Preface to 6th Edition 14apr07

This document is about Horizontal Ambisonic Decoders. Periphonic (with height) Decoders have bigger Shelfs. See **"Practical Periphony" - M Gerzon AES preprint 1571, 1980** I'll extend it when height is important.

It was originally written for DSP gurus designing the ultimate Ambisonic Surround Decoder. It's really a personal re-discovery (still to be completed) of arcane 20+ yr old arts. Difficult to find stuff I wanted as a software & hardware designer; of interest only to DSP and hardware freaks.

Section 2.0 is the most important and difficult part. There hasn't been an explanation of Shelf Filters and how they relate to Ambisonic Decoders for 20+ yrs if it ever existed. Most of it only available from close examination of old decoder designs, "General Metatheory .." & other Gerzon papers. Hard work. eg Fig 12 in Practical Periphony *obviously* translates to Eqn 4.4

Probably why Fons' "Happy Days" decoder is the only software decoder apart from the secret Meridian, with proper Shelf Filters. No one understands them. David McGriffy's VVMic has proper phase matched Shelf Filters for regular polygon decoders. Bruce Wiggins has an experimental decoder with phase matched Shelfs but his public WAD decoders are not phase matched yet.

IMHO, these are the only software decoders which MAG would deem Ambisonic. If you know of any others, please let me know.

Other novel bits in this paper are the

- description of various decoders as virtual mikes feeding a regular polygon decoder
- need to specify, Shelf Filters AND the associated Decoder. Practically all other treatments of Shelf Filters make no sense cos the associated Decoder is not clearly defined.
- **Rationalised Decoder** which is **best if you DON'T use Shelf Filters**.

Dr. David Pickett, a former director of Tonmeister programmes at Surrey and Indiana Universities, rightly upbraids me for obfuscation & lack of clarity. I beg his forgiveness for only dealing with some of his constructive comments. A simpler, clearer version is planned.

But for the multitudes who want to do their own decodes with the excellent DAW tools now available ...

... and who start at the last chapter of a whodunnit ...

The basic decoder is Eqn 4.2 Rationalised Square Decoder and Eqn 4.3 & 4.4 for the relevant speaker layouts.

The relative gains of W & XY make these **'Energy' optimised decoders. Use these if you DON'T use Shelf filters.**

The other 2 decoders involve just changing W by 3 dB in either direction.

Dropping W gives 'Velocity' optimisation which is good at LF but poor everywhere else. Tiny sweet spot & probably some 'in head' effects. "Probably", cos people vary in susceptibility. At Eric's & Aaron's prompting, I now use the term **'in head'** rather than **'phasiness'** cos Gerzon has a different definition of 'phasiness'.

Raising W gives 'Controlled Opposites' which is best for large audiences. A terrible name. I shall call this the **Cardioid** decoder from now on. *Dr. Pickett will probably forgive my abandoning the Pickett Cardioid moniker for this decoder but not my sloth in preparing a simpler clearer version of this document ... 8>D*

You might want to listen to the 3 basic decodes to see if I'm spouting BS. Easy to twiddle W.

What Shelf filters do is to change from 'Velocity' at LF to 'Energy' at HF around 380 Hz ... from the 20+ yr old listening tests by the original Ambisonic team.

The responses are in **Table 4.2 & Fig 4.3** Do a simple Shelf EQ with the tools you have on your DAW. This is what Bruce Wiggin's excellent WAD (early 2006) regular decoders do.

You apply these to WXY before the basic Eqn 4.2 Rationalised Square Decoder.

Where this approach falls down is that the Shelf filters really need to be phase matched ... again from the 20+ yr old listening tests but I don't think anyone has tested this since. Bruce Wiggins uses non phase matched IIR filters cos they are the easiest to do ! What Bruce & David Pickett have shown is that non phase matched Shelf Filters are better than no Shelf Filters.

You can't do Phase Matched filters with the usual DAW tools. (I may be wrong. Look for 'Linear Phase' filters or EQ in your DAW documentation.)

Section 8.0 dreams up XY48.wav W48.wav XY44.wav W44.wav (and shorter versions), the impulse responses of the Phase Matched filters at 48kHz & 44.1kHz.

I'm told CoolEditPro / Audition can download an impulse response like this into a "FFT Filter".

Thanks to Angelo, I might be able to confirm this in the near future.

So my advice would be to

a) Try listening to the 3 basic decodes
b) Try simple DAW minimum phase EQ for Shelf filters
c) If you can do a 'FFT Filter', try 'proper' phase matched Shelf filters.

If you can compare c) with b) I'd be VERY interested in what you hear. But see www.ai.sri.com/ajh/ambisonics

If you have a **large audience** (more than 6) and a big space, use Eqns 4.2 4.3 & 4.4 but raise W' by 3dB. This gives **Cardioid** decoding. **Don't use Shelf Filters** and you must read Grand Vizier Malham's

Experience with Large Area 3-D Ambisonic Sound Systems
www.dmalham.freeserve.co.uk/ioapaper1.pdf

IMPORTANT CORRECTION

The rectangular decoders in the early 5mar06 edition is wrong. Please destroy any copies of that document.

Eqn 4.4 Aaron Rationalised Rectangular Decoder is an exact solution to Fig 12 THE DESIGN MATHEMATICS from Gerzon's "Practical Periphony". It replaces my poor "1% accurate to speakers at +-15°" solution in the 2jul06 edition.

It is the first appearance in (virtual) print of the exact Rectangular Decoder. It came about from work on

"**Localization in Horizontal-Only Ambisonic Systems**" - Benjamin, Lee & Heller AES oct06 San Francisco

Read this for listening tests on various decoders. Apart from Bruce Wiggin's tests, these are possibly the only formal listening tests on different Ambisonic decoders since the original Ambisonic team +20 yrs ago.

Aaron has put an expanded version of the preprint on his website with Gnu Octave code to solve Fig 12 THE DESIGN MATHEMATICS for other layouts. There is an important caveat about using Fig 12 solutions. These result in 'Velocity' decoders so ONLY give good results with matching Shelf filters. Without Shelf filters, these give poor performance.

Because Fig 12 is a 'Velocity' decoder, you should use the section 4.0 Filters rather than the section 4.1 Shelf Filters..

www.ai.sri.com/ajh/ambisonics

The Aaron Rationalised Rectangular Decoder in Eqn 4.4, however, is a **Rationalised Energy decoder** so can be used "as is" without Shelf Filters though Shelf Filters will improve performance.

Richard Lee ricardo@justnet.com.au

the Old Postoffice
Cooktown
Q4895
Australia +61 (0)438 731 027

Acknowledgements A very long list. Please forgive me if I have left anyone out.

Aaron Heller, Angelo Farina, Bruce Wiggins, David McGriffy, David Pickett, Eric Benjamin, Etienne Deleflie, Fons Adriaensen, Geoff Barton, Paul Doombusch, Richard Dobson, Richard Furse and the members of sursound.

SHELF FILTERS for Ambisonic Decoders - Richard Lee

An Idiot's Guide to Designing Software Ambisonic Decoders. Discusses the design of both analogue and digital shelf filters. Should be read in conjunction with

http://www.ambisonia.com/Members/ricardo/ASD.htm/ : the Ambisonic Surround Decoder and its references. The most important of these is **General Metatheory of Auditory Localisation** - M Gerzon, preprint 3306 AES Vienna mar92

Simple square G-Format Nimbus 4.0 encoder (Square Ambisonic Decoder) described.

Proposes a '**Rationalised Decoder** and **Shelf Filter**' specification

A certain facility with DSP theory is assumed.

Also stuff about **Distance Compensation.**

1.0 Intro

Shelf filters are an important part of the Ambisonics myth. Even today, Ambisonic decoders are little understood. Some pseudo gurus believe an Ambisonic decoder simply simulates a number of cardioids pointing in the direction of the supposed speaker positions. While this naive form of surround encoding/decoding is a lot better than the crude pair wise panning used in early 'Quad' systems and naive multi-channel DAWs, it is far from optimum.

