AUDIBLE DESIGN

TREVOR WISHART

APPENDIX 2

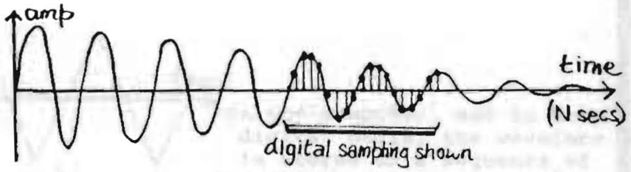
A diagrammatic guide
to sound compositional processes.

Published by Orpheus the Pantomime Ltd : 1994

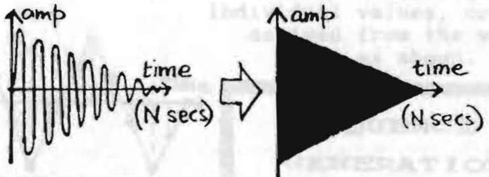
KEY

WAVEFORM :

sound represented in the time domain.

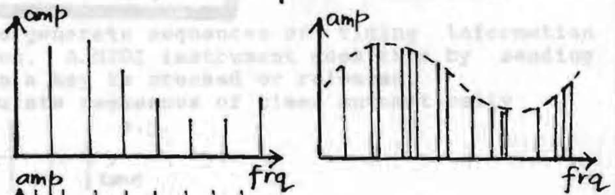


When looking at much longer blocks of time, waveform would be very compressed in display, so a different format is used.

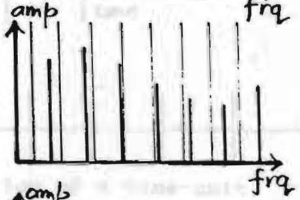


SPECTRUM :

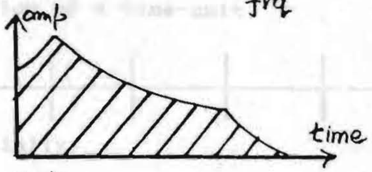
sound represented in the frequency domain. N.B. Frequency axis evenly spaced with respect to frequency, & NOT to pitch.



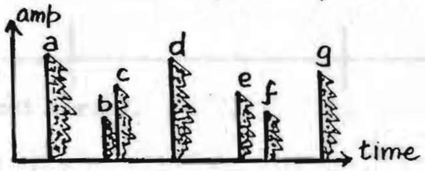
Spectrum, displaying analysis channels.



LOUDNESS TRAJECTORY : (loudness envelope) of a sound.



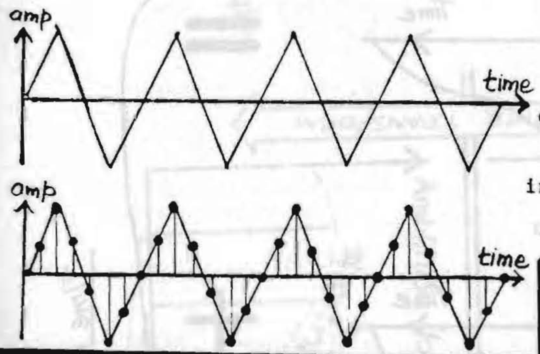
MIX PLACEMENT : represents the times at which sounds in a mix begin. Does NOT display the whole of each sound.



N.B. ALL REPRESENTATIONS ARE SCHEMATIC.

We do not attempt to represent realistic waveforms, spectra, loudness trajectories or mix placements, but only to illustrate the principles involved in the various compositional processes described. Waveform, spectral etc shapes have been chosen to make their transformations as clear as possible in diagrammatic form.

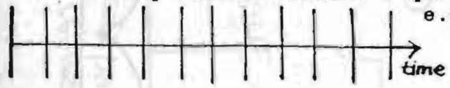
SAMPLING



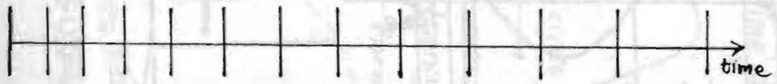
On the computer, and in other digital media, the waveform is stored as a sequence of individual values, or samples, derived from the waveform as shown.

SEQUENCE GENERATION

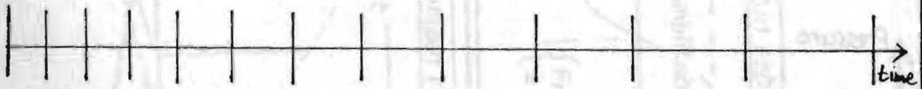
It is often useful to generate sequences of timing information for musical composition. A MIDI instrument does this by sending data when a key is pressed or released.
We may also calculate sequences of times automatically e.g.



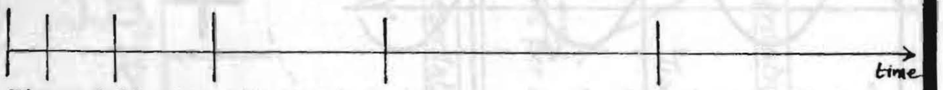
Regular times.



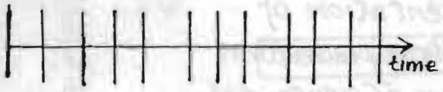
Times growing by addition of a time-unit.



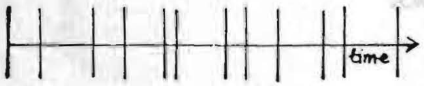
Times growing exponentially.



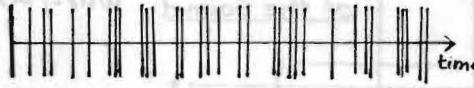
Times following Fibonacci series.



Regular times slightly randomised.

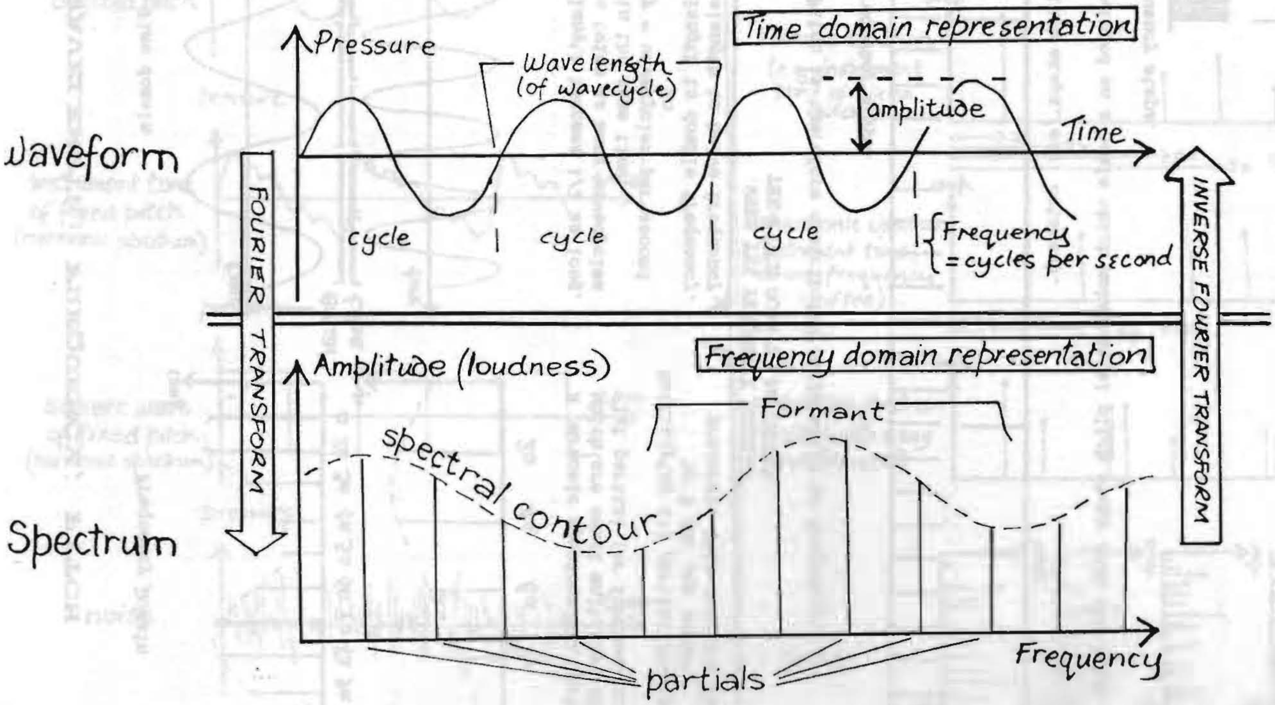


Times completely random.

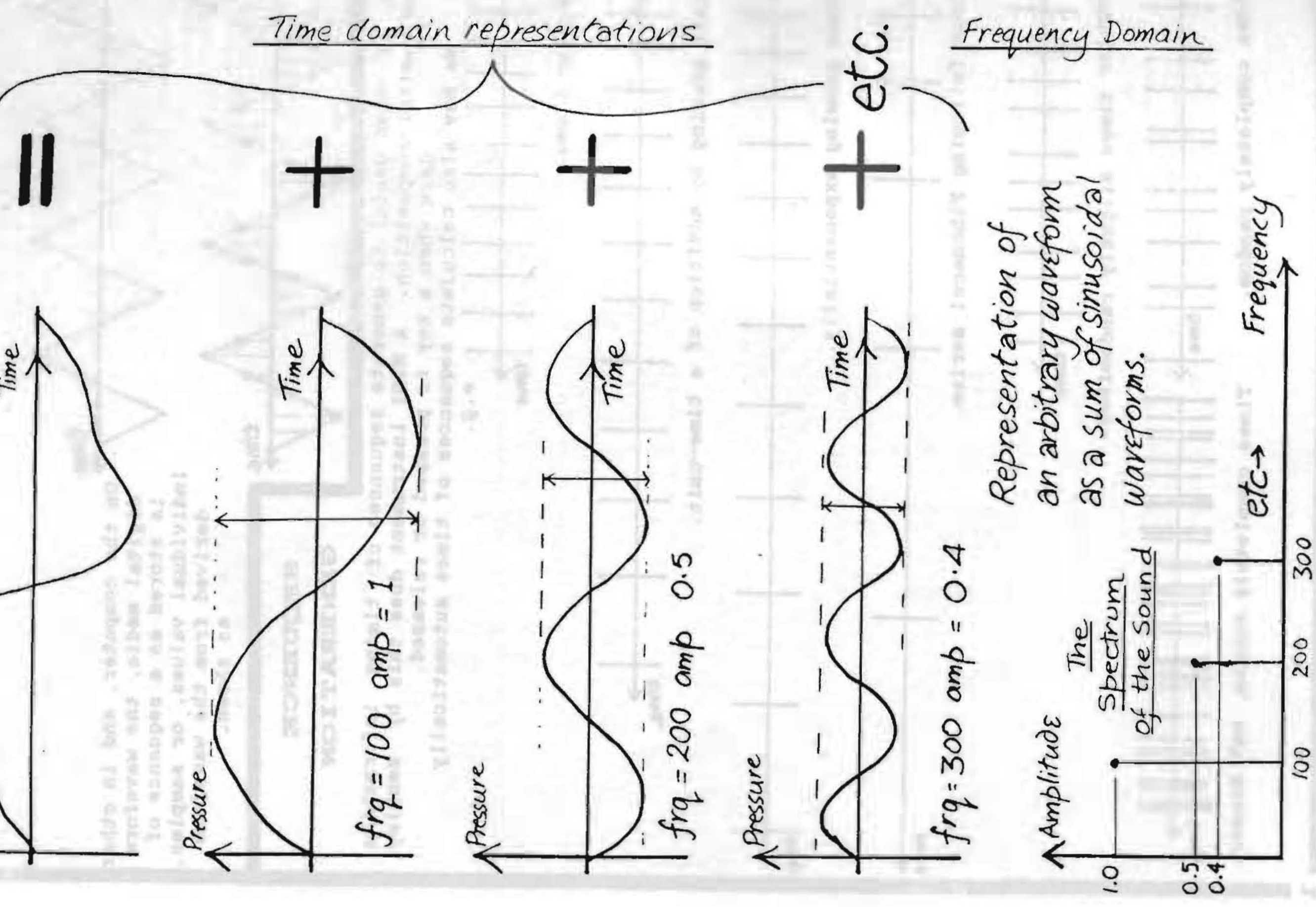


Times completely random, but denser.

TIME DOMAIN & FREQUENCY DOMAIN



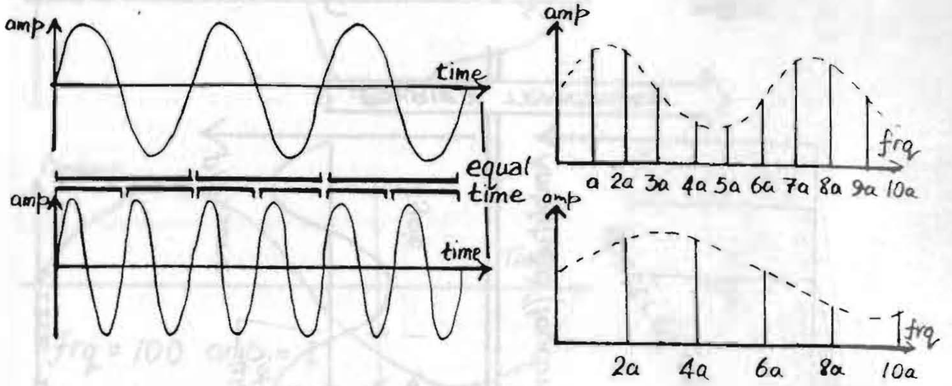
FOURIER ANALYSIS



WAVELENGTH, FREQUENCY, PITCH

Time domain

Frequency Domain



When wavelength becomes 1/2 as long,
there are twice as many wavecycles
in the same time.
Frequency = wavecycles-per-second
So
Halve wavelength to double frequency.
Double wavelength to halve frequency.

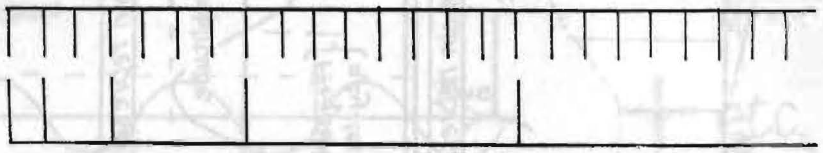
A harmonic spectrum has partials
which are exact multiples of the
1st partial (or fundamental).

Multiplying all partial frequencies
by 2 (or any number)
preserves this relationship.

WHEN THE FREQUENCY DOUBLES
THE PITCH GOES UP AN OCTAVE.

Pitch is therefore distributed differently to frequency.

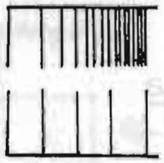
Equal frequency steps.



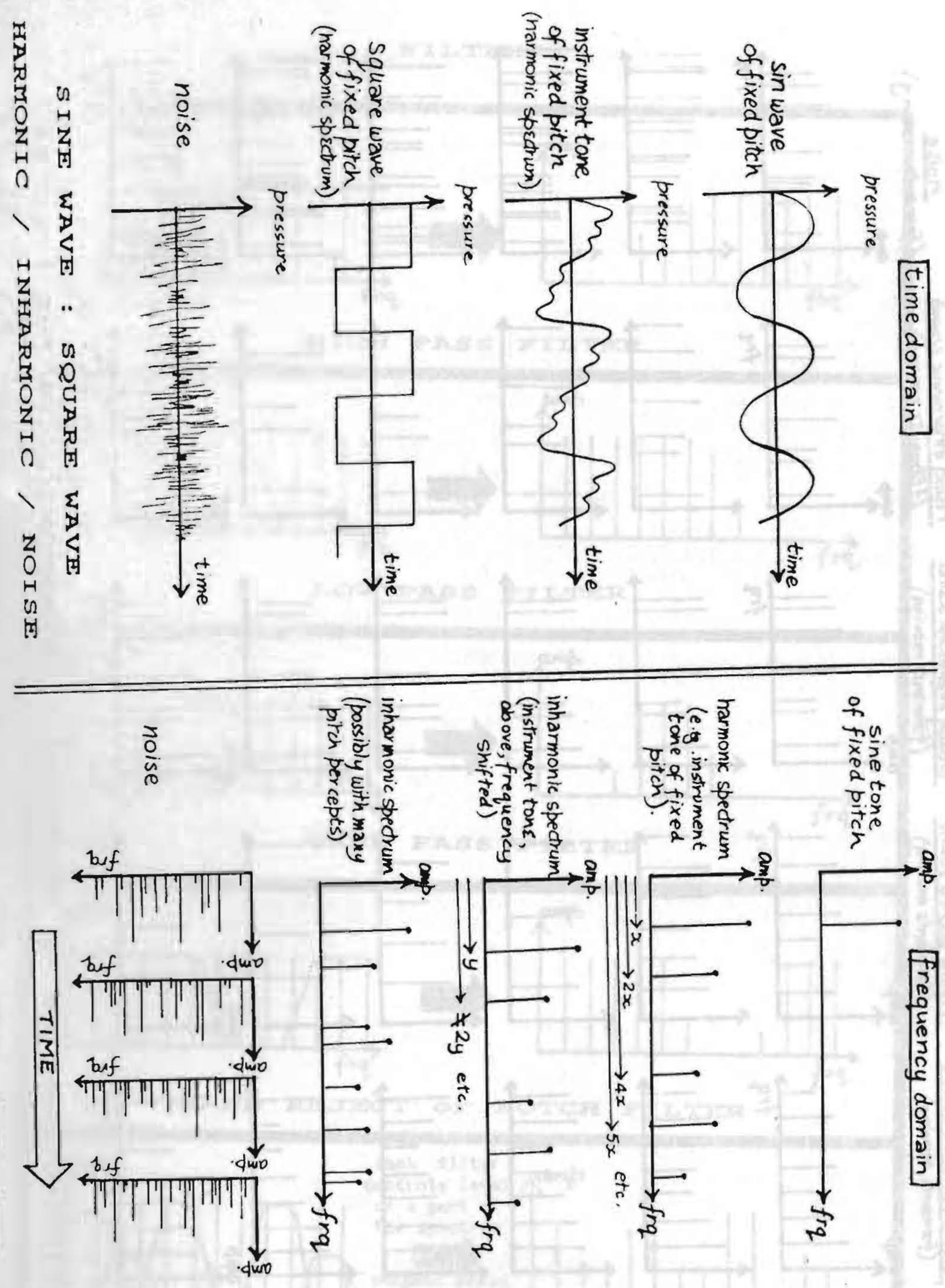
Equal pitch steps (e.g. octaves).

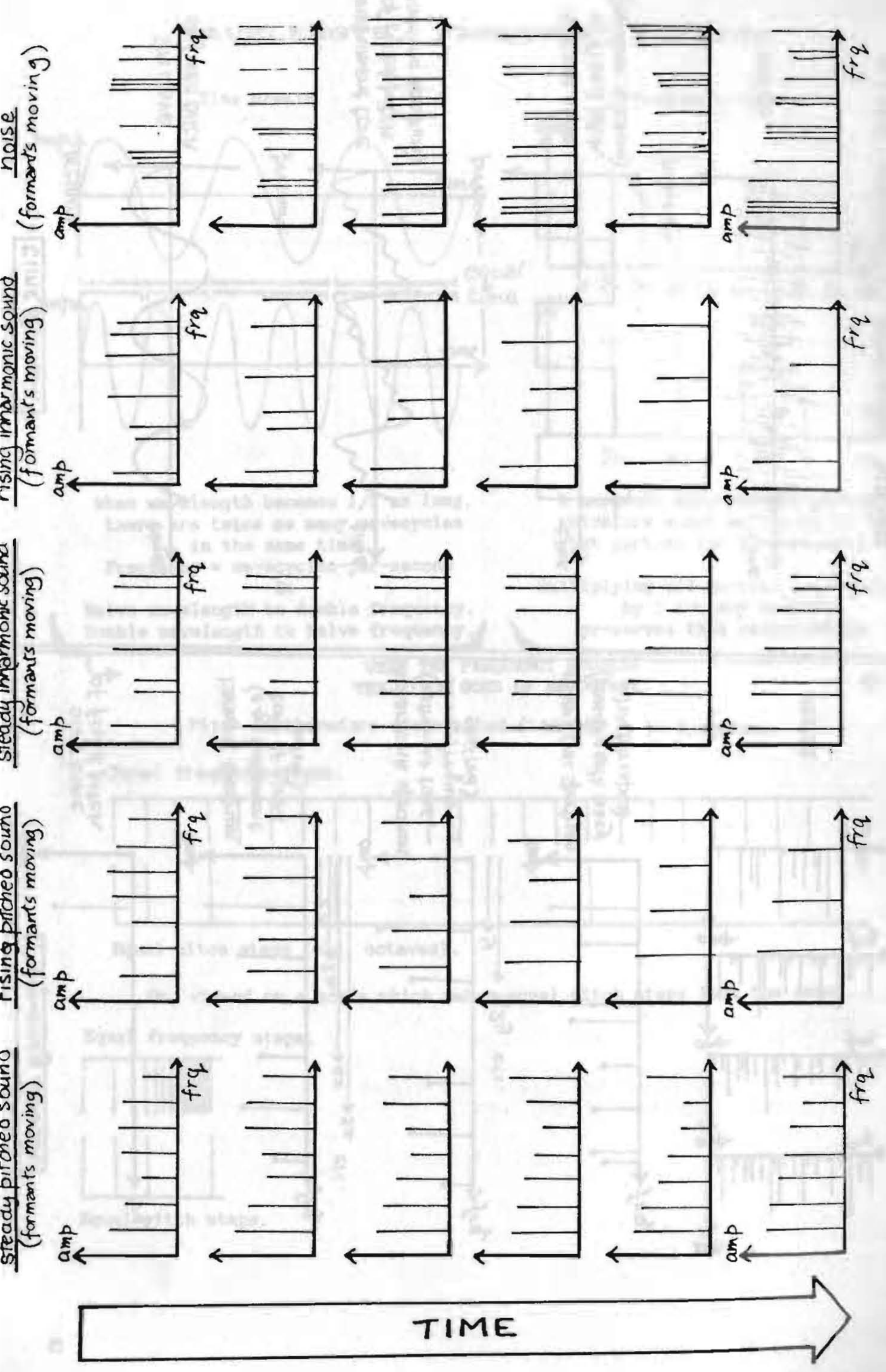
Or, viewed on a scale which makes equal <u>pitch</u> steps look the same...

Equal frequency steps.



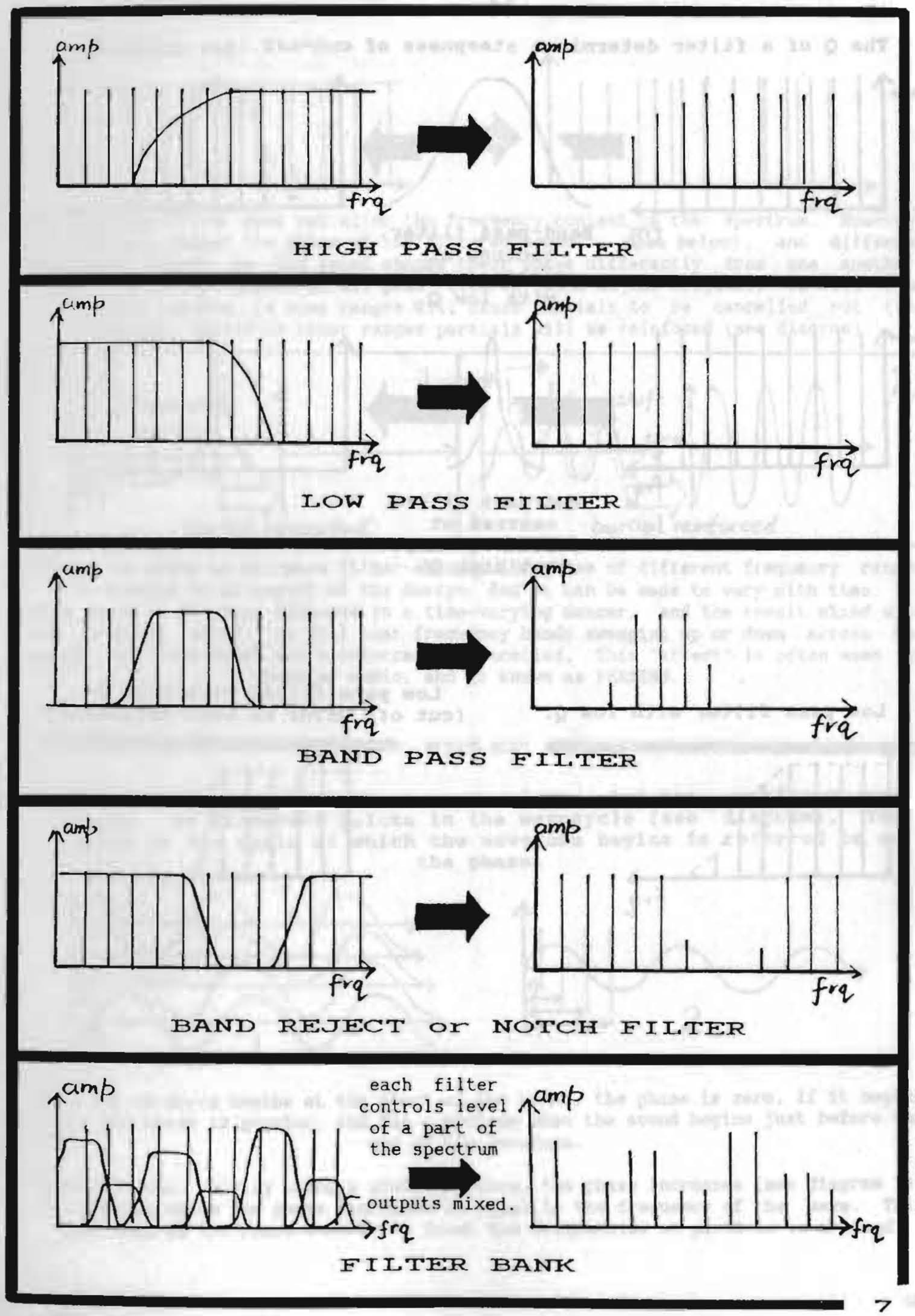
Equal pitch steps.





TIME-EVOLUTION OF HARMONIC / INHARMONIC / NOISE SPECTRA

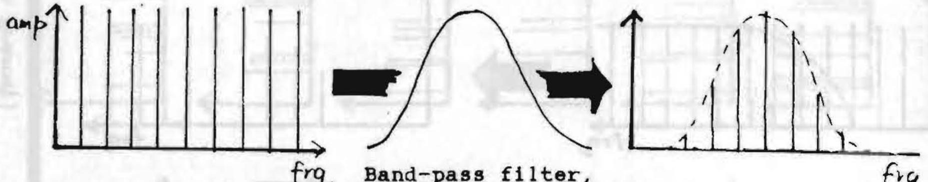
FILTERS



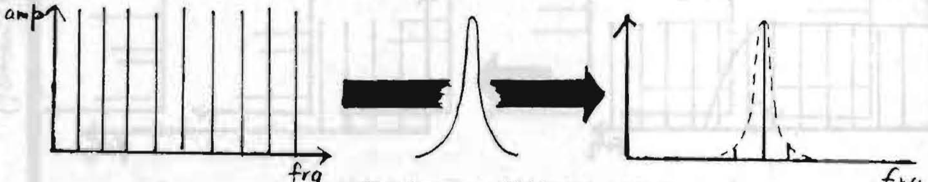
FILTERS continued

Q

The Q of a filter determines steepness of cut-off (see diagrams).

