ELECTRONOTES 218

Newsletter of the Musical Engineering Group

1016 Hanshaw Road, Ithaca, New York 14850

Volume 23, Number 218 October 2013

DISPERSION OF "TONE BURSTS" AS A MEANS OF FORMING MUSICAL SOUNDS

INTRODUCTION:

Just about everyone who first thinks about musical tone synthesis begins by supposing that there would be some merit to first considering the formation of tones from what we call "tone bursts" which are finite length segments of sinusoidal waveforms. For example, if we want a musical note at A = 440 Hz we might take 440 cycles of a sinewave and play them out during one entire second. That's of course 440 cycles/second, or 440 Hz. None of us actually expect this to do the whole job, but it is a start. What could go wrong? A lot.

Among the problems with tone bursts are the fact that even though they are derived (in theory) from continuous, infinite-duration sinusoidal waveforms of a single frequency, when truncated to finite length (as we always must do), they have additional frequency content – not just a single frequency. The notion of Fourier Analysis is an important mathematical tool for looking into this. But it is a tool, and does not tell us the whole story.

Complicating the story is the fact that (1) we always hear what "really" happens with our ears, and they are not compelled to be Fourier analyzers, and (2) we have a long and rich tradition (perhaps "cultural prejudice") that influences what we will accept as being a candidate for, or actual musical tone.

For example, we are likely to write “fail” to the 1 second, 440 Hz tone burst as a musical tone. Something funny (contrary to our listening experience) happens at the very start, and at the very end, and the middle is somehow boring. Here we worry about transient effects and steady state, as well as the idea that the ear needs to be “challenged” to hold our interest. As our readers likely already know, at the very least we would want to add an amplitude-controlling “envelope” to turn the tone on and off gradually, and we would probably want a starting signal that is rich in harmonics, and then we might well filter this dynamically as the tone progresses in time so as to produce a time-varying “spectrum”. In doing this, we are really imitating the traditional sounds we have come to expect. Sure some new sounds would seem to be invited in (intellectually thinking), but sadly, the synthesizer designer needs to sell equipment to musicians who in turn need to sell music to the public.

Accordingly the notion of starting with a simple tone burst is modified by starting with a harmonically richer wavefrom and adding amplitudes dynamics and spectral dynamics (the latter by means of filtering, modulations, distortions, etc.). Here we want to use the idea that a tone burst (of a sinewave or other waveshape) has spectral components that can be separated or “dispersed” in time.

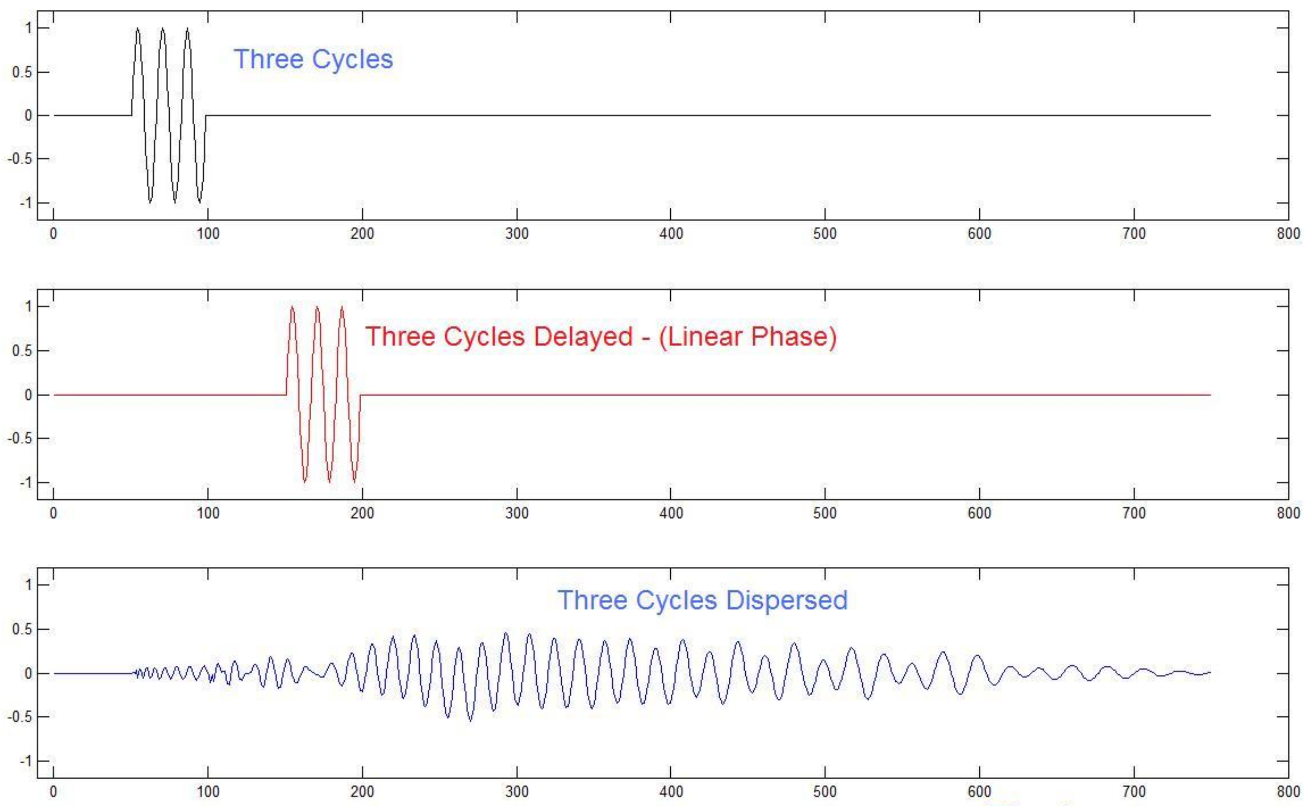


Fig. 1

Fig. 1 shows the general idea of what we have in mind. In the top panel we have simply three cycles of a sinewave tone burst, and the middle panel is not much more interesting – just showing what happens if we delay this burst. This simple time delay is deceptively complicated if we consider the problem in the frequency domain. In this case, in order to achieve a pure time delay, each frequency component (of which there are an infinite number) needs to receive an appropriate phase shift so that they all line up. This is actually a well-known "linear phase" requirement. But if we want a delay, we just think in terms of a time-domain realization, not in terms of linear phase.

Now, in all other cases, the alignment is lost! Different frequencies are delayed by different amounts, and are "dispersed", spread and shifted in time according to the system that is involved. Dispersion is a well-studied phenomenon, particularly in communications where a tone burst (famously a dot or dash of Morse code) is launched into a channel of the atmosphere to find its way to a receiver. In this case, if different frequencies are subject to different phases (and amplitude changes in the general case), the components of the tones are spread wider and shifted. Potentially this can result in garbling at the receiver.

In our synthesizer work, dispersion is virtually non-existent. (We don't engage in phase shifts of more than a few cycles at the very most). However, here we want to see if we can use dispersion to modify a simple tone burst so that it takes on the characteristics of signals we have come to find useful as musical tones. This is suggested in the bottom panel of Fig. 1. Note here that we have achieved, based on the three-cycle input, a much wider signal, one that has amplitude shaping, and one which seems to change in frequency (higher to lower) as the tone progresses. Thus we achieve some of the characteristics of a useful musical tone that we suggested we needed. Clearly this is interesting. Is it useful?

Before going on, we should say what we have done here. We repeatedly filtered the three-cycle input with an all-pass filter. This is equivalent to 30 all-pass filters in series. Here the all-pass filters were 2nd-order (two poles, two reciprocal zeros) and identical. Nothing prevents us from trying innumerable variations on this theme. The reader has likely noted here that the reference to the "reciprocal zeros" gives away that this is digital signal processing (as if 30 filters in series were not already a clue that this is really a digital computation).

Indeed the exercise here is going to be one that is largely a trial-and-error, note-by-note investigation. While the degree of computation is quite large, the ideas are very simple. We just make a simple filter (or incrementing parameters on successive filters) and loop the output of each one into the next – and look at and listen to what comes out. The view here is not so much a processing such as we get from a filter during a tone, but rather more of a "pre-processor" that gives us a better piece of raw material. Instead of starting with a tone burst, you will be starting with something more inherently varies and hopefully more interesting at the very beginning. Perhaps you can think of it as a sort of continuously-varying waveshaper.

THE ALL-PASS FILTER

Note:

Here we propose to simulate dispersion using all-pass filters. We choose all-pass because we want to leave the actual spectral content alone, even though the time alignment of the various components is to be intentionally upset. So we need to start with a complete understanding of the all-pass. We will be using second-order all-pass sections. But first, we should form the most useful perspective:

There are many variations on this theme, even within the confines of parameters chosen for our proposed dispersion study. Clearly, there are:

(a) The parameters (poles and reciprocal zeros of a particular all-pass section).

(b) The number of sections in series, and

(c) ……whether or not they all have the same parameters.

