Pivotuner

Pure intonation on keyboards in real time! Microtonal modulation! Strange and novel sonorities! Welcome to Pivotuner™! Here's a guide to get you up and running. Please let me know if you have any questions.

Introduction

As of writing, most music is composed in 12 tone equal temperament: twelve equally spaced notes per octave. However, music tuned this way is arguably not in-tune! What the ear hears as in-tune is actually frequencies with simple ratios between them: for example, a purely-tuned major third has a ratio of 5:4 from the top frequency to the bottom frequency. The top note is about 14 cents lower than the top note in a equally-tempered major third!

Pivotuner's magic is in converting notes from 12 tone equal temperament to pure intonation! It considers the notes currently played, and decides best how to tune them according to a selectable algorithm. It then outputs the note data, but tuned purely!

There is a side effect to be aware of, first discovered by Giambattista Benedetti in the 1500s. Using pure intonation can result in an unstable pitch center! For example, with certain chord progressions, the pitch of all the notes will slowly get higher and higher. This is called a microtonal modulation! (Also known as a cross-terrain modulation.)

Microtonal modulations are incredibly powerful musical devices! Listen to Jacob Collier's arrangements of *In the Bleak Midwinter* and *Moon River*, which pioneered their use! As demonstrated by those arrangements, microtonal modulations have a unique and powerful energy, which cannot be experienced in any other way.

Microtonal modulations can be difficult to perform precisely; they usually only happen due to painstaking deliberate effort, or accidentally in situations where there is no stable reference pitch (e.g. choir performances). But with Pivotuner, microtonal modulations become a relatively intuitive, controllable, and natural part of playing!

Pivotuner opens the door to many new musical possibilities! Previously impractical, now anyone can easily use purely tuned chords and achieve microtonal modulations (even in live performance)!

