# Low-Latency f0 Estimation for the Finger Plucked Electric Bass Guitar Using the Absolute Difference Function

**Christhian Fonseca [1][2], Tiago Tavares[1][2]**

[1]School of Electric and Computer Engineering (FEEC) – University of Campinas (Unicamp)
CEP 13083-852 – Av. Albert Einstein, 400 – Campinas – SP – Brazil

[2]Interdisciplinary Nucleus for Sound Studies (NICS – University of Campinas (Unicamp)
CEP 13083-872 – Rua da Reitoria, 165 – Campinas – SP – Brazil

christian_h@hotmail.com

***Abstract.*** *Audio-to-MIDI conversion can be used to allow digital musical control by means of an analog instrument. Audio-to-MIDI converters rely on fundamental frequency estimators that are frequently restricted to a minimum delay of two fundamental periods. This delay is perceptible for the case of bass notes. In this paper, we propose a low-latency fundamental frequency estimation method that relies on specific characteristics of the electric bass guitar. By means of physical modelling and signal acquisition, we show that the assumptions of the method relies on generalize throughout electric basses. We evaluate our method in a dataset with musical notes played by diverse bassists. Results show that our method outperforms the Yin method in low-latency settings, which indicates its suitability for low-latency audio-to-MIDI conversion of the electric bass sound.*

## 1 Introduction

Digital instruments and controllers commonly use the MIDI (Musical Interface for Digital Instruments) standard to communicate with each other. This allows combining different synthesizers, controllers, and effect racks, which expands the expressive possibilities related to timbres, musical performances, musical recordings and notations [1]. This toolchain can use analog instruments by means of audio-to-MIDI converters [2].

Audio-to-MIDI converters aim at identifying the notes played by the instrument. For such, they use a perceptual model in that relates the fundamental frequency (f0) of an audio signal to its pitch. There are many known algorithms that aim at estimating f0, such as the autocorrelation [3] and the Yin method [4].

f0 estimators commonly aim at finding periodicity in an audio signal $f_j$. The periodicity is based on the model

$$f_j = f_{j+kJ}, \tag{1}$$

where $J$ is the fundamental period of $f_j$ and $k \in \mathbb{Z}$. Methods that rely on this property commonly require analyzing at least two fundamental periods of the signal. This incurs in a lower-bound for the latency of Audio-to-MIDI conversion that can be close to 50 ms for the lowest notes (41.2 Hz) in standard 4-string electric basses. These long delays can harm the use of basses as a MIDI controller.

In this work, we aimed at attenuating this problem using an f0 estimation method especially crafted for the electric bass. The method exploits specific properties of the electric bass waveform. Our method allows f0 estimation with an algorithmic latency of 1.1 times the fundamental period of the signal.

Experimental results show that the method is effective with an error rate of 15%. This is half of error rate of the baseline method (Yin).

## 2 Related work

Pitch is an auditory sensation often related to the perception of a repetition rate of a waveform [5]. The repetition rate is called Fundamental Frequency (f0) and can be used to decompose harmonic complex tones into sinusoidal harmonic components whose frequencies at multiple integers of the fundamental frequency $f_0$, that is:

$$f(t) = \sum_{m=1}^{M} a_m \cos(2\pi m f_0 t + \phi_m). \tag{2}$$

The relative harmonic amplitudes $a_m$ are commonly associated to timbre differences and the fundamental frequency $f_0$ is closely related to the sensation of pitch [6]. In this study, we assume that the fundamental frequency is the physical counterpart of the sensation of pitch, hence estimating the fundamental frequency is equivalent to finding the pitch of a signal.

There are several methods that aim at finding the pitch of periodic signals, as discussed next.

### 2.1 Autocorrelation

A common method for estimating pitch of periodic signals is by detecting the greatest positive peak of the autocorrelation function $r_t$ [3], which is calculated by:

$$r_t(\tau) = \sum_{j=t+1}^{t+W} f_j f_{j+\tau} \tag{3}$$

The autocorrelation $r_t(\tau)$ is a measure of the similarity between the signal $f_j$ and a temporally shifted version $f_{f+\tau}$ of itself. It presents peaks in values of $\tau$ that correspond to the fundamental periods of $f_j$.

