A SIMPLIFIED, AUTOMATED SYSTEM FOR MEASURING ROOM ACOUSTIC RESPONSES.

Robert Walker, Consultant

1. INTRODUCTION.

Over the past few years one of the continuing topics at RS conferences has been the (mis)use of steady-state room responses and target response curves, particularly in the cinema industry. It is clear that the use of steady-state room responses leads to inconsistencies, disagreements and other anomalies. One of the reasons is that the steady-state room response has only a weak correlation with the perceived sound quality. On the other hand, the cinema industry in particular does not employ acoustic specialists to set up each venue individually and might anyway have difficulties with the costs and time do so. What is needed is a simplified measurement system useable by non-specialist staff to give a reasonable measure of the subjective sound quality.

For a number of years now papers have been presented on the 'limitations' of the methods used by the cinema industry to 'calibrate' their acoustic spaces [1, 2, 3]. In particular, the variable results obtained from steady-state amplitude response measurements and the implicit standardisation of the 'X-curve' have been highlighted [3]. Apparently[1], in order to satisfy certain audio 'authorities' and acquire licenses to use those authorities' names and logos, cinema operators have been routinely using narrow band equalisation (most often 1/3rd octave) to 'adjust' the responses to satisfy the rather tight limits specified. Such equalisation is not a remedy for 'imperfect' acoustic performance.

Substantial variability in the subjective sound quality results from this process. Firstly, the rooms differ greatly both in size and in acoustic treatment. The same measurement process might be applied to cinemas and to the (usually) rather better controlled production rooms such as mixing rooms, Foley suites and review theatres. Disparities between the sound characteristics in the production facilities and the eventual public presentation spaces are inevitable.

More recently, it has apparently been asserted that the X-curve is not a 'standard' to be met but merely provided "for guidance". This author at least, finds it hard to comprehend how a measurement, required for certification and which has rather tight tolerance limits, is not actually a 'standard', but that is a matter for the industry. Resolving that issue is not the purpose of this paper.

However, it is certainly true that the origins of the X-curve appear to be reasonable, at least for cinema spaces if not for the production spaces. Ref. 4 presents an explanation of the origins and application. In Ref. 5, Mark Yonge writes "*The X-curve was originally intended to predict [what steady-state measurement would give for] a flat first arrival sound in a normally-reverberant, normal-sized cinema. ... but only in a normally reverberant room, where the room is within a specific size range, where the microphone is relatively far from the loudspeaker [i.e. in the reverberant field] and where the loudspeaker is fed with continuous pink noise and not with any kind of intermittent signal. The X-curve is not scoped, or useful, for measuring small rooms. This excludes most recording studio control rooms, or rooms with unusual acoustic treatment, very large*

[1] This topic is not this author's speciality. It is included sole for background. The main purpose of the paper is to present a potential simplified method for making acoustic response measurements that includes the subjectively important and excludes the subjectively less important.

auditoria, stadia or open-air spaces". According to Yonge [6] "*These [limitations] were once well understood by the technicians implementing [the standard]*".

The ideal would be a method that could be applied in all types of room to give an objective measurement that was well correlated with the subjective quality of the sound in the room. This, of course, is expecting a great deal – probably the impossible, and something that the audio community has been seeking for decades. A system that rejected the acoustic effects of the room in a similar way to the human hearing system might be a step forwards. As the steady-state response is clearly deficient, such a system must perforce be based on some sort of time-domain selectivity.

Ref. 4 does mention the possible use of time-domain windows to selectively weight the direct sound components. There, the argument against is presented as "*First arrival measurement methods tend to be based on proprietary equipment, so caution must be taken if pursuing a first arrival standard ... selecting one or a few of the large number of competing measurement systems might ... disfavor a manufacturer of valid measurement tools*". It is certainly true that very many possibilities do exist, both in proprietary equipment and measurement parameters.

The purpose of the development described in this paper was to provide a simple, public-domain application for commonly-available hardware that might provide a route for measurement standardisation – to give more consistent, and meaningful, results in rooms of different types and which was, at the same time, simple to operate and free from commercial considerations.

2. BACKGROUND.

2.1. AES / SMPTE Liaison Project.

Recently, the Audio Engineering Society has set up a new Working Group (SC-04-08) on Measurement and Equalization of Sound Systems in Rooms. This is divided into four projects – AES-X215, *Liaison with SMPTE Project to Codify Current Procedures to Calibrate the Cinema B-Chain*, AES-X216, *Liaison with SMPTE Project on Calibration Pink-Noise Standard and Test File*, AES-X218 *Measurement and Calibration of Sound Systems in Rooms* and AES-X219 *Method of Measurement for Frequency response and Impulse response of Sound Systems in Auditoria*.

The work described in this paper was inspired by earlier discussions with P. Newell at previous Reproduced Sound Conferences and the author's membership of the AES Study Group SC-04-08.

However, this development has not been done as part of that group's activities or under its sponsorship or control. In fact, many of the contributors to that group have yet to acknowledge the limitations of steady-state measurements for this type of room assessment. Much of the group's discussion to date has actually been about the merits (or otherwise) of various steady state test signals. It is however true that SC-04-08 does have a wider remit than just cinema or film sound.

It is the author's hope that such a system as described here might be useful in furthering the development of practically useful standards.

2.2. Simplified Measurement of Acoustic Room Responses (SMARRS).

This paper outlines a simple device that could be substituted in an existing measurement system to give non-expert users a simple readout of the room direct response without any requirement to understand the technicalities of the acoustics. The assumption is made that cinema operators and others already have access to an adequately calibrated and standardised microphone chain – as they must in order to be making the currently-specified acoustic measurements, under SMPTE 202. It is intended that the new analysis system could be used to provide the excitation signal and process and display the results.

The experimental measurement system obtains the overall room Impulse Response (IR), identifies the total system and propagation time delay and then selects the early part of the received room response by applying a pre-determined time-domain window. The result is then converted to a frequency domain response in either narrow-band or 1/3rd octaves.

All of the measurement parameters, for example, the actual values of the time windows, are yet to be determined. The present system is intended only to provide an experimental basis for evaluation and development and as a tool for discussion and, possibly, the setting of the final parameters. To that end the current implementation can also display some intermediate parts of the calculation and some other response graphs (see Section 5.5).

One of the current limitations of the experimental system is that information about the absolute sound level is not retained. The existing steady-state measurement system would still have to be used where the absolute sound level or overall system sensitivity is required.

The system has been developed using a public domain (GNU) compiler system (g++) and using pubic domain Qt4 libraries. At present, it also uses a public domain FFT library (fftw3) because this author is unfamiliar with the capabilities of the Qt4 FFT module. It also uses a public domain audio I/O library (pulseaudio) because the Qt4 audio implementation for Linux is incomplete (this may be improved in Qt5). At the present time, it is assumed that the actual audio I/O would be a separate module, possibly re-written and re-compiled separately for each different platform. To make that simple, the audio I/O has only a single access function from the main application.

