

Hayward Tuning Vine – a means of exploring Just Intonation on Mac and PC

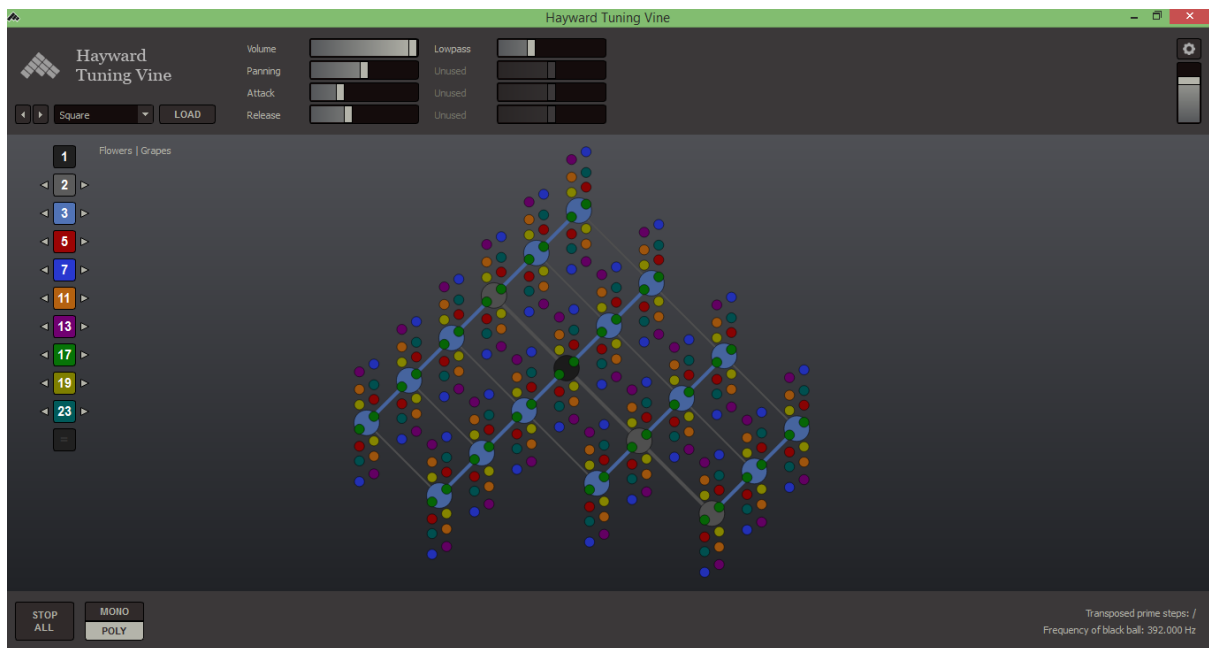
Review from 2015 published in Soundbytes magazine

Available from <http://www.tuningvine.com/> 45 Euros; 49 US Dollars

In the vast universe of microtonal tunings, one of the more interesting corners is that occupied by Just Intonation, that is, tunings made of whole-number ratios. Some of these can be very consonant, and in fact, the reason most people gravitate to Just Intonation is the fact that it can make “tonal” tunings sound even more tonal – where the structure of the scale reinforces the structures of the harmonies. The ratio-based tuning of just intonation, however, can also produce some really complex dissonances. And these dissonances, like the consonances made by just intonation, can sound distinctly different from anything the normal piano scale is capable of. As Harry Partch, the father-figure of just intonation in the 20th century said, “Just Intonation dissonance is a whole other serving of tapioca!”