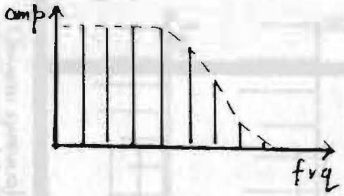


Band-pass filter,
centred at
some frequency,
with low Q.

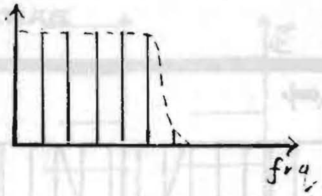


Band-pass filter,
centred at
same frequency,
with high Q.

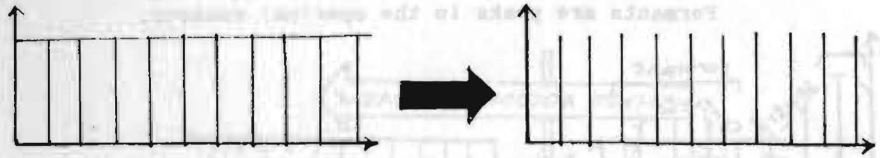
Low pass filter with low Q.



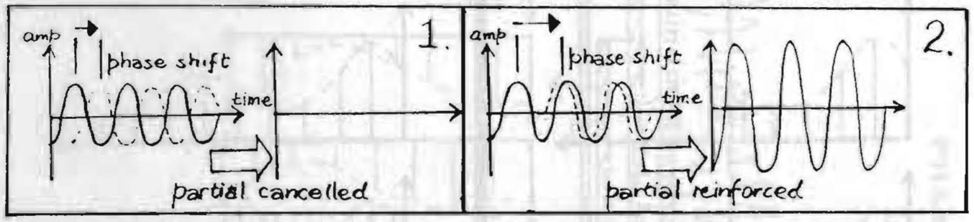
Low pass filter with high Q.
(cut off point at same frequency)



ALL PASS FILTER : PHASING



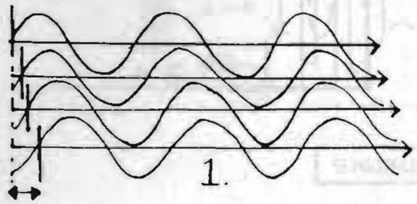
An all pass filter does not alter the frequency content of the spectrum. However, all filters change the phase of the filtered sound **(see below)**, and different frequency ranges in the sound change their phase differently from one another. Hence, if we superimpose an all-pass-filtered sound on the original, we will find that phase shifts in some ranges will cause partials to be cancelled out (see diagram), whilst in other ranges partials will be reinfoced (see diagram).

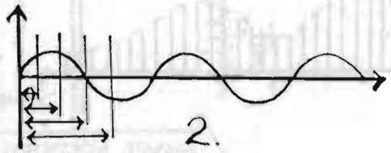


The way in which an all-pass filter changes the phase of different frequency ranges in a sound is an aspect of its design. And it can be made to vary with time. If a sound is all-pass-filtered in a time-varying manner, and the result mixed with the original sound, we will hear frequency bands sweeping up or down across the sound, as these bands are reinforced or cancelled. This 'effect' is often used in popular music, and is known as PHASING.

PHASE

Two sounds may have the same waveform, but the waveforms may begin at different points in the wavecycle (see diagram). The point in the cycle at which the waveform begins is referred to as the phase.



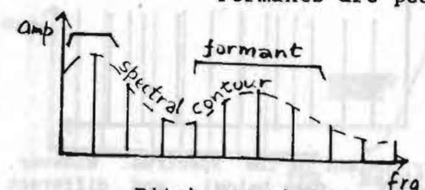


When the waveform begins at the start of the sound, the phase is zero. If it begins later the phase is greater, and has a maximum when the sound begins just before the end of the waveform.

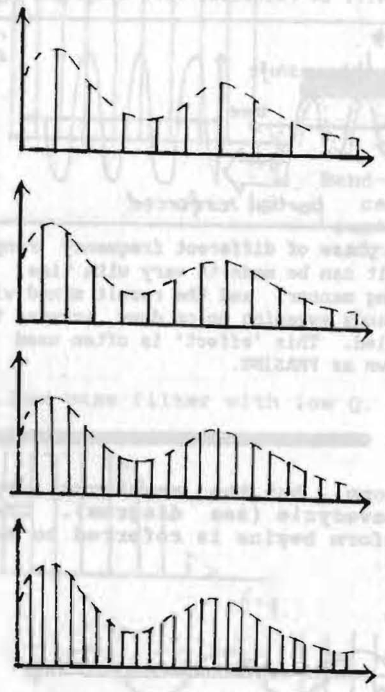
Alternatively, moving along a given waveform, the phase increases (see diagram 2). The rate at which the phase increases is equal to the frequency of the wave. This fact enables the Phase Vocoder to track the frequencies of partials in a sound.

FORMANTS

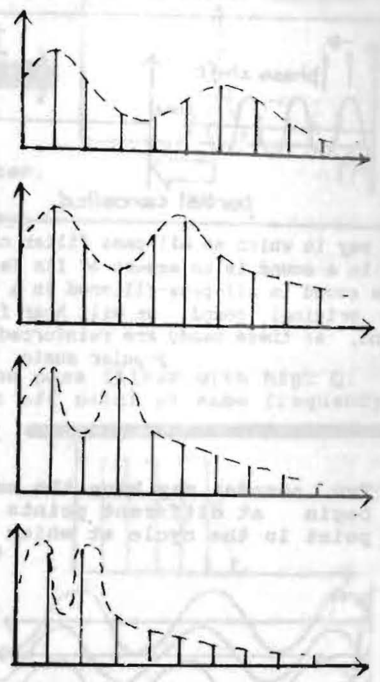
Formants are peaks in the spectral contour.



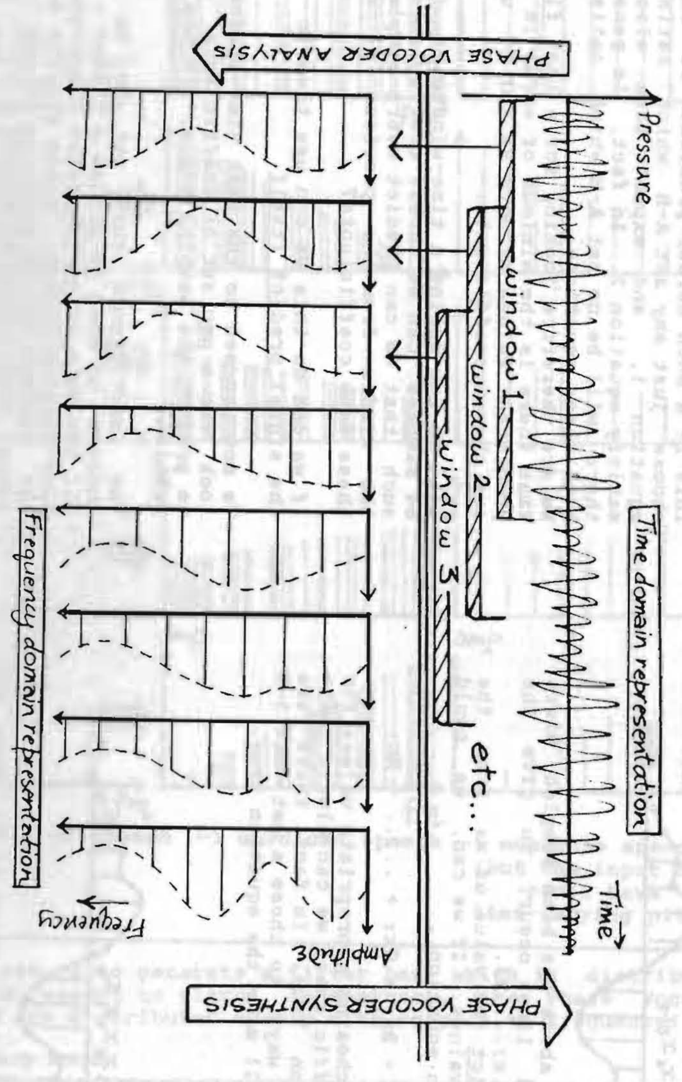
Pitch may change without changing the formants. e.g. sing a downward sliding pitch on a fixed vowel 'a'.

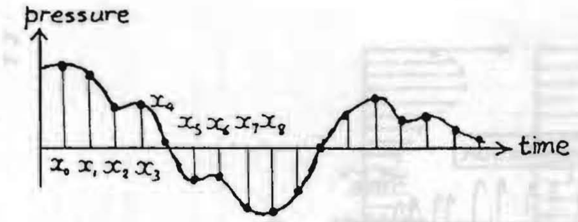


Formants may change without changing the pitch. e.g. sing 'a' -> 'u' on a fixed pitch.



PHASE VOCODER

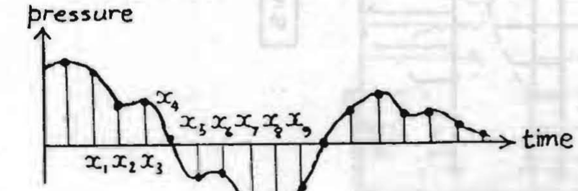




The waveform above has been sampled where the vertical lines occur, to give the values $x_0, x_1, x_2 \ldots x_8$.
Can we predict the value of x_8 from the previous 8 values. If we can, we could write down an equation...

$$(1)\ x_8 = Ax_0 + Bx_1 + Cx_2 + \ldots\ldots + Hx_7$$

Clearly, by chosing appropriate values for A-H (the coefficients) we can always make this equation true. In fact there are many, many ways to chose a set of A-H which will all make the equation true.



Can we predict the value of x_9 from the previous 8 values, using the same set of coeefficients A-H. If we can, we could write down an equation...

$$(2)\ x_9 = Ax_1 + Bx_2 + Cx_3 + \ldots\ldots + Hx_8$$

This is a much harder problem. We cannot choose just any set A-H which satisfies equation 1, and expect them also to satisfy equation 2. In fact, in general, there will be no set A-H which satisfies both these equations.
We are therefore looking for a best fit so that there is the minimum of error in the predicted values of both x_8 and x_9 using our chosen coefficients.

Let us now define a time-window of (say) 64 samples. Can we choose coefficients A-H such that we can predict every sample in the window from the previous 8, using these same coefficients?

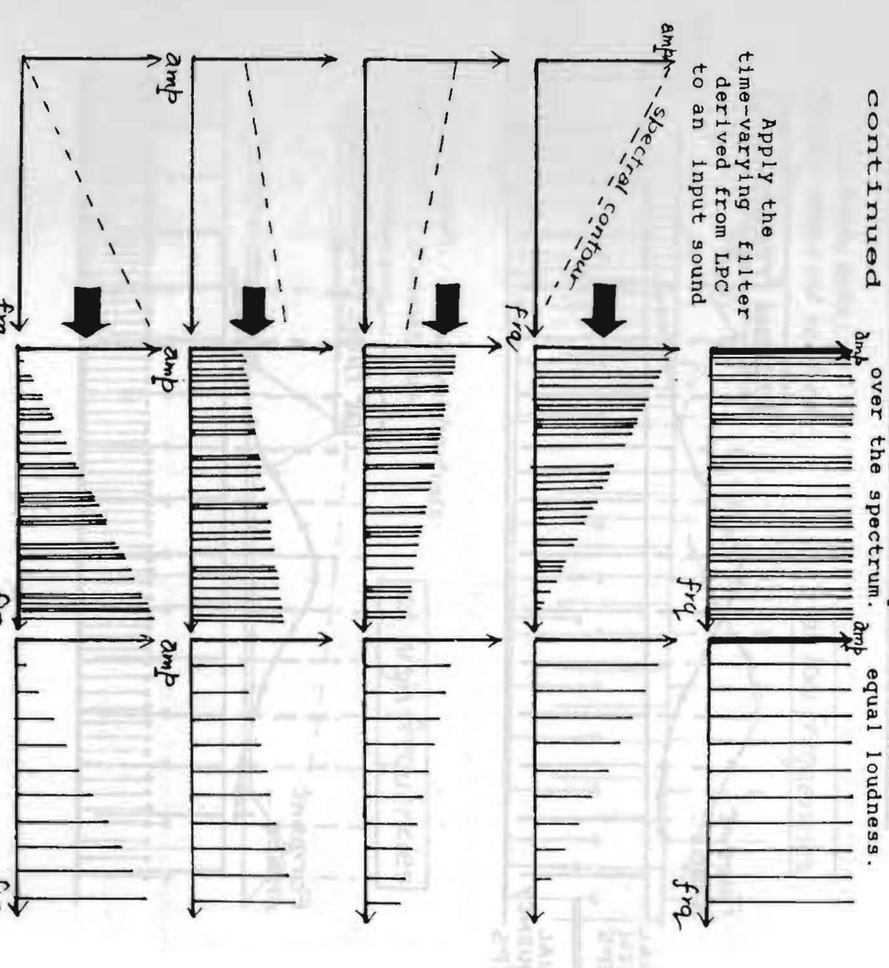
If we can do this we can use them to make the signal predict itself.

We now proceed to the next time-window and look for a new set of coefficients (I-P) to predict the samples there, in the same fashion.

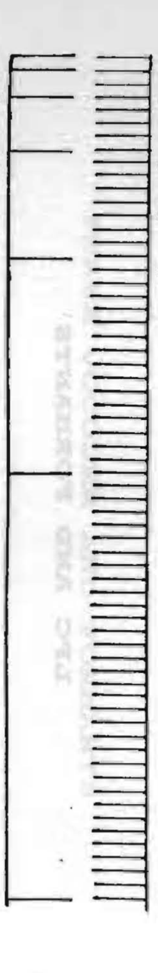
The coefficients turn out to be the numbers we need to define a filter - and all the sets of coefficients together, a time-varying filter. These filters define the contour of the spectrum of our input sound, from moment to moment.

In practice we need about 40 coefficients (predicting each sample in our window from the previous 40 samples) to achieve a reasonable resolution of the spectrum.

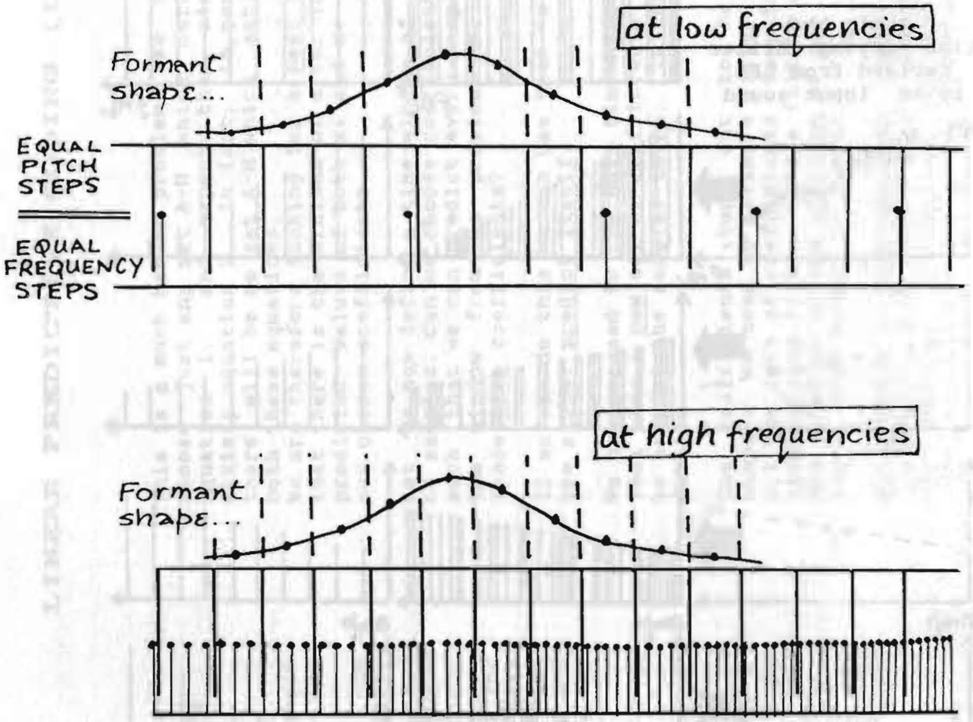
LINEAR PREDICTIVE CODING continued



LPC can be set up to generate a filter bank which is distributed evenly with respect to pitch. In contrast, the Phase Vocoder channels are distributed evenly with respect to frequency.



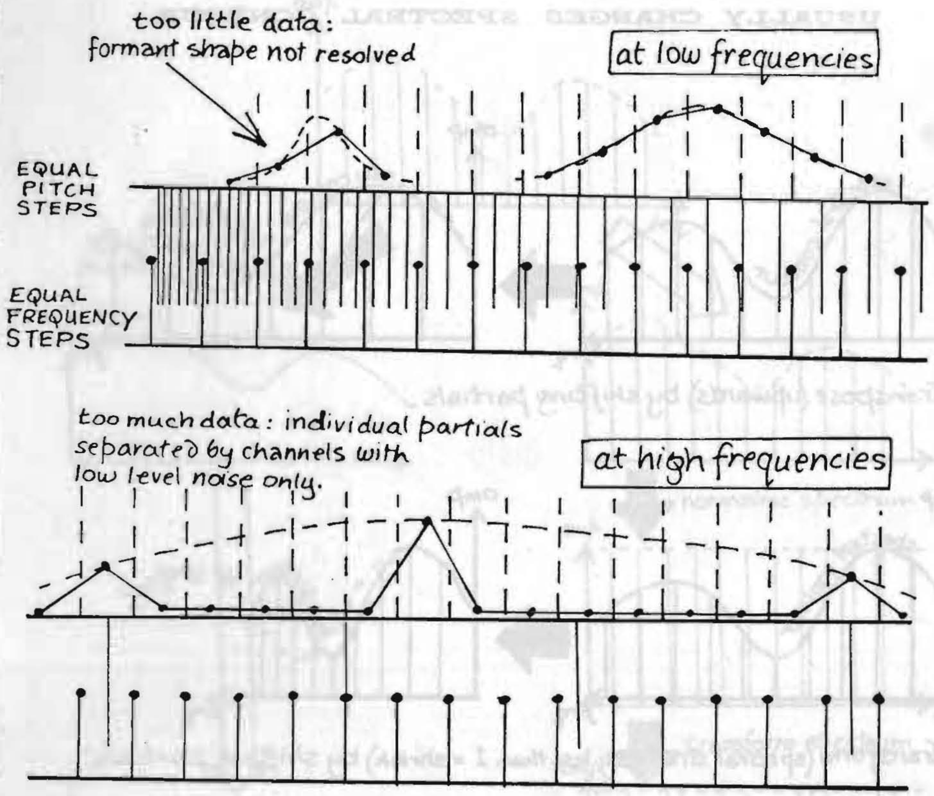
Distribution of Formants relative to Pitch



Formants have similar pitch-width. If analysis channels are equally spaced in pitch (as they may be, in Linear Predictive Coding) then formant envelopes are resolved equally well in all ranges.

LPC AND FORMANTS

Distribution of Formants relative to Frequency

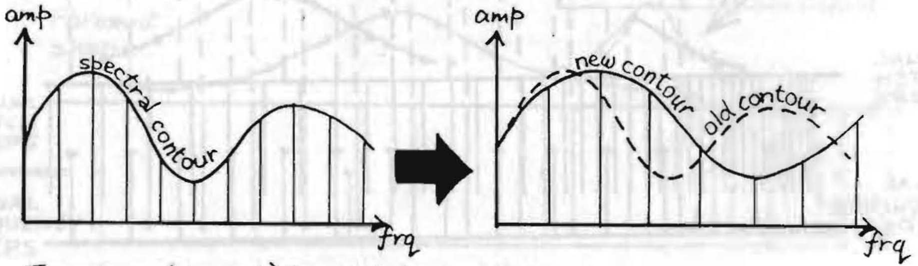


Formants have similar pitch-width. Hence their frequency-width increases as we go up the spectrum. The phase vocoder channels are equally spaced in frequency so do not resolve the formant envelopes well in all ranges.

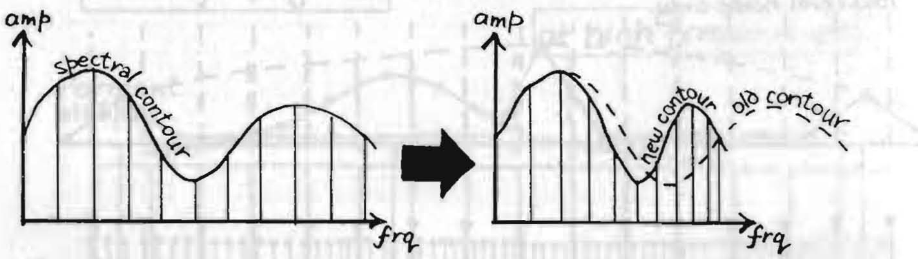
PHASE VOCODER AND FORMANTS

CHANGING THE SPECTRUM

USUALLY CHANGES SPECTRAL CONTOUR



Transpose (upwards) by shifting partials.

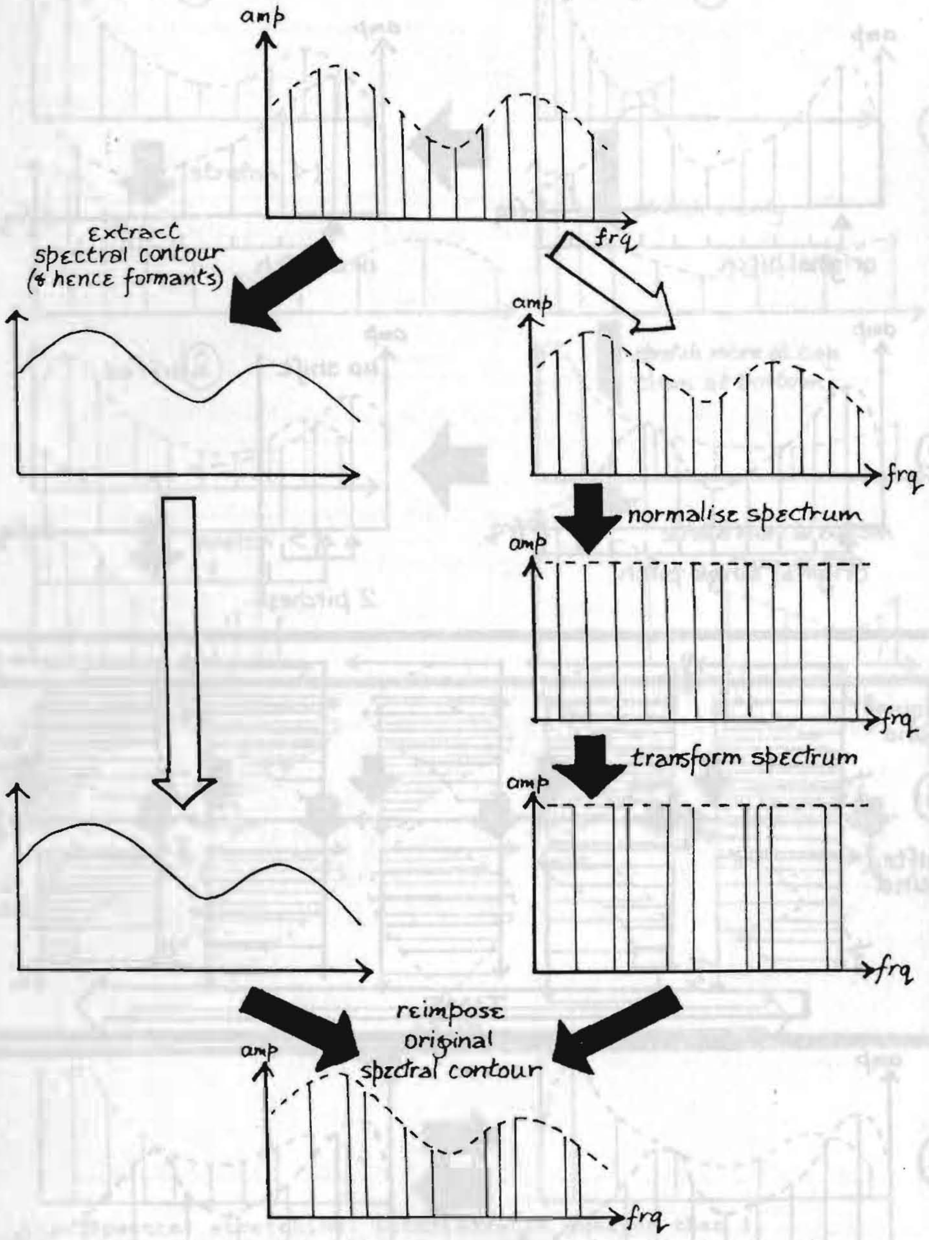


Transform (spectral stretch by less than 1 = shrink) by shifting partials.

Pitch transposition & spectral transformation do not normally preserve the spectral contour.

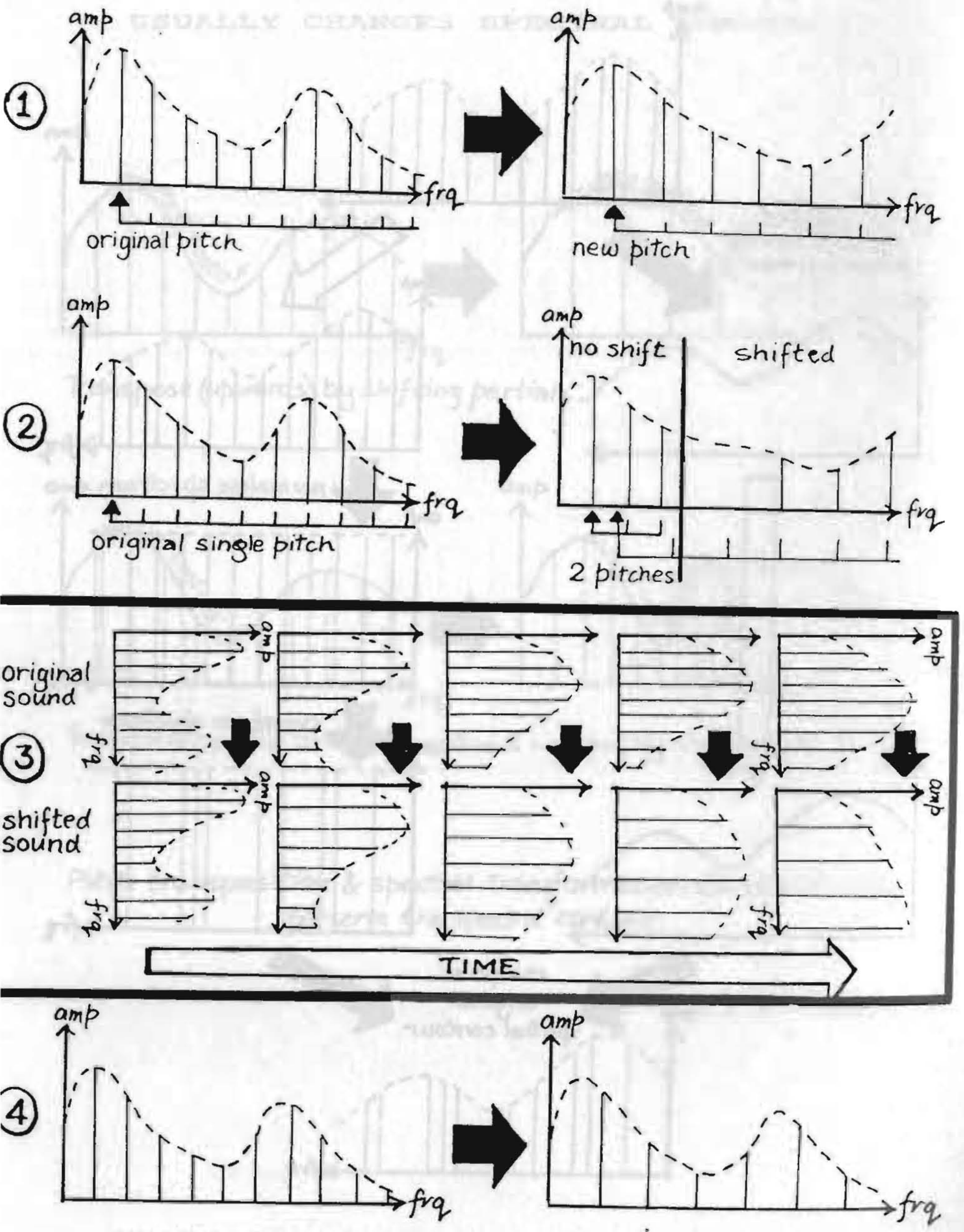
FORMANT PRESERVING SPECTRAL MANIPULATION

for every window in the frequency domain.....