The application has been developed using Qt Creator 4 under Linux (Ubuntu 12.04), but ought to be easily transferable to Microsoft ®, Apple ®, Android ® and other modern operating systems, so long as they have audio I/O capabilities and Qt libraries available. In fact, the Qt4 elements are only used for presentation and display and not for any critical operational functions. A simple, command-line application could be made without them but, today, the intended user is much more familiar with 'apps' and GUI operation than apparently esoteric command-line operation.

3. SYSTEM IDENTIFICATION USING IMPULSE RESPONSES.

For a linear, time-invariant (LTI) system the IR theoretically contains all of the information about the system transfer function. In that sense, it is almost entirely free of considerations and arguments about "the test signal", although there remain certain basic details that must be satisfied (see Section 4.1). The IR is a convenient starting point for the derivation of more familiar responses, such as the response magnitude as a function of frequency (amplitude response) or the response decay as a function of time (e.g. reverberation time).

In principle, to obtain a system IR an infinite-amplitude, zero-duration signal (delta function) could be applied to the system under test and the resulting time-domain response recorded directly. In practice, that leads to severe problems with system dynamic range and measurement signal-noise ratios [7]. No real system can respond linearly to an infinite input, even if one could be created, introducing (at least) amplitude distortion. In addition, the total energy in the test signal is small (unless of infinite, or at least, very great amplitude). Improvements to the signal noise ratio can be obtained by averaging many responses [8], but the finite amplitude limitations will always remain.

4. IMPULSE RESPONSE DERIVATION USING CORRELATION.

4.1. Principles.

The principles of IR measurement with deterministic, pseudo-random (PR) signals are well known [9]. They involve the use of real and practical signals, i.e. signals that are constrained in amplitude, bandwidth and duration, to excite the system under test. The resulting complex system response can be processed arithmetically to derive a close approximation to the infinite IR. The problems of amplitude non-linearity are significantly reduced, though they do still need consideration. The signal-noise ratio problems are reduced by the extended duration of the excitation, in which the average power in the test signal is increased relative to a delta function and which can thus contain much more total energy than an almost infinitesimally short pulse.

In practice, the finite bandwidth, discretisation of both the time domain and the amplitude response in modern digital signal generation and recording methods introduce additional limitations. However, those limitations are always calculable and can, in most applications, be arranged to allow adequate performance for the task – but they must always be taken into account when interpreting results of measurements. For example, the finite record length sets the frequency resolution – the ultimate resolution cannot be greater than the reciprocal of the total record length. In the frequency domain, the audio sampling process limits the upper frequency range to one-half of the sampling frequency. Within these well-understood limitations, the IR contains all of the information about the system under test. Further processing of the IR can be carried out to extract other parameters or functions [e.g. 10].

An earlier paper by this author [11] presented a detailed description of using modern personal computers (desktop, laptop and these days 'pads' of various types) to carry out room acoustic measurements based on impulse responses. A little of that is repeated here for completeness.

4.2. Method Outline.

A test signal is applied to the system under test and the response recorded. The overall impulse response is calculated from the cross-correlation of the output test signal with the received system response. If the autocorrelation function of the input test signal is (or is nearly enough for the purposes) a delta function, then that cross-correlation gives the system IR directly.

Calculating long cross-correlation functions requires a great number of arithmetic operations[1]. Over the years, algorithms have been devised to reduce processing times for signal transformation, especially for carrying out the Fourier transform. Those algorithms work by breaking up a long transform into shorter segments. It is much faster to process many shorter transforms and re-assemble the results than to carry out the longer transform directly. Carl Friedrich Gauss (around 1805) already knew of such algorithms. Various limited forms were rediscovered several times throughout the 19th and early 20th centuries. In 1965, Cooley and Tukey [12] published the modern form of the Fast Fourier Transform (FFT) for electronic computers. In 1972, this author was using the related, but arithmetically simpler, Fast Hadamard Transform (FHT) in hardware to process real-time video signals (110 Mb/s data rate) for bit-rate reduction [13, 14]. Today, the FFT and its close relative the Fast Cosine Transform are ubiquitous, widely used in video and audio coding (MP3, AAC, DTT, digital video, mobile phones, etc.) and in many other fields.

These fast transform algorithms can be used to implement the correlation function more economically than direct calculation. Convolution of two time-domain signals can be achieved by multiplying their Fourier transforms in the frequency domain and then using the Inverse FFT (IFFT) to return the result back to the time domain. If the complex conjugate of one of signals is used in the multiplication then the final result is their cross-correlation function.

4.3. Test Signals.

To give adequate frequency resolution, the test signal needs to be relatively long in duration (the frequency resolution is *1/duration*) and (self-evidently) needs to contain all of the frequency components needed for the measurement. In addition, because of the inherent properties of the Discrete Fourier Transform, the test signal also needs to be cyclically repetitive.

[1] An acoustically useful test sequence length of 2^{16} (about 1.4 s at 48kHz) would require about 8.5 x 10^9 multiply/accumulate operations for the autocorrelation function. Even today that would be significant.

A large number of potential test signal types fulfil these requirements. Generally, only two are common. One is a maximal-length binary sequence (MLS) of length N samples, where N is of the form 2^n-1, n being integer. They are easily generated in hardware in real time or by a simple calculation. For system testing they have some valuable properties. The sequence values are limited to ±1 only, giving a theoretical peak/rms ratio of 1.0 (the lowest possible) and all of the frequency components they contain are the same amplitude but of pseudo-random phase. They have a normalised autocorrelation function value of unity for zero shift and $1/N$ for all other shifts. Thus, their autocorrelation function is close to the required rectangular impulse. Typically, values of N of about 4,095 to 65,535 would be used in practice, giving values for $1/N$ ranging from -72 dB to -96 dB, good enough for nearly all acoustic or electronic applications.

Ideally, the MLS test signal amplitude could be set to peak at the system overload level. In practice, the test signal has essentially infinite bandwidth and the inevitable system bandwidth limitation effects (even in the signal generator hardware) cause a degree of low-pass filtering. That creates a more noise-like multilevel signal, which in turn requires some amplitude headroom (in most cases at least 12dB). That is still very much better than impulse measurements.

A second signal type is a linear frequency sweep ('chirp' – widely used in radar systems). It is a sinusoidal signal of unit amplitude whose rate of change of phase is proportional to time elapsed from the signal start (with some initial offset). Typically, the signal will start at some lower frequency and progress to some higher frequency linearly over the duration of the whole test signal. The advantage of a sinusoidal waveform is that the peak/rms ratio is that of a sine wave (≈ 3 dB). The system amplitude limits can be well defined in relation to the test signal and the energy at any frequency maximised. Theoretically, the test signal amplitude could be set to peak at the system overload level. In practice, response modulation effects in systems under test (especially due to additive interference from time-delayed response components) require some amplitude headroom (in most cases at least 6dB).

