Tuning a musical instrument with vibrato system: a mathematical framework to study mechanics and acoustics and to calculate optimal tuning strategies

D. Hebenstreit[1]

School of Life Sciences, University of Warwick, Gibbet Hill Rd, CV4 7AL, Coventry, UK[a)]

String instruments such as electric guitars are often equipped with a 'vibrato system', which allows varying the pitch of all strings as a musical effect. It is usually based on a mobile bridge that is kept in balance by the strings and a coiled spring. Tuning such an instrument is complex, since adjusting the tension on one string will alter all other strings' tensions. In practice, a heuristic method is used, where all strings are repeatedly tuned to their desired pitch, which appears to reliably yield correct pitches after a while. It is unclear why this method works; an analysis is lacking. I present here a mathematical model that allows studying this subject in detail; the model captures the underlying mechanics and acoustics and can be used to simulate a typical tuning process. I verify the model with experimental data and show that it permits calculation of optimal tuning strategies that use the least number of adjustment steps.

[a)] d.hebenstreit@warwick.ac.uk;

I. INTRODUCTION

Electric guitars are among the most popular musical instruments. They are commonly equipped with a 'vibrato system'[9], also known as 'tremolo system/bar/arm' or 'whammy bar/arm'.

The function of a vibrato system is that it allows to reversibly and in a controlled fashion alter the pitch of all strings for the purpose of musical expression. It is usually constructed by replacing the fixed bridge of a normal guitar with a movable bridge that is kept under tension by a coiled spring. The spring counteracts the strings' tensions and keeps the bridge at an equilibrium position, where the string and spring forces balance each other. A lever that is attached to the bridge allows applying force to move it and so to either increase or decrease the strings' tensions and thus pitches. Releasing the lever returns the bridge to its original position. Softly and repeatedly varying the pitch of a note in both directions is used for expressivity and is generally known as *vibrato* in music; this gave the vibrato system its name since it can be used for this, albeit some older vibrato systems permit only detuning in one direction.

Guitars are not the only instruments with vibrato systems; another example would be the Vietnamese đàn bầu, which is basically a monochord where one end of the string is fixed to a flexible stick that can be bent for vibrato effects[11]. However, a peculiar aspect of instruments with vibrato systems is relevant only if multiple strings are used: changing the tuning of one string will change the tuning of all other strings, since all strings are fixed to the movable bridge, and the latter will change position upon any changes to the force balance. How can the instrument be tuned in light of this?

Practical experience suggests that repeatedly adjusting strings will eventually result in the desired pitches for all strings, as the adjustments become successively smaller. This approach is usually adapted by the average guitarist as a result of trial and error, assumption, or personal communication, etc. However, there does not appear to be literature that establishes this method and/or explains why it succeeds. While several works establish basic physical principles involved in the acoustics of string instruments and guitars in particular (e.g.[3,4,7,10,12]), none appears to discuss vibrato systems in detail.

In this work, I want to address this issue. I construct a mathematical model that describes the most important features of the mechanics and acoustics of a string instrument with vibrato system. Based on this model, I derive an algorithm that captures the typical tuning process of such an instrument and which allows following changes in the underlying mechanics. I present some results from an application of the algorithm to an example setting. The model represents a crucial first step towards understanding the tuning process of instruments equipped with a vibrato system; it will provide a useful starting point for further studies. I furthermore demonstrate how the presented framework can be used to pre-calculate tuning frequencies for each string, which allows achieving a defined overall tuning with single adjustments at each string.

II. BASICS

The acoustic behaviour of a vibrating string on a vibrato system guitar is mainly governed by three laws or principles from physics. *Mersenne's law* (or, more precisely, one of several *M.*'s laws)[8] relates the string's vibration frequency (denoted f) to the string's length, L_0, the stretching force F acting on it, and a material-specific constant, μ, that corresponds to the string's mass per unit length:

$$f = \frac{1}{2L_0}\sqrt{\frac{F}{\mu}} \tag{1}$$

The force F can be factorised using *Young's modulus* in the following way[1]:

$$F = \frac{E\,A\,l}{L_0}, \quad l \geq 0. \tag{2}$$

Here, E is Young's modulus (modulus of elasticity) and A is the string's cross sectional area. If the string is extended by length l beyond its original length L_0, the stretching force F results.

This is an approximation, but describes a guitar string well. Hence, stretching a string further (using a machine head) by a factor a, so that $\bar{l} = a\,l$, will increase its vibration frequency by $\sqrt{a}$, if the vibrating length is kept constant (e.g. by the 'nut' or by fretting the string at a fixed position).

The distinguishing feature of a guitar with vibrato system is its movable bridge, which is not fixed, but rather under tension by a coiled spring that counteracts the string's tension. The force exerted by this spring, F_{spring}, scales with its extension x and a constant k according to *Hooke's law*[6,13]:

$$F_{spring} = kx, \quad x \geq 0. \tag{3}$$

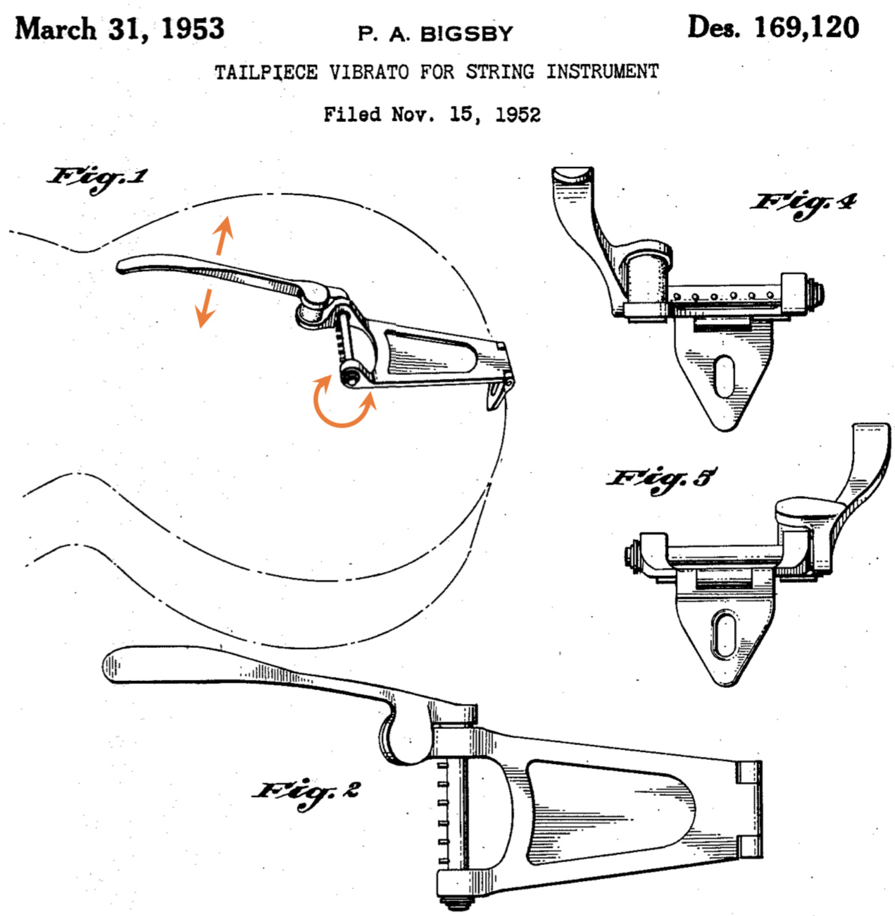

FIG. 1. Scheme illustrating the Bigsby vibrato system (color online). The image is taken from its patent application, with orange markings added to indicate movements. The arm can be moved vertically (i.e. in the direction normal to the soundboard), which turns the cylinder around which strings are wound, thereby changing their tensions.