For those unfamiliar with tuning theory, a brief explanation might be in order. All musical pitches are the result of frequencies vibrating at specific rates. For example, an A above middle C vibrates at 440 cycles per second. When we hear an interval, any interval, we’re hearing a sound that results from two pitches in a particular mathematical relationship. So if we hear the interval of a perfect 5th, for example, no matter what perfect 5th it is, the sound of that interval is produced by a particular mathematical ratio between the frequencies of the two pitches, in this case the ratio of 3:2. That is, the top note is vibrating at 1 and $\frac{1}{2}$ times the rate of the bottom pitch. A major 3rd, any major 3rd, is the result of 2 frequencies in a 5:4 ratio. That is, the top frequency is 1 and $\frac{1}{4}$ times the frequency of the bottom frequency. Using our normal piano scale, the ratios between notes are not quite those pure ratios given here. This is because our normal scale is not the result of pure ratios, but rather, divides the space of an octave (the ratio 2:1) into 12 equal parts. So for example, the perfect 5th on the piano is $\frac{1}{50}$ of a semitone smaller than the just 3:2 ratio. This is a very small difference, but one which is hearable. Even more different from the pure ratio is the major 3rd, which on the piano is $\frac{1}{6}$ of a semitone larger than the pure 5:4 ratio of the just major 3rd. If one wants to work with very pure intervals, then, one will not find these ratios in the piano scale, and one has to look at other resources, such as Just Intonation. For our purposes, we can say that a lot of the interesting Just Intonation ratios are made from elements of the harmonic and subharmonic series, based on certain prime numbered intervals, such as 2/1, 3/1, 5/1, 7/1, 11/1, 13/1, 17/1, 19/1 and 23/1. One can certainly go higher than this, and there are very interesting sounds to be found in prime numbered ratios larger than this, but for a start, these ratios, and their products (when you take a ratio of a ratio, for example the 7th harmonic of the 5th harmonic), will provide us with a very fertile field for sound exploration.

Robin Hayward, with programmer Bjorn Naesby, have developed a software for Mac or PC to explore the ratios of Just Intonation. It’s called the Hayward Tuning Vine, and it features a unique interface for accessing the numerical complexities of the ratios of Just Intonation. This interface at first glance looks quite complicated, is actually simplicity itself.



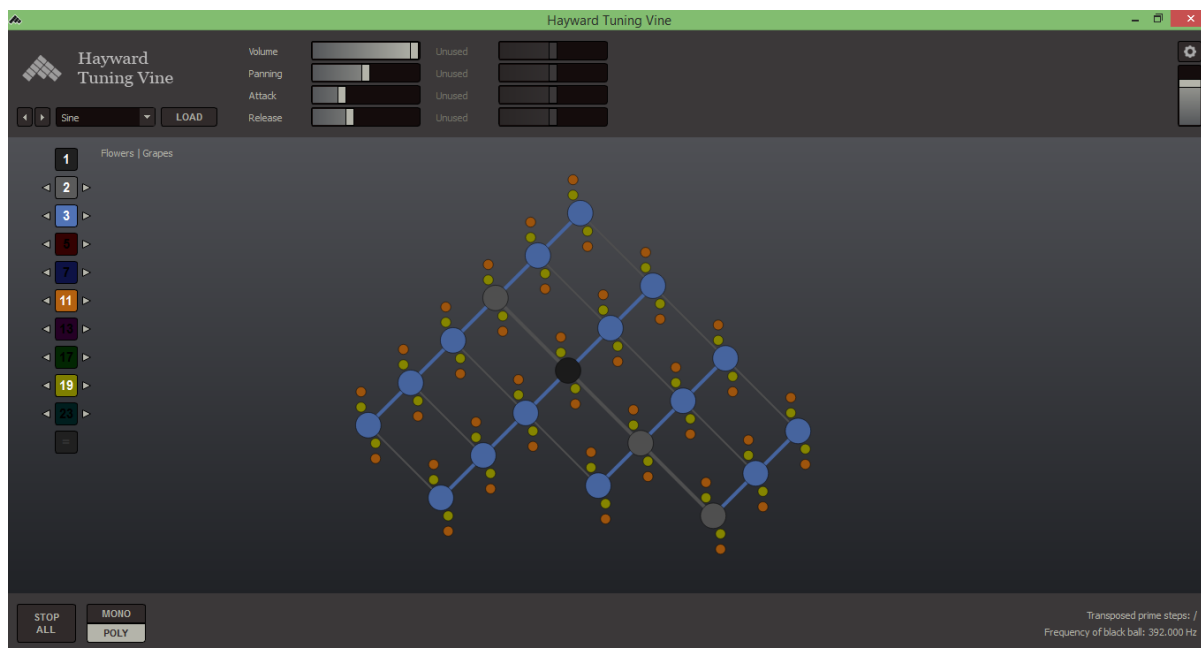
At the top, on the left, we see a waveform selector – this determines the timbre that the instrument plays with. The four basic waveforms, sine, triangle, sawtooth and square are given as options. However, since the behind-the-scenes programming is done with the public domain music programming language PD (which we'll have a full feature on in the next issue of Soundbytes), users can make their own PD patches and use those as sound sources. So although not included in the basic package, the use of, for example, samples, is quite possible with the Tuning Vine. The program actually loads 128 copies of the given PD patch into computer memory, giving you a potential of 128 voice polyphony. This can also stress out underpowered machines. For example, on my tiny ASUS notepad, the program works fine with sine waves, but other waves seem to have breakups in them. The developers tell me that in the next version of the program, this will be fixed – they will have an option to decrease polyphony to 64 or 32 voices. They say that this should accommodate less powerful machines quite nicely.