To meet the requirements for the test signal to be cyclically repetitive, the test signal duration must be longer than the system response (approximately the reverberation time in a room). This is in addition to any considerations for frequency resolution. It must also be repeated at least once and the system output response recorded during the second or subsequent repeat of the sequence. That is to allow the assumption of cyclical repetitiveness to be met. If the test signal duration is longer than the system reverberation time then any previous responses will be below –60dB.

4.4. Processing.

Fig. 1 shows the operation schematically for an MLS test signal. It should be noted that the MLS sequence length of 2^n-1 does not fit exactly with the FFT processes, which are significantly better matched to lengths of 2^n. The missing sample is usually filled by zero padding. That has no significant effect in practice, but it is needed to keep the processing optimised. In references to sequence lengths in the remainder of this paper the padded length will be intended.

All of this processing was used by Borish and Angell in 1982 [15] and subsequently incorporated into MLSSA™, developed by Douglas Rife and John Vanderkooy in the late 1980's [9, 10]. It has been much used and replicated ever since. Because of processor limitations, early systems used the FHT rather than the FFT as it was simpler to implement and quicker in operation [9]. Many still do. However today, with modern processors and maths co-processors that take no more time to carry out multiplication than addition, the FHT has no practical speed advantage [11][1].

[1] The HT decomposes the signal into the discrete, dyadic Walsh-Hadamard domain, based on Walsh functions (cal() and sal()) – not the common continuous frequency domain, based on sin() and cos().

5. EXPERIMENTAL MEASUREMENT SYSTEM.

5.1. Measurement Parameters.

The experimental system was intended as a basis for further experiments and tests, perhaps as part of the AES/SMPTE project group developments. It was not intended as any sort of final or user measurement system.

An external initialisation file ('smarrs.ini') is used to set any measurement parameters required to be different from their default values (default values in brackets). These are the audio sampling rate (48kHz), test sequence length (2^16, 65536) and the masking window settings (2 ms, 50 ms and 5 ms), together with some other minor parameters. At present, none of the measurement parameters is adjustable while the application is running. Other parameters, less likely to require frequent changes, are included in header files and require re-compilation to make changes.

The measurement system is available as a set of source code files, which can be re-compiled for a particular target platform. Compilation currently requires a Qt4 development environment and *fftw3* and *pulseaudio* libraries and header files.

The distribution package also includes a Linux-compatible binary file for those wanting to try it immediately on a compatible system. The executable should run directly on a typical Linux system with default audio I/O settings. However, this has only been verified on a few systems and some audio settings may need to be adjusted[1]. Being a development system, it does also require some run-time libraries (at least Qt4, fftw3 and portaudio) to be installed as well. Some of these are included anyway on a default Linux system installation (e.g. Ubuntu 12)[2].

5.2. Main Control Window.

Fig. 2 shows the control window on application start-up. Only one main control is enabled initially – "*Measure*" to start the measurement. Two groups of selectors, "*Impulse response Save/Load*" and "*Test signal type*" and one rolling number control "*Repeats*" are also active. These are discussed in Section 5.6 and 5.7 below.

On starting the application, the class modules for measurement, audio I/O and graphics display are instantiated with the system parameters (e.g. sample rate and sequence length). The system then awaits the command to start.

5.3. Signal Generator and Audio I/O.

With modern desktop, laptop or tablet computers, high quality replay and recording of audio signals is an integral part of the multimedia system. Standardised procedure calls can set up, output and record audio signals in high enough quality for room acoustic measurements and most other audio measurements. In principle, the operations are essentially trivial.

The output test signal is generated in software and stored in an audio-compatible format array (16-bit signed linear pcm) and as an fftw-compatible complex double array. (No attempt has been made to optimise the transforms as real-complex or complex-real because the additional storage and calculation overheads of complex-complex are not significant in this application). The output signal could also have been stored as the complex conjugate of its transform, for later re-use in multiple or repetitive measurements, but the savings in doing this would be negligible and that would not help in carrying out multiple, independent measurements (see Section 5.6).

[1] The application audio processing currently selects the default output and input audio devices (Device #0).

[2] Using a Kubuntu 12.x Live disk, only the portaudio library (*libportaudio.s.0*) was not present.

To meet the requirements for cyclical repetition, the pcm audio output sequence is duplicated in the second half of the double-length output array. In addition, because of an as yet unresolved bug in the portaudio system, a number of extra 'blocks' of silence have to be added at the end otherwise the output audio is truncated prematurely. ('blocks' are the subdivision of the audio I/O into smaller chunks for 'callback' buffer processing. They were set at 1024 samples, which was a compromise between processing and latency. For a sequence length of 2^{16} (65536), four additional blocks (4096 samples) were needed to guarantee no truncation.). Thus, for a test sequence length of 2^{16}, a total of 135,168 audio samples is stored.

On the command to carry out the measurement, the system generates the audio output data and the complex conjugate of its transform. It then calls the audio I/O module with the location of the audio output data, the address for the returned audio input data and the audio data length. This is the only interface between the main system and the audio I/O subsystem.

On return from the audio I/O function the system carries out a sequence of processing steps. First, the received audio sequence is converted to complex format and any dc component removed (some analogue audio input systems can have significant dc offsets.). Then, an FFT is carried out to transform it to the frequency domain and the two frequency-domain responses corresponding to the output and complex conjugate of the input audio signals are multiplied together. The multiplication routine includes masking by a frequency-domain window to set the bandwidth to about 16kHz and reject all higher frequencies. That reduces out-of-band noise components that would alias down into the measurement band. The masking is done with a raised cosine taper from unity at 16.8 kHz (adjustable in the 'ini' file) to zero at half-sample frequency (24 kHz). The product is then transformed back to the time domain using an IFFT to produce the 'raw' impulse response, which forms the basis of all further processing.

5.4. Peak Identification, Time-domain Masking and Low-frequency Limits.

The most critical component of this simplified measurement system is automatic location of the peak in the IR and setting a standardised masking window to select just the direct sound and a short period after it. That is the whole principle of and justification for the system.

It is assumed that the IR will be well enough defined to contain a distinctive peak at a point corresponding to the total system propagation time. That total time includes the computer processing delays, the audio system latency, the room's audio system processing time and the sound propagation from loudspeaker to measurement microphone. Of these, the computer audio system latency (maybe 40 – 50 ms) and the acoustic propagation delay ($\approx$ 3 ms per metre) will usually be the dominant components. If the IR response peak is not well defined then there are other acoustic problems not resolvable by such a measurement system[1].