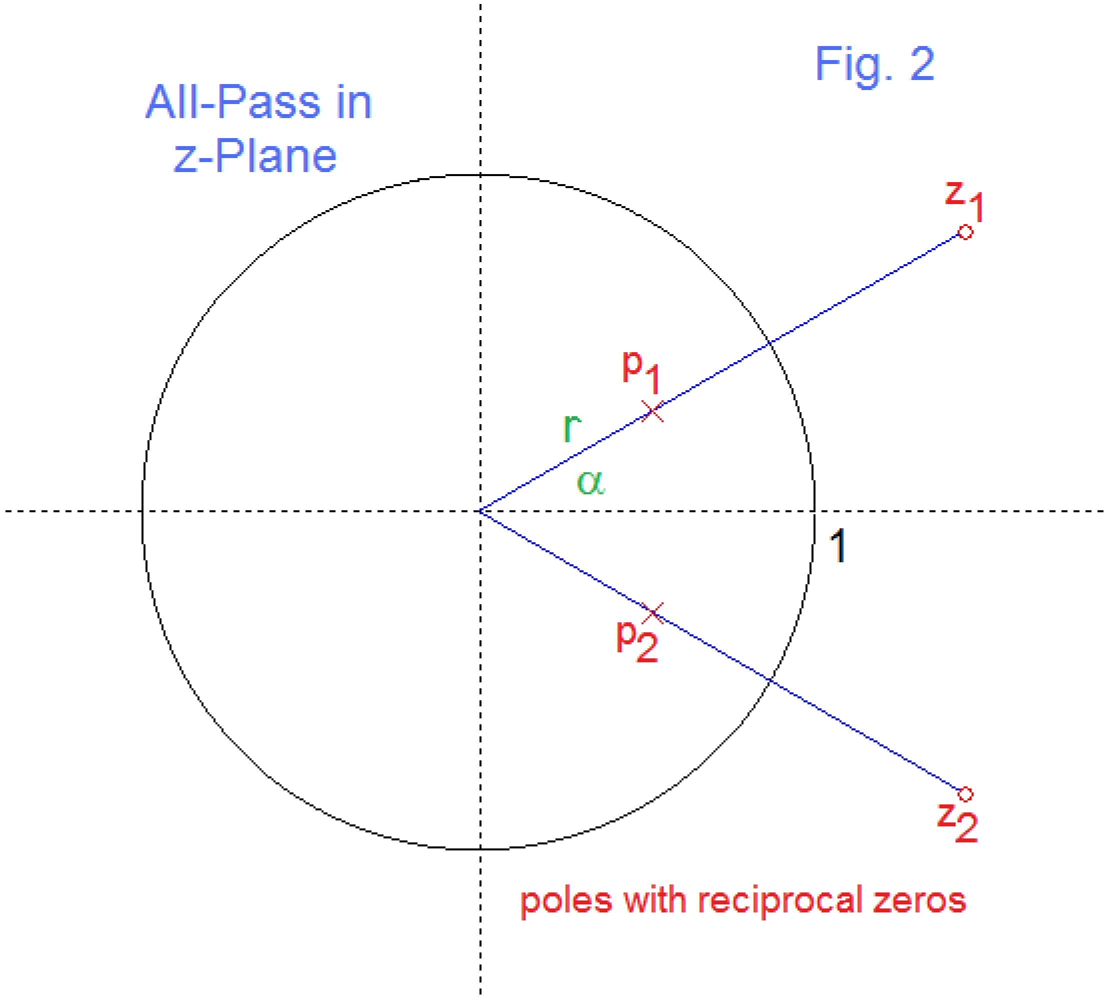
(d) The input signal, whether it is an impulse, a tone burst, or whatever.

(e) Any decision to depart from strict all-pass.

That's a lot of experimental room! If we were to examine the full space of the experiment, this study would revert to nothing more than a study of the use of all possible filters as "generalized resonators".

Here we will need to restrict ourselves to the development of "tools" and to cases where we can learn, and illustrate things. Thus we will generally be using a large number of all-pass sections in series (perhaps hundreds of sections) and our choice of input signal will be a tone-burst so that the input spectrum is well defined (but because it's lime limited, not just a single frequency). In this way we can see how the various frequencies are separated.

A digital all-pass has one or more poles inside the unit circle and each of these has a corresponding zero that is in a reciprocal position that is thus outside the unit circle. If the poles is at p = a +bj, the zero is at $1/(a+bj) = (a-bj)/(a^2 + b^2)$. Here $(a^2 + b^2)=r^2$ is the square of the pole radius r. Because we are interested in real signals, we almost always choose poles in complex conjugate pairs. Accordingly it is convenient to suggest that in total, that if we have a pole at an angle α and at radius r, we have a zero at angle α and radius 1/r, as well as conjugates of both (Fig. 2). One pole thus defines an entire (two pole, two zero) all pass.



Our dispersion-causing filter will be composed of multiple (tens or even hundreds) second-order sections, which may all have the same pole/zero array, or may be different (Fig. 3). Here we indicate a simple three-cycle tone burst as passed through N all-pass stages.

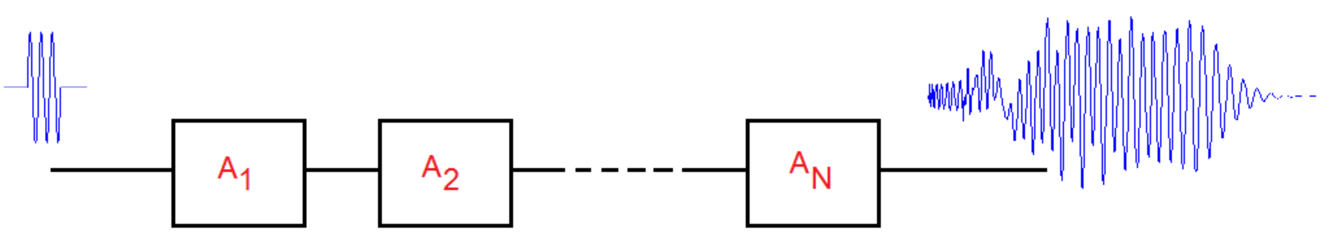


Fig. 3 All-Pass Cascade

The formulation of the transfer functions of the all-pass section is straightforward. The pole p_1 is at:

$$p_1 = r\cos(\alpha) + j\,\mathrm{r}\sin(\alpha) \tag{1}$$

$$p_2 = r\cos(\alpha) - j\,\mathrm{r}\sin(\alpha) \tag{2}$$

$$z_1 = \left(\frac{1}{r}\right)\cos(\alpha) + j\left(\frac{1}{r}\right)\sin(\alpha) \tag{3}$$

$$z_2 = \left(\frac{1}{r}\right)\cos(\alpha) - j\left(\frac{1}{r}\right)\sin(\alpha) \tag{4}$$

$$H(z) = \frac{(z-z_1)(z-z_2)}{(z-p_1)(z-p_2)} \tag{5a}$$

$$= \frac{z^2 - 2z\left(\frac{1}{r}\right)\cos(\alpha) + \left(\frac{1}{r^2}\right)}{z^2 - 2zr\cos(\alpha) + r^2} \tag{5b}$$

and H(z) can then be made unity gain by multiplying by r^2.

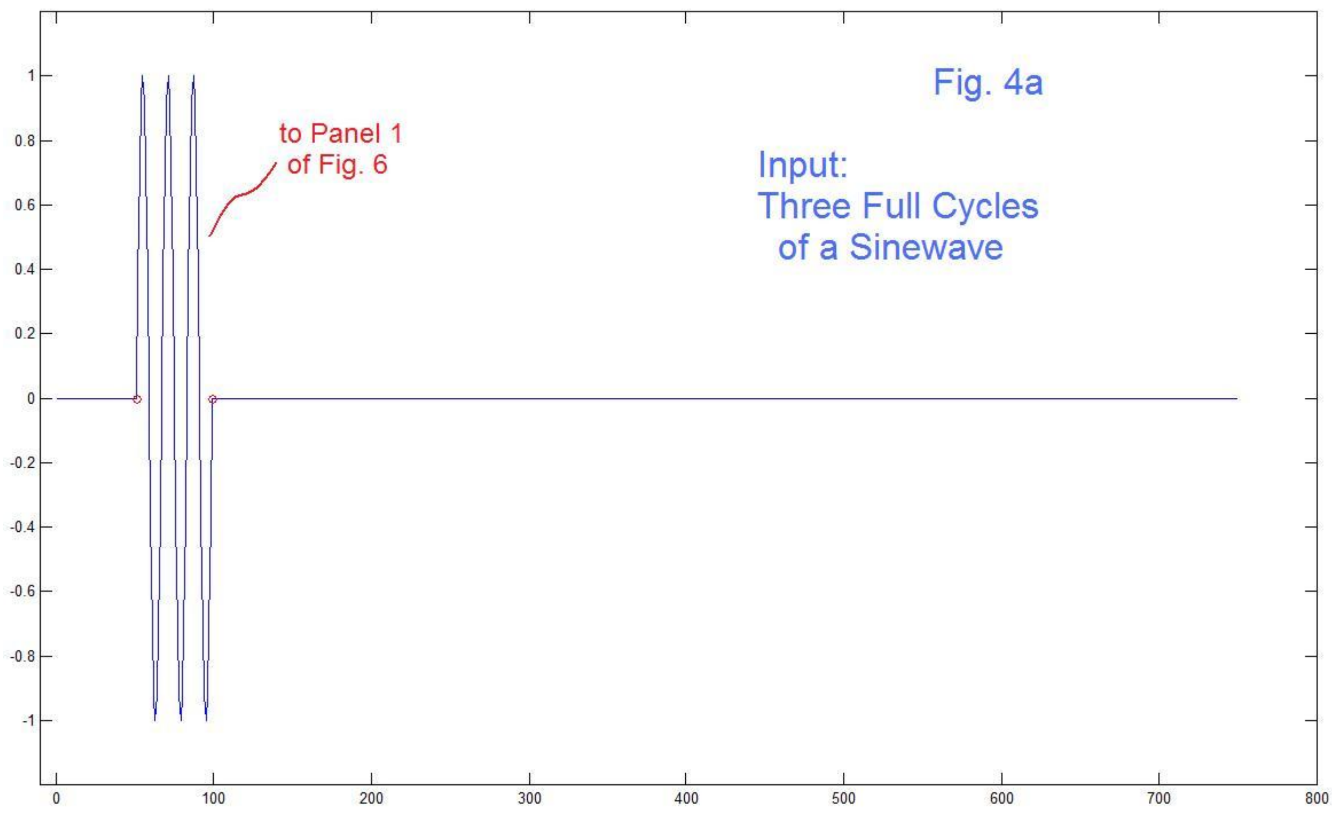
The Matlab code snippet below shows how the numerator and denominator can be set up from the pole radius r and angle alpha. It also shows how the signal x1 is filtered N times with this filter. (Note that taking real parts in the code is just to remove a tiny accidental numerical round-off imaginary component).

```
p1=r*cos(alpha)+j*r*sin(alpha)
p2=p1'
z1=(1/r)*cos(alpha)+j*(1/r)*sin(alpha)
z2=z1'
num=real(poly([z1 z2]))
den=real(poly([p1 p2]))
for k=1:N
     x1=(r^2)*filter(num,den,x1);
end
```

AN EXAMPLE CASE

Here we will look at an example case in great detail. This will illustrate the method and indicate the large amount of data that we have that we can examine, even from one test case. In fact, the output is so profuse that it also should suggest that the parameters of this test are somewhat arbitrary since it is hard to examine enough cases to choose a "best example".

