

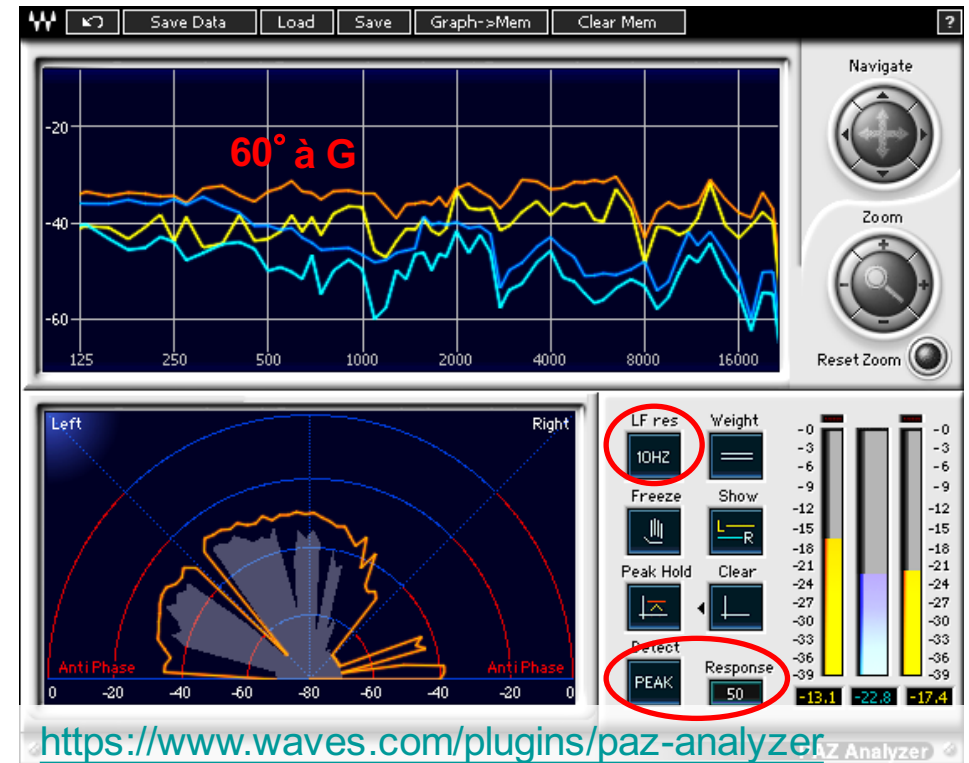
# **COMPARATIF DE PLUG-INS POUR "BINAURALISER" UN SON MONO OU STÉRÉO**

**BERNARD LAGNEL**  
**Ingénieur du son Radio France**  
**Février 2018**

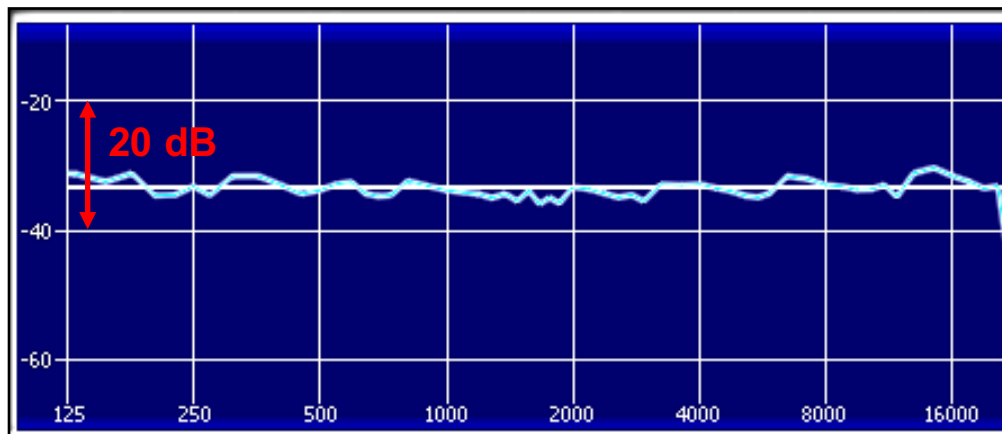
## **BINAURALIZER de NOISEMAKERS**



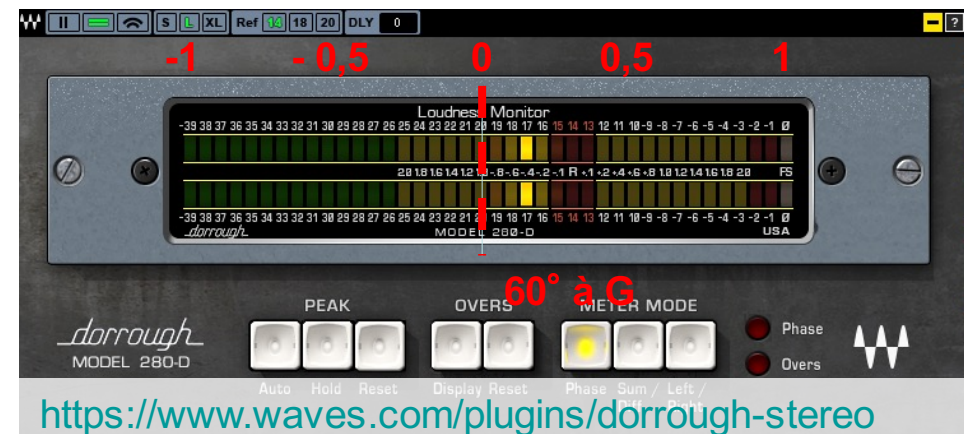
## **OUTPUT : PAZ de WAVES pour l'ILD**



## **INPUT : BRUIT ROSE**

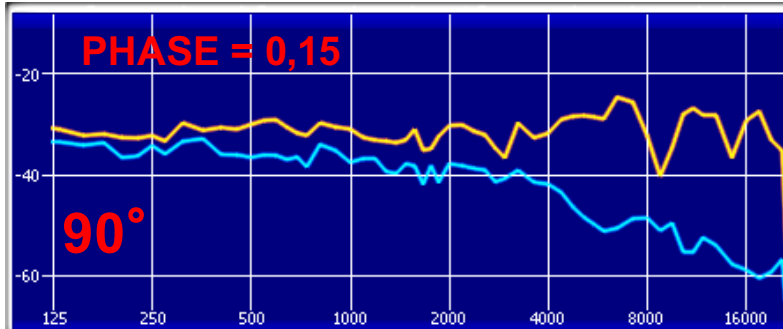
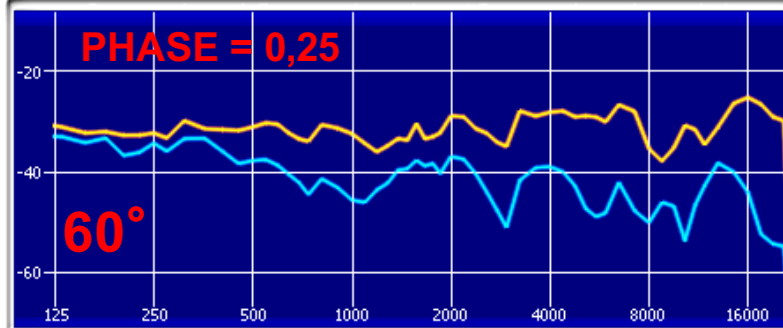
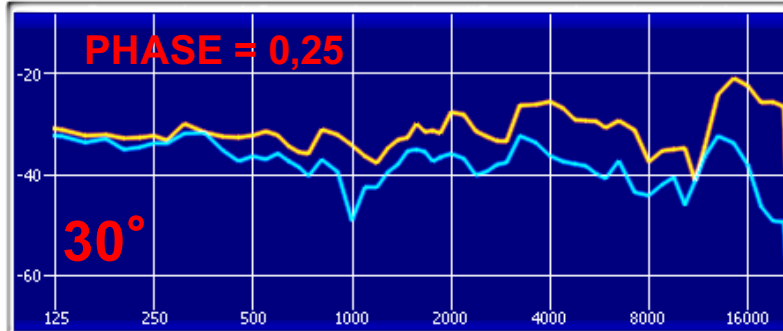
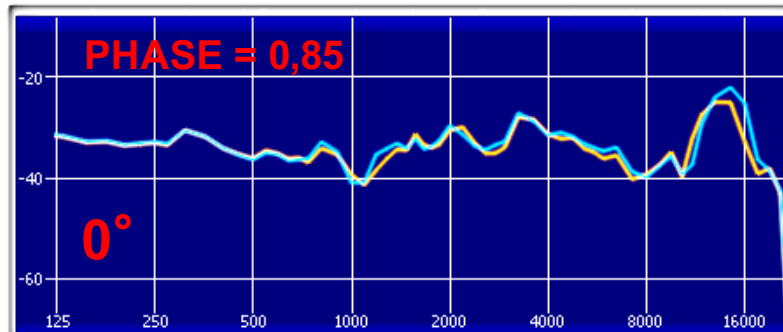


## **OUTPUT : Dorrough de WAVES pour la PHASE**

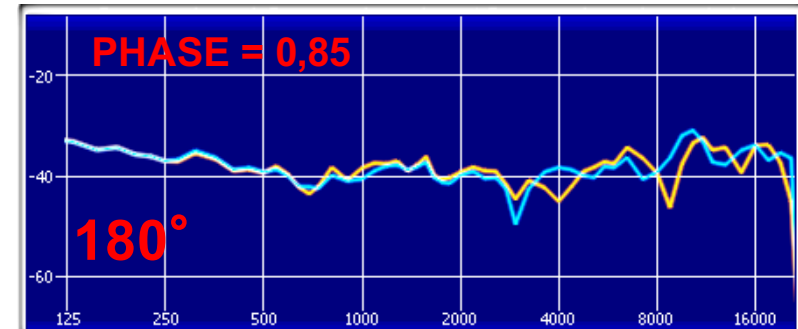
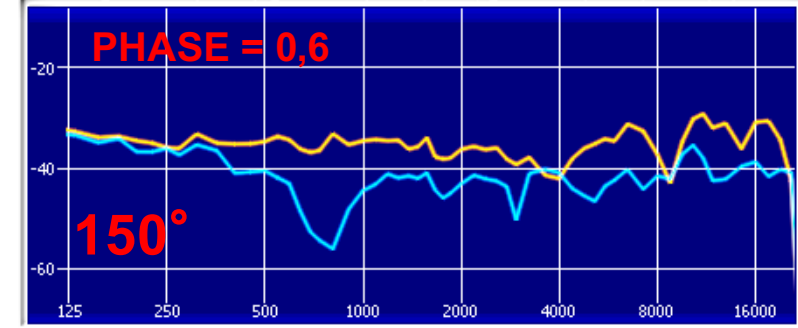
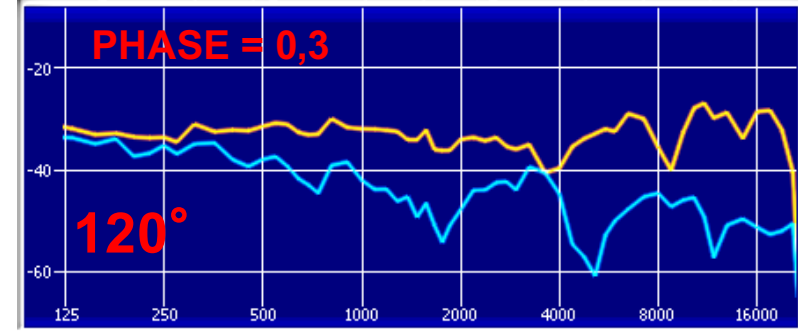
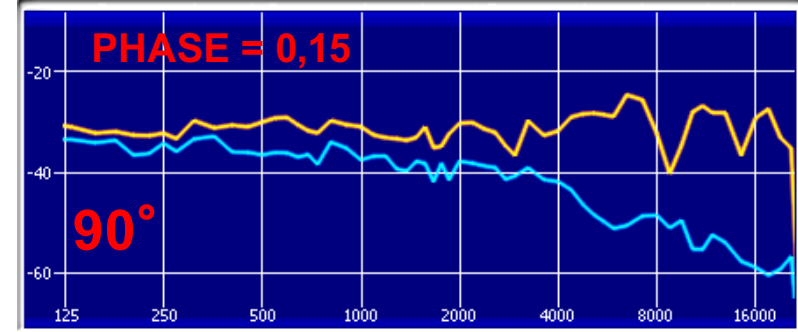


# PLUG-IN **HRTF 26** utilisé à Radio France

— Oreille Ipsilatéral  
— Oreille Contralatéral

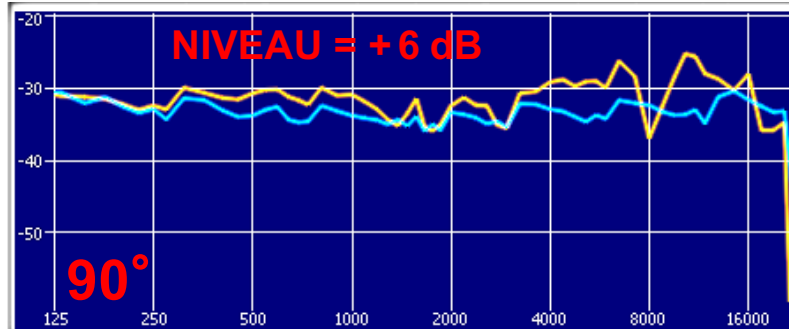
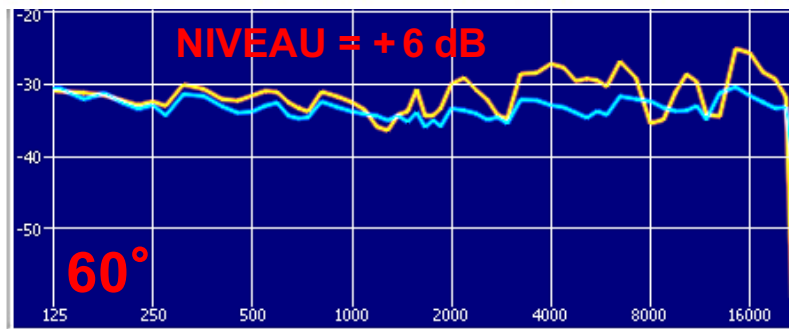
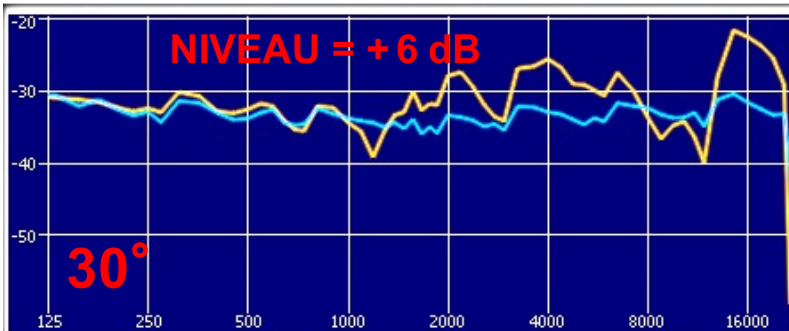
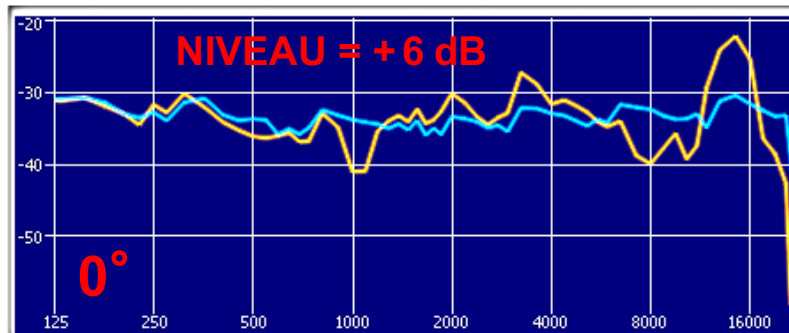


IRC\_512\_Meth5\_26\_48000.sofa

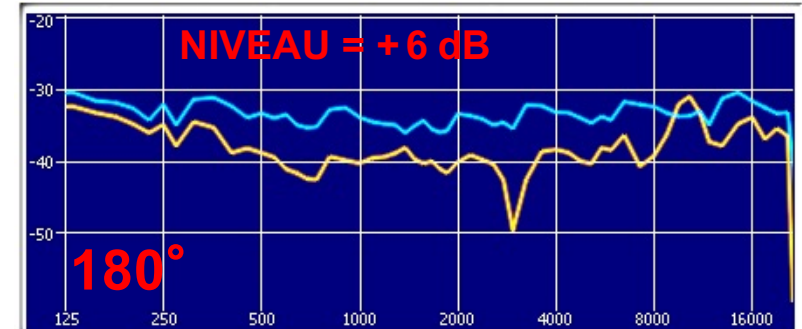
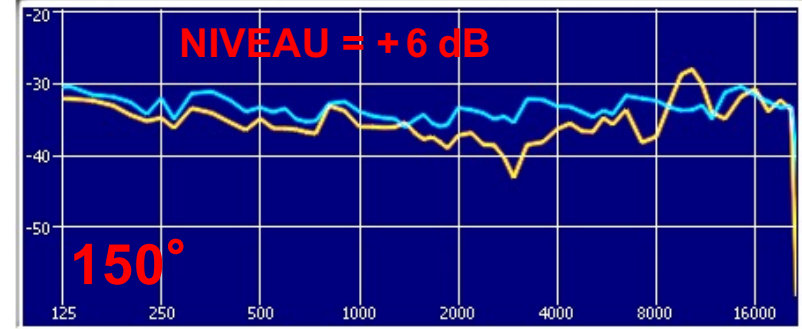
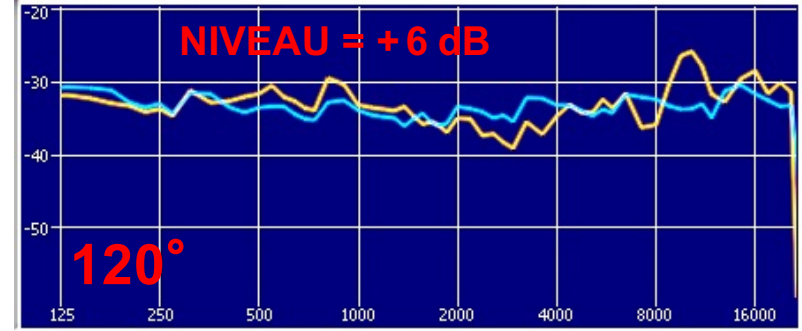
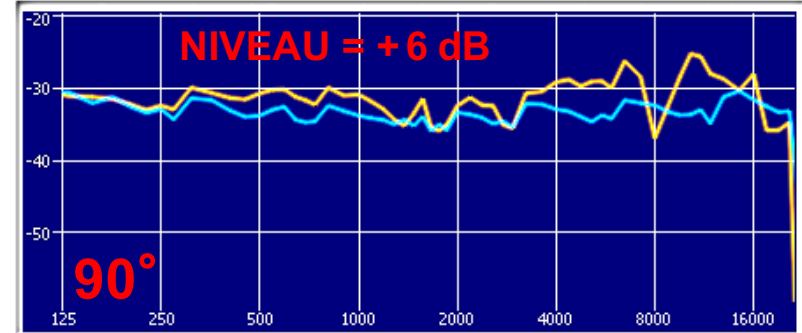


# HRTF et Bruit Rose : **RF 26** *Radio France*

— Oreille Ipsilatéral **L**  
— BRUIT ROSE IN



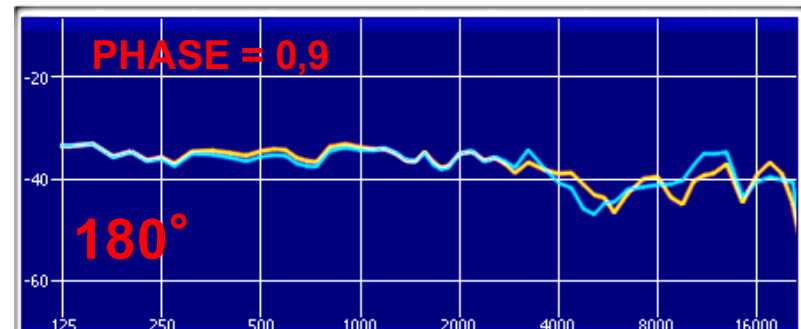
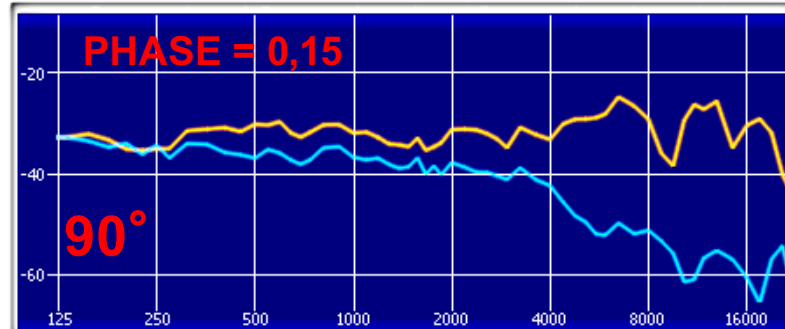
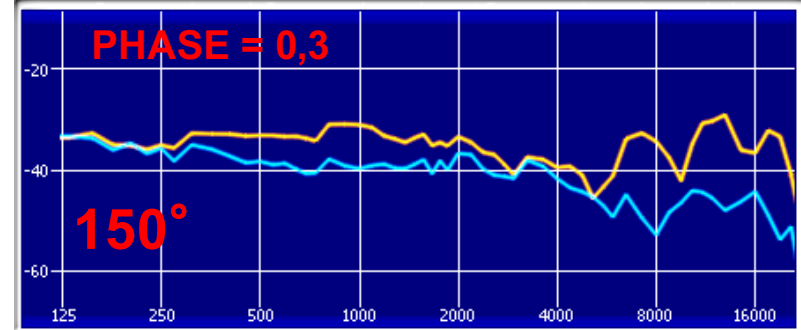
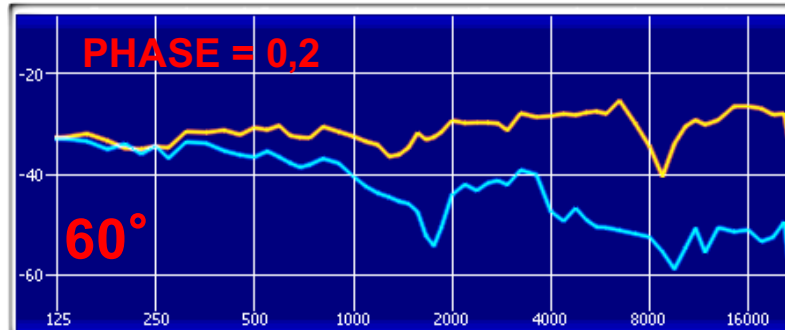
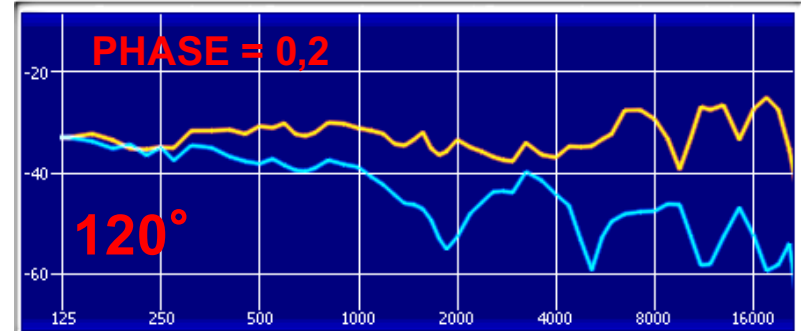
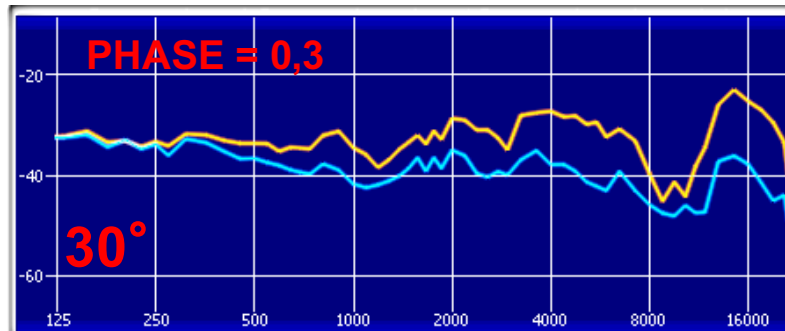
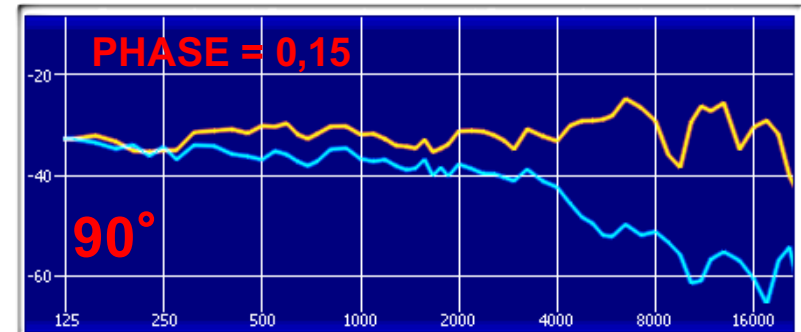
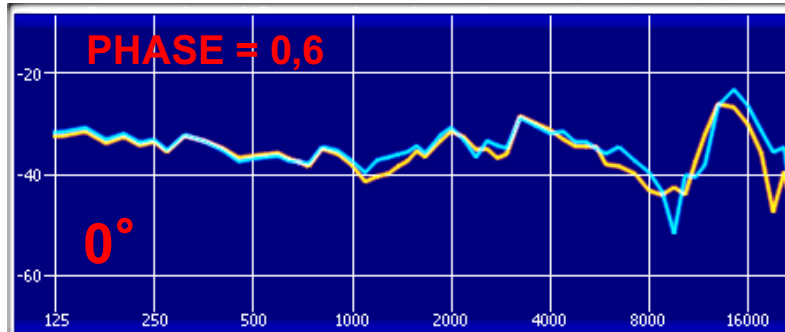
IRC\_512\_Meth5\_26\_48000.sofa  
Octobre 2015





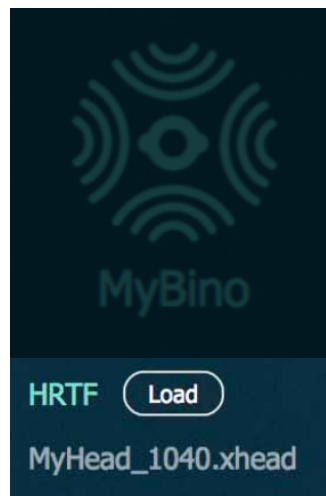
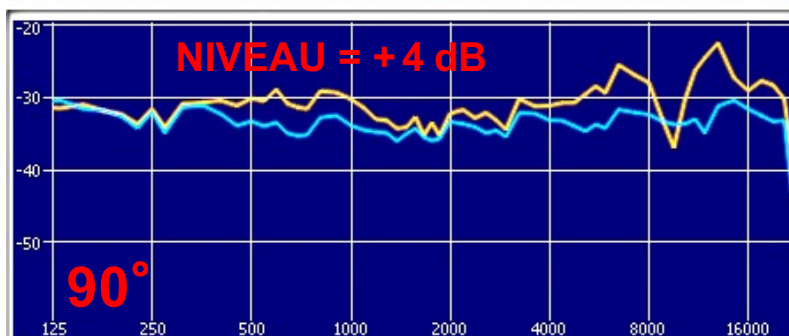
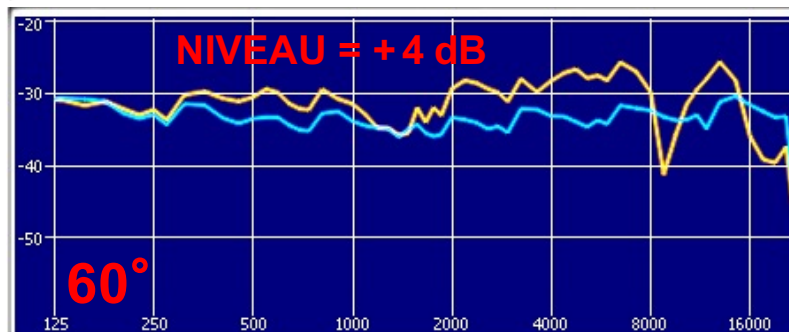
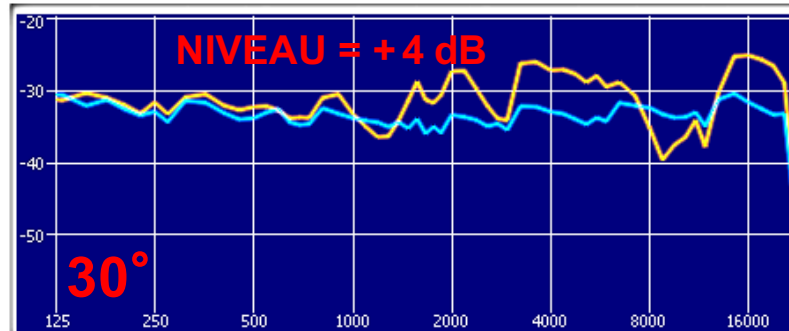
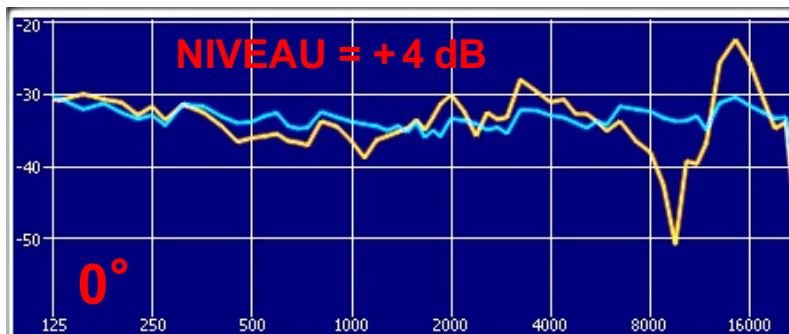
# PLUG-IN *MYBINO*