This relation has the same form as Eq. 2, a force that increases linearly with extension. In this model, back- and forth movements of the bridge are only permitted along the same direction as the string. This is probably a good representation of a system such as the relatively simple 'Bigsby vibrato tailpiece'[2] (Fig. 1), but many other vibrato system designs exist (see Conclusions).

III. THE FORCE BALANCE

Let us adapt the facts above to a guitar with n strings ($n > 1$). Numbering the strings and assuming their vibrating lengths the same (L from nut to bridge *viz.* vibrato system), Eq. 1 becomes

$$f_j = \frac{1}{2L}\sqrt{\frac{F_j}{\mu_j}}, \quad 1 \leq j \leq n. \tag{4}$$

and Eq. 2 becomes

$$F_j = \frac{E_j\, A_j\, l_j}{L_j}, \quad l_j \geq 0. \tag{5}$$

Note that L corresponds to the vibrating length only, while the L_j denote the total original (unextended) string lengths. To simplify things, I now collect the constants in Eq. 5 into single constants $k_j := \frac{E_j\, A_j}{L_j}$, which capture the physical characteristics of each string that contribute to the pitch and frequency content aside from the tensions they are under. Thus, Eq. 5 and Eq. 4 become

$$F_j = k_j\, l_j, \tag{6}$$

and

$$f_j = \frac{1}{2L}\sqrt{\frac{k_j\, l_j}{\mu_j}} \tag{7}$$

respectively.

The combined forces of the strings and the spring coil of the tremolo balance each other:

$$\sum_{j=1}^{n} F_j = F_{spring} \tag{8}$$

which, following Eq. 3, further becomes

$$\sum_{j=1}^{n} k_j l_j = kx. \tag{9}$$

IV. PERTURBING THE SYSTEM

How will this system of balanced forces change if a string's tuning of the vibrato system guitar is changed? Let us assume we start with a situation where at least one $l_j > 0$ and we want to change the pitch of string i by adjusting l_i. Such a change will alter the combined string force and will thus move the bridge's position, which in turn alters tension of the strings, and so forth.

Let Δl be the change in l_i and Δx the resulting change in x, the spring coil's extension (Figure 2). The new vibrato system force, $\bar{F}_{spring}$, will become

$$\bar{F}_{spring} = k(x + \Delta x) \tag{10}$$

$$= F_{spring}\Big(\frac{x + \Delta x}{x}\Big) \tag{11}$$

$$= \sum_{j=1}^{n} F_j \Big(\frac{x + \Delta x}{x}\Big) \tag{12}$$

$$= \sum_{j=1}^{n} k_j l_j \Big(\frac{x + \Delta x}{x}\Big), \tag{13}$$

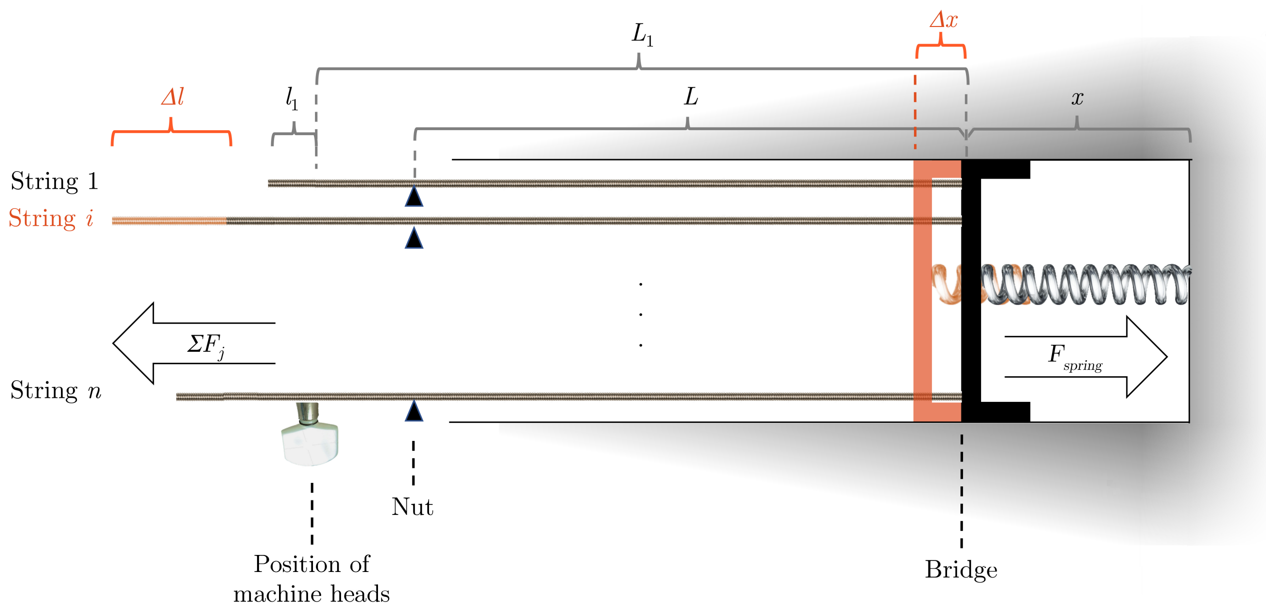

FIG. 2. Schematic overview of the model (color online). Three strings are shown as examples, string 1, i and n. The orange parts reflect the changing positions of elements upon extension by Δl of string i. Note that the strings may have different total lengths as indicated by the different endpoints of strings 1, i and n on the left, and that the string parts extending beyond the machine head position to the left correspond to the parts wound up at the machine head (l_1 in the case of string 1). L = vibrating length of all strings, L_1 = total original length (unextended) of string 1, l_1 = extension of string 1, i = index ('name') of string to be tuned, n = total number of strings and index/name of last string, j = index/name of any string not to be tuned (= not string i), Δl = extra extension of string i by tuning it, x = extension of coiled spring, Δx = resulting extra extension of coiled spring (i.e. change in the position of the bridge) by tuning string i, ΣF_j = combined force of strings pulling the bridge to the left, F_{spring} = force of coiled spring pulling the bridge to the right.

while the combined new string forces will become

$$\sum_{j=1}^{n} \bar{F}_j = \bar{F}_i + \sum_{j=1,\, j\neq i}^{n} \bar{F}_j \tag{14}$$

$$= k_i(l_i + \Delta l - \Delta x) + \sum_{j=1,\, j\neq i}^{n} k_j \max(l_j - \Delta x, 0). \tag{15}$$

The maximum function guarantees that a string's contribution to the total force disappears once it is relaxed to its original length. We skip the maximum function for the 'i' term and require $\Delta l > \Delta x - l_i$, as we are not interested in a complete detuning of string i.

Since strings and coiled spring must balance each other (Eq. 8), we get

$$k_i(l_i + \Delta l - \Delta x) + \sum_{j=1,\, j\neq i}^{n} k_j \max(l_j - \Delta x, 0) = \sum_{j=1}^{n} k_j l_j \Big(\frac{x + \Delta x}{x}\Big), \tag{16}$$

from Eq. 13 and 15. Here, $\Delta x > -x$, since the left hand side is strictly positive. This expression can be used to calculate Δx and thus the new balance of forces as a function of Δl. Before I do that, I make the following changes to the underlying assumptions to allow for a more powerful model:

1. Let us number the strings in order of length of the l_j, so that $l_j \leq l_k$ if $j < k$.

2. Let us permit some strings to be detuned even beyond complete relaxation. This means that the original lengths of the strings are significantly longer than the vibrating part, $L_j \gg L$, which is true in practice.