### 2.2 Harmonic Sum Spectrum

The Harmonic Sum Spectrum is obtained by adding the Fourier transform of the signal is to sub-sampled versions of itself [7]. This corresponds to calculating:

$$Y(\omega) = \sum_{m=1}^{M} F(m\omega), \tag{4}$$

where $F$ is the frequency spectrum of $f_j$, $Y$ is the Harmonic Sum Spectrum and $M$ is the number of downsampling processes to execute. After this processing, the argument of the global maximum of $Y$ corresponds to the f0.

### 2.3 Yin method

The Yin method was proposed by Cheveigné and Kawahara [4]. It is based on the same premises as the autocorrelation method, with the addition of a series of modifications that reduce errors. Two very important modifications are the substitution of the autocorrelation function by the difference function shown in Equation 5, and the application of the a cumulative mean normalized difference function shown in Equation 6.

$$d_t(\tau) = \sum_{j=1}^{W} (f_j - f_{j+\tau})^2. \tag{5}$$

$$d_t^{'}(\tau) = \left\{ \begin{array}{ll} 1 & \text{, if } \tau = 0 \\ \frac{d_t(\tau)}{(1/\tau)\sum_{j=1}^{\tau} d_t(j)} & \text{, otherwise} \end{array} \right\} \tag{6}$$

The shortest period between the local minima of $d_t^{'}$ that are lower than a pre-defined threshold is yielded as the fundamental period of $f_j$.

### 2.4 Discussion

All the methods discussed in this section directly rely on the periodicity property as stated in Equation 1 or the harmonic series model shown in Equation 2. This allows them to be applicable for the general case of finding pitch in periodic signals, but bounds them to a minimum delay of twice the fundamental period.

In this work, we propose a pitch detection method that relies on specific characteristics of the plucked electric bass string. This restricts our method to signals generated by this specific instrument. However, it allows reducing the delay to $1.1$ times the fundamental period.

This reduction is critical for the real-time pitch detection in low-pitch notes. In these notes, general-purpose methods require a delay of around 50ms to work properly. Our method allows detecting the same pitch with a delay of around 30ms.

The method proposed by [2] also indicates to estimate f0 close to the theoretical minimum latency, i.e. the fundamental period of the lowest observable pitch, but with higher computational complexity.

The proposed method is based on specific properties of the plucked electric bass signal. These properties are analyzed using a physical model, which guide its generalization possibilities. Then, the proposed model is compared to the Yin method using a dataset containing recordings from electric bass guitars.

## 3 Time-domain Behavior of a Plucked String

This section discusses the properties of the plucked string signal that were used as basis for our f0 estimation method. These properties were inferred by analyzing the audio signal of an electric bass strings, as shown in Section 3.1, then the physical model discussed in Section 3.2 was used to generalize these results, as shown in Section 3.3.

### 3.1 Plucking an Electric Bass String

The traditional electric bass guitar is an electro-acoustic instrument with a body and neck made of wood and four metal string tuned to E, A, G and D, which are fixed in a metal bridge on the body and in the nuts of the neck. The neck has a fingerboard with 20 to 24 frets which divides it in tonal areas. The index and middle fingers of the right hand are used to pluck the strings and the fingertips of the left hand are used to hold the strings against the fretted fingerboard. This changes the free length of the string, which modulates its natural oscillation frequency.

There are magnetic pickups placed on the instruments body, under the strings. They convert the string transverse velocity at its position into an electric voltage. The string transverse velocity can be see as a wave which propagates from the pluck position along the string length, reflecting and inverting when reach the string end, as shown in Figure 1.

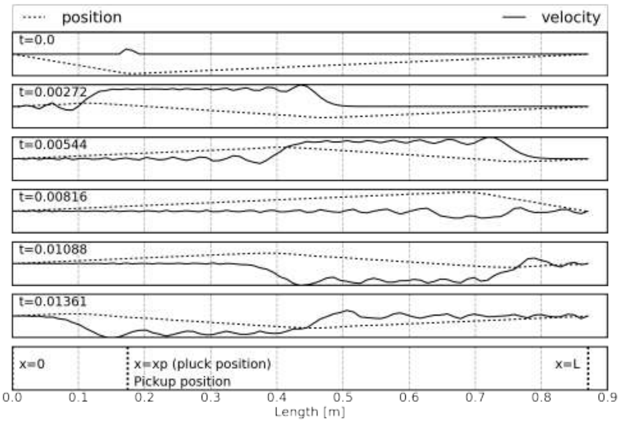

**Figure 1: Position and velocity of the string along the $x$ axis at different times $t$**

The waveforms of the voltage signal at the pickups, as shown in Figure 2, indicates repetitions of a peak at the beginning of each cycle. In order to confirm that this characteristic is maintained for all electric bass guitars (instead of being a characteristic of the specific instrument), the behavior of its string was mathematically modeled, as discussed in the next section.