The system is capable of handling total propagation delays approaching the length of the test sequence (and even negative delays) because all functions and data access pointers are implemented modulo the sequence length. This works because all of the processed data arrays are cyclically repetitive. However, delays approaching the sequence length will begin to breach the excitation repetition criterion. In most rooms, the current default length of about 1.4 s will be adequate. In any case, with very large rooms and/or long delays the LTI criterion will likely be breached anyway by acoustic variations (wind, draughts, temperature gradients, etc.).

The peak identification is done by simply scanning the data for the highest value. This has to be done separately for positive and negative values because many audio systems do not maintain absolute phase (including computer I/O systems). Some trials were made with smoothing the data before searching but these never produced any improvement. In any case, a small error of a few samples would not materially affect the subsequent processing.

[1] One reason might be that the room audio system includes non-time-invariant processing, such as dynamic source steering, noise reduction/masking, etc. These could contravene the LTI assumption and should be disabled for room acoustic measurements.

Having located the peak position, the IR is masked in the time domain to select the direct sound and a short period afterwards. To reduce the effects of abrupt window edges ('aliasing' and 'leakage'), the masking transitions are in the form of raised-cosine windows. The 'fade-in' segment is set at 2 ms. The 'fade-out' segment is set at 5 ms. The unity gain centre section is set at 50 ms. Fig. 3 illustrates the form of the complete window, where the cosine tapers have been exaggerated (10 ms and 40 ms) for illustration.

All of these window settings are subject to further refinement. In particular, the duration of the central unity-gain segment is a compromise between rejecting most of the room response and maintaining sufficient record length to give useful low frequency information. In any sort of measurement, the length of time for which the data is observed sets the lower limit of the frequency information that can be derived. There is an inevitable time-frequency product that cannot exceed unity and no information can be derived about frequencies lower than 1/*duration*. Theoretically, a 50 ms window would allow information down to 20 Hz to be obtained. In practice the limit is not so sharply defined and the lower frequency limit for a 50 ms window would be at least 50 Hz. This present system has a lower frequency limit set at the lower end of the 63 Hz 1/3rd octave band (see Section 5.5.2).

5.5. Post-processing.

After the 'raw' IR response has been obtained the remaining controls in the main window are enabled. These select the type of data display. The IR can be processed in the time domain to provide time-histories of the system response.

The ultimate purpose of the system is to present a very simplified set of results to the operator. That could be in the form of a simple text table. However, the purpose of this experimental system is to allow the measurement parameters and their effects to be observed. To that end, a number of additional data presentations have been included.

In the following display examples, most are measurements of the electronic system response, i.e. the electrical audio output connected directly to the electrical audio input (E-E). This illustrates the limiting system response, without any acoustic or transducer complications.

5.5.1. Impulse Response Displays.

In the experimental system, the 'raw' impulse response is displayed by default. It shows the entirety of the derived IR, for the whole length of the test sequence. An example is shown in Fig 4. This can be a very long sequence of data points (typically 65536) and it is unreasonable to expect to obtain much more information than the approximate location of the peak value and the general level of the rest of the response. It is, however, a useful indication that there is a well-defined peak and that it is located in a reasonable place within the overall response. If this is not the case then something has gone wrong with the measurement – possibly insufficient sensitivity of the loudspeaker/microphone chain or audio input selection.

After the window masking function has been applied to the raw IR, other time-domain responses can be displayed[1]. The second time-domain plot, "*Windowed IR*", shows the masked IR normalised in time so that beginning of the fade-in segment is at the origin. The length of the display is adjusted so that the end of the fade-out segment is at the end of the ordinate axis. An example is shown in Fig. 5. It also illustrates the modulation of the background by the fade-in and fade-out segments.

The next time-domain plot, "*Log Impulse Response*", shows the magnitude of the windowed IR on a logarithmic scale. An example is shown in Fig. 6. This is useful for closer examination of the later parts of the response and the room decay function (though the x-axis is not logarithmic and would

[1] This ability for the user to go between different versions of the IR at will requires that both are stored – in separate arrays (see Fig. 2).

not shown a uniform logarithmic decay as a straight line). The residual components, at around –60 dB relative to the peak value, show the residual random components resulting from the finite sequence length. The overall shape around the peak is a function of the time-domain masking windows. None of these would be significant for a real room response. As an illustration, Fig. 7 shows a comparable response measured in a studio control room.

5.5.2. Frequency Domain Displays.

Frequency domain responses can be calculated from the IR by applying an FFT to the masked IR data. Fig. 8 shows the system amplitude response. The somewhat irregular low-frequency fall-off below about 400 Hz is caused by the masking windows[1]. It is consistent and could be equalised out if required. That would be trivial but it can only be done after the final form of the masking window has been determined. The high frequency noise is again set by the test signal length limitation (the degree of departure from a pure delta function autocorrelation).

The final plot type is "*1/3rd octave FR*". An example is shown in Fig. 9. The narrowband frequency-domain data has been divided into blocks representing each of the ISO 1/3rd octave bands. The energy for each band was summed and the result normalised by dividing by the number of sample values ('bins') included in the band. That normalisation removes the 3 dB per octave increase in energy as the number of bins increases.

The source code includes a pre-defined parameter setting the minimum number of bins for a valid band. That sets the lowest frequency displayed. In the experimental system, that was set at 20. With a sequence length of 65536 and a sample rate of 48kHz the lowest band meeting that criterion is 63 Hz (ISO Band 18).

5.5.3. Text Display.

The control button "*Text results*" displays the same 1/3rd-octave data as above as a text list, as shown in Fig. 10. It is envisaged that this data would be the principle output of a final user system. Given the very large range of file format possibilities, it is difficult to imagine a useful electronic data interface that relatively unskilled operators could use to transfer the data to a report or form. The amount of data is very small and can easily be transcribed manually. An automated output process could be included to send the text results to a data file for pasting into a report but has not yet been implemented.

5.5.4. Measurements with Swept Sinewave.

All of the preceding results were generated using the MLS test signal. Measurements made using the swept sinewave were generally similar, though with much reduced spurious noise components. Fig 11 shows the results of the log impulse response display using swept sinewave. The noise components are clearly much lower than with the MLS signal, shown in Fig. 6.

Fig. 12 shows the narrowband frequency response. In comparison with Fig. 8, the high frequency results are much less affected by erratic noise components. The 1/3rd-octave results were essentially indistinguishable because of the averaging in the formation of the bands.

5.5.5. Measurements with Random Noise.

The experimental system includes the option for using pure random (white) noise as a test signal. This was included mainly as a demonstration of what happens if the test signal is not cyclically repetitive to match the discrete Fourier transform. The results are generally quite consistent with those from MLS or swept sinewave, though with much more uncertainty. Nevertheless, averaging random noise results over many tests (see Section 5.6) gave essentially the same average results.

[1] The 400Hz limitation is directly associated with the 2 ms fade-in window taper. Increasing the fade-in duration reduces the low-frequency attenuation, of course at the expense of time-domain resolution.

