277.29 -1.28?	323.63 -0.61?	370.028 +0.16?	416.491 +1.03?	463.028 +2.?	509.647 +3.08?	556.356 +4.25?	603.163 +5.53?	650.076 +6.91?	697.104 +8.38?	744.254 +9.95?
G1	48.9185 -2.86?	97.8452 -2.71?	146.789 -2.47?	195.757 -2.13?	244.758 -1.69?	293.8 -1.15?	342.892 -0.52?	392.041 +0.21?	441.257 +1.03?	490.546 +1.95?	539.918 +2.97?	589.38 +4.08?	638.941 +5.29?	688.608 +6.59?	738.39 +7.99?	788.295 +9.48?
G#1	51.8331 -2.66?	103.675 -2.53?	155.532 -2.3?	207.415 -1.97?	259.331 -1.56?	311.287 -1.06?	363.294 -0.46?	415.358 +0.23?	467.488 +1.?	519.691 +1.87?	571.977 +2.83?	624.354 +3.88?	676.828 +5.02?	729.409 +6.25?	782.105 +7.57?	834.924 +8.97?
A1	54.9206 -2.5?	109.85 -2.36?	164.795 -2.14?	219.766 -1.84?	274.77 -1.44?	329.816 -0.96?	384.912 -0.39?	440.067 +0.27?	495.288 +1.01?	550.584 +1.84?	605.963 +2.76?	661.434 +3.76?	717.005 +4.85?	772.683 +6.03?	828.478 +7.29?	884.397 +8.63?
A#1	58.1912 -2.35?	116.391 -2.22?	174.609 -2.?	232.853 -1.7?	291.133 -1.3?	349.456 -0.82?	407.833 -0.25?	466.271 +0.4?	524.78 +1.15?	583.368 +1.98?	642.044 +2.89?	700.817 +3.89?	759.695 +4.98?	818.688 +6.15?	877.803 +7.41?	937.049 +8.75?
B1	61.6557 -2.23?	123.321 -2.1?	185.004 -1.89?	246.714 -1.59?	308.46 -1.21?	370.252 -0.74?	432.099 -0.19?	494.009 +0.45?	555.991 +1.17?	618.055 +1.97?	680.21 +2.86?	742.464 +3.83?	804.827 +4.89?	867.307 +6.03?	929.913 +7.25?	992.654 +8.55?
C2	65.3258 -2.13?	130.66 -2.01?	196.012 -1.82?	261.39 -1.55?	326.803 -1.21?	392.259 -0.79?	457.767 -0.29?	523.337 +0.29?	588.975 +0.94?	654.692 +1.67?	720.495 +2.47?	786.393 +3.35?	852.395 +4.3?	918.51 +5.33?	984.745 +6.43?	1051.11 +7.61?
C#2	69.2137 -2.04?	138.437 -1.92?	207.681 -1.71?	276.954 -1.42?	346.268 -1.05?	415.631 -0.59?	485.053 -0.05?	554.546 +0.57?	624.118 +1.27?	693.779 +2.06?	763.539 +2.93?	833.409 +3.88?	903.398 +4.91?	973.515 +6.02?	1043.77 +7.21?	1114.17 +8.48?
D2	73.3322 -1.98?	146.677 -1.83?	220.046 -1.59?	293.453 -1.25?	366.908 -0.81?	440.426 -0.27?	514.018 +0.36?	587.696 +1.08?	661.472 +1.91?	735.36 +2.83?	809.371 +3.84?	883.517 +4.95?	957.81 +6.16?	1032.26 +7.46?	1106.89 +8.86?	1181.7 +10.34?
D#2	77.6953 -1.92?	155.405 -1.76?	233.143 -1.5?	310.924 -1.12?	388.763 -0.65?	466.672 -0.06?	544.668 +0.63?	622.764 +1.42?	700.973 +2.32?	779.311 +3.33?	857.792 +4.44?	936.429 +5.65?	1015.24 +6.97?	1094.23 +8.39?	1173.42 +9.91?	1252.82 +11.53?
E2	82.3174 -1.88?	164.651 -1.71?	247.017 -1.42?	329.431 -1.03?	411.91 -0.52?	494.47 +0.1?	577.126 +0.84?	659.895 +1.69?	742.794 +2.64?	825.837 +3.71?	909.04 +4.9?	992.421 +6.19?	1075.99 +7.59?	1159.77 +9.1?	1243.78 +10.72?	1328.02 +12.45?
F2	87.214 -1.84?	174.446 -1.66?	261.713 -1.37?	349.033 -0.96?	436.424 -0.43?	523.904 +0.21?	611.49 +0.97?	699.2 +1.85?	787.05 +2.84?	875.06 +3.95?	963.246 +5.17?	1051.63 +6.5?	1140.22 +7.96?	1229.04 +9.52?	1318.1 +11.2?	1407.43 +12.99?
F#2	92.4016 -1.81?	184.823 -1.62?	277.286 -1.31?	369.809 -0.86?	462.413 -0.29?	555.119 +0.4?	647.946 +1.22?	740.914 +2.17?	834.044 +3.24?	927.356 +4.43?	1020.87 +5.75?	1114.6 +7.2?	1208.58 +8.76?	1302.82 +10.45?	1397.33 +12.26?	1492.15 +14.19?
G2	97.8974 -1.79?	195.818 -1.58?	293.785 -1.24?	391.823 -0.76?	489.953 -0.14?	588.199 +0.61?	686.586 +1.5?	785.134 +2.53?	883.869 +3.69?	982.812 +4.99?	1081.99 +6.42?	1181.42 +7.98?	1281.12 +9.68?	1381.13 +11.5?	1481.46 +13.46?	1582.13 +15.55?
G#2	103.72 -1.77?	207.465 -1.56?	311.259 -1.21?	415.129 -0.73?	519.098 -0.11?	623.191 +0.66?	727.434 +1.55?	831.851 +2.59?	936.466 +3.76?	1041.3 +5.07?	1146.39 +6.51?	1251.75 +8.09?	1357.4 +9.81?	1463.38 +11.65?	1569.7 +13.63?	1676.38 +15.74?
A2	109.889 -1.75?	219.804 -1.54?	329.773 -1.19?	439.822 -0.69?	549.979 -0.06?	660.27 +0.71?	770.721 +1.62?	881.359 +2.68?	992.21 +3.87?	1103.3 +5.19?	1214.66 +6.66?	1326.31 +8.26?	1438.28 +10.?	1550.6 +11.88?	1663.28 +13.88?	1776.36 +16.02?
A#2	116.424 -1.73?	232.879 -1.5?	349.397 -1.11?	466.008 -0.57?	582.744 +0.12?	699.635 +0.97?	816.713 +1.97?	934.008 +3.12?	1051.55 +4.43?	1169.37 +5.88?	1287.5 +7.49?	1405.98 +9.25?	1524.82 +11.15?	1644.06 +13.2?	1763.73 +15.4?	1883.86 +17.75?
B2	123.349 -1.71?	246.734 -1.45?	370.193 -1.02?	493.763 -0.42?	617.481 +0.36?	741.382 +1.31?	865.505 +2.43?	989.885 +3.71?	1114.56 +5.17?	1239.56 +6.8?	1364.93 +8.59?	1490.71 +10.55?	1616.92 +12.68?	1743.6 +14.97?	1870.79 +17.42?	1998.53 +20.04?
C3	130.685 -1.69?	261.413 -1.4?	392.229 -0.92?	523.174 -0.25?	654.293 +0.61?	785.629 +1.66?	917.225 +2.91?	1049.12 +4.34?	1181.37 +5.96?	1314. +7.76?	1447.07 +9.75?	1580.6 +11.93?	1714.65 +14.29?	1849.26 +16.83?	1984.47 +19.55?	2120.31 +22.45?
C#3	138.458 -1.66?	276.967 -1.34?	415.579 -0.81?	554.345 -0.06?	693.316 +0.9?	832.544 +2.08?	972.079 +3.46?	1111.97 +5.06?	1252.27 +6.86?	1393.03 +8.87?	1534.3 +11.09?	1676.13 +13.52?	1818.56 +16.14?	1961.65 +18.97?	2105.45 +22.?	2250. +25.22?
D3	146.694 -1.63?	293.447 -1.28?	440.321 -0.69?	587.376 +0.14?	734.67 +1.2?	882.265 +2.5?	1030.22 +4.03?	1178.59 +5.79?	1327.44 +7.78?	1476.83 +10.01?	1626.81 +12.45?	1777.45 +15.13?	1928.79 +18.02?	2080.91 +21.14?	2233.84 +24.48?	2387.66 +28.03?
D#3	155.42 -1.6?	310.906 -1.22?	466.527 -0.6?	622.349 +0.27?	778.439 +1.39?	934.864 +2.75?	1091.69 +4.36?	1248.98 +6.22?	1406.81 +8.31?	1565.23 +10.65?	1724.32 +13.23?	1884.13 +16.04?	2044.74 +19.09?	2206.2 +22.37?	2368.58 +25.87?	2531.94 +29.61?
E3	164.665 -1.56?	329.406 -1.16?	494.299 -0.49?	659.42 +0.44?	824.844 +1.63?	990.646 +3.09?	1156.9 +4.81?	1323.69 +6.79?	1491.07 +9.03?	1659.14 +11.52?	1827.95 +14.27?	1997.59 +17.27?	2168.12 +20.52?	2339.62 +24.02?	2512.16 +27.76?	2685.8 +31.74?
F3	174.461 -1.51?	349.013 -1.06?	523.746 -0.31?	698.75 +0.73?	874.116 +2.08?	1049.93 +3.72?	1226.29 +5.65?	1403.28 +7.87?	1580.98 +10.39?	1759.49 +13.19?	1938.89 +16.27?	2119.27 +19.64?	2300.72 +23.29?	2483.31 +27.21?	2667.14 +31.4?	2852.28 +35.85?
F#3	184.841 -1.46?	369.789 -0.96?	554.95 -0.12?	740.432 +1.04?	926.34 +2.54?	1112.78 +4.36?	1299.86 +6.51?	1487.68 +8.99?	1676.34 +11.78?	1865.96 +14.9?	2056.62 +18.33?	2248.44 +22.07?	2441.52 +26.12?	2635.94 +30.47?	2831.82 +35.12?	3029.24 +40.06?