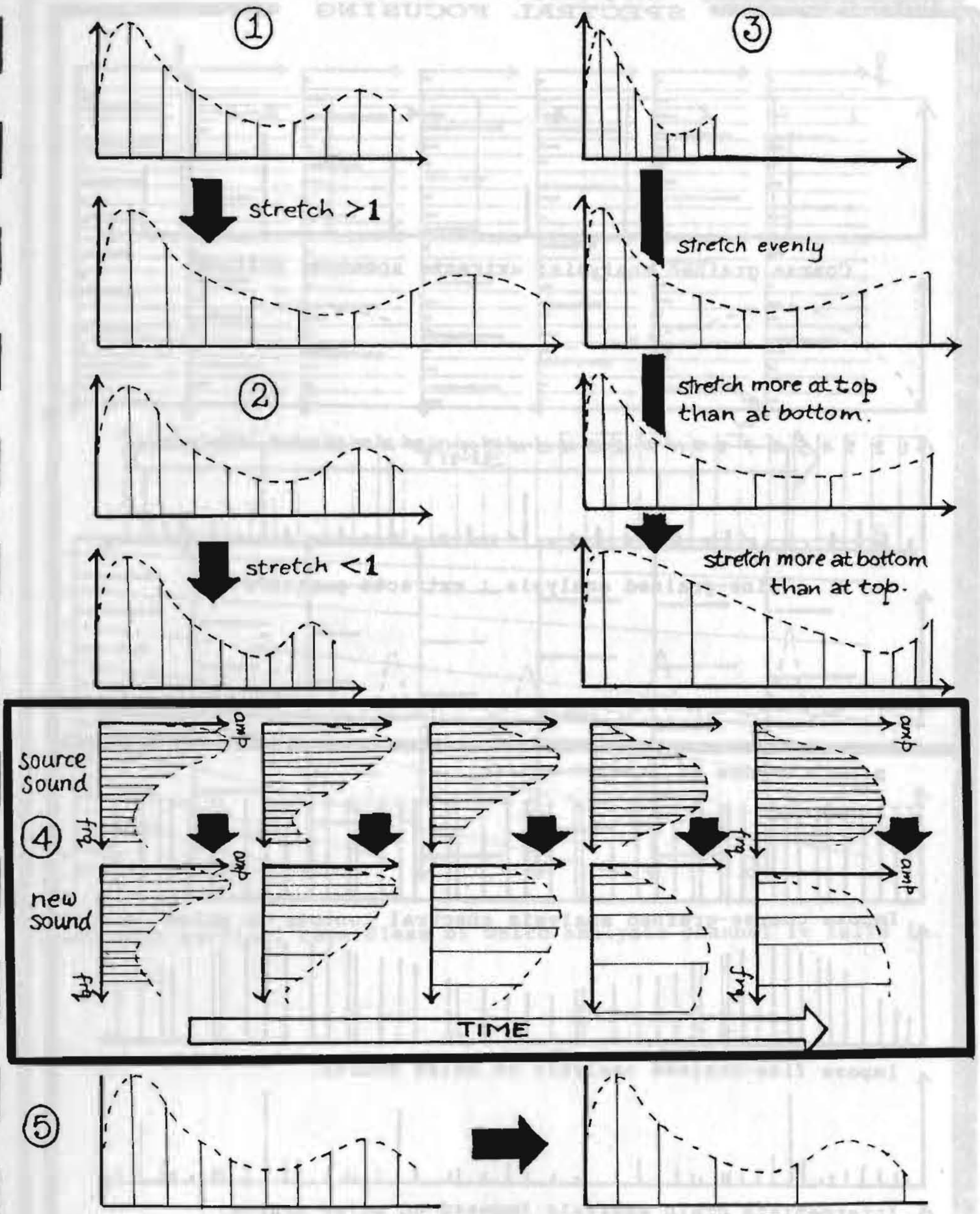
FORMANT PRESERVING PITCH SHIFTING
works similarly

SPECTRAL SHIFTING



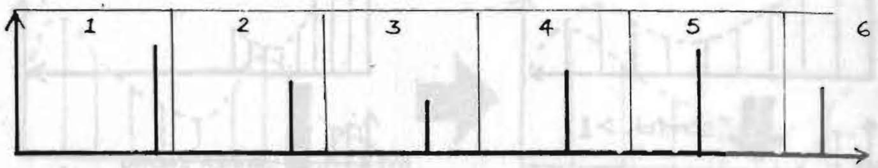
(1) Spectral shift.
(2) Spectral shift of part of spectrum only.
(3) Time-varying (increasing) spectral shift.
(4) Formant-preserving spectral shift.

SPECTRAL STRETCHING

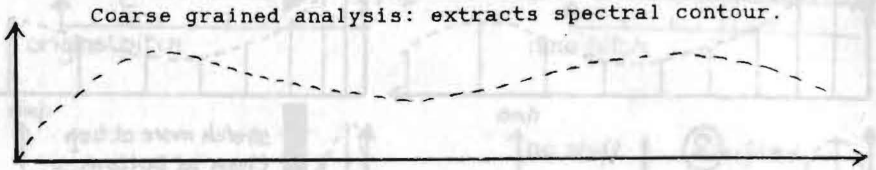


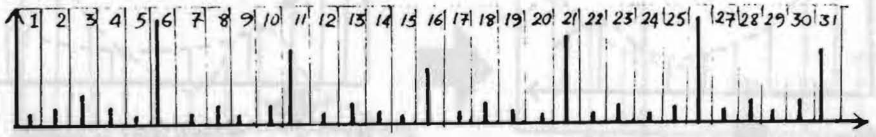
(1) Spectral stretching: total stretch greater than 1.
(2) Total stretch less than 1 (spectral shrinking).
(3) Different types of stretching.
(4) Time-variable (total stretch increasing) spectral stretching.
(5) Formant preserving spectral stretching.

SPECTRAL FOCUSING

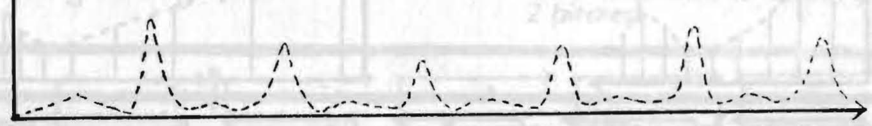


Coarse grained analysis: extracts spectral contour.

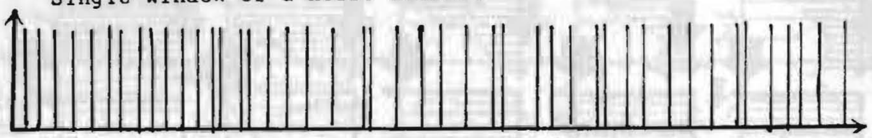




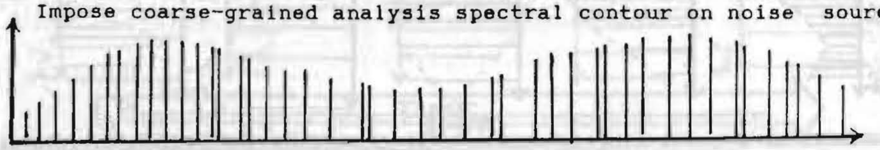
Fine-grained analysis : extracts partials.



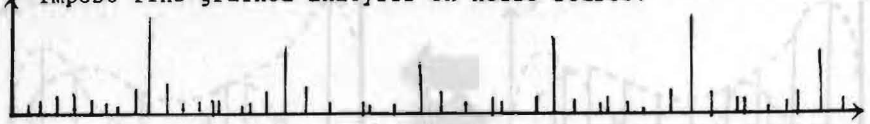
Single window of a noise source.



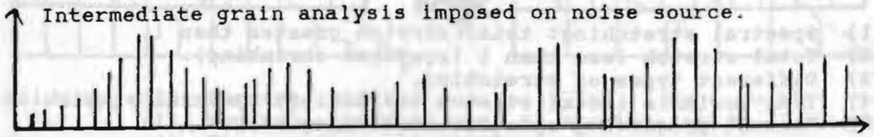
Impose coarse-grained analysis spectral contour on noise source



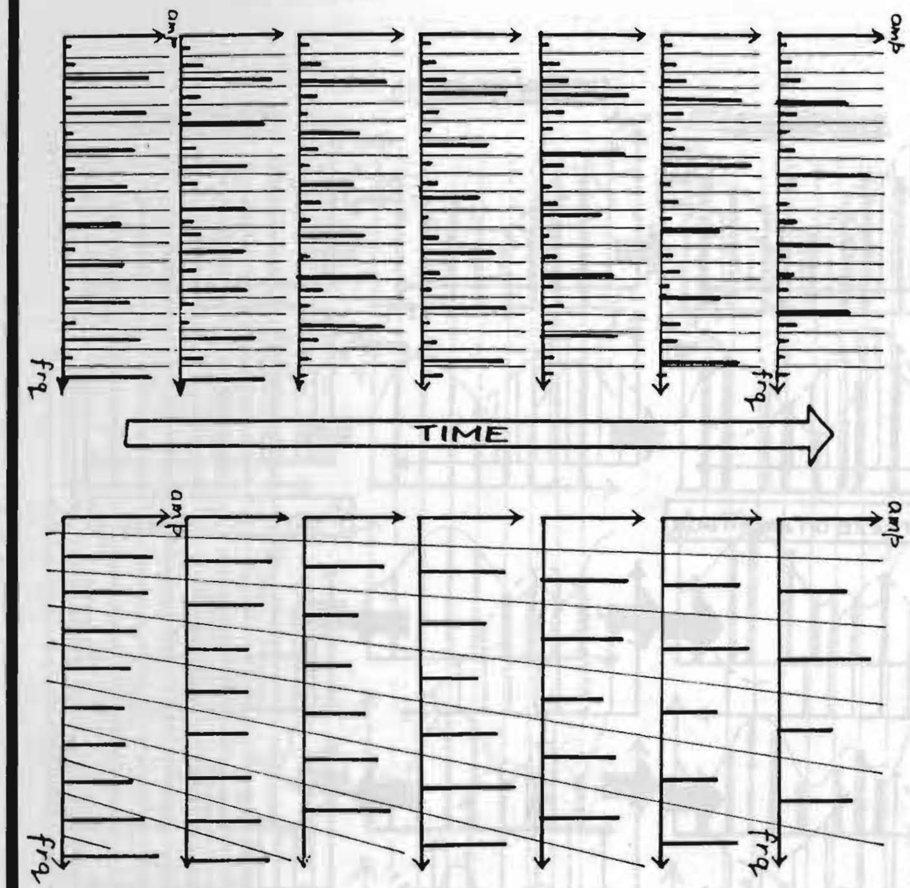
Impose fine-grained analysis on noise source.



Intermediate grain analysis imposed on noise source.

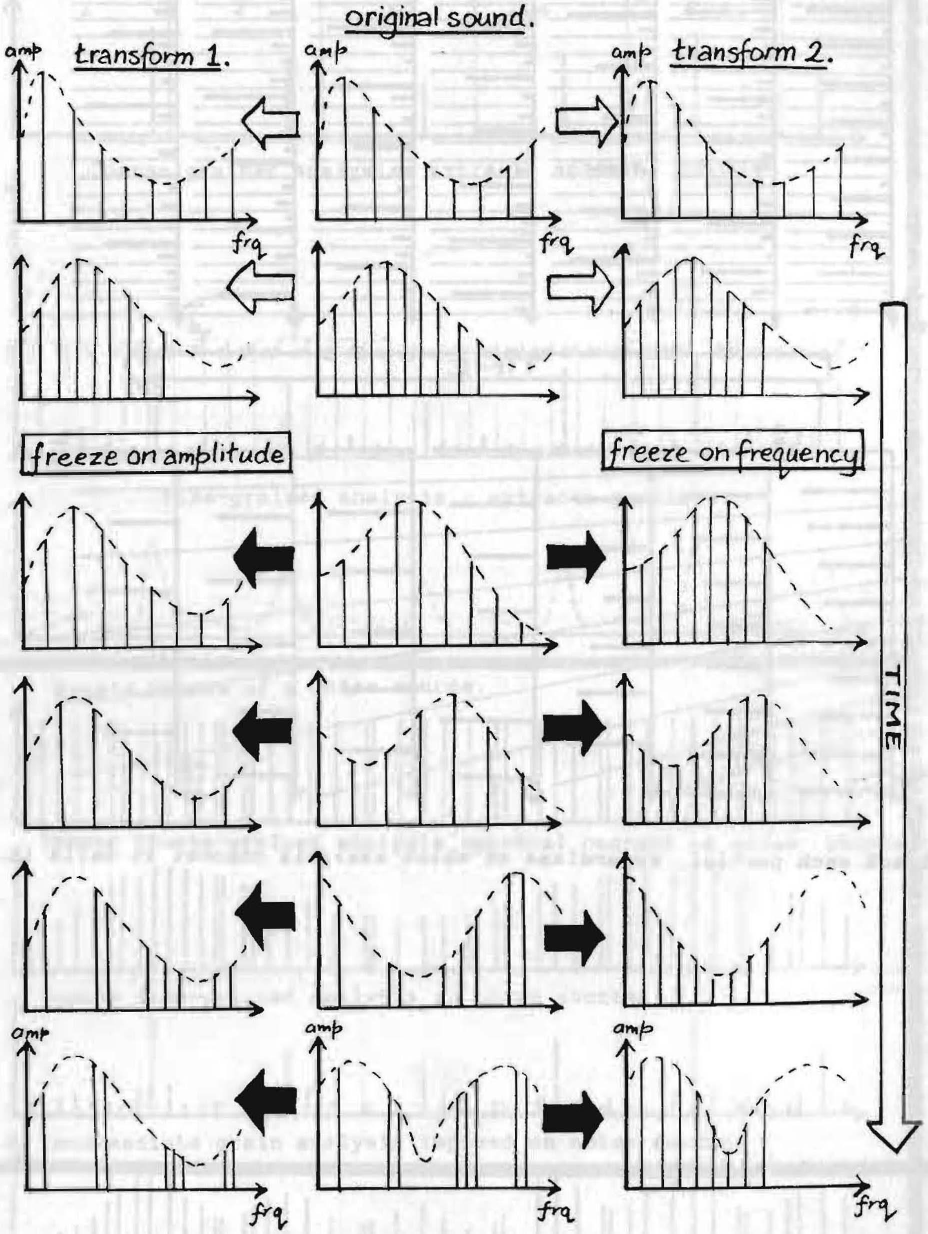


PARTIAL TRACKING

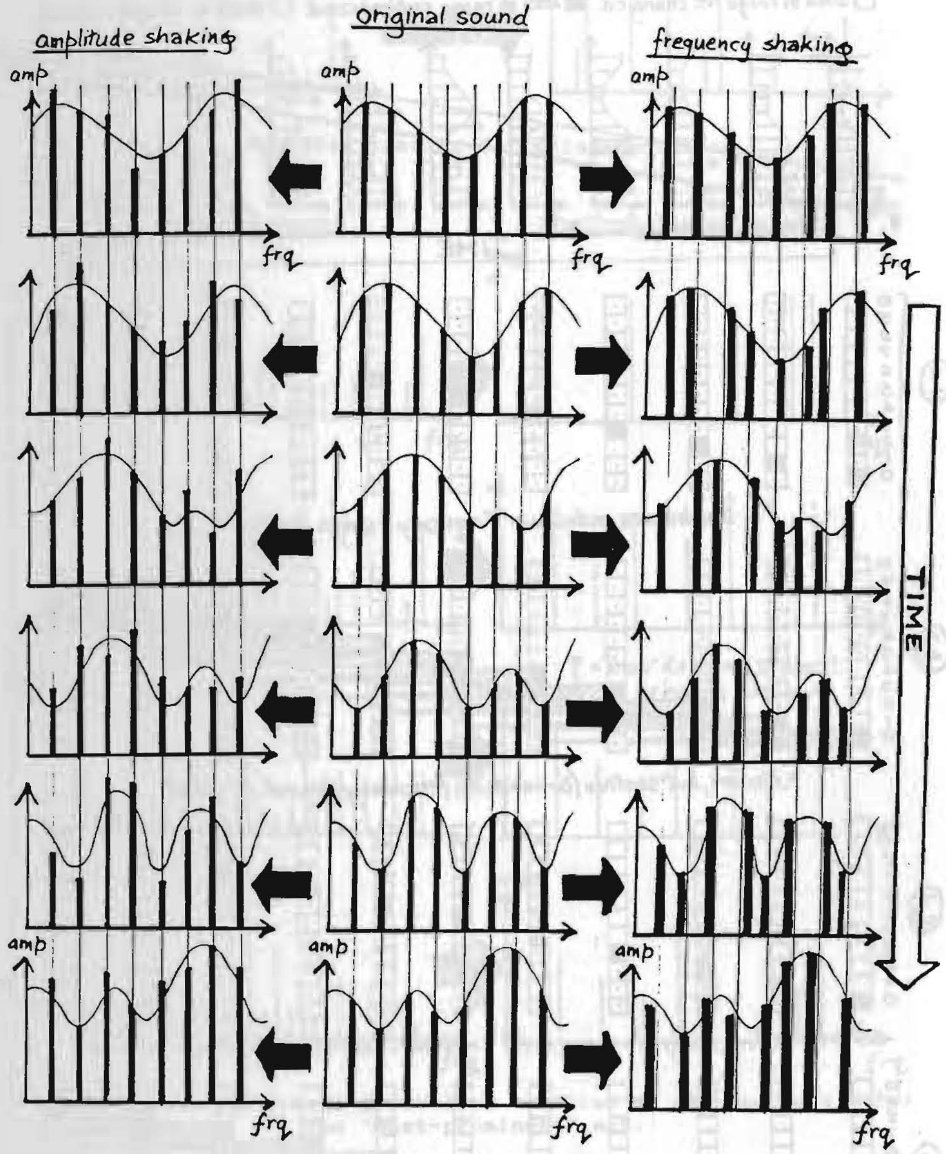


Track each partial, regardless of which analysis channel it falls in.

SPECTRAL FREEZING



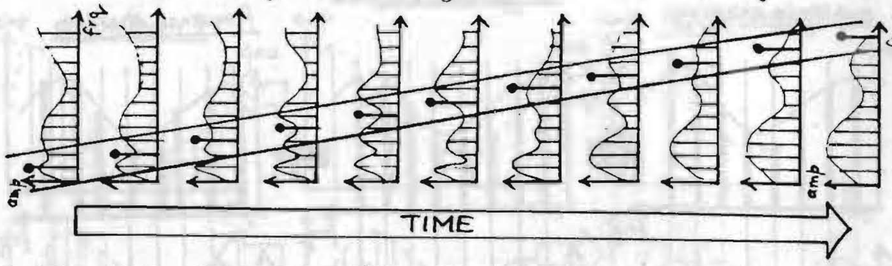
SPECTRAL SHAKING

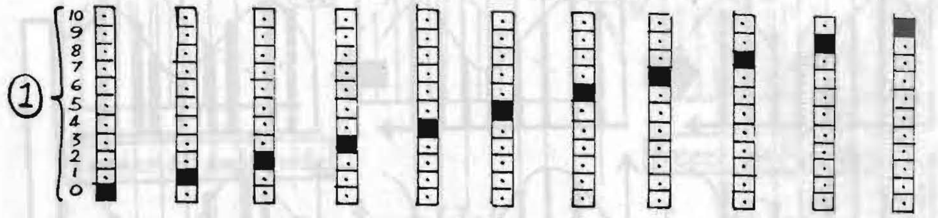


SPECTRAL ARPEGGIATION

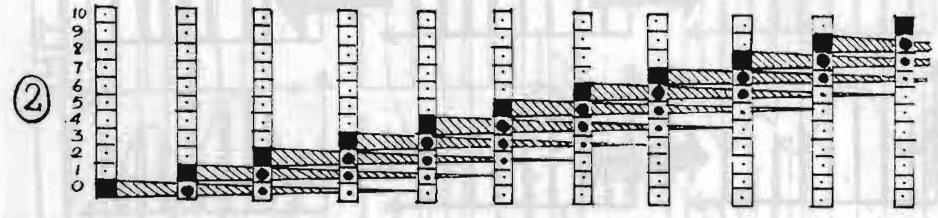
Emphasize different elements of spectrum in successive time-windows. Emphasis-by-loudness may sweep over phase vocoder channels or composer-defined frequency ranges. The sweeping motion may be in any composer-specified shape (e.g. sinewave, ramp etc.) and at any (possibly time-varying) speed.

Data in range not changed. Data in range emphasized. Data in range zeroed.

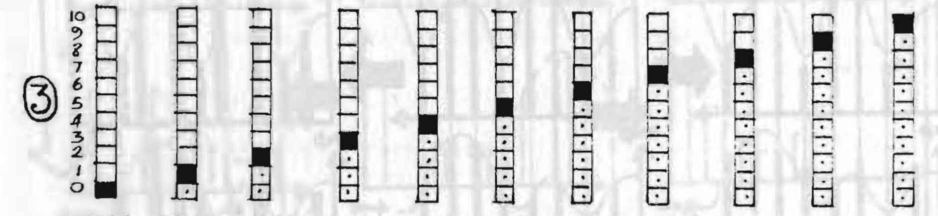




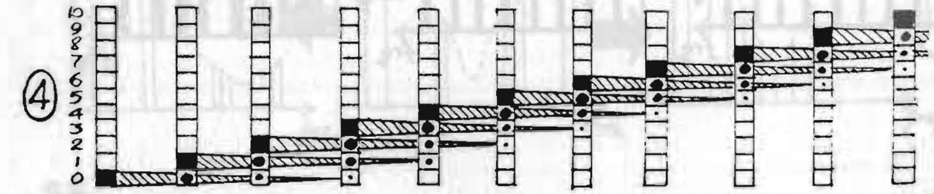
Emphasize individual frequency ranges, in turn.



As above, but sustain (diminuendo) frequency found at first.

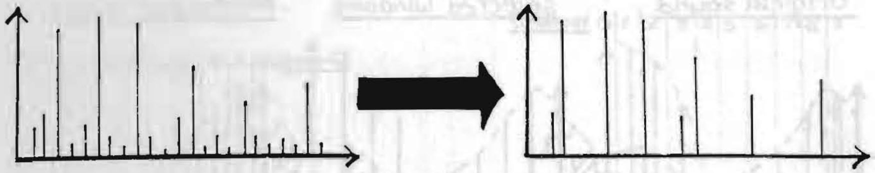


Set level in frq. ranges to zero, until emphasis arrives. Spectrum 'unfolds'.

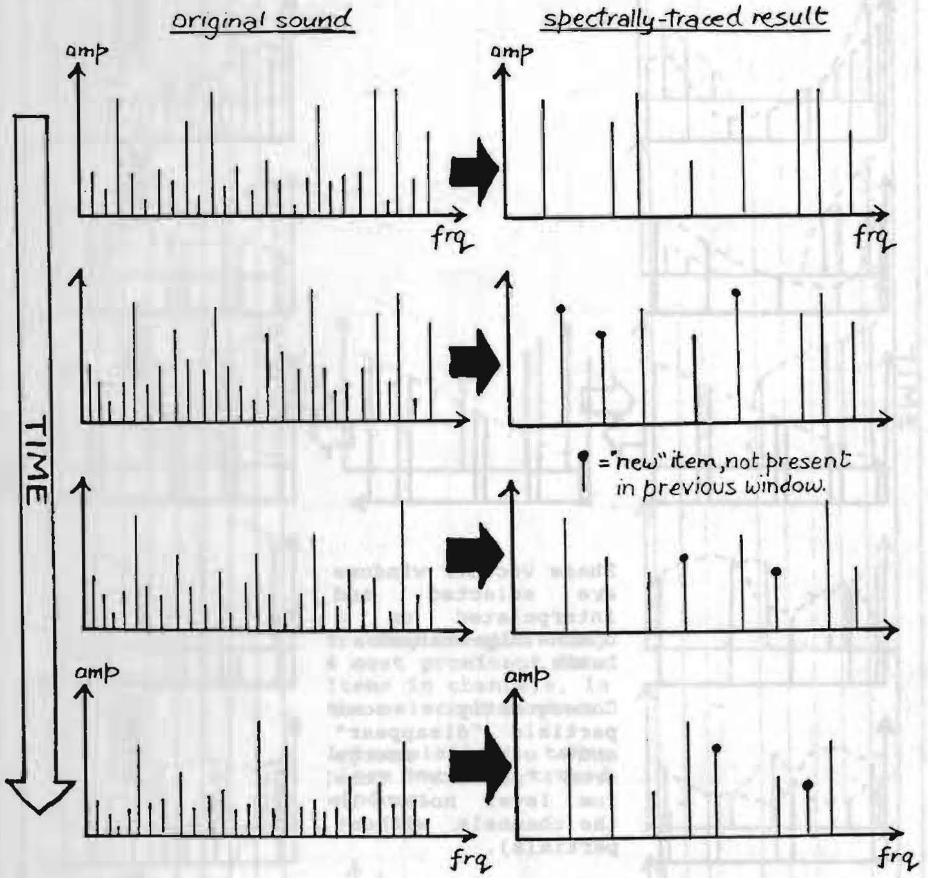


Turn off frq. range before & after emphasis (in this example, also sustain, as 2).