MyHead\_1040.xhead



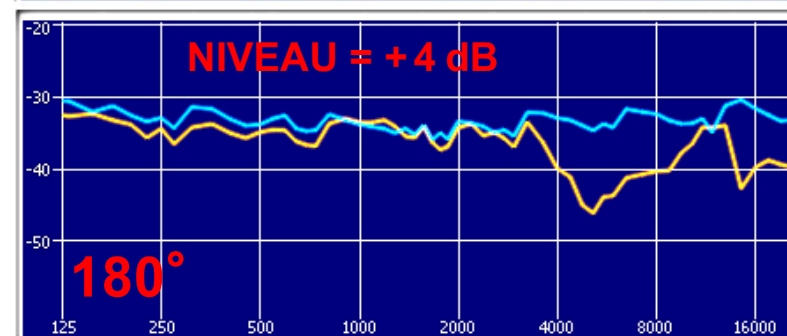
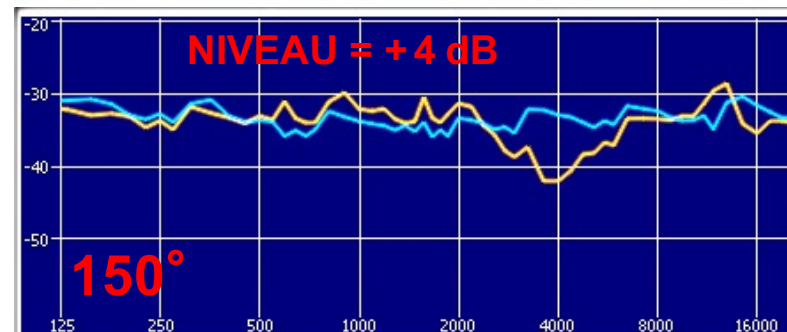
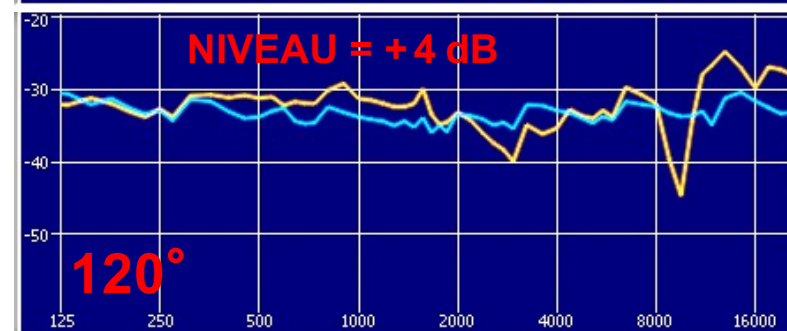
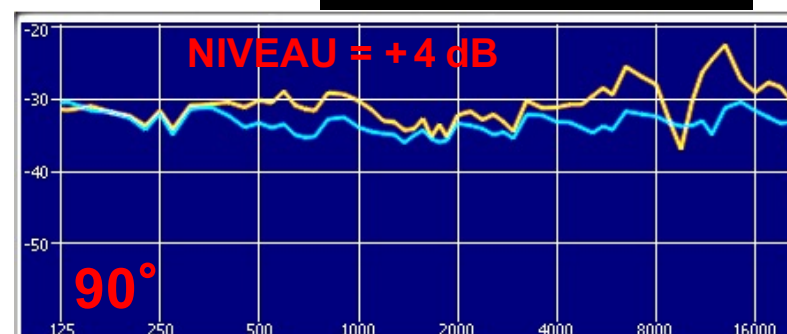
# HRTF et Bruit Rose : **MY BINO**

— Oreille Ipsilatéral **L**  
— BRUIT ROSE IN



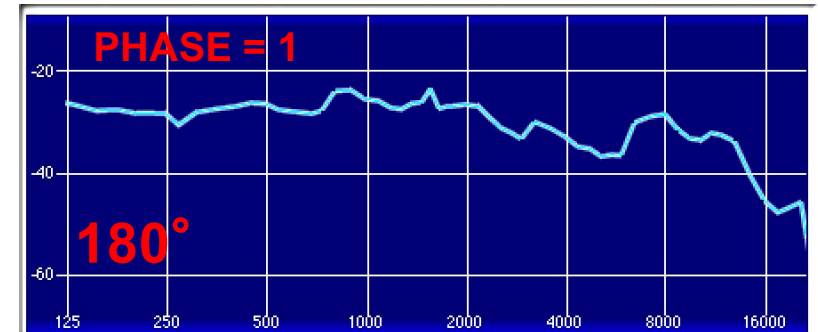
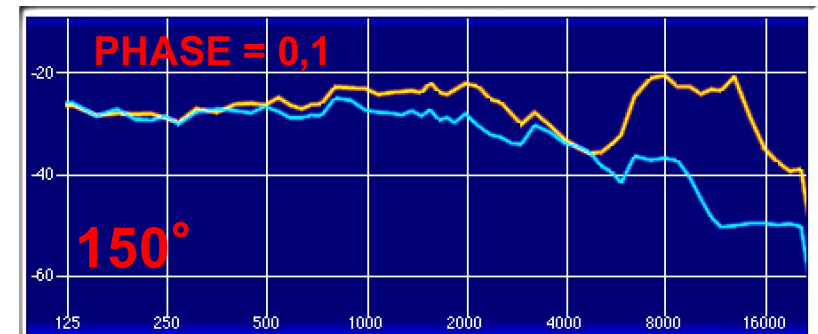
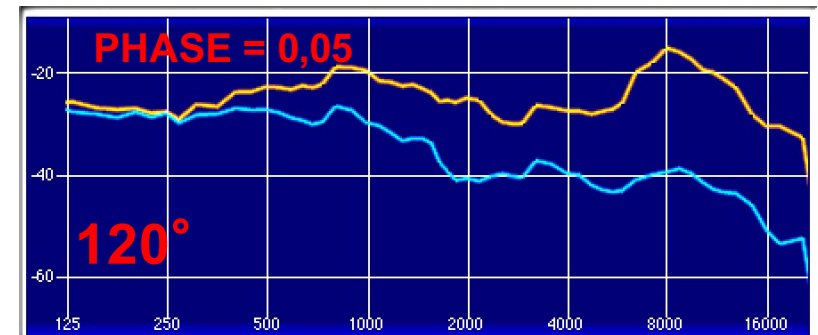
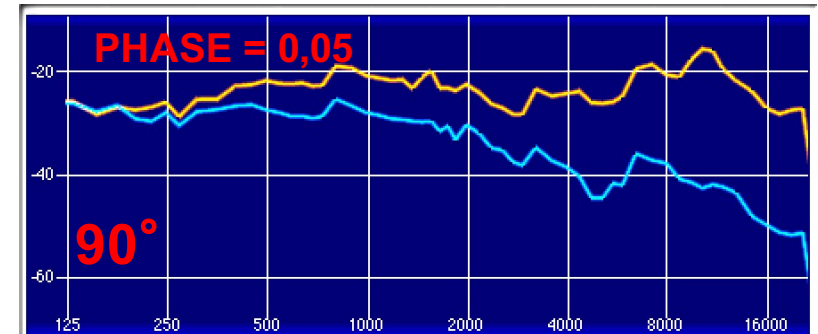
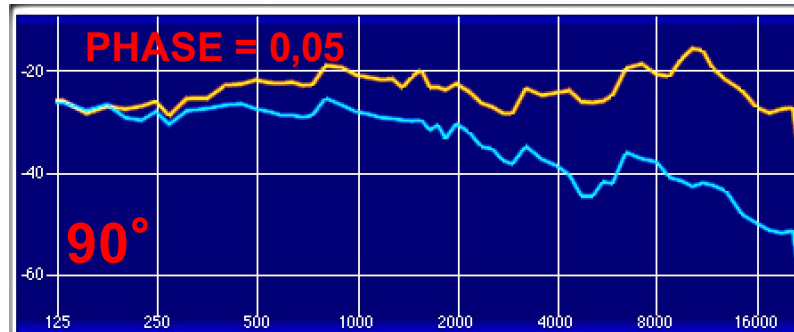
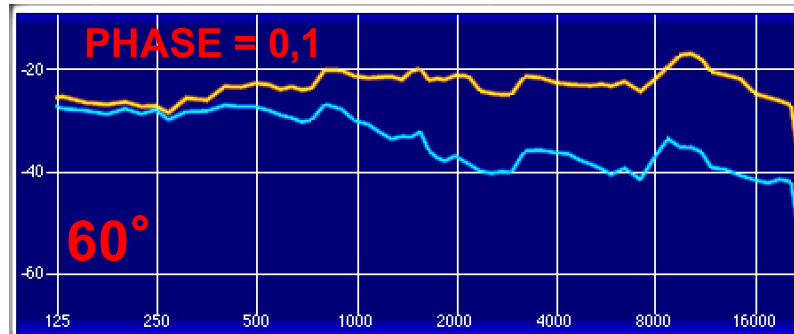
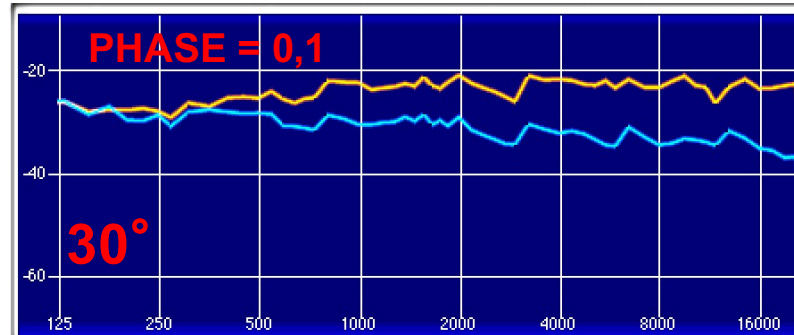
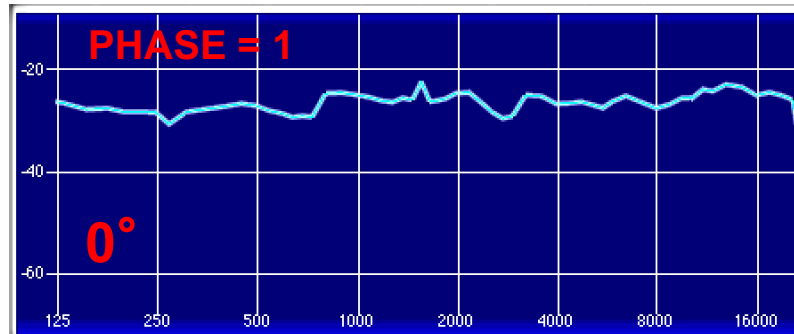
X-AUDIO

MyHead\_1040.xhead



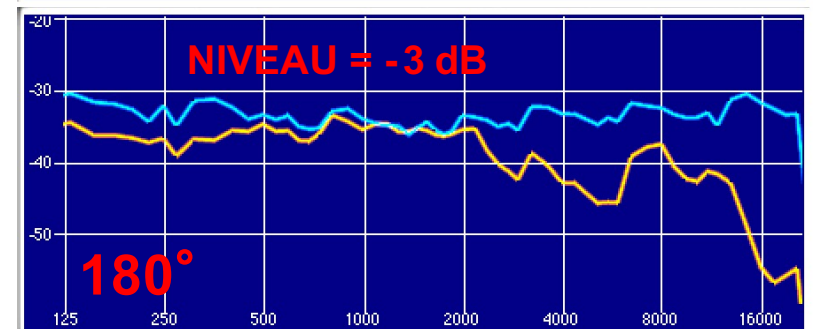
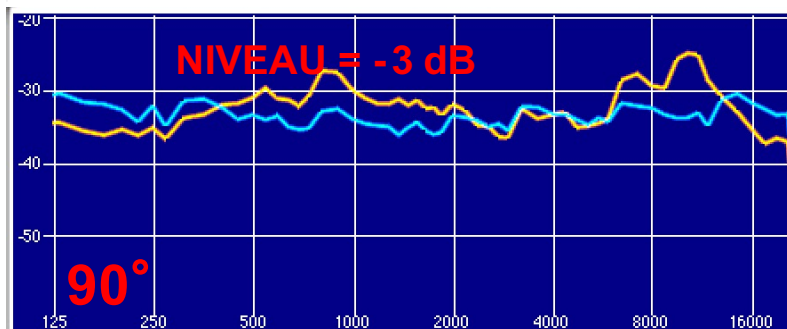
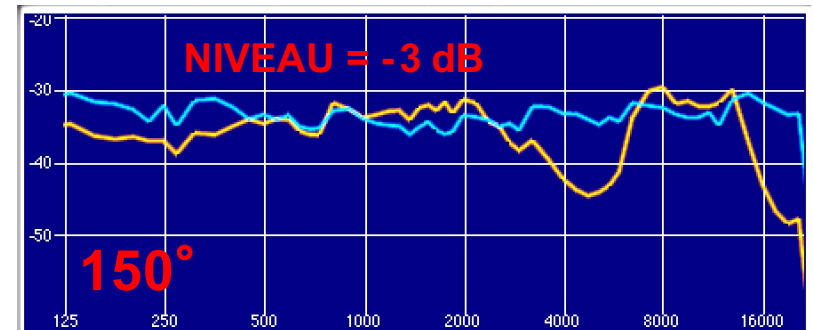
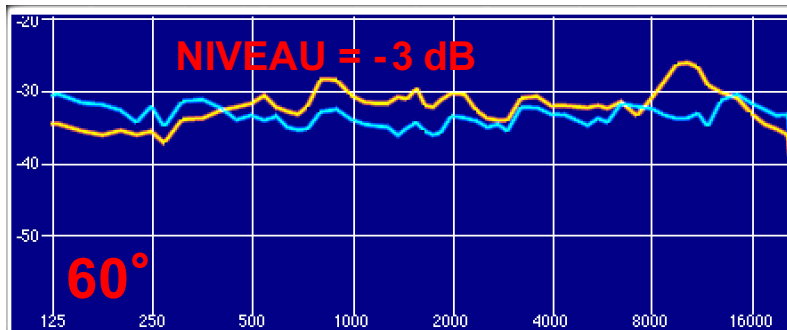
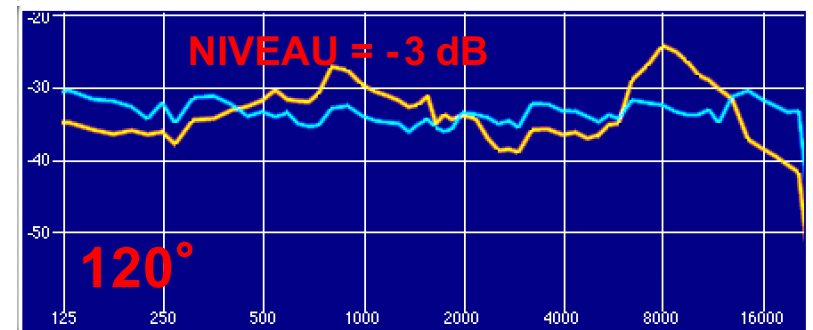
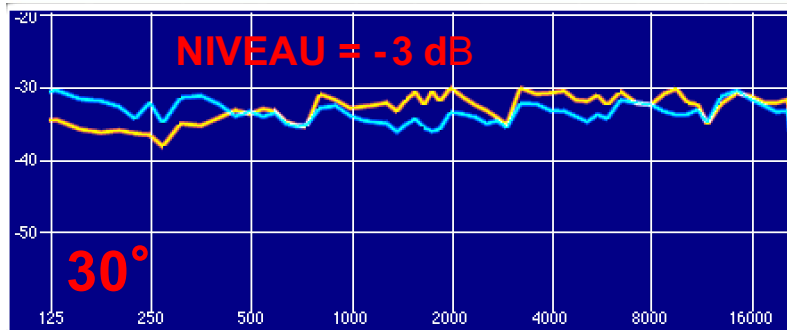
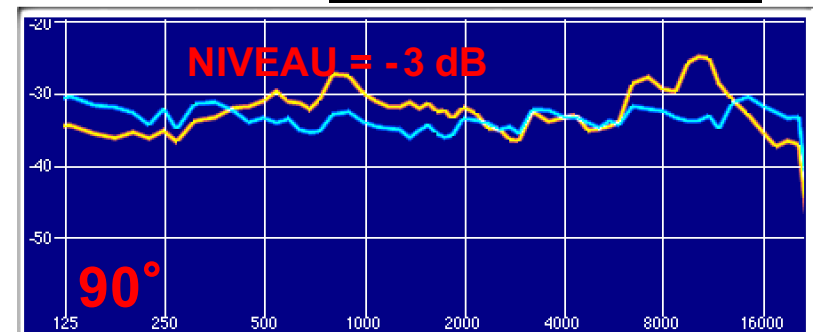
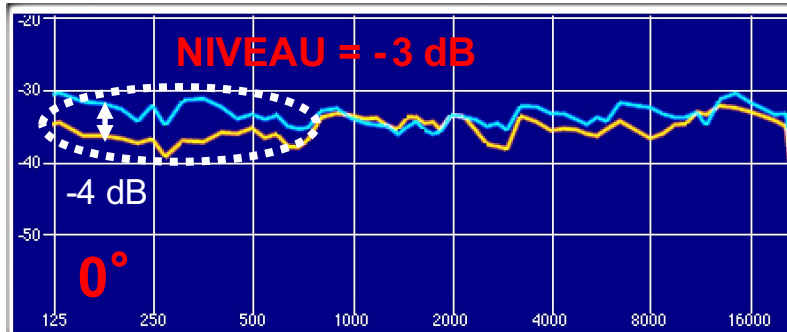
# PLUG-IN ***NX*** WAVES

NX 9-91-63-8 Build E 122366 (Cocoa)

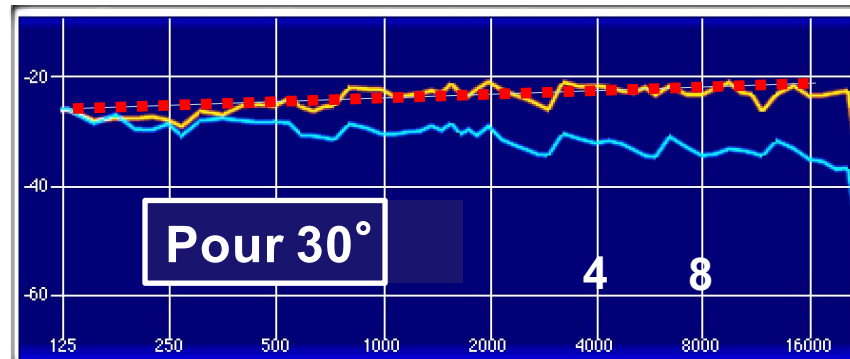


# HRTF et Bruit Rose : **Nx** de **WAVES**

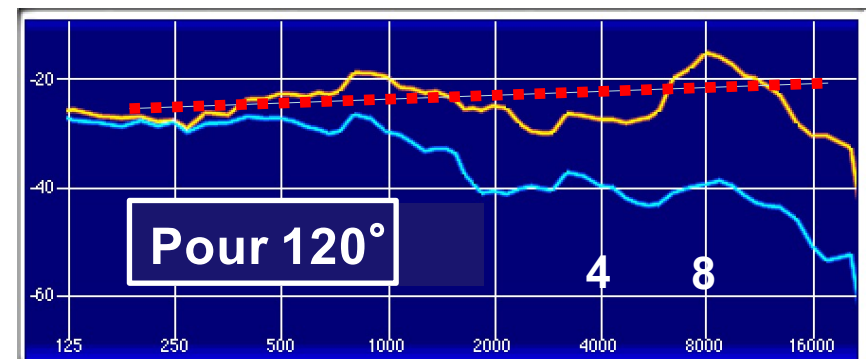
— Oreille Ipsilatéral L  
— BRUIT ROSE IN



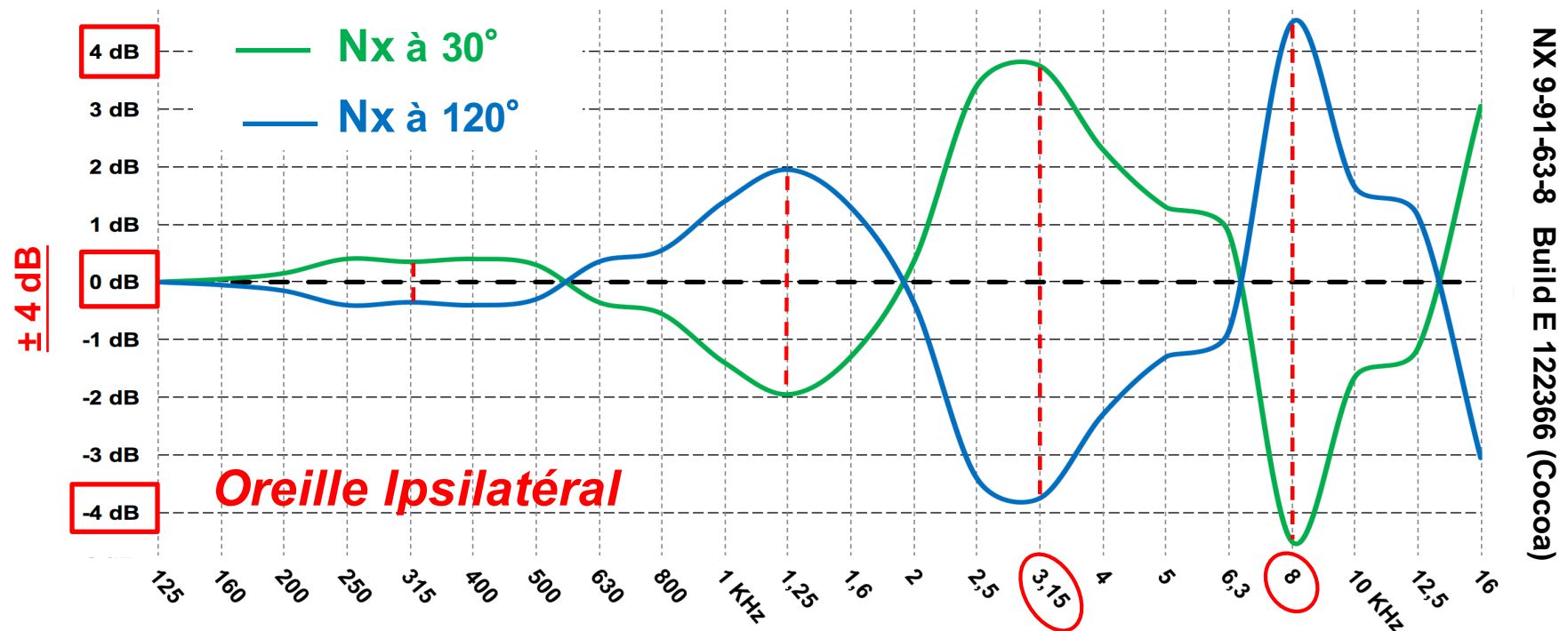
# Le Plug-in **Nx** de WAVES:



**En 5.0**



## Corrections pour « **binauraliser** » en **ILD** du multicanal **5.0**:





# En **1992** : **NEUMANN KU 100** (≈ 7400 €)

*BATT. pile 9v -- P48. fantôme 48v -- EXT. secteur 220v.*



Les Oreilles sont

***anthropométriques*** (G et D identiques pour les mesures).



(Sorties : 1 XLR 5 + 2 BNC)

2 micros du système **KM 100** (circuit de sortie sans transfo)

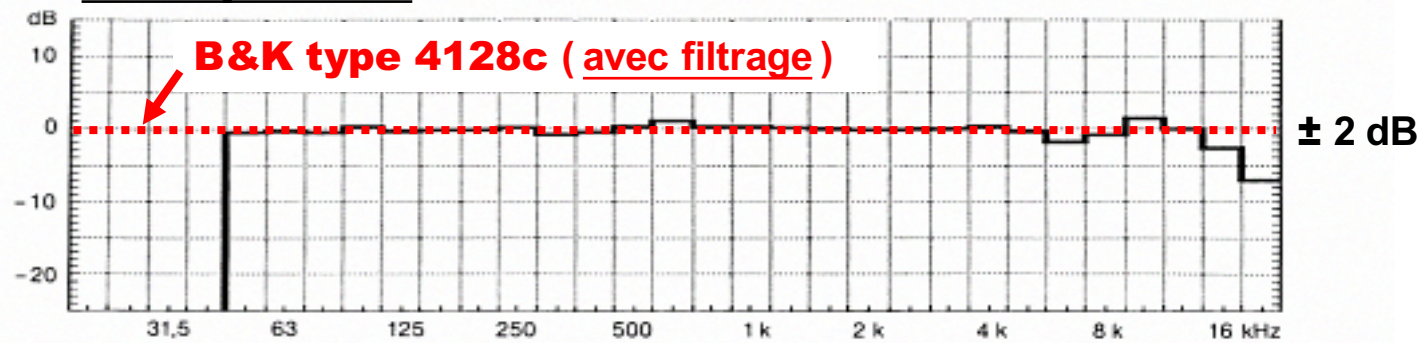
- Tête égalisée pour une réponse linéaire en champ diffus.

**↳ Compatible avec haut-parleurs**

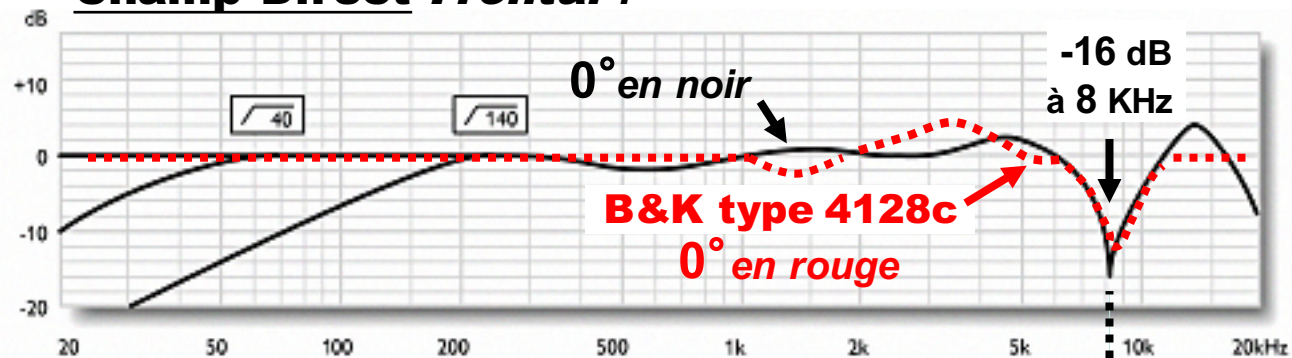
- Filtre coupe bas à 40 Hz ou 150 Hz et atténuation de 10 dB.