Interface
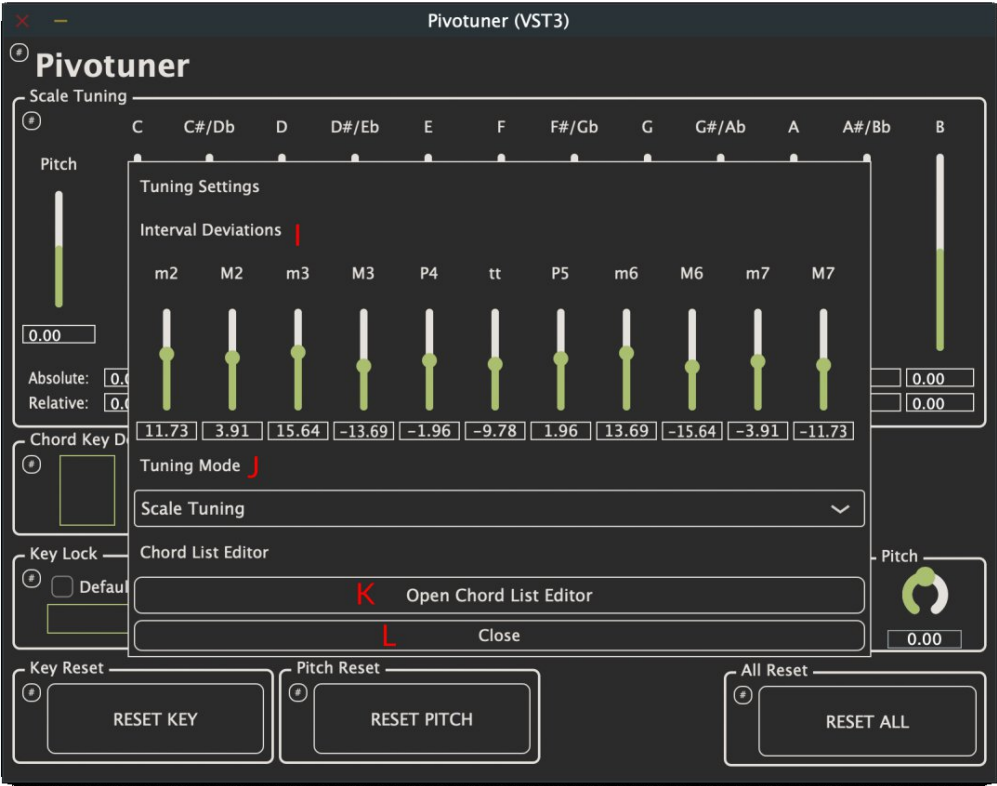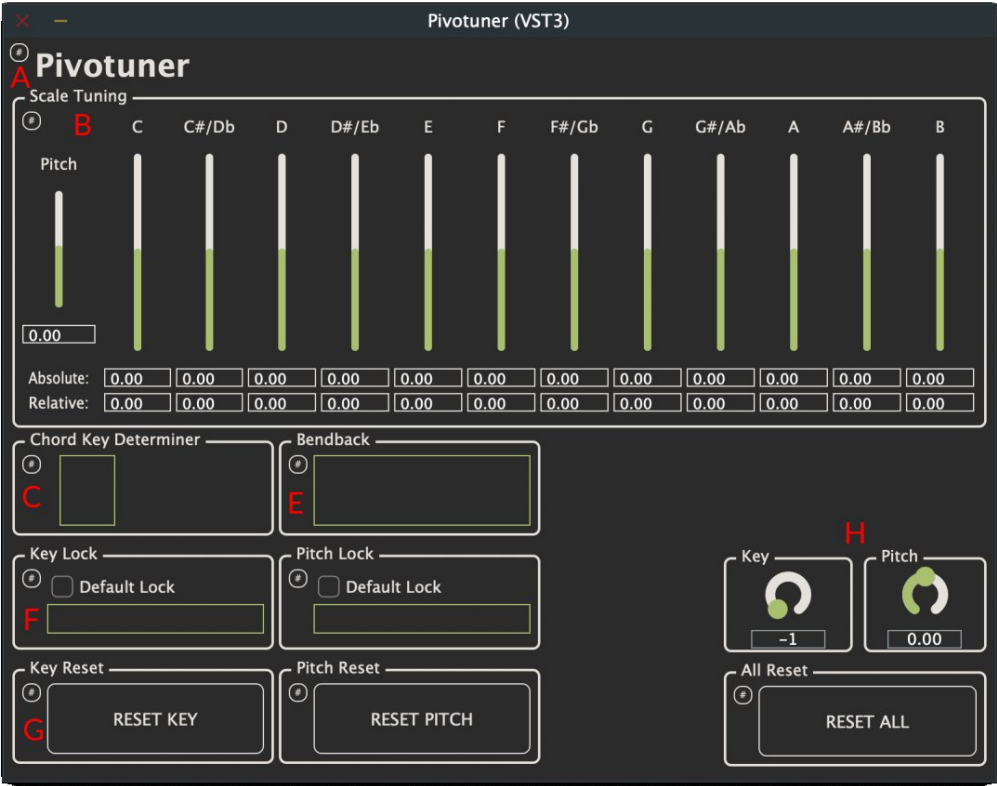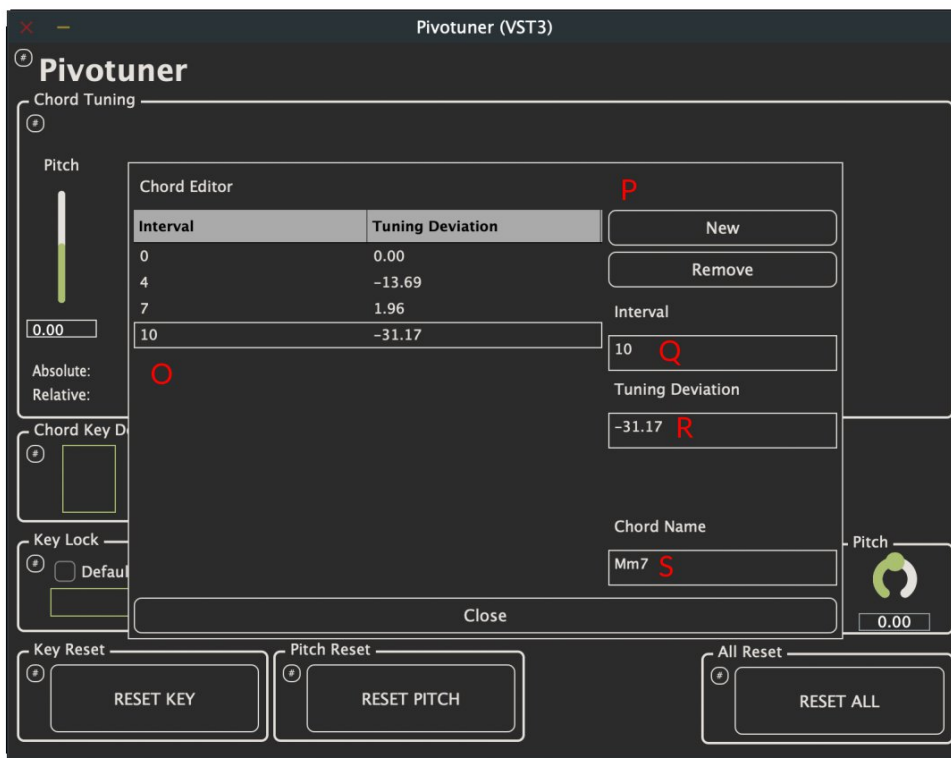