Here the input is chosen to be three full cycles of a sinewave that corresponds to a frequency of 1/16 of the sampling frequency (an angle in the z-plane of 22.5°). Thus there are 16 samples per cycle and 48 samples total. We choose 25 passes through the same all-pass filter for which we select a pole angle α=30° (or 1/12 the sampling frequency) and a pole radius of r = 0.9. Figure 4a shows the input, and Fig. 4b shows the output. The 48 samples of the input signal here are offset from zero, starting at 51, just for clarity. Further, the red circles and notations have been added for partitioning of FFT slices to be described later. For the moment, note that the output is of a much longer duration than the input and clearly shows a frequency that seems to be changing (increasing for this example). Note also that while the output is longer, it is also smaller in amplitude. The output is not finite duration (the all-pass filter is IIR), so each pass adds to the length by about the length of the all-pass impulse response that it significant. Here we have shown about 750 samples.



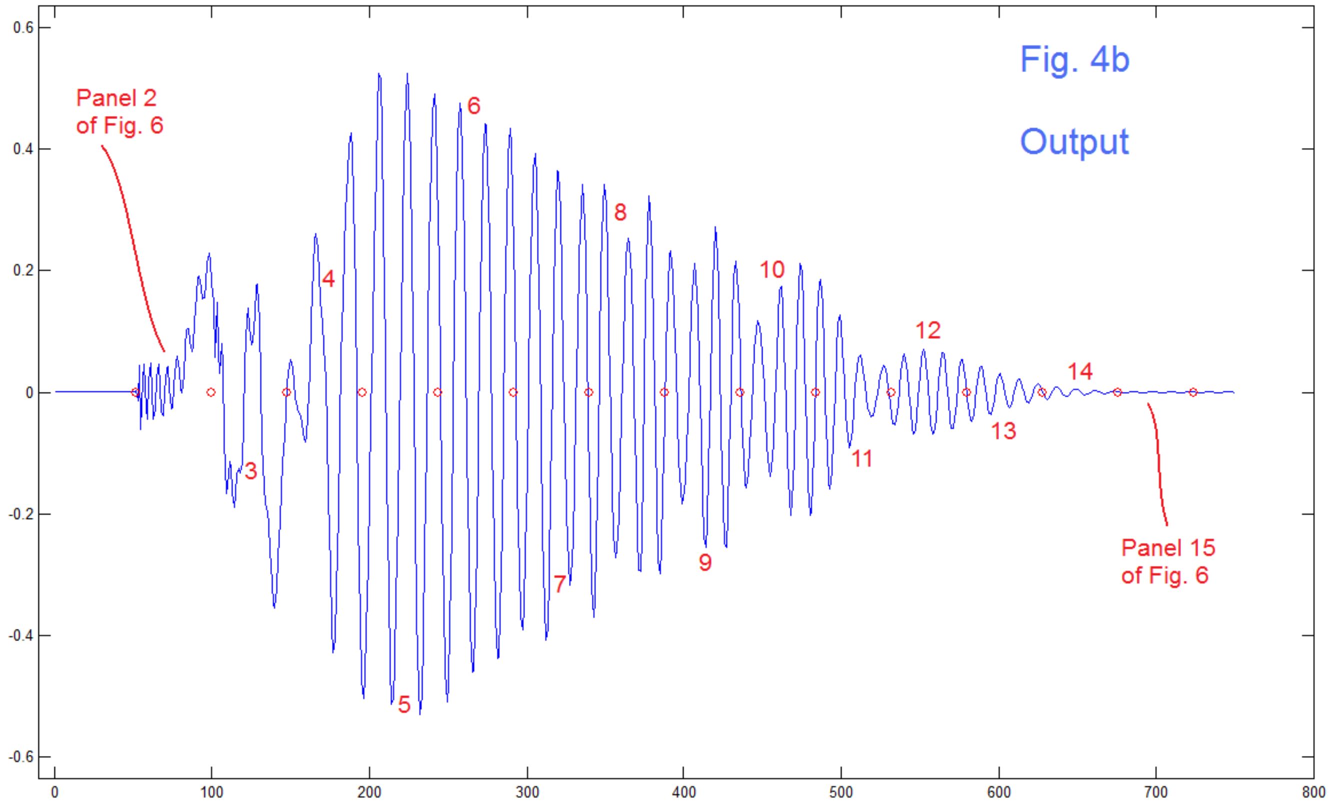
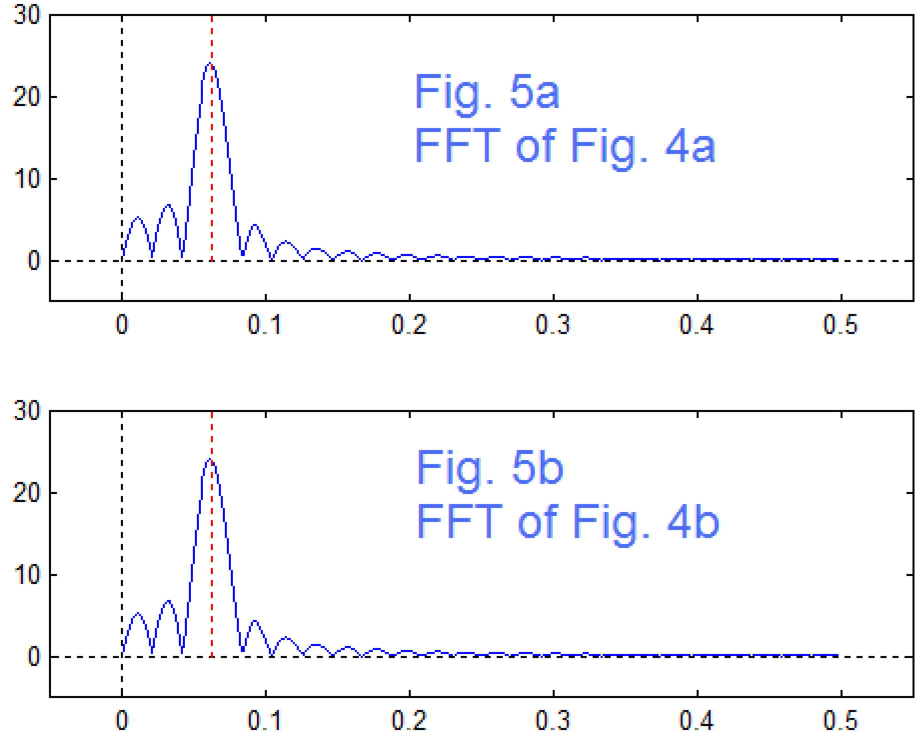


Fig. 4a and Fig. 4b look very different and sound quite different as well. Listening to Fig. 4a we hear the expected ever-so-slightly pitched short “thump”. It is short enough that we hear “one thing” as its duration is within the time-constant of the ear, and we do not hear what would be annoying turn-on and turn-off transients in a longer burst. In contrast, Fig. 4b is very noticeably pitched, and we detect a shifting pitch. However there are no turn-on or turn-off transients here as the signal is clearly self-enveloped. It is no surprise, perhaps, that these look and sound quite different. Is anything the same? Yes.

Figures 5a and 5b show the spectra of the two signals, corresponding to Fig. 4a and Fig. 4b. They are essentially identical. (There is a slight difference since 4b is, after all, ever so slightly truncated.) For certain, the FFTs of the two are different, but as is our usual practice, here we are looking at the magnitude FFTs and we have agreed that all the all-pass filters do is rearrange the phases. So this should not be any surprise. Indeed, if they were not the same, we would have found ourselves searching for an error.

In this light, we further understand the importance of the time constant of the ear. We often think of ourselves as hearing a Fourier Transform magnitude, the ear being considered “phase deaf” according to Ohm’s acoustic law. Here we have mathematical evidence of the magnitude spectra being the same – yet we hear a difference. Thus, we appreciate that the ear is most concerned with the newest material arriving, and in some sense discards older sound. This observation is old news. But it does indicate that if we want to examine this further, we are going to need to subdivide the output.



Before we go on to partition the output signal, some comments about the FFT calculation should probably be made. The FFTs in Fig. 5 were both of length 750 signals, and the horizontal frequency axis is in units of the sampling frequency (running from 0 to 0.5 as is traditional). When we use shorter segments (indeed, we shall use segments of length 48), shorter length-48 FFTs will be used and the frequency recalibrated in the usual way.

It has perhaps crossed the reader's mind that the FFT of the three cycle tone burst should have been represented in its lower half just by a single spike at k=3, which would have been, 0.0625 (1/16) of the sampling rate. (Indeed, a bit later we will see this in "Panel 1" of Fig. 6.) The FFT of the 48 samples would have been a single spike. This is not a single spike because it is a tone burst of length 48 surrounded (as seen in Fig. 4a) by 702 zeros. Indeed, if our available spectral analysis tool is just the FFT, and we need to <u>estimate</u> the Fourier Transform of a non-periodic signal (like just three cycles), our method is in fact to insert the signal ("zero pad") in many zeros and take the larger FFT. Here we did this same thing just to make sure the FFTs in comparing the input and the output were the same length. Note that we did this intuitively, and that if we had not, we would in fact have not seen the exact same spectra in Fig. 5. It would have been like using two different measuring tools.

Our next step is to partition Fig. 4b and take the FFTs of segments of length 48 instead of the entire 750 samples. This we look at as a means of simulating the finite window length of the ear and the way we hear the pitch changing. Fig. 5 shows both the original input and the dispersed output as having the same magnitude FFT, yet they appear in the graphs (Fig. 4) and sound different. Fig. 6 shows the segmented FFTs in a series of 15 panels total.