# NEUMANN KU100

## Champ Diffus :

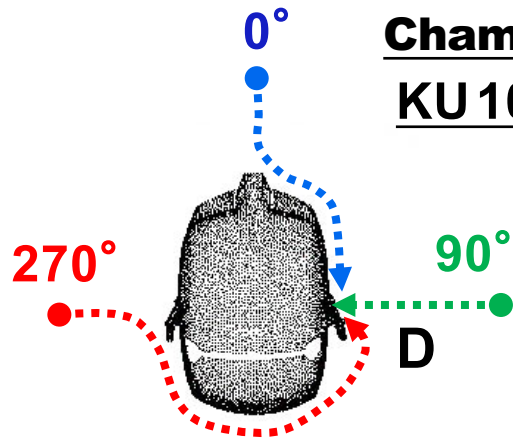
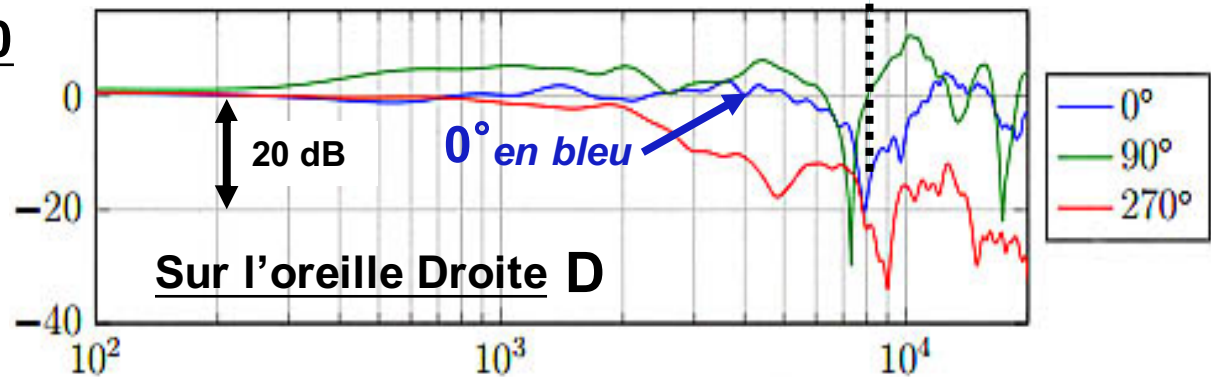


## Champ Direct Frontal :



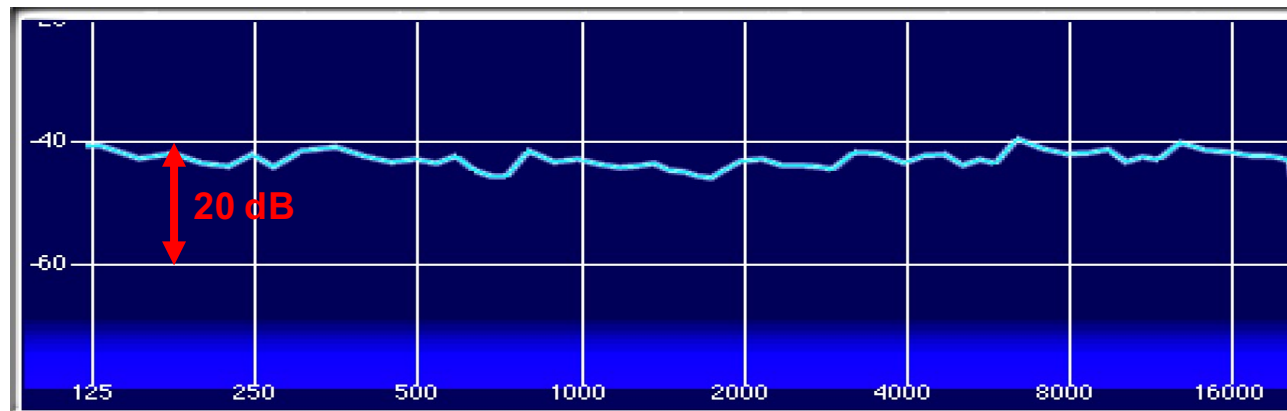
## Champ Direct Frontal :

**KU100**



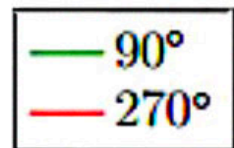
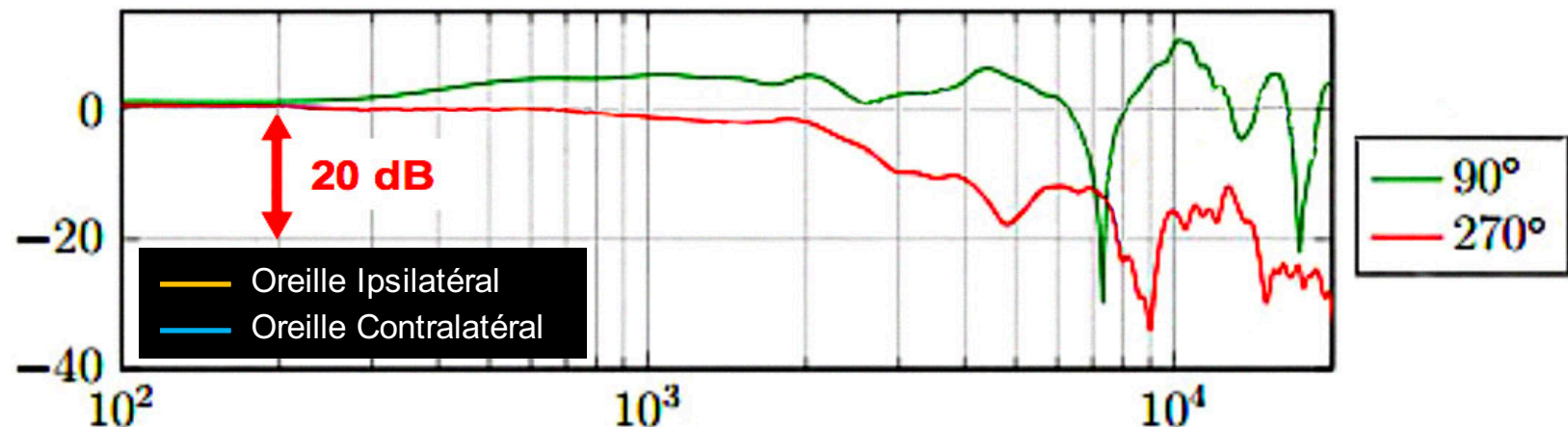
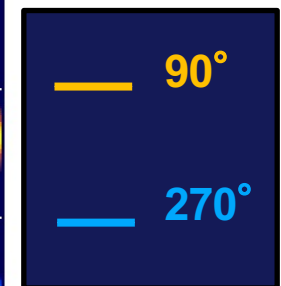
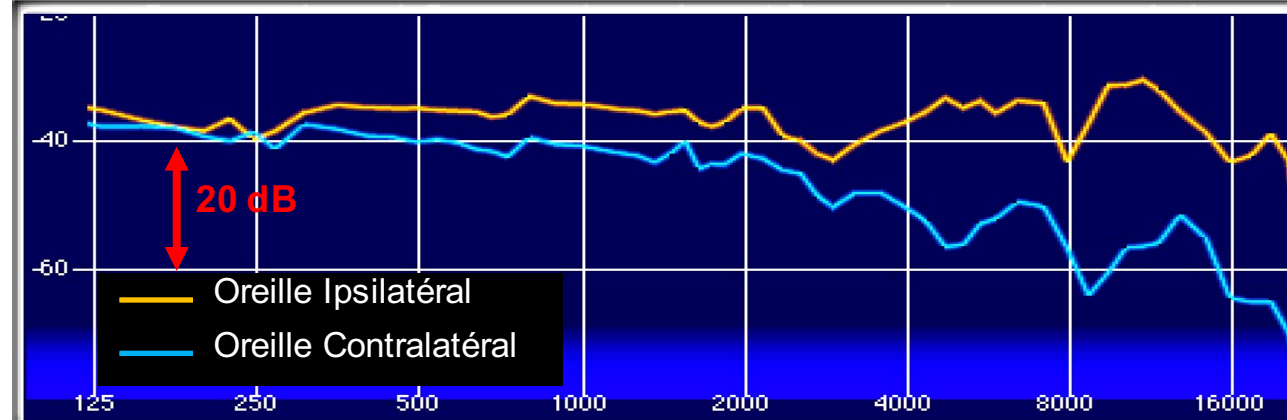
Documents Neumann  
Documents B&K

# IN Bruit Rose



# OUT ORBIT

Clarity = 0%

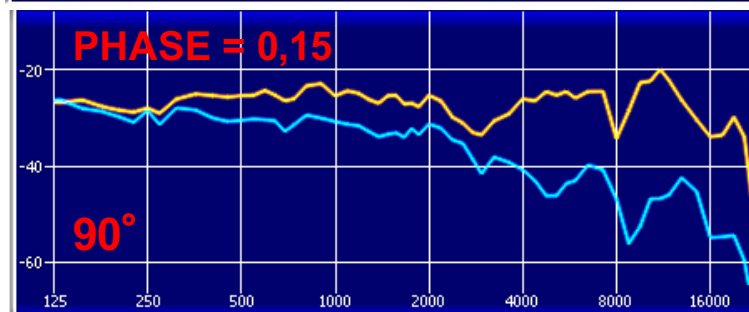
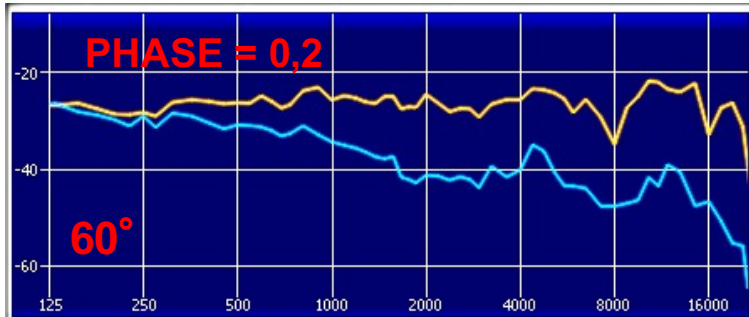
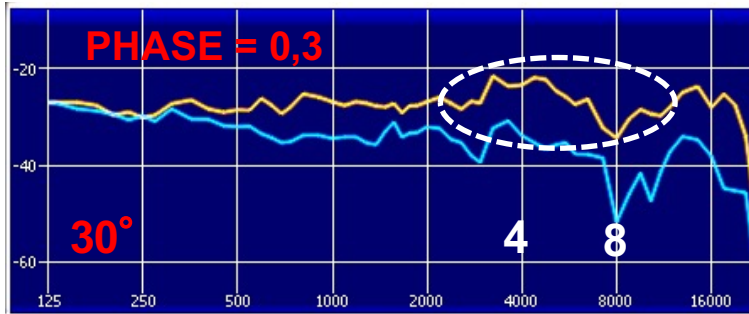
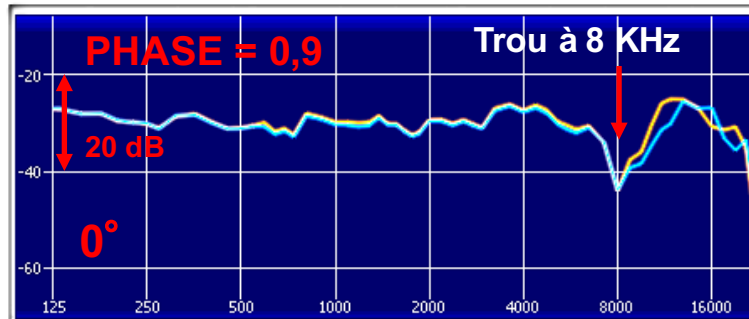


Jakob Vennerød  
Trondheim, June 2014

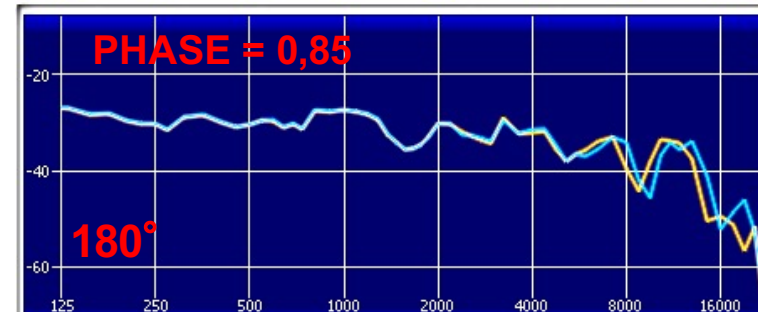
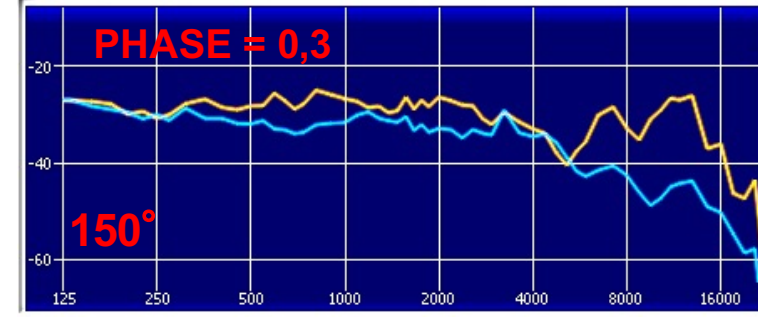
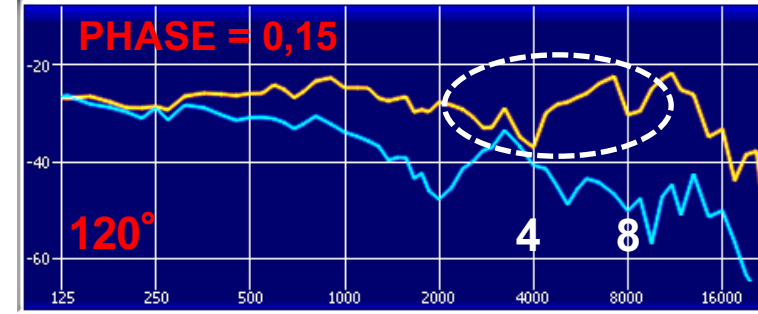
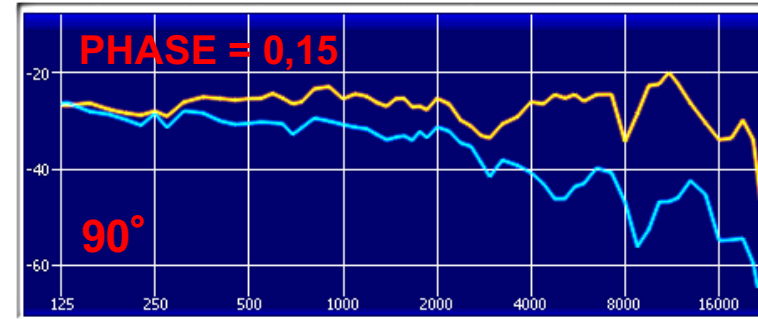
[https://www.researchgate.net/profile/Jakob\\_Vennerod/publication/278404811\\_Binaural\\_Reproduction\\_of\\_Higher\\_Order\\_Ambisonics\\_A\\_Real-Time\\_Implementation\\_and\\_Perceptual\\_Improvements/links/5580323408ae87edac4c9091/Binaural-Reproduction-of-Higher-Order-Ambisonics-A-Real-Time-Implementation-and-Perceptual-Improvements.pdf](https://www.researchgate.net/profile/Jakob_Vennerod/publication/278404811_Binaural_Reproduction_of_Higher_Order_Ambisonics_A_Real-Time_Implementation_and_Perceptual_Improvements/links/5580323408ae87edac4c9091/Binaural-Reproduction-of-Higher-Order-Ambisonics-A-Real-Time-Implementation-and-Perceptual-Improvements.pdf)



# ORBIT : HRTF KU100 pour une source Mono..... CLARITY 0 %



**5.0**



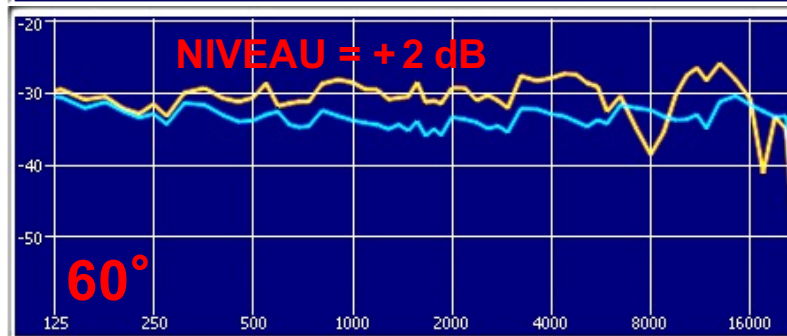
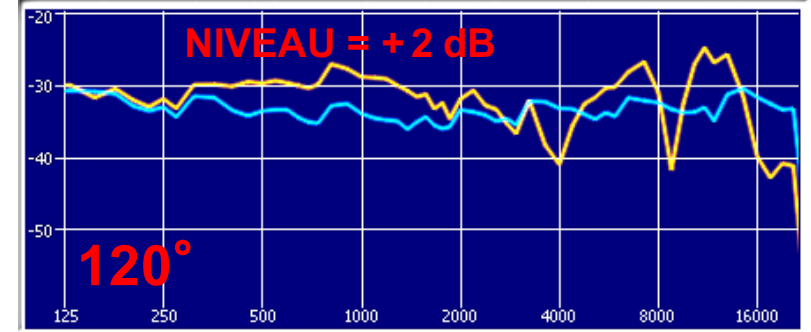
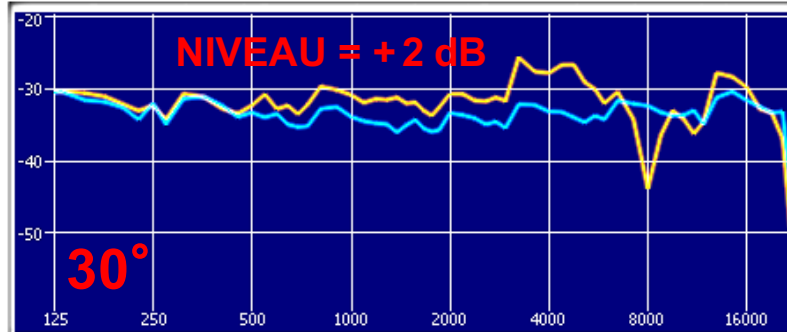
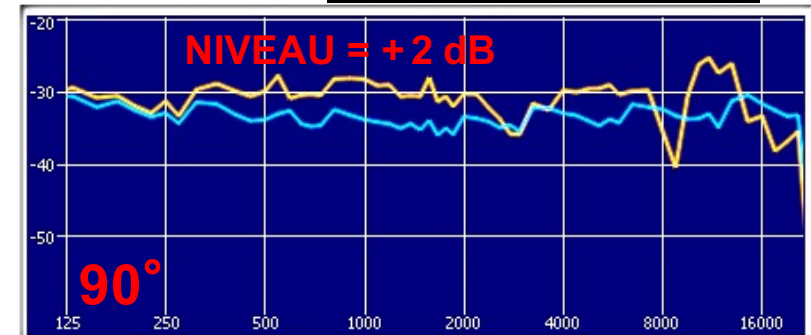
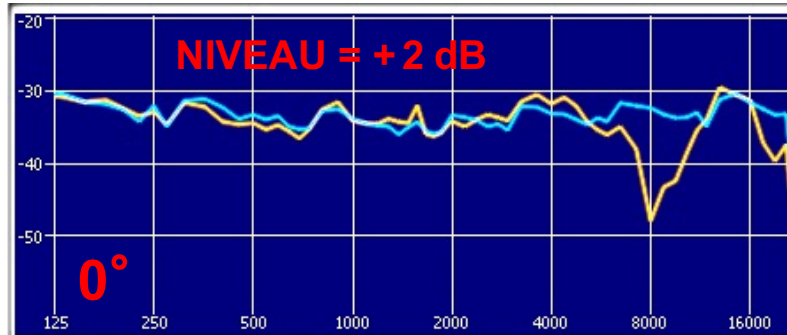
— Oreille Ipsilatéral  
 — Oreille Contralatéral

**CLARITY 0 % = IS**

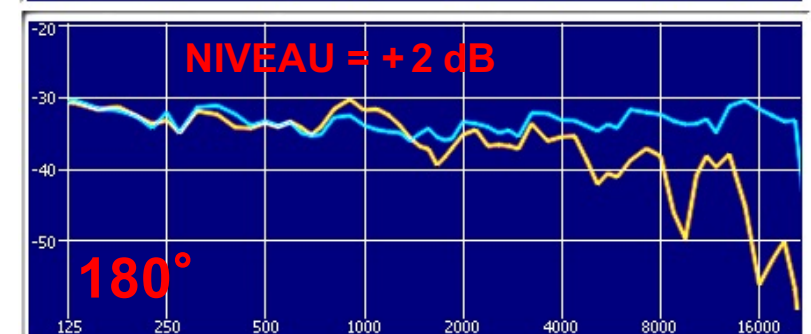
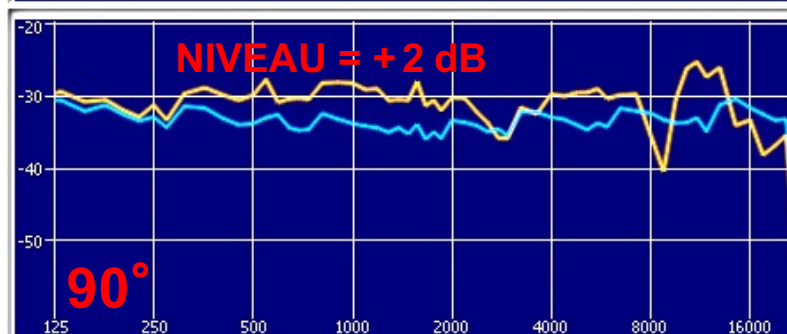
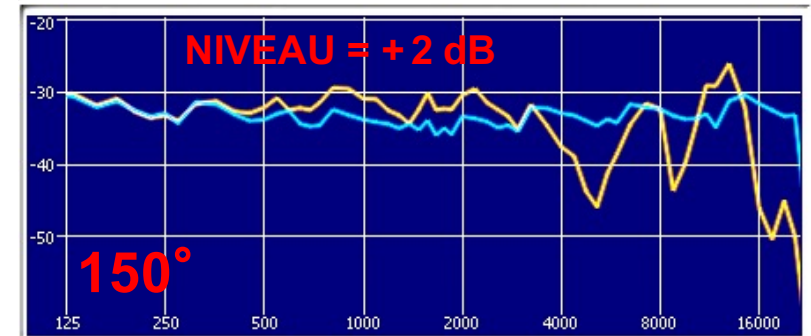
— Oreille Ipsilatéral  
 — Oreille Contralatéral

# HRTF et Bruit Rose : **ORBIT** *KU 100*

— Oreille Ipsilatéral **L**  
— BRUIT ROSE IN



**CLARITY 0 %**



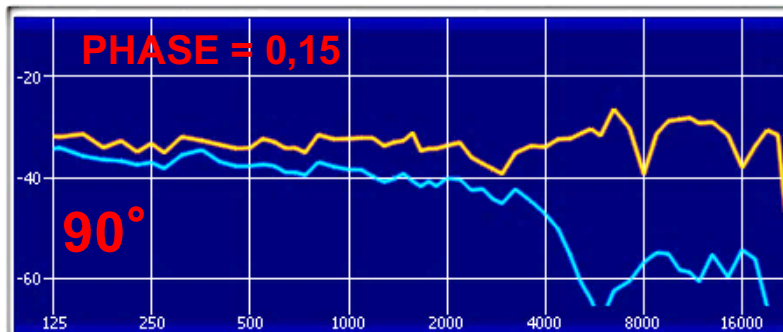
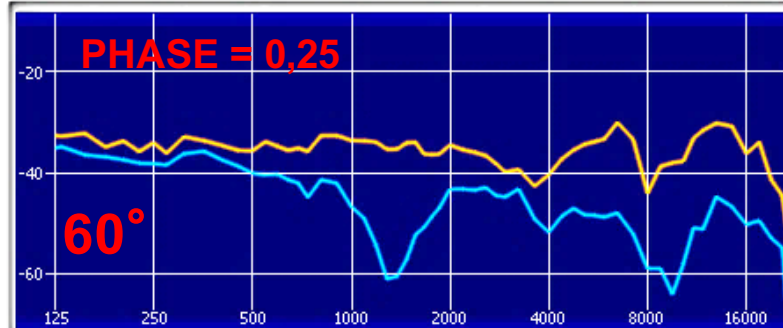
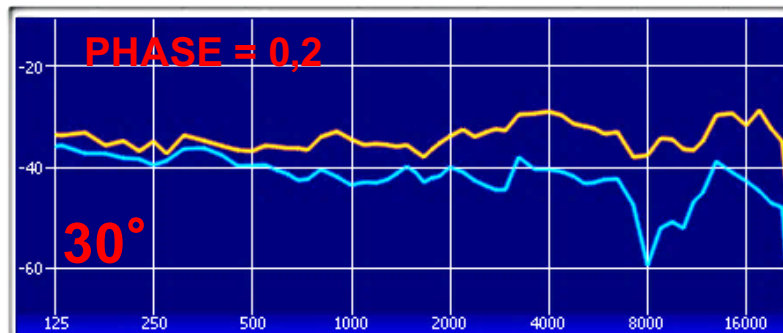
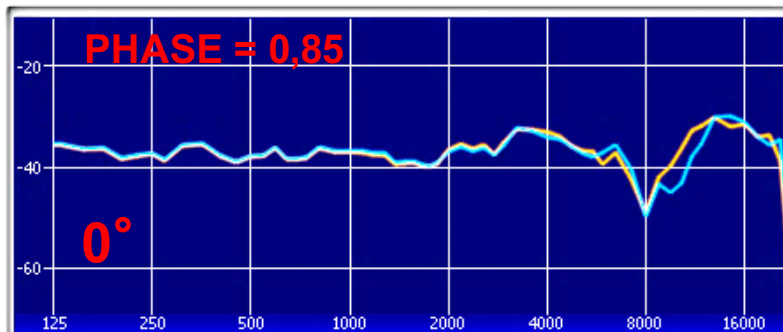


# PLUG-IN **KU 100** SADIE

*Adopté par:*

Youtube 360  
et Google VR

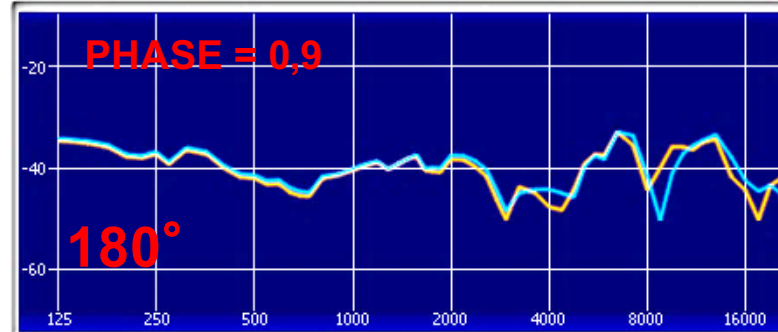
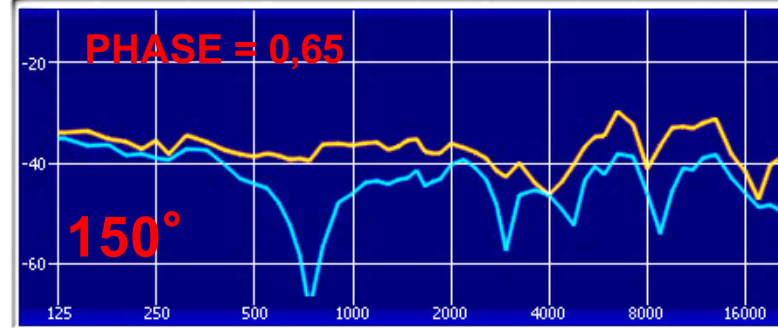
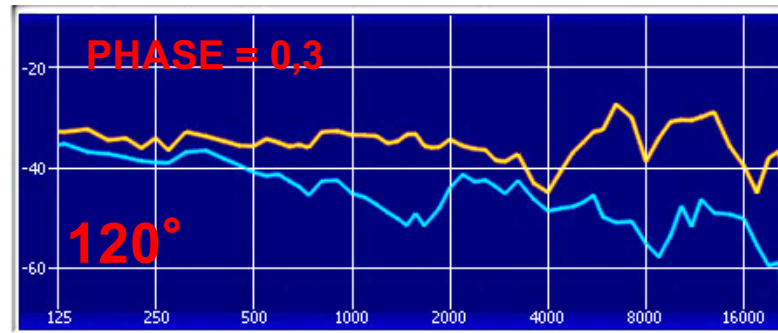
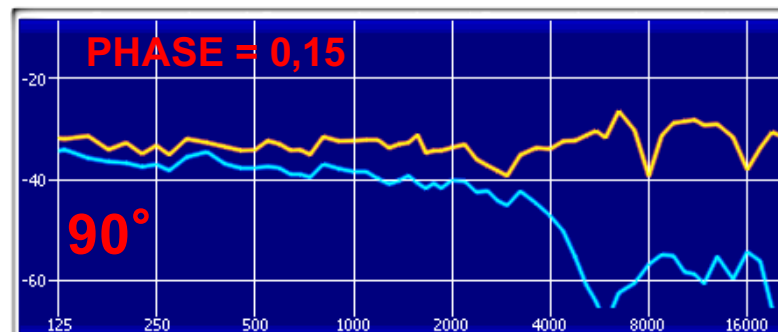
— Oreille Ipsilatéral  
— Oreille Contralatéral