3. In line with the previous two points, I reinterpret the l_j as the machine head setting, i.e. $l_j > 0$ stretches the string, while some $l_j \leq 0$ are permitted (at least one must be positive) and correspond to (incomplete-) unspooling of a relaxed string by length $|l_j|$.

These assumptions permit handling better situations where some strings are completely relaxed. For instance, if one string's pitch is strongly decreased, then the decreasing Δx might make one or more other relaxed strings gain tension. The negative l_j then 'remember' when these strings will start contributing.

The assumptions require the following modification of Eq. 16:

$$k_i(l_i + \Delta l - \Delta x) + \sum_{j=1,\, j\neq i}^{n} k_j \max(l_j - \Delta x, 0) = \sum_{j=1}^{n} k_j \max(l_j, 0)\Big(\frac{x + \Delta x}{x}\Big). \qquad (17)$$

Δx is a function of Δl (and of the remaining string parameters, which I will not write down explicitly since adjusting Δl will not change these). Before I derive this function explicitly, I make some observations about these variables and Eq. 17.

Lemma 1. *Δl and Δx have the same sign, and $|\Delta l| > |\Delta x|$ if $\Delta l \neq 0$.*

Proof. In every situation discussed below, the first term on the left hand side of Eq. 17 must be positive, since we ruled out leaving string i completely relaxed.

Let us first assume $l_i > 0$.

If $\Delta l = 0$, the equation is satisfied for $\Delta x = 0$. If we started from this state and wanted to increase Δx to some value $\Delta x > 0$, then the right-hand side would strictly increase. This means that Δl must increase as well (-for the sake of the argument; this cannot be interpreted causally of course, since moving the bridge would not turn the tuning peg), and by a larger amount than Δx, as Δx contributes negatively to the left-hand side. Therefore we have $\Delta x > 0 \implies \Delta l > \Delta x$.

The converse would happen if we decreased Δx instead of increasing it, yielding $\Delta x < 0 \implies \Delta l < \Delta x$.

Furthermore, equivalent results are obtained if we repeated the considerations above for de- or increased Δl instead of Δx: $\Delta l > 0 \implies \Delta l > \Delta x > 0$ and $\Delta l < 0 \implies \Delta l < \Delta x < 0$.

By implication, $\Delta l = 0 \iff \Delta x = 0$. Thus, Eq. 17 is consistent with the notion that $\Delta l = 0$ leaves the tuning unchanged and therefore must result in $\Delta x = 0$.

Let us now assume $l_i \leq 0$.

Δl and Δx cannot both equal zero because the leftmost term must be positive. We can also rule out $\Delta x \leq 0$, since string i is completely detuned on the right-hand side, but not on the left-hand side, requiring positive Δx if the rightmost factor is considered. If $\Delta x > 0$, we must have $\Delta l > \Delta x + |l_i| > 0$. □

V. Δx AS EXPLICIT FUNCTION OF $\Delta l, h(\Delta l)$

Δx can be obtained as an explicit function of Δl from Eq. 17, $\Delta x = h(\Delta l)$. This will be a piecewise linear function; Δx will scale linearly with changing Δl as long as the number of strings under tension remains the same. However, continuously increasing Δl will eventually move the bridge forward enough to completely detune the other strings, one after another (and *vice versa* for decreasing Δl when starting from a situation where some strings are completely detuned). This will lead to kinks in the graph of $h(\Delta l)$ that reflect the maximum functions in Eq. 17, and each linear section will have a different slope that is greater than 0 and less than 1 (both strictly). This requires the distinction of many relatively complex cases (shown in the Appendix), which would successively apply if Δl was changed continuously.

VI. CHANGES IN VIBRATION FREQUENCIES

I now study how the vibration frequencies of all strings will change upon altering string i's tuning. Following Eq. 7, we get

$$\bar{f}_i = \frac{1}{2(L-\Delta x)}\sqrt{\frac{k_i(l_i+\Delta l-\Delta x)}{\mu_i}}, \tag{18}$$

and

$$\bar{f}_j = \frac{1}{2(L-\Delta x)}\sqrt{\frac{k_j \max(l_j-\Delta x, 0)}{\mu_j}}.$$

Lemma 2. *$\bar{f}_i$ is a strictly monotonic function of Δl.*

Proof. According to Lemma 1, Δl and Δx have the same sign, and $|\Delta l| > |\Delta x|$ if $\Delta l \neq 0$, which, together with the form of $h(\Delta l)$, proves that $\bar{f}_i$ is a strictly monotonically increasing function in an interval where $\Delta x < L$ (and $\Delta l - \Delta x > -l_i$, as before). □

The singularity of $\bar{f}_i$ at $\Delta x = L$ corresponds to the exploding frequency predicted by *Mersenne's law* if the vibrating part of the string becomes tiny. This will not actually occur, of course, as the law will not be a realistic model then anymore. Furthermore, increasing Δl anywhere near L is also usually prohibited by the vibrato system's design and by string i's tensile strength; the string will snap much earlier.

Corollary 1. *Because of Lemma 2, a bijection exists between the target tuning of string* i *and the length Δl it needs to be adjusted by to achieve this tuning.*

We can thus define a function $g(\Delta l) = |f_i^* - \bar{f}_i| = d$, that yields the distance d of the string's vibration frequency to a desired target frequency f_i^*, and its inverse function $g^{-1}(d) = \Delta l$. Since $\bar{f}_i$ depends on Δx and Δx implicitly depends on the other strings' parameters, so will g and g^{-1}.

VII. TUNING ALGORITHM

The above information can be combined into an algorithm (Figure 3) that mirrors the tuning procedure of a guitar with vibration system in practice: each string is successively tuned to its target pitch and, once the last string is tuned, the cycle restarts with the first string. This procedure is repeated for as many cycles as necessary until the instrument is perceived as fully tuned. The algorithm corresponds to a multi-step, multidimensional fixed-point iteration over the n independent variables l_j, and x (or L). Questions relating to its convergence properties appear non-trivial.

I implemented this algorithm in *Mathematica 11*. The code numerically calculates the required machine head adjustments through $\Delta l = g^{-1}(0)$ and reorders the strings internally at each step so that function $h(\Delta l)$ can be used in accordance with its definition.

VIII. EXPERIMENTAL SETUP

I test the tuning algorithm by comparing its predictions with experimental data obtained with an electric guitar. I chose three typical situations as test scenarios: (i) detuning of a guitar in standard E tuning to a 'D tuning' (each string is tuned one whole tone lower);

TABLE I. String parameters used in Figures 4 to 6. The bottom three strings are wound.

j	E_j, Pa	A_j, m^2	$\mu_j, kg/m$	String name	String gauge, in	String diameter, m
1	179×10^9	5×10^{-8}	4×10^{-4}	high E	0.010	2.54×10^{-4}
2	188×10^9	9×10^{-8}	7×10^{-4}	B	0.013	3.3×10^{-4}
3	178×10^9	1.5×10^{-7}	1.1×10^{-3}	G	0.017	4.32×10^{-4}
4	62×10^9	3.6×10^{-7}	2.3×10^{-3}	D	0.026	6.6×10^{-4}
5	42×10^9	6.8×10^{-7}	4.3×10^{-3}	A	0.036	9.14×10^{-4}
6	33×10^9	1.10×10^{-6}	7.0×10^{-3}	E	0.046	11.7×10^{-4}

(ii) tuning the low E string of a guitar in standard E tuning down by one whole tone to D, known as ‘Drop D tuning’; (iii) tuning a guitar to standard E tuning after restringing, i.e. starting with no tension on the strings (all tunings are equal temperament). The guitar I used was a ‘Jackson Kelly Standard’, which is equipped with a ‘Floyd Rose’ vibrato system. The latter has a more complex geometry than the model is based on, but I assumed it would behave roughly linear over a small range (see Conclusions section). For the strings’ properties I referred to[5] (Table 1). I further used the guitar’s nominal scale length of 25.5" to set $L = 0.6477\ m$. All parameters of the tuning algorithm are thus fixed, with the exception of x, which cannot be measured without specialized equipment. I thus decided to leave this as a single free parameter and determined its value based on the best fit to the experimental data (see next section).