Some worthy software Ambisonic decoders do not incorporate Shelf filters cos their implementators "don't believe in Shelf filters". Close inspection of other implementations show that subtleties vital to 'good' Shelf filters; ie filters that improve localisation, sound and overal listening pleasure, have not been properly addressed.

I hope to explain with minimal (well slight) poly-syllabicity, what Shelf filters need to do and describe at least one 'good' software implementation. As Shelf filters are integral to the design of Classic Ambisonic Decoders, this has grown into an **Idiot's Guide to Designing Software Ambisonic Decoders**. I'm not really competent to pontificate on GUI OOP stuff, so I'll limit myself to the design of a simple command line **G-Format Nimbus 4.0 encoder**. I'll leave the all singing, all dancing Windoz players to the experts.
But I hope these prophets may be persuaded to incorporate proper **Shelf filters** and **Distance Compensation** in their designs.

2.0 W vs XY

Shelf filters adjust the ratio of W to Vel to optimise localisation for different applications. These ratios can be defined by the equivalent Virtual Mikes which feed a Regular Polygon Speaker Layout.

There are 3 important W vs XY ratios which differ by 3dB

Velocity Decode
is appropriate at LF and satisfies Gerzon's 1st order phase or 'velocity' localisation model. Adjusting W can set the Velocity Vector Magnitude to 1.0, same as a "real" source but this can only be achieved for the central listener. The Virtual Mike is a strong Hypercardioid with fig-8 component 6dB greater than omni on axis. Richard Furse calls this the 'Matching' decoder as it would give 'matching' WXY outputs from a First Order Soundfield Mike at the central position. I'll call it **'Velocity'** decode.

But Ambisonic Surround Decoders are not meant for Soundfield Mikes but the Mk1 Human Head, a slightly squashed ovoid with sensors surrounded by irregular flaps called pinnae on the sides slightly low and behind a L-R diameter. At LF, both a Soundfield Mike and the Mk1HH are small compared to the wavelengths so this decoding is appropriate.

At frequencies above 700 Hz, the size and shape of the Mk1HH means 'Velocity Optimisation' gives a small 'sweet spot' and some 'in head' effects (because of the processing inside the ovoid) 'Velocity Optimisation' is not suitable for the whole frequency range.

Attempting to match the output of a 1st order Soundfield Mike in the listening position, to the original B format signals results in this type of decoder which is not good for listening.

Energy Decode
is 'optimum' between 500 - 5k Hz. and maximises Gerzon's Energy Vector Magnitude. At these frequencies, the size of the Mk1HH means the phase between the two ears starts cycling and becomes unreliable as a guide to localisation. Instead, Inter-aural Level Differences can provide good cues so W is adjusted to 'make these as close to real life as possible' (maximise Gerzon's Energy Vector Magnitude). The maximum average value this can have is 0.707

The Virtual Mike is still Hyper but fig-8 only 3dB greater than omni on axis.

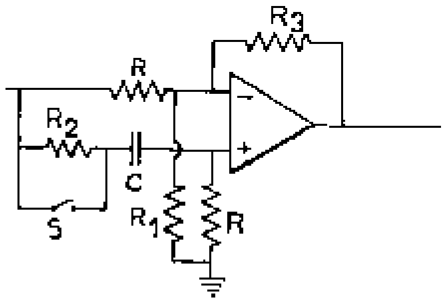
Energy Optimisation also gives good results for off-centre listeners in a domestic environment and is **suitable as a general purpose decoder**.

*I'll define '**domestic environment**' as the central 'sweet spot' plus 2 seats either side and the 3 seats behind.*

Cardioid Decode
Richard Furse calls this Controlled Opposites, a terrible name. Cardioid Virtual Mikes. Nothing from the opposite speaker. *Actually the virtual mikes are cardioid only for regular polygons. Attempting cardioid virtual mikes for irregular (but still diametric opposite) rigs, eg thin rectangles, gives very poor results.*

The previous Velocity & Energy decodes with the hypercardioid Virtual Mikes, have out-of-phase output from the opposite speaker to sharpen the image for central or near central listeners. But if you are sitting near a speaker, and a source is panned from that speaker to the opposite side, you never hear it get there because the speaker next to you still has output. Dave Malham calls this effect 'Bounce Back'.

Cardioid decode avoids this problem and is best for presentation to large audiences. Not as good as Energy for the domestic environment.



3.0 Fig 3.1 : Analogue Shelf Filters
used in all hardware Ambisonic Decoders from the Integrex and even before. Response is

$$(-kL + kH\, sT) / (1 + sT) \qquad \textbf{Eqn 3.1}$$

where T = sqrt (kL / kH) / (2 π F) F is the pivot frequency.
= C * (R + R2)
kL = R3 / R
kH' = R1 + R3 / (R1 * R3) Gain without R2
R2 = R (kH' - kH) / (kL + kH)

Introducing R2 drops the HF even further without changing the phase much.
If R1 is large, R2 = 0 and R3 = R, this becomes the standard 1st order All Pass Network with response

- (1 - sT) / (1 + sT) flat Amplitude Response but twice the phase shift of a simple pole.

The zero (1 - sT) is in the right half of the s-plane which make this Non Minimum Phase.

At HF, C is SC and if R2 = 0, the circuit has gain [R / (R + R2)] * (1 + R3/R1)
At LF, C is OC and it inverts with gain R3 / R.

The pivot is near T = RC. Different gains are obtained by spacing the pole and zero. We adjust T to get **matching phase** between W which has gain at HF and Vel which has loss. This **PHASE MATCHING** is vital to Shelf filters. The original listening tests started off with 700Hz for the pivot but ended up with **380Hz** as best.

Above from Rolv-Karsten Ronningstad's website
http://www.geocities.com/ambinutter/AMBISONIC_HOME_PAGE.html
Rolv has a useful **Excel spreadsheet** for designing Analogue Shelf filters. Read this section with it.

Also circuits and info for nearly all **hardware Ambisonic decoders past & present**. Of particular interest are the **Integrex** and Geoff Barton's designs which include the Minims, Troy, professional Cepiar and Audio & Design.

Table 3.2 Integrex Shelf Filters				Old Spec	
	W	XY	Pref	W	XY
R	22k	22k	22	22k	22k
R1	17k6	36k9	11.11	32k2	OC
R2	2k9	1k056	0	3k2	911
R3	82k	82k	30.3	25k4	25k41
C	15 nF	20n	11n	15n	20n55
F	385Hz	391Hz	400 Hz	380Hz	380 Hz

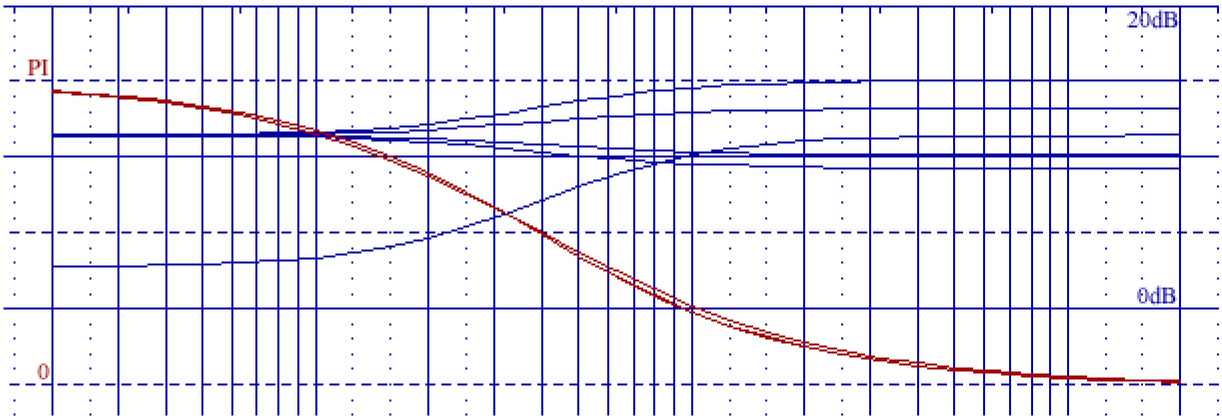


Fig 3.3 **Integrex Shelf filters** shows 5 curves : From top to bottom at LF Phase in Red