The first panel is special because it is the FFT of the original 48 samples of Fig. 4a. Note that as we mentioned, the result is a single spike at the frequency of 1/16=0.0625, as here we have no zero padding. This is expected.

Note from Fig. 4b that we have segmented the output into 14 partitions numbered 2 through 15. We then took the individual magnitude FFTs of these partitions and plotted them as panels of Fig. 6. In general, we note three things.

(1) They are not single spikes, although Panel 6 is very similar to the original input of Fig. 4a, except it is smaller.

(2) The earlier panels show clustering at lower frequencies (below 0.0625) while the later panels cluster somewhat above the original frequency of 0.0625,

(3) The energy decreases in the later panels (note well the change of scale) as is also clear from the time waveform of Fig. 4b.

Fig. 6

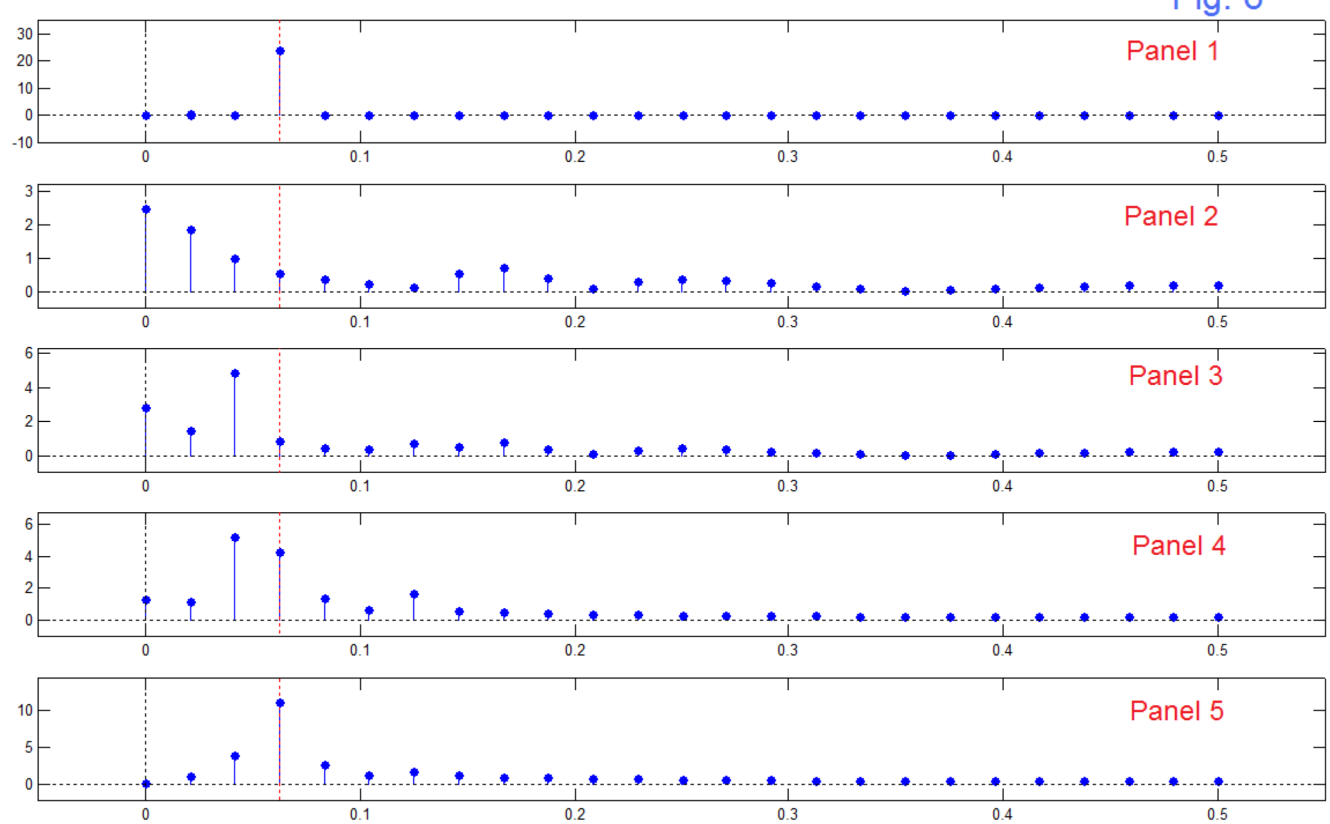


Fig. 6

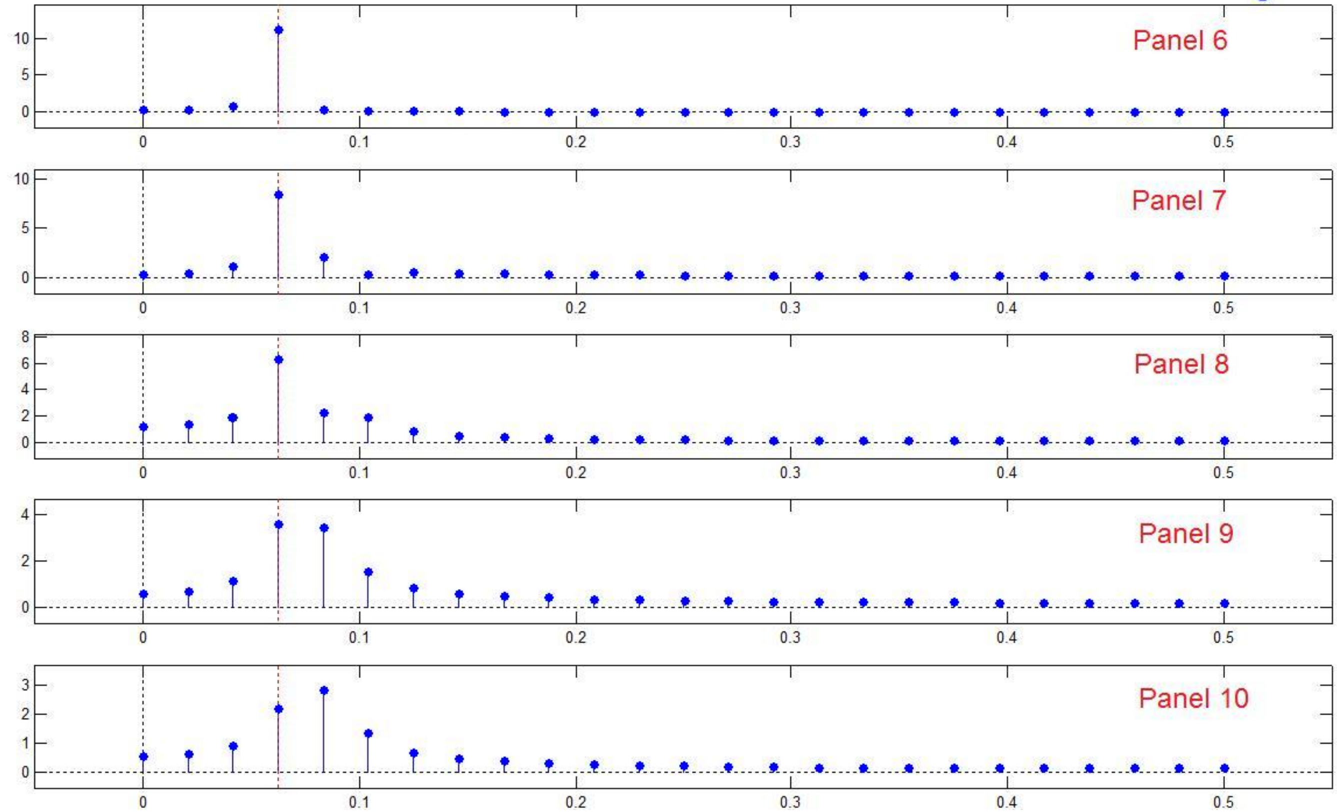
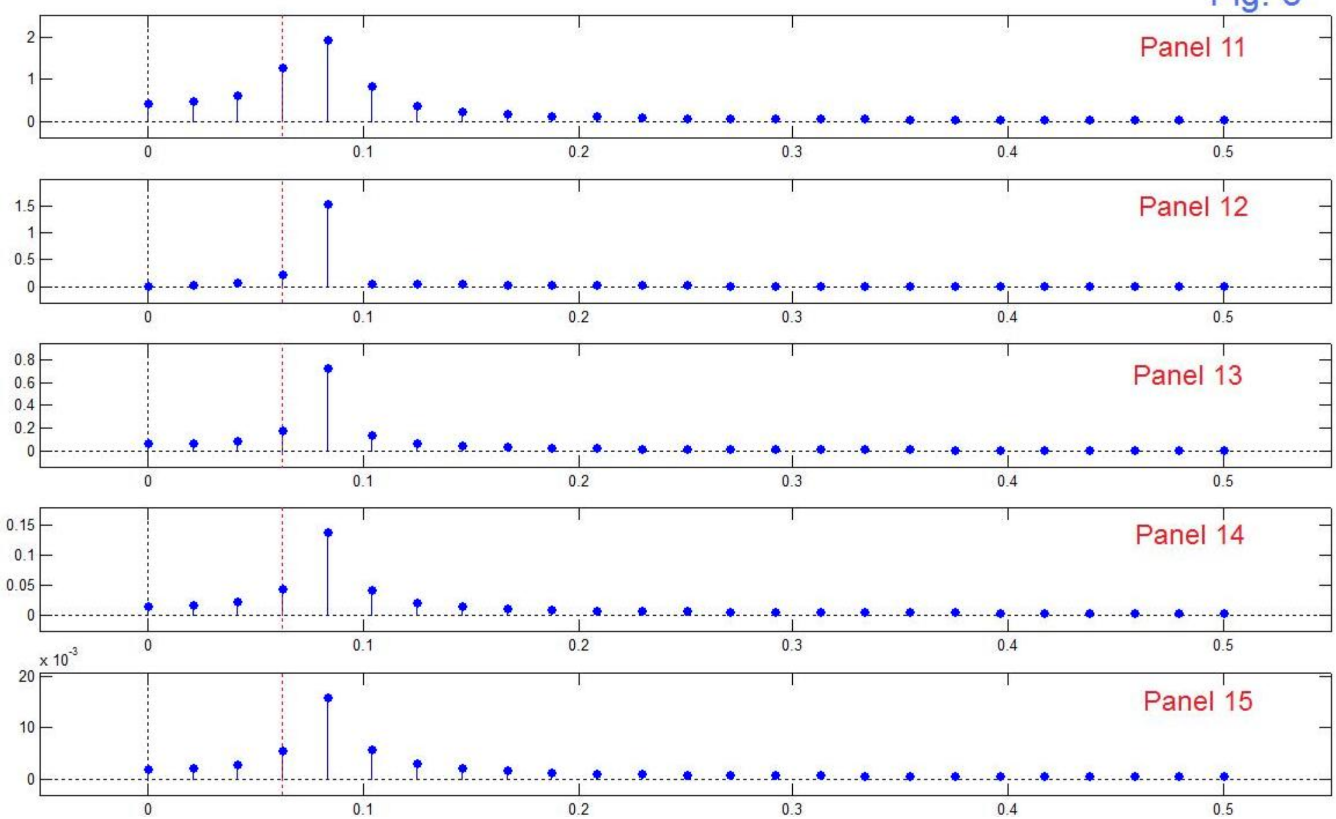