Algorithm 1 Tuning algorithm for guitar with vibrato system

Input
$S \leftarrow (1\ 2\ ...\ n)$ ▷ String names
x^0 ▷ Initial x
L^0 ▷ Initial L
$K \leftarrow (k_1\ k_2\ ...\ k_n)$
$M \leftarrow (\mu_1\ \mu_2\ ...\ \mu_n)$
$\Lambda^0 \leftarrow (l_1^0\ l_2^0\ ...\ l_n^0)$ ▷ Initial l_j
$f_1^*, f_2^*,\ ...\ f_n^*$ ▷ Target vibration frequencies
ϵ ▷ Error tolerance

Start
$\Lambda \leftarrow \Lambda^0$
$x \leftarrow x^0$
$L \leftarrow L^0$

function $\sigma(T)$
  $T' \leftarrow$ reorder T, so that for $t_p, t_q \in T' : p < q \implies l_p \leq l_q$ ▷ order T (can be $\subset S$) by l_j
  return T'
end function

while $(\exists j :\ |f_j - f_j^*| \geq \epsilon)$ **do** ▷ Cycles
  for $i \leftarrow 1, i \leq n$ **do**
    $\Delta l \leftarrow g^{-1}(0, f_{\sigma(i)}^*, x, \sigma(\Lambda), \sigma(K), \sigma(M))$ ▷ 'Inverse distance' function for $d = 0$
    $\Delta x \leftarrow h(\Delta l, x, \sigma(\Lambda), \sigma(K))$
    $l_i \leftarrow l_i + \Delta l$
    $\Lambda \leftarrow (l_1, ..., l_i, ...\ l_n)$ ▷ Update l_j
    $x \leftarrow x + \Delta x$ ▷ Update x
    $L \leftarrow L - \Delta x$ ▷ Update L
  end for
  return Λ, x, L
end while

End

FIG. 3. Tuning algorithm. $\sigma(T)$ is the function that orders the set T handed over to it based on the l_j as shown. The other individual variables are used as in the main text, with some additional letters added to refer to sets of these.

The only readout of the experimental setup were the fundamental frequencies of the strings' vibrations, which I measured using the 'n-Track Tuner' app on an iPhone 7 Plus, after amplifying the guitar's sound with a 'Marshall G 15R CD' amplifier. Correct function of the n-Track Tuner was verified using an online tone generator (http://onlinetonegenerator.com) and the `Play[]` function in *Mathematica 11*, confirming 0.1 Hz precision of the app.

IX. COMPARISON OF TUNING ALGORITHM AND EXPERIMENTAL RESULTS

As the first test scenario, I tuned the guitar to standard guitar tuning, i.e. string 1 to 6 were tuned to E_4 (329.6 Hz), B_3 (246.9 Hz), G_3 (196 Hz), D_3 (146.8 Hz), A_2 (110 Hz), and E_2 (82.4 Hz), respectively (Figure 4). I then successively tuned each string, from high to low strings, to its target frequency (Figure 4) in accordance with the tuning algorithm and measured the remaining strings' frequencies at each step for four full cycles. I carried out the experiment a total of three times at different days. I then determined x by minimizing the mean square deviations between the algorithm's output and the experimental data using the bisection method, obtaining a value of $x = 0.005\ m$. As an overlay of the algorithm's predictions (lines) on the experimental data (data with error bars) demonstrates, the agreement is excellent (Figure 4a). Output at each step of the algorithm demonstrates how machine head settings (Fig. 4b), L (Fig. 4c), and Δx (Fig. 4d) begin to converge after four cycles.

I repeated this approach for the second scenario, the 'Drop D tuning'. I started with standard tuning and then used three tuning cycles to tune the low E string to D while repeatedly tuning back the other strings (high to low) to their nominal standard pitches. I used the value for x as determined before. Since only a single string is being detuned, the frequency changes are much smaller. Agreement between experiment and theory was excellent again (Fig. 5).