SADIE\_KU100\_DFC\_256\_order\_fir\_48000.sofa

<https://www.york.ac.uk/sadie-project/binaural.html>

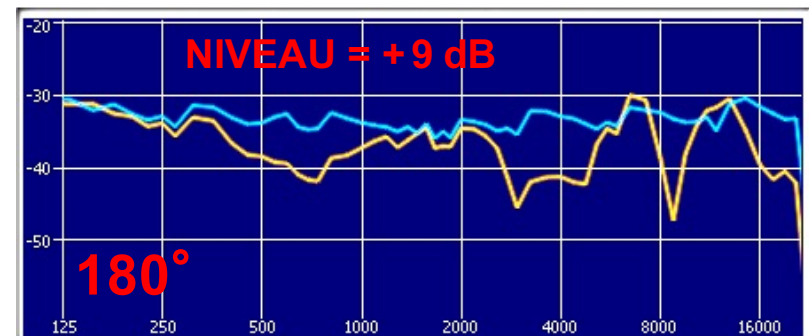
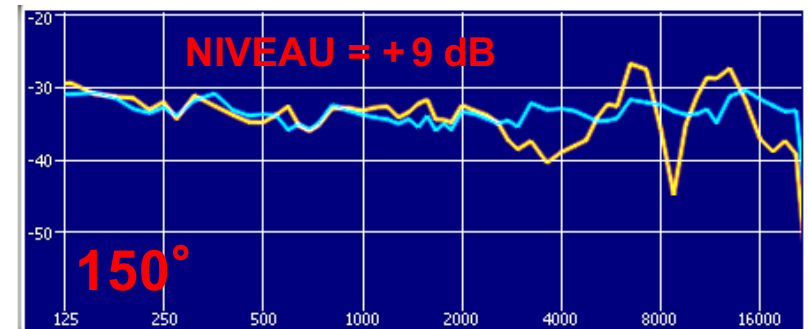
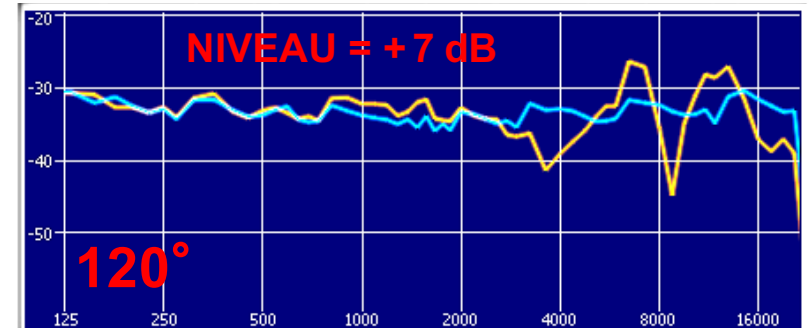
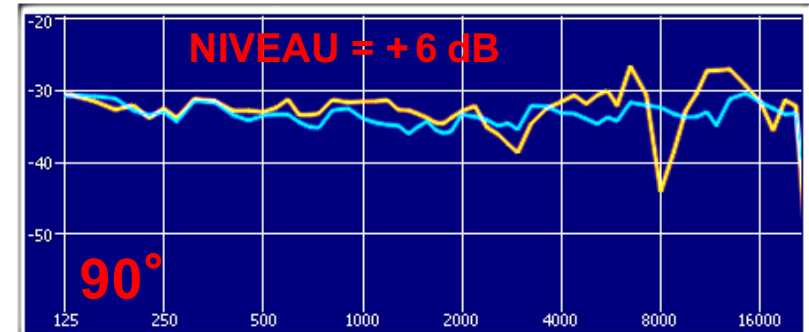
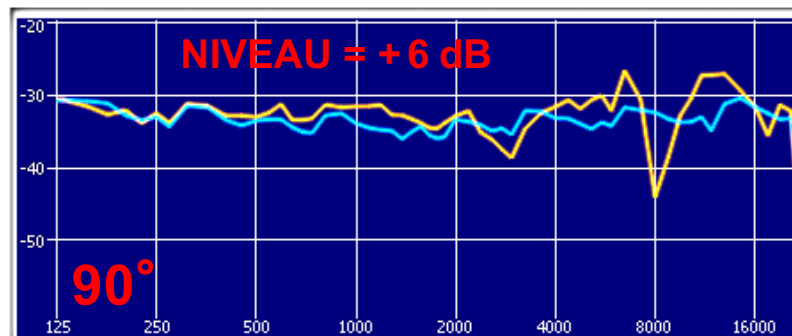
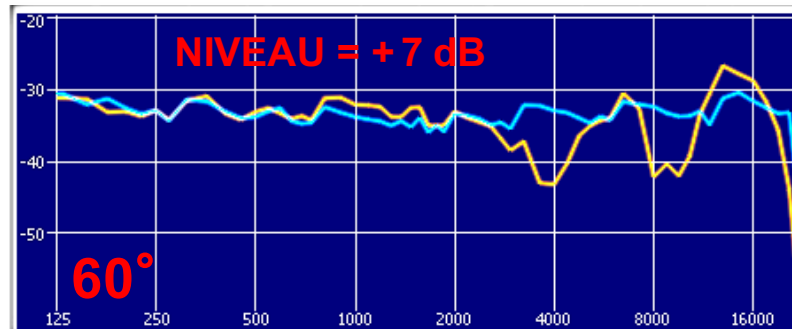
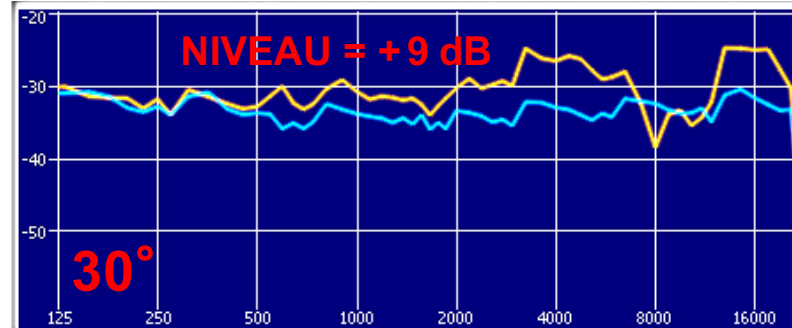
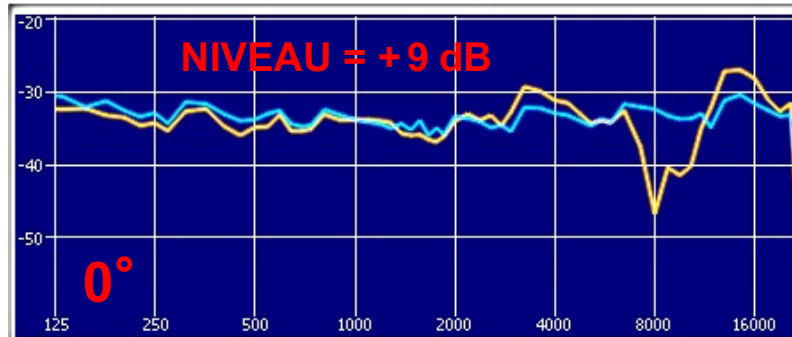
University of York



# HRTF et Bruit Rose : **SADIE** KU 100

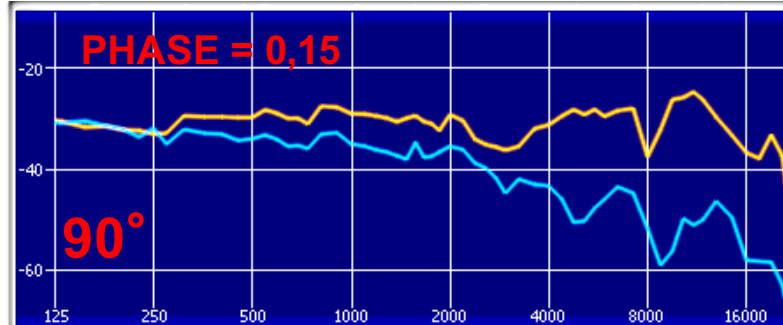
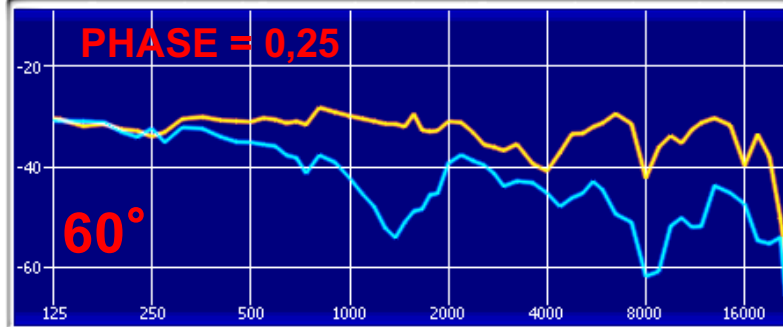
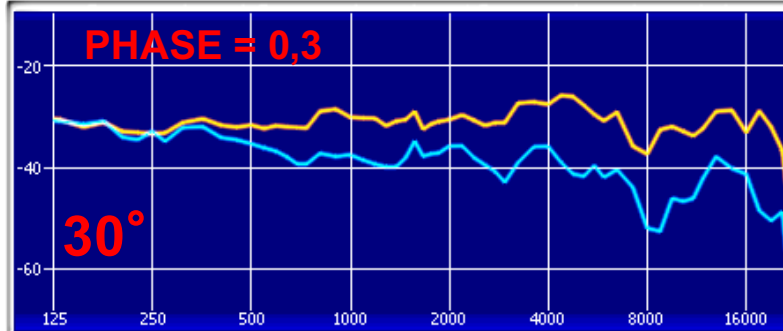
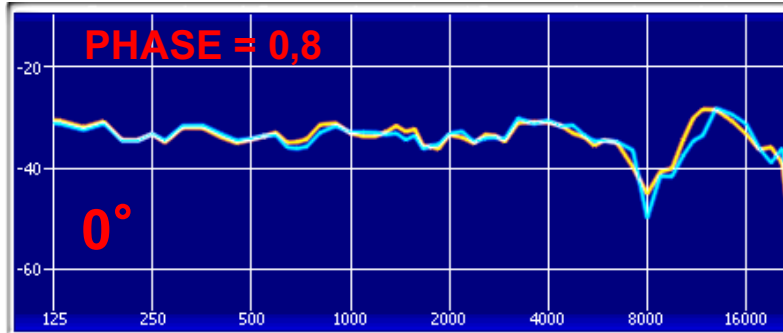
— Oreille Ipsilatéral **L**  
— BRUIT ROSE IN

University  
of York



SADIE\_KU100\_DFC\_256\_order  
\_fir\_48000.sofa

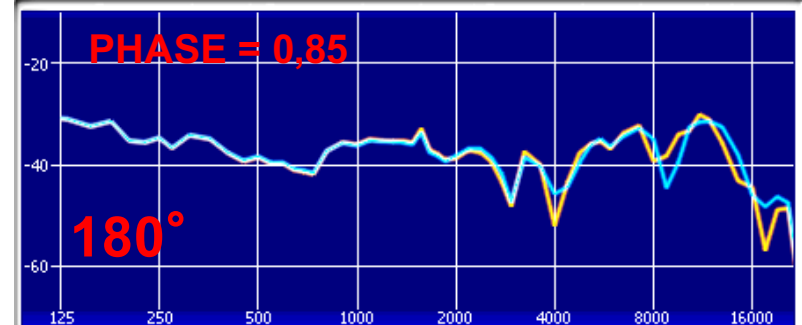
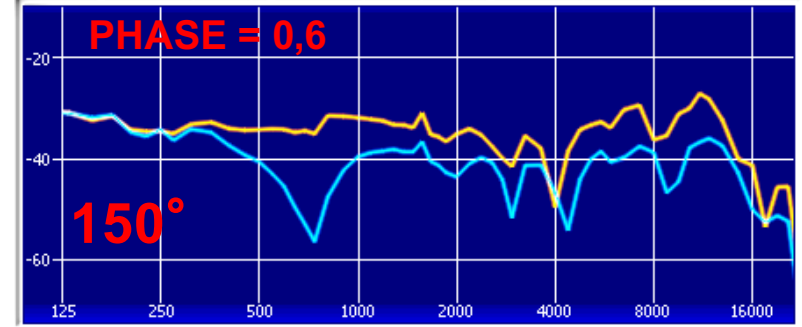
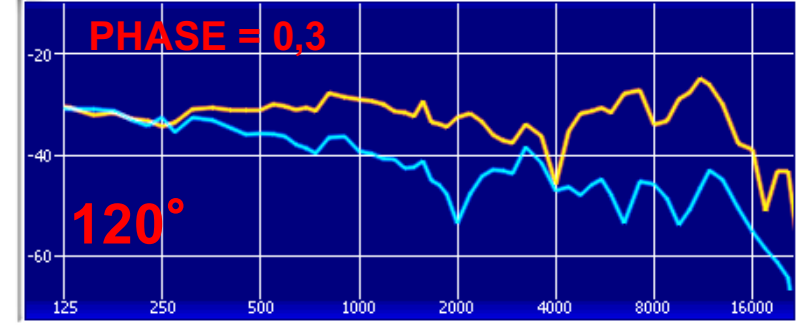
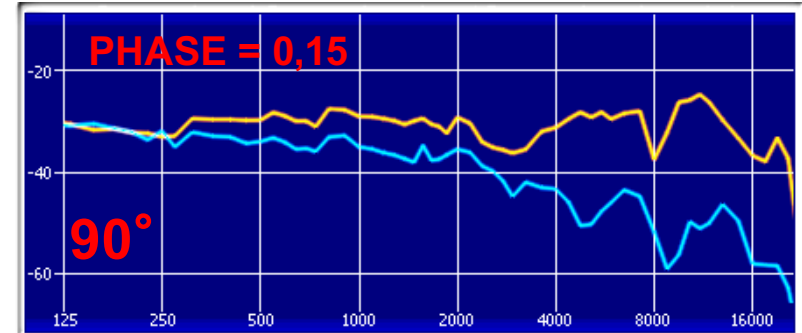
# PLUG-IN *KU 100* TH Köln



HRIR\_FULL2DEG.sofa

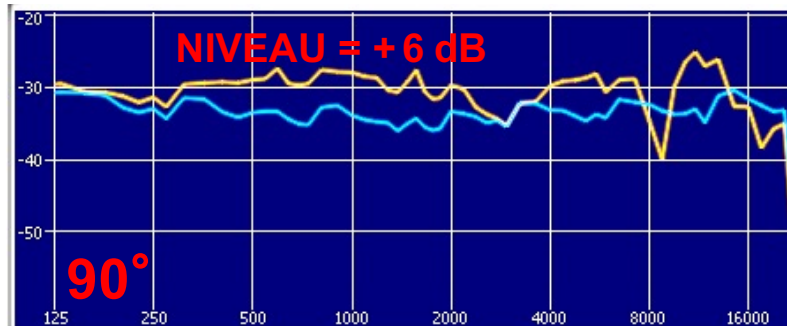
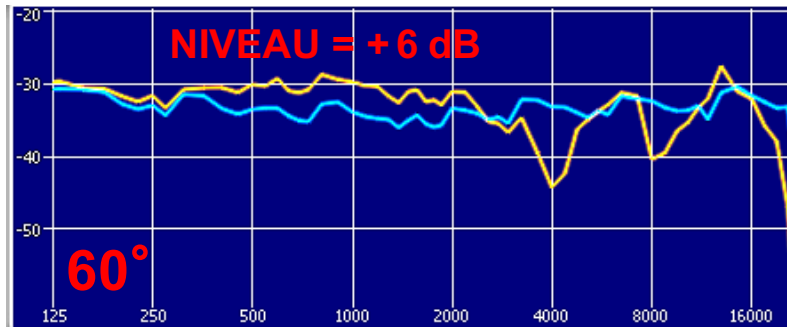
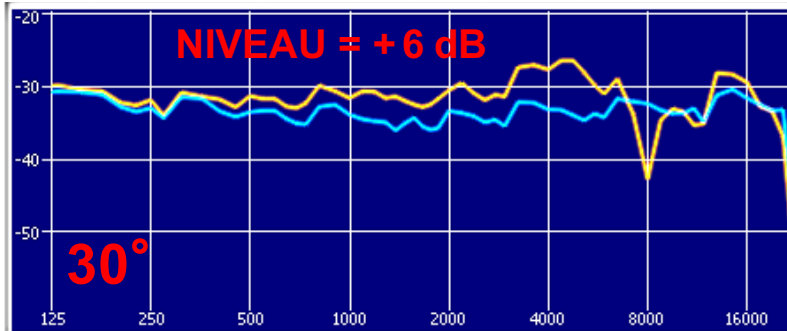
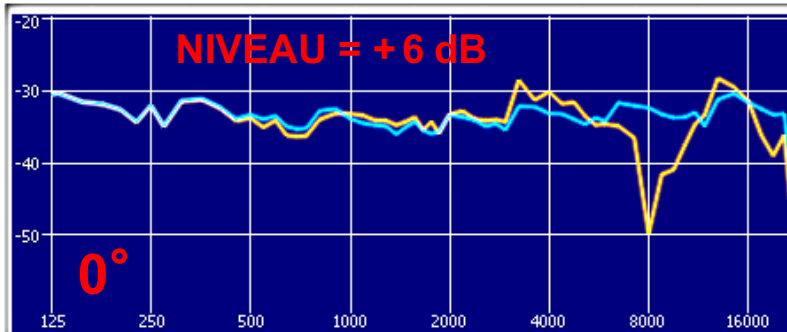
<http://sofacoustics.org/data/database/fhk/>

<http://audiogroup.web.th-koeln.de/ku100nfhrir.html>

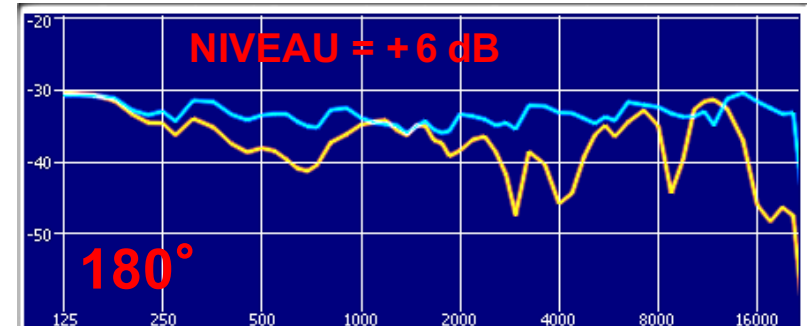
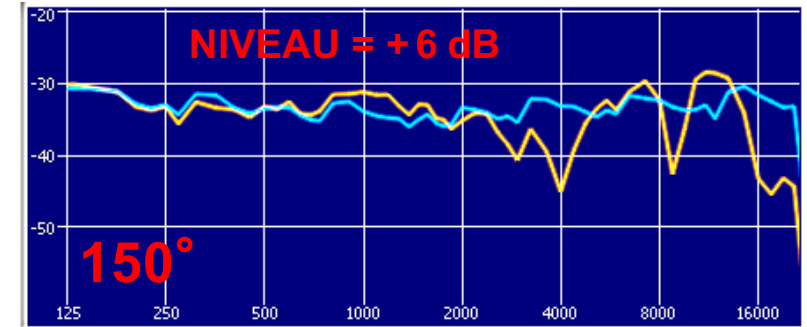
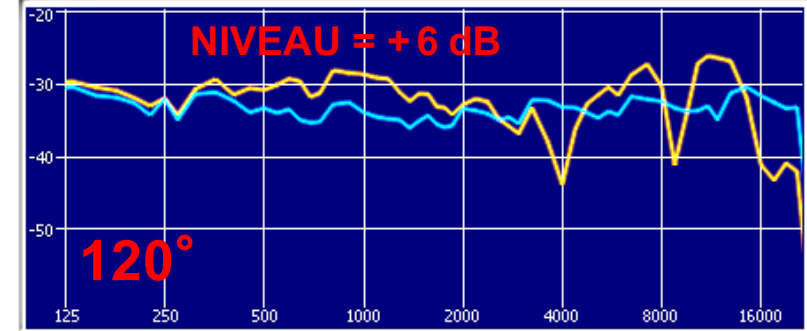
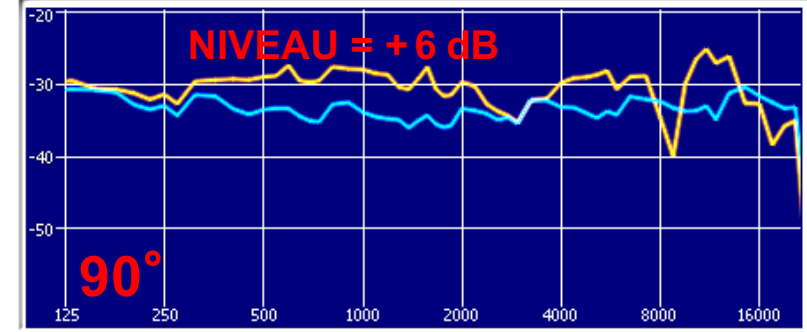


# HRTF et Bruit Rose : **KOLN** *KU 100*

— Oreille Ipsilatéral **L**  
— BRUIT ROSE IN



HRIR\_FULL2DEG.sofa





# En 1972 : Kemar (mannequin anthropométrique)

**40 years  
and still the same  
– but different**

En 1972, KEMAR a été introduite dans le monde par Knowles Electronics. Il a été le premier simulateur de torse et spécialement conçu pour la recherche acoustique et a permis aux laboratoires de prothèses auditives pour effectuer de simulation des mesures in situ de prothèses auditives.

1972  
2012



[http://kemar.us/KEMAR\\_Book.pdf](http://kemar.us/KEMAR_Book.pdf)

## MANIKIN MEASUREMENTS - KEMAR by GRAS

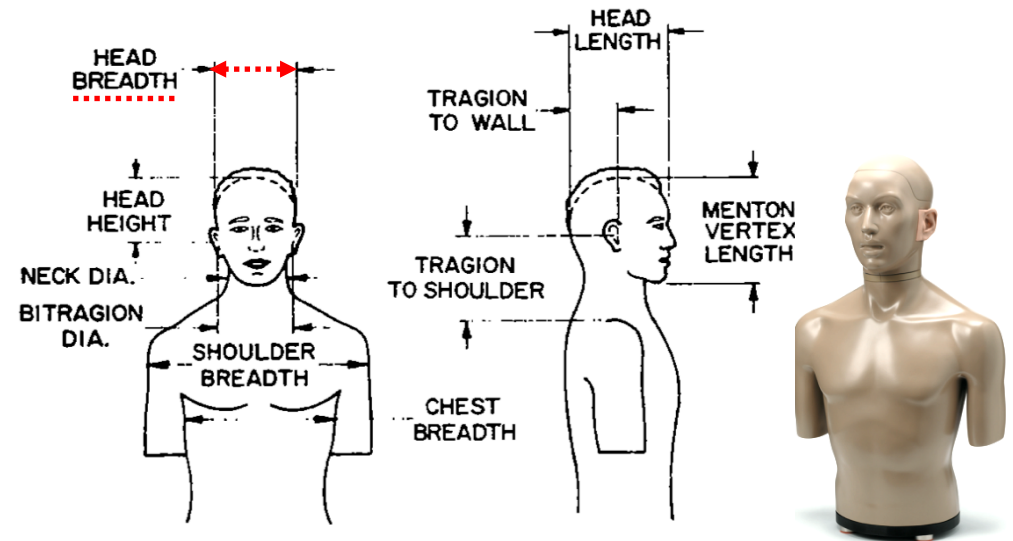


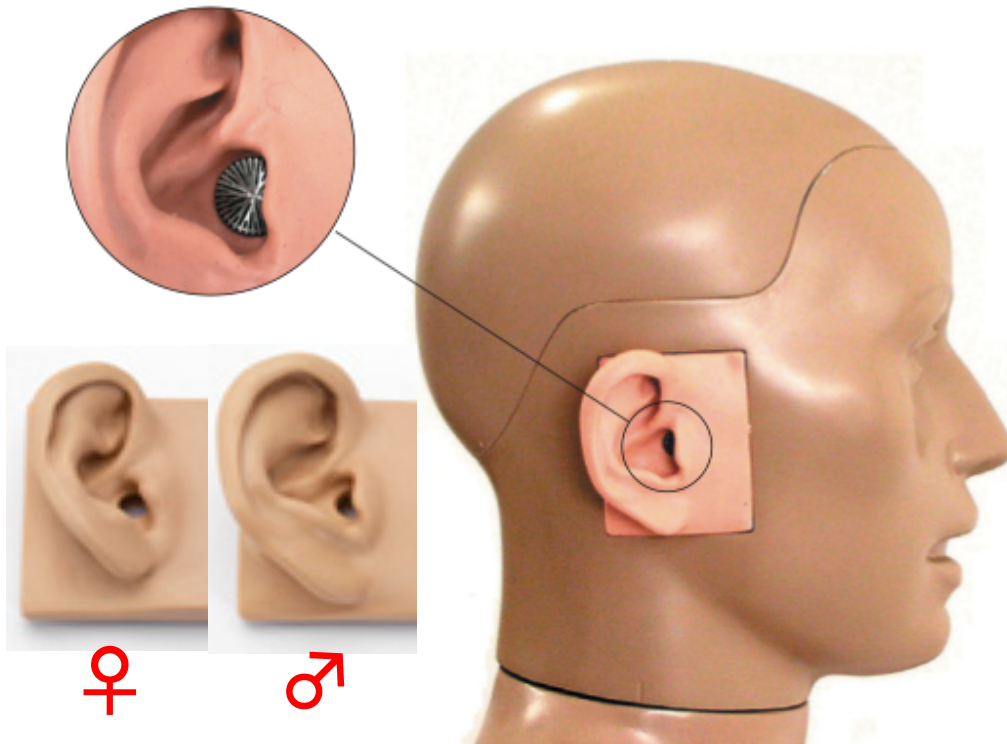
FIG. 1. Anthropometric measures used in design of KEMAR.

TABLE I. Dimensions for KEMAR and average human adults, in centimeters.

	Median male	Median female	Average human	KEMAR
Head breadth	15.5	14.7	15.1	15.2
Head length	19.6	18.0	18.8	19.1
Head height	13.0	13.0	13.0	12.5
Bitracion diameter	14.2	13.5	13.85	14.3
Tracion to wall	10.2	9.4	9.8	9.65
Tracion to shoulder	18.8	16.3	17.55	17.5 <sup>a</sup>
Neck diameter	12.1	10.3	11.2	11.3
Shoulder breadth	45.5	39.9	42.7	44.0
Chest breadth	30.5	27.7	29.1	28.2
Menton vertex length	23.2	21.1	22.15	22.4

<sup>a</sup>Adjustable over  $\pm 1.27$  cm.

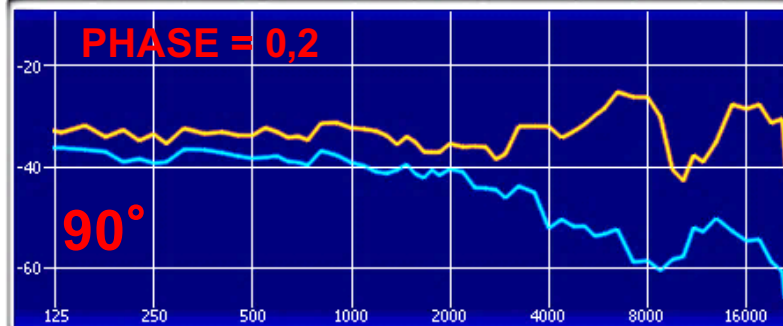
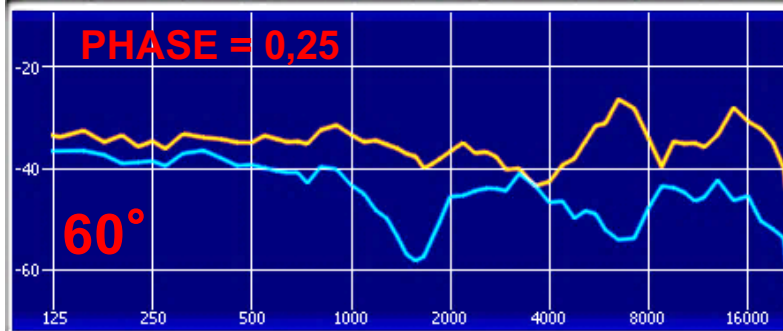
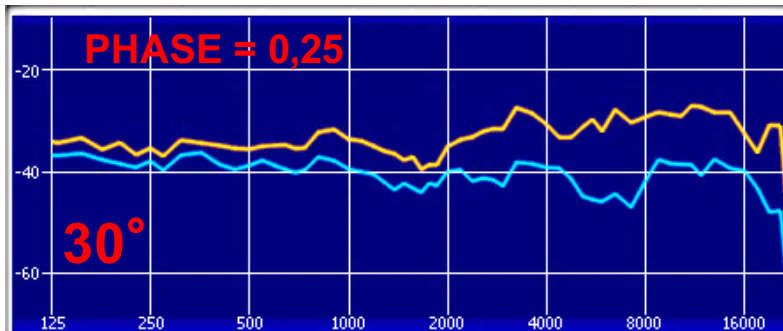
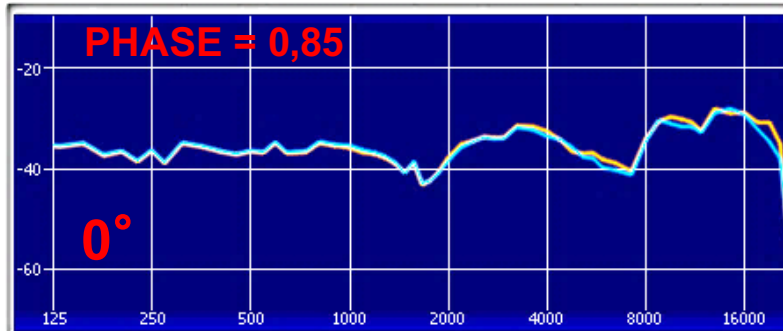
Photos : [www.gras.dk](http://www.gras.dk)