Fig. 13 shows a typical result for the log impulse response measured using random noise. It can be compared with the result for MLS (Fig. 6). The peak level is about 4 dB less (38 dB v. 42 dB) and the background level is about 12 dB higher (–8 dB v. –20 dB), giving a signal noise ratio 16 dB worse (46 dB v. 62 dB).

5.6. Multiple Measurements and Averaging.

In noisy environments, it could be advantageous to average a number of separate measurements in order to reduce the variability of the results. A "SpinBox" control, labelled "*Repeats*", can be used to set a number of measurements. The system will carry out that number of independent measurements. The audio test signal is created afresh for each measurement except for the swept sinewave, which is inherently determinate.

Averaging is restricted to the one-third octave band values only. In the primary purpose of the application (cinema/theatre response measurements) it is very likely that one-third octave bands will be specified as a "target" or "standard" response.

If other sorts of averaged responses were needed in the development of a standard or specification then these would better be obtained from a more generalised measurement system. The purpose of this present system is to provide a model or prototype for an eventual simplified measurement system. The purpose is not to usurp the existing excellent measurement systems currently available.

5.7. Loading and Saving Impulse Response Files.

Also as part of the SMARRS system development, it will very likely be useful to be able to save and re-load IR files. Two selection boxes are available to enable this (Fig. 2). These take control during the measurement process. If the "*Save*" selector is ticked then the measurement will take place as usual and then a standard "File save" window will then open to allow a file name and destination to be entered. If that is cancelled then the file save will not take place.

If the "*Load*" selector is ticked then SMARRS will not carry out a measurement but instead open a standard file selection window. A previously saved or externally generated IR file (see Section 5.9) will be loaded into memory instead of a measurement being made. If the load operation is cancelled then no measurement will be carried out and no file loaded.

(If both "*Load*" and "*Save*" are ticked then SMARRS will offer to save the measured IR and then re-load one - optionally a different one. That is not especially useful, but is a residue of as yet somewhat unrefined software.)

The file format is a simple text file, with two header lines and then one line per IR value (in a short floating-point format) and no index count. It is assumed that any application opening a saved file (e.g. gnuplot) will be able to cope with automatic re-numbering. On opening the file, the SMARRS application checks the first two lines to verify that the audio sampling frequency and sequence length contained in the file correspond with the current system settings. Those lines have an "#" at the beginning so that other programmes (e.g. gnuplot) can treat them as comments.

5.8. Processing Times.

The execution times were measured on a nine-year old ACER 4500 series laptop computer with an Intel Centrino M 1.8 GHz processor and 1GB RAM. This machine was only mid-range when purchased in 2004 and would now be considered venerable. Nevertheless, the processing times were entirely adequate for the purpose.

Measurements were made of the time taken for various processing operations[1]. A sample is shown in Table 1 below. This shows that all of the processing for a test sequence length of 65536 (1.365 s at 48kHz, resolution = 0.732 Hz) took 17.44 ms. The creation of the audio data took an additional 2.23 ms for pseudo random noise and 9.2 ms for swept sinewave. These timings were quite variable and could be 50% to 200% of these values (depending on other processes were occurring in that multi-window, multitasking machine). In any case, up to 50ms is a barely noticeable delay in the presentation of measurement results.

Operation	Value
Sequence length	65536
Sampling frequency, Hz	48,000
Frequency resolution, Hz	0.732422
Make audio data, μs	2234 (9189 for swept sine)
Forward transform, μs	9757
Average CPU load during audio I/O	0.082%
Audio data dc removal and conversion to FFTW Complex, μs	1552
Complex conjugate data, μs	1194
Complex multiply and filter, μs	3216
Find peak, μs	579
Data masking, μs	1140

Table 1, Timing and other measurements.

The table also shows the average cpu load during audio I/O (0.082%). That is useful for optimisation and diagnostics. It could be significantly greater on other systems (see Section 6). Even on that old processor, the audio I/O overhead was negligible.

Also of interest is the audio I/O delay (latency). Over a large number of trials that was within 2133 ± 2 ($\approx$ 42 ms), showing that the audio I/O synchronicity was very consistent.

5.9. Loading MLSSA® Files.

The file load facility can be used to load IR files made using other measurement systems. In general, these will not be compatible. However, a small command-line utility ("convert"), included in the software distribution, has been written to convert MLSSA files to the format required for SMARRS. (Typing "convert –h" will produce a brief help menu.).

The utility analyses the MLSSA file and displays its main parameters such as acquisition type, sample interval, record length, title and comments. The MLSSA IR data can then (optionally) be converted into a format compatible with SMARRS. In general, that requires interpolation because the input and output data samples are generally not on the same sampling grid. The algorithm used is a relatively simple one that introduces artefacts that can be serious if the sampling rates are too different. It can only be regarded as an approximation. It works reasonably well for MLSSA sampling rates that are a little higher than the 48kHz of SMARRS. It has not been tested and is unlikely to work at all well for MLSSA sampling rates that are very much lower.

Figs. 14 and 15 show a comparison between narrow-band frequency responses using this conversion and as processed by MLSSA itself[2]. Figs. 16 and 17 show the same comparison for 1/3rd octaves. In the latter case, MLSSA doesn't carry out the same calculation of 1/3rd octaves. Instead, it uses a running filter, which produces a continuous plot rather than a discontinuous one, but the same general features are visible in both plots.

[1] The system code includes macros to optionally time and print to standard output details of time taken by different processes. That can be sent to a file using the usual command-line re-directions.

[2] MLSSA uses a more complex graph drawing routine that interpolates smooth curves between widely-spaced data points. The real differences between Figs. 13 and 14 are not very great.

These two comparisons are included principally to verify the functionality of the experimental measurement system against an existing standard (and well accepted, though now elderly) one.

6. CONCLUSIONS AND FURTHER WORK.

The measurement system described in this paper was continued only to the point where the principle could be proved and the basic measurement performance assessed. The author's main objective was to provide a basis for development of the system, possibly in the pursuit of a new standardised measurement for cinema and other venues where consistency of subjective impression between venues is an important consideration.

Much development remains to be carried out. This author has so far only attempted run the application (successfully) on a few PC-based Linux systems (ix86/AMD) and on one alternative architecture operating system (Rasbian) on a Raspberry Pi single board computer. That ran the audio post-processing in about 170 ms. (about four times slower than the old laptop, but still entirely adequate for the purpose). That required re-compilation for the different type of processor (ARM).