1	W' without R2	UHJ
2	W' with R2	Horizontal B format 1.75dB
3	XY' without R2	Horizontal B format -1.25dB
4	XY' with R2	UHJ
5	Forward Preference	optional in UHJ

Phase for the first 4 match to better than 2°. The Preference filter is the most different with the shallowest phase slope as it has the biggest shelf. The first 4 have the same gain at LF

4.0 Old Specification

The traditional spec for Shelf filters comes from **"Practical Periphony" - M Gerzon AES preprint 1571, 1980**

Table 4.1 OLD SPEC	LF	HF
Horizontal W	0dB	1.2245x 1.76dB
XY	0dB	0.866x -1.25dB
Periphonic W	0dB	1.414x 3.01dB
XYZ	1.2247x 1.76dB	0dB

The horizontal coefficients preserve 'energy' = W'^2 + X'^2 + Y'^2 at HF & LF.
Are the periphonic coeffs correct? What is the periphonic speaker matrix?

http://www.geocities.com/ambinutter/UHJ_and_Ambisonic_equations.html *has different filters. (0.45dB more HF) though his Excel spreadsheet uses the Gerzon values.*

The output of the **Shelf filters in Fig 3.3** can **feed a speaker matrix** directly.

In the following discussion, I use 1.4 for sqrt(2) and 0.7 for 1/sqrt(2).

W' X' Y' are W X Y with Shelf Filters & Near Field Compensation applied.

*Remember **B-format defines W** as a perfect **omni** mike with gain **-3dB** wrt to the on-axis gains of **X Y Z**, perfect orthogonal **fig-8 velocity** mikes pointing forward, left and up. See*

http://www.york.ac.uk/inst/mustech/3d_audio/secondor.html

A regular polygon of speakers at angles ϕ_i measured anti-clockwise from CF, has X' & Y' are scaled by the speaker direction cosines, cos ϕ_i & sin ϕ_i.

Speaker Feed = W' + 1.4 cos ϕ_i X' + 1.4 sin ϕ_i Y' Eqn 4.1 old regular polygon matrix

A **Shelf Filter definition MUST** include the **associated Speaker Matrix**. Here, the complete definition is given by the Fig 3.3 Response Curves, Table 4.1 and Eqn 4.1 old regular polygon matrix

May I ask other Ambisonic Gurus to clearly show the matching Speaker Matrix when discussing Shelf Filters.

The associated Square Speaker Matrix decoder is

LB = W' - X' + Y' etc

But this old decoder, fed with unshelved WXY, gives 'Velocity' decode which has critical sweet spot and may sound 'in head'.

4.1 RATIONALISED DECODER SPECIFICATION

If instead we use a **Square Matrix decoder** with

LB = W' - 0.7X' + 0.7Y' etc **Eqn 4.2 Rationalised Square Decoder**

with unshelved WXY, the decoder optimizes 'Energy' vector magnitudes and this give good results for most frequencies and also off-centre listeners in a domestic environment. This is a **good general purpose decoder**; working well without and even better with shelf filters.

W' X' Y' are W X Y with Shelf Filters & Near Field Compensation (section 4.2) applied.

For enhanced performance, Shelf filters are now defined as

Table 4.2 RATIONALISED SHELFs	**LF**	**HF**
380 Hz Horizontal W	0.8167x -1.76dB	0dB
XY	1.155x 1.25dB	0dB

The **Rationalised Specification** consists of **Eqn 4.2 the Decoder**, **Table 4.2** the **Shelf specification** and

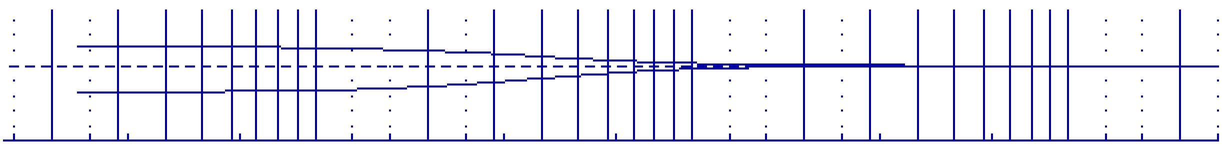


Fig 4.3 Rationalised Shelf Filters with 0dB gain at HF. Phase MUST match. Both centred around 380Hz
This gives the same results for Shelved signals as section 4.0 but will revert to the Energy decode for unshelved WXY. Gerzon says (and I agree) 'Energy' is appropriate for off-centre listening in domestic environments and is good in the centre.

This bla is just saying things in a different way for the true Ambisonic Surround Decoder. But as the Square Matrix decoder is also the "G-Format Nimbus 4.0 encoder", I want it to give good results even if an implementer doesn't believe in Shelf filters, doesn't know how to do them properly or the Nimbus 4.0 source material is unshelved.

I'll end this section on the **Rationalised Decoder** with the corresponding **Regular Polygon & Rectangular Decoders**.

The **Regular Polygon Decoder**, has X' & Y' scaled by the direction cosines of the speakers, cos ϕi & sin ϕi.

Speaker Feed = W' + cos ϕ_i X' + sin ϕ_i Y' **Eqn 4.3 Rationalised Regular Polygon decoder**

Speakers at angle ϕi measured anti-clockwise from CF

The layout **MUST** be regular to use **Eqn 4.3**. If not, something similar to Eqns 4.4 must be used and design becomes much more complicated. Further complexity if the speakers are not equidistant and as front / back symmetry is lost.

The **Rectangular Decoder** with front speakers at +- ϕ is

LB = W' + (- X' / cos ϕ + Y' / sin ϕ) / 2 **Eqn 4.4 Aaron Rationalised Rectangular decoder**
LF = W' + (X' / cos ϕ + Y' / sin ϕ) / 2
RF = W' + (X' / cos ϕ - Y' / sin ϕ) / 2
RB = W' + (- X' / cos ϕ - Y' / sin ϕ) / 2

Thanks to Professor Heller for this solution to Fig 12 in Gerzon's "Practical Periphony" AES preprint 1571 1980

This is the simplest example of a **Non-Regular Diametric Opposite Decoder**. Note that for the front speakers close together (ϕ small) we have to use MORE Y' and LESS X' which is NOT what a naive decoder might assume.

4.2 Near Field Compensation (formerly Distance Compensation)

acknowledges that the speakers are at a finite distance from the listener. This a simple **Minimum Phase HP** filter on **X' & Y'** to deal with "proximity effect" on the Velocity signals cos the speakers are 'near' the listener.

NFC can be done with a simple LF rolloff for X & Y at

f = c / (2 * PI * d) = 54.6 / d Hz c : speed of sound 343 m/s **Eqn 4.5 Frequency for NFC**
d : speaker distance m **HP Filter**

Not critical. A general 2-3m 27-15 Hz rolloff gives good results for a wide range of distances PROVIDED THE LISTENER IS NEARLY EQUIDISTANT. It will have to be an IIR as the low frequencies involved exclude short FIRs. **20 Hz (2.73m) is a good choice for G-Format Nimbus 4.0** and used in all known hardware decoders.