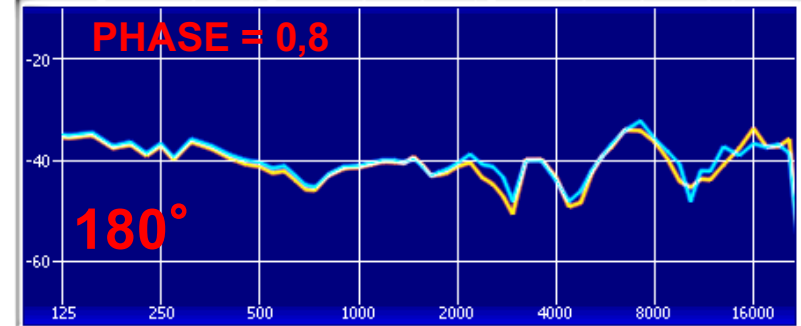
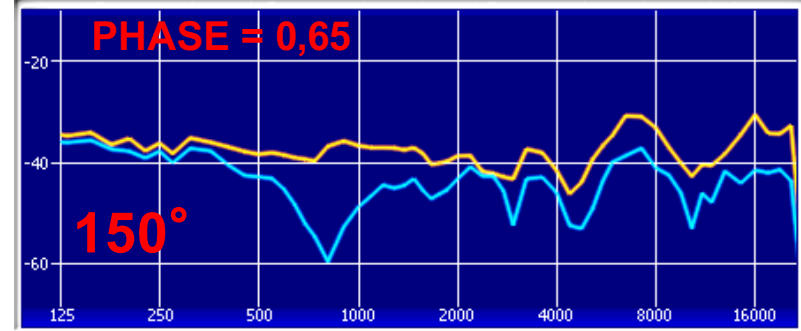
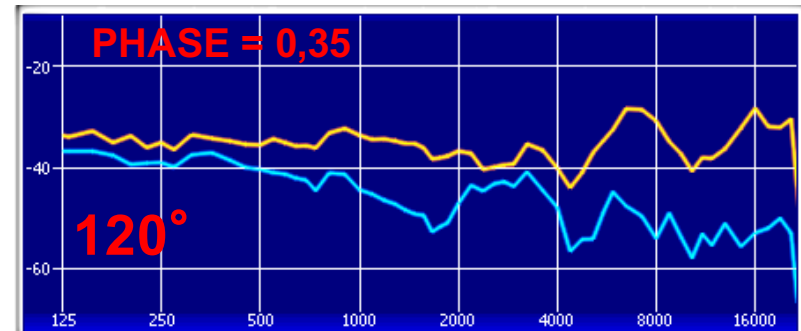
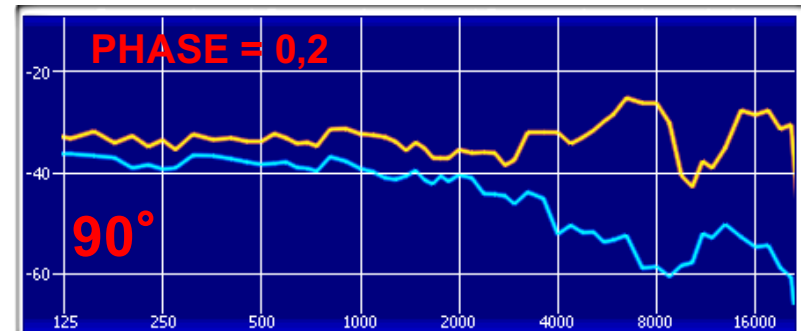
# PLUG-IN **KEMAR** de **SADIE**

— Oreille Ipsilatéral  
— Oreille Contralatéral

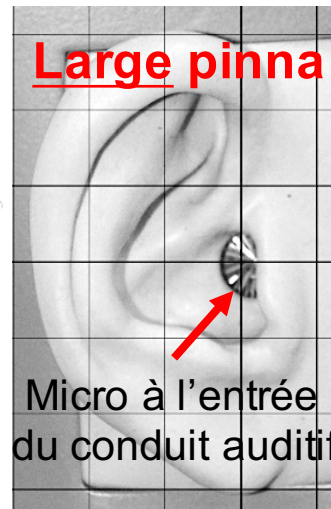


University  
of York

[https://www.york.ac.uk/sadie-  
project/binaural.html](https://www.york.ac.uk/sadie-project/binaural.html)  
SADIE\_KEMAR\_DFC\_256\_  
order\_fir\_48000.sofa



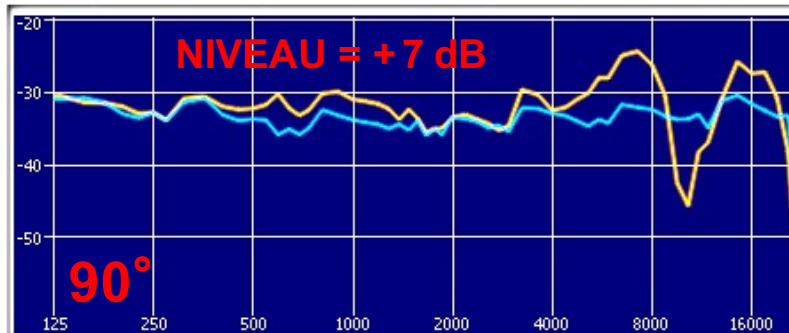
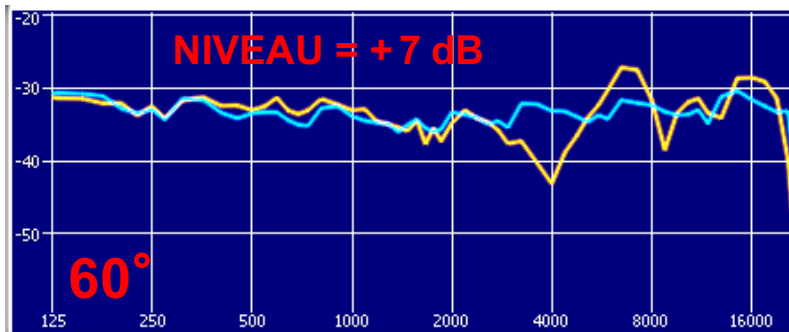
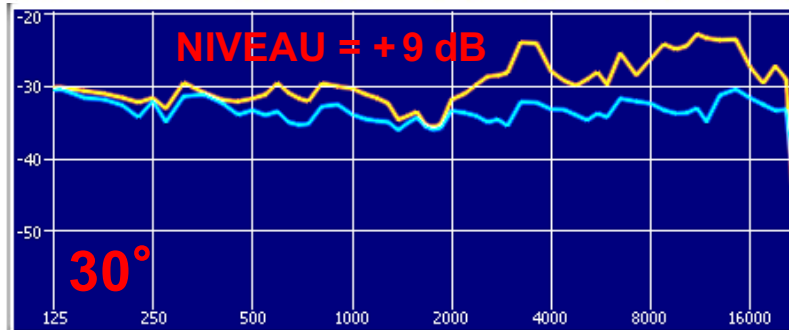
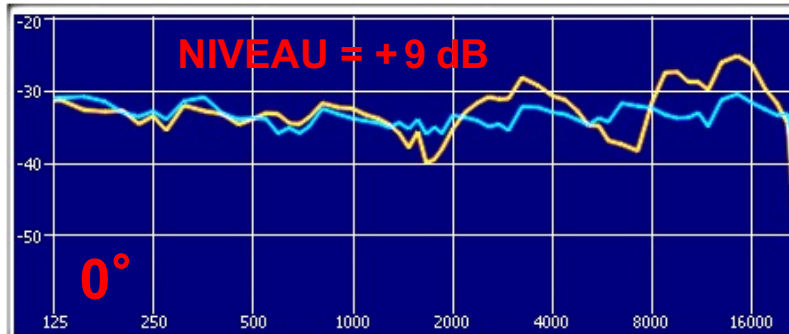
Large pinna



Micro à l'entrée  
du conduit auditif

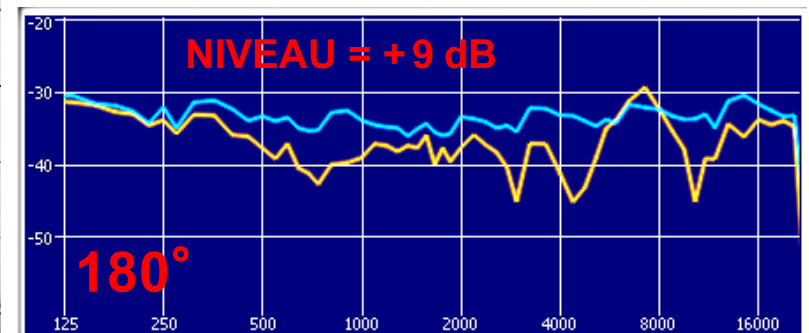
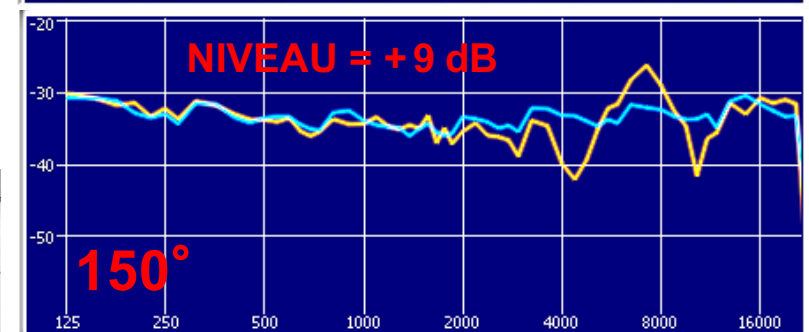
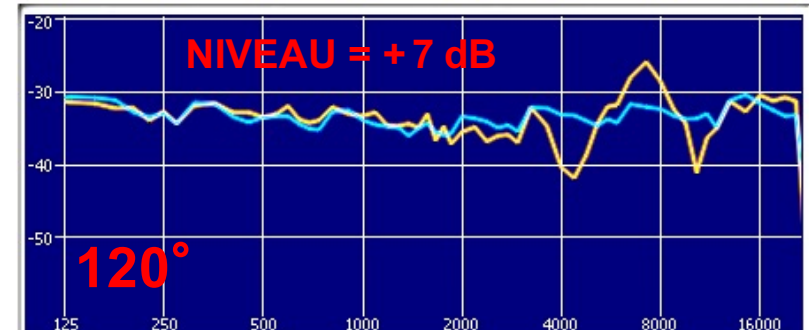
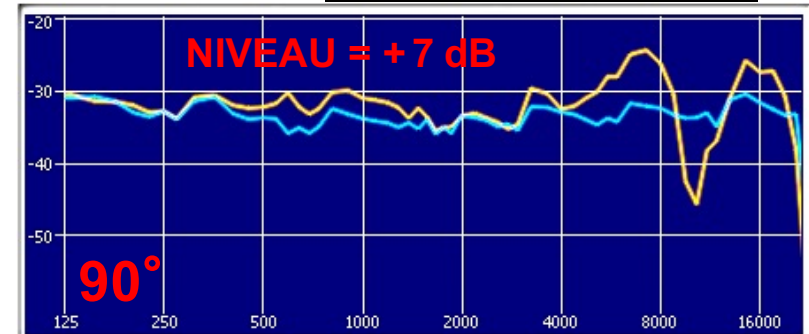
# HRTF et Bruit Rose **KEMAR** de **SADIE**

— Oreille Ipsilatéral **L**  
— BRUIT ROSE IN

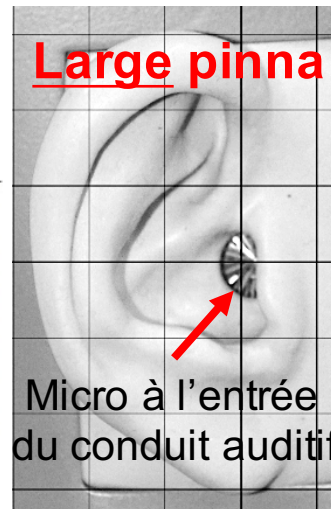


University  
of York

[https://www.york.ac.uk/sadie-  
project/binaural.html](https://www.york.ac.uk/sadie-project/binaural.html)  
SADIE\_KEMAR\_DFC\_256\_  
order\_fir\_48000.sofa



Large pinna



Micro à l'entrée  
du conduit auditif

# Electroacoustic Measurements of Headphones

Christopher J. Struck

CJS Labs  
San Francisco, CA – USA

2009

© Copyright 2009 CJS Labs – San Francisco, CA USA – [www.cjs-labs.com](http://www.cjs-labs.com) Email: [cjs@cjs-labs.com](mailto:cjs@cjs-labs.com)

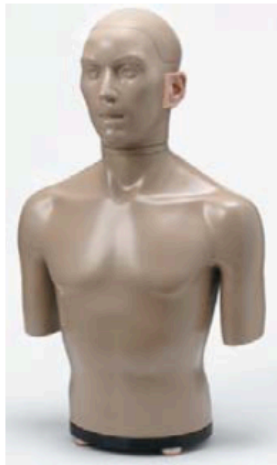
1



IEC 60711 Ear Simulator

## KEMAR

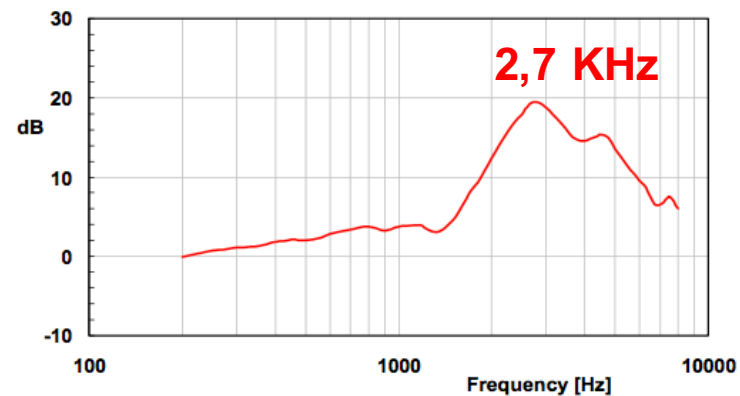
- Zwislocki OR IEC 60711 Ear Simulators
- Anatomical Pinnæ (soft or hard)
- Anthropomorphic Geometry
- Fulfills **ANSI S3.36** (and IEC 60959 and ITU-T Rec. P.58)



© Copyright 2009 CJS Labs – San Francisco, CA USA – [www.cjs-labs.com](http://www.cjs-labs.com) Email: [cjs@cjs-labs.com](mailto:cjs@cjs-labs.com)

23

## KEMAR Free Field Response with Zwislocki Coupler

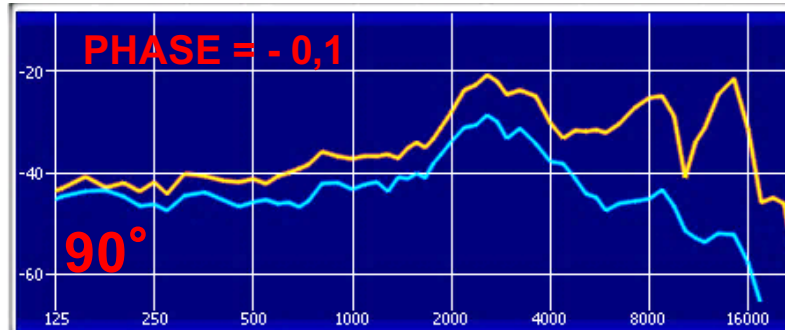
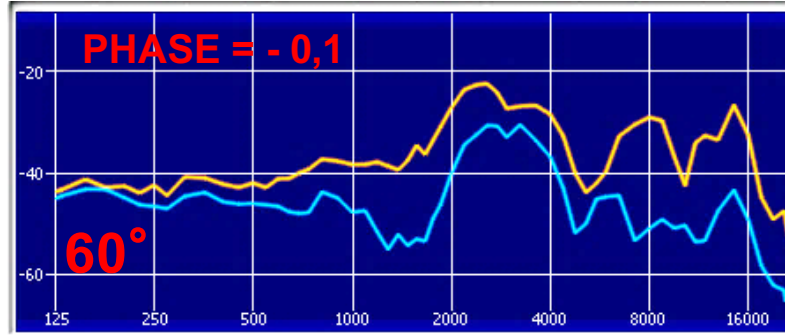
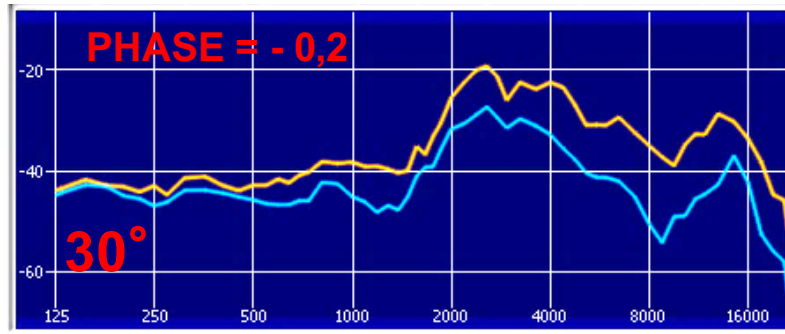
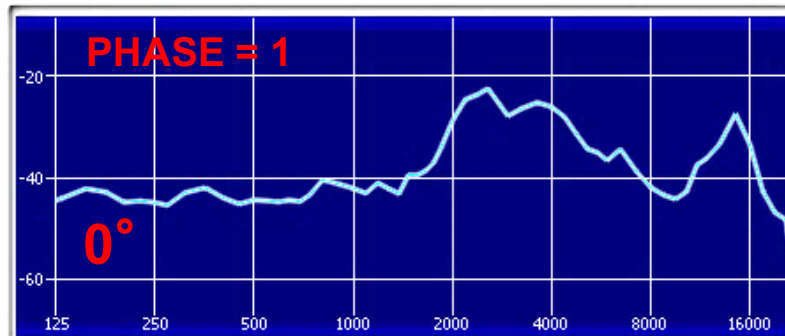


© Copyright 2009 CJS Labs – San Francisco, CA USA – [www.cjs-labs.com](http://www.cjs-labs.com) Email: [cjs@cjs-labs.com](mailto:cjs@cjs-labs.com)

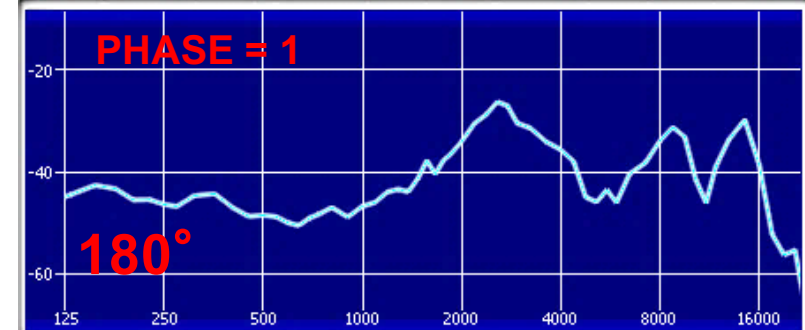
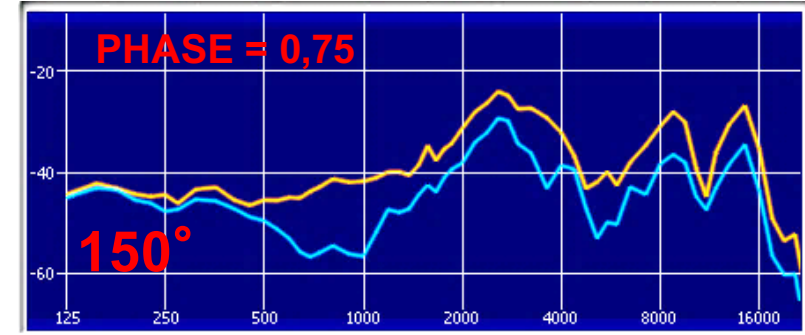
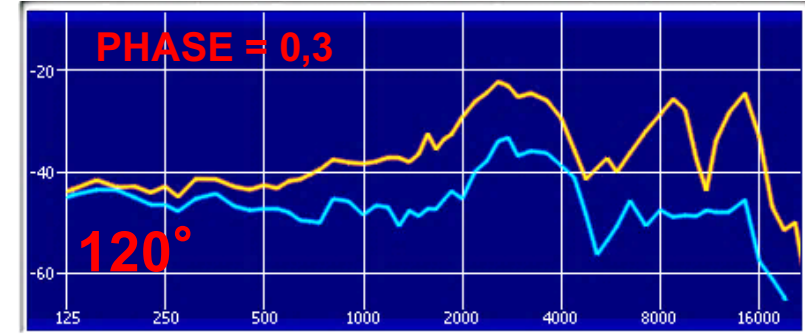
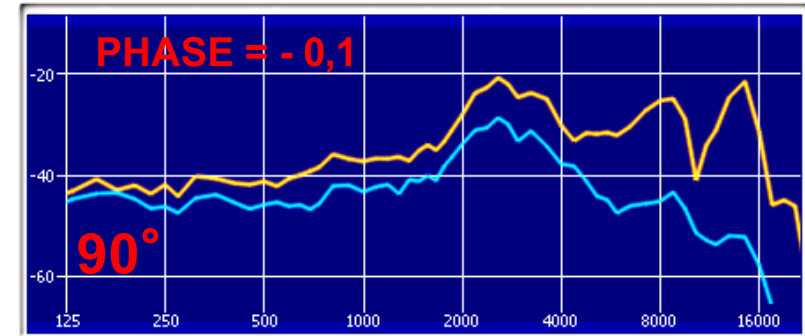
24

# PLUG-IN **KEMAR** de **MIT** Oreilles Normales

— Oreille Ipsilatéral  
— Oreille Contralatéral



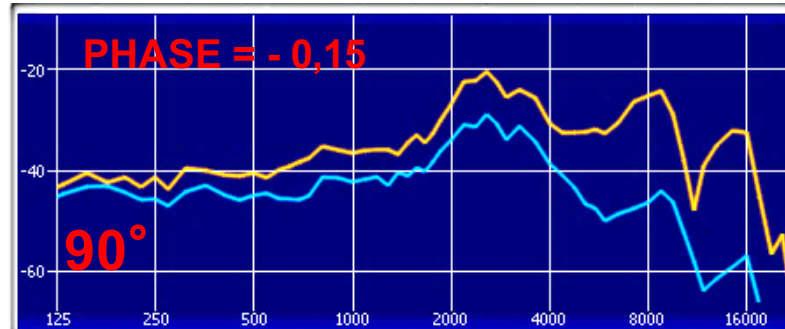
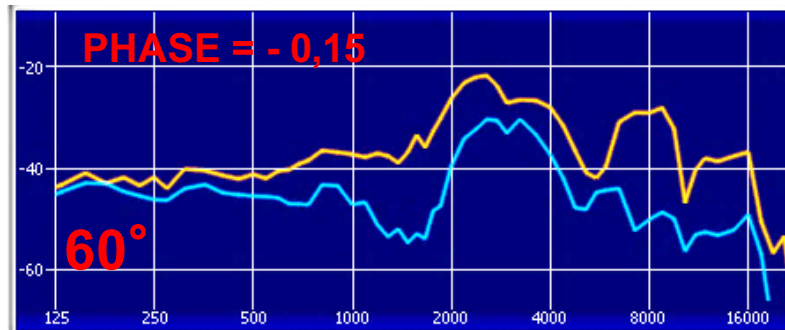
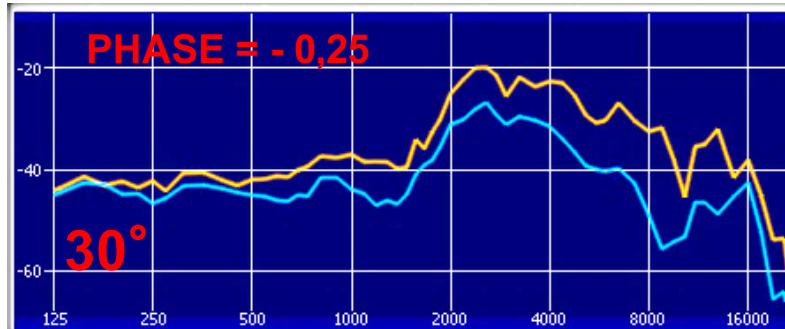
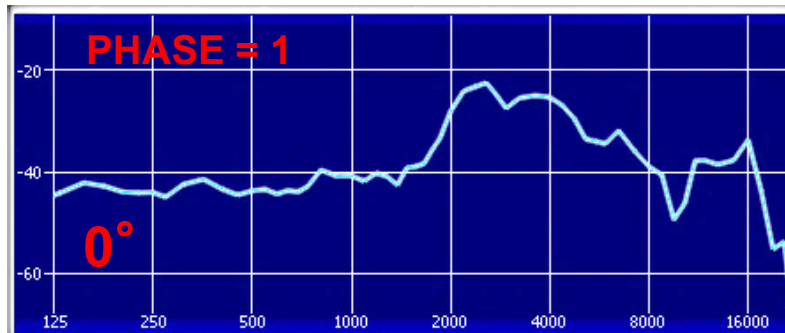
mit\_kemar\_normal\_pinna.sofa



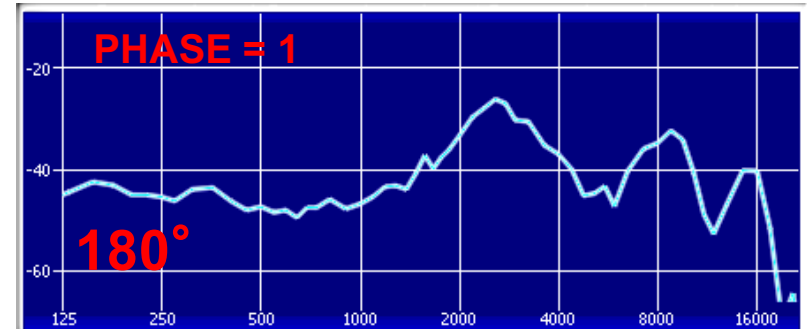
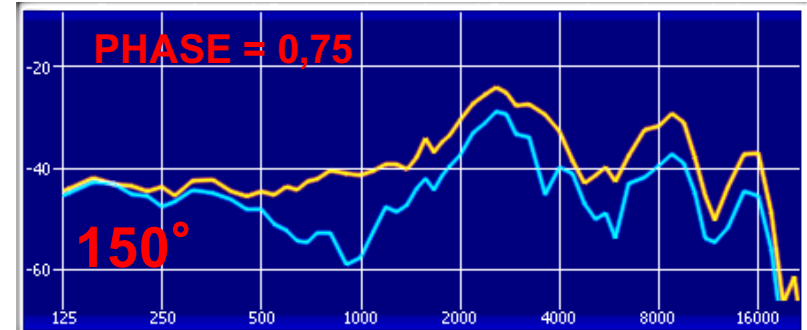
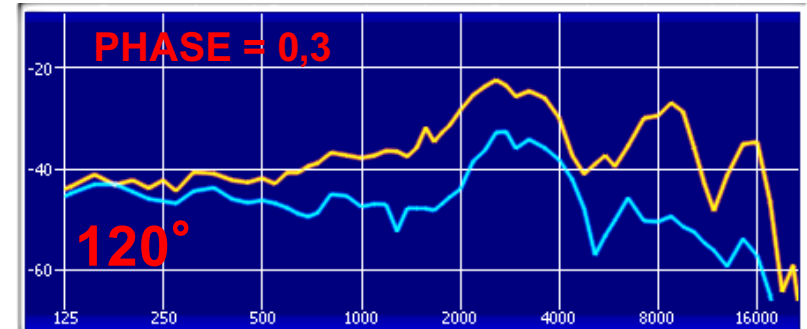
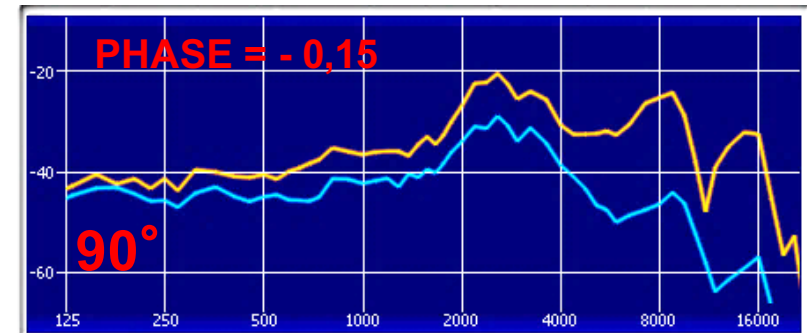


# PLUG-IN **KEMAR** de **MIT** Oreilles Larges

— Oreille Ipsilatéral  
— Oreille Contralatéral



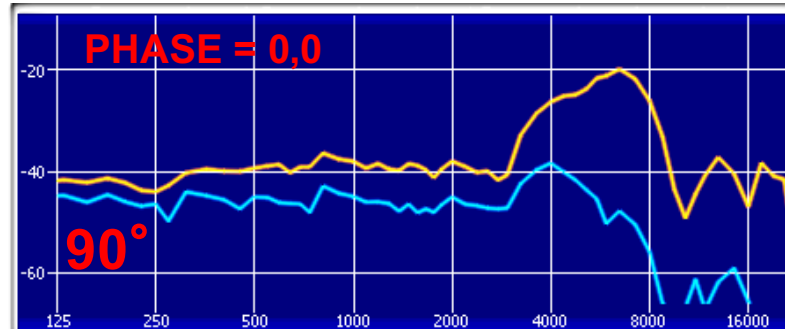
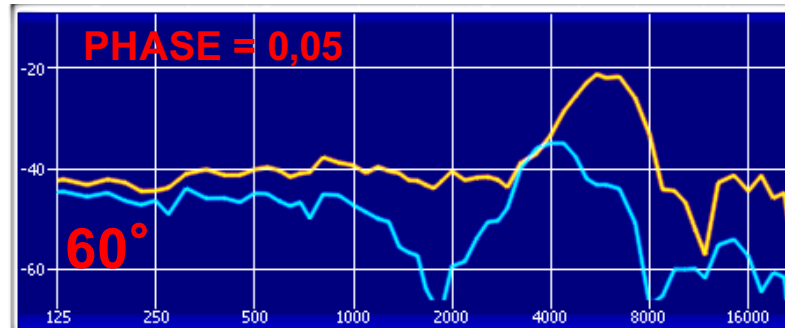
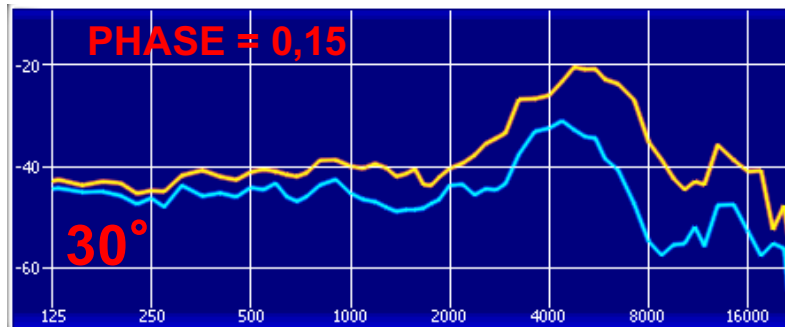
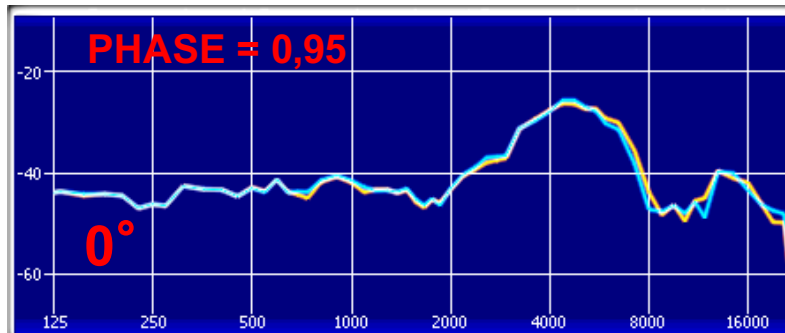
mit\_kemar\_large\_pinna.sofa