Standard E to D tuning

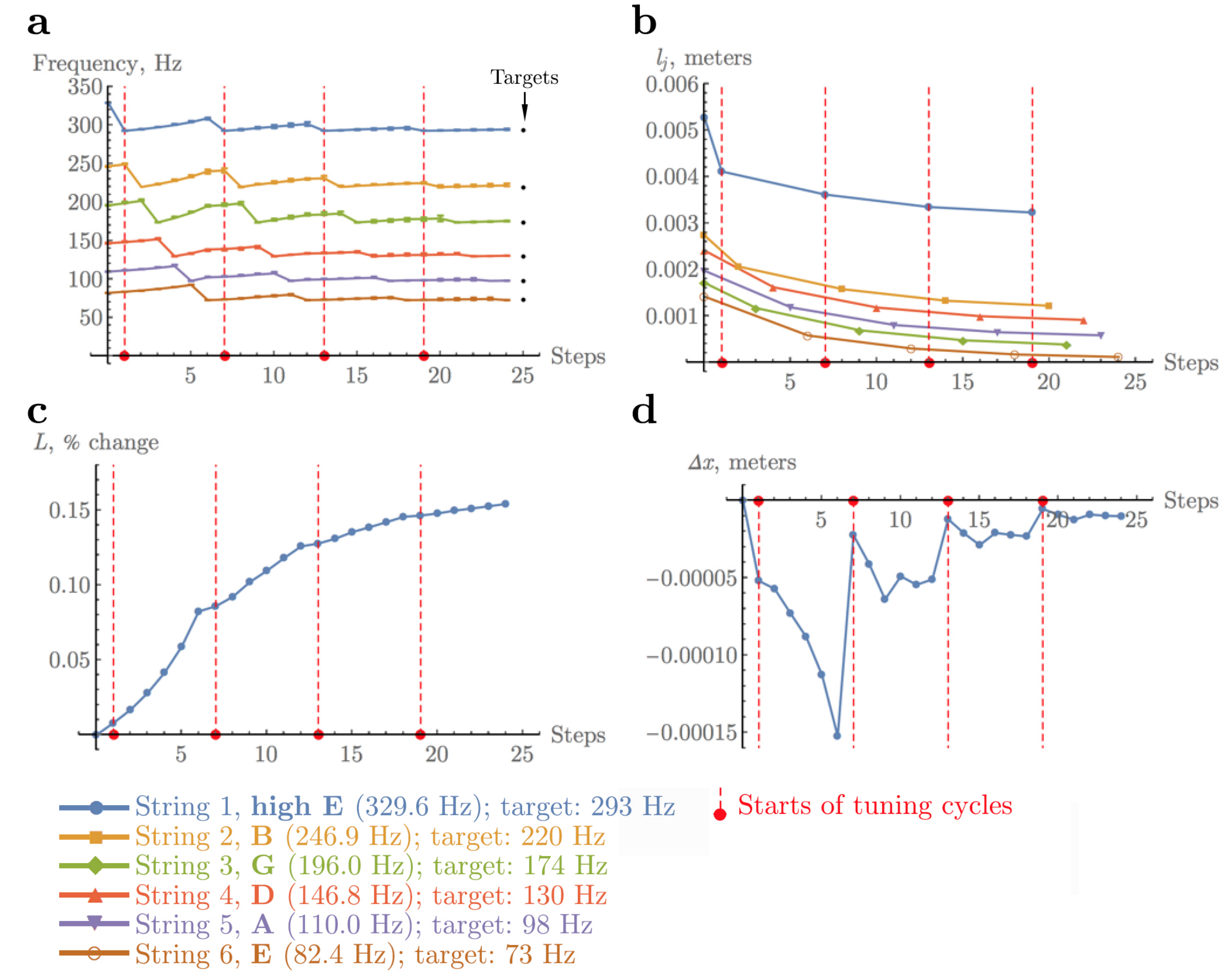

FIG. 4. Comparison of the tuning algorithm with experimental data for detuning a guitar from standard E tuning to D tuning (color online). Parameters used for the algorithm are shown in Table 1 and in the main text, while the value for x was derived from the best fit to the experimental data. (a), Overlay of experimental data (data with error bars) and predictions of the tuning algorithm (lines). The (tiny) error bars correspond to the data range of three independent experiments (i.e. maxima and minima). Individual strings are distinguished by colour, as indicated below figure. Target frequencies are indicated by black dots in (a) and are shown below the figure. The beginning of each tuning cycle at string 1 is indicated by red dots and dashed, red, vertical lines. (b), (c), and (d) show the algorithm's predicted machine head settings, l_j, the relative change in L, and Δx, respectively, during the procedure.

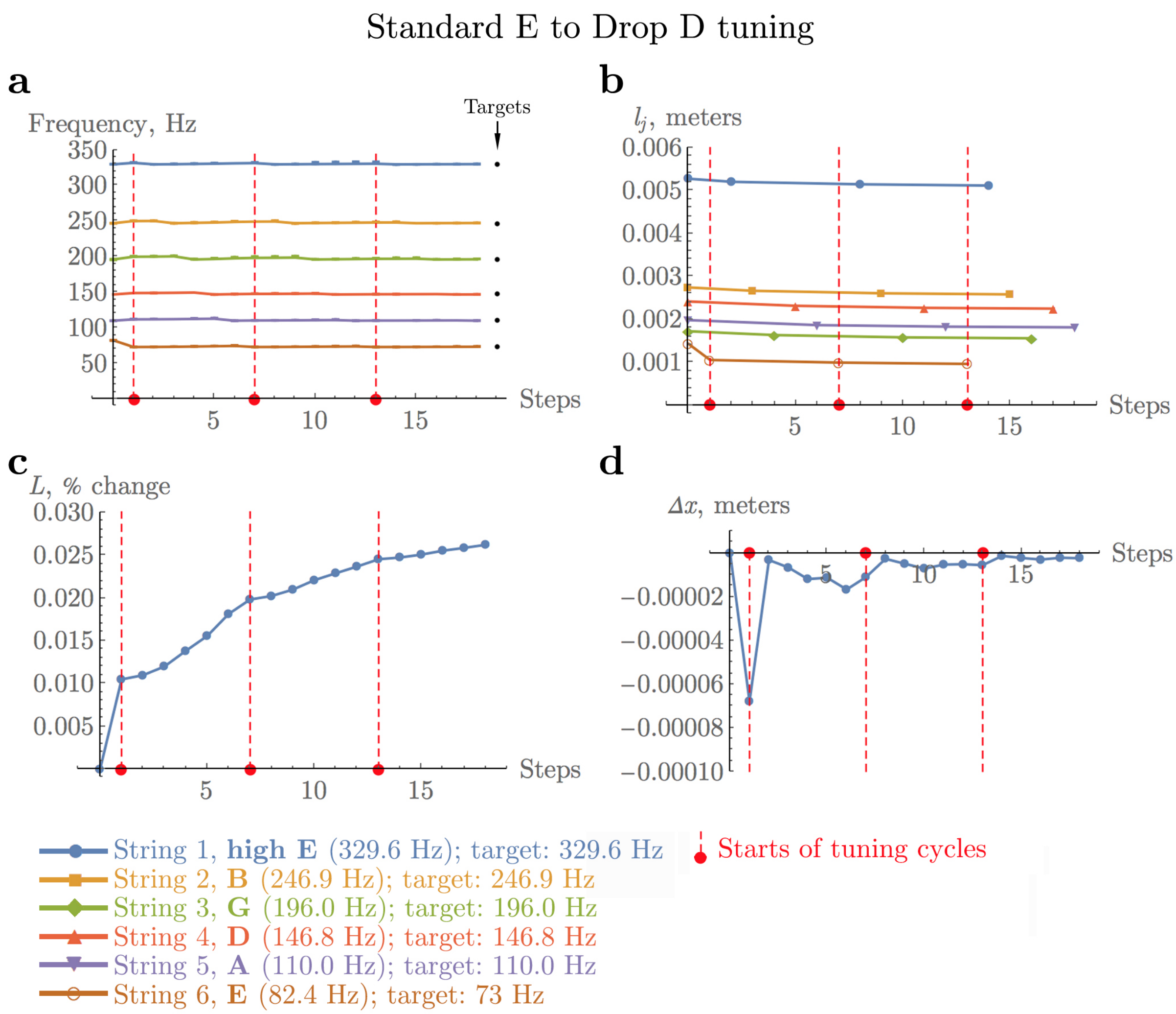

FIG. 5. Comparison of the tuning algorithm with experimental data for establishing a Drop D tuning from standard E tuning (color online). Panels and labels are equivalent to those of Figure 4. The (tiny) error bars in (a) denote the range of values from two independent experiments.

As the third test scenario, I detuned the guitar so that all strings were completely relaxed, and applied the tuning algorithm to re-establish standard E tuning (two independent experiments; strings were tuned from high to low in each cycle as before). This simulates the common situation of restringing the instrument. Again, I obtained the value for x based on which x yielded the best fit of the algorithm's output to the data. Agreement between the tuning algorithm's predictions and the experimental data is good again (Figure 6) but worse than with scenarios 1 & 2. This is probably due to the non-linear behaviour of strings and vibration system at very low tensions. Interestingly, convergence was achieved much faster in this situation (Figure 6).

These results demonstrate that the algorithm captures properties of a real instrument well. While its predictions are somewhat less precise at very low tension forces, it yields excellent fits when the bridge is close to its centre position.

X. TUNING STRATEGIES

The computational implementation of the tuning algorithm provides a tool to quickly and efficiently study different tuning strategies. For all practical matters, the fewer adjustment steps are necessary to achieve a certain tuning, the better. An obvious variation of the strategy used in the situations above concerns the order of string adjustments. Instead of tuning from highest to lowest string in each cycle, the reverse order can be used. For this analysis, I counted the number of cycles necessary for each string to deviate less than 0.1 Hz from its target frequency.