If the speakers are at different distances, the correct NFC MUST be applied to X' & Y' for each speaker. This is an added complication for irregular speaker arrangements. For a discussion see

http://www.ambisonicbootlegs.net/Members/ricardo/ASD.htm/#Real

For normal rectangular or regular polygons, Fons reports

*... this is really **essential for all except large rigs**. Material containing substantial **LF energy (e.g. organ music)** really sounds horrible without it.*

My experience is

People vary in their susceptibility to this and the recording must have clear 'phase' cues in the first place. I've heard this most clearly on my recording of the Fitzwilliam with the Soundfield within a metre of the players.

Switching Distance Compensation out doesn't change the timbre at all but cellos seem to have the wrong shape. As the cellist plays down the scale, the phase errors due to proximity change the localisation, so he appears to move.

I heard this to a lesser extent on other chamber music recordings. Perhaps cos the mike wasn't so close?

1st order HP H(s) = sT / (1 + sT) has an **IIR** **H(z) = (b0 + b1 z^{-1}) / (1 + a1 z^{-1})** Eqns 4.6

b0 = 1 / (k + 1) b1 = -b0 a1 = (k - 1) b0

where k is given by Eqns 7.3

5.0 Type 2 Digital FIR All Pass Shelf Filters

simply **match the Analogue 380 Hz All Pass** designs as Fig 3.3 These match "energy" gain at LF & HF.

For our first efforts, we'll do a simple 1024 point FIR. A true Type 2 Digital All Pass Shelf Filter would be an efficient IIR as in Section 7.1

Our All Pass network has components as in Table 3.2

We generate 1025 frequency points from 0 - 24000Hz and export the linear complex frequency response of our transfer functions. An Inverse FFT gives us the impulse response of our Shelf filter.

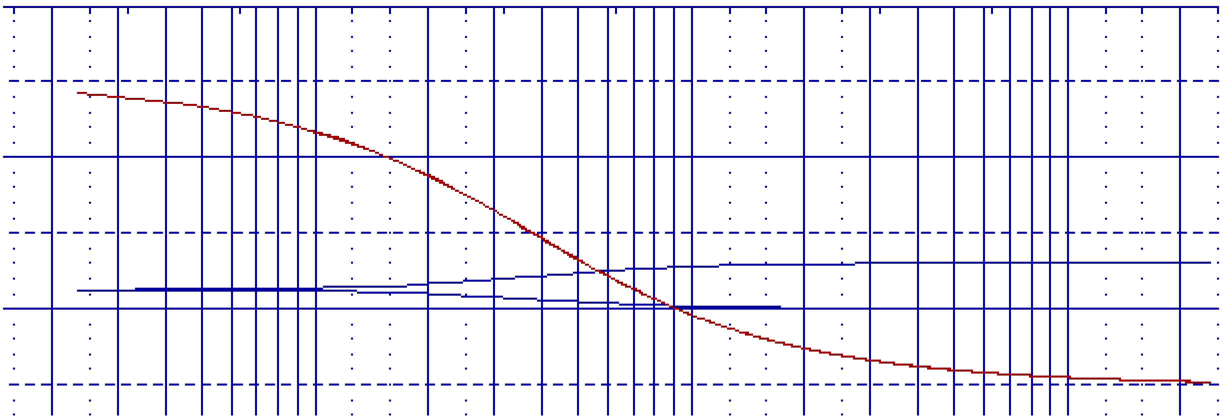


Fig 5.1 old Digital Shelf Filters. LF gain 1.25dB and pivot at 380 Hz Phase in Red : 180° at LF to 0° at HF

Because an FIR is limited in length, we have to look carefully at what this approximation does to our target function. Often we have to be very clever to hide these truncation or 'Windowing' effects and push them to where they are innocuous. Here we have it easy. Our 1025 linear frequency bins gives 2048 time samples. Halving the time sample and plotting phase & amplitude several times gives

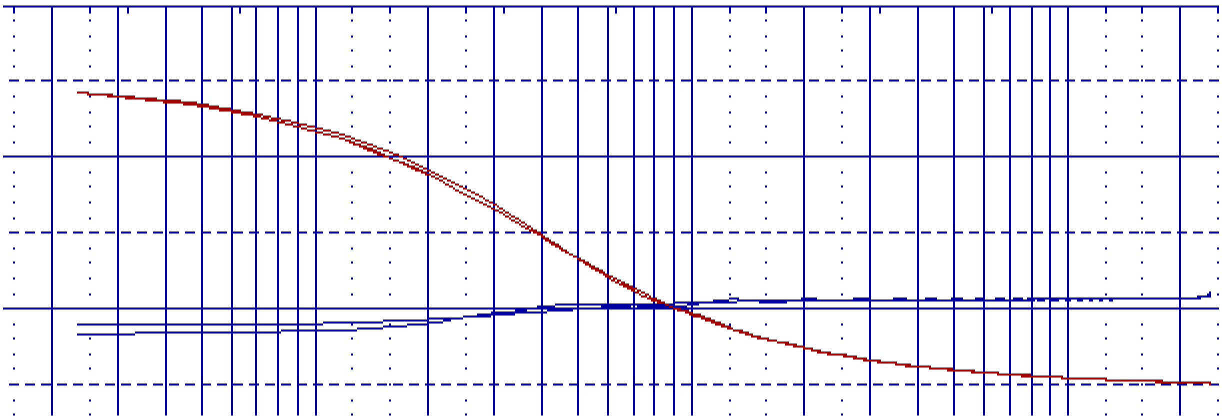


Fig 5.2 Digital W shelf filter 1024 to 64 taps. fs = 48kHz

Vertical resolution is 0.2dB and 1.8° At 128 samples, the phase curve just thickens in one spot but amplitude is still within 0.2dB No difference is seen for longer filters.

A **FIR of 128 samples** or longer is probably **good enough for the W filter** at **48kHz**. No further optimisation required. **256 samples** for the XY filters. Artifacts are below -96dBFS on the 1024 sample filter.

I'm not publishing these filters cos the work was done using the old specification.

6.0 Accuracy checks, truncation, dither ...

For each filter we hand carve from the solid bullshit, we spend a lot of time looking at its response and what happens when we truncate it.

You must be able to look at Phase & Amplitude critically with a large window FFT. Use Audacity, Audition (CoolEdit) or another wave editor to look at the waveform and blow it up to see what's happening below -96dB fsd.

Some of the (audible) windowing effects can appear in very subtle forms. eg the All Pass copies in section 5.0 were quite easy to do while the section 8.0 Linear Phase filters had to be tweaked for better accuracy with 'small' levels of truncation.

This tweaking, (application of Singular Value, LU Decomposition and other poly-syllabic algorithms to optimize in a Least Means Square sense ...) doesn't come for free. What happens is that you push the errors somewhere you hope they won't be seen or heard. In our case, we relax the tolerance on the DC gain for our Linear Phase filters.

Truncating a filter tells us many things besides how short it can be. eg the section 8.0 Linear Phase filters start showing artifacts slightly earlier than the 5.0 All Pass filters. However, the effects of severe truncation are less of a problem with the Linear Phase filters.

In the end, the Linear Phase filters are a more robust implementation if done properly.

It's vital to do this type of analysis when dreaming up FIR (and IIR) filters. I don't think you can automate Digital EQ for speakers or room IF YOU WANT THE BEST RESULTS ... even with the luxury of 32K bit convolution.

And don't forget dither. I'm almost certain most Windoz FFT fast convolution algorithms are not properly dithered. (but see http://www.fftw.org/ & http://www.ludd.luth.se/~torger/brutefir.html) Certain people under certain conditions can detect this in blind listening tests even on 'pop'.

7.0 IIR Minimum Phase Shelf Filters

Shelf filters implemented with an **IIR** should have the least computational load.

http://sparg.derby.ac.uk/SPARG/PDFs/eqdesign.pdf describes Shelf filters using Digital All Pass Networks. Their Figures 8.23, 8.25 and Figure 8.28 "Block diagram of the complete shelving equaliser" have an analogue equivalent.