G3	195.839 -1.4?	391.804 -0.84?	588.021 +0.09?	784.616 +1.38?	981.713 +3.05?	1179.44 +5.08?	1377.92 +7.47?	1577.27 +10.23?	1777.62 +13.34?	1979.08 +16.8?	2181.79 +20.61?	2385.85 +24.77?	2591.39 +29.26?	2798.52 +34.09?	3007.36 +39.24?	3218.01 +44.72?
G#3	207.492 -1.33?	415.137 -0.69?	623.088 +0.37?	831.498 +1.86?	1040.52 +3.76?	1250.3 +6.09?	1460.99 +8.83?	1672.75 +11.98?	1885.71 +15.54?	2100.03 +19.5?	2315.85 +23.85?	2533.32 +28.59?	2752.57 +33.72?	2973.74 +39.23?	3196.98 +45.1?	3422.41 +51.33?
A3	219.839 -1.26?	439.861 -0.54?	660.247 +0.65?	881.179 +2.32?	1102.84 +4.46?	1325.4 +7.08?	1549.05 +10.15?	1773.96 +13.69?	2000.31 +17.68?	2228.27 +22.11?	2458.02 +26.99?	2689.71 +32.3?	2923.53 +38.04?	3159.62 +44.19?	3398.16 +50.75?	3639.3 +57.71?
A#3	232.922 -1.18?	466.042 -0.45?	699.557 +0.77?	933.663 +2.48?	1168.56 +4.68?	1404.44 +7.35?	1641.49 +10.49?	1879.91 +14.11?	2119.89 +18.19?	2361.62 +22.73?	2605.28 +27.72?	2851.05 +33.15?	3099.12 +39.02?	3349.66 +45.31?	3602.85 +52.01?	3858.86 +59.12?
B3	246.784 -1.1?	493.785 -0.34?	741.219 +0.93?	989.302 +2.69?	1238.25 +4.96?	1488.27 +7.73?	1739.59 +10.98?	1992.4 +14.72?	2246.92 +18.94?	2503.35 +23.63?	2761.9 +28.79?	3022.77 +34.4?	3286.15 +40.46?	3552.23 +46.96?	3821.22 +53.89?	4093.29 +61.23?
C4	261.472 -1.01?	523.2 -0.17?	785.437 +1.24?	1048.44 +3.2?	1312.45 +5.72?	1577.74 +8.79?	1844.54 +12.4?	2113.1 +16.55?	2383.67 +21.22?	2656.48 +26.42?	2931.78 +32.13?	3209.8 +38.34?	3490.76 +45.04?	3774.9 +52.22?	4062.43 +59.86?	4353.57 +67.95?
C#4	277.035 -0.91?	554.368 -0.01?	832.294 +1.56?	1111.31 +3.71?	1391.11 +6.48?	1672.58 +9.85?	1955.82 +13.81?	2241.1 +18.36?	2528.72 +23.49?	2818.95 +29.19?	3112.07 +35.45?	3408.34 +42.24?	3708.03 +49.57?	4011.41 +57.42?	4318.71 +65.77?	4630.2 +74.6?
D4	293.526 -0.82?	587.392 +0.19?	881.94 +1.86?	1177.51 +4.2?	1474.43 +7.19?	1773.05 +10.84?	2073.69 +15.13?	2376.67 +20.05?	2682.34 +25.59?	2990.99 +31.75?	3302.94 +38.5?	3618.51 +45.84?	3937.99 +53.74?	4261.68 +62.19?	4589.86 +71.19?	4922.81 +80.69?
D#4	310.998 -0.71?	622.371 +0.33?	934.493 +2.07?	1247.74 +4.49?	1562.47 +7.59?	1879.06 +11.37?	2197.87 +15.82?	2519.27 +20.92?	2843.6 +26.67?	3171.21 +33.04?	3502.46 +40.04?	3837.66 +47.63?	4177.16 +55.81?	4521.27 +64.56?	4870.31 +73.86?	5224.59 +83.7?
E4	329.512 -0.6?	659.446 +0.51?	990.223 +2.35?	1322.26 +4.93?	1655.98 +8.22?	1991.79 +12.24?	2330.1 +16.96?	2671.32 +22.38?	3015.84 +28.48?	3364.05 +35.24?	3716.35 +42.66?	4073.09 +50.71?	4434.66 +59.37?	4801.41 +68.64?	5173.69 +78.48?	5551.84 +88.87?
F4	349.129 -0.49?	698.766 +0.77?	1049.42 +2.87?	1401.6 +5.8?	1755.8 +9.55?	2112.52 +14.12?	2472.25 +19.48?	2835.48 +25.63?	3202.69 +32.54?	3574.33 +40.21?	3950.87 +48.6?	4332.75 +57.7?	4720.41 +67.48?	5114.27 +77.92?	5514.75 +89.?	5922.24 +100.69?
F#4	369.914 -0.37?	740.425 +1.03?	1112.13 +3.35?	1485.62 +6.59?	1861.48 +10.74?	2240.29 +15.79?	2622.64 +21.71?	3009.07 +28.5?	3400.15 +36.12?	3796.43 +44.57?	4198.42 +53.81?	4606.65 +63.82?	5021.62 +74.57?	5443.82 +86.03?	5873.72 +98.18?	6311.78 +110.97?
G4	391.938 -0.25?	784.57 +1.28?	1178.59 +3.84?	1574.69 +7.39?	1973.55 +11.95?	2375.84 +17.48?	2782.23 +23.98?	3193.37 +31.41?	3609.91 +39.76?	4032.47 +49.?	4461.66 +59.09?	4898.08 +70.02?	5342.29 +81.74?	5794.87 +94.22?	6256.34 +107.43?	6727.22 +121.33?
G#4	415.274 -0.12?	831.382 +1.61?	1249.16 +4.5?	1669.42 +8.53?	2093. +13.69?	2520.69 +19.94?	2953.3 +27.28?	3391.59 +35.67?	3836.34 +45.08?	4288.28 +55.48?	4748.14 +66.83?	5216.6 +79.09?	5694.36 +92.23?	6182.05 +106.19?	6680.31 +120.94?	7189.73 +136.44?
A4	[440.Hz]	880.998 +1.97?	1323.99 +5.23?	1769.95 +9.77?	2219.87 +15.57?	2674.71 +22.62?	3135.4 +30.87?	3602.87 +40.29?	4078.02 +50.85?	4561.72 +62.49?	5054.82 +75.19?	5558.14 +88.88?	6072.46 +103.52?	6598.54 +119.06?	7137.1 +135.45?	7688.83 +152.63?
A#4	466.2 +0.14?	933.555 +2.28?	1403.21 +5.84?	1876.32 +10.8?	2353.99 +17.13?	2837.34 +24.81?	3327.46 +33.79?	3825.4 +44.05?	4332.2 +55.53?	4848.86 +68.17?	5376.33 +81.94?	5915.56 +96.78?	6467.42 +112.62?	7032.78 +129.4?	7612.45 +147.08?	8207.2 +165.58?
B4	493.962 +0.28?	989.239 +2.58?	1487.14 +6.41?	1988.96 +11.73?	2495.99 +18.53?	3009.48 +26.77?	3530.66 +36.41?	4060.73 +47.4?	4600.86 +59.69?	5152.18 +73.22?	5715.76 +87.93?	6292.66 +103.76?	6883.87 +120.65?	7490.34 +138.53?	8112.97 +157.32?	8752.62 +176.97?
C5	523.379 +0.43?	1048.24 +2.88?	1576.06 +6.95?	2108.3 +12.61?	2646.4 +19.84?	3191.78 +28.59?	3745.82 +38.83?	4309.87 +50.49?	4885.22 +63.51?	5473.15 +77.85?	6074.84 +93.41?	6691.48 +110.15?	7324.15 +127.98?	7973.92 +146.83?	8641.77 +166.64?	9328.66 +187.32?
C#5	554.549 +0.58?	1110.87 +3.34?	1670.71 +7.91?	2235.81 +14.27?	2807.89 +22.39?	3388.63 +32.2?	3979.65 +43.66?	4582.54 +56.69?	5198.84 +71.23?	5830. +87.2?	6477.44 +104.51?	7142.5 +123.07?	7826.45 +142.82?	8530.49 +163.64?	9255.77 +185.47?	10003.3 +208.21?
D5	587.58 +0.74?	1177.31 +3.9?	1771.32 +9.15?	2371.72 +16.43?	2980.58 +25.72?	3599.94 +36.92?	4231.73 +49.99?	4877.88 +64.82?	5540.19 +81.33?	6220.41 +99.41?	6920.19 +118.97?	7641.1 +139.9?	8384.63 +162.08?	9152.15 +185.42?	9944.98 +209.81?	10764.3 +235.14?
D#5	622.581 +0.91?	1247.68 +4.42?	1877.82 +10.23?	2515.45 +18.29?	3163. +28.55?	3822.83 +40.93?	4497.24 +55.33?	5188.4 +71.66?	5898.43 +89.8?	6629.33 +109.64?	7382.99 +131.04?	8161.19 +153.89?	8965.61 +178.07?	9797.81 +203.44?	10659.2 +229.89?	11551.3 +257.29?
E5	659.672 +1.1?	1322.26 +4.93?	1990.67 +11.27?	2667.75 +20.06?	3356.3 +31.25?	4059.03 +44.72?	4778.55 +60.38?	5517.4 +78.1?	6277.94 +97.75?	7062.47 +119.21?	7873.1 +142.32?	8711.86 +166.94?	9580.6 +192.93?	10481.1 +220.15?	11414.9 +248.46?	12383.5 +277.73?
F5	698.978 +1.3?	1401.29 +5.42?	2110.24 +12.25?	2829.1 +21.72?	3561.03 +33.75?	4309.12 +48.23?	5076.34 +65.03?	5865.53 +84.03?	6679.37 +105.06?	7520.42 +127.98?	8391.06 +152.62?	9293.52 +178.83?	10229.9 +206.45?	11202. +235.31?	12211.8 +265.28?	13260.7 +296.21?