### 3.2 Physical model

The behavior of the bass string can be modelled using an ideal string along the coordinate $x$ with fixed ends at $x = 0$ and $x = L$, which give us the following boundary conditions:

$$y(x = 0, t) = 0. \tag{7}$$

$$y(x = L, t) = 0. \tag{8}$$

The string has linear density $\mu$ and is stretched with a force $T$. It is initially at rest and is plucked in the position $x = x_p$ with amplitude $y(x_p, 0) = A$. In this situation, the initial transverse displacement $y(x, 0)$ can be expressed by

$$y(x, t = 0) = \left\{ \begin{array}{cc} A(\frac{x}{x_p}) & \text{, if } x < x_p \\ A(1 - \frac{x - x_p}{L - x_p}) & \text{, otherwise} \end{array} \right\} \tag{9}$$

and the velocity distribution $y'(0, x)$ is

$$y'(x, t = 0) = 0. \tag{10}$$

For a short segment of this string between $x$ and $\Delta x$ there is a slope $\delta y/\delta x = \tan(\theta)$ and a vertical force $F$ defined by:

$$F = T \sin(\theta)(x + \Delta x) - T \sin(\theta(x)) \tag{11}$$

If $y$ corresponds to a small displacement, $\theta$ is also small and can be approximated using $\cos(\theta) \approx 1$ and $\sin(\theta) \approx \tan(\theta)$. This allows re-writing Equation (11) as:

$$F = T(\frac{\partial y}{\partial x}(x + \Delta x) - \frac{\partial y}{\partial x}(x)) \tag{12}$$

Using the Newton's second law:

$$F = m\frac{\partial^2 y}{\partial t^2} \tag{13}$$

and knowing that the mass for this string segment is $m = \mu \Delta x$, we have:

$$T(\frac{\partial y}{\partial x}(x + \Delta x) - \frac{\partial y}{\partial x}(x)) = \mu \Delta x \frac{\partial^2 y}{\partial t^2} \tag{14}$$

dividing both sides of Equation (14) by $\Delta x$, and making $c = \sqrt{T/\mu}$, it becomes the wave equation:

$$\frac{\partial^2 y}{\partial t^2} = c^2 \frac{\partial^2 y}{\partial x^2}, \qquad x \in (0, L), t \in (0, T] \tag{15}$$

This model was used to simulate plucked strings and the resulting waveforms were compared to measured waveforms, as discussed in Section 3.3.

### 3.3 Plucked string simulation

Equation 15 was numerically solved using the finite difference method [8] and the algorithmic steps used by Langtangen [9]. The Taylor series expansion was used to approximate it as:

$$\begin{gathered} \frac{y(x + \partial x, t) - 2y(x, t) + y(x - \partial x, t)}{\partial x^2} = \\ \frac{1}{c^2} \frac{y(x, t + \partial t) - 2y(x, t) + y(x, t + \partial t)}{\partial x^2} \end{gathered} \tag{16}$$

Using the $i, j$ notation such that $y(x, t) = y_{ij}$, inserting the wave number $C = \frac{c\partial t}{\partial x}$ and rearranging Equation 16 yields:

$$y_{i,j+1} = C^2 y_{i-1,j} + 2(1 - C^2) y_{i,j} + C^2 y_{i+1,j} - y_{i,j-1}. \tag{17}$$

To calculate the value of this function in the first time step, $y_{i,j-1}$ must be determined. This can done using the initial velocity in Equation 10 and Tailor's series as follows:

$$\frac{y(x, t + \partial t) - y(x, t - \partial t)}{2\partial t} = 0. \tag{18}$$