To the right of that are basic controls for volume, panning, attack and decay of the sounds, as well as four controls for aspects of the sound. In this version of Tuning Vine, that's simply a low-pass filter control for the three non-sine basic waveforms. If you design your own PD patches, you can have access to GUI control of three more parameters of your choice. These controls can be changed while performing the Tuning Vine, and the results will be heard with the next note played.

On the far right is a cog wheel for preferences, and then a final volume control.

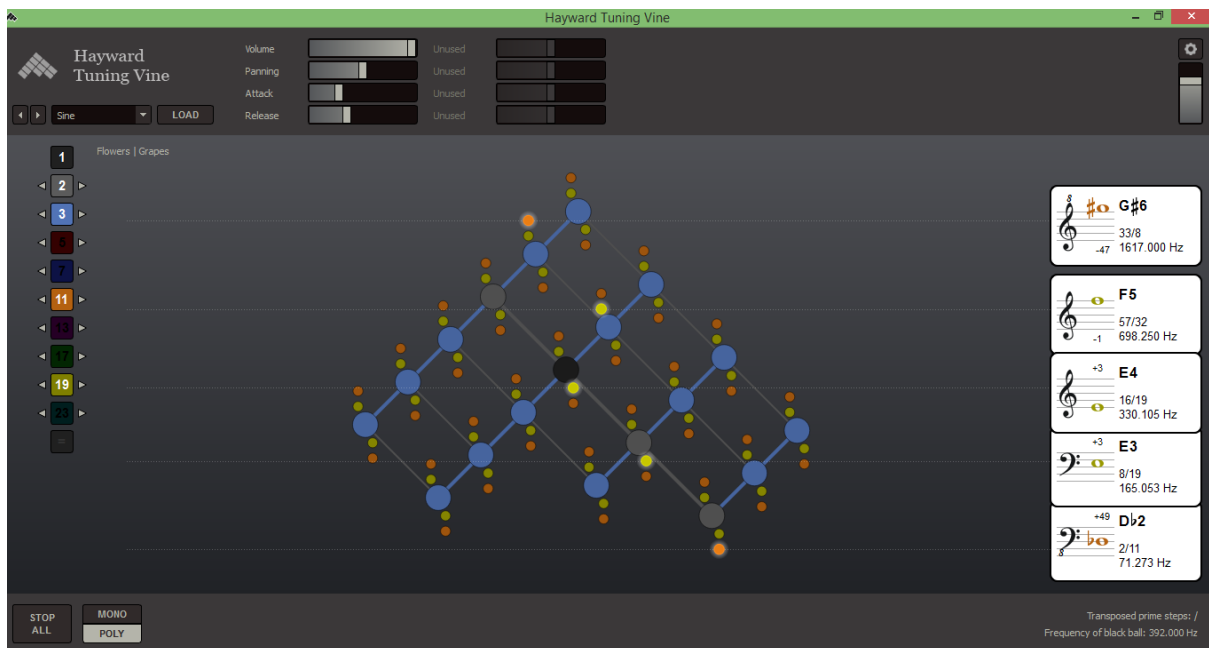
In the main window, on the left, is where things get interesting. There are a series of colored, numbered boxes. These turn on or off the various-numbered harmonics, or harmonic ratios, that one's tuning frequencies will be made of. Clicking to the left or right of these transposes the entire Tuning Vine up or down by the ratio given by the number. More on that later. In the center of the screen is the Tuning Vine itself, a grid with many colored balls ascending and descending from each major node of the grid. In the center of the grid is a black ball, which represents the fundamental pitch for the whole Vine. This is called 1/1, and it's the pitch against which all the other ratios are measured. Moving upward to the left, or downward to the right, the pitches produced by the large balls are related by octaves, that is, by the ratio 2/1. Moving upwards to the right, or downwards to the left, and the ball's frequencies are related by 3:2, or perfect 5ths. So the basic grid is one of octaves and perfect 5ths. That is to say, it's a grid with axes on 2 and 3.