7. REFERENCES.

1 Newell, P., Holland, K., Desborough, M., Fazenda, B., Neskov, B., Castro, S., Valdigem, E., Newell, J. "Room to room compatibility of the low frequency content of mixes using resonant and transient signals", Reproduced Sound 25 Conference, Brighton, 2009.
2 Newell, P., Holland, K., Torres-Guijarro, S., Sergio Castro, Valdigem, E. "Cinema sound: a new look at old concepts", Reproduced Sound 2010 Conference, Cardiff, 2010.
3 Newell, P., Leembruggen, G., Holland, K., TorresOGuijarro, S., Gilfillan, D., Newell, J, Santos, D., Castro, S., "Does 1/3rd octave equalisation improve the sound in a typical cinema?", Reproduced Sound 2011 Conference, Brighton, 2011.
4 Karagosian, M, "X-Curve Is Not An EQ Curve", SMPTE, 2013.
5 Younge, M. "Standards News", JAES, **61**, 6, p. 450.
6 ibid, p. 451.
7 Berman, J. M. and Fincham, L.R., "The Application of Digital Techniques to the Measurement of Loudspeakers", J. Audio Eng. Soc., **25**, 6/77, p. 370.
8 Fincham, L. R., "Refinements in the Impulse Testing of Loudspeakers", J. Audio Eng. Soc., **33**, 3/85, p. 133.
9 Rife, D. D. and Vanderkooy, J., "Transfer-Function Measurement with Maximum-Length Sequences", J. Audio Eng. Soc., **37**, 6/89, p. 419.
10 Rife D. "Modulation Transfer Function Measurement with Maximum-Length Sequences", J. Audio Eng. Soc. **40**, No. 10, October, 1992.
11 Walker, R. "Maximal Length Sequence measurements – a contemporary methodology for Windows® systems", Reproduced Sound 25 Conference, Oxford, 2006.
12 Cooley, J.W. and Tukey, J.W., "An algorithm for the machine calculation of complex Fourier series," Math. Comput. **19**, 297-301 (1965).
13 Walker, R. "Hadmard Transformation: a real time transformer for broadcast standard p.c.m television". BBC RD Report No. 1974/7 (1974).
14 Walker, R and Clarke, C.K.P. "Walsh-Hadamard transformation of television pictures". BBC RD Report No. 1974/13 (1974).
15 Borish, J. and Angell, J.B., "An Efficient Algorithm for Measuring the Impulse Response Using Pseudorandom Noise", J. Audio Eng. Soc., **31** 7/83, p. 478.

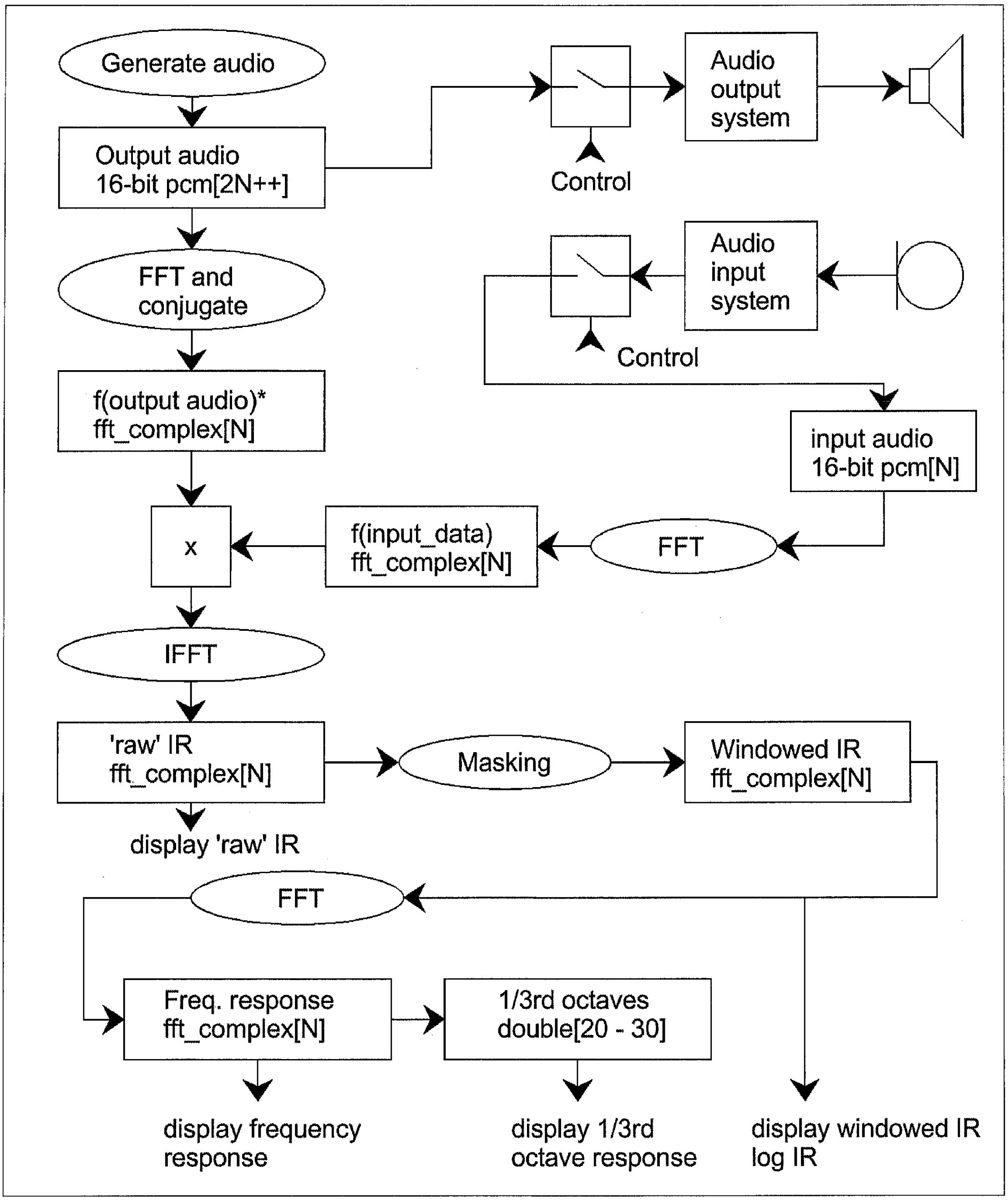


Fig. 1. Processing schematic.

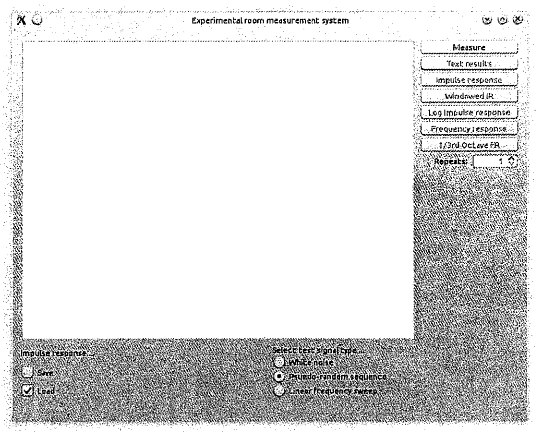


Fig. 2 Main application control window, on startup.