Panel 6
Panel 7
Panel 8
Panel 9
Panel 10

Fig. 6


Panel 11
Panel 12
Panel 13
Panel 14
Panel 15

This display of FFTs as a series of panels is instructive as well as laborious. That is – laborious to examine – once the program code is done you just press enter for each test. Perhaps more to the point, much of what we learn from the FFT series was already evident to us by an examination of the time waveforms. For one example, we note that the nicely resolved spike of Panel 6 corresponded to a segment of the time waveform that was very similar to Panel 1 from Fig. 4a. For Panel 6, somewhere in the middle, the output looked like the input.

As another example, we have said there is a general trend from low frequencies to higher ones as we go through the panels. Indeed, the first partition of Fig. 4b looks to be little more than 1/4 of a cycle, and the corresponding FFT shows lots of energy at k=0, 1, and 2, as expected. Yet, it is also clear from the time domain that there is a good amount of much higher frequency material (onset transients?) in those first few partitions of Fig. 4b, and this energy is seen in the FFTs – like in Panels 2, 3, and 4.

Thus in our follow-up studies here, we rejoice in the ability of the eye to examine the time-domain plots for the essential details. We don't need the FFT so much. Clearly we do not need to take FFTs of the whole output, as we see (Fig. 5) that it is just the same as that of the input in all cases. Accordingly plots such as Fig. 4b will serve us well. Not to mention that these plots are immediately interesting and informative, and complex enough to be entertaining (a fun program).

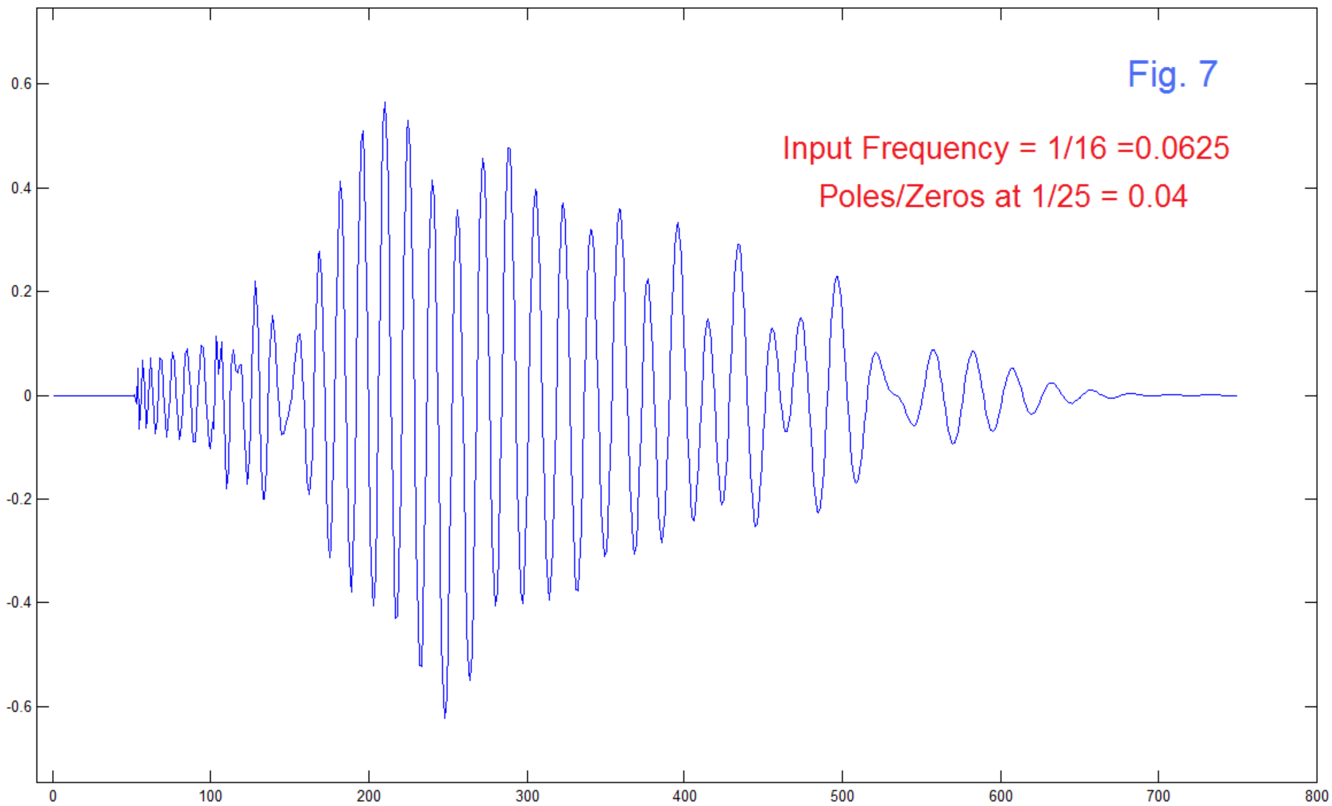
REVERSING THE FREQUENCY TREND

We do not wish to claim that the dispersion results are inherently useful as musical tones. Possibly each one is interesting and the outputs are at least as interesting as the inputs, and in must cases more so as they have a more dynamic (evolving) spectral structure. So our scheme here is to use both a visual and a sound presentation and to ask if we might well use the output as one tone of a musical presentation. The answer seems to be a general "yes" with the proviso that the sounds, having a detectable varying pitch to some degree, tend to be tones of a lighter if not a comical nature.

[It is worth noting that in the largest picture, we might look at the dispersion filter not as a tone-by-tone "generator" but as a processor such as a more traditional filter bank. That is, our musical input could be a series of tone bursts (a melody if you prefer) input to the all-pass bank with each different burst getting a different "twist", much as a traditional filter bank provides a fixed background "formant" to characterize the impression of an individual instrument.]

If we are concerned about individual tones, we need to ask about how well a particular generation method treats our general "low-pass" model of a musical sound as we generally expect them to occur. That is, acoustical instruments (physical instruments made of usual mechanical materials such as strings, wood, metal, plastic, air columns, etc., very generally behave in a manner of a low-pass filter. Higher frequencies decay (mechanically damp) faster than low frequencies. Thus all frequencies may occur at the beginning, but it is the lower frequencies that remain at the middle and end of a tone. Because we traditionally have this with our mechanical instruments, we also find this as a preferred implementation with musical tone synthesis. In this light, we might suppose that Fig. 4b has it backward (low frequencies at the beginning, high ones at the end). This is really a matter of placing the all-pass filter relative to the input tone burst frequency.

Fig. 4b had the input signal at a frequency of 1/16 (=0.0625) the sampling rate while the all-pass poles (and zeros) were at a higher frequency of 1/12 (=0.08333) the sampling rate. What if we were to move the poles (and zeros) to a lower frequency. For example, Fig. 7 shows the case where the poles (and zeros) are at a frequency of 1/25 (0.04 of the sampling frequency), thus below the input frequency of 0.0625.



In this case, we clearly see a reversal of the situation of Fig. 4b in that we start with very high frequencies and encounter lower frequencies as the tone progresses. (Yes – we remember that high frequency buzz in the first partition of Fig. 4b.)

Based on very limited data, the examples of Fig. 4b and Fig. 7 and a few more, is the case where the all-pass frequency is below the input frequency “more musical”? This is a hard question to answer, but I think most listeners would say it is, in the sense of relating to our experiences. Fig. 4b is, as we suggested, slightly more comical. But you can have both available of course. This is parameterized software – not solder!

A DYNAMICALLY VARYING ALL-PASS PROCESS

It is possible and easy to make different passes through the all-pass section have different pole (and zero) frequencies. Essentially the code at the bottom of page (6) is modified so that the filter numerator and denominator are calculated and updated inside the loop. Figure 8 shows the case where the angle starts at 0, and increments by 0.002 times the sampling frequency on each iteration until it reaches 0.048 during the 25th iteration.