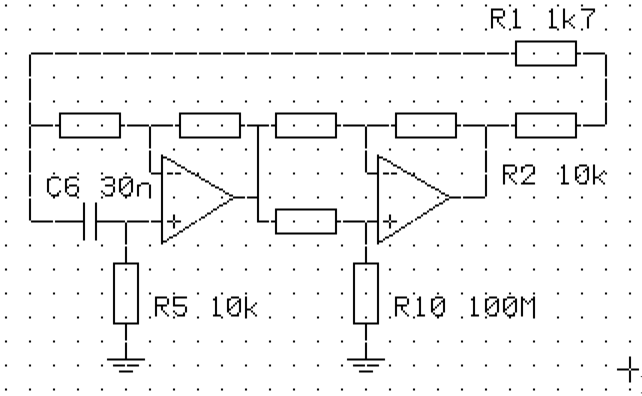


Fig 7.1 Analog of SPARG IIR Shelf Filter

R10 is either OC or SC to give HP or LP. R1 = 10k for a filter and 1k7 for 3dB shelf

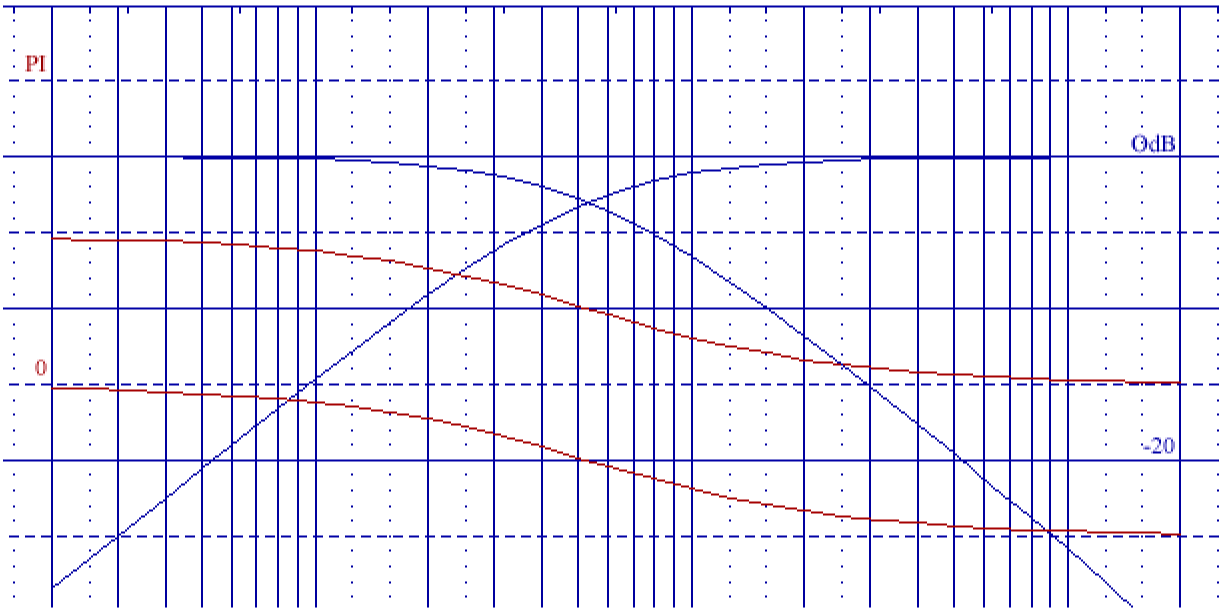


Fig 7.2 HP & LP filters implemented with Fig 7.1 network R1 = 10k

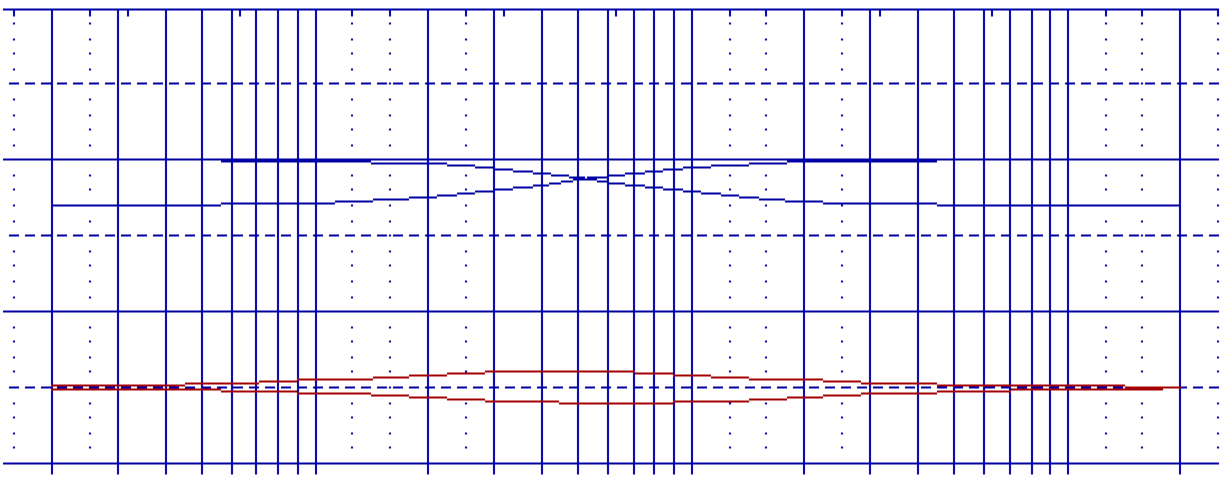


Fig 7.3 Arbitary 3dB Shelf Filters implemented with Fig 7.1 R1 = 1k7
Both the filters and the shelfs are **Minimum Phase** and not **phase matched**. The shelfs have about 18° difference at the pivot frequency. Horizontal B Format Shelf Filters, the equivalent of Fig 3.3, would have about 9° phase mismatch.

While this appears small, it is significant. Dr Geoffrey Barton of the original Ambisonic team comments

... *believe me you do not want to go that route* ...

So the method in the above **SPARG** document "**Parametric equalisers**" is **unsuitable** for Ambisonic Shelf filter sets. The phase in the W and XYZ sections don't match.

7.1 IIR All-Pass Shelfs with Phase Matching

It is possible to use **IIR Digital All-Pass Networks** in the same manner as the Analogue All-Pass Shelf Filters.

A simple Minimum Phase Shelf is a pole zero pair with s & z function (IIR filter) representations as below

pole zero pair $H(s) = (\mathbf{1 + sT1}) / (1 + sT2)$ becomes $\mathbf{H(z) = (b0 + b1\ z^{-1}) / (1 + a1\ z^{-1})}$ Eqn 7.1

$b0 = \dfrac{k2\ (\mathbf{k1 + 1})}{k1\ (k2 + 1)}$ $b1 = \dfrac{k2\ (\mathbf{k1 - 1})}{k1\ (k2 + 1)}$ $a1 = \dfrac{(k2 - 1)}{(k2 + 1)}$ Eqns 7.2

We obtain k1 & k2 by applying the Bilinear Transform which approximates s = jw

by $s = \dfrac{1\ (z - 1)}{k\ (z + 1)}$ where **k = tan(Pi Fc / fs)** **Eqns 7.3**
Fc : pole or zero frequency = 1/(2 π T)
fs : sampling freq

http://en.wikipedia.org/wiki/Bilinear_transform

An **All-Pass Shelf** , the equivalent of Eqn 3.1 is as below with the corresponding IIR filter coefficients.