SPECTRAL TRACING



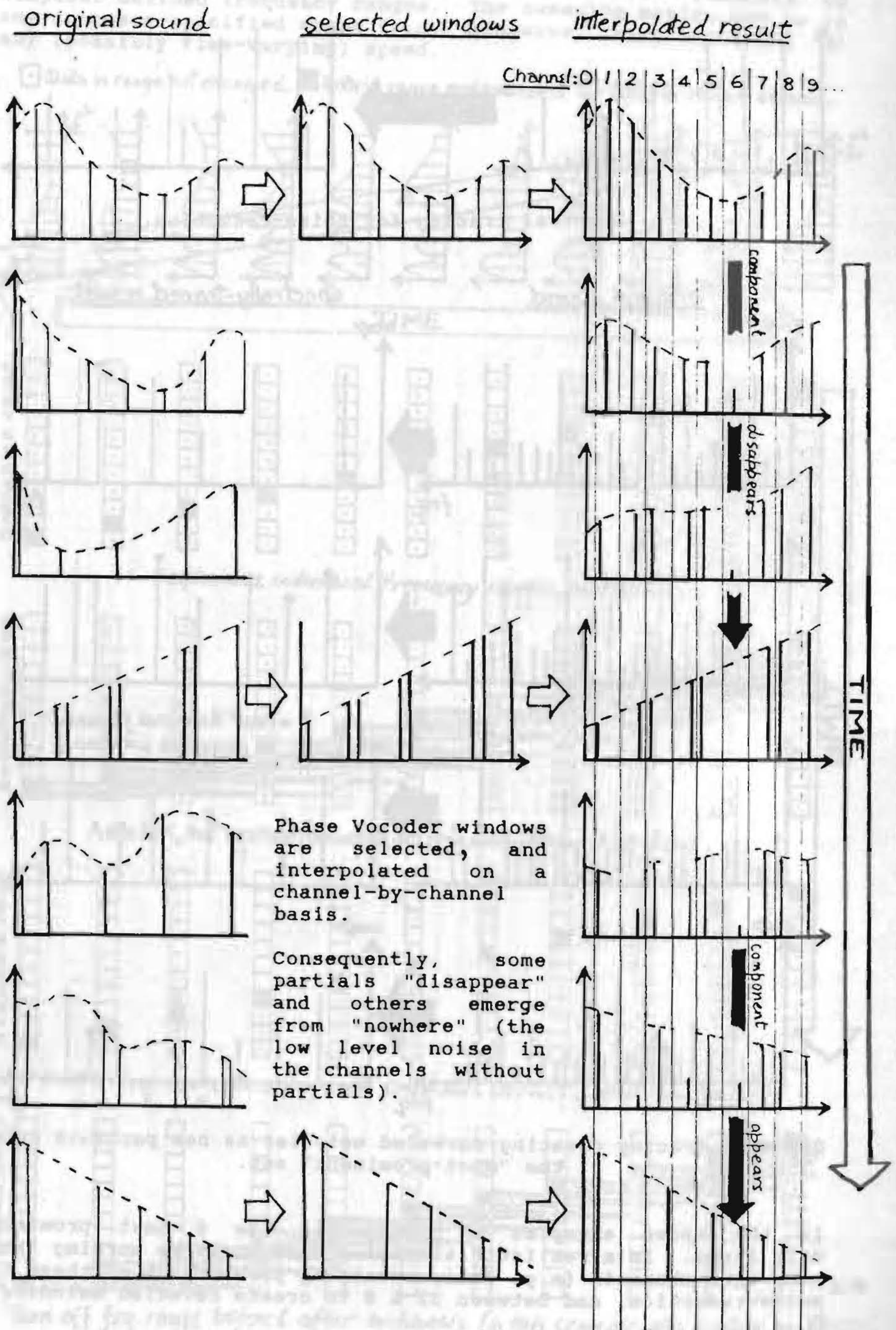
Spectral tracing for noise-reduction.



Spectral tracing creating revealed melodies as new partials enter the "most-prominent" set.

In the above examples we retain only the 8 most prominent data items. In a realistic situation, we would be working with many more channels (e.g. 512), retaining perhaps 128 of these for noise-reduction, and between 32 & 8 to create revealed melodies.

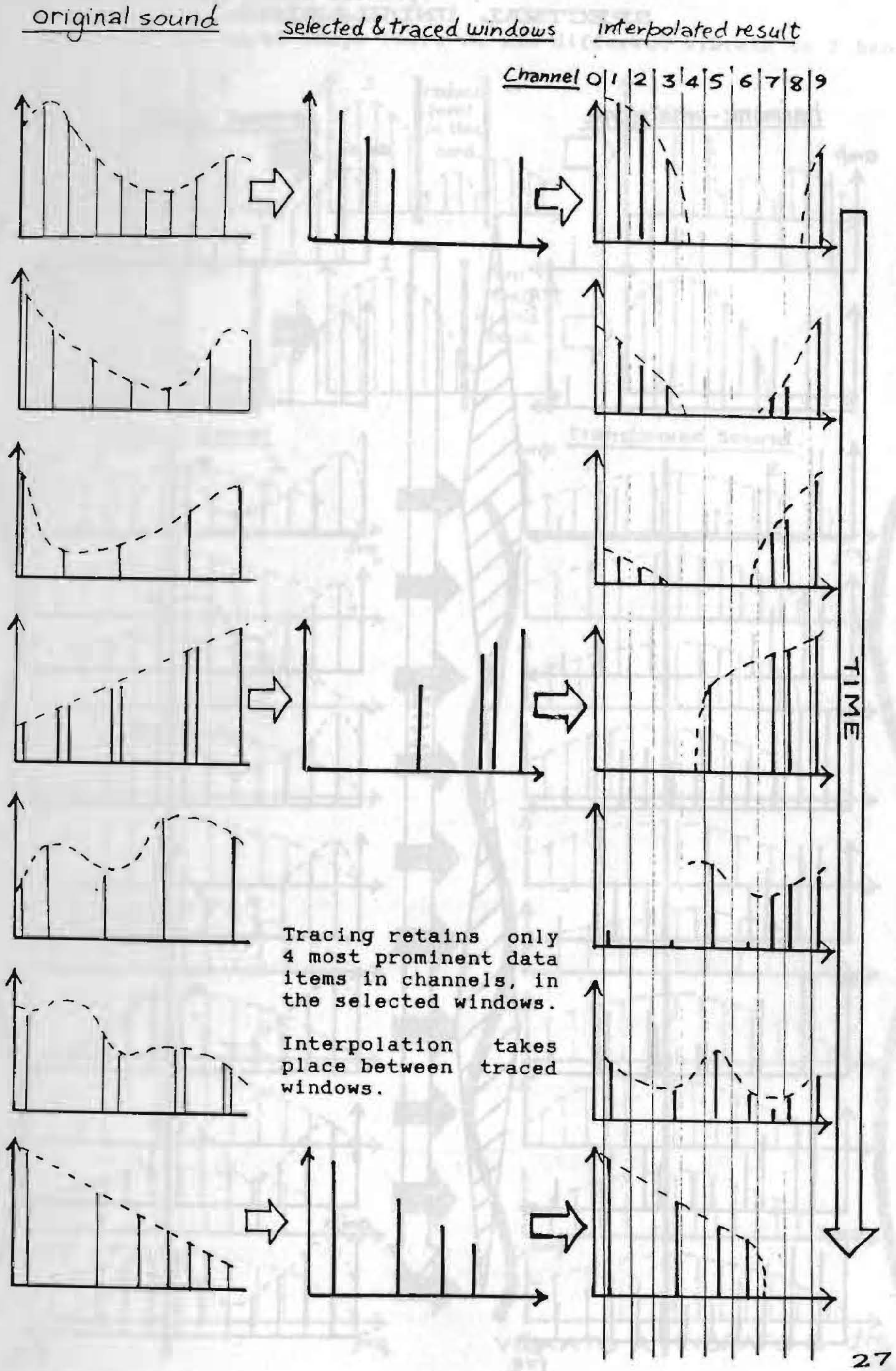
SPECTRAL BLURRING



Phase Vocoder windows are selected, and interpolated on a channel-by-channel basis.

Consequently, some partials "disappear" and others emerge from "nowhere" (the low level noise in the channels without partials).

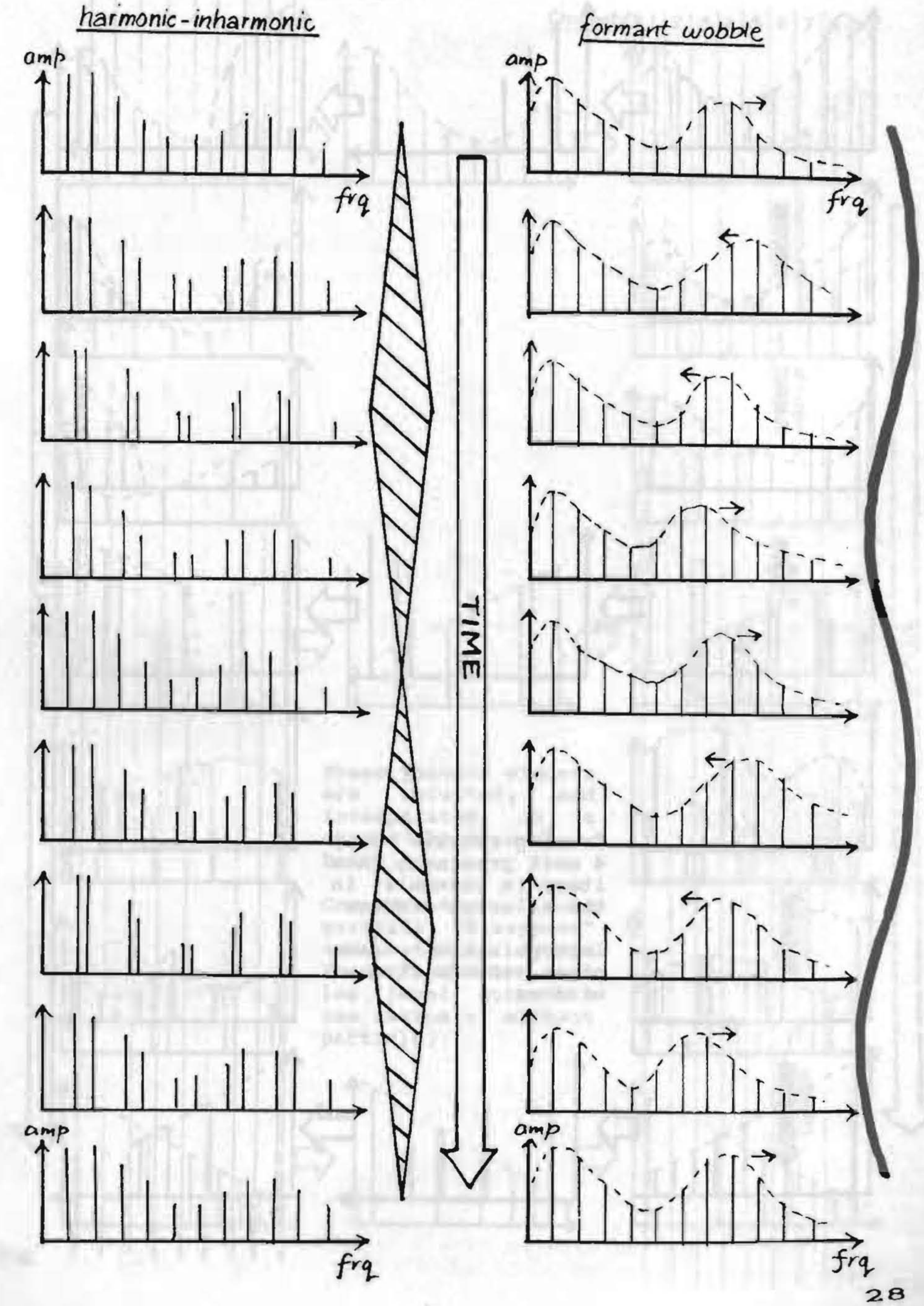
SPECTRAL TRACE AND BLUR



Tracing retains only 4 most prominent data items in channels, in the selected windows.

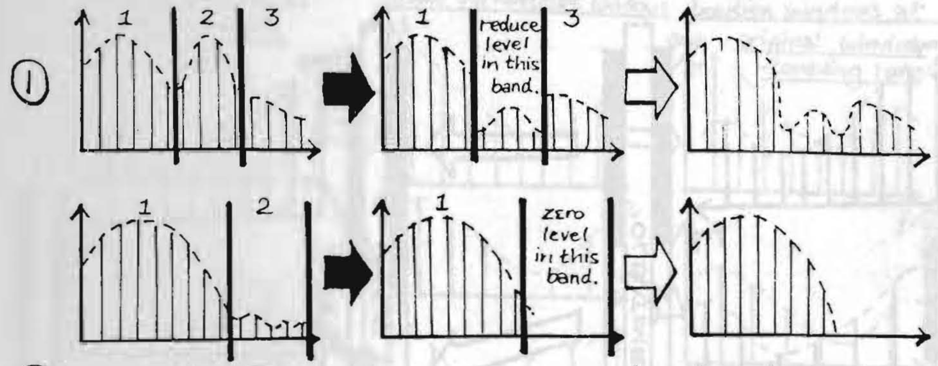
Interpolation takes place between traced windows.

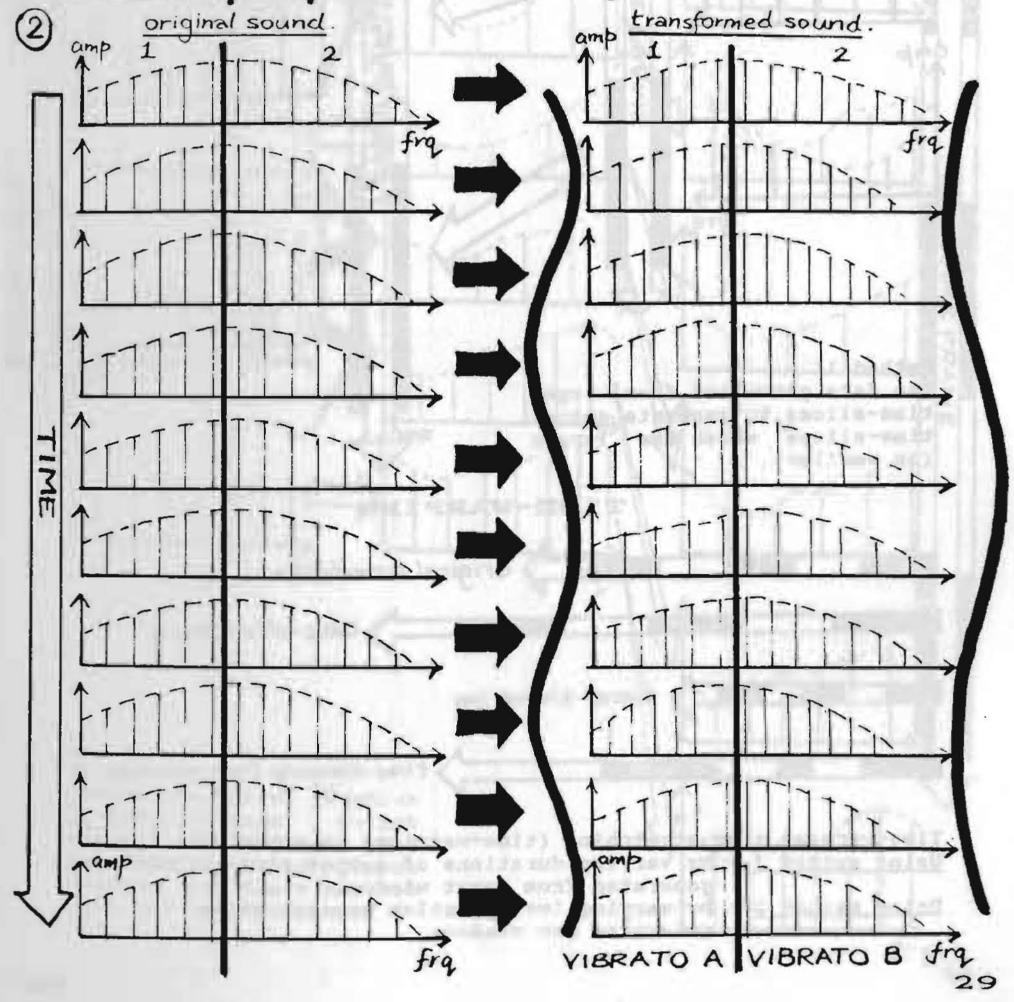
SPECTRAL UNDULATION



SPECTRAL SPLITTING

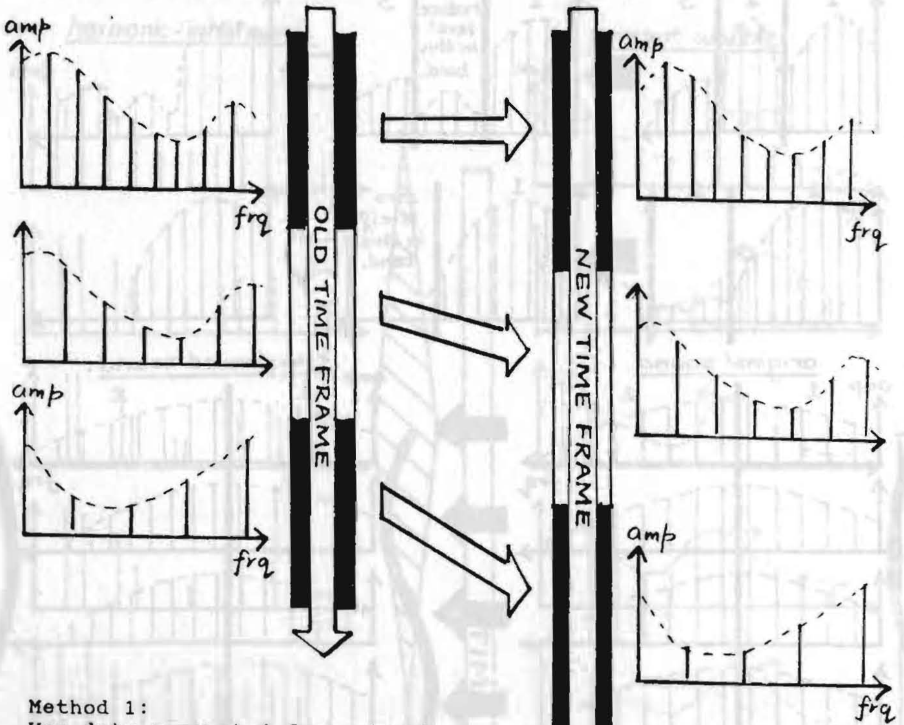
Divide spectrum into bands for e.g.
(1) Filtering, or noise removal.
(2) Splitting the aural image (here we add different vibrato to 2 bands).





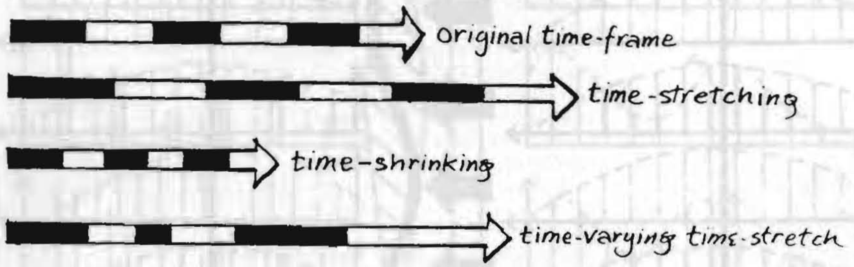
SPECTRAL TIME-STRETCHING

Method 1



Method 1:
Use data generated from input time-slices, to generate output time-slices which are larger (or smaller).

TIME-WARPING



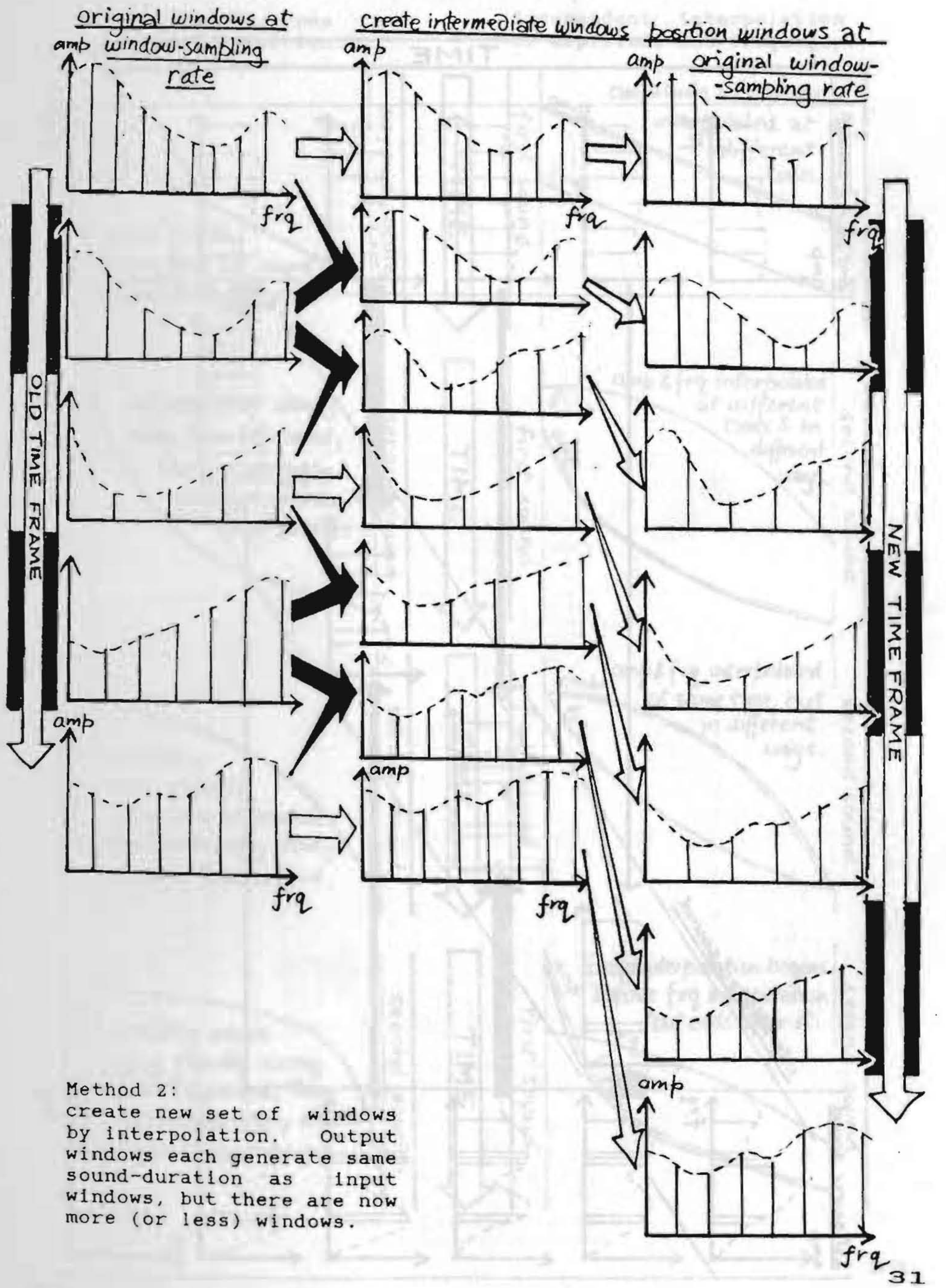
Time-varying time-stretching (time-warping) is achieved ...

<u>Using method 1</u> : by varying durations of output time-slices generated from input windows.

<u>Using method 2</u> : by varying interpolation process which generates new windows.

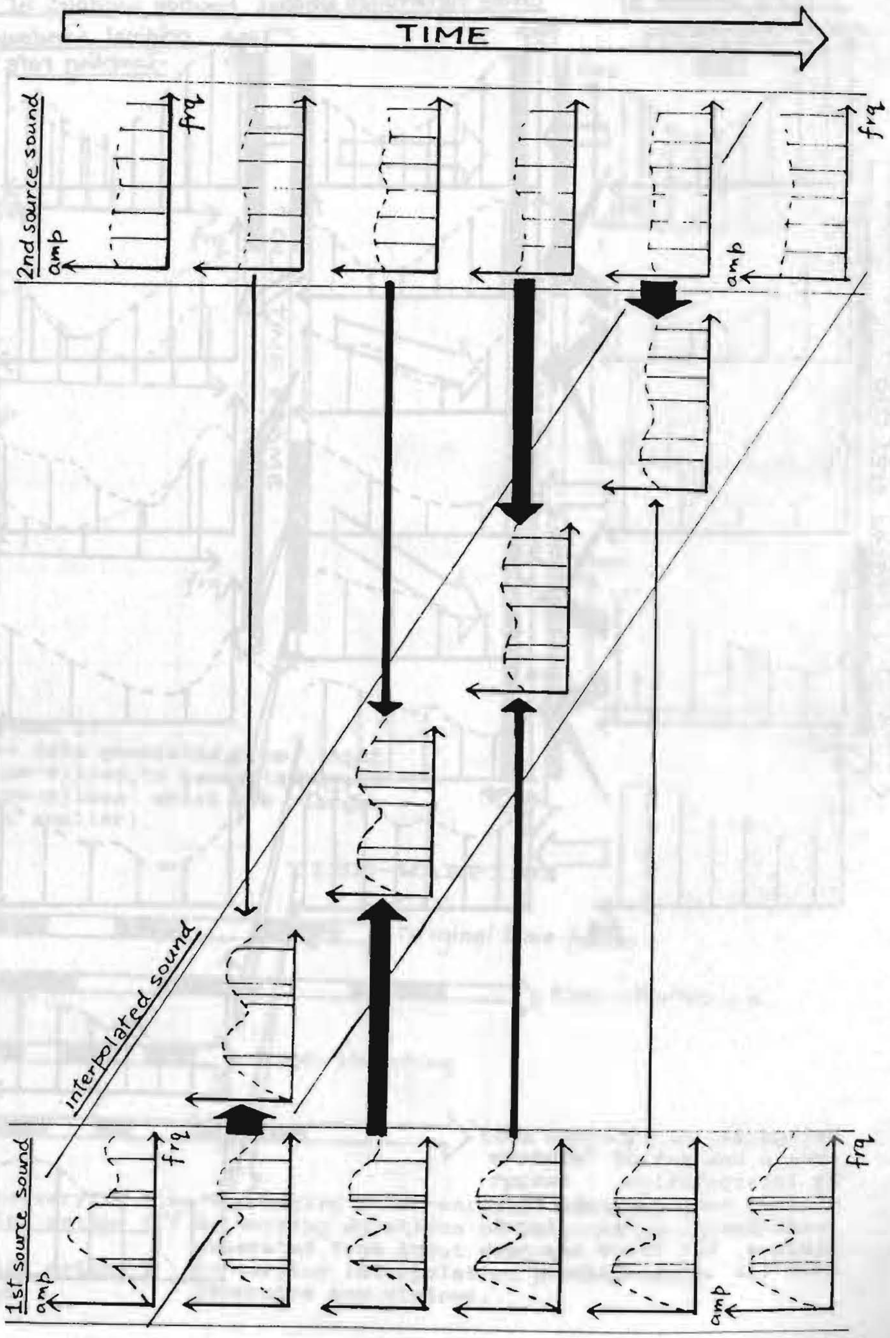
SPECTRAL TIME-STRETCHING

Method 2



Method 2:
create new set of windows by interpolation. Output windows each generate same sound-duration as input windows, but there are now more (or less) windows.