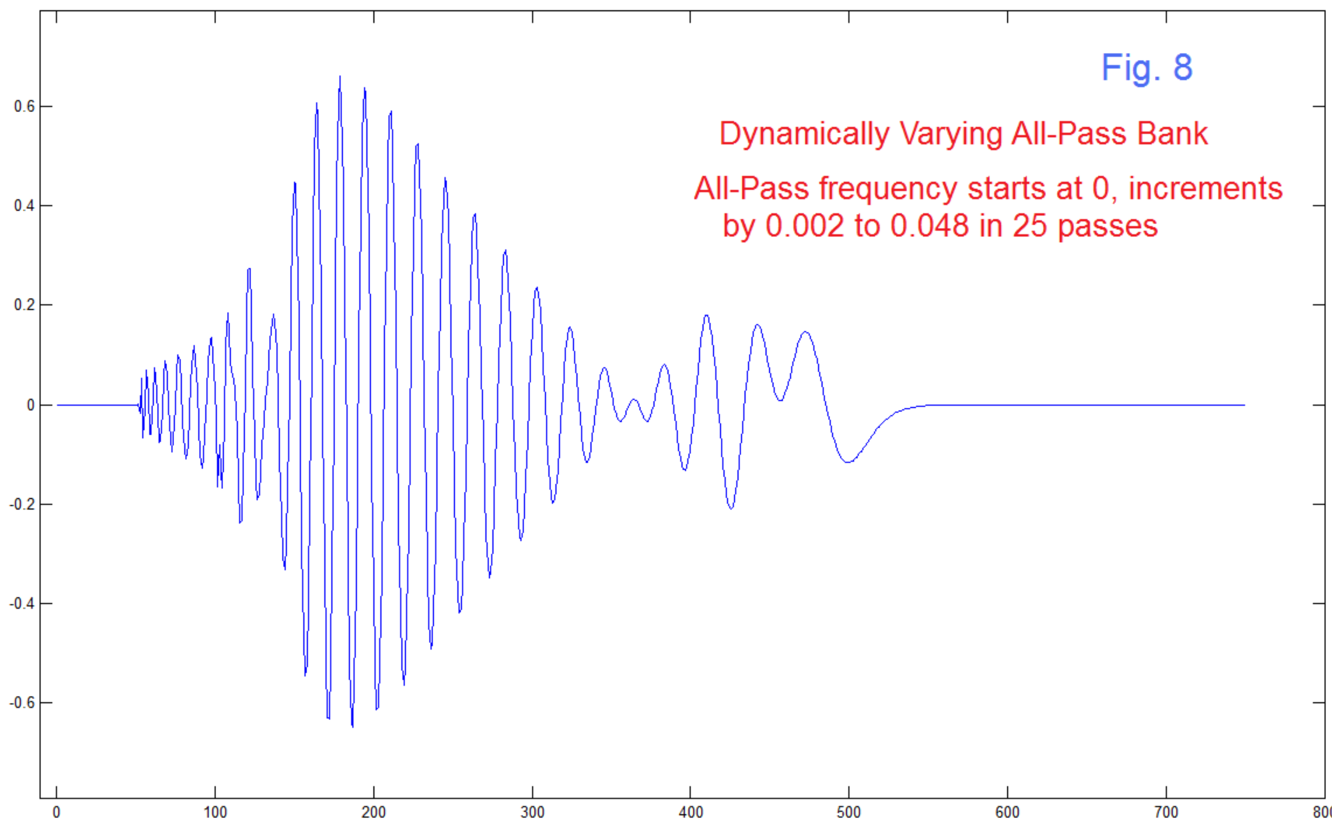


Fig. 8 is not all that different from Fig. 7. Before we forget, keep in mind that all of Fig. 4b, Fig. 7, and Fig. 8 have the same Fourier Transform (Fig. 5).

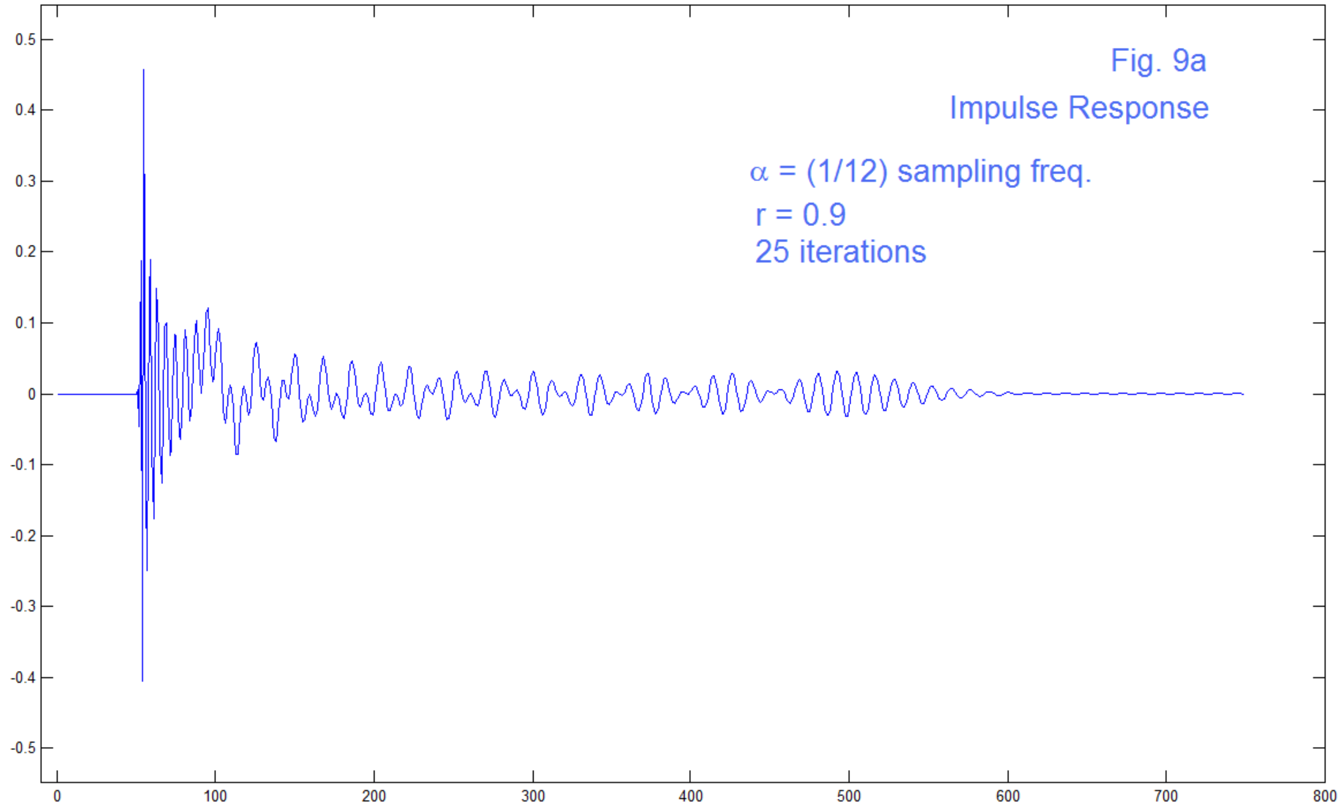


Fig. 9a
Impulse Response
α = (1/12) sampling freq.
r = 0.9
25 iterations
0.5
0.4
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
-0.4
-0.5
0
100
200
300
400
500
600
700
800