F#5	740.633 +1.51?	1485.04 +5.92?	2236.97 +13.22?	3000.11 +23.33?	3778.03 +36.16?	4574.22 +51.59?	5392.02 +69.48?	6234.6 +89.67?	7104.96 +112.?	8005.95 +136.29?	8940.2 +162.36?	9910.17 +190.05?	10918.2 +219.18?	11966.2 +249.57?	13056.3 +281.06?	14190.2 +313.51?
G5	784.779 +1.75?	1573.85 +6.47?	2371.45 +14.28?	3181.77 +25.11?	4008.85 +38.83?	4856.63 +55.31?	5728.86 +74.39?	6629.1 +95.89?	7560.72 +119.63?	8526.87 +145.42?	9530.49 +173.06?	10574.3 +202.34?	11660.8 +233.09?	12792.2 +265.12?	13970.8 +298.25?	15198.3 +332.32?
G#5	831.568 +2.?	1668.11 +7.17?	2514.55 +15.72?	3375.71 +27.54?	4256.28 +42.51?	5160.78 +60.47?	6093.5 +81.21?	7058.51 +104.55?	8059.63 +130.26?	9100.41 +158.12?	10184.2 +187.9?	11313.9 +219.39?	12492.4 +252.36?	13722.2 +286.61?	15005.5 +321.95?	
A5	881.159 +2.28?	1768.13 +7.99?	2666.68 +17.41?	3582.43 +30.44?	4520.82 +46.9?	5487.09 +66.61?	6486.2 +89.34?	7522.8 +114.84?	8601.25 +142.86?	9725.57 +173.14?	10899.5 +205.42?	12126.3 +239.45?	13409.2 +274.97?	14751. +311.77?		
A#5	933.723 +2.59?	1874.29 +8.93?	2828.47 +19.38?	3802.85 +33.81?	4803.82 +52.02?	5837.44 +73.76?	6909.45 +98.77?	8025.22 +126.76?	9189.71 +157.43?	10407.5 +190.46?	11682.7 +225.56?	13019.2 +262.44?	14420.3 +300.82?	15889.1 +340.44?		
B5	989.441 +2.94?	1986.89 +9.93?	3000.26 +21.46?	4037.23 +37.35?	5105.21 +57.37?	6211.23 +81.21?	7361.86 +108.57?	8563.26 +139.11?	9821.03 +172.45?	11140.3 +208.27?	12525.8 +246.2?	13981.7 +285.92?	15511.6 +327.12?			
C6	1048.51 +3.32?	2106.25 +10.92?	3182.32 +23.45?	4285.56 +40.69?	5424.44 +62.37?	6606.96 +88.14?	7840.6 +117.65?	9132.28 +150.48?	10488.3 +186.25?	11914.4 +224.56?	13415.6 +265.01?	14996.7 +307.25?				
C#6	1111.13 +3.74?	2232.9 +12.02?	3375.81 +25.64?	4550. +44.36?	5765.18 +67.84?	7030.46 +95.7?	8354.31 +127.51?	9744.52 +162.82?	11208.1 +201.17?	12751.5 +242.12?	14380.3 +285.22?					
D6	1177.52 +4.22?	2367.45 +13.32?	3581.99 +28.27?	4832.92 +48.79?	6131.44 +74.47?	7488.02 +104.86?	8912.33 +139.45?	10413.2 +177.72?	11998.5 +219.14?	13675.3 +263.2?	15449.8 +309.42?					
D#6	1247.92 +4.74?	2510.82 +15.1?	3803.39 +32.1?	5139.75 +55.35?	6533.24 +84.36?	7996.2 +118.54?	9539.94 +157.27?	11174.6 +199.9?	12909.3 +245.81?	14751.8 +294.39?						
E6	1322.57 +5.33?	2663.22 +17.12?	4039.62 +36.42?	5468.69 +62.75?	6966.24 +95.46?	8546.82 +133.83?	10223.6 +177.08?	12008.2 +224.45?	13910.8 +275.17?	15940.5 +328.55?						
F6	1401.74 +5.98?	2824.85 +19.12?	4290.16 +40.6?	5817.51 +69.8?	7425.33 +105.95?	9130.41 +148.17?	10947.7 +195.56?	12890.5 +247.19?	14970.1 +302.22?							
F#6	1485.71 +6.7?	2996.68 +21.35?	4557.47 +45.24?	6191.32 +77.61?	7919.66 +117.53?	9761.82 +163.94?	11734.9 +215.76?	13853.6 +271.94?								
G6	1574.78 +7.5?	3179.38 +23.81?	4842.7 +50.33?	6591.92 +86.15?	8451.89 +130.13?	10444.8 +181.01?	12589.9 +237.53?	14903.9 +298.46?								
G#6	1669.27 +8.38?	3373.65 +26.49?	5147.04 +55.85?	7021.12 +95.36?	9024.59 +143.63?	11182.8 +199.2?	13517.4 +260.58?									
A6	1769.53 +9.36?	3580.22 +29.37?	5471.67 +61.74?	7480.63 +105.11?	9640.07 +157.85?	11978.7 +218.24?	14521. +284.57?									
A#6	1875.92 +10.43?	3799.82 +32.43?	5817.71 +67.9?	7971.96 +115.24?	10300.2 +172.52?	12834.8 +237.75?	15603. +309.?									
B6	1988.83 +11.62?	4033.44 +35.73?	6187.13 +74.48?	8498.59 +125.98?	11010.6 +187.99?	13759.6 +258.2?										
C7	2108.67 +12.91?	4282.08 +39.29?	6581.8 +81.54?	9063.63 +137.42?	11776.1 +204.35?	14759.9 +279.69?										
C#7	2235.89 +14.33?	4546.82 +43.14?	7003.8 +89.13?	9670.67 +149.65?	12602.3 +221.73?	15843.9 +302.38?										
D7	2370.97 +15.89?	4828.86 +47.33?	7455.52 +97.33?	10323.8 +162.8?	13495.6 +240.3?											
D#7	2514.41 +17.58?	5129.51 +51.9?	7939.62 +106.24?	11027.9 +177.01?	14463.8 +260.25?											
E7	2666.75 +19.41?	5450.26 +56.9?	8459.22 +115.99?	11788.4 +192.47?	15515.9 +281.81?											
F7	2828.57 +21.4?	5792.73 +62.41?	9017.88 +126.7?	12612. +209.38?												
F#7	3000.49 +23.55?	6158.79 +68.49?	9619.76 +138.56?	13506.5 +228.01?												
G7	3183.18 +25.88?	6550.54 +75.25?	10269.8 +151.76?	14481.2 +248.64?												
G#7	3377.32 +28.37?	6970.39 +82.8?	10973.8 +166.54?	15547.5 +271.64?												
A7	3583.69 +31.05?	7421.14 +91.28?	11738.7 +183.2?													
A#7	3803.07 +33.91?	7906.05 +100.86?	12573.1 +202.08?													
B7	4036.34 +36.97?	8421.62 +110.23?	13459.6 +220.04?													
C8	4284.41 +40.23?	8974.67 +120.35?	14419.6 +239.31?													

Figure 7-1 Tuning Table for Grand Piano

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
A0	27.1975 -19.15?	54.4302 -18.02?	81.7335 -16.16?	109.142 -13.55?	136.691 -10.21?	164.415 -6.14?	192.348 -1.36?	220.524 +4.12?	248.975 +10.29?	277.736 +17.14?	306.837 +24.65?	336.31 +32.79?	366.185 +41.56?	396.494 +50.93?	427.264 +60.89?	458.525 +71.4?
A#0	28.8329 -18.05?	57.7015 -16.98?	86.6414 -15.2?	115.688 -12.71?	144.877 -9.52?	174.243 -5.63?	203.821 -1.07?	233.644 +4.18?	263.747 +10.08?	294.163 +16.63?	324.925 +23.81?	356.063 +31.61?	387.611 +40.?	419.597 +48.98?	452.053 +58.52?	485.006 +68.6?
B0	30.5664 -16.98?	61.1693 -15.94?	91.845 -14.23?	122.63 -11.83?	153.56 -8.75?	184.671 -5.01?	215.998 -0.61?	247.576 +4.45?	279.44 +10.14?	311.624 +16.46?	344.162 +23.39?	377.085 +30.91?	410.426 +39.02?	444.216 +47.69?	478.487 +56.91?	513.267 +66.65?
C1	32.4038 -15.92?	64.8424 -14.99?	97.3508 -13.44?	129.964 -11.27?	162.715 -8.49?	195.64 -5.11?	228.773 -1.13?	262.146 +3.44?	295.793 +8.59?	329.746 +14.31?	364.039 +20.59?	398.702 +27.42?	433.768 +34.78?	469.265 +42.66?	505.224 +51.04?	541.674 +59.91?
C#1	34.3511 -14.89?	68.7389 -13.96?	103.2 -12.42?	137.771 -10.27?	172.488 -7.51?	207.388 -4.16?	242.506 -0.2?	277.877 +4.34?	313.536 +9.45?	349.519 +15.13?	385.858 +21.37?	422.587 +28.14?	459.739 +35.45?	497.347 +43.28?	535.44 +51.6?	574.051 +60.41?
D1	36.4148 -13.88?	72.8629 -13.09?	109.378 -11.77?	145.992 -9.93?	182.74 -7.56?	219.654 -4.68?	256.766 -1.28?	294.108 +2.62?	331.714 +7.02?	369.613 +11.91?	407.839 +17.28?	446.42 +23.13?	485.388 +29.44?	524.773 +36.21?	564.602 +43.41?	604.906 +51.05?
D#1	38.6017 -12.92?	77.2388 -12.12?	115.947 -10.8?	154.761 -8.95?	193.716 -6.58?	232.847 -3.7?	272.188 -0.3?	311.775 +3.61?	351.641 +8.01?	391.819 +12.91?	432.342 +18.29?	473.244 +24.15?	514.556 +30.47?	556.309 +37.24?	598.536 +44.46?	641.265 +52.11?
E1	40.9191 -11.99?	81.8729 -11.25?	122.896 -10.02?	164.024 -8.31?	205.29 -6.12?	246.729 -3.44?	288.375 -0.29?	330.263 +3.34?	372.424 +7.43?	414.893 +11.97?	457.701 +16.97?	500.882 +22.41?	544.467 +28.29?	588.487 +34.59?	632.973 +41.3?	677.955 +48.43?