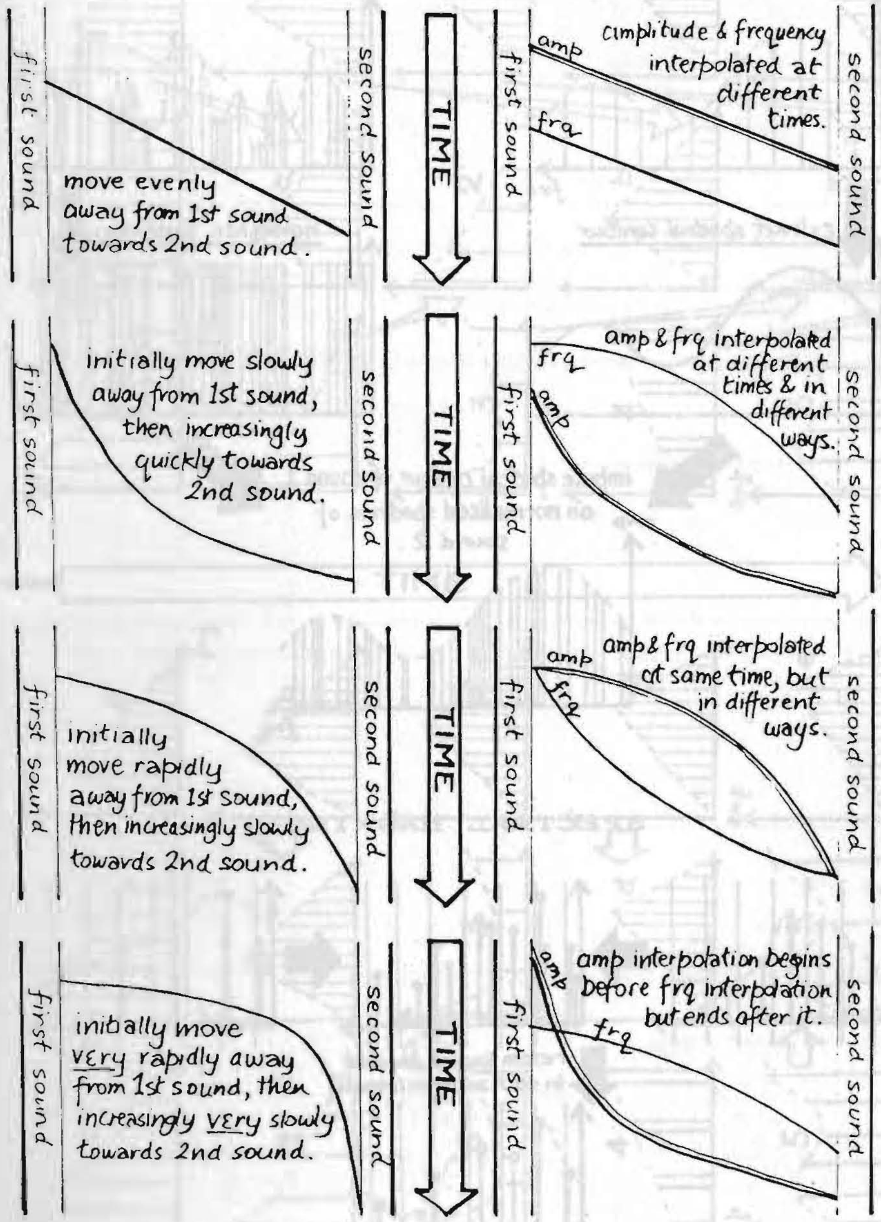
SPECTRAL INTERPOLATION



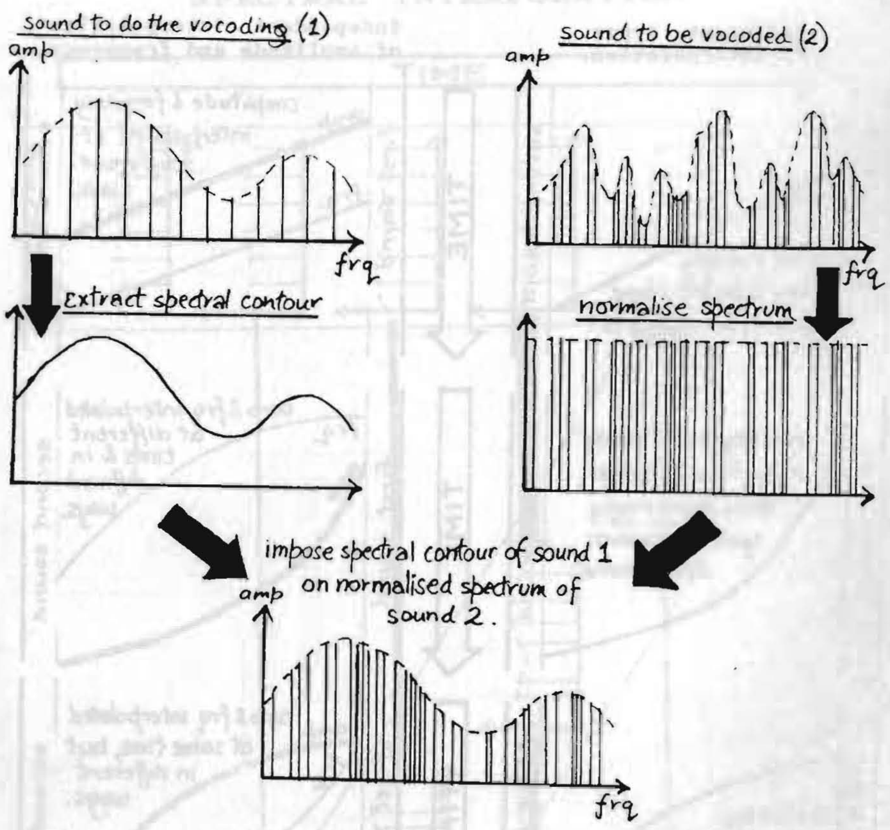
TYPES OF SPECTRAL INTERPOLATION

Different types of interpolation.

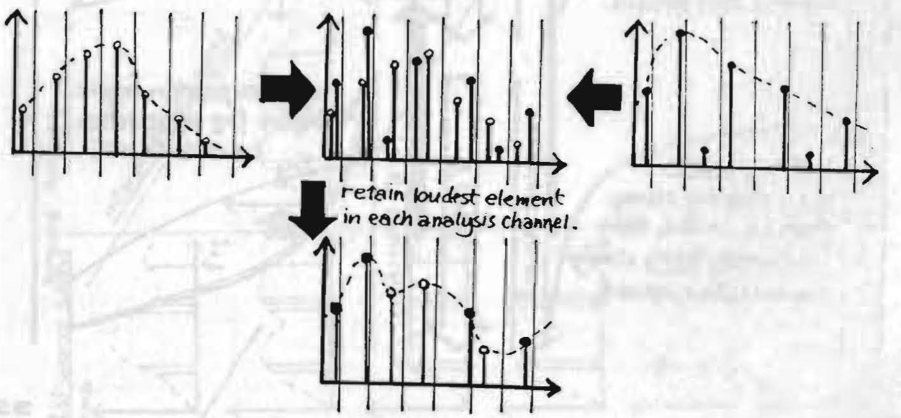
Independent interpolation of amplitude and frequency.



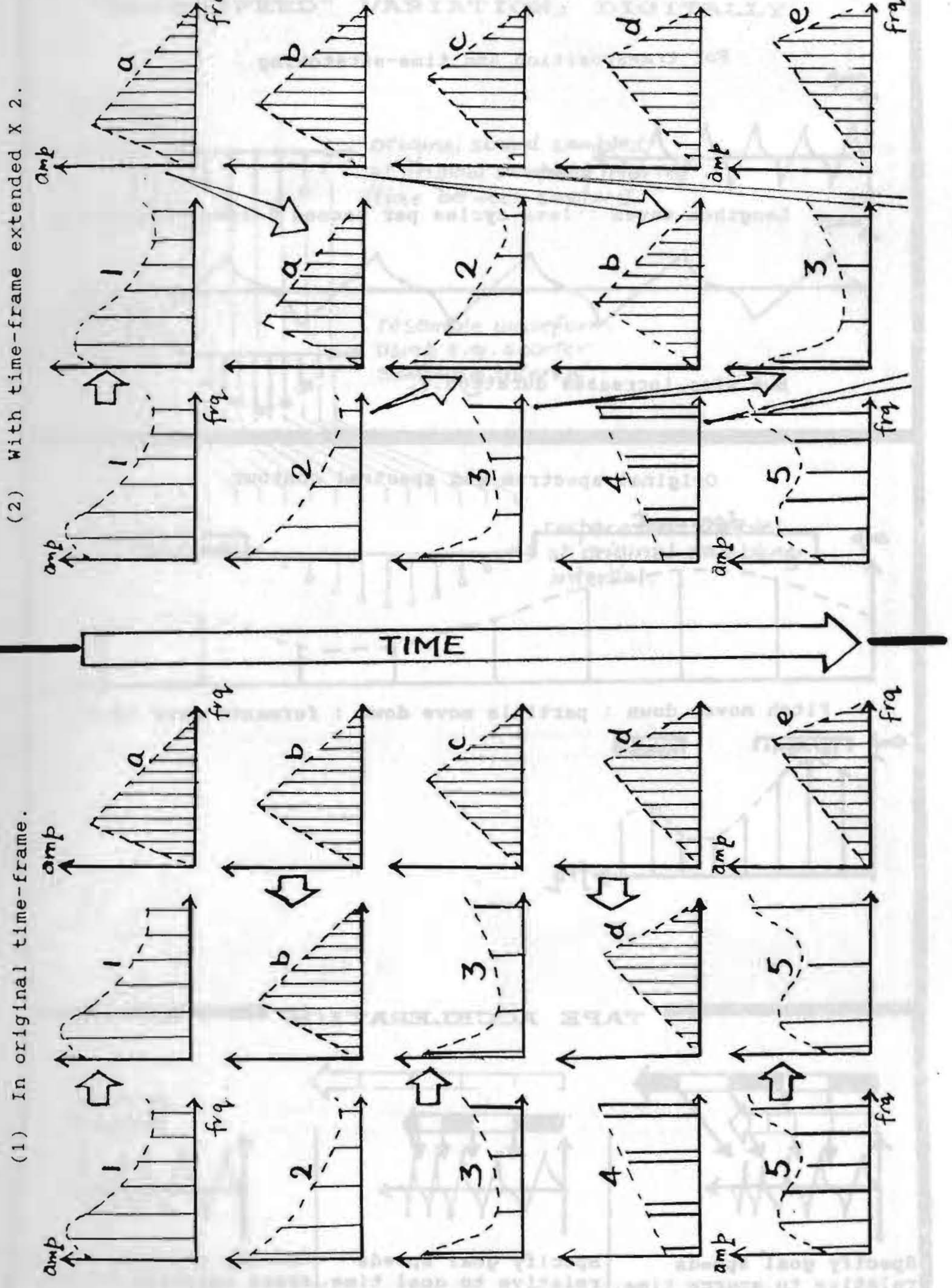
VOCODING



SPECTRAL MASKING

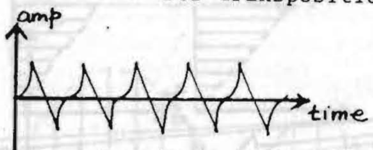


SPECTRAL INTERLEAVING

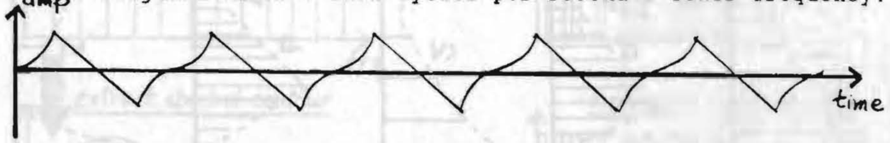


TAPE-SPEED VARIATION

For transposition and time-stretching.

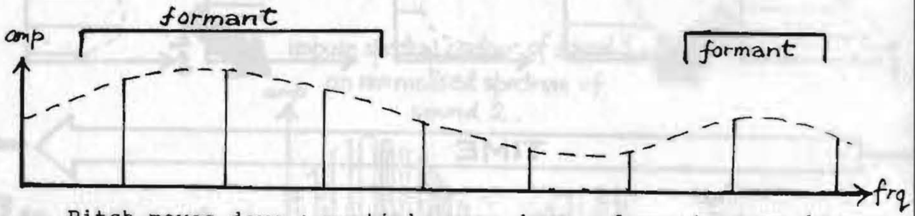


Lengthen waves : less cycles per second : lower frequency.

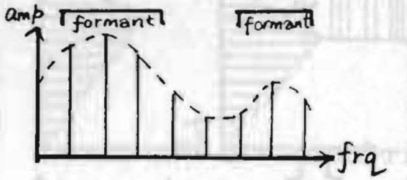


but also increases duration...

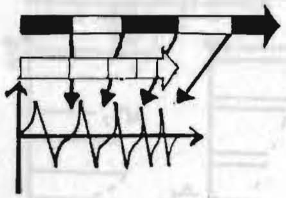
Original spectrum and spectral contour.



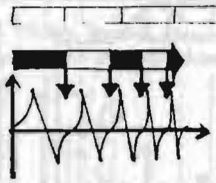
Pitch moves down : partials move down : formants move down.



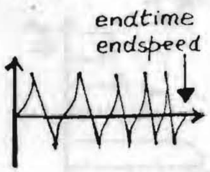
TAPE ACCELERATION



Specify goal speeds relative to source time.

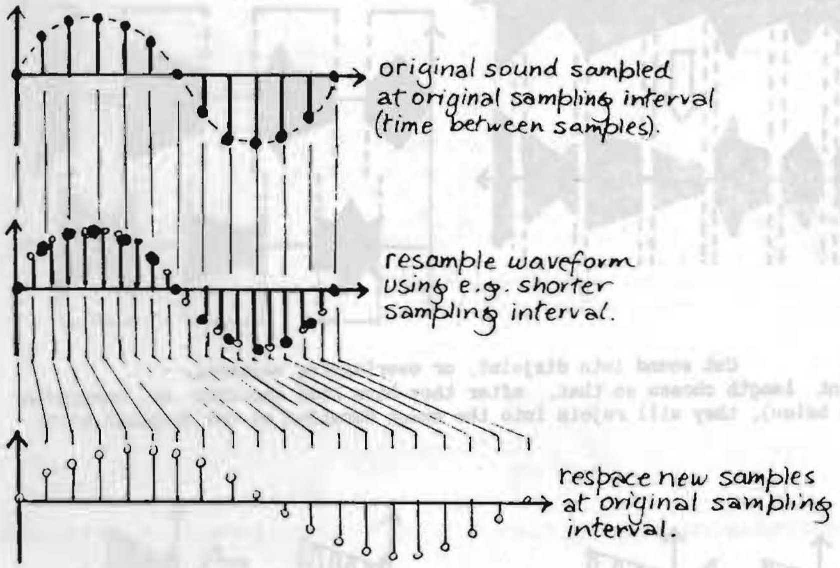


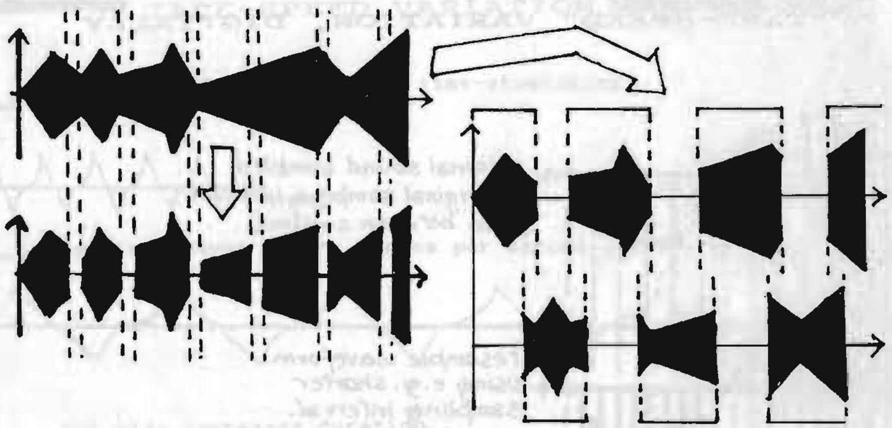
Specify goal speeds relative to goal time.



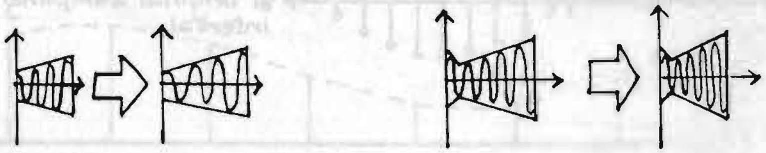
Specify end time & end speed relative to start

"TAPE-SPEED" VARIATION, DIGITALLY

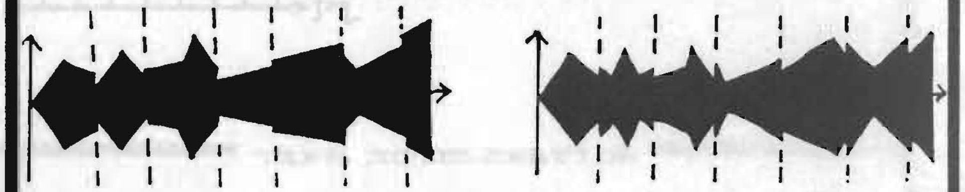




Cut sound into disjoint, or overlapping segments.
Segment length chosen so that, after they have been expanded or contracted (see below), they will rejoin into the exact duration of the original sound.

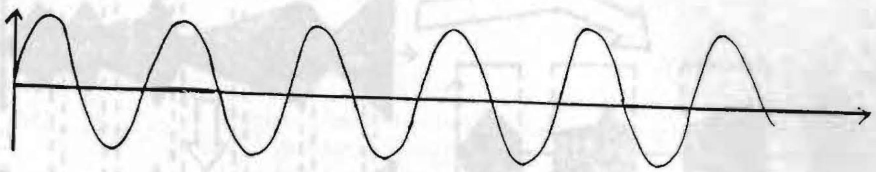


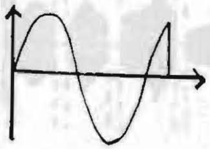
Change segment pitch by tape-speed variation, which also modifies segment lengths.



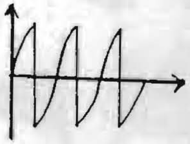
Rejoin changed-length segments, with no overlaps or gaps.
They now fit exactly into the original duration.

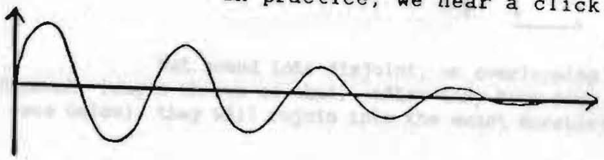
CUTTING





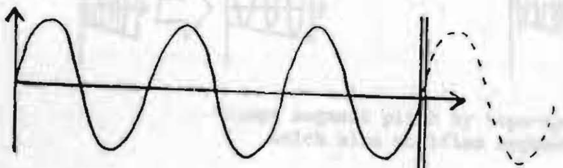
We cannot cut a sound simply by stopping it at an arbitrary point. Usually this produces a sharp edge (see left), equivalent to a fragment of a new waveform (see right) with many high partials.
In practice, we hear a click.





A normal cut is a brief fade-out to zero loudness.

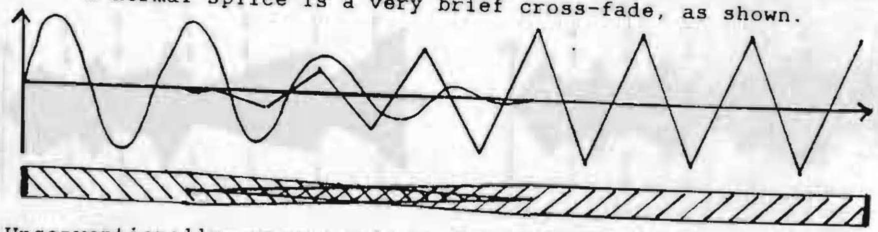
ZERO CUTTING



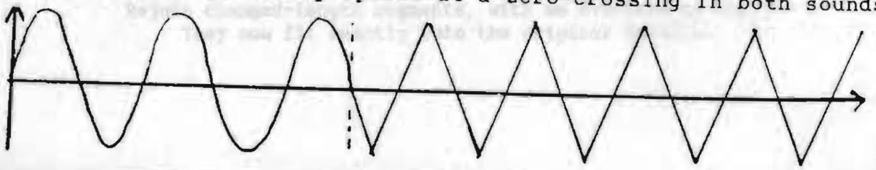
Unconventionally, we may cut a sound at a zero-crossing.

SPLICING

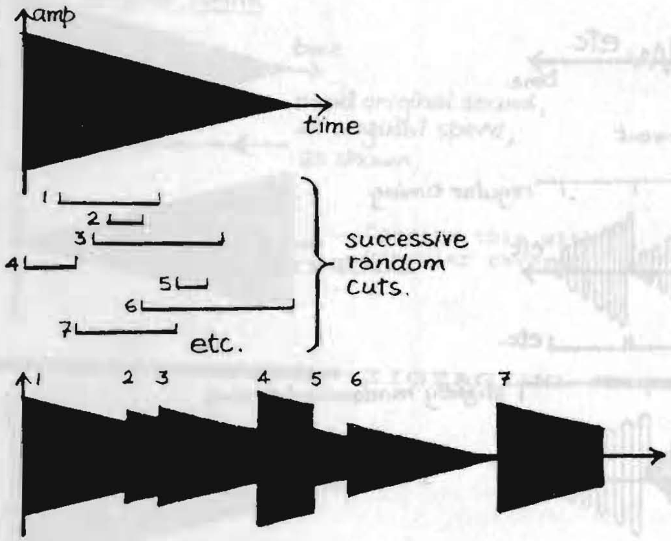
A normal splice is a very brief cross-fade, as shown.



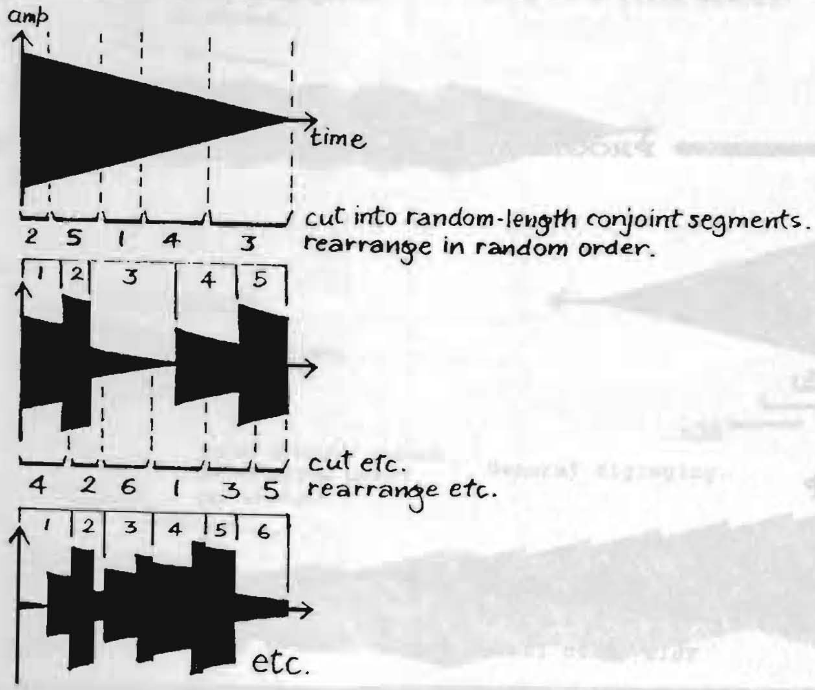
Unconventionally, we may splice at a zero-crossing in both sounds.



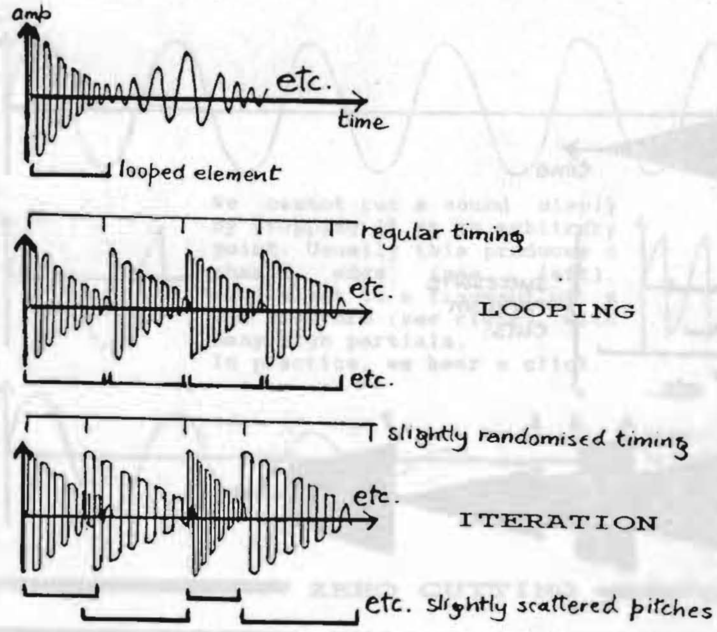
RANDOM CUTTING



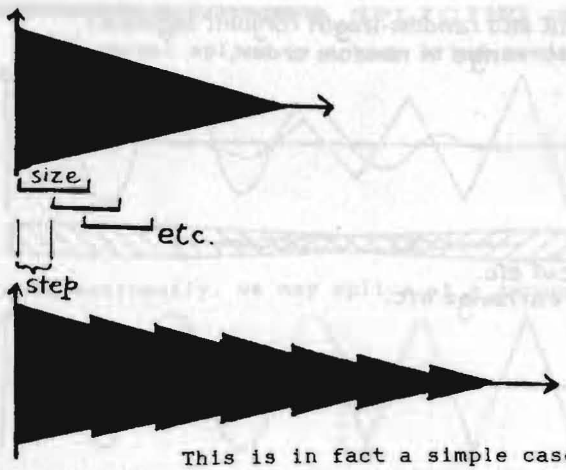
SHREDDING



LOOPING AND ITERATION

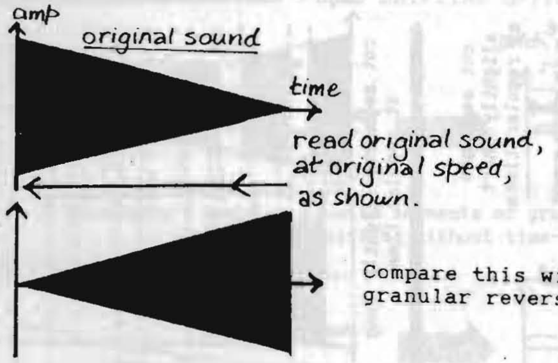


PROGRESSIVE LOOPING



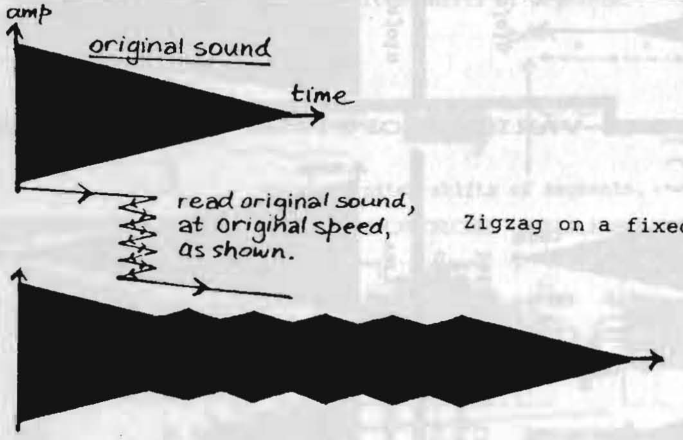
This is in fact a simple case of Brassage.