Each of the major balls of this grid has a series of smaller balls ascending and descending from it. These color-coded balls represent the ratios shown in the list of selection buttons on the left of the screen. If we were to turn off all of the ratios above 3, except for, say 11 and 19, the grid would become more sparse, as shown here:



In this picture, you can see that there are only yellow and orange balls above and below each of the major balls in the grid. The yellow balls have a relationship of being 19/16 either above or below the central ball in pitch, while the orange balls have the relationship of being 11/8 either above or below the central ball's pitch. With the central black ball, which has the name of 1/1, the orange ball above it has the ratio of 11/8. But with the ball immediately to the right and above the central ball, which represents the ratio of 3/2, the orange ball above it would represent an 11/8 above that 3/2. That is, $(11/8 \times 3/2)$ the ratio 33/16. By exploring the various ratios - turning them on and off - and selecting them in relation to each of the main balls of the 2-3 grid, one can explore a wide range of tuning possibilities.

But wait, there's more, as they say on the late night TV ads. Every time a pitch is selected, a small card appears at the right of the screen, which identifies which pitch it is that is being played. For example, here's a fairly widely spaced chord with its note cards showing. Notice that each colored ball is arrayed in terms of the height of the pitches, from top to bottom.



Each tuning card shows the name of the pitch, the ratio of it above or below 1/1, its frequency in cycles per second, and the pitch's deviation (in cents, 1/100th of a semitone), either above or below the nearest 12-tone-equal-temperament pitch. Notice also that they played notes are glowing when they are selected.

By clicking on the left or right of the ratio selector buttons on the left of the screen, one can transpose the whole grid by the ratio of the given factor. So clicking once on the left of the 11 button transposes the whole grid up by 11/8. These buttons can be clicked while performing, so for example, one can have pitches from one grid sustaining, while pitches from a new transposition can then be added to the mix of tones. Clicking twice will transpose the grid up by two of the selected intervals. So clicking to the right of the 3 button twice will transpose the grid up by two perfect fifths. That is, it will transpose the grid up by a major 9th, because $3/2 \times 3/2 = 9/4$. Or if you're having trouble understanding that, think of it in terms of traditional note names - C to G is one perfect 5th, and G to d is another. One problem is that once one transposes up more than one or two levels for intervals greater than 3, the lit-up selected notes disappear off the screen. They keep sounding, but they are no longer visible. The developers say that this is a feature that will be fixed in the next version of the software.