The tuning algorithm predicts that the specifics of the situation determines which strategy is sensible; both strategies perform equally for the D tuning scenario, while restringing is quicker when adjustments are made from low to high string in each cycle. I also tested a strategy where adjustments are performed in random order in each cycle. In 100 trials each, the average random strategy takes longer than both 'ordered' strategies in the D

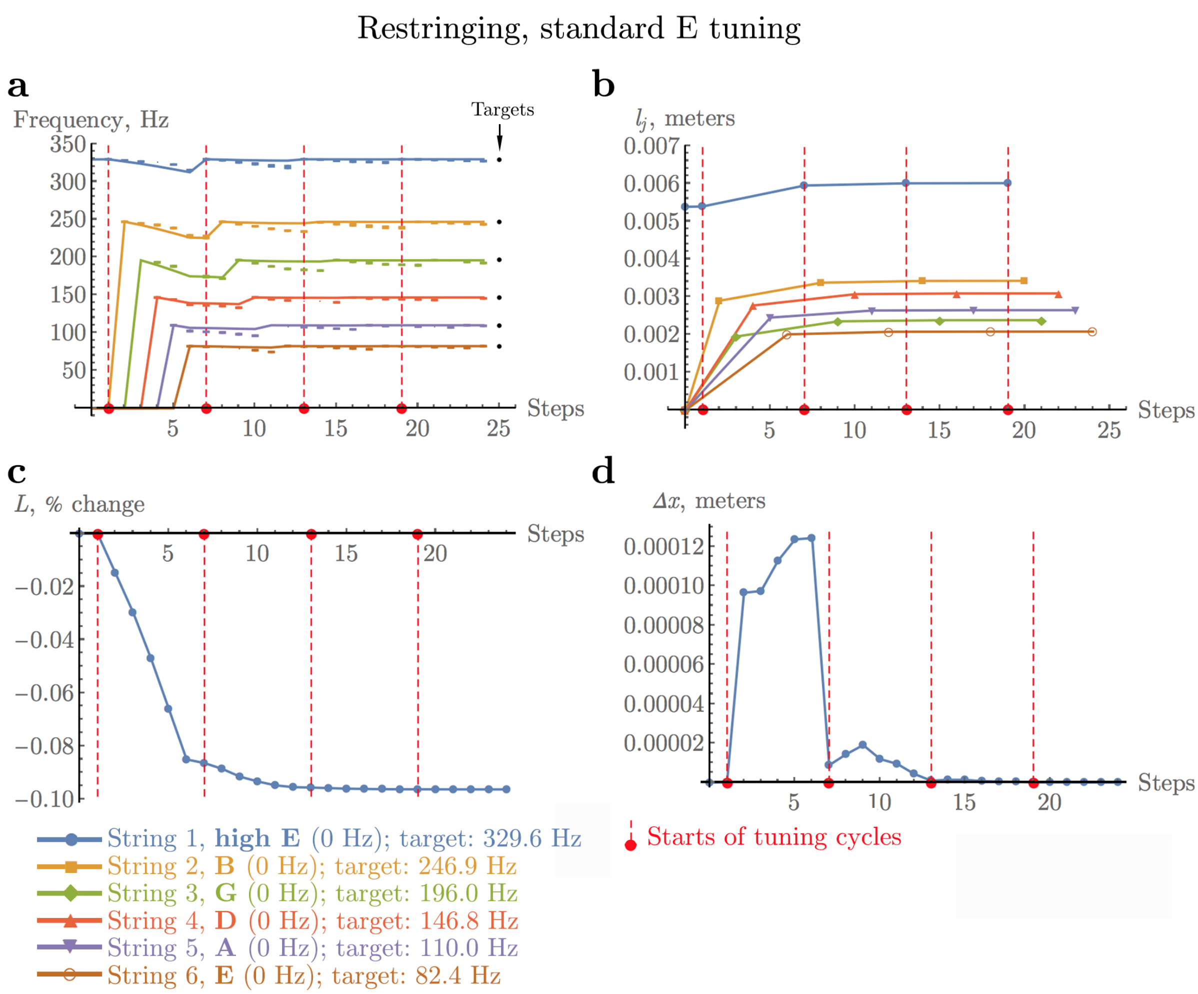

FIG. 6. Comparison of the tuning algorithm with experimental data for establishing standard E tuning after restringing (color online). Panels and labels are equivalent to those of Figures 4 & 5. The (tiny) error bars in (a) denote the range of values from two independent experiments.

tuning setting, but slightly outperforms the slower high-to-low strategy in the restringing case (Figure 7a).

Finally, an optimal tuning strategy can be devised. Letting the algorithm run to convergence yields the final machine head settings $\bar{l}_j$ for a desired tuning. This allows calculating the frequency each string needs to be tuned to in each step of a single cycle, if the order of string adjustments is decided on in advance. In other words, for each tuning step, Δl can be calculated from $\Delta l = \bar{l}_i - l_i$, which in turn yields $\Delta x = h(\Delta l)$. Δl and Δx can then be inserted into Eq. 18 to obtain the frequency $\bar{f}_i$ the string needs to be tuned to. I used this procedure to pre-calculate frequencies each string needs to be tuned to if using the high-to-low tuning order for the D tuning scenario (Figure 7b). The predicted string frequencies at each step are shown in Figure 7c, theoretically achieving the target tuning in a single cycle.

To test this in practice, I applied the exact tuning strategy based on these figures to the guitar and measured the final frequency of each string at the end. The results demonstrate that this strategy indeed achieves the desired tuning in a single cycle, with only minor deviations from the target frequencies remaining (Figure 7d).

XI. CONCLUSIONS

I have introduced here a framework that allows exploring the acoustic, mechanical, and procedural aspects of the tuning of an instrument with a vibrato system. I illustrate its application based on experimental examples, which demonstrate how the main features of a real tuning process are captured by the model. The underlying algorithm is also relevant from a mathematical viewpoint and represents an interesting case of a relatively complex fixed-point iteration.

The presented framework can be used to find optimal tuning strategies as demonstrated and could be helpful in the design of future instruments. This paper can also serve as a starting point for further work in this direction; many different designs for vibrato systems