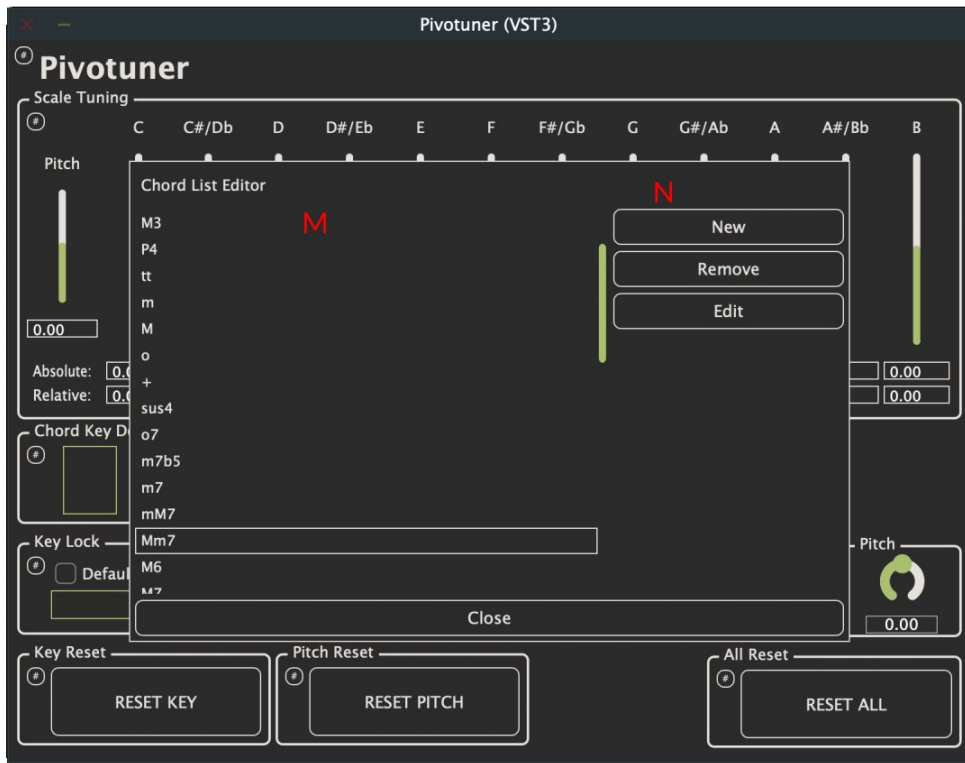
A) Pivotuner settings button (note that most components also have settings buttons).