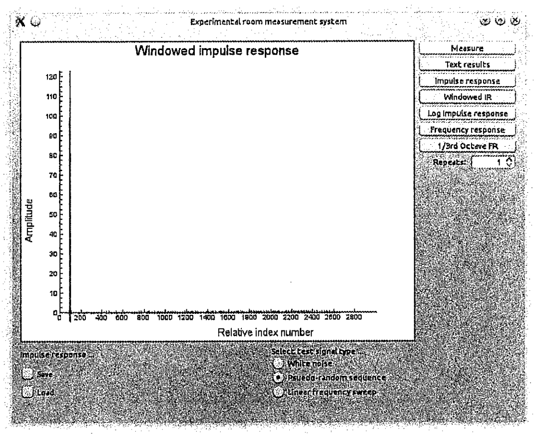


Fig. 5 Windowed impulse response, MLS test signal.

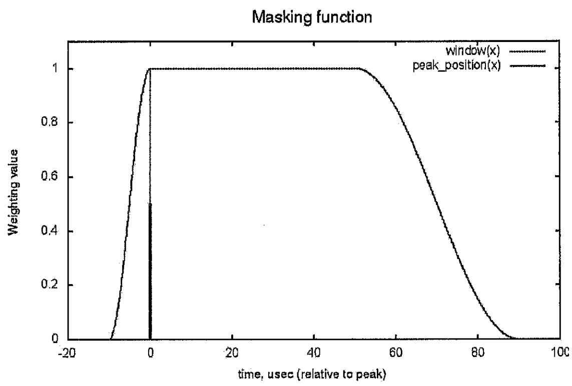


Fig. 3 Impulse Response masking window (schematic).

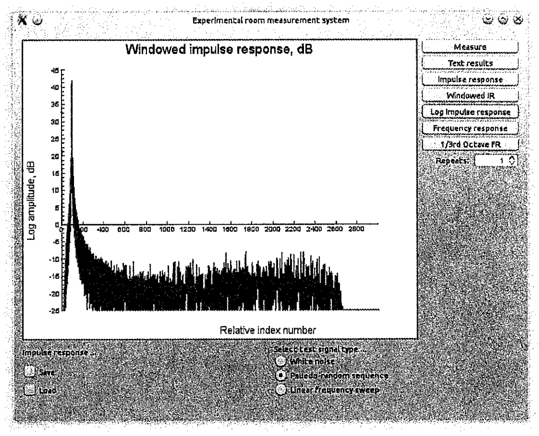


Fig. 6 Logarithmic impulse response, MLS test signal.

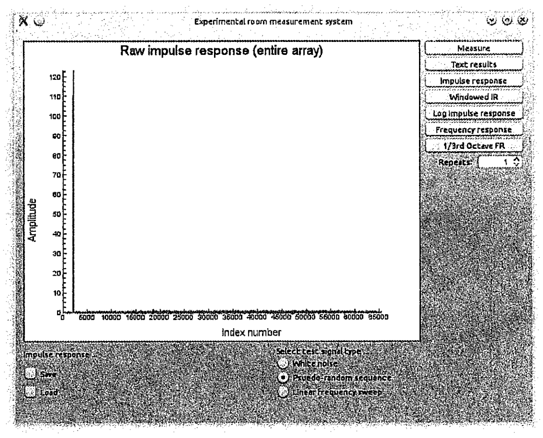


Fig. 4 'Raw' impulse response, MLS test signal.

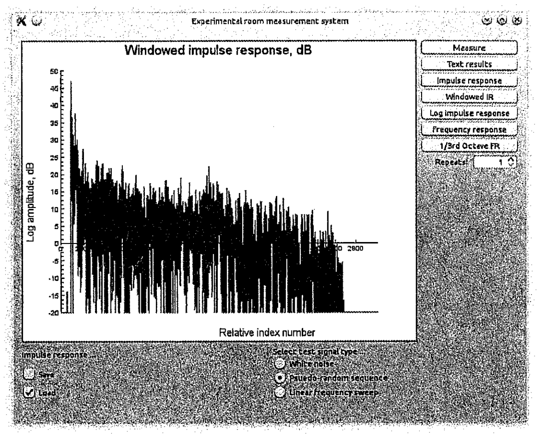


Fig. 7 Log. impulse response – studio control room, MLS test signal.

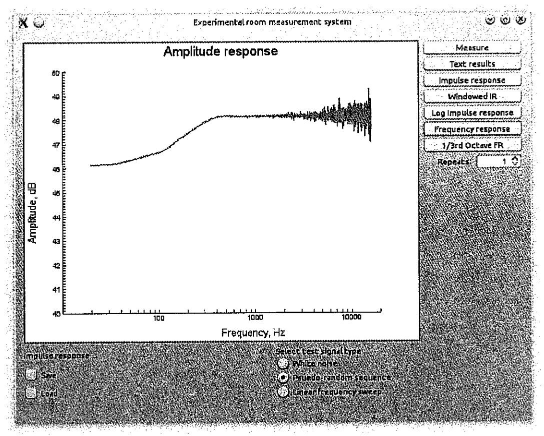


Fig. 8 Narrowband amplitude response, MLS test signal.

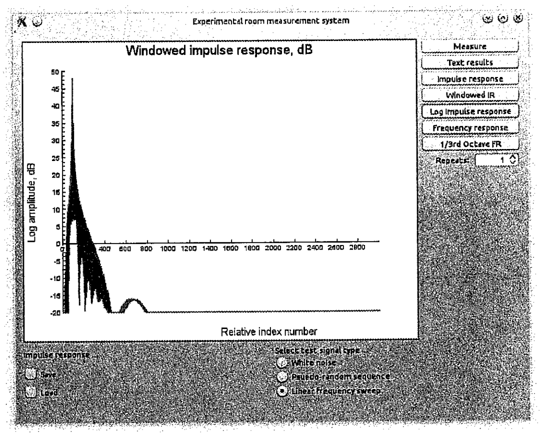


Fig. 11 Logarithmic impulse response, swept sine wave test signal.

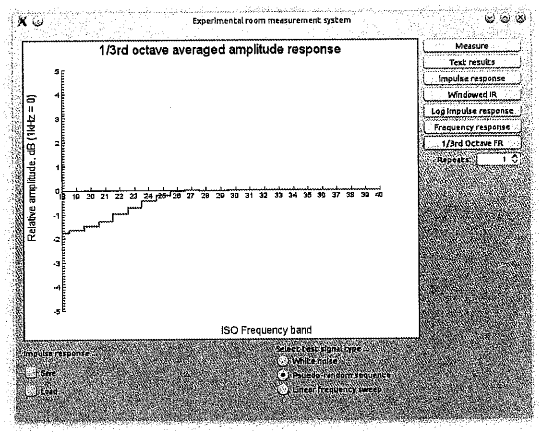


Fig. 9 1/3rd octave amplitude response, MLS test signal.