F1	43.3745 -11.1?	86.7867 -10.34?	130.274 -9.09?	173.875 -7.34?	217.625 -5.1?	261.563 -2.37?	305.725 +0.85?	350.147 +4.55?	394.866 +8.73?	439.919 +13.37?	485.339 +18.47?	531.162 +24.03?	577.422 +30.02?	624.152 +36.45?	671.387 +43.31?	719.159 +50.57?
F#1	45.9762 -10.25?	91.9917 -9.51?	138.085 -8.28?	184.296 -6.57?	230.663 -4.37?	277.225 -1.68?	324.02 +1.48?	371.086 +5.11?	418.461 +9.2?	466.182 +13.76?	514.286 +18.77?	562.809 +24.22?	611.787 +30.11?	661.254 +36.42?	711.246 +43.15?	761.797 +50.29?
G1	48.7328 -9.44?	97.4972 -8.88?	146.325 -7.95?	195.247 -6.64?	244.295 -4.96?	293.501 -2.91?	342.896 -0.5?	392.509 +2.27?	442.373 +5.41?	492.517 +8.9?	542.972 +12.74?	593.768 +16.92?	644.933 +21.45?	696.497 +26.32?	748.489 +31.51?	800.937 +37.03?
G#1	51.6533 -8.68?	103.337 -8.16?	155.084 -7.3?	206.923 -6.09?	258.885 -4.54?	311.002 -2.64?	363.304 -0.41?	415.821 +2.16?	468.583 +5.06?	521.62 +8.29?	574.963 +11.84?	628.639 +15.72?	682.679 +19.92?	737.11 +24.43?	791.962 +29.25?	847.262 +34.37?
A1	54.7473 -7.97?	109.528 -7.45?	164.374 -6.58?	219.319 -5.36?	274.395 -3.81?	329.636 -1.91?	385.074 +0.34?	440.74 +2.91?	496.667 +5.83?	552.888 +9.07?	609.432 +12.64?	666.332 +16.54?	723.619 +20.75?	781.323 +25.28?	839.474 +30.11?	898.101 +35.25?
A#1	58.0252 -7.3?	116.084 -6.8?	174.209 -5.98?	232.433 -4.82?	290.79 -3.34?	349.313 -1.53?	408.035 +0.6?	466.987 +3.06?	526.202 +5.83?	585.714 +8.92?	645.552 +12.32?	705.749 +16.03?	766.337 +20.05?	827.346 +24.36?	888.806 +28.97?	950.748 +33.88?
B1	61.4978 -6.67?	123.04 -6.05?	184.671 -5.01?	246.434 -3.56?	308.374 -1.69?	370.535 +0.58?	432.96 +3.26?	495.692 +6.33?	558.775 +9.81?	622.25 +13.68?	686.16 +17.94?	750.546 +22.57?	815.449 +27.59?	880.91 +32.97?	946.969 +38.71?	1013.66 +44.81?
C2	65.1765 -6.09?	130.394 -5.55?	195.694 -4.64?	261.116 -3.37?	326.703 -1.74?	392.494 +0.25?	458.53 +2.6?	524.852 +5.29?	591.498 +8.34?	658.509 +11.73?	725.924 +15.47?	793.782 +19.54?	862.121 +23.94?	930.979 +28.67?	1000.39 +33.73?	1070.4 +39.09?
C#2	69.0735 -5.55?	138.184 -5.09?	207.368 -4.32?	276.664 -3.24?	346.106 -1.85?	415.734 -0.16?	485.581 +1.83?	555.686 +4.13?	626.085 +6.72?	696.812 +9.61?	767.903 +12.79?	839.395 +16.27?	911.321 +20.02?	983.716 +24.07?	1056.61 +28.38?	1130.05 +32.98?
D2	73.2017 -5.06?	146.445 -4.57?	219.772 -3.74?	293.224 -2.59?	366.843 -1.12?	440.669 +0.68?	514.745 +2.8?	589.111 +5.25?	663.807 +8.01?	738.875 +11.08?	814.353 +14.47?	890.282 +18.16?	966.701 +22.16?	1043.65 +26.45?	1121.16 +31.04?	1199.28 +35.92?
D#2	77.5749 -4.61?	155.188 -4.18?	232.876 -3.48?	310.678 -2.5?	388.631 -1.23?	466.772 +0.31?	545.14 +2.13?	623.771 +4.22?	702.702 +6.59?	781.97 +9.22?	861.611 +12.13?	941.663 +15.3?	1022.16 +18.73?	1103.14 +22.42?	1184.63 +26.37?	1266.68 +30.57?
E2	82.2074 -4.19?	164.456 -3.76?	246.786 -3.04?	329.24 -2.03?	411.857 -0.74?	494.679 +0.84?	577.745 +2.7?	661.097 +4.84?	744.775 +7.26?	828.817 +9.95?	913.264 +12.92?	998.156 +16.16?	1083.53 +19.67?	1169.43 +23.45?	1255.88 +27.48?	1342.93 +31.78?
F2	87.1147 -3.81?	174.271 -3.4?	261.511 -2.71?	348.877 -1.74?	436.409 -0.5?	524.149 +1.02?	612.139 +2.81?	700.42 +4.86?	789.032 +7.19?	878.016 +9.78?	967.413 +12.64?	1057.26 +15.76?	1147.6 +19.14?	1238.48 +22.77?	1329.92 +26.65?	1421.97 +30.79?
F#2	92.3131 -3.47?	184.671 -3.05?	277.12 -2.34?	369.703 -1.36?	462.466 -0.09?	555.454 +1.45?	648.712 +3.27?	742.283 +5.36?	836.211 +7.73?	930.541 +10.37?	1025.32 +13.28?	1120.58 +16.45?	1216.37 +19.89?	1312.74 +23.58?	1409.72 +27.54?	1507.36 +31.74?
G2	97.8198 -3.16?	195.687 -2.74?	293.649 -2.04?	391.754 -1.06?	490.048 +0.19?	588.579 +1.73?	687.393 +3.54?	786.538 +5.62?	886.06 +7.98?	986.004 +10.6?	1086.42 +13.49?	1187.35 +16.65?	1288.83 +20.06?	1390.92 +23.74?	1493.66 +27.67?	1597.09 +31.85?
G#2	103.653 -2.88?	207.358 -2.45?	311.168 -1.72?	415.133 -0.71?	519.307 +0.59?	623.741 +2.18?	728.486 +4.06?	833.593 +6.21?	939.114 +8.65?	1045.1 +11.37?	1151.6 +14.36?	1258.66 +17.63?	1366.34 +21.16?	1474.68 +24.97?	1583.73 +29.03?	1693.53 +33.36?
A2	109.832 -2.63?	219.716 -2.24?	329.7 -1.57?	439.836 -0.64?	550.173 +0.55?	660.764 +2.01?	771.656 +3.72?	882.9 +5.7?	994.546 +7.94?	1106.64 +10.43?	1219.24 +13.17?	1332.38 +16.17?	1446.12 +19.42?	1560.51 +22.91?	1675.59 +26.64?	1791.4 +30.62?
A#2	116.378 -2.41?	232.812 -2.?	349.356 -1.31?	466.066 -0.36?	582.997 +0.87?	700.204 +2.38?	817.741 +4.15?	935.662 +6.19?	1054.02 +8.49?	1172.88 +11.06?	1292.27 +13.89?	1412.27 +16.98?	1532.92 +20.33?	1654.28 +23.93?	1776.38 +27.78?	1899.3 +31.87?
B2	123.313 -2.21?	246.683 -1.81?	370.167 -1.14?	493.824 -0.2?	617.709 +1.?	741.88 +2.47?	866.393 +4.2?	991.305 +6.2?	1116.67 +8.45?	1242.55 +10.97?	1368.99 +13.74?	1496.06 +16.76?	1623.8 +20.03?	1752.27 +23.56?	1881.52 +27.32?	2011.61 +31.33?
C3	130.659 -2.04?	261.375 -1.65?	392.208 -1.01?	523.214 -0.12?	654.452 +1.03?	785.979 +2.44?	917.853 +4.09?	1050.13 +6.?	1182.87 +8.15?	1316.12 +10.55?	1449.95 +13.2?	1584.41 +16.09?	1719.55 +19.22?	1855.43 +22.59?	1992.1 +26.2?	2129.62 +30.03?
C#3	138.44 -1.88?	276.94 -1.51?	415.557 -0.9?	554.35 -0.04?	693.378 +1.06?	832.7 +2.4?	972.373 +3.99?	1112.46 +5.81?	1253. +7.88?	1394.08 +10.18?	1535.74 +12.71?	1678.03 +15.48?	1821.02 +18.48?	1964.76 +21.71?	2109.3 +25.16?	2254.7 +28.84?
D3	146.684 -1.74?	293.403 -1.54?	440.191 -1.2?	587.082 -0.73?	734.11 -0.12?	881.311 +0.63?	1028.72 +1.5?	1176.36 +2.52?	1324.29 +3.66?	1472.52 +4.94?	1621.09 +6.35?	1770.04 +7.89?	1919.39 +9.57?	2069.19 +11.37?	2219.47 +13.3?	2370.25 +15.37?
D#3	155.418 -1.62?	310.872 -1.42?	466.399 -1.08?	622.034 -0.61?	777.816 0.?	933.778 +0.74?	1089.96 +1.62?	1246.39 +2.63?	1403.12 +3.77?	1560.17 +5.04?	1717.58 +6.45?	1875.38 +7.98?	2033.62 +9.65?	2192.33 +11.45?	2351.54 +13.37?	2511.29 +15.43?
E3	164.67 -1.51?	329.383 -1.28?	494.181 -0.91?	659.109 -0.38?	824.208 +0.3?	989.522 +1.12?	1155.09 +2.1?	1320.96 +3.22?	1487.18 +4.5?	1653.78 +5.92?	1820.8 +7.48?	1988.29 +9.2?	2156.3 +11.06?	2324.86 +13.06?	2494. +15.2?	2663.79 +17.49?
F3	174.472 -1.41?	349.055 -0.85?	523.863 +0.07?	699.006 +1.37?	874.595 +3.03?	1050.74 +5.05?	1227.56 +7.44?	1405.15 +10.18?	1583.63 +13.29?	1763.1 +16.74?	1943.67 +20.54?	2125.45 +24.68?	2308.54 +29.17?	2493.05 +33.98?	2679.07 +39.12?	2866.7 +44.59?