Rearranging equation 18 and rewriting in the $i, j$ notation, we find that:

$$y_{i,j-1} = y_{i,j+1}. \tag{19}$$

Finally, replacing $y_{i,j-1}$ by $y_{i,j+1}$ in Equation 17, isolating $y_{i,j-1}$ and dividing both sides by 2, we have:

$$y_{i,j+1} = \frac{C^2}{2} y_{i-1,j} + (1 - C^2) y_{i,j} + \frac{C^2}{2} y_{i+1,j}, \tag{20}$$

which is the finite difference scheme. The numerical simulation was executed over the discrete spatial domain [0,L] equally spaced by $\partial x$ and over the discrete temporal domain $[0, T]$ equally spaced by $\partial t$.

The model's pluck position $x_p = L/5$ and the string length $L = 0.87m$ were directly measured from the strings of an electric bass. The wave velocity $c$ was calculated using $c = f/(2L)$ [10] related to note E0. The simulation time was define as $T = 0.05s$.

Over the spatial domain, the algorithm computes $y_{i,0}$ using Equation 9 and $y_{i,1}$ using Equation 20 and applying the boundary conditions from Equations 7 and 8. Then, for each element $j$ from temporal domain, apply Equation 17 to find $y_{i,j+1}$ for each element $i$ from the spatial domain, applying the boundary conditions again.

The output simulated signal was retrieved from the string velocity in the position $x = L/5$, approximately the pick up position and was yielded to a 5th order low-pass Butterworth filter with a $150Hz$ cutoff frequency. This simulates the smoother bend of the string due to its stiffness and the soft touch from the fingertip, which are responsible for generating tones with weaker high-frequency

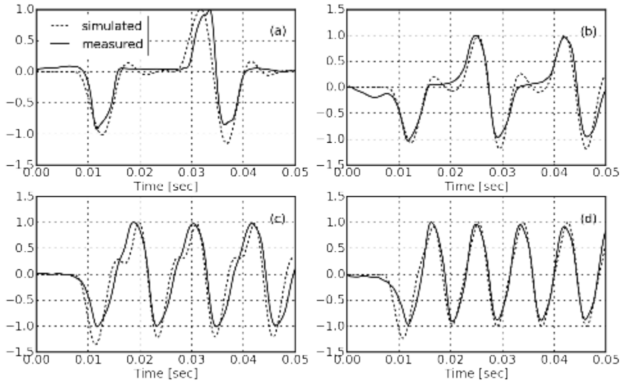

Figure 2: Simulated and measured notes played on string E of the electric bass guitar (a) E0 (b) A♯0 (c) F1 (d) A1

components [11]. The resulting signals were compared to the recorded signals, as shown in Figure 2.

Figure 2 shows that the physical model generates shapes that are similar to those found in the acquired signals. This means that the peak behavior is not a particular behavior of the specific electric basses that were used in our acquisitions. Rather, this behavior can be expected to appear in electric basses in general, hence it can be used for further steps in fundamental frequency estimation.

# 4 Fundamental Frequency Estimation

The simulated and measured waveforms in Figure 2 show that there is a peak at the onset of the note and in the beginning of each cycle after it. These peaks have approximately the same width, regardless of the note's frequency, and the note's fundamental frequency occurs due to the rate in which peaks appear in the signal. The proposed method is based on these two characteristics, as follows.

The proposed method consists of the application of the signal to an absolute difference function:

$$d_t(\tau) = \sum_{j=1}^{W} |f_j - f_{j+\tau}|, \tag{21}$$

with a window size $W$ shorter than half width of this first peak as shown in figure 4(a). This thin window plays an important role to make possible for the method find f0 after 1.1 times the fundamental period, whereas the Yin method needs more than two fundamental periods [4], as shown in Figure 3.

The f0 can be estimated from the interval between the two bigger dips in the absolute difference function, which are illustrated in Figure 4(b). In order to reduce error rates, the absolute difference function was normalized to the $[0, 1]$ range and a maximum threshold was applied. This allows ignoring the dips whose absolute value is too high.

If the specific string being played is known *a priori*, minimum and maximum frequencies can be used to reduce errors. They allow ignoring the dips that lead to frequency estimations out of this range.