All Pass $H(s) = (\mathbf{1 - sT1}) / (1 + sT2)$

$b0 = \dfrac{k2\ (\mathbf{k1 - 1})}{k1\ (k2 + 1)}$ $b1 = \dfrac{k2\ (\mathbf{k1 + 1})}{k1\ (k2 + 1)}$ $a1 = \dfrac{(k2 - 1)}{(k2 + 1)}$ **Eqns 7.4**

Note the small sign change for calculating the IIR coefficients from k1 & k2 compared to Eqns 7.2

I omit all scaling in the above discussion cos I expect only DSP gurus to read this section. Get the gains from Table 4.2 and Fig 4.3

Yus gurus will know there are important caveats in implementing the above. At LF, k1 & k2 are small and the IIR filter coefficients become nearly = 1.

An IIR may be self-dithering. I'd appreciate an email if anyone has done work or has any info on this.

This Shelf filter is **best if computing power is limited. Efficient & phase matched.**

7.2 IIR multi-band crossover Shelfs

Guru Fons points out there is another IIR arrangement which has advantages if more than 1 step is used as in 10.0 This is to treat LF and HF as you would LF & HF units in a loudspeaker. See his AmbiDec source code for a fiercely efficient implementation with Regalia-Mitra filters of what I describe here.

Certain speaker crossover circuits have matched phase at both LF & HF. We don't want the complex filters with high slopes cos Shelfs should be gradual.

The simplest arrangement and the most useful for us is as follows

$$LF(s) = \frac{1}{[\,1 + 2\,sT + (sT)^2\,]} \qquad HF(s) = \frac{(sT)^2}{[\,1 + 2\,sT + (sT)^2\,]}$$ **Eqns 7.5**

simple 2nd order sections which are -6dB at the xover frequency, w = 1/T rad/s

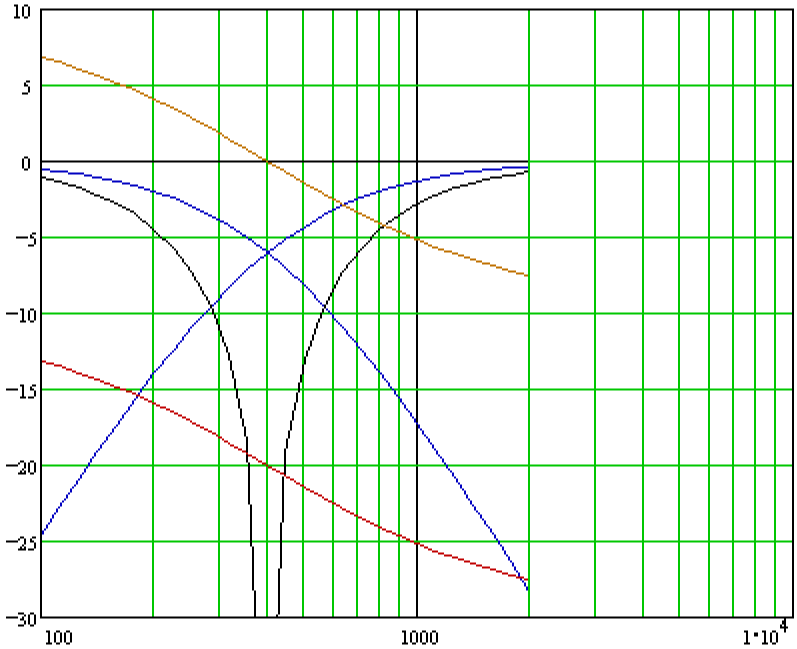


Fig 7.3 Xover type IIR Shelfs at 400Hz. Phase +/- π

The upper red phase curve is for the HF section, out-of-phase at LF and 0 phase at HF. The bottom LF curve has 0 phase at LF. If you combine these, you have a big suckout at the crossover frequency cos the 2 sections are out of phase (black curve). If you reverse the phase of the HF section, its phase now matches the LF section and there is no cancellation at the crossover frequency.

The output is $$\text{Total}(s) = \frac{1 - (sT)^2}{[\,1 + 2\,sT + (sT)^2\,]} = \frac{(1 + sT)\,(1 - sT)}{(1 + sT)\,(1 + sT)} = \frac{(1 - sT)}{(1 + sT)}$$ **Eqn 7.6**

ie a 1st order All Pass network as section 7.1 This phase response is as the LF section and is maintained **regardless of the relative levels** of the LF & HF sections.

Making the LF & HF sections different in amplitude gives our phase matched Shelf Filters. Somewhat easier to twiddle different Shelf heights at the expense of more complicated Bi-quads. But pivot shifts with Shelf height.

2nd order pole $H(s) = 1 / [\,1 + DsT + (sT)^2\,]$ http://www.earlevel.com/Digital%20Audio/Bilinear.html

becomes $H(z) = (b0 + b1\,z^{-1} + b2\,z^{-2}) / (1 + a1\,z^{-1} + a2\,z^{-2})$ **Eqns 7.7**

$b0 = k^2 / (k^2 + Dk + 1)$ $b1 = 2\,b0$ $b2 = b0$
$a1 = 2\,(k^2 - 1) / (k^2 + Dk + 1)$ $a2 = (k^2 - Dk + 1) / (k^2 + Dk + 1)$

2nd order HP $H(s) = (sT)^2 / [\,1 + DsT + (sT)^2\,]$ **Eqns 7.8**

$b0 = 1 / (k^2 + Dk + 1)$ $b1 = -2\,b0$ $b2 = b0$
a1 & a2 as Eqns 7.7

In each case **k = tan(Pi Fc / fs)** as Eqn 7.3

8.0 Type 3 Linear Phase filters

I'm suspicious of non-causal filters of which "Linear Phase" filters are the most common useful variant.

But from Fig 7.3, we see that the Minimum Phase filters have quite small phase response and will be quite short. The "Linear Phase" filters (which WILL be phase matched) will also be quite short. **Shorter than Section 5.0 Type 2 Digital FIR All Pass Shelf Filters** but not shorter than the Minimum Phase filters.

Linear Phase filters are favoured by Eric Benjamin of Dolby Labs. who says,

The first was implemented in CoolEdit's "FFT filter". You enter data points and it uses cubic splines to fit a filter function to your curve. I was able to get quite close <0.1 dB) to the magnitude response of s-domain shelf filters. It's implemented in 32-bit arithmetic so it's quite precise.

The next step was to derive a linear phase version from the IIR shelf filter by taking the tail of the Impulse Response of the shelf filter, reversing it in time and prepending it to the IR. This gives a linear phase version of the filter, but with twice the attenuation. So starting with a filter with half the attenuation gives the desired result. I do the actual
filtering with a shortened version of this filter, and a DOS filtering utility written by my friend Mark Davis.

As we now have to twiddle the results, we might as well try to meet the exact specification. From Table 4.2, the longest filter will be for W as the step is 1.76dB

First we derive some Minimum Phase Shelf filters just to get our amplitude response. We won't actually use them.

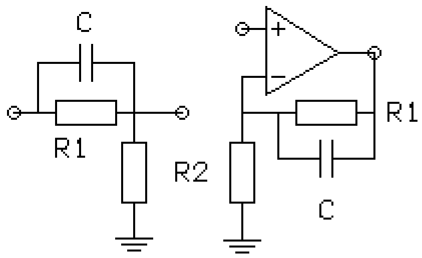


Fig 8.1 Minimum Phase Shelf filters for W & Vel R2 = 10k with transfer functions

[R2 / (R1+R2)](1 + sCR1C) / [1 + sCR1R2 / (R1 + R2)] and its inverse

and pivot frequency $2 \pi F = 1 / (C\ R1 sqrt[R2 / (R1 + R2)])$ **Eqn 8.1**

	Shelf	R1	C		Shelf	R1	C			
W Shelfs	0.8167	2k246	206n3	Vel	1.1548	1k548	290n8	-1.76	&	1.25dB

We use these to generate the correct Amplitude response *.FRQ for our Linear Phase filters.