F#3	184.856 -1.32?	369.834 -0.74?	555.058 +0.21?	740.65 +1.55?	926.732 +3.27?	1113.42 +5.36?	1300.85 +7.83?	1489.13 +10.67?	1678.37 +13.88?	1868.71 +17.45?	2060.25 +21.38?	2253.11 +25.66?	2447.41 +30.3?	2643.26 +35.27?	2840.77 +40.58?	3040.04 +46.22?
G3	195.857 -1.24?	391.85 -0.64?	588.114 +0.36?	784.785 +1.76?	981.997 +3.55?	1179.88 +5.73?	1378.58 +8.31?	1578.21 +11.27?	1778.92 +14.61?	1980.83 +18.33?	2184.08 +22.42?	2388.78 +26.89?	2595.06 +31.71?	2803.06 +36.89?	3012.88 +42.42?	3224.66 +48.29?
G#3	207.513 -1.16?	415.18 -0.51?	623.156 +0.56?	831.593 +2.05?	1040.65 +3.98?	1250.47 +6.32?	1461.21 +9.08?	1673.01 +12.25?	1886.04 +15.84?	2100.43 +19.82?	2316.34 +24.21?	2533.9 +28.99?	2753.26 +34.15?	2974.56 +39.7?	3197.93 +45.61?	3423.52 +51.89?
A3	219.862 -1.08?	439.902 -0.38?	660.298 +0.79?	881.23 +2.42?	1102.87 +4.52?	1325.4 +7.08?	1548.99 +10.09?	1773.82 +13.55?	2000.06 +17.46?	2227.87 +21.8?	2457.43 +26.58?	2688.9 +31.78?	2922.45 +37.4?	3158.23 +43.43?	3396.41 +49.86?	3637.14 +56.68?
A#3	232.945 -1.01?	466.098 -0.24?	699.666 +1.04?	933.856 +2.84?	1168.87 +5.14?	1404.93 +7.95?	1642.21 +11.26?	1880.94 +15.06?	2121.3 +19.34?	2363.5 +24.11?	2607.72 +29.34?	2854.16 +35.04?	3103.02 +41.19?	3354.47 +47.79?	3608.69 +54.82?	3865.87 +62.27?
B3	246.807 -0.94?	493.845 -0.13?	741.344 +1.22?	989.536 +3.1?	1238.65 +5.52?	1488.91 +8.47?	1740.54 +11.93?	1993.77 +15.92?	2248.83 +20.41?	2505.92 +25.41?	2765.26 +30.89?	3027.06 +36.86?	3291.55 +43.31?	3558.91 +50.21?	3829.35 +57.57?	4103.07 +65.36?
C4	261.494 -0.87?	523.253 +0.01?	785.544 +1.48?	1048.63 +3.52?	1312.78 +6.15?	1578.25 +9.35?	1845.3 +13.11?	2114.19 +17.44?	2385.17 +22.31?	2658.49 +27.73?	2934.4 +33.67?	3213.14 +40.14?	3494.95 +47.11?	3780.07 +54.58?	4068.72 +62.54?	4361.13 +70.96?
C#4	277.055 -0.79?	554.41 +0.14?	832.365 +1.7?	1111.22 +3.89?	1391.27 +6.68?	1672.81 +10.09?	1956.13 +14.09?	2241.53 +18.69?	2529.28 +23.88?	2819.68 +29.64?	3112.99 +35.96?	3409.49 +42.82?	3709.44 +50.23?	4013.12 +58.16?	4320.77 +66.59?	4632.64 +75.52?
D4	293.542 -0.72?	587.437 +0.32?	882.035 +2.05?	1177.69 +4.46?	1474.74 +7.55?	1773.54 +11.32?	2074.43 +15.75?	2377.75 +20.83?	2683.82 +26.55?	2992.99 +32.91?	3305.56 +39.87?	3621.87 +47.44?	3942.2 +55.59?	4266.88 +64.31?	4596.19 +73.57?	4930.43 +83.37?
D#4	311.012 -0.63?	622.412 +0.44?	934.586 +2.24?	1247.92 +4.75?	1562.8 +7.96?	1879.6 +11.87?	2198.7 +16.47?	2520.47 +21.75?	2845.28 +27.69?	3173.48 +34.28?	3505.44 +41.51?	3841.49 +49.36?	4181.98 +57.81?	4527.24 +66.85?	4877.59 +76.45?	5233.35 +86.6?
E4	329.522 -0.55?	659.494 +0.63?	990.364 +2.6?	1322.58 +5.34?	1656.58 +8.85?	1992.81 +13.12?	2331.7 +18.14?	2673.67 +23.9?	3019.17 +30.38?	3368.58 +37.57?	3722.33 +45.45?	4080.81 +53.99?	4444.41 +63.18?	4813.51 +73.?	5188.48 +83.42?	5569.67 +94.42?
F4	349.136 -0.45?	698.788 +0.83?	1049.47 +2.95?	1401.7 +5.92?	1755.98 +9.73?	2112.81 +14.35?	2472.69 +19.79?	2836.12 +26.02?	3203.58 +33.03?	3575.53 +40.79?	3952.45 +49.29?	4334.78 +58.51?	4722.96 +68.42?	5117.43 +78.99?	5518.6 +90.21?	5926.87 +102.04?
F#4	369.918 -0.35?	740.398 +0.96?	1112. +3.15?	1485.28 +6.19?	1860.79 +10.1?	2239.08 +14.84?	2620.69 +20.42?	3006.15 +26.81?	3395.98 +34.?	3790.7 +41.96?	4190.82 +50.67?	4596.81 +60.12?	5009.17 +70.27?	5428.34 +81.1?	5854.79 +92.59?	6288.95 +104.7?
G4	391.94 -0.24?	784.531 +1.2?	1178.43 +3.59?	1574.27 +6.93?	1972.71 +11.21?	2374.37 +16.41?	2779.88 +22.52?	3189.86 +29.51?	3604.92 +37.36?	4025.63 +46.06?	4452.59 +55.57?	4886.35 +65.87?	5327.46 +76.93?	5776.46 +88.71?	6233.84 +101.19?	6700.11 +114.34?
G#4	415.274 -0.12?	831.291 +1.42?	1248.79 +4.?	1668.51 +7.59?	2091.18 +12.18?	2517.52 +17.77?	2948.25 +24.32?	3384.05 +31.82?	3825.62 +40.24?	4273.62 +49.55?	4728.71 +59.73?	5191.52 +70.75?	5662.66 +82.56?	6142.73 +95.14?	6632.3 +108.46?	7131.93 +122.47?
A4	[440.Hz]	880.879 +1.73?	1323.51 +4.61?	1768.77 +8.61?	2217.51 +13.73?	2670.6 +19.95?	3128.85 +27.25?	3593.1 +35.59?	4064.14 +44.94?	4542.76 +55.28?	5029.71 +66.57?	5525.74 +78.76?	6031.55 +91.82?	6547.83 +105.71?	7075.25 +120.38?	7614.43 +135.8?
A#4	466.201 +0.14?	933.401 +2.?	1402.6 +5.08?	1874.78 +9.38?	2350.92 +14.87?	2831.98 +21.53?	3318.92 +29.35?	3812.67 +38.28?	4314.12 +48.28?	4824.16 +59.33?	5343.64 +71.39?	5873.4 +84.39?	6414.23 +98.32?	6966.89 +113.11?	7532.13 +128.71?	8110.64 +145.09?
B4	493.965 +0.29?	989.077 +2.3?	1486.48 +5.63?	1987.3 +10.28?	2492.66 +16.22?	3003.66 +23.42?	3521.39 +31.86?	4046.9 +41.5?	4581.23 +52.29?	5125.37 +64.19?	5680.3 +77.16?	6246.94 +91.14?	6826.2 +106.09?	7418.95 +121.95?	8025.99 +138.66?	8648.13 +156.18?
C5	523.387 +0.45?	1048.12 +2.68?	1575.55 +6.39?	2107.01 +11.55?	2643.81 +18.14?	3187.24 +26.13?	3738.58 +35.48?	4299.05 +46.14?	4869.86 +58.06?	5452.17 +71.2?	6047.1 +85.49?	6655.72 +100.87?	7279.08 +117.29?	7918.15 +134.68?	8573.87 +152.98?	9247.13 +172.12?
C#5	554.564 +0.63?	1110.67 +3.03?	1669.86 +7.03?	2233.65 +12.6?	2803.55 +19.71?	3381.03 +28.31?	3967.53 +38.38?	4564.45 +49.84?	5173.16 +62.66?	5794.96 +76.76?	6431.13 +92.08?	7082.87 +108.56?	7751.35 +126.12?	8437.65 +144.7?	9142.82 +164.21?	9867.87 +184.6?
D5	587.603 +0.81?	1177. +3.44?	1769.96 +7.82?	2368.25 +13.9?	2973.61 +21.66?	3587.74 +31.05?	4212.3 +42.02?	4848.89 +54.5?	5499.09 +68.44?	6164.39 +83.75?	6846.22 +100.37?	7545.96 +118.21?	8264.93 +137.19?	9004.34 +157.23?	9765.39 +178.26?	10549.2 +200.18?
D#5	622.615 +1.01?	1247.24 +3.8?	1875.87 +8.44?	2510.5 +14.88?	3153.05 +23.1?	3805.45 +33.04?	4469.54 +44.64?	5147.12 +57.83?	5839.94 +72.55?	6549.66 +88.7?	7277.87 +106.22?	8026.1 +125.?	8795.78 +144.96?	9588.29 +166.02?	10404.9 +188.08?	11246.8 +211.05?
E5	659.719 +1.22?	1321.72 +4.22?	1988.29 +9.19?	2661.67 +16.11?	3344.06 +24.92?	4037.63 +35.57?	4744.47 +47.99?	5466.63 +62.1?	6206.05 +77.81?	6964.6 +95.05?	7744.08 +113.71?	8546.18 +133.7?	9372.5 +154.91?	10224.6 +177.25?	11103.8 +200.62?	12011.4 +224.92?