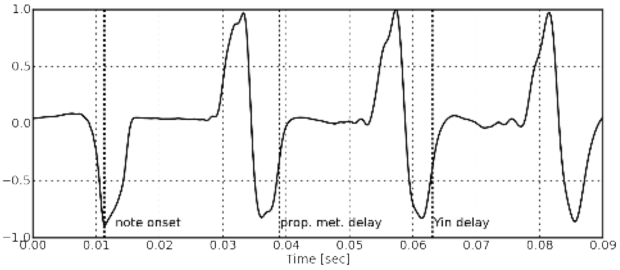

Figure 3: Algorithmic delay for the proposed method and for the Yin method.

Also, for frequencies higher than twice the minimum frequency parameter, there are more than two dips. In this case, the fundamental period is estimated from the average of the intervals between the dips of the difference absolute function, as the illustrated in Figure 4(d).

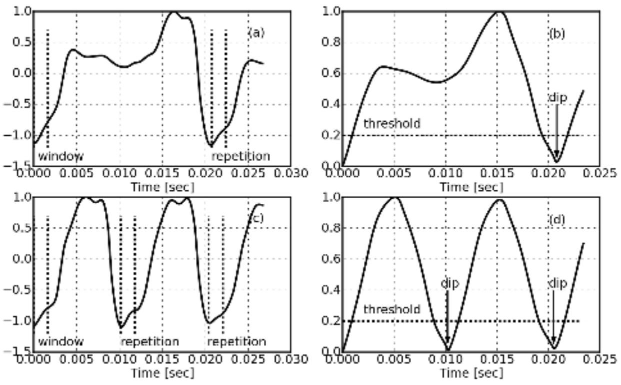

Figure 4: Sample analyzed $f(t)$ and window size $W$ for the E string notes: (a)$G0$ and (c)$G1$. Absolute difference function $d(\tau)$ and threshold: (b)$G0$ and (d)$G1$.

The next section discusses experiments regarding the proposed method.

# 5 Experiments and results

## 5.1 Dataset

The proposed method was tested using a set of audio recordings acquired from 3 different electric bass guitars. Each of them was played by a different musician, and all of them used the finger-plucking technique. All notes within the instrument's range were recorded from each of the guitars, using two different instrument equalizations (full bass and full treble). This yielded 528 recordings, which were all manually cropped to start at the note onset.

## 5.2 Experiments

This section describes experiments that compare the proposed method to the Yin method [4], as implemented by Guyot [12]. The experiments comprised executing both the proposed method and the Yin method to estimate the f0 in the dataset samples.

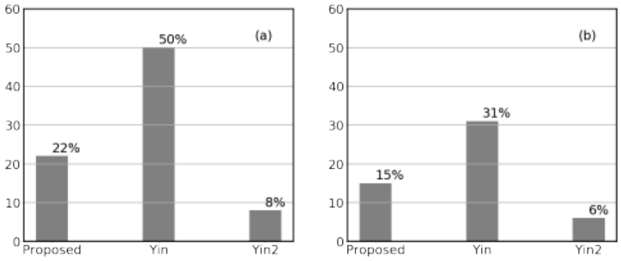

**Figure 5: (a) Test 1 error rates . (b) Test 2 error rates**

### 5.2.1 Test 1 - sample length for note

In this first test, the sample length provided as input parameters for the algorithms are equal to $1.1 \times T_{t1} \times fs$, being $T_{t1}$ the fundamental period of the expected note and $fs$ the frequency of sampling of the digital audio signal. To serve as reference, the test was repeated for the Yin method with a sample length equal to $2.1 \times T_{t1} \times fs$ and is referenced as ”Yin2” in figure 5 (a).

### 5.2.2 Test 2 - sample length for string

This second test is a more common application for a pitch detector in a string instrument, where the fundamental frequency should be estimated from a range of approximately 2 octaves. So, the sample length provided as input parameters for the algorithms are equal to $1.1 \times T_{t2} \times fs$, being $T_{t2}$ the fundamental period of the lower note from the specific string to which the recorded note belong. Also in this case, the test was repeated for the Yin method with a sample length equal to $2.1 \times T_{t2} \times fs$ and is referenced as ”Yin2” in figure 5 (b).