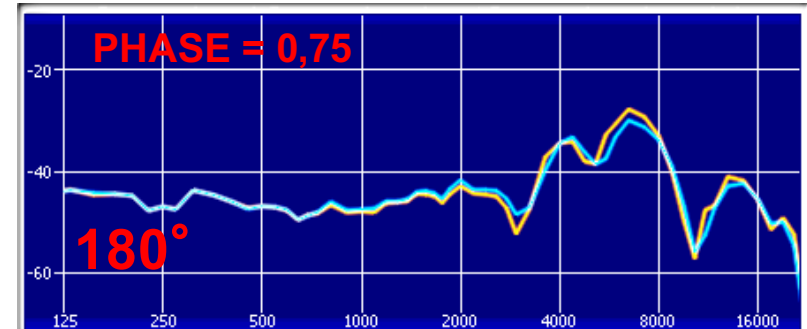
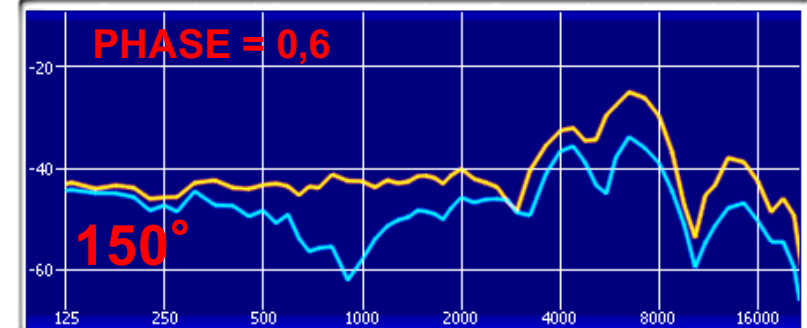
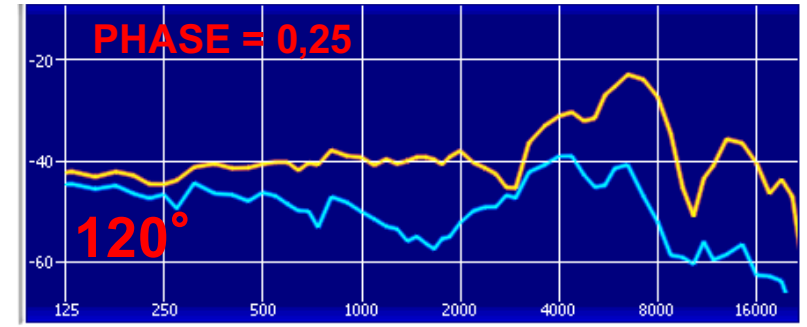
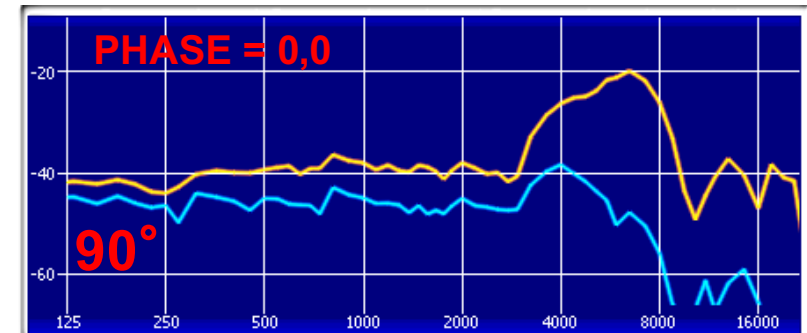
# PLUG-IN **KEMAR** de **QU TWO EARS**

— Oreille Ipsilatéral  
— Oreille Contralatéral



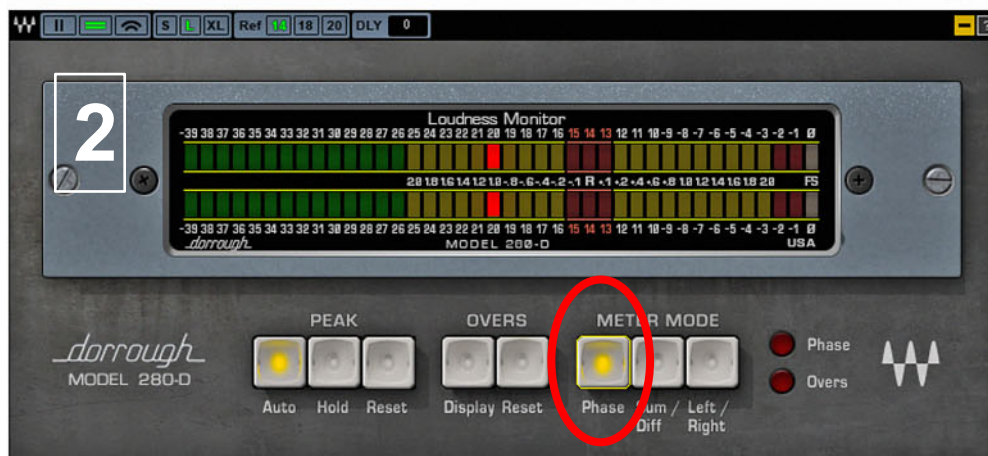
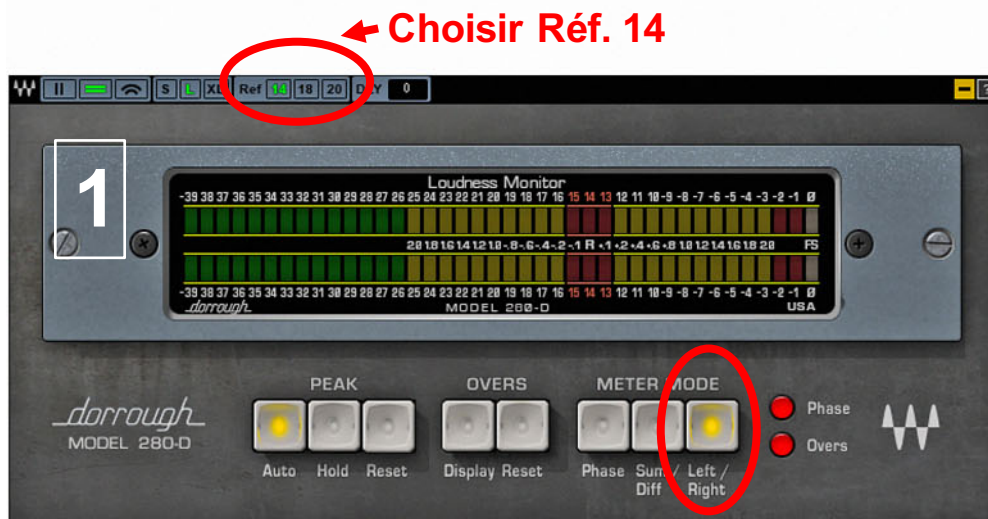
QU\_KEMAR\_anechoic.sofa

<http://docs.tweears.eu/en/latest/database/impulse-responses/hirs/>



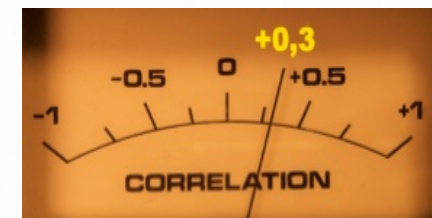
**Annexes :**

# Le Plug-in Waves **Dorrough** Stéréo utilisé comme Phasemètre :



## Caractéristiques Techniques :

- Temps d'intégration  $\approx$  **500 ms**
- Seuil de sensibilité pour une réponse exacte  $\approx$  - 32 dBFS (Affichage de la même valeur pour des écarts max de 32 dBFS d'IDL)
- Réponse linéaire de la phase et non logarithmique comme sur la plupart des *Phasemètres Plug-ins*...
- Phasemètre Plug-in comparable aux phasemètres analogiques du siècle dernier...

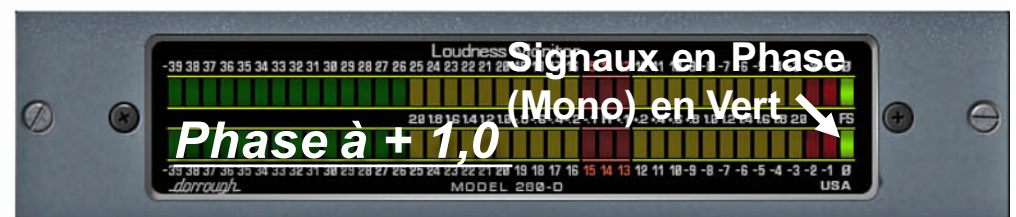
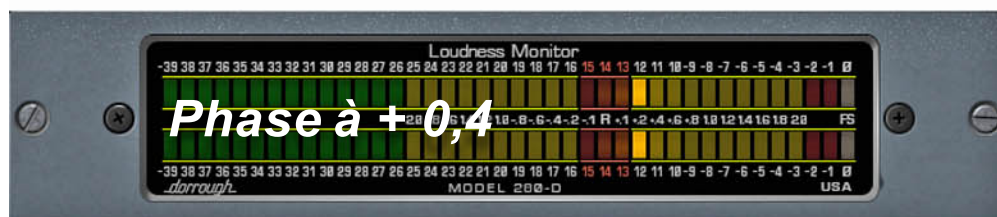
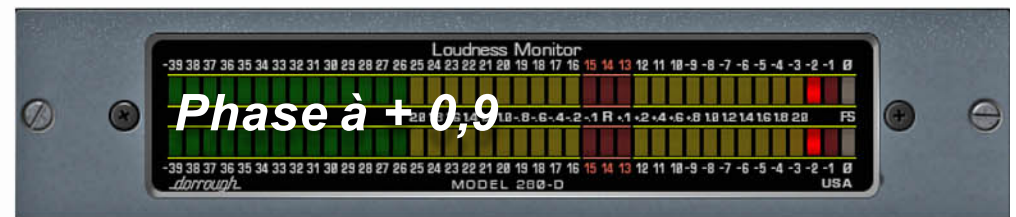
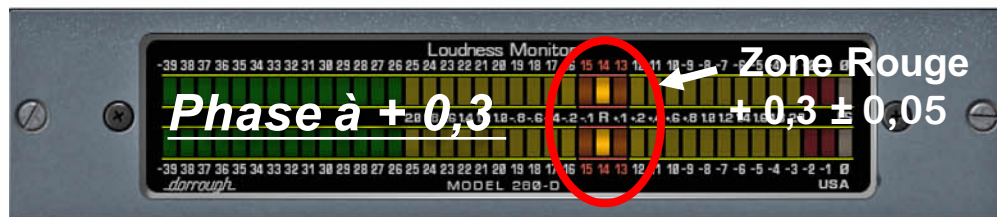
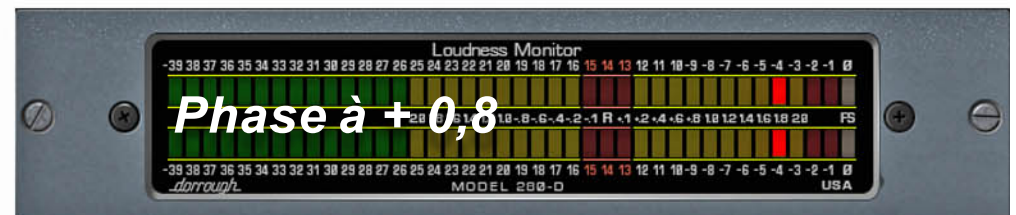
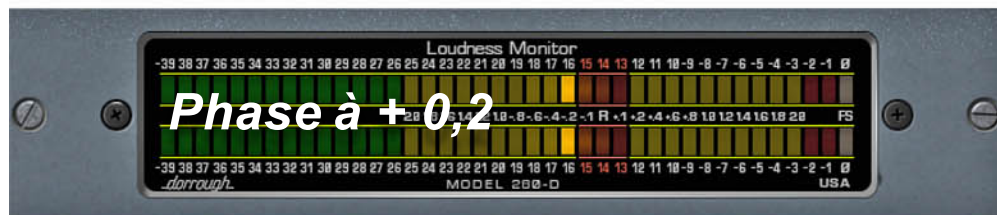
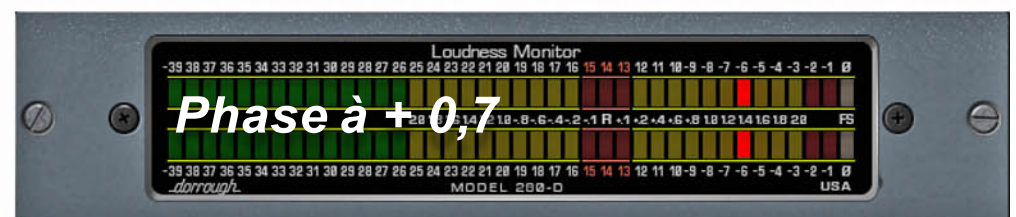
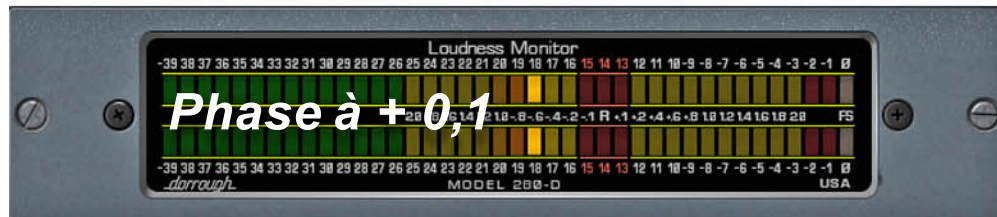
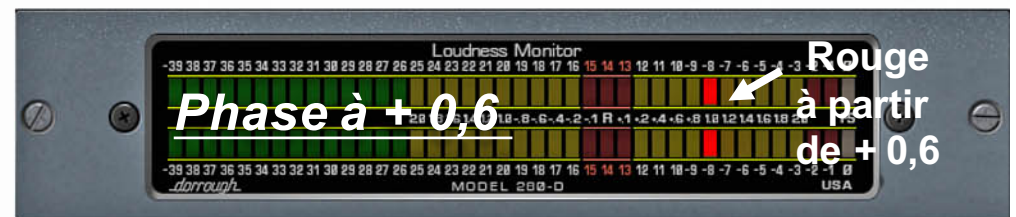
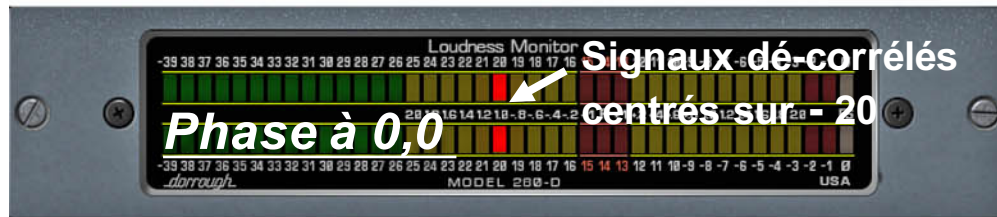
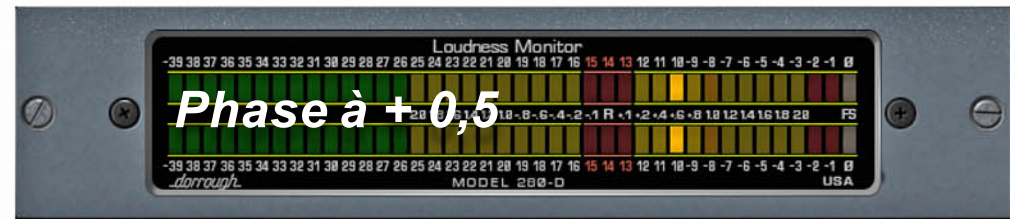


**+ 0,3 = répartition Stéréo homogène**

Étude psycho acoustique faite à Radio France sur du **bruit rose** : ( valable pour la musique classique et les ambiances )



# Indications linéaires de la phase sur le Plug-in Waves **Dorrough** Stéréo :







# Table of Contents

---

**Chapter 1 ..... Quick start .... 2**

**Chapter 2 ..... PAZ component plug-ins .... 3**

**Chapter 3 ..... The PAZ interfaces .... 4**

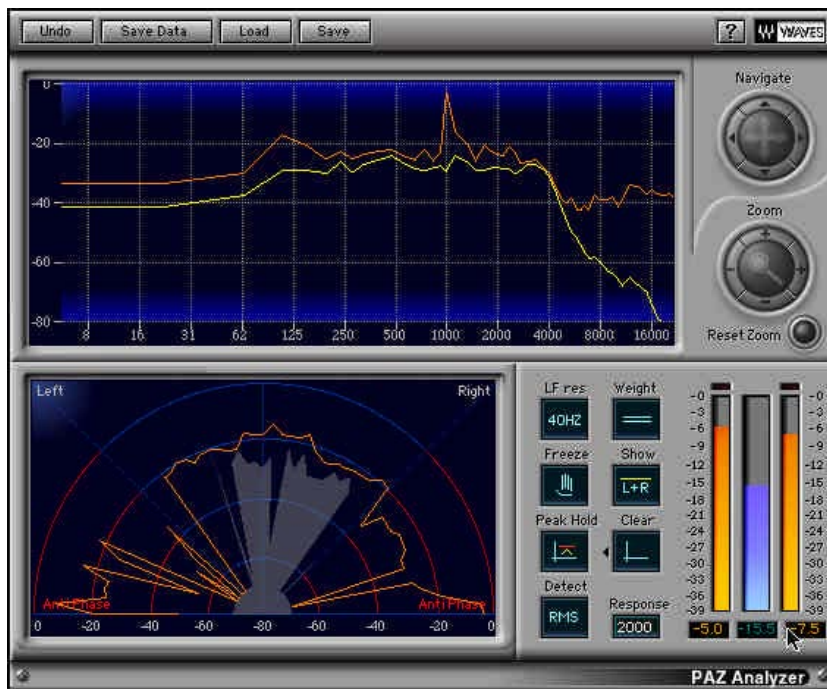
Frequency window (RTA) ..... 4

Stereo Position Display (SPD) ..... 6

Level meters ..... 7

**Chapter 4 ..... For fast graphics .... 8**

## Chapter 1 - Quick start



PAZ is so easy to use, you can almost work from the illustration shown above! However, for a more complete understanding of what the PAZ does and why we chose certain functions, please read this manual.

We suggest you use only the plug-in you need; this will give you the fastest graphic response time (if this is important to your use).

## Chapter 2 - PAZ component plugs

---

On most host applications and platforms you can select just the meter you need. In the case of the PAZ, no DSP savings are accomplished, but graphic speed is increased by using smaller component plug-ins. The PAZ has several plug-in components in the menu. They are all mono and stereo compatible and do not change the audio at all, as they are fully bit-transparent (24bit fixed point or 32bit floating point clean).

Here are the current components as of this printing. New ones might be added in future updates, and will be noted in the plug-in Read Me files. Some examples for each component are listed:

- **PAZ-Analyzer** — This is the full plug-in with all 3 measurement functions (Frequency, Stereo Position, and Level); it has the biggest demand on the computer's graphic power.
- **PAZ-Position** — Shows only the Stereo Position Display with the controls related to it.
- **PAZ-Frequency**— This shows only the real-time frequency analysis with related controls.
- **PAZ-Meters**— Shows only the meters for Left, Right, and Sum. The Sum meter can be selected for Peak or RMS display.



## Chapter 3 - The PAZ interfaces

### Frequency window (RTA)

Here is the PAZ frequency analyzer graph. The screen shot is taken from the PAZ-Frequency component. In other components, the row of buttons below the graph may be arranged differently, but they are functionally identical.



The PAZ frequency analysis is done in bands very similar to those of the ear. The default setting of the **LF resolution** control is 40Hz, which gives 52 bands, and most closely approximates the constant-Q critical frequency bands of the ear. The result is a good correlation between what we heard and what we see in the PAZ frequency analyzer.

For more detailed or technical uses, LF resolution below 250Hz can be set to 20 or 10Hz, providing up to 68 bands; the button's options are 40, 20, and 10Hz. Above 250Hz, the “engineering Q” (or width) of the bands are about 10.0, which as mentioned, are similar to the resolution of our hearing. The 40Hz LF default also closely matches the ears characteristics, plus has the best time response in the meter itself.

The **Peak/RMS** and **Response** controls let you choose the type of analysis and response time. In the **Peak** mode, peaks in each frequency band are displayed and the Response simply controls the release time. Under the RMS mode, the energy is averaged over time, and the Response value controls the length of this time.

Optimal time and frequency resolution in the PAZ is achieved by using wavelet techniques (as opposed to FFTs). This

lets each band update independently as fast as possible for its frequency resolution.

The real-time frequency analysis can be used for mono, dual channel, or total stereo energy by using the **LR** button, which switches between dual channel (**L/R**, independent channels) and total stereo energy (**L+R**). The L+R mode is not a simple summation of L and R signals, but of L and R energies. This more closely models our ears' responses to loudness of frequencies, as the gain and pan position of a sound are independent. The dual channel mode may be selected for precision analysis or troubleshooting.

The display can also be set for either of 3 weighting curves (dBA, dBB, or dBC), or unweighted. When unweighted, the absolute 'electrical energy' values are shown. Like in standard loudness meters, the weighting curves can be used for noise and loudness measurement applications that require them. Precise definitions of the monitoring levels for these weighting curves are beyond the scope of this manual, but in short, low frequencies offer little contribution to the overall loudness when monitoring at low sound levels. In other words, we are less sensitive to them. So an 'A' weighting is used which reduces the contribution of these LF to the meter reading. 'B' is for intermediate levels, and 'C' is nearly flat, for high level monitor loudness measurements. Unweighted is for precise, "what you see is what is there", measurements.

**Option-dragging** in the Frequency window will zoom the selected area. Click the **Reset Zoom** button to reset to maximum display of all frequencies and dynamic values.

The **crosshair cursor** can be shown by simply clicking once in the Frequency window. You can drag it around or click anywhere to show it. **Drag the crosshair to the upper corners** to hide the crosshair cursor.

At any time, the Frequency graph can be stopped by clicking the **Freeze** button. To release and return to realtime graphing, click the button again.

The maximum graphed value of the Frequency analysis can be shown by clicking the **Show/Hide** button. Maximum values are tracked even when in Hide mode. To reset the max graph line, click the **Clear** button.

**A note on Pink Noise measurements:** Pink noise is defined as noise that has constant energy inside any frequency band with a constant Q (i.e. frequency/bandwidth=a constant value). Pink noise is an important audio signal since it represents the average spectrum of a typical audio signal, and because the ears analyze the sound in approximately constant-Q bands. Because PAZ has almost constant-Q bands, a pink noise input will show up as a flat spectrum when no weighting is applied.

## Stereo Position Display (SPD)

The SPD shows you the stereo energy in your mix, clearly and intuitively. This unique audio meter analyzes the distribution of loudness across the stereo stage, as well as detecting out-of-phase components. It differs from the common ‘phase display’ where left/right channels are displayed against each other on an oscilloscope screen.

For those familiar with the Waves S1 Stereo Imager, the Stereo Position Display will seem familiar, as both use a polar coordinates graph. Loudness is shown by length of the line and stereo position is shown by the angle (direction). The concentric lines are 20dB step indications.

In other words, a mono signal goes straight up (as will anything in the center of a stereo mix). A hard left or right sound will go 45 degrees to the respective sides. Most stereo mixes will have a fan-shaped display centered around the vertical line with occasional little spikes shooting out at greater angles.



Out-of-phase signals are shown between 45 and 90 degrees on each side. In the area beyond 60 degrees on each side have a red color in the background to indicate that signals in this range might be disturbing to the ears.

## The controls

They are the same for the SPD as for the Frequency Display, except that when using the component Frequency plug-in, only the related controls are shown.

### About Anti-phase

It is a bit tricky to define when a signal is shown by the Stereo Position Display or heard by the ears as “out-of-phase”. This is not simply when the left and right channels happen to have opposite polarity at any given instant. Such a situation can happen if independent instruments are panned to different stereo positions, and it just happens, statistically, that the signal of each channel has opposite polarities. The meter will show a quick spike in the anti-phase area. This can also happen if a stereo delay or reverb is applied to an instrument and the delays arrive to each

channel at different times.

In these examples, the ears do not hear a disturbing ‘out-of-phase’ sound. For us to hear an out-of-phase sound, there generally needs to be a consistent opposite polarity over a period of time, which the meter will show.

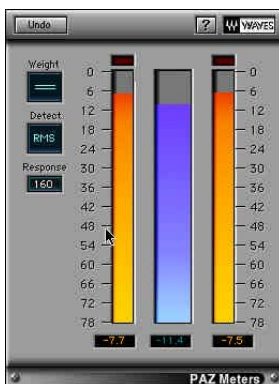
The Stereo Position Display analysis takes this into account. Still there will be occasional fast spikes in the anti-phase region, and you can ignore them if they are not consistent, especially if you know you are using stereo delay or similar effects. You might be really bothered if the energy in the out-of-phase region is consistently significant compared with the energy of the ‘in-phase’ region.

It is advised that you take the time to explore how the SPD looks when you intentionally flip the phase on one channel of a stereo signal. (You can do this by using the Q1 or one of the S1 components in front of the SPD).

## Level meters

**Peak** and **Loudness** meters are shown beside each other; when in stereo, the L and R channels have separate Peak meters, with a summed Loudness meter which has 6dB subtracted from it (L+R-6dB). Although the true energy the ears are exposed to by stereo signals is indeed the sum of L and R energies, by subtracting 6dB from the loudness meter, it’s easier to evaluate the Peak/RMS relations of the signal. This is because summed left/right signals can have peaks as much as 6dB higher than either channel separately.

You can select **Peak** or **RMS** measurement for the Loudness meter, which is also affected by the selected **weighting curve** (unweighted, dBA, dBB, dBC).



For those of you who own the L1-UltraMaximizer or the L2 hardware Ultramaximizer, it will be striking to see what it does to the Peak-to-RMS measurements of your audio.

Each of the 3 meters have **peak-hold values** at the bottom of the meter. Each can be **independently reset** by simply clicking directly on the peak hold value.

**Clip lights** are only provided on the full plug-in, not in the component Meter. They can be reset by clicking directly on the light.



## Chapter 4 - For faster graphics

---

### Fast graphic hints

---

Lower the number of colors in your Monitors control panel (256 instead of thousands or millions).

Use only the component of the Analyzer that you really need. For instance, if all you need is the frequency analysis, then use only the “PAZ-frequency” plug-in. The full plug-in with all 3 meters will have the heaviest demand on the host computer.

Under ProTools, if you click and hold on any control, such as the Release control, the Analyzer graphic will take all the computer “idle time” for graphic updates. This will freeze all Pro Tools meters, but will not affect any functions of Pro Tools (such as playback or recording).

# A Spherical Far Field HRIR/HRTF Compilation of the Neumann KU 100

Benjamin Bernschütz<sup>1,2</sup>

<sup>1</sup>*Cologne University of Applied Sciences - Institute of Communication Systems, Germany*

<sup>2</sup>*Berlin University of Technology - Audio Communication Group, Germany*

benjamin.bernschuetz@fh-koeln.de

## Introduction

A full sphere far field HRIR/HRTF (head related impulse response and its Fourier transform head related transfer function) [1] compilation of a Neumann KU 100 dummy head is presented. The compilation can be used for different purposes; it is, for instance, very convenient for applications in the field of spherical Fourier acoustics [2]. Different sampling configurations were captured and the compilation on the whole involves around 22.000 post-processed binaural impulse responses. The data is organized and stored in MATLAB® objects, offering different processing features and including gateways to the SOFiA sound field analysis toolbox [3] and to the Sound Scape Renderer [4]. Associated headphone compensation filters for around 20 common headphones were created. Additionally, a method to optimize and extend the low frequency response of measured HRIRs/HRTFs is discussed.