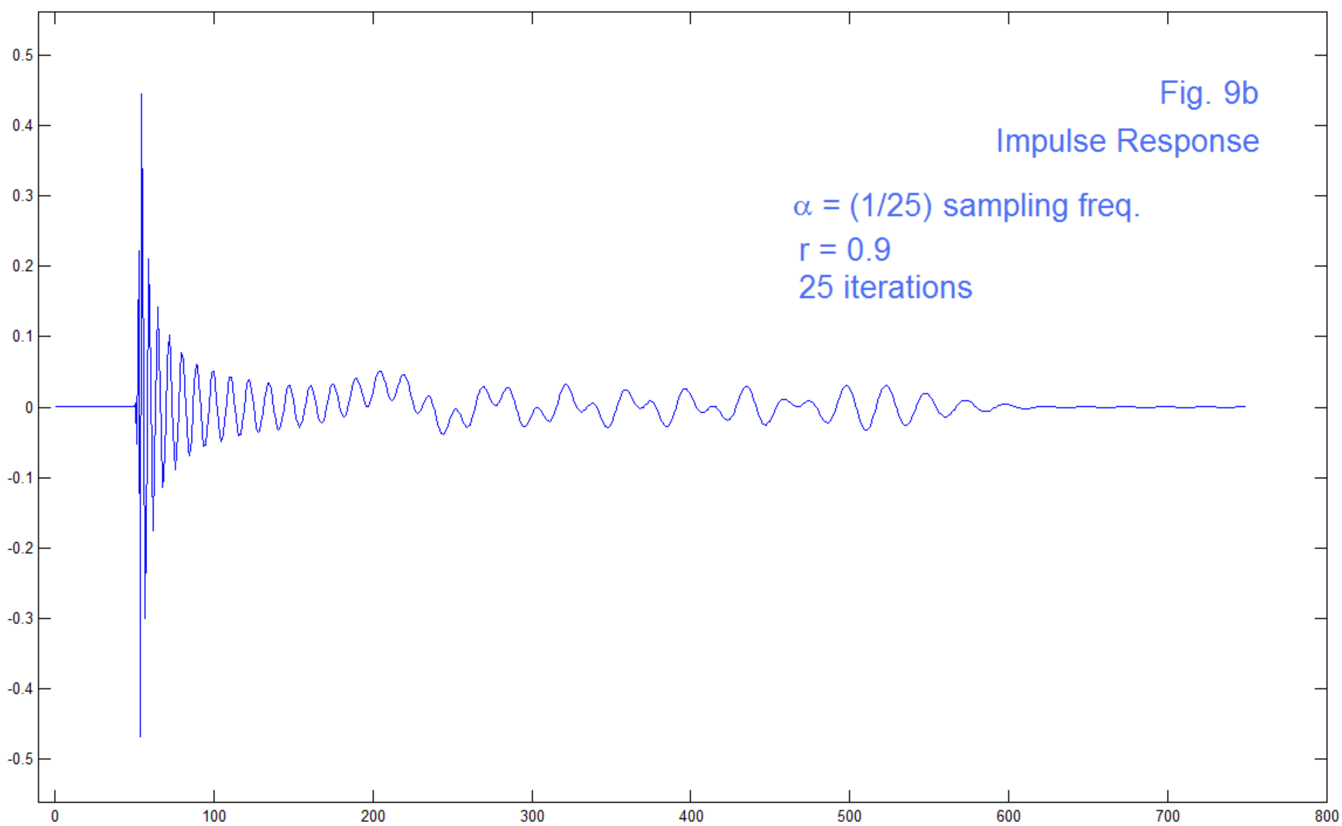


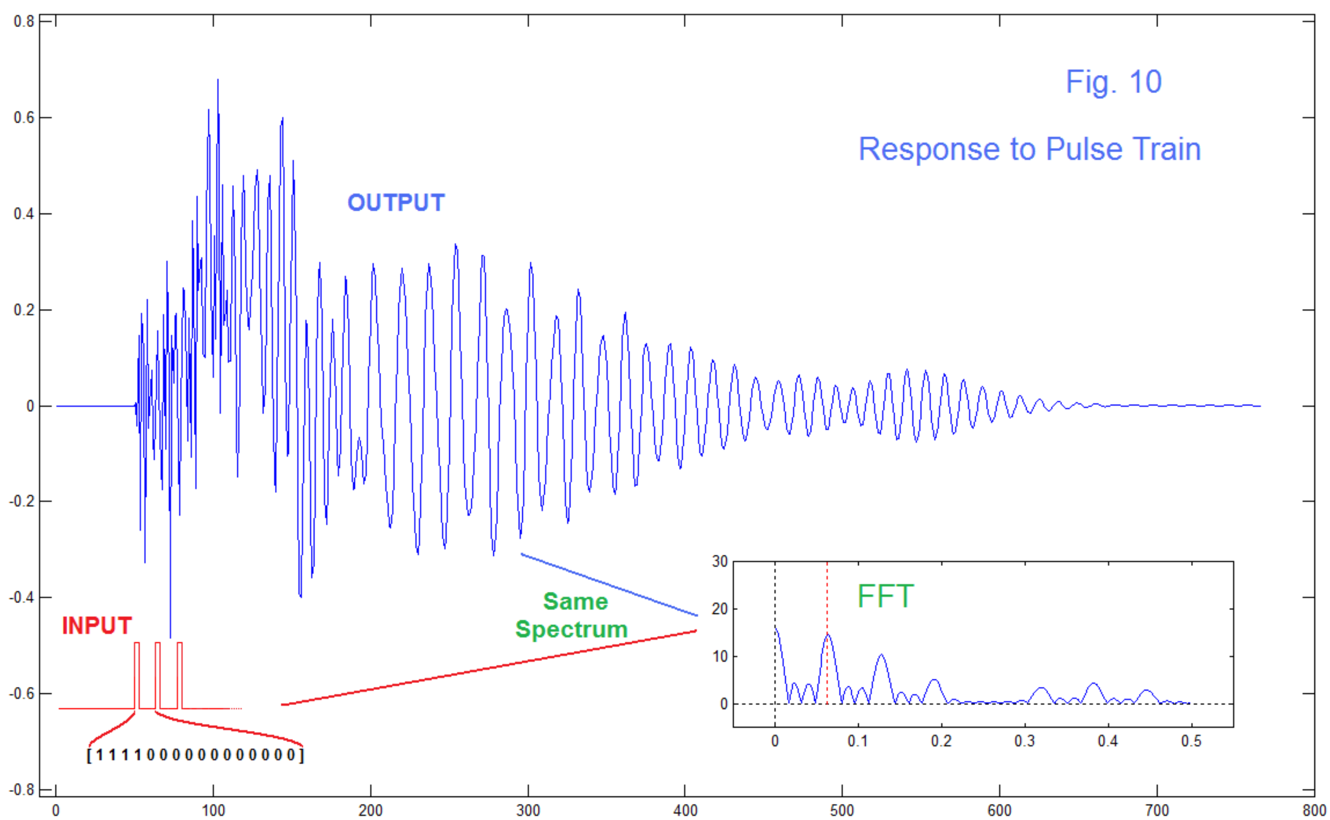
Fig. 9b
Impulse Response
α = (1/25) sampling freq.
r = 0.9
25 iterations
0.5
0.4
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
-0.4
-0.5
0
100
200
300
400
500
600
700
800

THE IMPULSE RESPONSE VIEW

We generally think of the notion of an impulse response of a filter as being fundamental, and this is something we need to look at here too. So far, our inputs have been tone bursts, so there was a frequency associated with the inputs. An impulse has all frequencies equally, so one "dimension" of our investigation is removed. Fig 9a and Fig. 9b show two examples of impulse response for the case where the radius is 0.9 and the angles are 1/12 and 1/25 times the sampling rate, respectively. Very roughly the two are the same except for pole frequency. Keep in mind that there are two poles, and that for the smaller angle they are closer together. So we still have two "dimensions" to the study. What we see is that both are from higher frequencies to lower ones, but the total view is much more complex and needs further study.

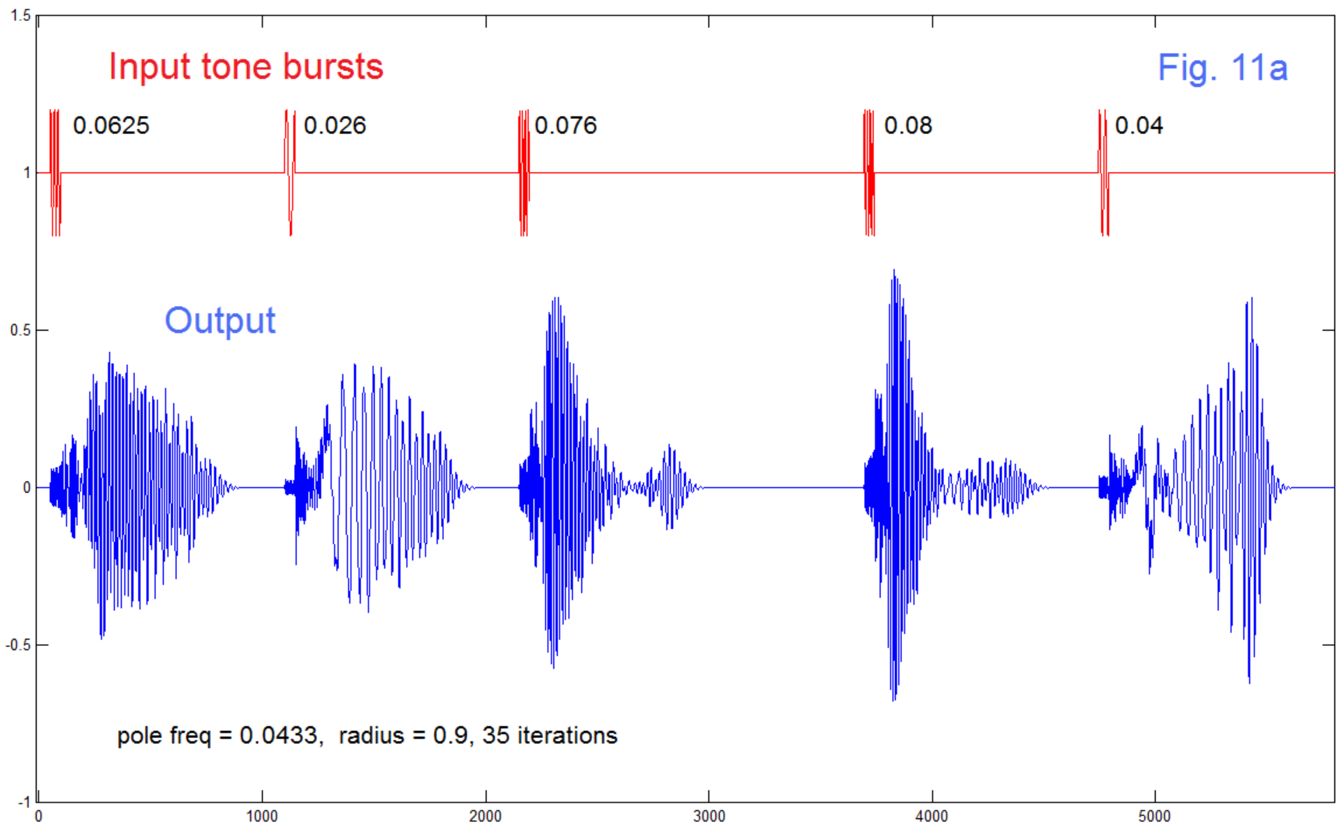
A COMPLEX WAVEFORM IN (PULSE TRAIN)

Another "can or worms" appears when we add a "dimension" to the study by letting the input be a complex waveform rather than just a sinewave burst. Here we have just shown one example, where the input is three cycles (16 samples each cycle) of a 1/4 duty cycle pulse. The reference here is the inset figure showing the spectrum of the output (same as input).