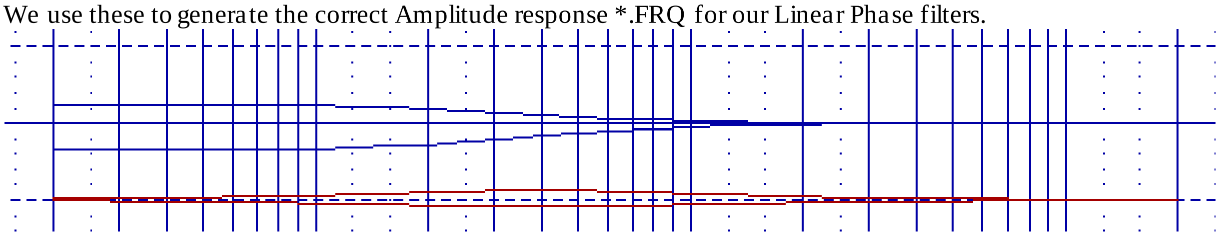


Fig 8.2 Response of Minimum Phase Shelf filters

Take the complex log magnitude of the *.FRQ file and zero the phase. Convert back to a symmetrical Impulse Response and truncate several times to check efficiency.

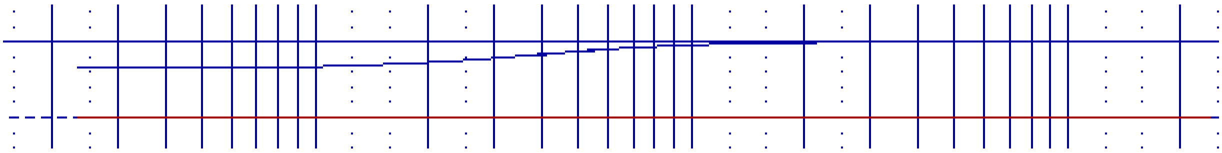


Fig 8.3 "Linear Phase" W Shelf filters 2048 to 128 taps.

shows amplitude just exceeds 0.2dB for 128 taps. Phase within 1° of zero. Reference time = 0 is at the maximum of the Impulse Response. The Impulse response concentrates much more energy at time = 0 compared to Section 5.0 filters. To turn these FFT blocks into FIR filters, we shift time = 0 to the middle of the block.

XY48.wav, W48.wav, XY44.wav & W44.wav have **time = 0 in the middle** of the **2048 sample** file. For shorter filters, truncating these is more accurate than calculating from scratch. Truncation should be from both ends to maintain symmetrical "Linear Phase" response. Look at the WAV files with a waveform editor to clarify.

XY48sht.wav, W48sht.wav, XY44sht.wav & W44sht.wav are the short, truncated 128 pt filters. Fig 8.4 shows their amplitude response is just over 0.2dB from the 2048 pt filters. Note the phase cycling cos time = 0 is now at sample 64, the middle of the filter impulse response. But the phase is matched to better than 1°

These filter sets are more accurate and may sound better than Section 5.0 filters as they do less damage to the signal. *Section 7.1 filters can be more efficient*. Use 256 samples for better than 0.2dB accuracy. Deterioration when truncated further eg with 64 samples is still acceptable compared to Section 5.0 filters.

I still have an irrational suspicion of the sound of non-causal filters except for anti-aliasing.

These are non-causal symmetrical filters so both **must be same length** unless you use a delay on the shorter filter.

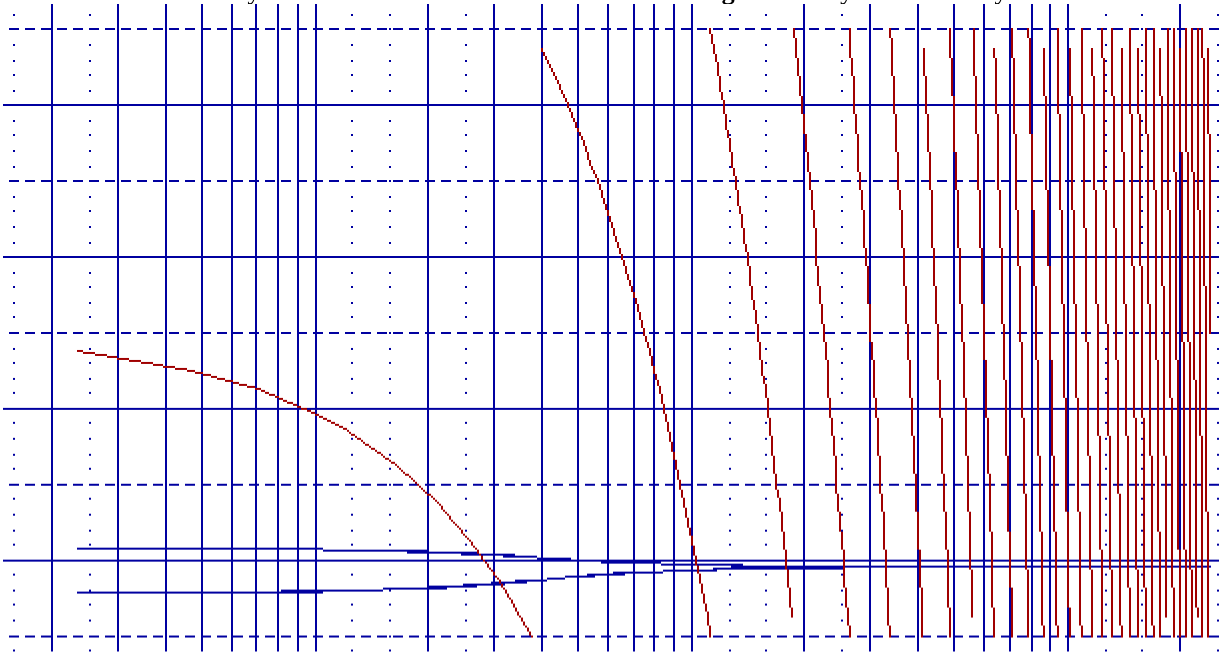


Fig 8.4 48kHz "Linear Phase" Shelf Filters truncated to 128pt. Amplitude of 2048 pt filters superimposed.

9.0 Simple G-Format Nimbus 4.0 encoder & Square Speaker Layout Ambisonic Decoder

These are one and the same. For Nimbus 4.0, the Ambisonic Surround Decoder generates speaker feeds for a square layout and sends them to the LF RF RB & LB inputs of a Dolby Digital or other good surround encoder.

W X Y Horizontal B format

> Convolve these with **Shelf Filters** W48.wav and XY48.wav (for 48kHz) from Section 8.0 Dither properly
> Or use a Section 7.1 All-Pass IIR.

Apply **Distance Compensation** to X' & Y'. Simple IIR filter with response sT / (1 + sT) see Section 4.2

W' X' Y'
The **Rationalised Square Speaker Matrix Decoder** Eqn 4.2 turns these into

LF RF RB LB Speaker Feeds

All operations properly dithered where necessary.

This simple system is probably still 'state of the art' for horizontal surround sound on 4 speakers. http://www.ambisonicbootlegs.net/Members/ricardo/ASD.htm/#Real suggests it is likely to outperform most 5.0 systems. It is also very easy for the user to set up and to get the best out of it.

Just arrange the speakers in a square and sit in the centre.

It degrades gracefully. eg

- if no Shelf filters are used, it should still give better results than most 5.0 systems in a domestic environment.
- it's not necessary to get Distance Compensation exactly right.

These benefits are very important for Nimbus 4.0 and someone trying Ambisonics for the first time.

10.0 2 STEP SHELFS *This bit is experimental (feb07) so don't bother looking for 2W44 etc.*

Work on Option 1, a dipole stereo speaker, optimized for larger "sweet spot" suggests it may be beneficial to have a further stage of Shelf filtering taking the decode to "Cardioid" above 5kHz.