There are a number of other pitch resources to explore in the Tuning Vine. For example, there is an alternative mode of notating the ratios higher than 3, called Flowers, where the ratios above and below each central ball are displayed as a circle of colored balls rather than an ascending or descending collection of colored balls. There are other features as well, but in a brief look such as this, the examples given here will give you a hint of the possibilities inherent in the Tuning Vine.

At the bottom left of the screen is a "Stop All" button, which does what it says - it terminates all pitches. The selection button to its right sets the mode to Mono or Polyphonic for the grid. These modes can coexist. For example, one could have a sustained chord of several notes, made with the Poly mode, then one could switch to the Mono mode, in order to play a single line melody over the sustained chord. Transposition information and the frequency of the starting pitch are given in the lower right of the screen.

The program comes with a quite extensive and very readable manual (in pdf format), which is designed to guide beginning users of Just Intonation through the ins and outs of the tuning. Later chapters, such as how to set up harmonic and subharmonic series on the Tuning Vine, are quite elaborate, but well worth working through.

At the moment, the Tuning Vine is best used to make drones or sustained chords, and there is no way to control it except with the computer screen controls. Several improvements are scheduled for version 2, including a chord memory, MIDI in (mappable), the ability for notes to remain visible through many degrees of transposition (mentioned above), and a facility for recording the output of the program. Already on the Mac, one can use Soundflower to route the output of Tuning Vine to a DAW or sound recording program, but on the PC, unless your sound drivers support audio loopback, you're out of luck.

It's a very interesting program, quite versatile, and quite usable – a great tool for exploring Just Intonation sounds. I immediately plunged into exploring the universe of beating, closely tuned sounds possible with it. For example, if you were to activate two tones, one at $7/12$ at 228.627 hz, and another at $19/32$ at 232.750 hz, these will merge into a beating tone which beats around 4 times a second. Another two tones, one at $19/12$ at 620.667 hz and another at $8/5$ at 627.200 hz merge into a beating tone which beats at around 6.5 times a second. Playing these two sets of beating tones together results in a nicely complex timbre, one that can be used in a sampler tuned to any other scale imaginable. For those interesting in exploring Just Intonation, the resources offered by the Hayward Tuning Vine are considerable, with implications for use of the sounds outside the program that are quite intriguing.

What improvements would I like to see in the program? First of all would be either the ability to record its sound output, or even better, putting the program in a VST and AU wrapper, so that it can be used in an integrated patching environment with other programs. For example, taking the output of Tuning Vine and processing it through Illformed's Glitch2 could provide quite interesting soundscapes.

Second, the ability to save the tunings is essential. This could take the form of a "chord memory," which the developers are talking about, but I would also make the case for having the program save chords as Scala files, as well, so that the tunings developed in the Tuning Vine could be immediately loaded into other software synthesizers. At the moment, if you find a tuning you like in the Tuning Vine, and you want to use it in another program, you have to write out the tuning by hand, and transfer that into a program like Scala, which will make a broadly applicable tuning file to use in many other programs.

Thirdly, MIDI input would be quite useful. This could take two forms – first, with a mapping utility, so that a keyboard could play melodies of selected notes in the chosen scale. Second, as the developers are suggesting, mapping the MIDI input to the proposed "chord memory," so that the incoming MIDI keys select from a set of already saved chords. And having the ability to switch between these in real time, so that one could select a chord, then switch to melodic playing with another chord, would be quite useful. With these improvements, the program would graduate from being a very useful program, into something that would have spectacular possibilities for real-time performing and improvising.

In short, if you're interesting in exploring the intricacies of Just Intonation, the Hayward Tuning Vine is highly recommended. At the moment, it's a very useful program. If the developers implement some of the changes their talking about, it will become truly spectacular.