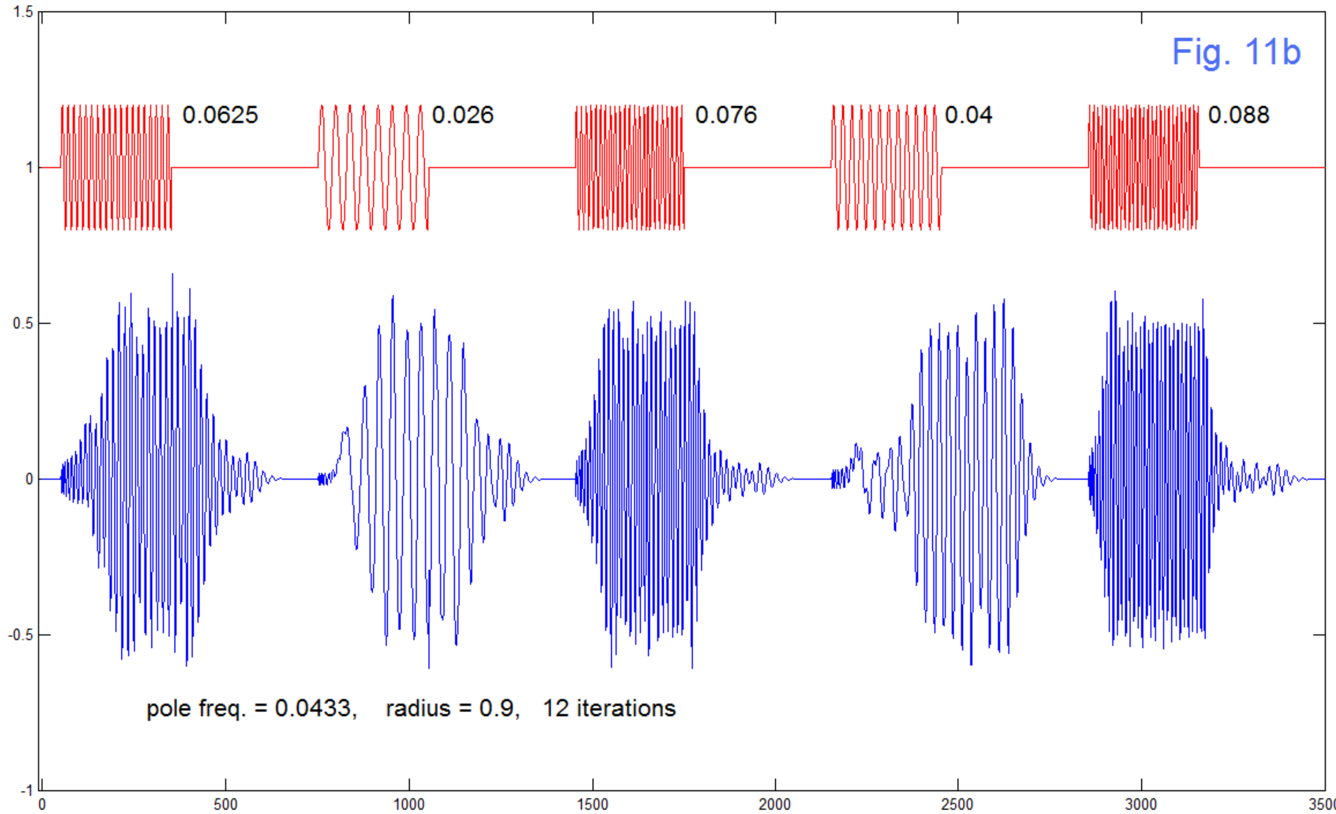
THE DISPERSION "FILTER BANK"

Back on page 12 (the red text), we suggested using the all-pass filters in the manner of a filter bank. In this application, a series of simple sinewave tone bursts would constitute an input sequence of notes to be processed (a "melody"). Each "note" would then be used to from an output sequence of more complex tones, as the individual burst happens to be processed. Fig. 11a shows an elementary example. Here the input tone bursts are all relatively short – just a few cycles, with frequencies as indicated (relative to a sampling frequency of 1). The tone bursts are 48 samples long. There are 35 passes through the all-pass. We see considerable differences in the shapes of the output tones (blue).



Above we showed that the positions of the frequencies of the tone burst (relative to the pole frequency of the all-pass filters), spread the tones and skewed the original frequency alignment. Here changing the burst frequency relative to the fixed dispersion filter bank makes each tone an individual item visually, and aurally to a lesser apparent extent. The details of the processing of the notes are not clear at this scale. Indeed, each of the five tones shown could be analyzed in detail as in Fig. 4 and Fig. 6. It is clear that we have added an individual character to inputs that were just a few cycles of sine waves. (Scales vary for plotting purposes.) Of course each blue tone has the same spectrum as the red tone burst above.

In the example of Fig. 11a we have chosen a short tone burst and a relatively large number of all-pass iterations, and the results are visually dramatic. It is interesting if we change these input parameters a bit, as in Fig. 11b.

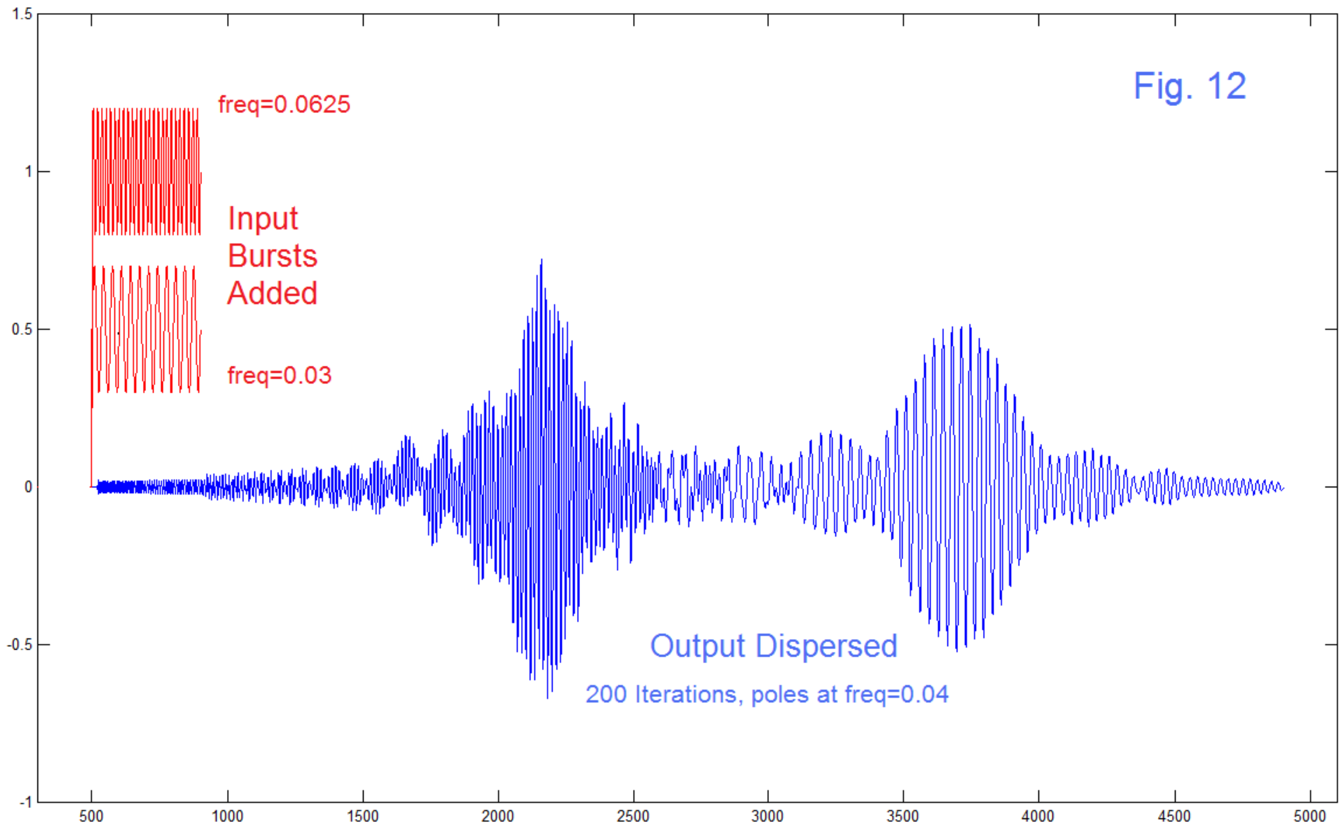


Here a couple of input frequencies are changed, but mainly we have made the red bursts longer (length 301) and decreased the number of all-pass iterations to 12. Visually, there is far less variations (blue in Fig. 11b relative to blue in Fig. 11a). Aurally, there are lesser but still distinct differences in the sounds of the individual notes. (A lot depends on the playback rate.) What is perhaps most striking here is the time shifts of the bursts. The intervals between bursts are exactly equal in the red bursts of Fig. 11b, but clearly they vary their spacing in the blue versions. One possible idea as that an exact regularity in an electronically generated sequence would be broken up to something more like the irregularities found in human playing – but this is just a suggestion.

One possible advantage to this bank is that it is a series structure. This we can readily do because the filters are all-pass. This is in contrast to a more familiar bandpass filter bank, which usually must be a parallel structure. The problem with parallel structures is that when we recombine (sum) the channels, phase matters and is a big issue. In a series bandpass structure, each channel is significantly (often strongly) attenuating all other channels, making signal/noise an overriding issue. In theory, a series structure might be possible with a digital bandpass (floating point) implementation.

DISPERSION LEADING TO SEPARATION OF TONES

The classic example of dispersion would be to have two different frequencies, initially in alignment, each with a significant number of cycles, dispersed to the extent of being actually seen to be separated. This we show in Fig. 13 for an interesting example. Here we have a high frequency (0.0625 times the sampling rate) and a low frequency (0.03 times the sampling rate) as shown in red. The signals are added and subjected to 200 passes though the all-pass filter with a pole frequency of 0.04 (chosen between 0.03 and 0.0625) times the sampling rate (radius 0.9) . The separation (and spreading) of the individual tones is clearly shown. This resulting transition of parallel tones to a somewhat stepped structure may be of musical interest.



DISCUSSION

Not much to add here. I don't recall if we ever discussed the possibilities of using dispersion before. Certainly the applications of pure time delay are well studies as various phasers and flangers. Bill Hartman summarized these effects and related them to time delay in a much appreciated paper [1]. Dispersion as such in communications is much studied and is related to such things as "whistlers" in radio communications (VLF radio artifacts of lightning strikes) and indeed, to the spreading of light in a prism. So – it's not new – we just have ignored it.

Here we have been talking about digital implementation. In fact, we have remarked that a study of the phenomenon is largely unthinkable in an analog contest. This is true, and we can consider digital processing of files of acoustically-produced music in this way as well as our notions of synthesizing new electronic sounds. Is an analog realization (as opposed to a study) possible? Sure. We could envision an analog module that has one input. This could lead to series of analog all-pass filters [2]. We would make no attempt to vary the poles/zeros – these would be fixed filters. Presumably some digital study would suggest how to choose the general design parameters. Then perhaps we would have outputs (or a rotary selector switch) along the series, perhaps at 1, 2, 4, 8, 16 networks. Not that much panel space. At some point, you would insert this in the signal path, much as we would a filter bank, "special effects unit", or animator.

REFERENCES

[1] W. M. Hartmann, "Flanging and Phasers", ***J. Audio Eng. Soc***., Vol 26, No. 6, June 1978, pp 439-443

[2] B. Hutchins, "Second-Order All-Pass Networks 1 – 4", Electronotes Application Notes 167-170, March 11, March 18, March 25, April 5, 1980.