F5	699.04 +1.45?	1400.82 +4.84?	2108.07 +10.47?	2823.48 +18.28?	3549.68 +28.23?	4289.25 +40.23?	5044.68 +54.2?	5818.36 +70.05?	6612.6 +87.66?	7429.58 +106.94?	8271.39 +127.75?	9139.98 +149.99?	10037.2 +173.53?	10964.8 +198.25?	11924.3 +224.04?	12917.3 +250.79?
F#5	740.712 +1.7?	1484.38 +5.14?	2233.92 +10.86?	2992.25 +18.79?	3762.2 +28.89?	4546.52 +41.07?	5347.9 +55.26?	6168.92 +71.34?	7012.05 +89.21?	7879.63 +108.75?	8773.88 +129.86?	9696.92 +152.39?	10650.7 +176.24?	11637.1 +201.28?	12657.8 +227.39?	13714.4 +254.46?
G5	784.875 +1.96?	1573.4 +5.98?	2369.21 +12.65?	3175.86 +21.89?	3996.85 +33.64?	4835.56 +47.78?	5695.25 +64.2?	6579.04 +82.77?	7489.89 +103.34?	8430.62 +125.77?	9403.85 +149.9?	10412.1 +175.58?	11457.5 +202.65?	12542.4 +230.97?	13668.6 +260.39?	14837.9 +290.78?
G#5	831.681 +2.24?	1667.59 +6.63?	2511.93 +13.92?	3368.82 +24.01?	4242.28 +36.81?	5136.21 +52.21?	6054.35 +70.05?	7000.25 +90.2?	7977.29 +112.48?	8988.64 +136.72?	10037.3 +162.75?	11125.9 +190.39?	12257.2 +219.46?	13433.4 +249.79?	14656.7 +281.24?	15929.1 +313.63?
A5	881.288 +2.54?	1767.31 +7.18?	2662.74 +14.86?	3572.21 +25.49?	4500.18 +38.98?	5451. +55.18?	6428.81 +73.95?	7437.57 +95.11?	8480.99 +118.48?	9562.58 +143.88?	10685.6 +171.11?	11853. +199.98?	13067.7 +230.31?	14332.2 +261.91?	15648.7 +294.62?	
A#5	933.865 +2.86?	1873.19 +7.91?	2823.38 +16.27?	3789.75 +27.84?	4777.44 +42.49?	5791.42 +60.06?	6836.43 +80.38?	7916.95 +103.25?	9037.19 +128.45?	10201.1 +155.78?	11412.3 +185.02?	12674.2 +215.95?	13989.8 +248.35?	15362. +282.04?		
B5	989.593 +3.2?	1985.71 +8.9?	2994.81 +18.32?	4023.2 +31.33?	5077.01 +47.78?	6162.09 +67.46?	7284. +90.16?	8447.98 +115.64?	9658.91 +143.64?	10921.3 +173.89?	12239.3 +206.14?	13616.8 +240.14?	15057.1 +275.64?			
C6	1048.66 +3.57?	2104.68 +9.63?	3175.31 +19.64?	4267.66 +33.45?	5388.59 +50.89?	6544.67 +71.74?	7742.09 +95.76?	8986.67 +122.66?	10283.8 +152.17?	11638.6 +184.01?	13055.4 +217.88?	14538.5 +253.53?				
C#6	1111.27 +3.97?	2231.37 +10.83?	3369. +22.15?	4532.65 +37.74?	5730.48 +57.39?	6970.24 +80.81?	8259.23 +107.7?	9604.23 +137.72?	11011.5 +170.53?	12486.7 +205.79?	14035.1 +243.15?	15661.2 +282.31?				
D6	1177.65 +4.4?	2365.3 +11.74?	3572.83 +23.84?	4809.84 +40.5?	6085.52 +61.46?	7408.61 +86.4?	8787.27 +114.99?	10229. +146.83?	11740.9 +181.57?	13329.1 +218.8?	14999.2 +258.18?					
D#6	1248.01 +4.87?	2506.93 +12.42?	3787.52 +24.86?	5100.22 +41.99?	6455.07 +63.52?	7861.52 +89.13?	9328.43 +118.45?	10864. +151.09?	12475.6 +186.66?	14170.2 +224.75?	15953.8 +264.99?					
E6	1322.6 +5.37?	2658.08 +13.77?	4019.1 +27.61?	5417.94 +46.61?	6866.31 +70.44?	8375.2 +98.71?	9954.83 +130.96?	11614.6 +166.75?	13362.9 +205.59?	15207.4 +247.04?						
F6	1401.69 +5.91?	2818.59 +15.28?	4265.65 +30.68?	5757.3 +51.79?	7307.22 +78.19?	8928.19 +109.4?	10632. +144.89?	12429.3 +184.12?	14329.7 +226.52?							
F#6	1485.54 +6.5?	2988.54 +16.64?	4526.12 +33.29?	6114.76 +56.07?	7770.02 +84.5?	9506.38 +118.03?	11337. +156.06?	13274. +197.95?	15327.8 +243.1?							
G6	1574.46 +7.14?	3168.81 +18.04?	4802.54 +35.92?	6494.33 +60.33?	8261.79 +90.75?	10121.2 +126.52?	12087.3 +166.99?	14173.3 +211.44?								
G#6	1668.75 +7.84?	3359.66 +19.3?	5094.43 +38.06?	6893.79 +63.67?	8777.25 +95.52?	10762.7 +132.93?	12866.5 +175.14?	15102.9 +221.42?								
A6	1768.75 +8.59?	3565.3 +22.14?	5416.7 +44.26?	7348.69 +74.3?	9385.13 +111.45?	11547.7 +154.8?	13855.7 +203.38?									
A#6	1874.83 +9.42?	3783.63 +25.04?	5759.38 +50.45?	7833.16 +84.83?	10033.5 +127.11?	12386. +176.12?	14913. +230.69?									
B6	1987.35 +10.33?	4015.24 +27.9?	6122.88 +56.41?	8346.96 +94.82?	10720.8 +141.81?	13274. +195.99?										
C7	2106.73 +11.32?	4255.33 +28.44?	6486.34 +56.24?	8837.75 +93.73?	11344.2 +139.65?	14036.3 +192.67?										
C#7	2233.39 +12.4?	4518.63 +32.38?	6905.62 +64.68?	9440.65 +107.98?	12165.3 +160.64?	15115.7 +220.93?										
D7	2367.82 +13.58?	4798.2 +36.31?	7351.06 +72.9?	10081.4 +121.67?	13038. +180.58?											
D#7	2510.5 +14.88?	5083.42 +36.27?	7778.7 +70.79?	10651.6 +116.92?	13751.5 +172.83?											
E7	2661.96 +16.3?	5396.27 +39.67?	8272.08 +77.26?	11352.8 +127.29?	14694.4 +187.63?											
F7	2822.79 +17.86?	5735.27 +45.15?	8822.54 +88.79?	12161.6 +146.42?	15818.9 +215.29?											
F#7	2993.59 +19.57?	6092.43 +49.74?	9395.87 +97.79?	12992.8 +160.88?												
G7	3175.03 +21.44?	6471.88 +54.34?	10005. +106.53?	13875.6 +174.69?												
G#7	3367.81 +23.49?	6874.91 +58.92?	10651.5 +114.94?	14811.8 +187.72?												
A7	3572.7 +25.73?	7302.41 +63.36?	11335.3 +122.66?	15798.6 +199.39?												
A#7	3790.51 +28.18?	7755.35 +67.54?	12056.3 +129.41?													
B7	4022.13 +30.87?	8230.93 +70.58?	12799.4 +132.97?													
C8	4268.53 +33.8?	8741.5 +74.77?	13608. +139.02?													

Figure 7-2 Tuning Table for Upright Piano

	1	2	3	4	5	6	7	8	9	10	11	12
A0	27.4413 -3.7?	54.6261 -11.81?	82.0674 -9.1?	109.765 -3.7?	137.463 -0.46?	165.417 +4.37?	193.884 +12.41?	222.095 +16.41?	251.075 +24.83?	280.055 +31.54?	309.804 +41.31?	339.297 +48.11?
A#0	28.8409 -17.57?	57.6818 -17.57?	86.5226 -17.57?	115.579 -14.35?	144.635 -12.41?	173.906 -8.98?	204.038 +0.78?	233.525 +3.29?	264.088 +12.31?	294.65 +19.49?	325.428 +26.49?	356.422 +33.35?
B0	30.6368 -13.?	61.2735 -13.?	91.9103 -13.?	122.746 -10.19?	153.98 -4.03?	185.213 +0.07?	216.845 +6.17?	248.675 +12.11?	280.903 +19.18?	313.33 +25.91?	345.758 +31.4?	378.981 +39.59?
C1	32.4352 -14.24?	64.8703 -14.24?	97.4898 -10.97?	130.109 -9.33?	163.097 -4.43?	196.454 +2.07?	229.626 +5.32?	263.72 +13.81?	297.445 +18.24?	331.539 +23.7?	365.633 +28.16?	400.648 +35.85?
C#1	34.1767 -23.7?	68.502 -19.94?	102.679 -21.19?	137.301 -16.19?	171.924 -13.19?	207.14 -6.22?	242.209 -2.32?	277.723 +3.38?	313.385 +8.62?	349.494 +15.01?	385.751 +20.89?	422.305 +26.99?
D1	36.5828 -5.91?	73.0243 -9.26?	109.748 -5.91?	146.614 -2.58?	183.338 -1.91?	220.768 +4.08?	258.339 +9.3?	296.052 +14.02?	334.189 +19.89?	372.184 +23.91?	410.744 +29.57?	449.304 +34.28?
D#1	38.6613 -10.25?	77.3226 -10.25?	115.984 -10.25?	155.044 -5.79?	194.303 -1.34?	233.761 +3.09?	273.419 +7.51?	313.276 +11.92?	353.532 +17.3?	393.787 +21.59?	434.441 +26.68?	475.295 +31.63?
E1	40.8872 -13.33?	81.9168 -10.32?	122.946 -9.32?	164.404 -4.31?	205.718 -2.51?	247.46 +1.68?	289.06 +3.82?	331.371 +9.14?	373.826 +13.93?	416.85 +20.12?	459.304 +23.02?	502.613 +28.38?