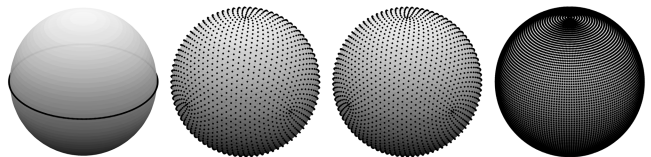
## HRIR Measurements

The HRIR measurements were conducted in the anechoic chamber at Cologne University of Applied Sciences. The respective chamber has dimensions of 4.5 x 11.7 x 2.3 m and a lower frequency boundary of around 200 Hz. All surfaces are absorptive. A custom mount was constructed to install the Neumann KU 100 dummy head on the highly accurate VariSphear motion system [5], offering motor position deviations  $< 0.01^\circ$ . The resulting orientation entailed a transform of the usual VariSphear coordinate system. The impulse response measurements and the motion control were performed using the proprietary VariSphear software. The measurement core of the software has an integrated error detection stage to ensure the validity of all measured impulse responses. An emphasized sine sweep with +20dB low-shelf at 100 Hz of  $2^{19}$  samples at 48kHz temporal sampling rate was used for excitation. The long sweep duration (approx. 11 s) made the measurements robust against background noise and entailed a good overall signal to noise ratio. A RME Fireface UCX including its internal low-noise microphone preamps and AD/DA converters was used as audio interface. The sound source was a Genelec 8260A active 3-way speaker system. The speaker was driven around -3dB below the internal limiter threshold. No relevant climate changes in the chamber or time-variance of the speaker system due to driver heating were observed during the session. The temperature was constantly tracked and a static microphone was used to detect time variances. The frequency response of the speaker at the

physical origin (center of the head) was tuned to be flat within a tolerance of less than  $\pm 1$ dB for the entire involved spectrum using a digital signal processor. The speaker system was high-passed around 60 Hz in order to reduce the power load. This was feasible as the final low frequency components were generated during the post processing. The physical origin was placed in the vertical acoustic center of the speaker, where the phase response of the single speaker components is perfectly aligned. The head position adjustment was conducted with high accuracy using a cross-grid laser combined with real-time analysis of the phase response differences at the ear channels. The distance between speaker system and center of the head was approximately 3.25 m, which can be considered as far field referring to the speaker dimensions.



**Figure 1:** Genelec 8260A speaker and Neumann KU100 dummy head mounted on the VariSphear motion system [5] in the anechoic chamber. [Photo: P. Stade]



**Figure 2:** Spatial sampling configurations: Circular  $1^\circ$ , Lebedev 2354, Lebedev 2702 and Gauss-Legendre  $2^\circ$ .

Several different sampling configurations were captured. First of all, two circular horizontal turns with an angle resolution of  $1^\circ$  were run. The first circle was recorded using a simple thin microphone stand on the rotating ground-plane and the second circle using the complete 3D robot arm and rotation mount that was also used for all subsequent spherical grids. Two different equidistant spherical Lebedev grids (2354 and 2702 nodes) were recorded. These are convenient for typical applications in spherical acoustics. The grids were simulated in advance using the SOFiA sound field analysis toolbox [3]. Both of the presented Lebedev grids enable a stable transform

into the spherical wave spectrum domain (spherical Fourier transform based on surface spherical harmonics) [2] with low contribution of spatial aliasing on the entire audio spectrum up to  $>20$  kHz on the KU100 head radius. Finally a full sphere equiangular  $2^\circ$  Gauss quadrature with 16020 nodes was captured. There are no grid discontinuities as all grid nodes could be measured. This is particularly relevant in spherical acoustics. In addition to the HRIRs, single impulse responses were captured at the physical origin using an omnidirectional Microtech Gefell M296S microphone.

### *miro* Data Format

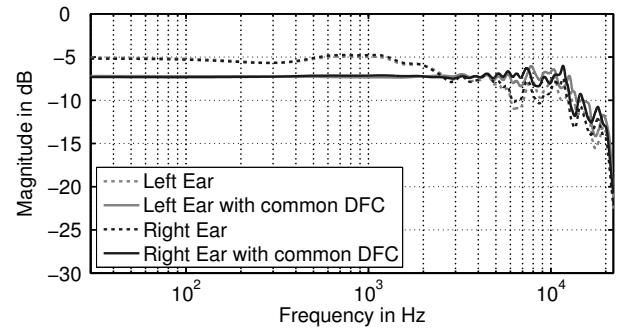
A proprietary data type was developed in order to store, organize and access the datasets in a comfortable way. It is a simple object based MATLAB® data type called *miro* (measured impulse response object); *miro* integrates three stages: It combines the storage of raw impulse responses, the access to around 50 meta information properties and a method set for the access to the impulse response as well as for the treatment and conversion of the signals. It offers interfaces to the SOFiA sound field analysis toolbox [3] for spherical harmonics processing and to the Sound Scape Renderer [4] for dynamic binaural synthesis.

### Headphone and Diffuse Field Filters

Even headphones and dummy heads explicitly advertising free field equalization [6] do usually still not meet the requirements for a highly natural and color-free reproduction that is needed for an immersive binaural listening experience. Hence the frequency response of the headphones, respectively the full transducer chain, must be explicitly compensated by application of appropriate compensation filters [7]. In order to make this highly specific compensation filters accessible to a broad range of different users, around 20 different common headphone models were analyzed to create specifically adapted compensation filters. To create the filters, each headphone was put on the KU100 head and replaced 12 times in order to capture stable and representative transfer functions. Additionally, a common free field compensation filter has been derived from spherical magnitude averaging, compare Figure 3. This filter compensates the dummy head itself and does not refer to a specific headphone. The compensation filters were computed based on a semi-automatic log-spline inversion algorithm. The proprietary *miro* data type enables a direct inline processing of the headphone and free field compensation filters.

### The low frequency range of HRIRs

In a system for binaural synthesis, source signals are convolved with HRIRs. Hence all magnitude and phase properties of the HRIRs are directly imprinted on the resulting audio signal. Besides the desired magnitude and phase properties that are inherent to binaural techniques [1] [8], additional undesired parasitic magnitude and phase changes may arise which originate from the process of capturing the HRIRs. Especially at the



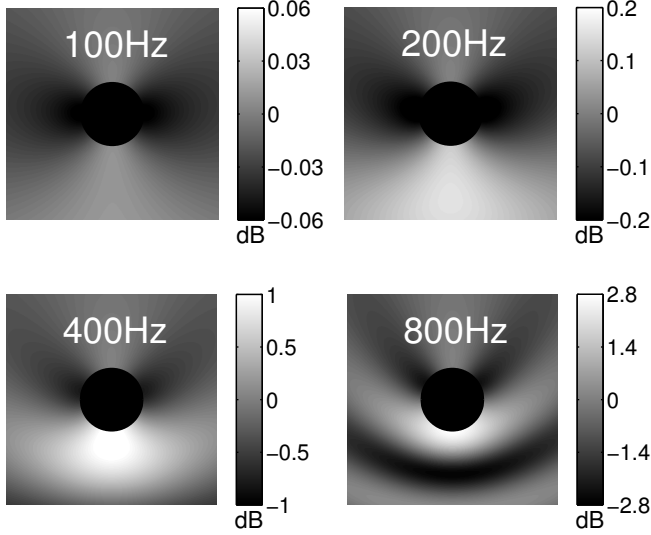
**Figure 3:** Averaged and weighted magnitude responses over the full sphere for the left and the right ear with and without the common diffuse field compensation filter (common DFC).

low frequency end, below 100 Hz for instance, it is difficult to capture HRIRs without undesired artifacts due to several reasons. Small studio monitors that are typically used for HRIR measurements, mostly cannot reproduce frequencies e.g. below 60 Hz with sufficient sound pressure level. This lack of low frequencies cannot be compensated using filters or equalizers. As a consequence, the measured HRIRs cannot transmit low frequencies either. A possible solution is to use a bigger and more powerful speaker system, which otherwise brings along an increased cabinet sizes and larger spatial distributions of the single drivers. Especially for near-field measurements this is not desirable. A less obvious problem of speaker systems is the typical surge of group delay towards low frequencies. The group delay rises due to different mechanical reasons, bass-reflex constructions and filter networks [9]. Besides of the question at which dimensions group delay distortions lead to any noticeable impairment of the audio signal [10], there is a simple and very practical problem with it: Group delay differences directly come along with a spread of the time domain signal. Thus the HRIRs need more filter taps to transmit the full audio spectrum and demand for more computational power. While the problems concerning speaker systems are widely solvable with some effort, the anechoic chamber can bring up much more fundamental problems due to finite room dimensions and the limited length of absorption wedges. Below certain boundary frequency, reflections and room modes arise. As the low frequencies do not contribute significantly to localization [8], the respective reflections do usually not directly affect the binaural hearing. But reflections and room modes entail an overlay of the direct sound with reflected waves at the transducers. Depending on the phase relation this can lead to an amplification or to cancellation of the respective frequency. This has dramatic influences on the frequency response of the system. Furthermore room modes bring along a very excessive surge of group delay, as the decay of energy is slow. If these issues are not considered and the effects are eliminated, the measured HRIRs do not transmit low frequencies properly.

### Adaptive low frequency extension (LFE)

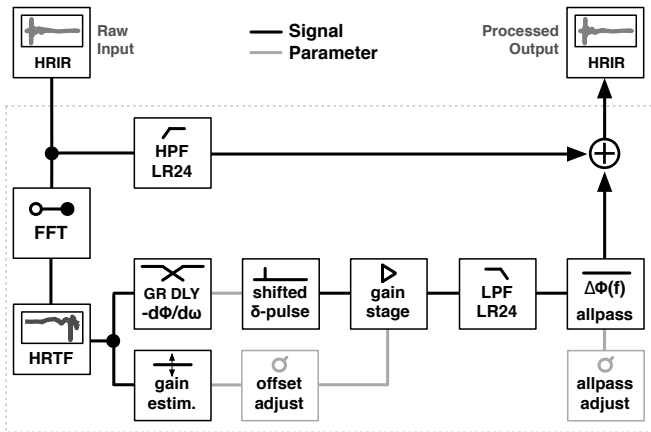
In order to avoid negative influences on the low frequencies of the measured HRIRs, the entire low frequency

range is replaced by an analytic extension. A similar approach has been presented in [11]. At high frequencies the head as a rigid body evokes shading and scattering of the sound field and the pinna and ear canal work as filters [1] [8]. Whereas at frequencies below e.g. 400 Hz, pinna and ear canal filters do barely influence the signal and the head itself does only have minor influence on the sound field. Analytic simulations of a plane wave



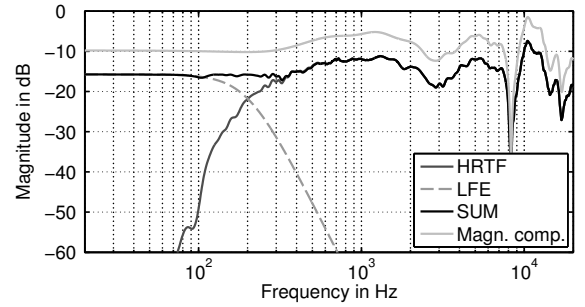
**Figure 4:** Analytic simulation of the pressure magnitude at different frequencies for a plane wave impact with unit gain from south to a rigid sphere with a typical diameter of 17.5 cm serving as a simplified model of the dummy head. Below the LFE crossover frequency of 200 Hz, the magnitude deviations on the sphere surface are already less than  $\pm 0.2$  dB.

impact on a rigid sphere as a simplified model of the dummy head (Figure 4) indicate that there are only neglectable differences in the pressure magnitude around the sphere below 200 Hz. Thus it is feasible to replace the low frequency band using simplified analytic descriptions including a flat frequency response. An adaptive low



**Figure 5:** Basic structure of the adaptive LFE algorithm. frequency extension (LFE) algorithm has been employed that generates the low frequency component and matches gain and phase, see Figure 5. Each HRIR and each ear is processed independently. At certain crossover frequency (200Hz) the original HRIR is high-passed

(24 dB/oct Linkwitz-Riley) and a matched low frequency path is added. To generate the LFE path, the HRIR is transformed into a HRTF using the Fourier transform. The group delay of the original HRTF is evaluated around the crossover frequency and a shifted  $\delta$ -pulse is generated accordingly. The required gain for the pulse is estimated by an analysis of the original HRTF gain around the crossover frequency on the one hand and either evaluation of single selected stable bins at lower frequencies or the simulation of a sphere model on the other hand. This is done as besides residual shading effects, even in the far field the different ear distances to the source bring along certain attenuation of the sound pressure level. This effect obviously becomes much more important for near-field measurements. In a next



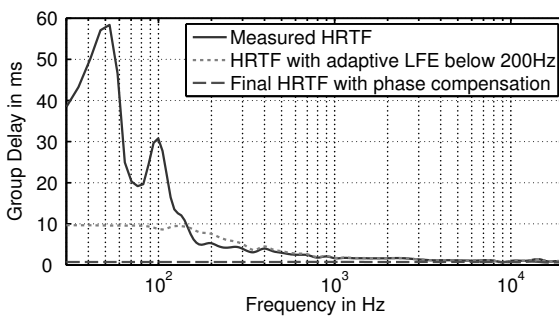
**Figure 6:** Post-processed HRTF split into the original high-passed HRTF path and the adaptive LFE path. Additionally the final magnitude compensated HRTF is depicted.

step the impulse is lowpass filtered. Finally an allpass filter is employed in order to match the phase slope of the original high-passed HRTF path and the LFE path around the crossover frequency. Out of this range, the LR24 crossover filters attenuate the signal sufficiently. The resulting phase drop of the generated signal is much lower than the phase drop of the original measured HRIR below the crossover frequency. This implies a lower group delay, as depicted in Figure 7. Hence the resulting HRIRs have optimized properties concerning frequency response and group delay. In order to eliminate some remaining minor reflection contributions from the floor framework and the VariSpear base plate, the back section of the HRIRs was slightly smoothed using floating average filters. The HRIRs at that point could directly be employed without further processing besides head and tail windowing. The windowing is performed in the processing core within the *miro* data type.

## Magnitude and Phase Compensation

In a final optimization step the remaining group delay differences  $\Delta\tau_{gr}$  were removed and the minor frequency response deviations (around  $\pm 1$  dB) of the speaker system were compensated. For that purpose a specific FIR (finite impulse response) filter was designed and applied. FIR filters generally admit to design magnitude and phase responses independently. The respective filter was derived from the post-processed (LFE) center impulse response by appropriate magnitude and phase inversions and windowing in the time domain similar to [12]. The inherent non-causal portions of the FIR compensation

filter clear away in the processed signal and the ensuing overall latency can be easily removed. The influence of the compensation filter on the magnitude response of an exemplary HRTF can be observed in Figure 6. The phase compensation has a direct impact on the group delay that is depicted in Figure 7. At this point all substantial magnitude and phase influences of the speaker and the chamber are eliminated and the final HRIRs are contracted to the tightest possible time domain signal. The HRIRs were reduced to 128 taps while maintaining the full spectral bandwidth. (Whereas already around 30 taps are covering the sonic runtime difference between the ears only.) Besides an improved signal integrity due to the phase compensation, the reduced number of filter taps directly entails decreased computational demands for the convolution.



**Figure 7:** Group delay  $\tau_{gr}$  of a measured and a post-processed HRTF. The measured HRTF shows a strong surge of  $\tau_{gr}$  according to the speaker system and additional room resonances in the anechoic chamber at low frequencies (50, 100 Hz). Besides a little increase of  $\tau_{gr}$  around the crossover frequency due to the filters, the processed HRTF has a considerably reduced  $\Delta\tau_{gr}$ . The final phase compensated HRTF has a nearly constant  $\tau_{gr}$  and hence  $\Delta\tau_{gr} \approx 0$  leading to a perfectly tight time domain signal.

## Conclusion

A free spherical far field HRIR/HRTF compilation of the Neumann KU100 dummy head including several headphone compensation filters is presented. The compilation is useful for applications in spherical acoustics as different dedicated sampling grids are included. It is suitable for high quality audio applications and music production due to the excellent signal quality and the wide frequency response. Owing to the phase compensation, the final HRIRs could be reduced to 128 filter taps while maintaining the full bandwidth. This saves computational power and makes the HRIRs applicable e.g. in mobile devices, where processing power is a highly important issue.

## License and Access

The compilation is freely available under a Creative Commons CC BY-SA 3.0 license and can be downloaded at: <http://www.audiogroup.web.fh-koeln.de>

## Acknowledgements

The author likes to thank Philipp Stade for supporting the headphone measurements and for taking the pictures. Thanks to the MUSIC STORE professional

GmbH in Cologne for the hospitality and for providing the headphones to create several compensation filters. Thanks to Markus Haßler from the WDR radio production department for providing the speaker system. The author particularly thanks Christoph Pörschmann, professor for acoustics and audio signal processing at Cologne University of Applied Sciences and head of the research group for his valuable contributions and ideas, and for making all of the presented work possible.

## Funding

The research activities are funded by the Federal Ministry of Education and Research in Germany. Support Code: 17009X11 MARA. We appreciate the support.

## References

- [1] Møller H., "Fundamentals of binaural technology," In: *Applied Acoustics* 36, pp 171-218, 1992.
- [2] William E.G., "Fourier Acoustics: Sound Radiation and Nearfield Acoustical Holography," Academic Press, San Diego, 1999.
- [3] Bernschütz B., Pörschmann C., Spors S. and Weinzierl S., "SOFiA Sound Field Analysis Toolbox," In: *Proceedings of the International Conference on Spatial Audio (ICSA)*, pp 7-15, 2011.
- [4] Geier M., Ahrens J. and Spors S., "The Sound Scape Renderer: A Unified Spatial Audio Reproduction Framework for Arbitrary Rendering Methods," In: *Proceedings of the 124th Convention of the AES*, Convention Paper 7330, 2008.
- [5] Bernschütz B., Pörschmann C., Spors S. and Weinzierl S., "Entwurf und Aufbau eines sphärischen Mikrofonarrays für Forschungsanwendungen in Raumakustik und Virtual Audio," In: *Forschritte der Akustik (DAGA)*, pp 717-718, 2010.
- [6] Theile G., "On the Standardization of the Frequency Response of High-Quality Studio Headphones," In: *Journal of the AES*, Volume 34(12), pp. 956- 969, 1986.
- [7] Schärer Z. and Lindau A., "Evaluation of Equalization Methods for Binaural Signals," In: *Proceedings of the 126th Convention of the AES*, Preprint No. 7721, 2009.
- [8] Blauert J., "Spatial Hearing," The MIT Press, Cambridge/MA, 1997.
- [9] Goertz A., "Lautsprecher," In: *Handbuch der Audiotechnik*, Weinzierl S. (ed.), pp 421-490, 2008.
- [10] Blauert J. and Laws P., "Group Delay Distortions in Electroacoustical Systems," In: *Journal of the Acoustical Society of America*, Volume 63, Number 5, pp 1478-1483, 1978.
- [11] Xie B., "On the low frequency characteristics of head-related transfer function," In: *Chinese Journal of Acoustics* 28(2), pp 1-13, 2009.
- [12] Müller S., "Digitale Signalverarbeitung für Lautsprecher," Doctoral Dissertation, RWTH Aachen University, 1999.



**miro :: measured impulse response object**

data type description

**Technology  
Arts Sciences  
TH Köln**

## Document Information

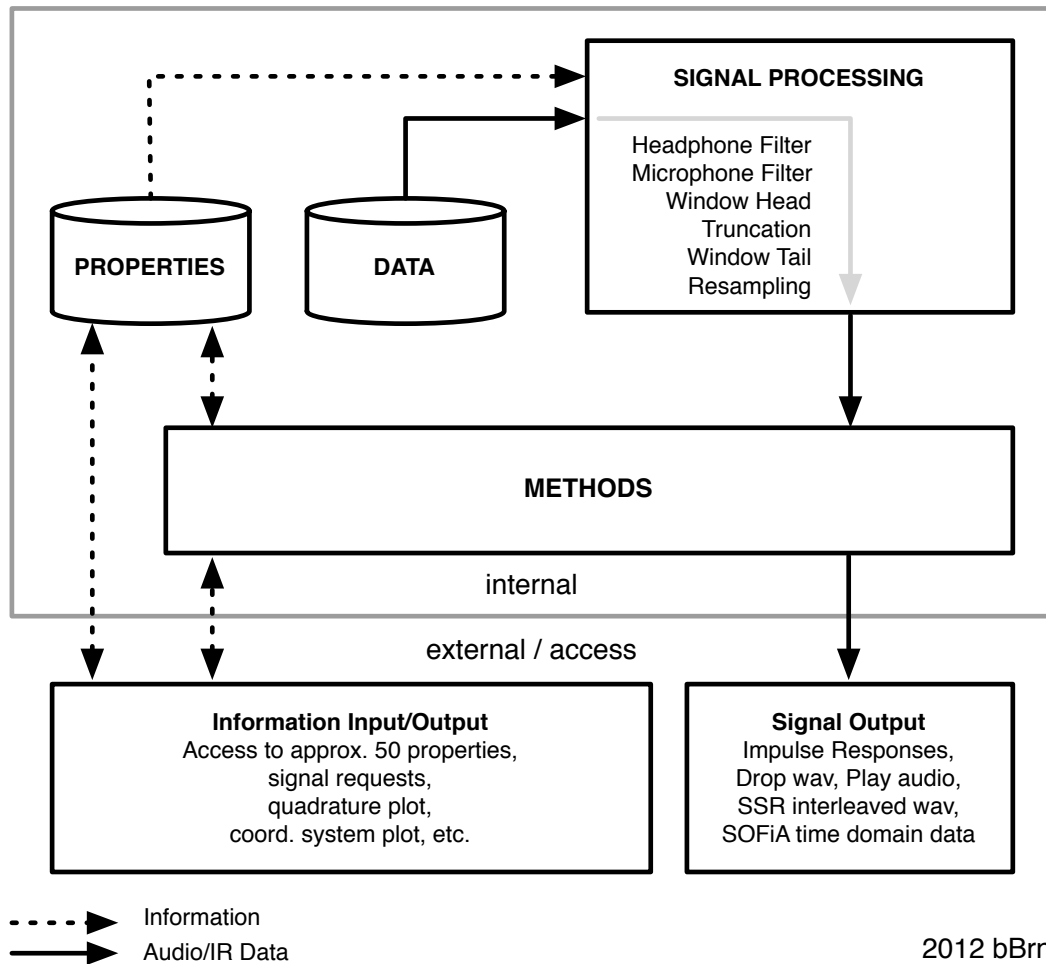
Author	Benjamin Bernschütz Johannes M. Arend Christoph Pörschmann
Source	<a href="http://www.audiogroup.web.th-koeln.de">http://www.audiogroup.web.th-koeln.de</a>
<i>miro</i> Version	1.06
Document Revision	1.3
Date	20/September/2017
Institution	TH Köln - University of Applied Sciences, Institute of Communications Engineering, Betzdorfer Str. 2, 50679 Cologne, Germany
Mail	christoph.poerschmann@th-koeln.de
Phone	+49 221 8275 2495

## Introduction

*miro* is a simple object oriented data type for the storage and handling of measured audio impulse responses, especially designed e.g. for complex microphone array or rotated dummy head datasets. *miro* works under MATLAB® and has been developed for the storage and distribution of the WDR Spatial Audio Impulse Response Collection and the Neumann KU100 spherical HRIR datasets that were captured by the audio group of Cologne University of Applied Sciences during the summer of 2012. But the *miro* datatype will also be used for future work and can naturally be used and modified/optimized by the audio community. Each measurement position or session (e.g. microphone array or rotated dummy head measurement) is stored in a separate *miro* instance. The *miro* class combines three basic elements:

1. **PROPERTIES:** Description and detail information on the content and the measurement session that are stored in the properties.
2. **METHODS:** Basic methods for the extraction and treatment of the measured data.
3. **DATA:** Measured Data (Raw Impulse Responses).

## *miro* datatype overview



### Important general information:

- Besides to the object file itself, MATLAB® needs to have access to the *miro* class definition that is stored in the file *miro.m*. This file must be put in the object's directory or to an accessible MATLAB® searchpath.
- Miro has a basic signal-processing core (inside the *obj.getIR* method). All impulse responses run through this core before they are returned, written to a file or played back. Thus the processing settings are globally valid and affect all impulse responses

automatically. The processing blocks (headphone filters, windowing/truncation, resampling) can be bypassed or adapted via the properties.

- Every impulse response has a specific ID (*irID*). A stereo impulse response (e.g. from a dummy head) has two channels but one single *irID* only. The numbering/indexing of the logical contiguous impulse response sets starts at *irID* = 1.
- Every *miro* instance contains a corresponding center impulse response that is captured using an omnidirectional microphone (*obj.centerMicrophone*) at the physical center (origin) of e.g. the microphone array or dummy head. The center impulse response can be addressed via *irID* = 0. Be careful to distinguish this particular ID.
- For the ease of use, all main methods and properties are intentionally set *public*<sup>1</sup>.
- *Miro* is a *Value Class* by default (recommended), but can be changed to a *Handle Class* if needed. For more information on that topic refer to the *Mathworks* website.
- The default angles are in RAD but the object can be set to operate in DEG.

---

<sup>1</sup>Thus the risk of misuse and damaging an object is high. But this is not an important problem in the present case as the original objects can always be reloaded if the current instance is broken for some reason. If the properties and methods were set private to offer an increased protection, the objects would become much more complex to use and would require e.g. specific getter and setter methods for every property. This is not very convenient in the present case. Easy handling was rated to be more valuable than a high protection level here.

## Properties

Properties	Description	Type
name	name/description of the IR set	string
context	identifier of a larger measurement session	string
location	recording location	string
date	recording date	string
engineer	recording engineer	string
contact	email/telephone for requests	string
comments	comment(s)	string
miroVersion	version of the miro class definition	1x1 single
type	$\{HRIR, BRIR, MICARRAY, SINGULAR\}$	string
fs	audio sampling rate in Hz	1x1 single
taps	number of taps	1x1 single
nIr	number of impulse responses in dataset	1x1 single
excitationSignal	excitation signal	string
gapTime	gap time in s	1x1 single
microphone	manufacturer/type microphone or dummy head	string
source	manufacturer/type source loudspeaker	string
audioInterface	manufacturer/type audio interface	string
micPreamp	manufacturer/type mic preamp	string
capturingSystem	manufacturer/type of the capturing system	string
systemLoopLatency	electrical system I/O loop latency	1x1 single
latencyCompensated	$\{true, false\}$ system loop latency compensation	1x1 bool
headCut	number of empty leading head samples that were cut	1x1 single
sourcePosition	may contain coordinates or verbal description e.g. 0AZ, 90EL or Left, Center	string
sourceDistance	distance between microphone and source in m	1x1 single



avgAirTemp	average air temperature in °C	1x1 single
avgRelHumidity	average relative air humidity in %	1x1 single
positionReference	<i>{Virtual Source, Head Rotation, Microphone}</i>	string
postProcessing	description of the post processing	string
nomalization	normalization factor (1=no normalization)	1x1 single
ctIRnomalization	normalization factor (1=no normalization)	1x1 single
quadGrid	name/description of the quadrature grid	string
scatterer	<i>{true, false}</i> (for a dummy head set true)	1x1 bool
radius	microphone radius/radii in m	1x[1,2] single
azimuth	position azimuth	[1xnIR]
elevation	position elevation	[1xnIR]
quadWeight	quadrature weighting	[1xnIR]
chOne	content description e.g. <i>Left Ear</i>	string
chTwo	content description e.g. <i>Right Ear</i> or []	string
irChOne	impulse responses	[tapsxnIR]
irChTwo	impulse responses or []	[tapsxnIR]
centerMicrophone	manufacturer/type of the omni center microphone	string
irCenter	omni center impulse response at the coordinate origin	[tapsx1]
returnTaps	number of taps to be returned ( $\leq$ taps)	1x1 single
resampleToFS	target sampling frequency, [] = no resampling	1x1 single
headWin	head window length	1x1 single
tailWin	tail window length	1x1 single
headPhone	manufacturer/type of the headphone	string
hpcfKernel	headphone compensation filter kernel	string
headPhoneComp	true, false enables/disables HP compensation	1x1 bool

mic	manufacturer/type of the microphone	string
mcfKernel	microphone compensation filter kernel	string
micComp	true, false enables/disables MIC compensation	1x1 bool
shutUp	true, false if set true no console messages are printed	1x1 bool
angles	RAD, DEG RAD is default	string

## Methods

```
[ir, azimuth, elevation, quadWeight] = getIR(obj, irID)
```

Returns the impulse response, angles and the quadrature weight for a specific ID number *irID*. The returned impulse response can have one or two channels depending on the respective object. This method involves the signal processing core.

```
[irID, azimuth, elevation] = closestIr(obj, az_approx, el_approx)
```

Returns the ID number *irID* of the closest angle to *az\_approx*, *el\_approx* that is available within the object. *azimuth* and *elevation* return the corresponding closest fitting angles. The angles can be defined either in *RAD* or *DEG*, depending on the *obj.angles* status.

```
quadrature = getQuadrature(obj)
```

Returns the quadrature including azimuth and elevation angles and weights.

```
[] = plotQuadrature(obj)
```

Shows a sphere plot of the quadrature.

```
[] = miroCoordinates(obj)
```

Illustrates the coordinate system that is used by *miro*.

```
obj = setDEG(obj)
```

Changes the object's angle reference to *DEG*. All angle handling is then in *DEG*. (The default angle reference is *RAD*.)

```
obj = setRAD(obj)
```

Changes the object's angle reference to *RAD*. All angle handling is then in *RAD*. (The default angle reference is *RAD*.)

```
obj = setReturnTaps(obj, returnTaps, [tailWin])
```

Allows for changing the number of returned impulse response taps. By default, all available taps are returned (*obj.returnTaps* = taps). Setting the *returnedTaps* property to a different value will cause that all returned impulse responses are cut off at the value defined by *obj.returnTaps* and are by default windowed using a half-sided Hann window of the size (*obj.returnedTaps*/8). The size of the window can be defined by *tailWin*. Using *tailWin* = 0 turns off windowing.

```
obj = setResampling(obj, targetFS)
```

Sets the *resampleToFS* property and can be used to extract the IRs at a different audio sampling rate. *targetFS* defines the target sampling rate, e.g. 44100Hz or 96000Hz. If called without the *targetFS* argument, the *obj.resampleToFS* property is set to [] (default) and disabled. The resampling process is done within the *getIR* method after truncation and windowing. The returned IRs then do NOT have the amount of samples set in *returnTaps* property as these refer to the original FS. The resampling is based on the native MATLAB® *resample()* method which is included in the Signal Processing Library. **ADVICE:** If not urgently necessary try to avoid resampling.

```
dropWaveFile(obj, irID, [nBits], [filename])
```

Drops a wave audio file containing the impulse response for a specific ID number *irID*. The arguments *nBits* and *filename* are optional. Defaults: *nBits* = 16, *filename* = *obj.name*, '\_IR', *num2str(irID)*, '-AZ', *az*, 'EL', *num2str(el)*, *obj.angles*, [*obj.headphone*]. No dithering/noiseshaping is applied. Depending on the MATLAB© version, the function *wavwrite* or *audiowrite* is used to write the wave audio file.

```
[] = playAudio(obj, irID, audioSignal)
```

Plays an audio signal convolved with the preprocessed impulse response(s) given by *irID*. The method serves for a quick pre-listening of the datasets. If the signal *audioSignal* has more than one channel, the first channel is taken.

```
obj = setHeadPhones(obj, hpFilter, [linearPhase])
```

Sets headphone compensation filters for HRIR and BRIR datasets. Once the kernel is loaded the filters are applied to all outgoing impulse responses as long as *obj.headphoneComp* is set *true*. The *linearPhase* flag {*true*, *false*} can be set *true* to enable linear phase compensation (Careful: High Latency). By default the flag status is *false* and a minimum phase filter is applied.

```
obj = setMic(obj, mcFilter, [linearPhase])
```

Sets microphone compensation filters for HRIR and BRIR datasets. Once the kernel is loaded the filters are applied to all outgoing impulse responses as long as *obj.micComp* is set *true*. The *linearPhase* flag {*true*, *false*} can be set *true* to enable linear phase compensation (Careful: High Latency). By default the flag status is *false* and a minimum phase filter is applied.

```
[ ] = miroToSSR(obj, [mirror], [nBits], [filename], [normalize])
```

Writes a 720-channel interleaved wave file for the Sound Scape Renderer (SSR). The SSR can e.g. be used for dynamic binaural synthesis.