SOUND REVERSING

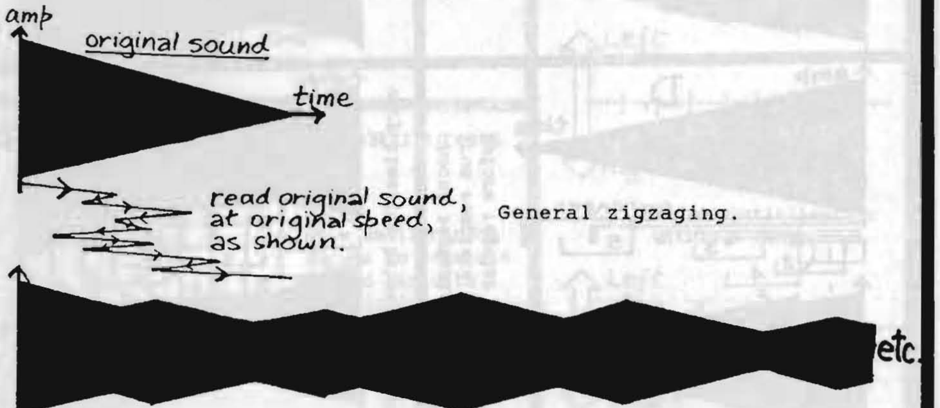


Compare this with granular reversal.

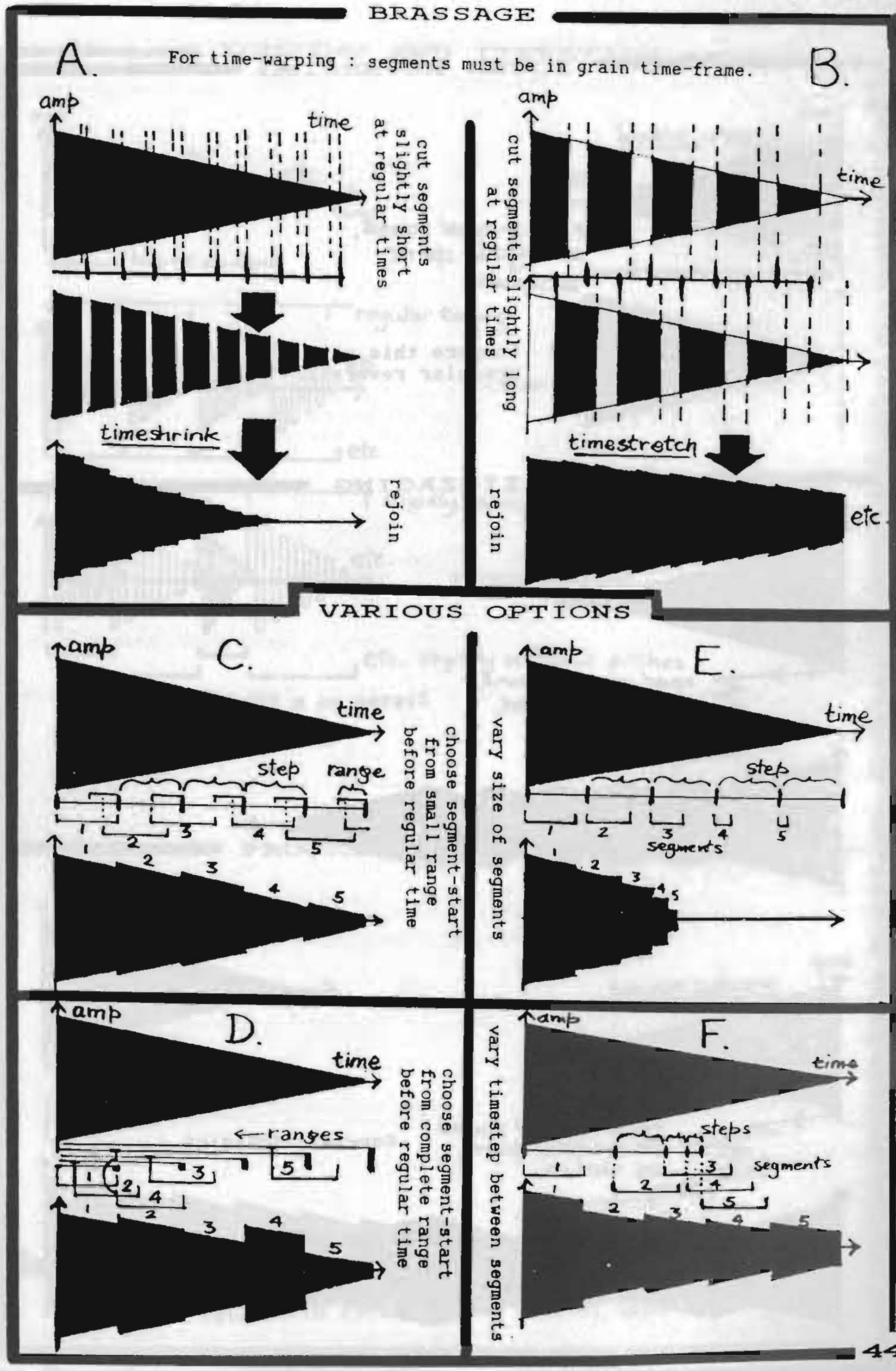
ZIGZAGING



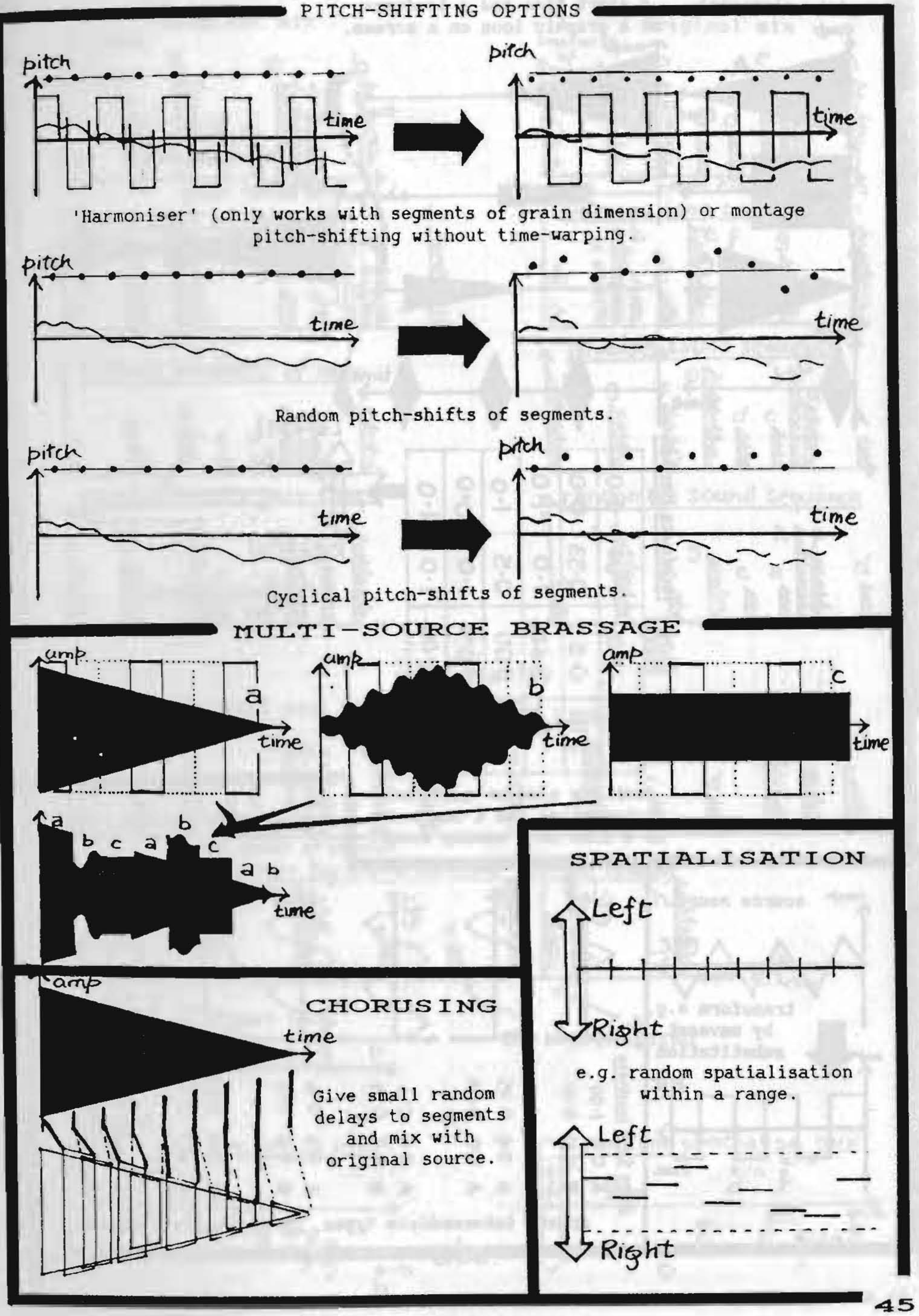
Zigzag on a fixed element.



General zigzaging.



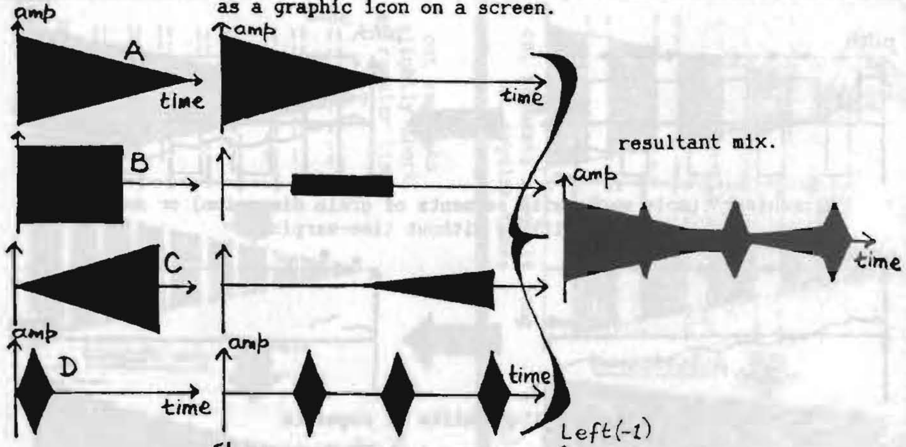
BRASSAGE (continued)



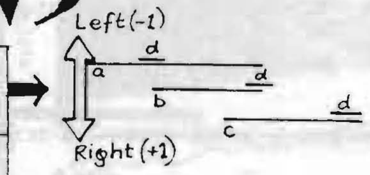
MIXING

Source sounds (mix elements)

each mix element assigned a start time and a loudness as a graphic icon on a screen.



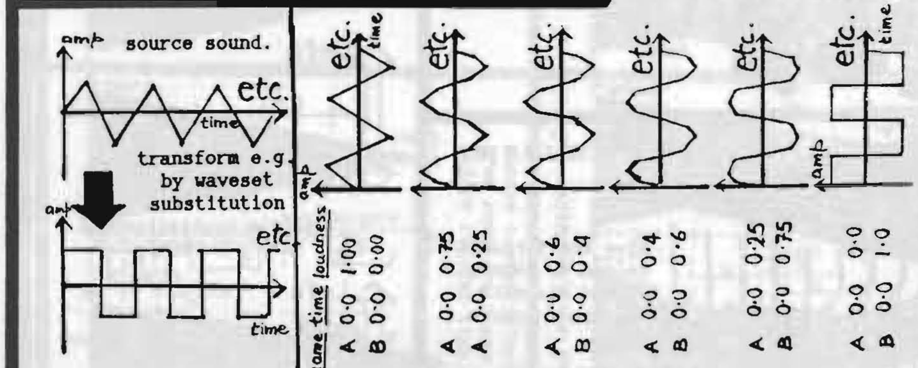
name	time (secs)	loudness	position
A	0.0	1.00	-1.0
B	0.5	0.33	0.0
D	0.625	1.0	-1.0
C	1.00	0.5	1.0
D	1.25	1.0	0.0
D	2.00	1.0	1.0



spatialisation of mix.

each mix element assigned a start time and a loudness on a line in a mixing score.

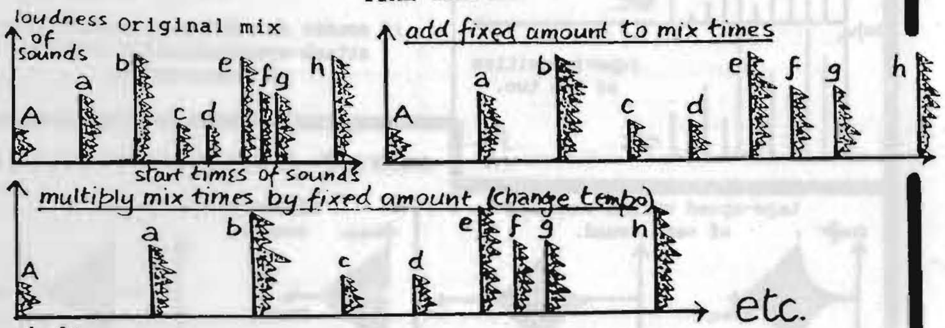
IN-BETWEENING



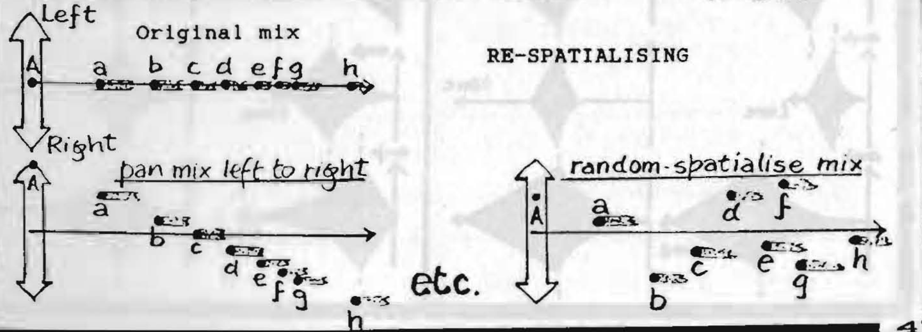
create intermediate types, by appropriate mixing.

MIX SHUFFLING

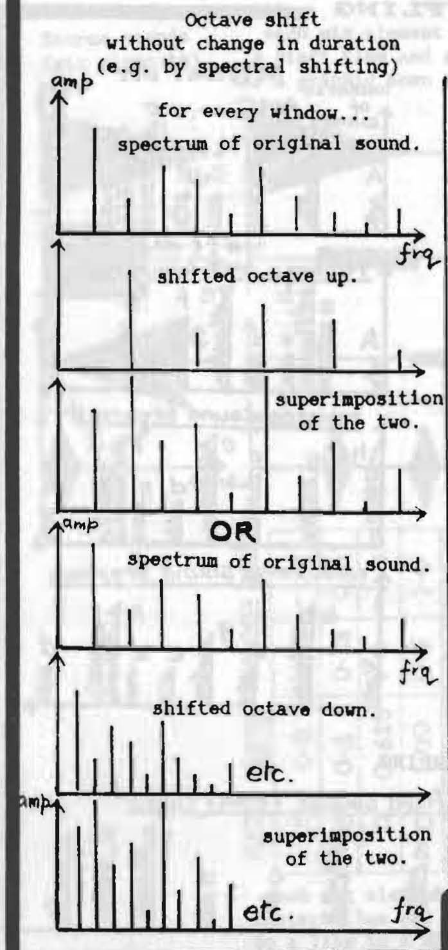
TIME WARPING

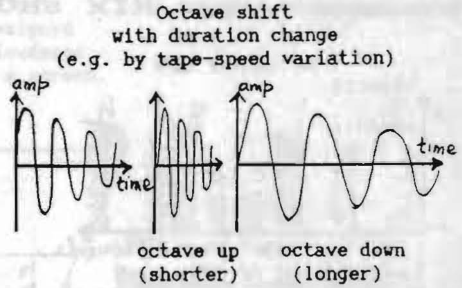


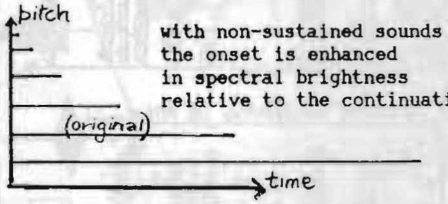
RE-SPATIALISING



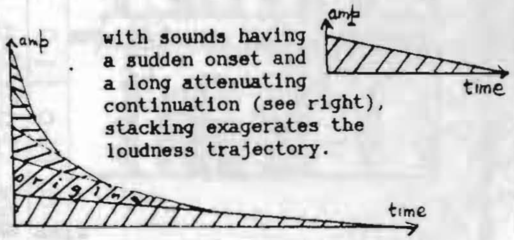
OCTAVE STACKING







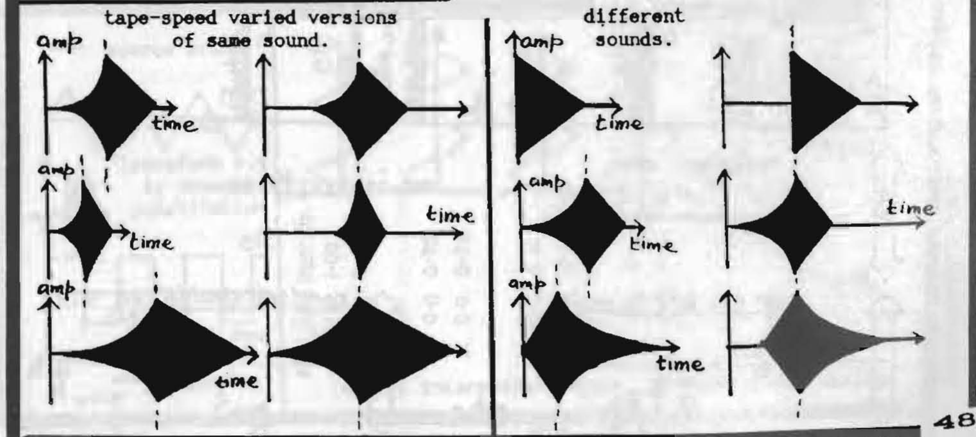
with non-sustained sounds the onset is enhanced in spectral brightness relative to the continuation.



with sounds having a sudden onset and a long attenuating continuation (see right), stacking exagerates the loudness trajectory.

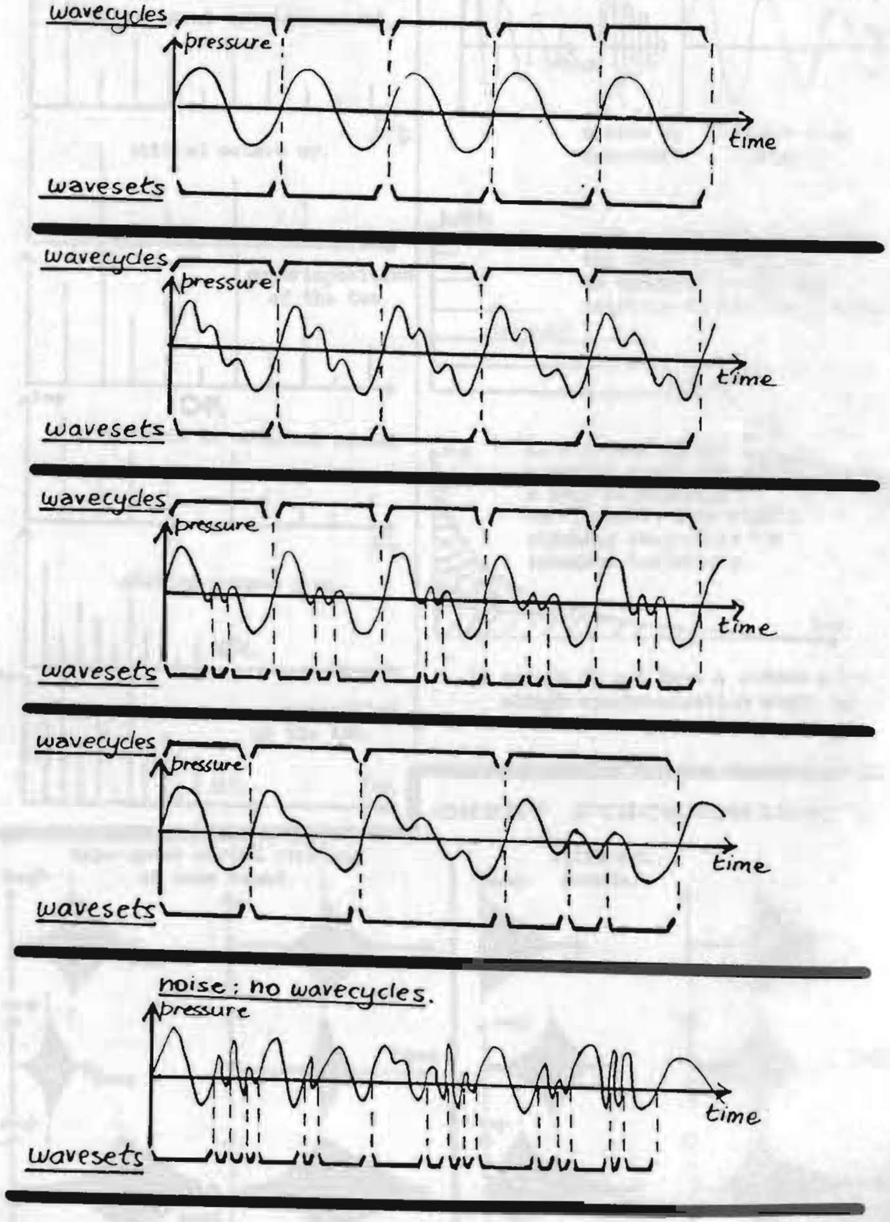
If sounds do not have a sudden onset, attack-synchronisation might be more appropriate.

ONSET SYNCHRONISATION

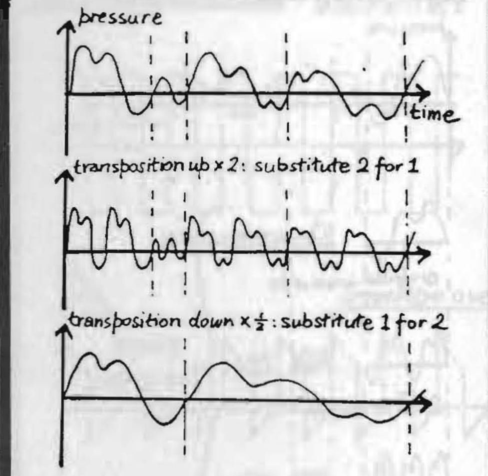


WAVECYCLES AND WAVESETS

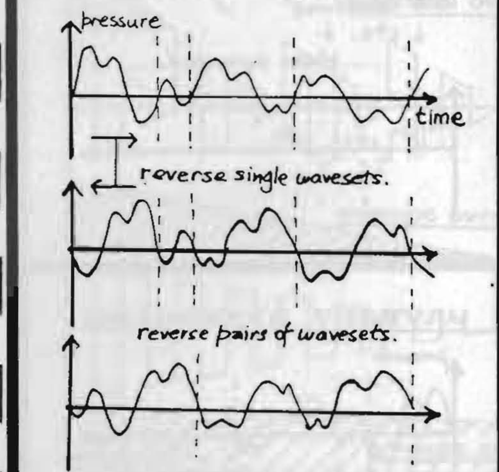
WAVECYCLE : Wavelength of sound, where clearly pitched.
WAVESET : Distance from a zero-crossing to a 3rd zero-crossing.