F1	43.2485 -16.13?	86.6554 -12.97?	130.062 -11.91?	173.786 -8.22?	217.668 -4.76?	261.867 -0.35?	306.225 +3.68?	350.899 +8.27?	395.415 +11.13?	440.723 +16.53?	486.031 +20.94?	531.972 +26.67?
F#1	45.9597 -10.87?	91.9195 -10.87?	138.012 -9.2?	184.372 -5.86?	230.731 -3.86?	277.623 +0.8?	324.516 +4.12?	371.808 +8.47?	419.366 +12.94?	467.191 +17.5?	515.282 +22.12?	564.039 +28.?
G1	48.8379 -5.71?	97.5044 -8.75?	146.514 -5.71?	195.694 -2.68?	244.703 -2.07?	293.713 -1.67?	343.236 +1.22?	392.931 +4.13?	442.968 +7.73?	493.005 +10.61?	543.557 +14.6?	594.451 +18.92?
G#1	51.5976 -10.55?	103.379 -7.47?	154.976 -8.5?	206.941 -5.93?	258.722 -5.63?	310.687 -4.4?	363.019 -1.77?	415.351 +0.2?	465.664 -5.76?	520.199 +3.56?	574.367 +10.05?	627.985 +13.92?
A1	54.5567 -14.?	109.325 -10.65?	164.093 -9.54?	218.861 -8.98?	273.841 -7.31?	329.032 -5.08?	384.435 -2.54?	440.049 +0.2?	495.874 +3.06?	551.7 +5.35?	608.371 +9.63?	665.254 +13.73?
A#1	57.9676 -9.02?	115.935 -9.02?	174.057 -7.48?	232.487 -4.42?	290.609 -4.42?	349.039 -2.89?	407.623 -1.14?	466.515 +1.31?	525.562 +3.72?	585.38 +7.93?	645.352 +11.79?	705.632 +15.74?
B1	61.2533 -13.57?	122.864 -8.53?	184.474 -6.85?	246.442 -3.5?	308.41 -1.5?	370.556 +0.68?	432.524 +1.51?	495.027 +4.01?	558.423 +8.72?	621.641 +11.99?	685.573 +16.46?	749.862 +21.?
C2	65.2473 -4.21?	130.223 -7.82?	195.742 -4.21?	261.261 -2.41?	326.78 -1.33?	392.571 +0.59?	458.906 +4.01?	524.969 +5.68?	591.576 +8.57?	658.454 +11.59?	726.148 +16.?	794.386 +20.85?
C#2	69.1197 -4.4?	137.766 -10.34?	207.043 -7.03?	276.321 -5.39?	345.441 -5.19?	415.034 -3.08?	484.785 -1.01?	554.693 +1.03?	625.233 +4.36?	695.931 +7.42?	766.629 +9.92?	838.589 +14.6?
D2	73.2158 -4.73?	146.432 -4.73?	220.086 -1.28?	293.301 -2.14?	367.394 +1.48?	440.829 +1.31?	515.36 +4.87?	589.453 +6.25?	664.641 +10.18?	739.83 +13.32?	815.457 +16.81?	890.865 +19.29?
D#2	77.3853 -8.84?	155.26 -3.38?	232.523 -6.11?	310.642 -2.7?	387.905 -4.47?	465.779 -3.38?	544.876 +1.29?	622.995 +2.06?	702.336 +5.68?	780.822 +6.68?	860.407 +9.71?	940.482 +13.13?
E2	82.0508 -7.49?	164.102 -7.49?	246.583 -4.47?	328.921 -3.71?	411.402 -2.66?	493.739 -2.45?	576.794 -0.16?	660.136 +2.32?	744.052 +5.57?	827.537 +7.28?	912.313 +11.12?	997.09 +14.31?
F2	86.8265 -9.55?	173.957 -6.52?	261.544 -2.49?	348.827 -1.99?	436.261 -1.08?	523.088 -2.49?	610.979 -0.48?	699.326 +2.16?	785.088 -1.48?	876.78 +7.35?	966.04 +10.18?	1056.21 +14.04?
F#2	91.9479 -10.33?	184.156 -7.89?	276.623 -5.45?	368.831 -5.45?	461.298 -4.47?	554.025 -3.01?	647.012 -1.27?	740.518 +1.24?	834.284 +3.74?	928.57 +6.7?	1022.86 +9.12?	1117.66 +11.94?
G2	97.5987 -7.08?	195.746 -2.22?	293.482 -3.03?	391.629 -1.62?	489.227 -2.71?	587.511 -1.41?	686.481 +1.24?	785.862 +4.13?	885.379 +6.65?	985.171 +9.14?	1085.51 +12.05?	1186.26 +15.07?
G#2	103.234 -9.89?	207.236 -3.47?	311.046 -2.4?	414.856 -1.87?	518.283 -2.83?	622.86 -0.26?	727.821 +2.48?	832.207 +3.33?	938.128 +6.83?	1043.66 +8.99?	1150.16 +12.2?	1257.04 +15.4?
A2	109.361 -10.08?	219.307 -5.45?	329.643 -1.87?	439.59 -1.61?	549.926 -0.23?	659.677 -0.84?	770.403 +0.91?	882.298 +4.52?	993.024 +5.29?	1104.92 +7.73?	1218.18 +11.67?	1330.47 +13.68?
A#2	115.706 -12.44?	232.675 -3.02?	348.855 -3.8?	465.351 -3.02?	582.478 -0.67?	698.658 -1.45?	816.258 +1.01?	933.859 +2.85?	1051.93 +5.06?	1170.95 +8.22?	1289.19 +9.75?	1409.63 +13.74?
B2	123.074 -5.57?	246.49 -3.17?	369.563 -3.97?	493.321 -1.97?	617.078 -0.77?	741.007 +0.43?	865.448 +2.31?	989.89 +3.72?	1115.19 +6.15?	1240.99 +8.8?	1367.32 +11.61?	1493.81 +14.15?
C3	130.389 -5.61?	261.693 +0.45?	392.692 +1.12?	523.081 -0.56?	654.384 +0.85?	785.992 +2.46?	917.6 +3.61?	1050.43 +6.49?	1182.64 +7.82?	1316.08 +10.5?	1449.82 +13.05?	1584.78 +16.5?
C#3	137.462 -14.16?	276.483 -4.37?	414.88 -3.72?	553.278 -3.39?	691.363 -3.98?	830.072 -3.07?	969.405 -1.31?	1109.67 +1.48?	1249.32 +2.77?	1389.9 +4.97?	1531.1 +7.48?	1672.92 +10.21?
D3	146.502 -3.9?	293.308 -2.1?	440.42 -0.3?	587.531 +0.6?	734.642 +1.14?	881.753 +1.49?	1029.17 +2.26?	1176.28 +2.39?	1324.3 +3.69?	1472.94 +5.44?	1621.57 +6.87?	1770.51 +8.36?
D#3	155.115 -4.99?	310.646 -2.67?	466.039 -2.41?	621.57 -1.9?	777.239 -1.28?	932.908 -0.87?	1088.85 -0.14?	1245.08 +0.8?	1401.58 +1.87?	1558.49 +3.18?	1714.99 +3.84?	1873.02 +5.8?
E3	164.043 -8.12?	328.759 -4.56?	493.475 -3.38?	658.528 -1.91?	823.244 -1.73?	988.297 -1.02?	1153.69 -0.01?	1319.08 +0.75?	1484.47 +1.34?	1650.53 +2.52?	1811.54 -1.34?	1985.02 +6.34?
F3	173.983 -6.26?	348.892 -1.66?	523.8 -0.13?	699.171 +1.78?	874.542 +2.92?	1050.38 +4.45?	1227.6 +7.5?	1404.82 +9.78?	1583.43 +13.07?	1762.5 +16.15?	1942.5 +19.5?	2124.35 +23.79?
F#3	183.896 -10.33?	369.662 -1.55?	554.181 -2.53?	739.323 -1.55?	925.713 +1.36?	1112.1 +3.31?	1299.11 +5.52?	1486.75 +7.91?	1675.63 +11.05?	1865.76 +14.72?	2057.14 +18.76?	2249.14 +22.61?
G3	195.775 -1.97?	391.55 -1.97?	587.727 -0.78?	784.308 +0.7?	981.694 +3.01?	1179.48 +5.14?	1378.08 +7.68?	1577.48 +10.46?	1778.09 +13.79?	1979.9 +17.52?	2183.33 +21.83?	2387.57 +26.01?
G#3	206.993 -5.5?	413.985 -5.5?	621.475 -4.12?	829.463 -2.39?	1038.94 +1.14?	1247.43 +2.11?	1457.9 +5.16?	1669.38 +8.48?	1881.34 +11.52?	2094.8 +15.18?	2311.25 +20.41?	2527.2 +24.41?
A3	219.729 -2.13?	439.795 -0.8?	660.198 +0.52?	880.602 +1.19?	1102.69 +4.24?	1325.12 +6.71?	1548.56 +9.6?	1773.35 +13.09?	1999.16 +16.68?	2226.65 +20.85?	2456.17 +25.69?	2687.37 +30.79?
A#3	232.696 -2.87?	465.911 -0.93?	699.127 -0.29?	933.122 +1.48?	1167.38 +2.93?	1403.71 +6.46?	1640.57 +9.52?	1878.72 +13.02?	2119.48 +17.86?	2360.75 +22.1?	2604.63 +27.29?	2850.59 +32.87?
B3	246.266 -4.74?	493.394 -1.71?	740.951 +0.3?	988.94 +2.06?	1237.79 +4.32?	1487.93 +7.33?	1738.5 +9.9?	1992.52 +14.83?	2246.1 +18.31?	2503.13 +23.48?	2761.88 +28.78?	3023.22 +34.66?
C4	261.469 -1.03?	523.505 +0.85?	786.111 +2.72?	1049.28 +4.6?	1313.6 +7.23?	1579.04 +10.22?	1846.19 +13.95?	2115.05 +18.15?	2386.18 +23.05?	2660.16 +28.81?	2935.84 +34.52?	3213.79 +40.49?
C#4	276.395 -4.92?	553.749 -1.92?	831.104 -0.92?	1109.42 +1.08?	1389.65 +4.67?	1670.85 +8.05?	1953.96 +12.17?	2238.99 +16.73?	2528.82 +23.56?	2816.73 +27.83?	3110.4 +34.52?	3405.99 +41.05?