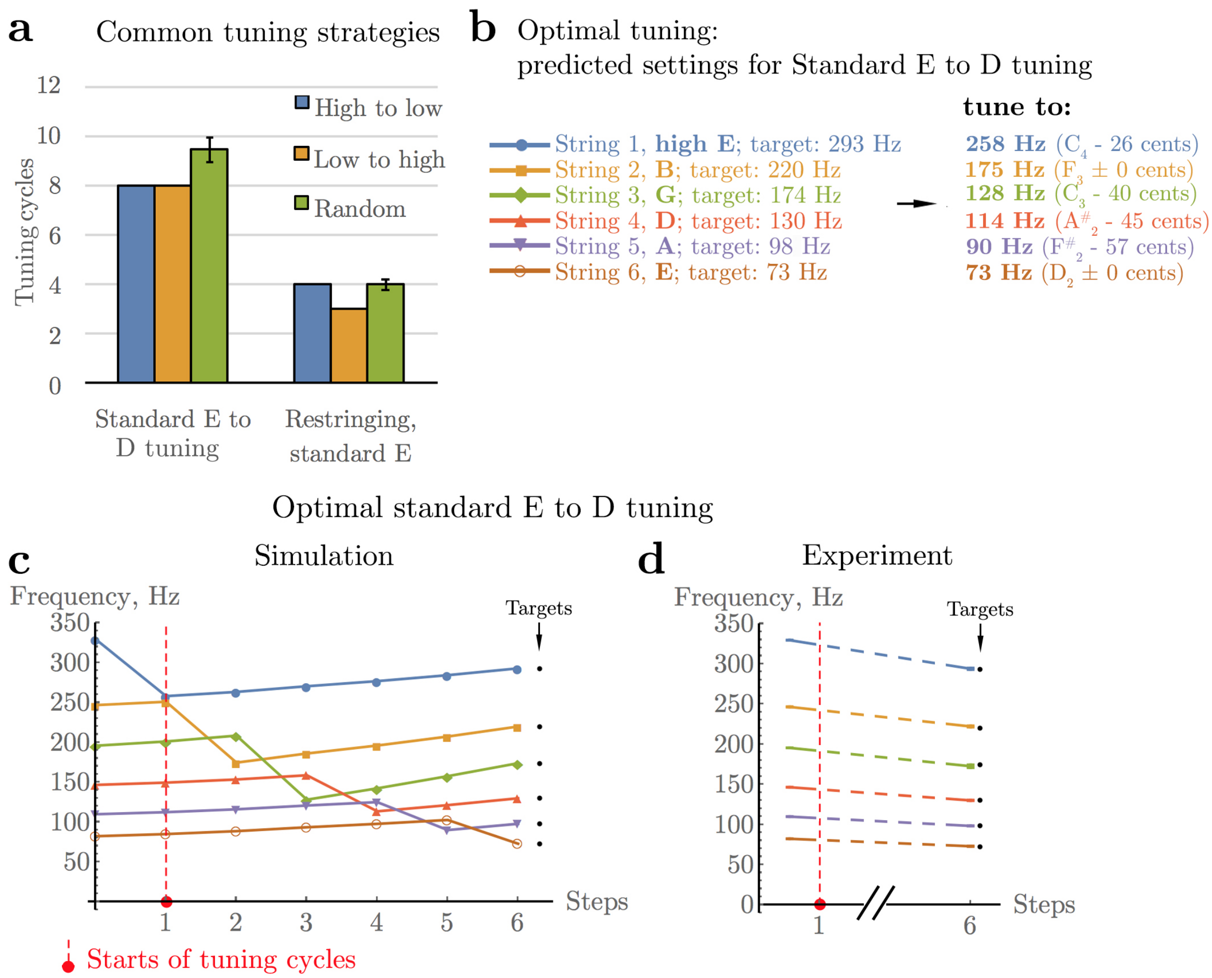

FIG. 7. Comparison of tuning strategies based on computational predictions of the tuning algorithm (color online). (a) Three different strategies were used in both test scenarios; 'High to low' and vice versa correspond to ordered adjustments in each cycle, while 'Random' corresponds to randomly unordered adjustments. The latter is shown as the average of 100 trials, with the error bars denoting the standard deviations from the average. (b) Pre-calculated target frequencies for an optimal tuning strategy for the D tuning scenario, using a high-to-low tuning order. (c) Predicted frequency changes of all strings at each step of the optimal strategy described in (b). (d) Experimental test of the optimal strategy shown in (b) and (c). The (tiny) error bars denote the range of values from two independent experiments.

exist and frequently have more complex geometries than the one assumed here; often, the bridge does not move in a linear, one-dimensional fashion, but rather pivots, leading also to minor vertical movements of the strings' endpoints, as it is the case for the guitar used in the experiments. It is straightforward to adapt the model presented here to the specifics of a particular instrument and/or vibrato system.

ACKNOWLEDGMENTS

I would like to thank Martin Hebenstreit, who read an earlier version of the manuscript and provided feedback.

APPENDIX: DERIVATION OF $\Delta x = h(\Delta l)$

To derive the piecewise linear function $\Delta x = h(\Delta l)$, I first distinguish cases depending on the magnitudes of Δx and l_j and which string is to be tuned. The l_j (at least one positive) are ordered as explained in the main text and $j \in \{1, ..., n\}$. Let $m \in \{1, ..., n+1\}$ be defined so that $l_j - \Delta x \leq 0$ for all $j < m$, and $l_j - \Delta x > 0$ for all $j \geq m$.

If $i \geq m$, Eq. 17 becomes:

$$k_i(l_i + \Delta l - \Delta x) + \textstyle\sum_{j=m,\, j\neq i}^{n} k_j(l_j - \Delta x) = \sum_{j=1}^{n} k_j \max(l_j, 0)\left(\frac{x+\Delta x}{x}\right),$$

which further becomes:

$$k_i l_i + k_i \Delta l - k_i \Delta x + \textstyle\sum_{j=m,\, j\neq i}^{n} k_j l_j - \sum_{j=m,\, j\neq i}^{n} k_j \Delta x = \sum_{j=1}^{n} k_j \max(l_j, 0) + \frac{\Delta x}{x} \sum_{j=1}^{n} k_j \max(l_j, 0).$$

We can collect the Δx terms and rearrange this to get:

$$
\begin{aligned}
\Delta x \left[k_i + \sum_{j=m,\, j\neq i}^{n} k_j + \frac{1}{x}\left(\sum_{j=1}^{n} k_j \max(l_j, 0)\right)\right] &= k_i \Delta l + k_i l_i + \sum_{j=m, j\neq i}^{n} k_j l_j - \sum_{j=1}^{n} k_j \max(l_j, 0) \\
\Delta x \left[\sum_{j=m}^{n} k_j + \frac{1}{x}\left(\sum_{j=1}^{n} k_j \max(l_j, 0)\right)\right] &= k_i \Delta l + \sum_{j=m}^{n} k_j l_j - \sum_{j=1}^{n} k_j \max(l_j, 0) \\
\Delta x \left[\sum_{j=m}^{n} k_j + \frac{1}{x}\left(\sum_{j=1}^{n} k_j \max(l_j, 0)\right)\right] &= k_i \Delta l - \sum_{j=1}^{m-1} k_j \max(l_j, 0) + \sum_{j=m}^{n} k_j \min(l_j, 0) \\
\Delta x &= \frac{x\left[k_i \Delta l - \sum_{j=1}^{m-1} k_j \max(l_j, 0) + \sum_{j=m}^{n} k_j \min(l_j, 0)\right]}{x \sum_{j=m}^{n} k_j + \sum_{j=1}^{n} k_j \max(l_j, 0)}.
\end{aligned}
$$

Similarly, if $i < m$, Eq. 17 becomes:

$$k_i(l_i + \Delta l - \Delta x) + \textstyle\sum_{j=m}^{n} k_j(l_j - \Delta x) = \sum_{j=1}^{n} k_j \max(l_j, 0)\left(\frac{x+\Delta x}{x}\right),$$

which further becomes:

$$k_i l_i + k_i \Delta l - k_i \Delta x + \textstyle\sum_{j=m}^{n} k_j l_j - \sum_{j=m}^{n} k_j \Delta x = \sum_{j=1}^{n} k_j \max(l_j, 0) + \frac{\Delta x}{x} \sum_{j=1}^{n} k_j \max(l_j, 0).$$

Collecting the Δx terms and rearranging yields:

$$
\begin{aligned}
\Delta x \left[k_i + \sum_{j=m}^{n} k_j + \frac{1}{x}\left(\sum_{j=1}^{n} k_j \max(l_j, 0)\right)\right] &= k_i \Delta l + k_i l_i + \sum_{j=m}^{n} k_j l_j - \sum_{j=1}^{n} k_j \max(l_j, 0) \\
\Delta x \left[k_i + \sum_{j=m}^{n} k_j + \frac{1}{x}\left(\sum_{j=1}^{n} k_j \max(l_j, 0)\right)\right] &= k_i \Delta l + k_i l_i - \sum_{j=1}^{m-1} k_j \max(l_j, 0) + \sum_{j=m}^{n} k_j \min(l_j, 0) \\
\Delta x &= \frac{x\left[k_i \Delta l + k_i l_i - \sum_{j=1}^{m-1} k_j \max(l_j, 0) + \sum_{j=m}^{n} k_j \min(l_j, 0)\right]}{x k_i + x \sum_{j=m}^{n} k_j + \sum_{j=1}^{n} k_j \max(l_j, 0)}.
\end{aligned}
$$