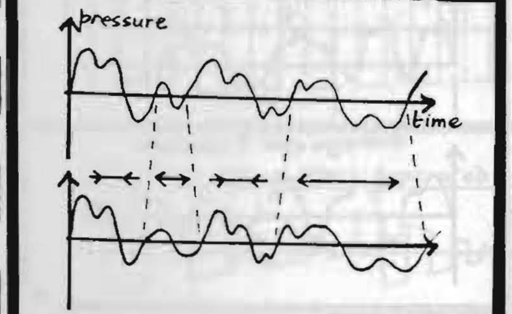
WAVESET TRANSPOSITION



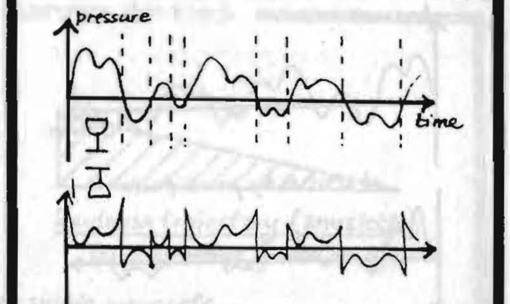
WAVESET REVERSAL



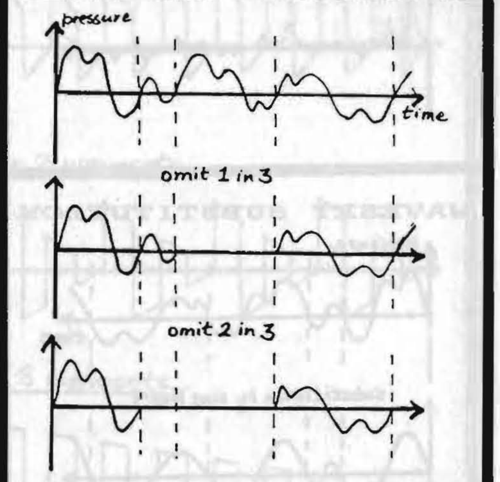
WAVESET SHAKING



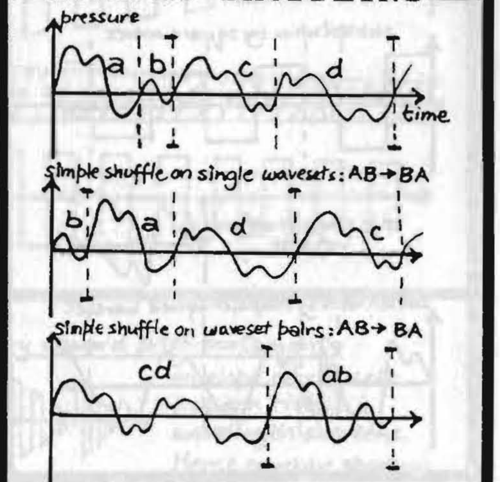
WAVESET INVERSION



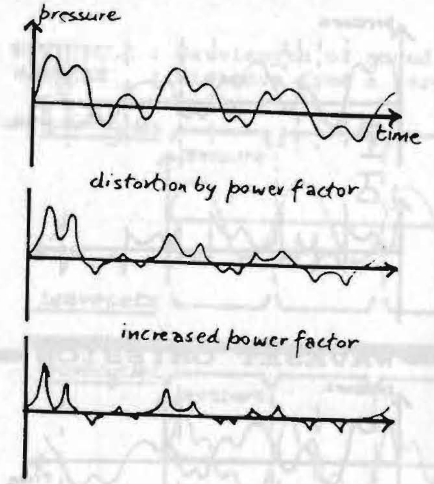
WAVESET OMISSION



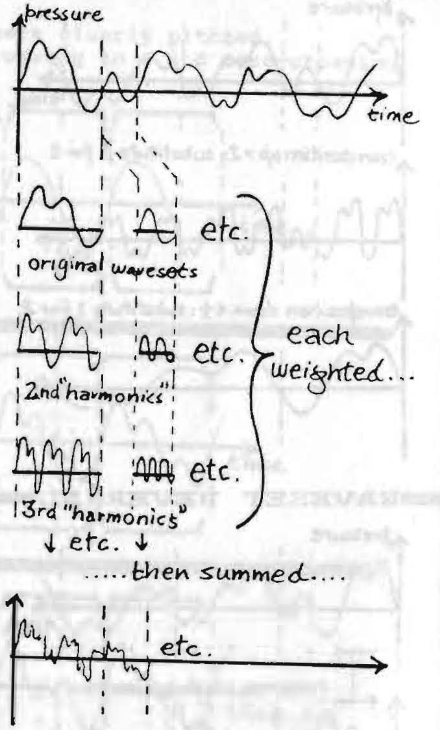
WAVESET SHUFFLING



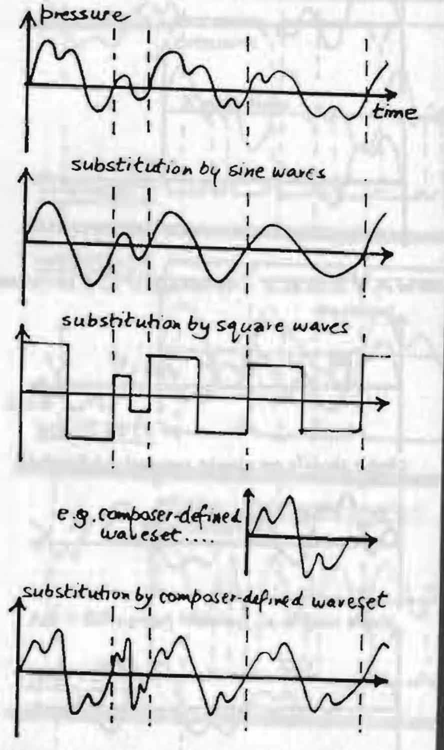
WAVESET DISTORTION



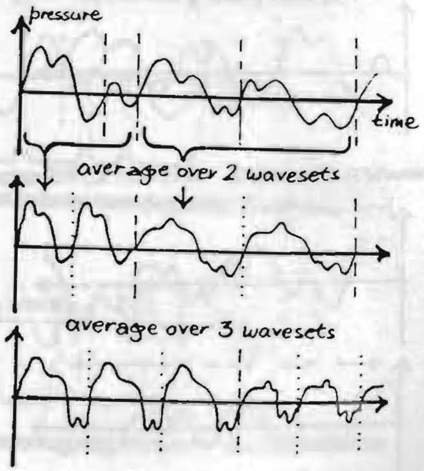
WAVESET HARMONIC DISTORTION



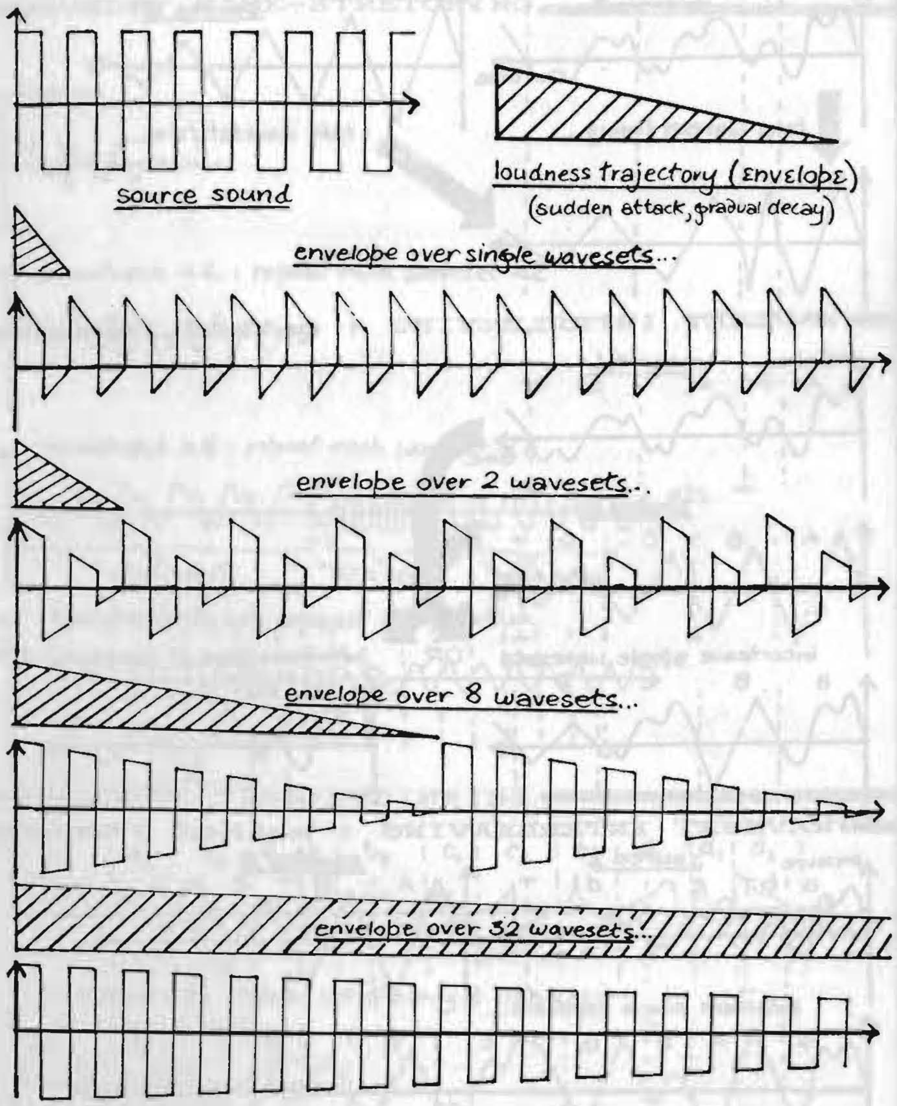
WAVESET SUBSTITUTION

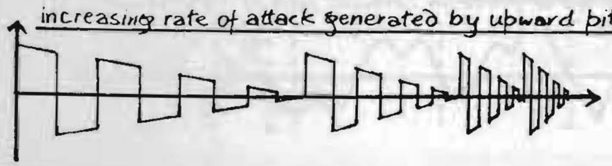


WAVESET AVERAGING



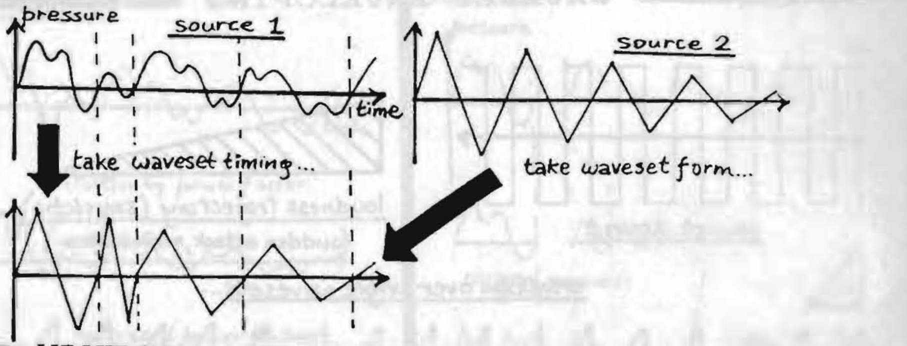
WAVESET ENVELOPING



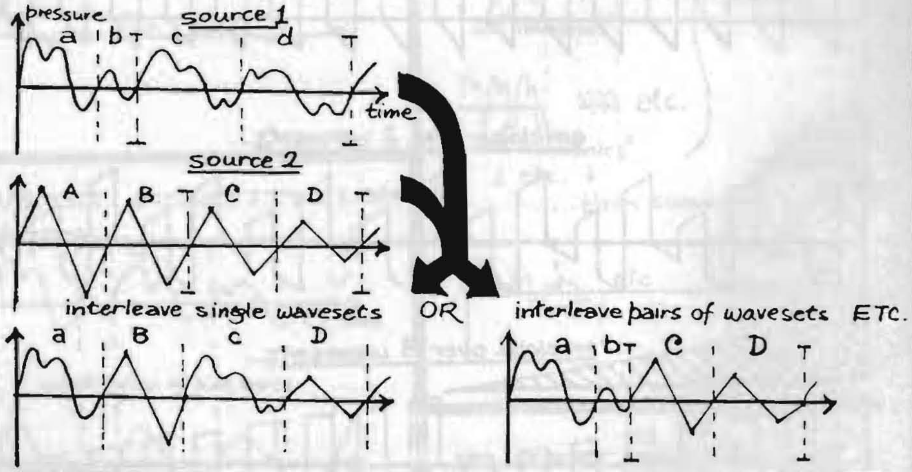


envelope over 4 sets.
As pitch rises,
wavelength shortens.
Hence envelope shortens.

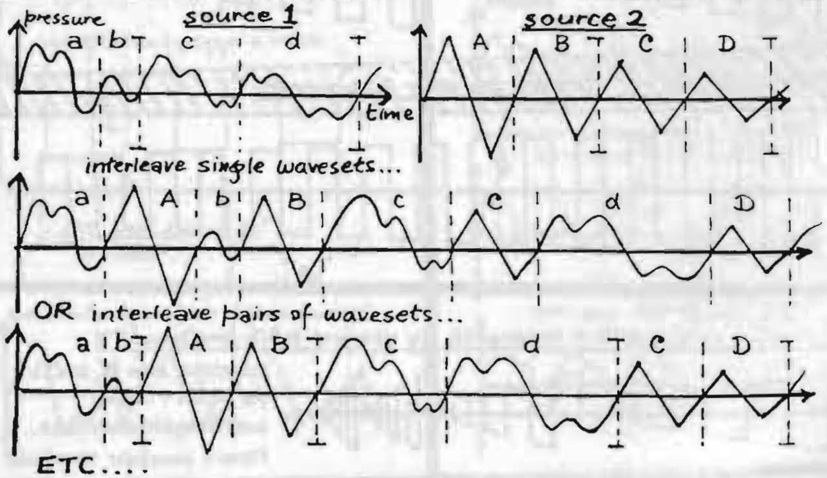
WAVESET TRANSFER



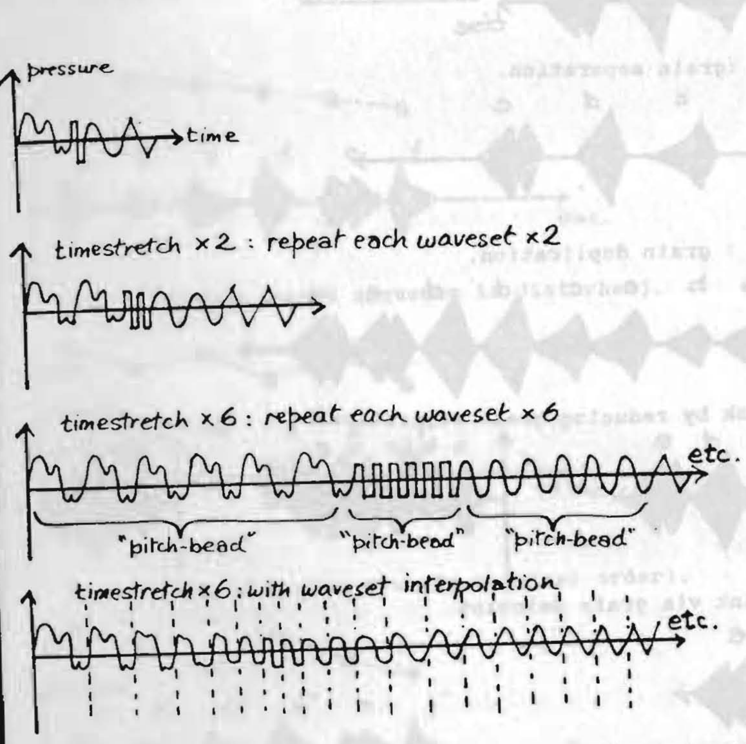
WAVESET INTERLEAVING : method 1



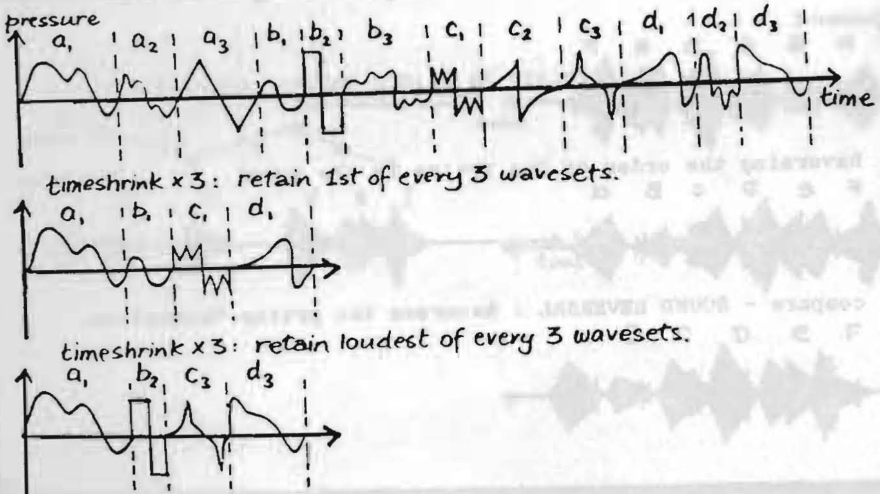
WAVESET INTERLEAVING : method 2



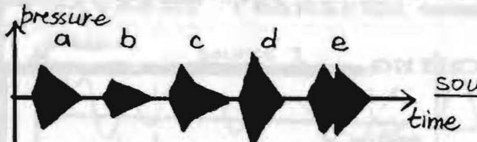
WAVESET TIME-STRETCHING



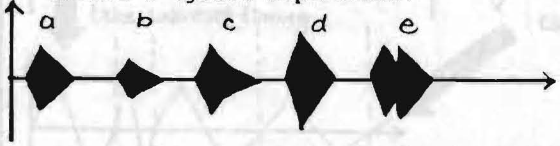
WAVESET TIME-SHRINKING



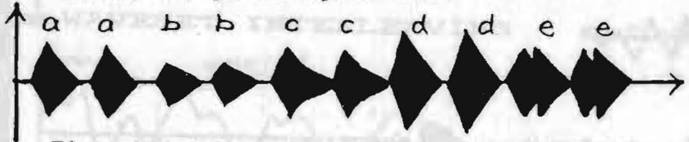
GRANULAR TIME-STRETCHING



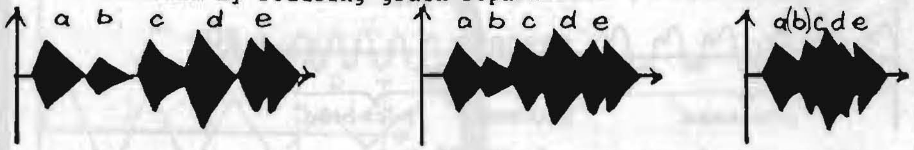
Method 1 :grain separation.



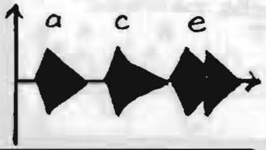
Method 2 : grain duplication.



Time-shrink by reducing grain separation.

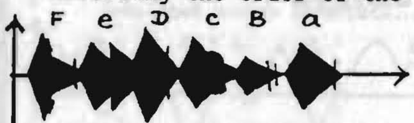


Time-shrink via grain deletion.

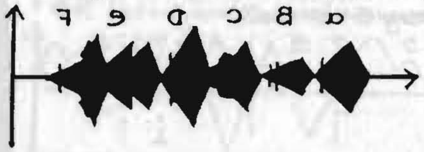


GRANULAR REVERSAL

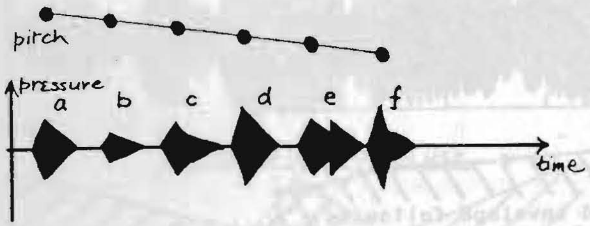
Reversing the order of the grains in the sound.



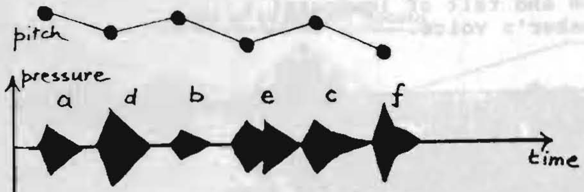
compare - SOUND REVERSAL : Reverses the grains themselves.



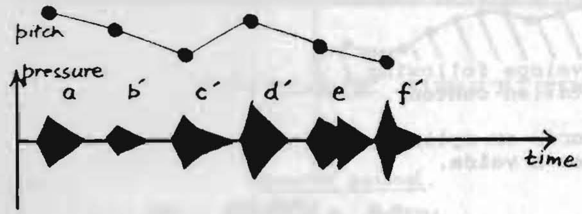
GRANULAR REORDERING



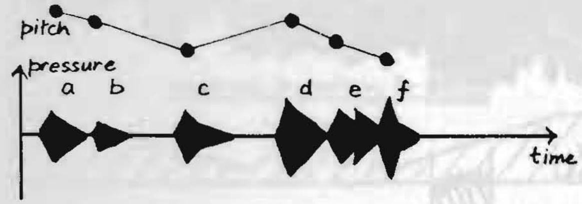
Reorder grains (pitch sequence is disturbed).



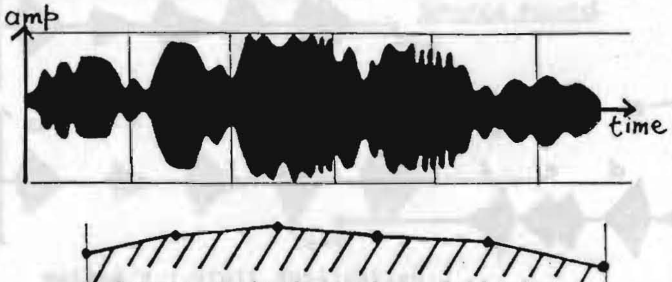
Alter pitch of grains (retain original order).



Alter pitch and scatter timing of grains.

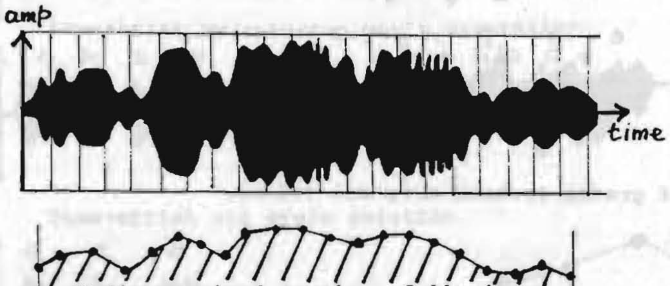


ENVELOPE FOLLOWING



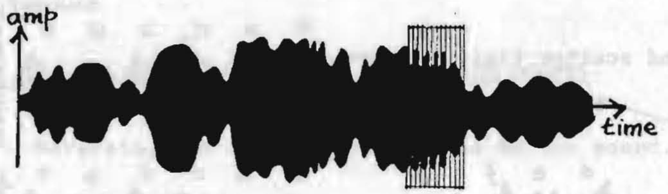
Coarse-grained envelope following.
Follows overall contour.

e.g. tracks rise and fall of loudness of speaker's voice.



Medium-grained envelope following.
Follows more detailed contour.

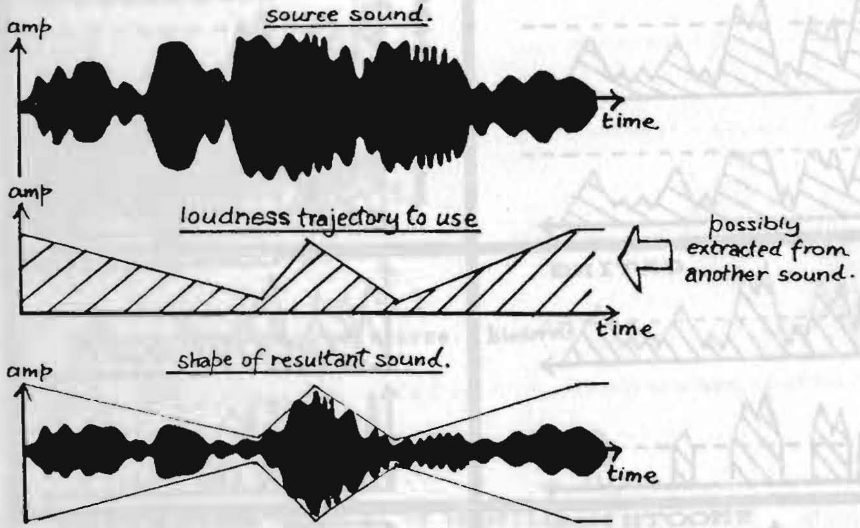
e.g. separates words or syllables in speaker's voice.



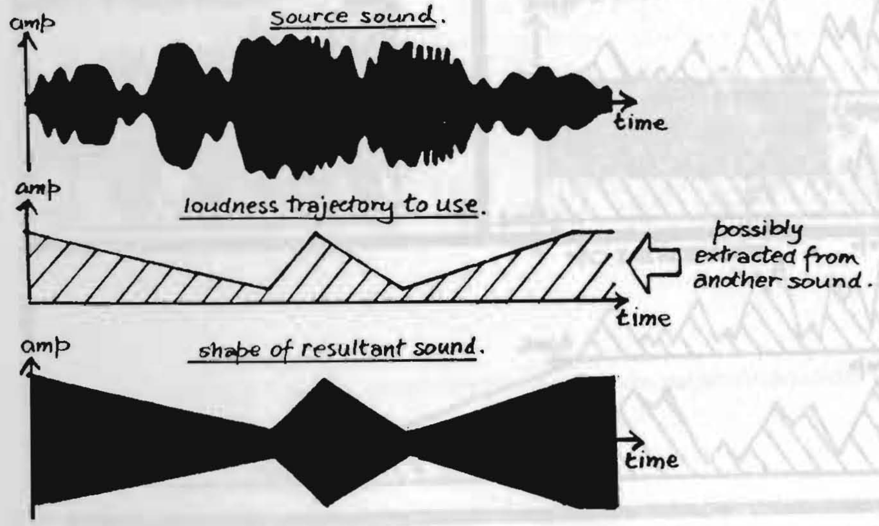
Fine-grained envelope following.
Follows finest detail of contour.

e.g. separates individual grains of a rolled 'r' in speaker's voice.

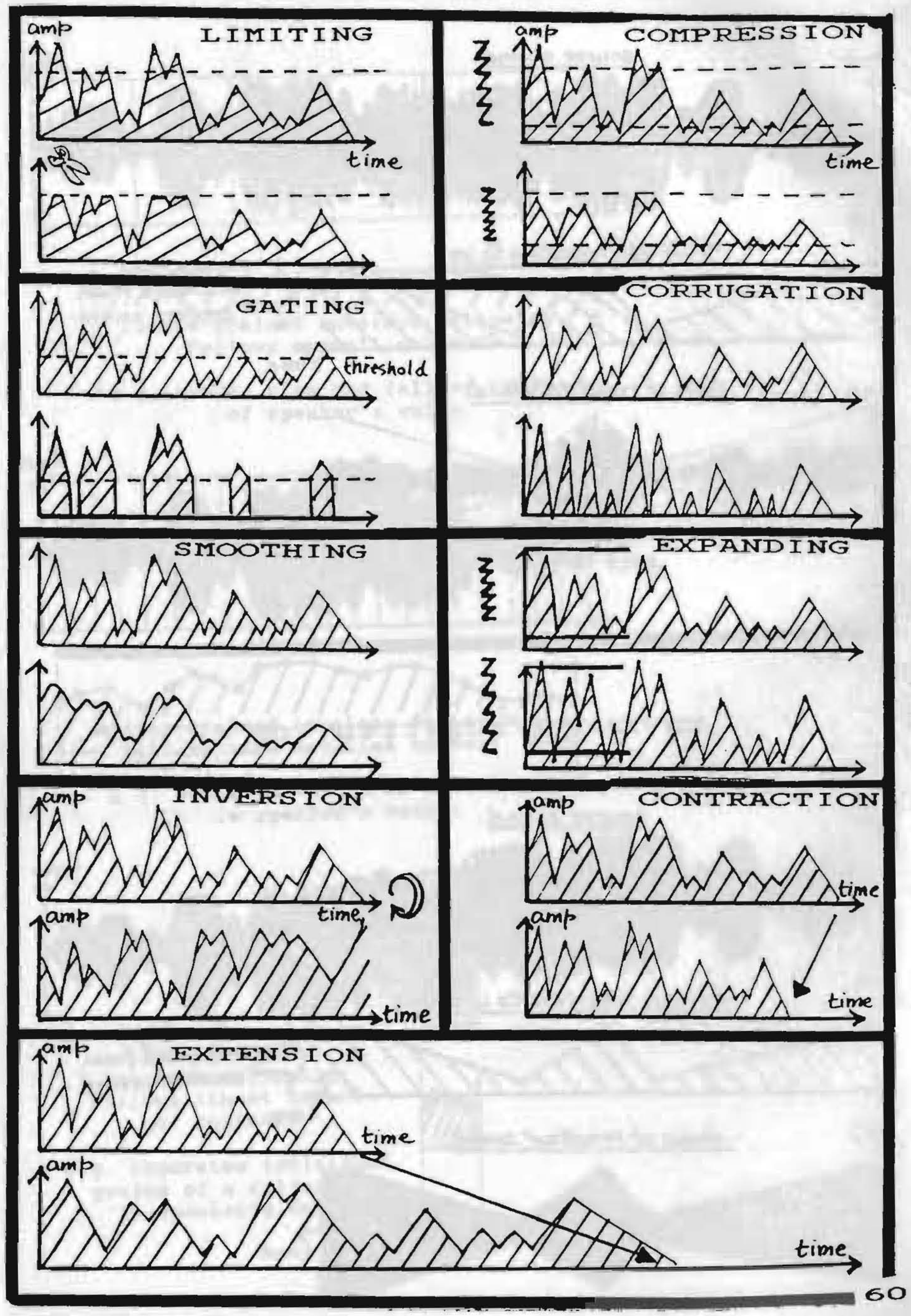
ENVELOPING



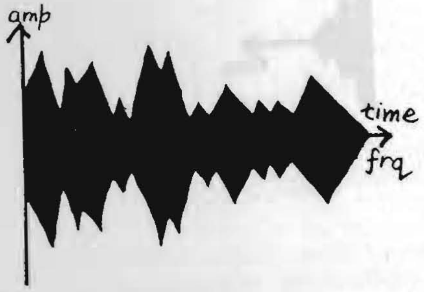
ENVELOPE SUBSTITUTION



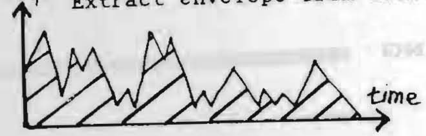
ENVELOPE TRANSFORMATION



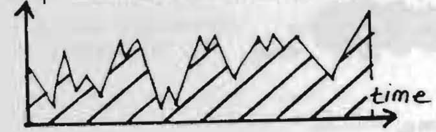
ENVELOPE TRANSFORMATION & SUBSTITUTION



Extract envelope from source.