B) Tuning component (contents change depending on tuning mode).

C) Key Determiner component (contents change depending on key determiner mode).

D) Does not seem to exist in the above images. Spooky.

E) Bendback component. shows current state of bendback.

F) Key Lock component. Allows toggling the default lock, and shows the state of the current lock. Note that the Pitch Lock component, directly right, is almost identical.

G) Key Reset component. Allows resetting the key by clicking the button. Note that the Pitch Reset and All Reset components, to the right, are almost identical.

H) Key and Pitch Automator components. Allow the key and pitch to be explicitly set and automated.

I) Interval Deviations setting. How each interval is tuned relative to equal temperament, in cents, can be customized here.

J) Tuning mode setting. Can select the tuning mode: Scale Tuning, Interval Tuning, Mirror Scale Tuning, or Chord Tuning.

K) Open Chord List Editor button. Opens the Chord List Editor.

L) Close button. Closes the popup window. All popup windows have these. (The tweaked settings are saved automatically).

M) Chord List. Each chord here is selectable.

N) Chord List buttons. Adds an empty chord to the list, removes the currently selected chord, or edits the currently selected chord in the Chord Editor.

O) Chord Editor List. Shows each interval and its tuning of the currently edited chord.

P) Chord Editor List buttons. Adds or removes a new empty interval to the chord.

Q) Interval box. Edit the currently selected interval (0 is unison, 7 is P5, etc.)

R) Tuning Deviation box. Edit the currently selected interval's tuning, in terms of deviation in cents from equal temperament.

S) Chord Name box. Edit the name of the chord.

Not every view is shown here, but a lot of the windows are similar to each other so as long as you have an understanding of these layouts. What each of these things does will be explained below, in the "Using Pivotuner" section.

Installation and Setup

Installing on macOS

Unzip and open the .dmg file, then inside of it, drag each file (the AudioUnit and VST file) to the corresponding folder (which should be the closest folder to each file). If this

does not work, then you may manually need to create a folder at /Library/Audio/Plug-Ins/VST3 first. You also may need to open the folders first, then drag into the new window. If any issues persist, please contact support.

Installing on Windows

Open the zip, and within the opened folder, drag Pivotuner to the VST3 folder, which is a link to the correct location. If this does not work, then you may need to manually create a folder at C:\Program Files (x86)\Common Files\VST3 first. If any issues persist, please contact support.

Installing on Linux

Currently there is no Linux support for Pivotuner, but I can add it if there's interest.

Uninstalling

Manually locate the plugin folder and delete the plugin from there. Here are the locations:

macOS AU: /Library/Audio/Plug-Ins/Components
macOS VST3: /Library/Audio/Plug-Ins/VST3
Windows VST3: C:\Program Files (x86)\Common Files\VST3

License Activation

When Pivotuner is first launched, it will ask for a License Key. This License key should have been emailed to you by Gumroad upon your Pivotuner purchase. Please make sure you are connected to the internet, and enter the License Key verbatim (copy and paste is most convenient). If issues occur, or if you would like to activate Pivotuner on a computer without internet access, please contact support.

Compatible Software

Pivotuner should work in any DAW with multichannel MIDI support, and work with any virtual instrument that supports with MPE, though not everything has been tested. Here's a non-comprehensive list of software which has been verified to work with

Pivotuner. Again, if something is not on the list below, it still most likely works! If you find a piece of software not on the list which works with Pivotuner, please feel free to contact support so that the list may be updated!

DAWs: Ableton, Logic Pro, Reaper.