Loudspeakers and the Stereo Image - Gareth Millward, Hi-Fi News & Record Review nov84

Table 10.1 RATIONALISED 2 STEP SHELFs	below 380Hz	MF	above 5kHz
Horizontal W	0.8167x -1.76dB	0dB	1.155x 1.25dB
XY	1.155x 1.25dB	0dB	0.8167x -1.76dB

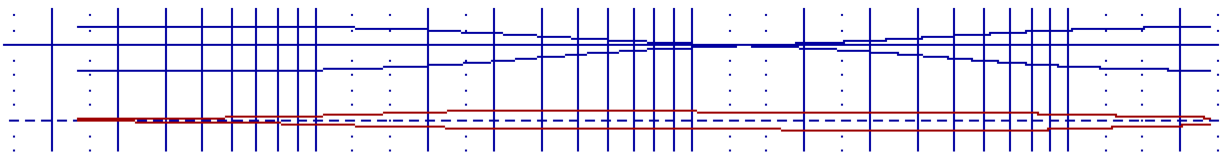


Fig 10.1 2 step Minimum Phase Shelfs centred at 380 & 5k Hz. Exact

Maximum phase for W is 6.4° and for XY -6.5°. The maximum difference 11.4°

The **Linear Phase** filters are 2W44.wav 2XY44.wav 2W48.wav 2XY48.wav and were dreamt up in a similar fashion to W44.wav XY44.wav W48.wav XY48.wav in Section 8.0

They are accurate to better than 0.1dB and 0.1 degree phase.

The filters do not have an overall gain of 1.0x cos the W impulse responses will exceed 1.0 which causes problems with programs like Audacity 1.2.3 which expect 32fp WAV files to have maximum amplitude of 1.0

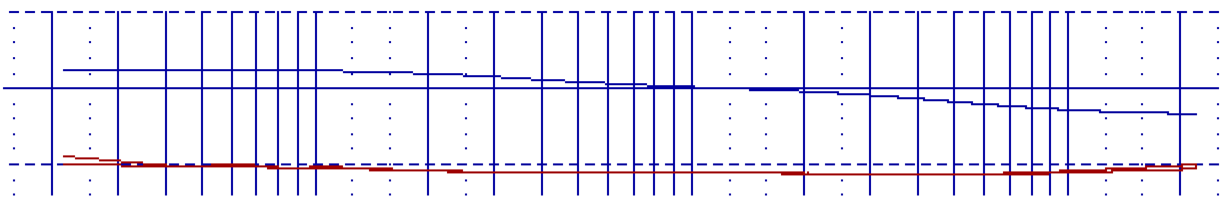


Fig 10.2 Response & Phase of target 2 step **Minimum Phase 44kHz XY filter** and 2048 pt FIR approximation

Response is within 0.1dB of the target but the phase differs from true minimum phase by 7.2° at 21.53Hz and 3.6° at 22.05khz. A more sophisticated method for Minimum Phase could reduce the error at LF but at the Nyquist frequency, the phase of an FIR filter must be 0° or 180°.

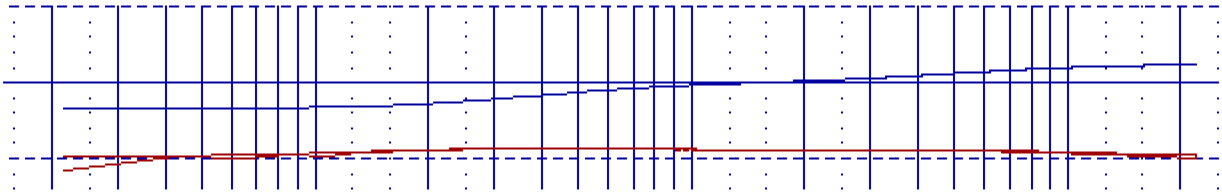


Fig 10.3 Response & Phase of 2 step **Minimum Phase W filter** 10dB & 90 degrees between dashed lines

The W filter is 12.6° out at 21.53Hz. The biggest phase difference between these 2 filters is at LF; about 19.8°.

Need to change all Shelfs when 2W44 2W44min 2W48 2W48min uploaded ************************

Appendix A : Final filters are 2048 or 128 x 32b FP samples in mono.WAV files. Scaled to 1.0 max. See the individual sections for truncation issues.

The CANONICAL WAV format abandoned by Microsoft. Many older programs eg Audacity & Audition (CoolEdit) will read, and some will write this format. Now an uncontrolled standard, some recent programs will refuse to read it; notably Windoz Media Player. I use this cos I dunno how to write a proper WAVE_X file.

The 44byte header is strictly legal only for 16b PCM. I use fmt_tag (byte20) = 3 to specify 32b FP like some versions of Audacity & CoolEdit. The rest of the format is described in

http://www.lightlink.com/tjweber/index.html I'm not specifying any integer coefficients cos

- FIRs filters MUST be dithered to sound good. If you do all your processing in 32b FP, there **might** be sufficient extra bits so that when the 32b FP signal finally goes to your (dithered) D/A there's a chance that the signal will be properly dithered when it gets reduced to 16b or so.
- Any non trivial DSP operation needs to be dithered. An FIR done "old fashion" definitely. Also an FFT block convolution but this is complicated and really needs dithering at each butterfly stage. An integer FFT loses precision AND needs dither at each stage even with careful scaling.
- Using 24b integers MAY alleviate this but the arithmetic becomes too hard for me. 32b FP coefficients have about 24b precision so you may scale and convert my coefficients if you are a proper 24b integer DSP guru. Small inaccuracies in the coefficients have little effect on sound unlike the lack of proper dithering.

CANONICAL WAV 44byte header for W48.WAV & XY48.WAV

```
struct  wavhead
{
char    rif[4];
long int filesz_8;              /* file length - 8 length of rest of file */
char    wav[8];                 /* fmt chunk */

/* byte 16 */

long int fmt_len;               /* length of 16b fmt chunk */
int     fmt_tag;                /* 1:PCM  <>1:compressed ?  3:fp? */
int     channels;               /* stereo:0x02  1:mono  ?:multi */
long int sample_sec;            /* 44100: 0xAC44  4800:0xBB80 */
long int bytes_sec;

/* byte 32 */

int     bytes_frame;            /* channels * bits_sample / 8 */
                                /* block_align ed frame size  */

int     bits_sample;            /* 16b:0x10  20b:0x14  24b:0x18  32fp:0x20 */

/* data chunk */

char    data[4];

/* Next is long int datasize then Data starts at byte 44 */

}  wav44hdr = {
        "RIFF",
        0x2024,         /* file size - 8 ie length of rest of file */
        "WAVE"
        "fmt ",         /* fmt chunk */

        0x10,           /* length of 16b fmt chunk */
        0x03,           /* 1:PCM  <>1:compressed ?  3:fp? */
        0x01,           /* stereo:0x02  1:mono  ?:multi */
        0xBB80,         /* sample_sec 44100:0xAC44  48000:0xBB80 */
        0x02EE00,       /* bytes_sec = sample_sec * bytes_frame */

/* byte 32 */

        1 * 32 / 8,     /* bytes_frame = channels * bits_sample / 8 */
                        /* block_align ed frame size  */

        0x20,   /* bits_sample 16b:0x10  20b:0x14  24b:0x18  32fp:0x20 */

/* data chunk */

        "data",
```

```
};
```

History

5mar06 1st publication on Etienne's site
2jul preface to **2nd Ed**. **Correct rectangle Eqn 4.4** etc Change Eqn 4.1 to polygon
22sep **Aaron Rationalised Rectangular decoder**. 'In head' replaces 'phasiness'. **IIR All Pass Shelfs & HP**
25dec06 mention Periphonic Shelfs are larger. 2 step Shelfs in Section 10.0 AH EB
add Min Phase 2 step Shelfs AH EB
7feb07 4th Ed to ambisonicbootlegs.net
17mar07 corrected typo in Eqn 4.4
14apr08 change to NFC. Start on xover Shelfs. To ambisonia.com, Eric & Aaron. Needs tidying.
26may08 corrected NFC Eqn 4.6