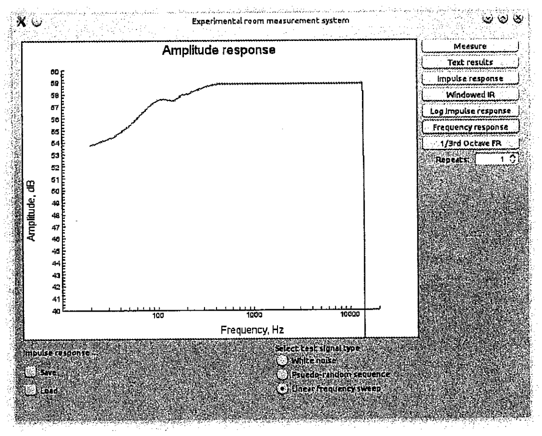
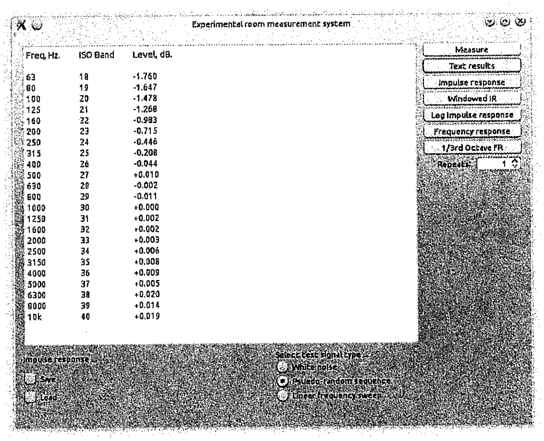


Fig. 12 Narrowband frequency response, swept sine wave test signal.



Freq, Hz.	ISO Band	Level, dB.
63	18	-1.760
80	19	-1.647
100	20	-1.478
125	21	-1.268
160	22	-0.983
200	23	-0.715
250	24	-0.446
315	25	-0.208
400	26	-0.044
500	27	+0.010
630	28	-0.002
800	29	-0.011
1000	30	+0.000
1250	31	+0.002
1600	32	+0.002
2000	33	+0.003
2500	34	+0.006
3150	35	+0.008
4000	36	+0.009
5000	37	+0.005
6300	38	+0.020
8000	39	+0.014
10k	40	+0.019

Fig. 10 Text results for 1/3rd octave bands, MLS test signal.

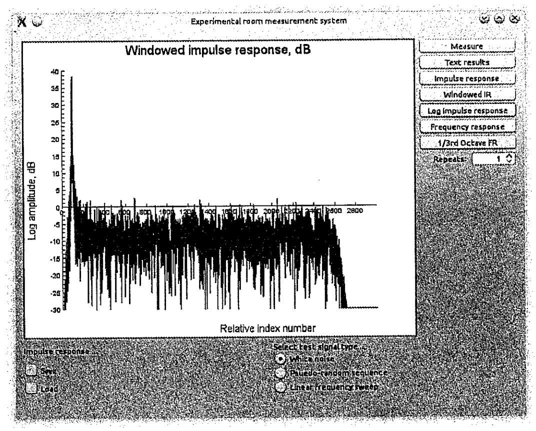


Fig. 13 Logarithmic impulse response, random white noise.

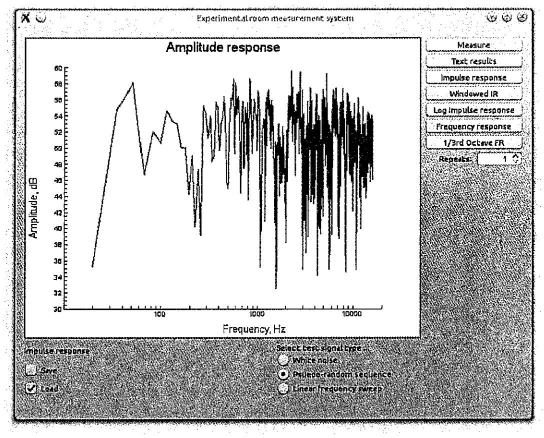


Fig. 14 Studio control room narrowband frequency response converted from original MLSSA file.

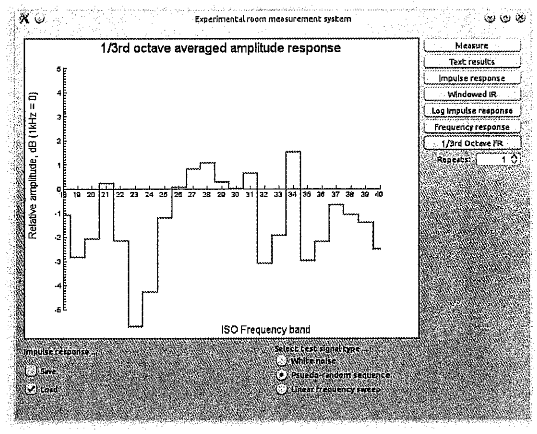


Fig. 16 Studio control room 1/3rd octave frequency response converted from original MLSSA file.

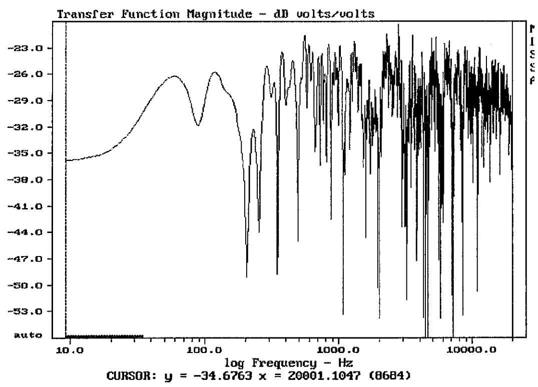


Fig. 15 Studio control room narrowband frequency response – MLSSA.

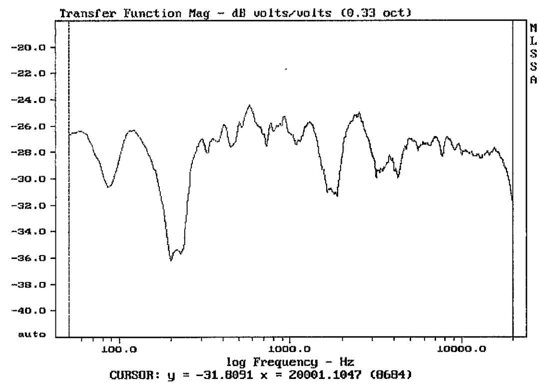


Fig. 17 Studio control room 1/3rd octave frequency response – MLSSA.

R. Walker
4th October, 2013.