Instruments: Ace, Alchemy, Alpha, Blade2, Continua, Diva, EFM1, ES2, F-em, FRMS, Karp, Legend, Msoundfactory, Obsession, Omega, Phonec, Pigments, Quanta, Quick Sampler, Retro Synth, Sampler (Logic), Sculpture, Sektor, Serum, SQ80 V, Surge XT, Swam, Tal Bassline, Tal J 8, Tal Mod, Tal Sample, Tal u no lx, Vecto, Vintage Clav, Vital, Lion.

Setting Up in Logic

1) Create a software instrument track.
2) In MIDI FX, select Audio Units > dmitrivolkov > Pivotuner.
3) Select an MPE-compatible instrument, such as Sampler, Alchemy, EFM1, ES2, EXS24, Retro Synth, Sculpture, or Vintage Clav. Alternatively, add an external instrument plugin (see "Setting Up Instrument Plugins" below).
4) In the dropdown at the bottom of the selected instrument's GUI window, make sure to enable MIDI Mono Mode and that the pitch bend range matches that set in Pivotuner.
   a) Common base channel 1 corresponds to low MPE base channel in Pivotuner.
   b) the pitch bend range should be 48, both for the Logic instrument and Pivotuner.

Setting Up in Reaper

1) Create a new track.
2) Open up the FX rack and add Pivotuner.
3) Add an MPE-compatible virtual instrument to the FX rack after Pivotuner. See "Setting Up Instrument Plugins" below.
4) Make sure the MPE settings match between the virtual instrument and Pivotuner!

Setting Up in Ableton

Ableton does not allow AU plugins to output MIDI, so the VST3 version of Pivotuner must be used.

1) Create a new MIDI track and add Pivotuner.
2) Create another MIDI track
   a) Set the input to be from Pivotuner's track, Post FX
   b) Make sure to set monitoring to "In"
3) Add an MPE-compatible virtual instrument to the 2nd after Pivotuner. See "Setting Up Instrument Plugins" below.
4) Make sure the MPE settings match between the virtual instrument and Pivotuner!
5) Put MIDI data on the 1st MIDI track

Setting Up in Bitwig

Bitwig does not expect plugins to return MPE data, so setting up Pivotuner requires using an external program to use Pivotuner as a virtual MIDI device, essentially. It is very inconvenient, so I unfortunately, I recommend trying to avoid using Pivotuner with Bitwig for now.

Setting Up in Other DAWs

1) Create a MIDI track or equivalent.
2) Insert Pivotuner as a MIDI effect, or at the front of the effects chain.
3) Insert an MPE-compatible virtual instrument and make sure the MPE settings match with Pivotuner.

This is the general idea, but the exact process may differ from DAW to DAW. Some DAWs do not allow placing VSTs in front of instruments in the chain, and may require two tracks; placing Pivotuner on one and MIDI clips on one, then routing the MIDI output from that track to the instrument track. If confused, please contact me and I will try to help!

Setting Up Instrument Plugins

For a plugin to work with Pivotuner, it must be MPE-compatible. Some plugins may call this "midi mono mode." Make sure to set the MPE settings in your instrument the same as the MPE settings in Pivotuner!

If you're looking for a free Pivotuner-compatible plugin, I recommend the Vital synth plugin: https://vital.audio/ . Make sure to enable MPE under "Advanced," and to make sure that the pitch bend range matches what is set in Pivotuner!

Setting Up Pivotuner

Set up Pivotuner in a DAW or other audio plugin host as specified above. Make sure Pivotuner's MPE settings (accessible from the Pivotuner settings button) match the MPE settings of the DAW or audio plugin host you are using! If "Low MPE Base Channel" corresponds to an MPE main channel of 1, and unchecked corresponds to an MPE main channel of 16.

Next, set the default pitch shift by setting the Pitch Automator. For tuning around A=440hz, this should be 0.0c. For tuning around A=432hz, this should be -31.77c. For tuning around A=442hz, this should be 7.85c. For tuning around A=444hz, this should be 15.67c.