To determined if the method fails, the MIDI note correspondent to the fundamental frequency estimated is calculated as:

$$M_{note} = 12 \log \left( \frac{f_0}{16.351597} \right) \frac{1}{\log(2)} + 0.5, \qquad (22)$$

being $f_0$ the estimated fundamental frequency, 16.351597 the $f_0$ for the MIDI note = 0 and 0.5 as tolerance, as the result will be truncated. If the calculated MIDI note differs from the expected one, it is counted as one error.

### 5.3 Discussion

The error rates presented in Figure 5 show that the proposed method had less than half of Yin method's error rate, so having a better performance estimating f0 on both tests. As expected, the Yin method is a better solution when sample length is longer than 2 cycles of the fundamental period, but for the string E of a electric bass guitar, only the algorithmic delay should be higher than 50 ms ($2/f0 = 2,1/41.20Hz \approx 0,051s$), which is perceptible for a bass player, making it harder to play a bass guitar with real-time MIDI outputs.

The next section shows conclusive remarks.

## 6 Conclusion

A method based on the absolute difference function and on the waveforms from a finger plucked strings of a electric bass guitar was presented. It was tested over 528 notes recorded from three different bass guitars and it shows to be capable to estimate these notes from samples with length equal to 1.1 times their fundamental periods, while our reference method, Yin, under the same conditions, had double the error rate. This shorter algorithmic delay, near the minimal theoretical delay (one fundamental period) and low computational complexity, makes the proposed method suitable for real time applications for the electric bass guitar, such as a MIDI bass guitar.

However the method missed 15% of the notes on test 2, which is a similar application, so future studies should be made to improve this results. Also, new recordings in which the bass players always pluck the string smoothly in order to keep the first cycles of the signal similar to the modeled ones, can show an alternative way to a MIDI bass guitar. This imposes a limited way to play in exchange for a more precise note detection and lower latency. Lastly, the method was not tested for notes played on a vibrating string which certainly should make harder to estimate the correct f0. This case will be approached in future work.

## 7 Acknowledgements

Thanks are due to Mario Junior Patreze from ”Escola de Música de Piracicaba Maestro Ernst Mahle”, Marcio H. Goldschmidt and Giovani Guerra from musical studio ”Esgoto” for the records from their own electric bass guitars which compose the database of this work. This study was financed in part by the ”Coordenação de Aperfeiçoamento de Pessoal de Nível Superior” - Brasil (CAPES) - Finance Code 001.

## References

[1] Warren A. Gibson, J. *The MIDI Standard*. http://www.indiana.edu/ emusic/361/midi.htm, accessed 05/9/2019.

[2] O. Derrien. A very low latency pitch tracker for audio to midi conversion. *17th International Conference on Digital Audio Effects (DAFx-14)*, 2014.

[3] L. Rabiner. On the use of autocorrelation analysis for pitch detection. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 25(1):24–33, February 1977.

[4] Alain de Cheveigné and Hideki Kawahara. Yin, a fundamental frequency estimator for speech and music. *The Journal of the Acoustical Society of America*, 111(4):1917–1930, 2002.

[5] E. J. Heller. *Why You Hear What You Hear*. Princeton University Press, 2012. (Chapter 23; pp. 437-504).

[6] Andrew J. Oxenham. Pitch perception. *Journal of Neuroscience*, 32(39):13335–13338, 26 September 2012.

[7] A. M. NOLL. Pitch determination of human speech by the harmonic product spectrum, the harmonic surn spectrum, and a maximum likelihood estimate. *Symposium on Computer Processing in Communication, ed.*, 19:779–797, 1970.

[8] M.K. Jain. *Numerical Methods for Scientific and Engineering Computation*. New Age International, 1st ed. edition, 2003. ISBN-10: 8122414613. pp: 844.

[9] H.P. Langtangen. *Finite difference methods for wave motion*. Department of Informatics, University of Oslo, preliminary version edition, 2016.

[10] F. Iazzetta. *Tutoriais de Audio e Acustica*. http://www2.eca.usp.br/prof/iazzetta/tutor/acustica, accessed 04/25/2019.

[11] E. Jansson. *Acoustics for Violin and Guitar Makers*. http://www.speech.kth.se/music/acviguit4/, 4th ed. edition, 2002. (Chapter 4; pp. 16-18).

[12] Guyot P. Fast python implementation of the yin algorithm. *http://doi.org/10.5281/zenodo.1220947*, 2018. accessed 01/02/2018.