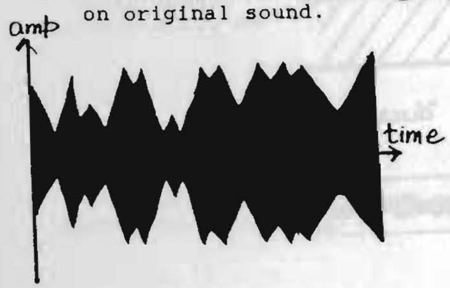
Transform envelope e.g. by inversion.

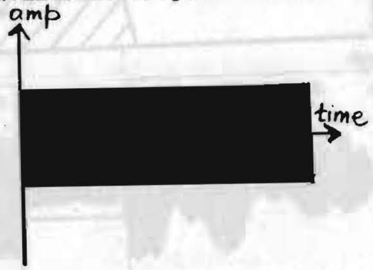


SUBSTITUTE new envelope on original sound.

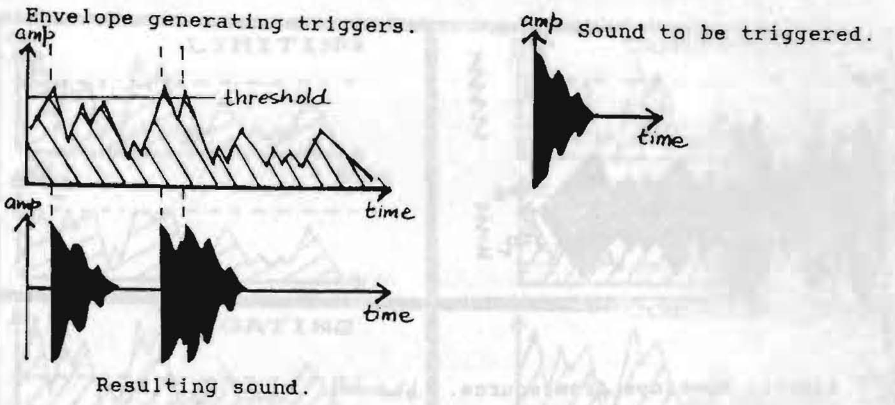
OR

Use new envelope for ENVELOPING original sound.

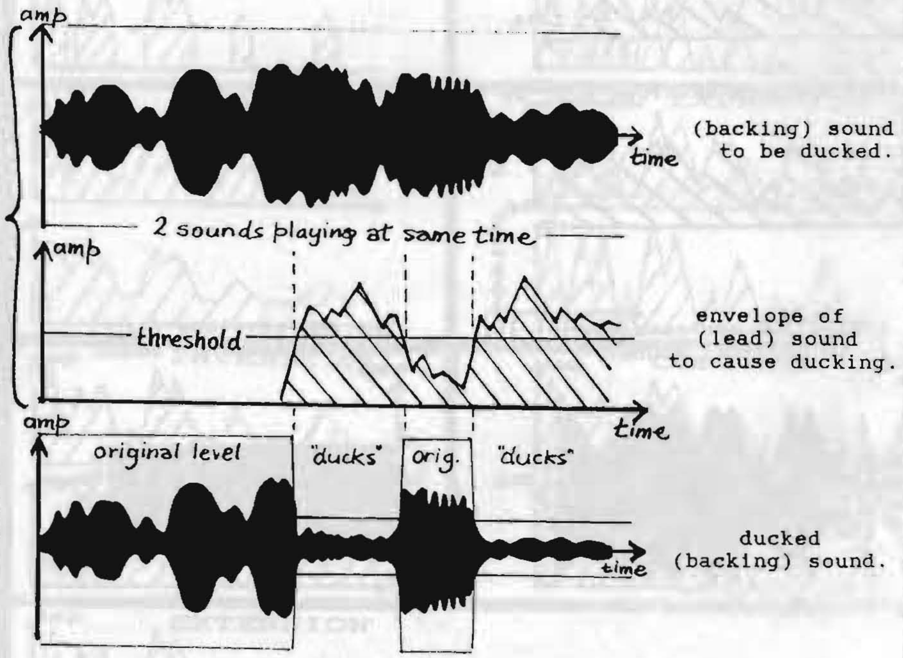




TRIGGERING

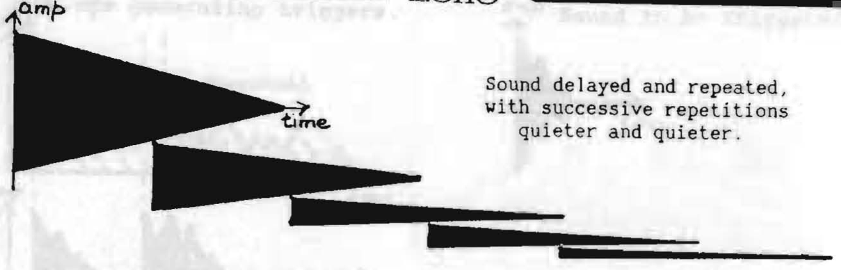


DUCKING



DELAY

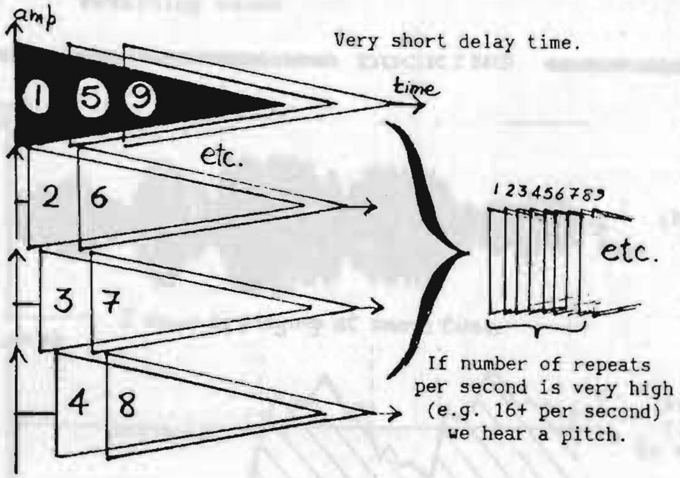
ECHO



Sound delayed and repeated,
with successive repetitions
quieter and quieter.

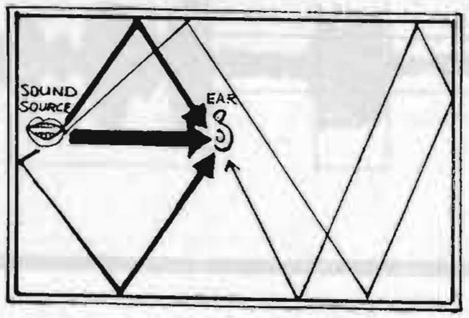
COMB FILTERING

Very short delay time.

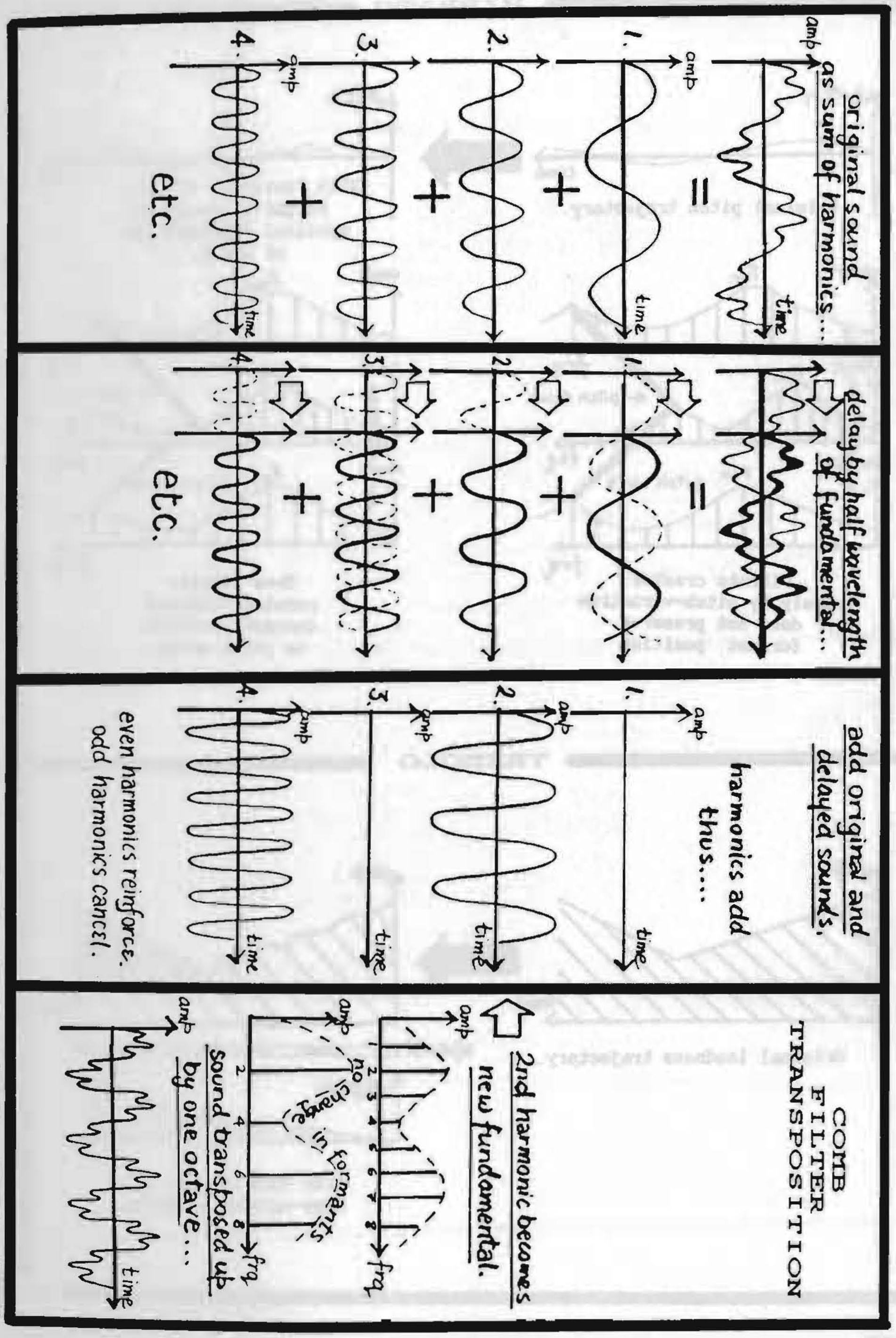


If number of repeats
per second is very high
(e.g. 16+ per second)
we hear a pitch.

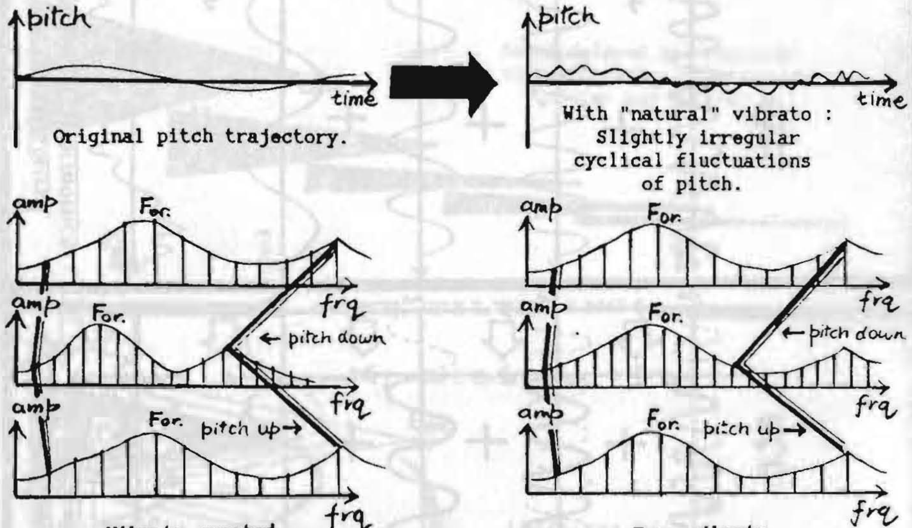
REVERBERATION



In an enclosed space,
sound reaching the ear directly
is mixed with many
differently delayed
versions of the sound.
The delays are caused by
the different distances
the sound must travel
on its various paths to the ear.
The delayed versions
are (usually) quieter
than the direct sound,
& have been changed in quality
through the filtering effects
of reflection from
different kinds of surfaces.



VIBRATO



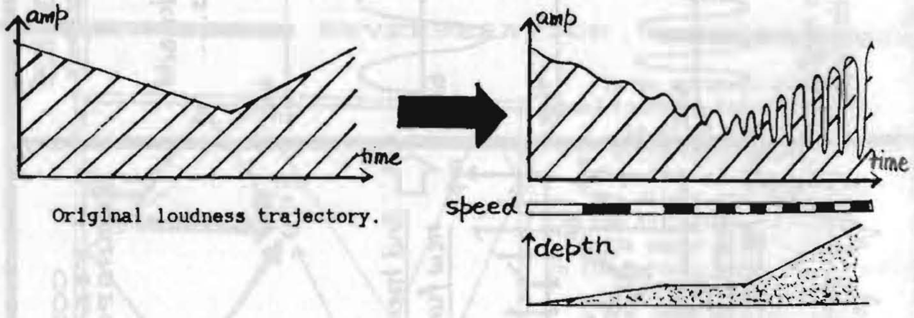
Original pitch trajectory.

With "natural" vibrato : Slightly irregular cyclical fluctuations of pitch.

Vibrato created merely by pitch-variation does not preserve formant position

True vibrato retains original formant position as pitch moves

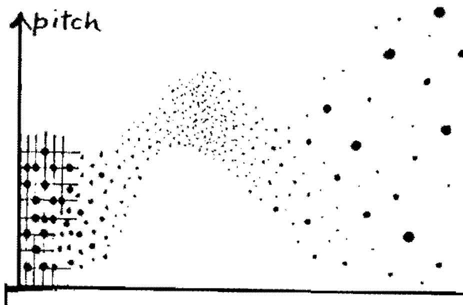
TREMOLO



Original loudness trajectory.

With (one example of) time-varying tremolo.

TEXTURE



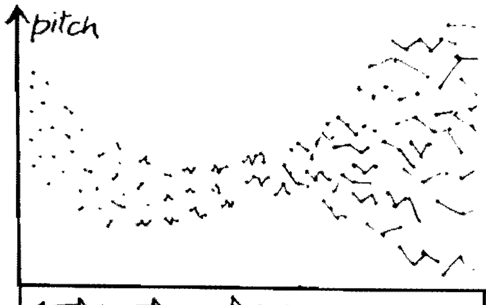
specify variation of....

- event density
- event regularity
- time quantisation grid
- number of sound sources
- hi & lo limits of loudness
- hi & lo limits of event duration
- hi & lo limits of pitch
- harmonic field (or not)
- range of event spatial position

medium ⇨ high ⇨ low	density
loud ⇨ quiet ⇨ mixed	loudness
low ⇨ high ⇨ full range	pitch range
quantised ⇨ random	time
harmonic field ⇨ random	pitch

{ · quiet event
{ • loud event

TEXTURE OF GROUPS



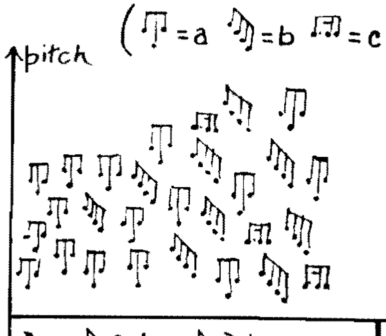
additionally specify variation of group properties...

- loudness trajectory
- hi & lo limits of size
- hi & lo limits of speed
- hi & lo limits of pitchrange
- harmonic field (or not)
- spatial form (e.g. moving left)

1 ⇨ 1-3 ⇨ 2-4 ⇨ 3-6	group size
fast ⇨ slow	group speed

etc.

TEXTURE OF MOTIFS



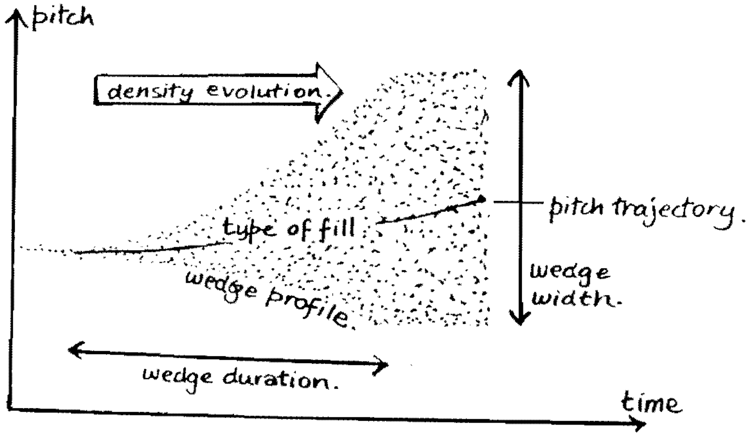
additionally specify variation of motif properties...

- motif type
- number of motifs

a ⇨ a,b ⇨ a,b,c	motif type & number

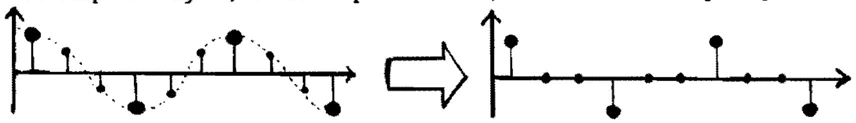
etc.

WEDGEING

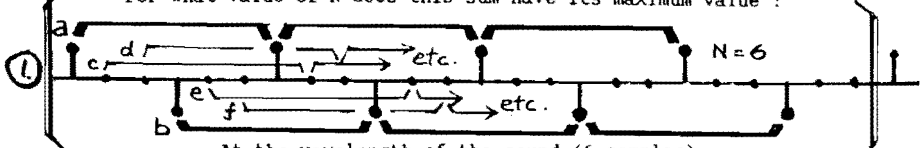


PITCH-TRACKING BY AUTO-CORRELATION

Example of an elementary auto-correlation procedure.
First clip the signal, so local peaks remain, but all other samples go to zero.

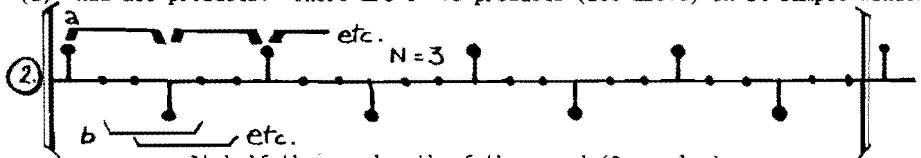


Add : the products of all pairs of samples separated by a step of N samples.
For what value of N does this sum have its maximum value ?



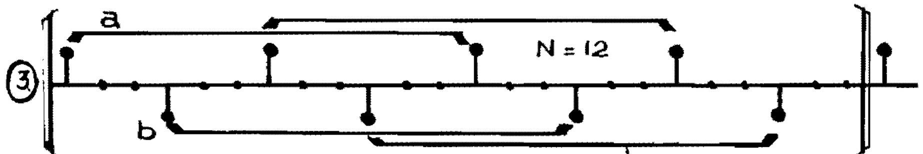
At the wavelength of the sound (6 samples).

(1) Multiply each sample value by value 6 samples ahead, within the window.
(a) All these products are [+ve value] X [+ve value] = [+ve value].
(b) All these products are [-ve value] X [-ve value] = [+ve value].
(c-f) All these products are zero X zero = zero.
(2) Add all products. There are 6 +ve products (see above) in 24-sample window.



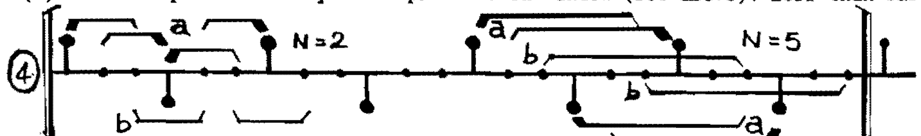
At half the wavelength of the sound (3 samples).

(1) Multiply each sample value by value 3 samples ahead, within the window.
(a) All these products are [+ve value] X [-ve value] = [-ve value].
(b) All these products are zero X zero = zero.
(2) Add all products. Total must be a -ve value, & hence less than case 1.



At twice the wavelength of the sound (12 samples).

(1) Multiply each sample value by value 12 samples ahead, within the window.
(a) All these products are [+ve value] X [+ve value] = [+ve value].
(b) All these products are [-ve value] X [-ve value] = [+ve value].
(c) All other products are zero X zero = zero.
(2) Add all products. Only 4 +ve products in window (see above). Less than case 1.

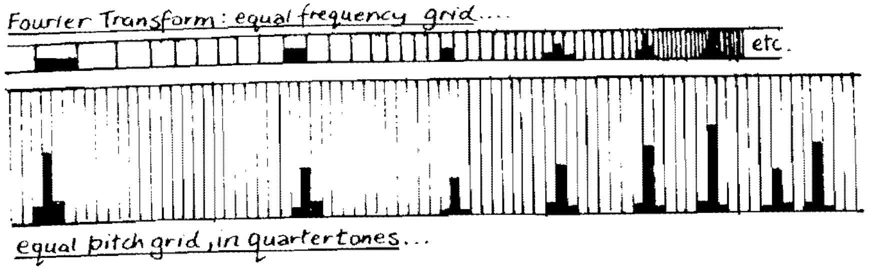


At all other sample steps.

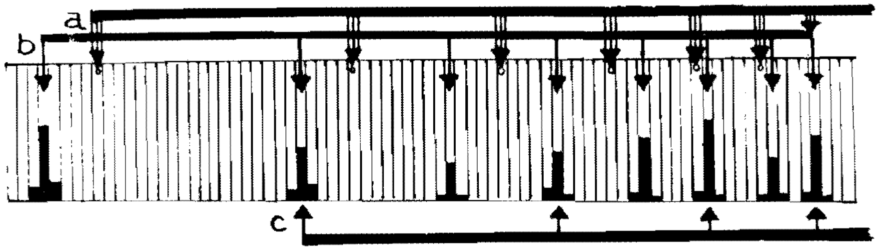
(1) Multiply each sample value by value N samples ahead, within the window.
(a) All these products are [+ve or -ve value] X zero = zero
(b) All these products are zero X zero = zero.
(2) Add all products. Sum is zero, and therefore less than case 1.

Hence sum has maximum value when step is wavelength of sound (case 1). We can thus track wavelength, & hence frequency. With more complex waveforms, transient subpeaks tend to cancel each other in the sum, over a large window, but as pitch may be changing, window can't be too big. Wrong-by-an-octave errors are a problem.

PITCH-TRACKING BY PARTIAL ANALYSIS



For each time-window
transfer spectral data from the equal frequency grid of the Fourier Transform, to an equal-pitch grid, spaced in quarter-tones.



We cannot merely take the lowest peak in the spectrum as the pitch. For example, there may be no pitch present in the particular window we are considering, so that the position of the minimum spectral peak has no pitch-tracking significance.

To find a pitch effectively we must proceed as follows.

For each quarter-tone channel on the pitch-grid:

(1) Find the value in that channel.
(2) Find values in those channels which fall on the harmonic-series template above that channel (a,b,c).
(3) Form a weighted sum of these values.

Most sums of this kind will miss all or most partials in spectrum (e.g. 'a' in diagram)

Templates which fall on the partials (e.g. 'b' & 'c' in diagram) will yield significant values for the sum.

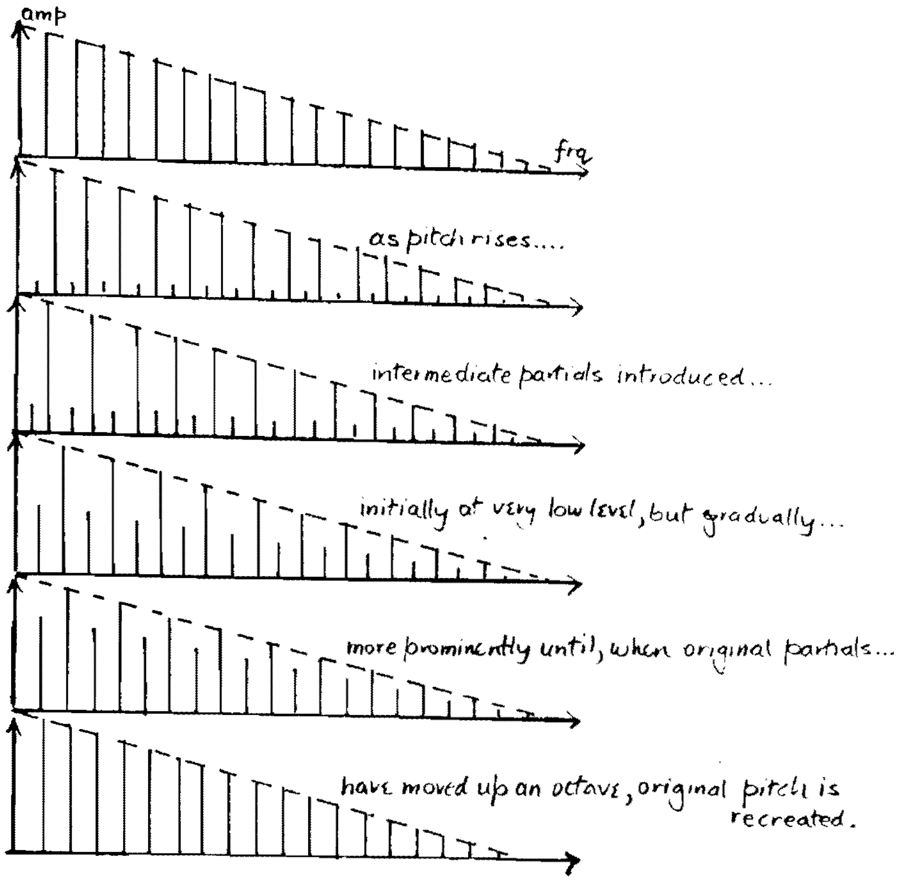
The sums given by 'b' & 'c' can only be differentiated if appropriate weightings are given to the contributions of each partial in the (suspected) spectrum.

Weightings for specific instruments (or other sound sources) may be well known.

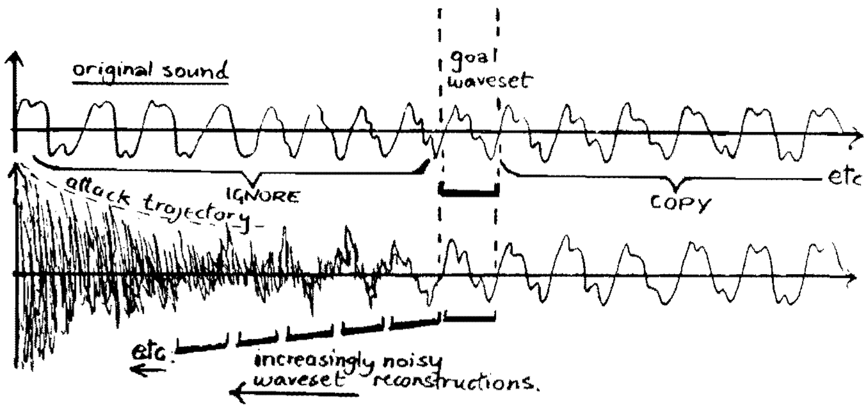
Difficulties arise with sources, like the voice, which can change their spectral contour from window to window.

SHEPARD TONES

In this example, a tone of rising pitch always stays in the same tessitura



SOUND PLUCKING



GRANULAR RECONSTRUCTION

Step is (average) distance between segments in source sound.
Output density = 1 joins segments end to end as in Brassage.