Next, set the MIDI CCs which control key center and pitch center locking (accessible from the Key and Pitch Lock settings buttons). I recommend controlling key center locking is controlled by MIDI CC 64 (sustain pedal) and pitch center locking is controlled by MIDI CC 1 (mod wheel). The default CC of -1 indicates that they are not set up to be controlled by anything. The details of how key and pitch center locking is explained below.

Like the locking, it is possible to set the MIDI CCs which control resetting. The same principals apply.

To adjust the tuning of each interval, open the Tuning Settings and adjust the corresponding sliders. To adjust chord tunings, open the tuning settings, and open the chord list editor. From here, new chords can be added, removed, and edited. For each chord, intervals can be added, removed, or edited as well.

Using Pivotuner

Core Concepts of Pivotuner

Pure intonation: a way of tuning notes such that they sound in-tune to the human ear. While "pure intonation" is a nebulous concept with no single concrete definition, this

plugin attempts to achieve one form of it by tuning all notes in pure intervals to a key center.

Key center or key: in the context of Pivotuner, this is the note to which all the other notes are being justly tuned.

Pitch center or pitch: in the context of Pivotuner, this is the exact cent value to which all notes are being justly tuned.

Pitch deviations: the difference (usually expressed in cents) between how a note is being tuned and how a note is usually tuned in equal temperament. Pitch deviation can be relative to the current pitch center, or to a default pitch center such as A=440hz.

Microtonal modulation: When the pitch center changes due to changes in key center. For example, changing from a key center of C to a key center of Eb results in the pitch center increasing by 15.54 cents (because Eb justly tuned to C has a pitch deviation of 15.54 cents).

Key center locking: This prevents the key center from changing, and maintains the pitch deviations tuned to the key center. This effectively disables microtonal modulates, but also enables interestingly detuned chords (e.g. locking the key to C then playing an A minor).

Pitch center locking: This prevents the pitch center from changing. This effectively disables microtonal modulation, but the key center can still change.

Interval numbers: a number corresponding to the number of semitones in an interval. 0 corresponds to a unison, 1 corresponds to a half step, 7 to a perfect fifth, 12 to an octave, and so on.

Note numbers: the number (from 0 to 127) corresponding to a note. 0 corresponds to C-2, 60 corresponds to C4, 61 corresponds to C#4, 127 corresponds to G9, and so on.

Using Pivotuner

Pivotuner generally works as follows: first, it determines the current key (using a "key determiner", then from there determines how to tune the rest of the notes to that key

(using a "tuning mode"). Often, when the key changes microtonal modulation will occur, but this can be prevented with pitch locking. Similarly, the key can be locked. Furthermore, both the key and pitch can be reset to a predefined value. The key and pitch can also be manually automated. Finally, "bendback" can be used when resetting the pitch to glide the notes back, instead of the instant snap which normally occurs.

Key Determiners

Key determiners figure out which note the rest of the notes in the chord should be tuned to. Pivotuner has 3 key determiners.

- Length Key Determiner: tunes to whichever key has been held down the longest. Arpeggiating a C major triad upward (while keeping the notes held) would tune to C, but arpeggiating downward would tune to G, because those notes would be played first, and therefore held down the longest.
- Bass Key Determiner: tunes to the lowest note.
- Chord Key Determiner: attempts to detect the chord from the chord library, and sets the key as the root. Best used in conjunction with Chord Tuning (see below).

I generally would recommend using Length Key Determiner.

Tuning Modes

Tuning Modes determine how the chord is tuned.