D4	293.461 -1.2?	587.402 +0.22?	881.823 +1.63?	1177.21 +3.75?	1474.03 +6.72?	1773.25 +11.04?	2073.92 +15.32?	2376.5 +19.92?	2683.89 +26.6?	2992.24 +32.47?	3304.92 +39.53?	3619.03 +46.08?
D#4	310.756 -2.06?	621.512 -2.06?	933.128 -0.46?	1245.17 +0.93?	1559.37 +4.15?	1875.71 +8.29?	2193.35 +12.25?	2515.28 +18.17?	2841.94 +25.65?	3168.6 +31.61?	3500.84 +39.24?	3834.38 +46.15?
E4	329.096 -2.79?	659.116 -0.36?	989.137 +0.45?	1321.47 +3.88?	1655.65 +7.88?	1991.21 +11.74?	2329.09 +16.21?	2671.13 +22.26?	3016.87 +29.07?	3364.91 +35.68?	3718.51 +43.66?	4077.18 +52.45?
F4	348.817 -2.04?	698.29 -0.41?	1048.42 +1.22?	1400.52 +4.47?	1754.92 +8.69?	2110.97 +12.84?	2470.62 +18.34?	2832.58 +23.85?	3203.07 +32.75?	3571.92 +39.04?	3947.34 +47.06?	4330.65 +56.86?
F#4	369.811 -0.85?	739.622 -0.85?	1110.68 +1.09?	1483.61 +4.25?	1859.03 +8.46?	2237.58 +13.68?	2617.37 +18.23?	3002.77 +24.87?	3396.28 +34.15?	3786.67 +40.12?	4187.04 +49.11?	4590.52 +57.75?
G4	391.691 -1.34?	784.275 +0.63?	1178.05 +3.04?	1574.2 +6.86?	1972.14 +10.71?	2373.64 +15.88?	2779.31 +22.16?	3187.66 +28.31?	3606.71 +38.23?	4026.36 +46.37?	4454.64 +56.37?	4885.58 +65.6?
G#4	414.798 -2.11?	829.597 -2.11?	1246.18 +0.37?	1664.53 +3.46?	2086.45 +8.26?	2511.93 +13.92?	2942.75 +21.09?	3371.79 +25.53?	3822.2 +38.69?	4267.26 +46.97?	4719.44 +56.34?	5178.75 +66.48?
A4	[440.Hz]	881.137 +2.24?	1324.55 +5.96?	1769.1 +8.93?	2219.33 +15.15?	2671.83 +20.76?	3131.16 +28.53?	3595.04 +36.52?	4069.15 +47.08?	4547.8 +57.2?	5035.56 +68.58?	5525.58 +78.71?
A#4	465.705 -1.7?	932.19 -0.25?	1401.01 +3.13?	1872.18 +6.98?	2348.03 +12.74?	2828.55 +19.44?	3316.88 +28.28?	3803.65 +34.17?	4312.26 +47.54?	4819.3 +57.59?	5340.39 +70.33?	5863.04 +81.34?
B4	493.727 -0.54?	987.974 +0.37?	1484.82 +3.71?	1985.31 +8.55?	2490.49 +14.71?	3000.86 +21.81?	3518.52 +30.45?	4039.82 +38.46?	4578.81 +51.37?	5122.48 +63.21?	5678.64 +76.65?	6244.69 +90.51?
C5	524.062 +2.68?	1048.12 +2.68?	1577.04 +8.03?	2108.74 +12.97?	2645.3 +19.12?	3189.49 +27.35?	3739.93 +36.1?	4301.47 +47.11?	4872.73 +59.08?	5456.49 +72.57?	6045.1 +84.92?	6660.09 +102.01?
C#5	553.989 -1.17?	1110.06 +2.08?	1668.2 +5.31?	2231.55 +10.97?	2801.13 +18.21?	3376.94 +26.22?	3963.15 +36.47?	4556.64 +46.88?	5169.87 +61.56?	5793.5 +76.32?	6428.56 +91.39?	7072.98 +106.14?
D5	587.407 +0.23?	1176.73 +3.06?	1769.9 +7.76?	2367.86 +13.62?	2973.51 +21.6?	3588.75 +31.54?	4212.63 +42.15?	4849.95 +54.88?	5500.7 +68.94?	6164.89 +83.89?	6844.44 +99.92?	7544.15 +117.79?
D#5	621.43 -2.29?	1247.02 +3.5?	1875.39 +7.99?	2507.91 +13.1?	3151.54 +22.27?	3804.87 +32.78?	4467.92 +44.01?	5142.06 +56.13?	5838.39 +72.09?	6543.05 +86.96?	7274.06 +105.31?	
E5	659.347 +0.25?	1318.69 +0.25?	1986.36 +7.51?	2660.27 +15.2?	3342.5 +24.11?	4035.12 +34.49?	4738.15 +45.68?	5466.13 +61.94?	6200.36 +76.23?	6963.71 +94.83?	7735.37 +111.76?	
F5	698.14 -0.78?	1400.02 +3.85?	2104.39 +7.45?	2821.23 +16.91?	3545.56 +26.21?	4282.34 +37.44?	5036.58 +51.42?	5817. +69.65?	6607.4 +86.3?	7418.99 +104.47?	8262.99 +125.99?	
F#5	739.78 -0.48?	1481.48 +1.76?	2230.84 +8.46?	2990.74 +17.92?	3762.14 +28.87?	4549.84 +42.34?	5351.9 +56.55?	6177.93 +73.86?	7025.03 +92.41?	7900.89 +113.42?	8755.66 +126.26?	
G5	786.154 +4.78?	1576.47 +9.36?	2373.74 +15.95?	3179.34 +23.79?	4003. +36.3?	4840.54 +49.56?	5701.7 +66.16?	6590.64 +85.82?	7505.97 +107.05?	8437.96 +127.27?	9419.96 +152.86?	
G#5	831.792 +2.47?	1666.36 +5.35?	2510.62 +13.02?	3367.37 +23.26?	4239.37 +35.62?	5133.54 +51.3?	6049.9 +68.78?	6996.76 +89.34?	7975.5 +112.09?	8984.74 +135.97?		
A5	881.357 +2.67?	1766.12 +6.02?	2663.38 +15.27?	3573.13 +25.94?	4496.51 +37.57?	5451.69 +55.4?	6427.32 +73.55?	7442.7 +96.31?	8485.33 +119.37?	9539.33 +139.67?		
A#5	933.078 +1.4?	1873.96 +8.62?	2824.2 +16.77?	3790.04 +27.97?	4774.61 +41.46?	5793.51 +60.68?	6831.13 +79.04?	7920.24 +103.97?	9038.99 +128.8?			
B5	989.254 +2.61?	1988.5 +11.33?	2995.24 +18.57?	4024.46 +31.87?	5081.17 +49.19?	6165.35 +68.38?	7284.5 +90.29?	8453.62 +116.8?	9652.72 +142.53?			
C6	1048.61 +3.48?	2108.33 +12.63?	3180.55 +22.49?	4272.21 +35.29?	5391.65 +51.87?	6559.7 +75.72?	7761.09 +100.?	8983.3 +122.01?				
C#6	1110.51 +2.78?	2230.72 +10.32?	3367.56 +21.41?	4530.75 +37.02?	5735.54 +58.91?	6966.66 +79.92?	8253.24 +106.44?	9599.43 +136.85?				
D6	1178.43 +5.55?	2368.09 +13.79?	3577.73 +26.21?	4819.83 +44.09?	6096.88 +64.68?	7421.36 +89.38?	8797.03 +116.91?					
D#6	1246.25 +2.42?	2502.88 +9.62?	3784.44 +23.45?	5093. +39.53?	6451.41 +62.54?	7845.13 +85.52?	9319.85 +116.86?					
E6	1323.47 +6.5?	2656.92 +13.02?	4022.84 +29.22?	5421.22 +47.65?	6872.03 +71.89?	8382.78 +100.27?						
F6	1399.17 +2.8?	2817.04 +14.33?	4266.11 +30.86?	5756.75 +51.62?	7305.6 +77.8?	8920.98 +108.?						
F#6	1489.74 +11.38?	2994.47 +20.08?	4536.7 +37.33?	6128.92 +60.07?	7796.12 +90.31?	9528.32 +122.03?						
G6	1576.77 +9.68?	3171.02 +19.25?	4815.26 +40.5?	6509.47 +64.36?	8273.65 +93.23?							
G#6	1665.85 +4.83?	3361.63 +20.31?	5094.81 +38.19?	6892.84 +63.43?	8788.12 +97.67?							
A6	1769.53 +9.35?	3544.05 +11.79?	5420.89 +45.6?	7345.16 +73.47?	9376.75 +109.91?							
A#6	1872.73 +7.48?	3780.37 +23.55?	5750.35 +47.74?	7800.13 +77.51?								
B6	1986.02 +9.17?	3972.05 +9.17?	6110.85 +53.01?	8310.44 +87.22?								
C7	2108.65 +12.9?	4259.78 +30.25?	6498.35 +59.45?	8834.36 +93.06?								
C#7	2234.48 +13.24?	4461.48 +10.34?	6870.33 +55.81?	9381.31 +97.06?								
D7	2366.71 +12.77?	4798.33 +36.35?	7344.79 +71.42?									
D#7	2504.12 +10.48?	5072.96 +32.71?	7763.77 +67.47?									
E7	2659.56 +14.74?	5309.16 +11.49?	8130.43 +47.36?									
F7	2816.41 +13.94?	5724.95 +42.03?	8805.32 +85.41?									
F#7	2992.84 +19.13?	5988.17 +19.85?	9061.02 +34.97?									
G7	3175.77 +21.84?	6471.38 +54.2?										
G#7	3360.55 +19.75?	6726.09 +21.04?										
A7	3571.5 +25.15?	7138.01 +23.94?										
A#7	3782.77 +24.65?	7735.22 +63.04?										
B7	4010.53 +25.87?	8026.05 +26.94?										
C8	4259.89 +30.29?	8731.77 +72.84?										

Figure 7-3 Entropy Tuning for Upright Piano