Given the premise, both of these expressions for Δx hold if $l_m > \Delta x \geq l_{m-1}$. Both sides of this inequality can be rearranged for both, $i \geq m$ and $i < m$, to yield boundaries for Δl, which define the individual linear sections of $h(\Delta l)$. I show this for the example $l_{m-1} \leq \Delta x$, $i \geq m$, $m > 1$, while the other boundaries can be derived in the same, simple way:

$$\frac{x\left[k_i\Delta l - \sum_{j=1}^{m-1} k_j \max(l_j,0) + \sum_{j=m}^{n} k_j \min(l_j,0)\right]}{x\sum_{j=m}^{n} k_j + \sum_{j=1}^{n} k_j \max(l_j,0)} = \Delta x \geq l_{m-1}$$

$$xk_i\Delta l - x\left[\sum_{j=1}^{m-1} k_j \max(l_j,0) - \sum_{j=m}^{n} k_j \min(l_j,0)\right] \geq l_{m-1}\left[x\sum_{j=m}^{n} k_j + \sum_{j=1}^{n} k_j \max(l_j,0))\right]$$

$$xk_i\Delta l \geq l_{m-1}\left[x\sum_{j=m}^{n} k_j + \sum_{j=1}^{n} k_j \max(l_j,0))\right] + x\left[\sum_{j=1}^{m-1} k_j \max(l_j,0) - \sum_{j=m}^{n} k_j \min(l_j,0)\right]$$

$$\Delta l \geq \frac{l_{m-1}\left[x\sum_{j=m}^{n} k_j + \sum_{j=1}^{n} k_j \max(l_j,0))\right] + x\left[\sum_{j=1}^{m-1} k_j \max(l_j,0) - \sum_{j=m}^{n} k_j \min(l_j,0)\right]}{xk_i}.$$

The expressions for Δx can further be inserted into the additional assumption of $\Delta l > \Delta x - l_i$, which adds another condition for Δl for each case. Finally, the following cases for $\Delta x = h(\Delta l)$ result (I treat $i = 1$ and $m = n + 1$ as separate, boundary cases):

Case 1.

$$\begin{aligned}
&\text{If} \quad i = 1, \\
&\text{and} \quad \Delta l > \frac{x\sum_{j=1}^{n} k_j \min(l_j,0) - l_1 \sum_{j=1}^{n} k_j[x + \max(l_j,0)]}{\sum_{j=1}^{n} k_j[x + \max(l_j,0)] - xk_1}, \\
&\text{and} \quad \Delta l < \frac{l_1[x\sum_{j=2}^{n} k_j + \sum_{j=1}^{n} k_j \max(l_j,0)] + xk_1 \max(l_1,0) - x\sum_{j=2}^{n} k_j \min(l_j,0)}{xk_1}, \\
&\text{then} \quad \Delta x = \frac{x[k_1\Delta l + \sum_{j=1}^{n} k_j \min(l_j,0)]}{\sum_{j=1}^{n} k_j[x + \max(l_j,0)]}.
\end{aligned}$$

Case(s) 2.

Let $m \in \{2, ..., n\}$.

If $i \geq m$,

$$\text{and } \Delta l > \frac{x[\sum_{j=m}^{n} k_j \min(l_j,0) - \sum_{j=1}^{m-1} k_j \max(l_j,0)] - l_i[x\sum_{j=m}^{n} k_j + \sum_{j=1}^{n} k_j \max(l_j,0)]}{x\sum_{j=m}^{n} k_j + \sum_{j=1}^{n} k_j \max(l_j,0) - xk_i},$$

$$\text{and } \Delta l \geq \frac{l_{m-1}[x\sum_{j=m}^{n} k_j + \sum_{j=1}^{n} k_j \max(l_j,0)] + x\sum_{j=1}^{m-1} k_j \max(l_j,0) - x\sum_{j=m}^{n} k_j \min(l_j,0)}{xk_i},$$

$$\text{and } \Delta l < \frac{l_{m}[x\sum_{j=m}^{n} k_j + \sum_{j=1}^{n} k_j \max(l_j,0)] + x\sum_{j=1}^{m-1} k_j \max(l_j,0) - x\sum_{j=m}^{n} k_j \min(l_j,0)}{xk_i},$$

$$\text{then} \quad \Delta x = \frac{x[k_i \Delta l - \sum_{j=1}^{m-1} k_j \max(l_j,0) + \sum_{j=m}^{n} k_j \min(l_j,0)]}{x\sum_{j=m}^{n} k_j + \sum_{j=1}^{n} k_j \max(l_j,0)}.$$

Case(s) 3.

Let $m \in \{2, ..., n\}$.

If $i < m$,

$$\text{and } \Delta l > \frac{x[\sum_{j=m}^{n} k_j \min(l_j,0) - \sum_{j=1}^{m-1} k_j \max(l_j,0)] - l_i[x\sum_{j=m}^{n} k_j + \sum_{j=1}^{n} k_j \max(l_j,0)]}{x\sum_{j=m}^{n} k_j + \sum_{j=1}^{n} k_j \max(l_j,0)},$$

$$\text{and } \Delta l \geq \frac{l_{m-1}[xk_i + x\sum_{j=m}^{n} k_j + \sum_{j=1}^{n} k_j \max(l_j,0)] + x\sum_{j=1}^{m-1} k_j \max(l_j,0)}{xk_i} - \frac{x\sum_{j=m}^{n} k_j \min(l_j,0) + xk_i l_i}{xk_i},$$

$$\text{and } \Delta l < \frac{l_{m}[xk_i + x\sum_{j=m}^{n} k_j + \sum_{j=1}^{n} k_j \max(l_j,0)] + x\sum_{j=1}^{m-1} k_j \max(l_j,0)}{xk_i}, - \frac{x\sum_{j=m}^{n} k_j \min(l_j,0) + xk_i l_i}{xk_i},$$

$$\text{then} \quad \Delta x = \frac{x[k_i \Delta l + k_i l_i - \sum_{j=1}^{m-1} k_j \max(l_j,0) + \sum_{j=m}^{n} k_j \min(l_j,0)]}{xk_i + x\sum_{j=m}^{n} k_j + \sum_{j=1}^{n} k_j \max(l_j,0)}.$$

Case 4.

$$\text{If} \quad \Delta l > -x - l_i,$$
$$\text{and} \quad \Delta l \geq \frac{l_n[xk_i + xk_n + \sum_{j=1}^{n} k_j \max(l_j, 0)] + x\sum_{j=1}^{n-1} k_j \max(l_j, 0) - xk_n \min(l_n, 0) - xk_i l_i}{xk_i},$$
$$\text{then} \quad \Delta x = \frac{x[k_i \Delta l + k_i l_i - \sum_{j=1}^{n} k_j \max(l_j, 0)]}{xk_i + \sum_{j=1}^{n} k_j \max(l_j, 0)}.$$

This completes the function definition for $\Delta x = h(\Delta l)$.

References

[1]Dave Benson. *Music: A Mathematical Offering, 2007.* Cambridge University Press, UK and USA.

[2]P.A. Bigsby. Tailpiece vibrato for string instrument, March 31 1953. US Patent D169,120.

[3]Richard Brice. *Music engineering.* Newnes, 2001.

[4]Richard Mark French. *Engineering the guitar: theory and practice.* Springer Science & Business Media, 2008.

[5]David Robert Grimes. String theory-the physics of string-bending and other electric guitar techniques. *PloS one*, 9(7):e102088, 2014.

[6]R Hooke. De potentia restitutiva, or of spring explaining the power of springing bodies, vol. 1678. *London, UK: John Martyn*, page 23.

[7] Jarmo Lähdevaara. *The science of electric guitars and guitar electronics*. BoD-Books on Demand, 2012.

[8] Marin Mersenne. *Harmonie universelle, contenant la théorie et la pratique de la musique (Éd. 1636)*. Hachette LIVRE-BnF, 2013.

[9] André Millard. *The electric guitar: a history of an American icon*. JHU Press, 2004.

[10] Harry Ferdinand Olson. *Music, physics and engineering*, volume 1769. Courier Corporation, 1967.

[11] Randy Raine-Reusch. *Play the World: The 101 World Instrument Primer*. Mel Bay Publications, 2010.

[12] Takis Kasparis and John E Lane. *A frequency error model for fretted string instruments*. S. Hirzel Verlag, 2012.

[13] Frank WK Firk. *Essential physics: relativity, particle dynamics, gravitation, and wave motion*. Yale University, 2000.