- Scale Tuning: tunes each note to create a pure interval with the key.
- Interval Tuning: tunes each note to create a pure interval with the next-closest note.
- Mirror Scale Tuning: tunes each note to create a pure interval with the key, with notes below the key inverted. This allows the m7 interval to be tuned differently from the -M2 interval, which can make a big difference when tuning Mm7 chords.
- Chord Tuning: tunes each note according to a chord in the chord library, defaulting to equal temperament if no chord is found. Accounts for inversions. Best used in conjunction with the Chord Key Determiner (see above).

I generally would recommend using Mirror Scale Tuning or Chord Tuning, though Scale Tuning is the easiest to understand when starting out.

Locks

Locks prevent key or pitch changes from occurring. The default lock can be set, then toggled with a custom MIDI CC message (customizable in the lock settings). Convenient CCs to use are 1 (mod wheel) and 64 (sustain pedal). A potential use case would be to unlock the pitch temporarily during a microtonal modulation, then re-locking it upon reaching a new section which should have a constant pitch.

Resets

Pitch reset sets the pitch to a default value, specified by the pitch automation value (see below). Key reset sets the key to -1, corresponding to no key/equal temperament. Resets can be triggered by clicking the button, or with a customizable MIDI CC value.

Bendback

Bendback is used along side pitch reset for a more gradual shift. By setting the MIDI CC which controls bendback to something with gradual shifts, such as the mod wheel, and also to the same CC as pitch reset, then by activating and slowly deactivating the MIDI CC, a gradual pitch bend of all notes to the specified pitch can be achieved.

Key and Pitch Automation

Key and Pitch are DAW-automatable parameters. Whenever the key or pitch parameter changes, they set the key and pitch used for tuning accordingly. That said, because the key and pitch are also determined progammatically by Key Determiners and Tuning Modes, the key and pitch can become out-of-sync with the key and pitch automatable parameters. If this is not desired, the key and pitch should be locked.

The pitch automatable parameter is also used to determine the target pitch for resets and bendback.

Chords

Chords in Pivotuner are sets of intervals and their corresponding tunings, in cents relative to equal temperament. Chords are used for key detection by the Chord Key Determiner, as well as for tuning by Chord Tuning.

Chords are specified as intervals relative to a root (which is represented by an interval of 0). Intervals here are represented by the interval number (e.g. number of semitones). Note that negative intervals (such as -1, or a minor 2nd below the root) and intervals greater than an octave (such as 13, or a minor 9th) are permitted!

To edit the chords, open the Chord List Editor from the Tuning Settings. This will open the Chord List Editor, to which you can add new chords (by default empty), edit chords, and remove chords, using the corresponding buttons. To edit a chord, first select it, then click the "Edit Chord" button, which will open the Chord Editor.

The Chord Editor displays a list of intervals and their tunings. To edit an interval and/or it's tuning, select it in the list and change the numbers on the side. Intervals can also be added and removed (except for the root). The name of the chord can also be changed; this is the name that shows up in the Chord Key Determiner.

Tips

Modulation Tips

The following tips assume the default interval tunings are being used, and that the key and pitch are not locked.

Arpeggiating an diminished chord upward modulates upward, and arpeggiating downward modulates downward.

Arpeggiating an augmented chord upward modulates downward, and arpeggiating downward modulates upward.

Diatonic or pentatonic scales played upward modulate upward, and played downward modulate downward.

Support

If you would like support, or have any thoughts or suggestions, please contact dmitri.s.volkov@gmail.com.

Credits

Pivotuner, created by Dmitri Volkov

Huge thanks to Ben Bloomberg for his mentorship and help!

Pivotuner is heavily inspired by Jacob Collier's arrangements of In the Bleak Midwinter and Moon River, which pioneered using microtonal modulations as musical devices!

Thank you to the following people for helping test Pivotuner!

Auri Luve, Barak Schmool, Ben Bloomberg, Erin Blake, Ethan French, Eugene Martynec, Florian Mrugalla, Israel Strom, Jacob Collier, JazzyJasperZ, Joel Werleman, Jordan Olson, NJ Helder, Nick Revel, Sonosaurus, Yoni Silver!