SSR Website: <http://spatialaudio.net/ssr/>

The object must be a circular HRIR or BRIR set to be exported to an SSR wave file. Defaults: *mirror*  $\{true, false\} = false$  (This option can be useful for symmetrical venues to mirror a source from left to right or vice versa.), *nBits* = 16, *filename* = 'SSR\_', *obj.name*, [*obj.headphone*], *normalize*  $\{true, false\} = true$ . No dithering/noiseshaping is applied. To write the multichannel wave file, the internal *miro\_wavwrite* function is used, which allows to write a wave file with an arbitrary (high) number of channels.

```
[timeData1,timeData2] = miroToSOFiA(obj)
```

Returns a struct that is readable by the F/D/T function of the SOFiA sound field analysis toolbox. The following S/T/C spatial transform core transforms the object's data into the spherical harmonics domain.

SOFiA Website: [http://audiogroup.web.th-koeln.de/SOFiA\\_wiki/WELCOME.html](http://audiogroup.web.th-koeln.de/SOFiA_wiki/WELCOME.html)

## Examples/Tutorial

Application examples are available at <http://audiogroup.web.th-koeln.de>.



# Boundary element method modelling of KEMAR for binaural rendering: Mesh production and validation

KAT YOUNG<sup>1</sup>, TONY TEW<sup>2</sup> AND GAVIN KEARNEY<sup>3</sup>

<sup>1-3</sup>AudioLab, Department of Electronics, University of York, Heslington, UK

<sup>1</sup>e-mail: [kaey500@york.ac.uk](mailto:kaey500@york.ac.uk)

<sup>2</sup>e-mail: [tony.tew@york.ac.uk](mailto:tony.tew@york.ac.uk)

<sup>3</sup>e-mail: [gavin. Kearney@york.ac.uk](mailto:gavin. Kearney@york.ac.uk)

September 23<sup>rd</sup> 2016.

## Abstract

Head and torso simulators are used extensively within acoustic research, often in place of human subjects in time-consuming or repetitive experiments. Particularly common is the Knowles Electronics Manikin for Acoustic Research (KEMAR), which has the acoustic auditory properties of an average human head. As an alternative to physical acoustic measurements, the boundary element method (BEM) is widely used to calculate the propagation of sound using computational models of a scenario. Combining this technique with a compatible 3D surface mesh of KEMAR would allow for detailed binaural analysis of speaker distributions and decoder design - without the disadvantages associated with making physical measurements.

This paper details the development and validation of a BEM-compatible mesh model of KEMAR, based on the original computer-aided design (CAD) file and valid up to 20 kHz. Use of the CAD file potentially allows a very close match to be achieved between the mesh and the physical manikin. The mesh is consistent with the original CAD description, both in terms of overall volume and of local topology, and the numerical requirements for BEM compatibility have been met. Computational limitations restrict usage of the mesh in its current state, so simulation accuracy cannot as yet be compared with acoustically measured HRTFs. Future work will address the production of meshes suitable for use in BEM with lower computational requirements, using the process validated in this work.

## 1 Introduction

The perception of spatial sound is known to be a complex phenomenon. In order to determine the location of a sound source, the human auditory system combines information from a number of cues; the differences between the signals at the ears (interaural level and time difference, ITD and ILD respectively) and the filtering applied by the outer ear, as well as any available visual cues. The ITD, ILD and the pinnae cues are all contained in a head-related transfer function (HRTF) pair. The contribution of interaural differences to localisation is well understood and easily implemented within spatial audio reproduction systems, however including accurate information from the whole HRTF, including pinnae cues, is more difficult. HRTFs are unique to each individual as the filtering is due to the shape of the listener's head and

ears, whilst also varying as a function of sound source direction. However, when HRTFs are used correctly and effectively the sound field they generate at the entrance to the listener's ears is ideally identical to that produced by a physical sound source.

Much of binaural research centres around the accurate capture, analysis and synthesis of HRTFs. Unfortunately, the experimental procedure for the capture of HRTFs is time-consuming, complex and repetitive, requiring subjects to remain as still as possible for long periods of time. If HRTFs are required within an audio reproduction system, one solution is to use generic HRTFs captured from a head and torso simulator (HATS) in place of the human subject. A particularly common HATS is the Knowles Electronics Manikin for Acoustic Research (KEMAR) seen in Fig. 1, which has acoustical properties derived from statistical research of the average human body, mean-

ing that KEMAR has the same acoustic properties as an average human [5].

Whilst physical acoustic measurement is perhaps the most accurate approach to capturing HRTF data, computational modelling techniques provide an appealing alternative by solving the wave equation subject to certain boundary conditions. The boundary element method (BEM) is the most common of these numerical techniques used in the calculation of HRTFs. BEM uses a surface mesh model of a scenario and defined sources and receivers to calculate the propagation of sound through the scenario. Therefore, a change in workflow to use a BEM compatible 3D surface mesh of a HATS such as KEMAR would allow for binaural analysis of speaker distributions and spatial decoder designs in a very similar way to current techniques, but without the disadvantages associated with physical measurements of human subjects. This could then be expanded to BEM calculations using mesh models of the human subjects if such meshes were available.



Figure 1: KEMAR model 45BC

This paper describes the development and validation of a mesh model of KEMAR based on the original KEMAR computer-aided-design (CAD) file. To facilitate the production of a variety of mesh models suitable for BEM calculation in the future, a workflow is needed to ensure the resulting mesh is consistent with the original whilst also meeting the requirements for BEM calculation. This process has been validated using a number of analysis techniques, and opens the door for the creation of a family of meshes optimised for different computational situations.

## 1.1 Boundary Element Method

To avoid measuring HRTFs, analytical and numerical methods can be used. The simplest solution for HRTF calculation is the analytical solution for scattering on a sphere, where the head is approximated as a rigid sphere without the pinnae and torso [2]. Without the pinnae the produced results are only valid for low frequencies, as the pinnae begin to have an influence above approximately 5 kHz [7]. This solution can be extended to include the torso in what is known as the ‘snowman model’ [1] but this is still missing the details in the pinnae.

To account for this complex geometry, researchers developed various numerical methods, the most common of which is the boundary element method, or BEM. In BEM the boundary problem of the wave equation is converted into a surface integral, which is then discretised into a number of elements. This set of simultaneous equations can then be solved to find the pressure at a point on the surface.

There are two BEM techniques used in acoustic computation: direct and indirect BEM (DBEM and IBEM respectively). DBEM is based on the Helmholtz formula, which relates the pressure in the fluid domain (in this case air) to the pressure and its normal on the boundary, and creates a non-symmetric matrix of equations. IBEM assumes that the pressure field is caused by a monopole distribution on the boundary surface, and creates a symmetric set of equations. These equations contain a component for every element in the mesh, and there is an equation for each element. Hence, with dense meshes this equation matrix can get very large. Determining which method to use depends on the problem size. For problems over a few hundred elements the direct method is better, as it is optimised for speed. However, as it solves a full set of simultaneous equations the storage required is large. The indirect method is slower, but requires less storage [12].

In order to be used within a BEM calculation, the surface must both be closed and discretised into a mesh. This discretisation converts a smooth surface into a number of smaller planar elements, the size of which determines the maximum valid frequency of the resulting calculation. It is generally acknowledged that a limit of 6 elements per wavelength is acceptable, although this can be pushed to 4 per wavelength. The maximum frequency is then determined by:

$$f_{max} = \frac{c}{edge_{max}} \quad (1)$$

where  $c$  is the speed of sound in  $\text{ms}^{-1}$  and  $edge_{max}$  is the length of the longest edge in m. This means

that for a mesh to be valid at 10 kHz, the maximum edge length anywhere in the mesh must be less than 5.67 mm. For 20 kHz validity, this maximum is 2.83 mm.

BEM has been historically restricted to fairly low frequency calculations because of the storage requirements, but advances in computational resources are continually raising the upper frequency limit.

## 2 Historic Work

Weinrich was the first to attempt modelling the sound field around the head in 1984 using a number of numerical techniques [17]. The ear canal was modelled as a series of cylinders of varying radius, and the sound field calculated using transmission line theory. The sound field of a simple two-dimensional geometric model of the pinna was calculated using a finite difference time domain approach, and showed roughly the dependence of the first HRTF notch on sound source elevation. A coarse surface mesh for the head was also created, minus the pinnae, with the response calculated using BEM. The maximum validity of the mesh was only 1.7 kHz, but the results in this range compared favourably with measurements made on a physical replica of the BEM model.

In the early 2000s Katz used BEM to calculate individualised HRTFs, focussing on the contribution of head and pinnae shape to the HRTF [13]. Modelling the head in a BEM environment allows for modifications such as the removal of the pinnae - crucial to investigating the contribution made to the HRTF, but not something possible with a real human subject. The work was limited by available computational resources, with 5.4 kHz being the maximum valid frequency: even after simplifying the requirements the full calculation took 50 days of CPU time.

As the optical scanner used by Katz was restricted to line-of-sight only, behind the ears, the cavities of the ear, and the ear canal were all viewed as filled. This is not a problem in the case of the ear canal, as most HRTF measurements are done using a blocked ear canal. It has been shown that the ear canal does not provide any directionally dependent information [6], and ear canal-related resonance would only need to be included if the sound reproduction site were to be at the eardrum itself. The filled-in nature of the ear cavities will probably have caused errors in the final solution results, but these would likely have been above the valid frequency of the calculation. Modifications of the mesh outputted by the optical scanner were also required. The top and bottom of the mesh were closed, as the scanner could not

handle perpendicular surfaces or those which extend beyond the machine, and holes in the mesh were filled in. The mesh was then coarsened to make the computation more manageable and refined in regions where large elements existed. Katz could not locate any existing software to do this coarsening and refining, so a brute force approach was taken.

Jin et al. used Fast Multipole BEM (FM-BEM) to calculate the HRTFs of a large database of subjects scanned using magnetic resonance imaging (MRI), culminating in the Sydney York Morphological and Acoustic Recordings of Ears (SYMARE) database [8]. The database contains both the meshes suitable for use within BEM at a range of frequencies and the HRTFs calculated from them. Various software applications were used in the processing workflow to perform steps similar to those described by Katz [13]. The SYMARE database is available at a number of frequency resolutions: 6 kHz, 8 kHz, 10 kHz, 12 kHz, and 16 kHz, whilst the head-only mesh is available at 15 kHz, 16 kHz and 20 kHz. An average mesh has approximately 130,000 elements. The Amira software [3] was used to extract the surface meshes from the scan data. Geomagic Studio (now discontinued, replaced by Geomagic Wrap [4]) was then used to clean the mesh and fill in holes (ear canal, nose etc). Geomagic Studio and the open source software MeshLab [14] were used to complete the various alignments and merges required across mesh resolutions. The open source software ACVD [16] was then used to improve uniformity of the surface elements across the mesh, and Geomagic Studio used again to reduce the number of elements by applying a small amount of smoothing.

In a large body of work culminating in [10], Kahana and Nelson used a laser-scanned model of KEMAR in BEM calculations to look at contributions made by the pinnae to the HRTF (known as pinna-related transfer functions: PRTFs) by using both KEMAR and a baffled pinnae, in similar work to Katz. Their results were also limited in frequency due to the available computational resources, with the calculations valid below 10 kHz.

The scanned surface was decimated using an algorithm by Johnson and Hebert [9] to create a more homogenous distribution of nodes and elements, creating a mesh with 23,000 elements valid up to 10 kHz (using the rule-of-thumb of 6 elements per wavelength). Their work showed the feasibility of BEM usage up to higher frequencies, but full mesh calculation was still limited by computation requirements. The baffled pinnae mesh was valid up to 20 kHz due to the physically smaller size of the mesh.

Previous work using KEMAR within BEM has

relied on a scanned version of some form. This has been shown to have certain limitations. Often scanners are restricted to line-of-sight, meaning occluded areas cannot be captured. This leads to regions such as behind the ear and resonant cavities within the ear being seen as filled [13], an obvious difference when compared to an actual pinna. Important morphological details can also be lost if the scan is not of a high enough resolution [10], as well as requiring post processing to combine scans from multiple directions. In the present work, however, the original CAD file of KEMAR used to produce the physical dummy is used rather than a scan, meaning complete accuracy can in principle be obtained.

### 3 Methodology

Following the work of Jin et al. [8], a number of software applications were used to create a mesh suitable for BEM usage from the initial CAD file. The CAD information describes the entirety of KEMAR, including torso and base. Perfect consistency with the physical KEMAR was therefore theoretically possible, and the model should not suffer from the disadvantages associated with some scanning techniques. Once created by the designer, the 3D shape in the CAD file is described using a series of large faces, defined by cartesian coordinate points and curved splines known as boundaries. These large curved faces can be seen in Fig. 2. This type of definition is unsuitable for BEM calculation, so conversion was required to create a discretised planar polygonal mesh.

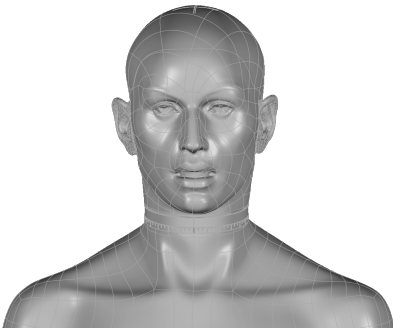
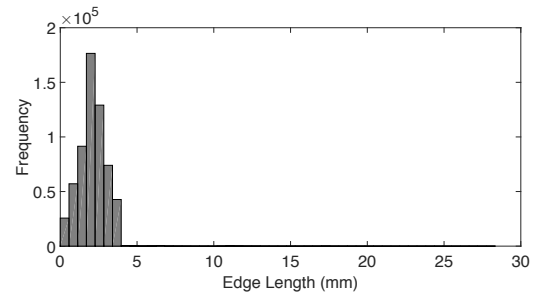


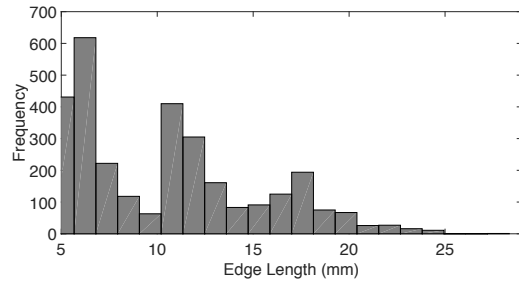
Figure 2: Head portion of the KEMAR CAD file

Geomagic Wrap [4] was used to convert the CAD file to a polygonal mesh, defined by information contained within the STEP CAD file format. The resulting mesh had 200,014 vertices and 400,001 faces, but with an irregular and undesirable distribution of shape and size. The maximum edge length in this mesh was 28.3 mm (as can be seen in Figures 3a

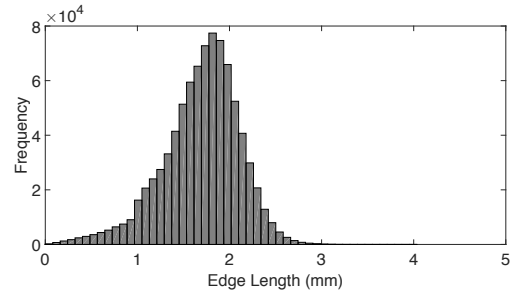
and 3b), most likely due to larger flatter areas of the torso defined using fewer mesh elements. As the maximum valid frequency is dependent on the largest edge length present in the mesh, a remeshing procedure was required to create a mesh that is valid up to 20 kHz by breaking up these large flat regions and reducing the edge lengths.



(a) Prior to remeshing: full histogram



(b) Prior to remeshing: tail of histogram



(c) After remeshing: full histogram

Figure 3: Distribution of edge lengths in mesh before and after remeshing

The ‘remesh’ feature of Geomagic Wrap was used to redefine the surface mesh with a target edge length of 2 mm. This process resulted in a mesh with a much improved distribution of edge lengths (as seen in Fig. 3c); whilst not all are under the target length of 2 mm, the largest is only 4.04 mm, giving a maximum valid frequency of 14 kHz when assuming 6 edges per wavelength. This is better than the original mesh, but still not ideal. The shape of some faces was also undesirable, in that they were too long and

thin. Equilateral triangles are much preferred over long thin triangles.

To create more nearly equilateral triangles across the mesh, the open source software ACVD [16] was used. This improved the distribution of edge lengths, as seen in Fig. 4. The maximum edge length was reduced to 2.49 mm (valid to 22.7 kHz when assuming 6 edges per wavelength) but at the expense of many more vertices and faces.

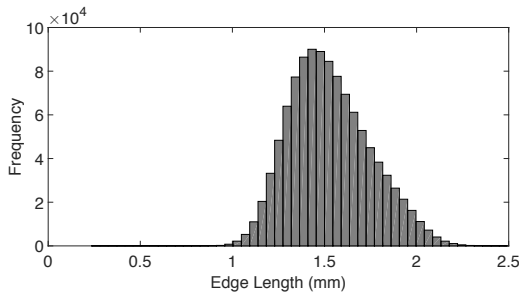


Figure 4: Distribution of edge lengths in final mesh

The remeshing process introduced a large number of unreferenced vertices: more specifically, the procedure did not remove the vertices associated with the previous arrangement of triangles, resulting in more vertices than were used. This did not affect the structure of the mesh, but would require a BEM solver to do more calculations than was strictly needed. The open source software MeshLab [14] was used between Geomagic Wrap and ACVD to remove these unreferenced vertices.

The final mesh had 360,017 vertices and 720,015 faces, a portion of which including the pinnae appears in Fig. 5. It can be seen that the mesh contains consistently small equilateral triangles.

The volume of the mesh at each stage of processing was also calculated to ensure no large discrepancies were introduced, as a change in volume from the original mesh invalidates any BEM calculations. These volume calculations, and their difference from the original CAD file, can be seen in Table 1. There is a very small loss of volume at each stage that can be attributed to the slight rounding of sharp corners during the remeshing and redistribution processes, however no particular stage introduced a large variation in volume, suggesting that all software applications and processes used were valid. This is investigated further in the next section.

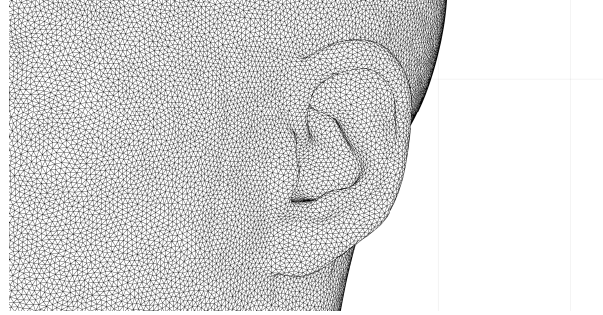


Figure 5: Pinna portion of the final mesh

## 4 Validation

A number of validation steps were required during the process of mesh production to inform decisions. As detailed in the previous section, maximum edge length and volume were the primary criteria for validity at each stage of the process.

In addition to the maximum edge length criterion, BEM solvers often have a minimum and maximum internal angle requirement to help avoid the inclusion of long thin triangles. In this and related future work, the PACSYS PAFEC-FE software [15] is the intended BEM solver, which enforces angles to be between 15° and 150°. All angles in the final mesh were between these limits (as seen in Fig. 6), with the minimum at 16.0° and maximum at 146.7°.

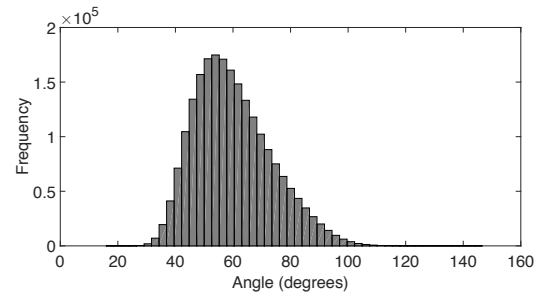


Figure 6: Distribution of angles in the final mesh

The local topology of the mesh was also of importance to ensure no regions were distorted by the process, so the distances between the faces in the original CAD data and the processed mesh were calculated. These values show that the local topology is consistent with the original, with all values less than 0.64 mm and the majority less than 0.1 mm; some faces had zero distance between them. This distribution can be seen in Fig. 7. The larger distances lie in regions such as the edge of the base and within the eyes, where the radius of curvature can be relatively small and the shape has been rounded during

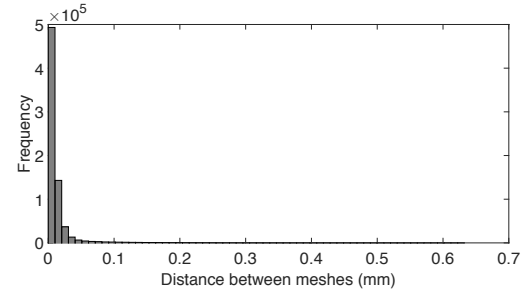


Mesh	Volume (mm <sup>3</sup> )	Difference from original
Original CAD file	28492978	0%
Initial mesh	28489372	-0.0127%
Remeshed at target edge length	28488486	-0.0158%
Redistributed mesh	28486546	-0.0226%

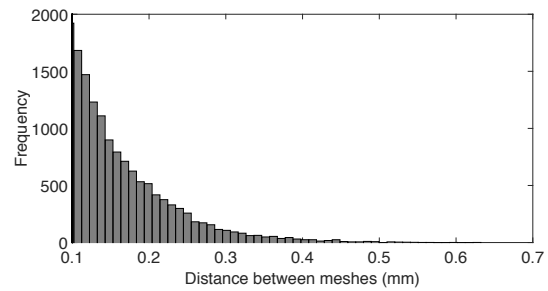
Table 1: Volume calculations for each stage of processing - mm<sup>3</sup> is required for the differences to be visible.

the remeshing or redistribution process. These regions are not of critical importance to the BEM calculation however; the values in the region around the pinnae are of higher importance. Whilst not zero, the values here are small enough to be considered valid, although further testing is required to confirm this for different applications. Fig. 8 shows the distances between the faces as a function of colour, with black at zero and white at the maximum value of 0.63mm. Whilst the pinna lights up a small amount, there is not as much white as in other regions of the mesh.

Unfortunately, whilst the mesh is numerically valid for BEM and consistent with the original CAD file, the size of the mesh renders it unusable by current solvers. A mesh consisting of 720,015 faces would require 7725GB of RAM to store the necessary equations to then solve using the PAFEC-FE software. This means that HRTFs of KEMAR cannot be calculated using this version of the mesh: a physically smaller mesh or one with a lower frequency limit would be required. Additionally, even if the mesh could be used for BEM calculation of HRTFs, this mesh includes the ear canals, and there are no databases of acoustically measured KEMAR HRTFs which include the ear canal to compare against.



(a) Full histogram



(b) Tail of histogram

Figure 7: Distribution of distances between the two meshes



Figure 8: Distances in mm between the two meshes, plotted on the final mesh as a colourmap

## 5 Conclusion

The original KEMAR CAD file was converted using a number of processing stages to a mesh which meets the topological constraints for BEM compatibility. The original 3D surface data has been discretised, remeshed and redistributed to meet the edge length requirement for a 20 kHz maximum valid frequency and the angle limitations imposed by the BEM solver. The volume and topology of the meshes prior to and after processing have been compared to ensure consistency between the original and the final mesh, and whilst there are small differences these are likely to be within the margin of error introduced when comparing a calculated result with a physically measured result. The workflow created in this paper is applicable more generally to the processing of meshes edited from an original, with consistency and validity assumed.

### 5.1 Further Work

Due to computational limitations the final mesh is not usable in BEM calculations in its current state. The primary aim of this work was to determine the workflow required to accurately convert the CAD information into a BEM-friendly mesh; further work is needed to address the problem of computation and usage. This could include a reduction in maximum valid frequency, by permitting longer edges and therefore larger faces, or by using different meshes for dif-

ferent frequency ranges. Physical mesh size could also be adjusted: a mesh consisting of only head and shoulders would be sufficient for the majority of HRTF calculations, whereas the entire torso of KEMAR is present here. A mesh without ear canals would also be of more use than the current mesh, allowing comparison between typical acoustically measured HRTFs and those calculated using BEM.

## 6 Acknowledgements

This work is supported by Meridian Audio Ltd and MQA Ltd. Thanks also to Mr Laurence Hobden, Mr Patrick Macey and Mr John King from PACSYS, and to G.R.A.S for supplying the CAD file of KEMAR.

## References

- [1] V Ralph Algazi, Richard O Duda, Ramani Duraiswami, Nail A Gumerov, and Zhihui Tang. Approximating the head-related transfer function using simple geometric models of the head and torso. *The Journal of the Acoustical Society of America*, 112(5):2053–2064, 2002.
- [2] Richard O. Duda and William L Martens. Range dependence of the response of a spherical head model. *The Journal of the Acoustical Society of America*, 104(5):3048, nov 1998.

- [3] FEI. Amira 3D Software for Life Sciences, 2016.
- [4] Geomagic. Geomagic Wrap 3D Imaging software, 2016.
- [5] G.R.A.S. Head & Torso simulators, 2016.
- [6] D Hammershøi and H Møller. Sound transmission to and within the human ear canal. *The Journal of the Acoustical Society of America*, 100(1):408–27, 1996.
- [7] David Howard and Jamie Angus. Pinnae and head movement effects. In *Acoustics and Psychacoustics*, chapter 2.6.3, page 103. Focal Press, 2 edition, 2001.
- [8] Craig Jin, Pierre Guillon, Nicolas Epain, Reza Zolfaghari, André Van Schaik, Anthony I Tew, Carl T Hetherington, and Jonathan B Thorpe. Creating the Sydney York Morphological and Acoustic Recordings of Ears Database. *IEEE Transactions on Multimedia*, 16(1):37–46, 2014.
- [9] Andrew E Johnson and Martial Hebert. Control of polygonal mesh resolution for 3-d computer vision. *Graphical Models and Image Processing*, 60(4):261–285, 1998.
- [10] Yuvi Kahana and Philip A Nelson. Boundary element simulations of the transfer function of human heads and baffled pinnae using accurate geometric models. *Journal of Sound and Vibration*, 300:552–579, 2007.
- [11] Brian F. G. Katz. Acoustic absorption measurement of human hair and skin within the audible frequency range. *The Journal of the Acoustical Society of America*, 108(5):2238, 2000.
- [12] Brian F. G. Katz. Boundary element method calculation of individual head-related transfer function. I. Rigid model calculation. *The Journal of the Acoustical Society of America*, 110(5):2440, 2001.
- [13] Brian F. G. Katz. Boundary element method calculation of individual head-related transfer function. II. Impedance effects and comparisons to real measurements. *The Journal of the Acoustical Society of America*, 110(5 Pt 1):2449–55, nov 2001.
- [14] MeshLab. MeshLab, 2016.
- [15] PACSYS Ltd. PAFEC-FE, 2016.
- [16] S. Valette, J.-M. Chassery, and R. Prost. Generic Remeshing of 3D Triangular Meshes with Metric-Dependent Discrete Voronoi Diagrams. *IEEE Transactions on Visualization and Computer Graphics*, 14(2):369–381, 2008.
- [17] Søren Gert Weinrich. *Sound field calculations around the human head*. Acoustics Laboratory, Technical University of Denmark, 1984.