Audio Engineering Society

Convention Paper 10247

Presented at the AES 159th Convention
2025 October 23–25, Long Beach, CA, USA

This paper was peer-reviewed as a complete manuscript for presentation at this convention. This paper is available in the AES E-Library (http://www.aes.org/e-lib), all rights reserved. Reproduction of this paper, or any portion thereof, is not permitted without direct permission from the Journal of the Audio Engineering Society.

Application of Low-Complexity Neural Equalizer for Adaptive Sound Equalization in Wireless Earbuds.

Andri Bezzola[1] and Yuan Li[1]

[1] *Samsung Research America, Audio Lab, Valencia CA 91355*

Correspondence should be addressed to Andri Bezzola (andri.b@samsung.com)

ABSTRACT

This paper presents an application of a neural equalization method of low complexity for the adaptive equalization of a wireless insert earbud. The task at hand is the calculation of optimal gains for a cascade of six biquadratic peaking filters with fixed center-frequencies and quality factors. Realistic equalization targets are obtained using measurements of the sound pressure response at the eardrum reference point with an earbud inserted into several synthetic ears and varying degrees of fit between the earbud and ear canal. We compare the accuracy of a neural equalizer that is based on a feed-forward neural network to a baseline method that relies on the self-similarity of biquadratic filters in the log-magnitude scale and calculates the optimal gain values using a least-squares approach, and to a neural equalizer trained on random polynomials to estimate coefficients of fully parametric biquad cascades. We show that for the limited number of six biquadratic filters, ten neurons in the hidden layer of the neural equalizer can provide a measurable improvement in equalization performance over the baseline method. In its published form, the fully parametric equalizer has over 5 Million parameters, which is orders of magnitude larger than the tested feedforward neural equalizer with 136 parameters. Yet it is only in the higher octaves of the equalization range that the larger model outperforms the lean model in the earbud equalization task.

1 Introduction

Wireless earbuds have become ubiquitous in personal audio consumption, offering users convenience and mobility [1]. However, the variability in fit, seal and ear canal anatomy introduces significant inconsistencies in the sound frequency response of the pressure level (SPL) at the eardrum [2], leading to sub-optimal audio quality and potential listening fatigue [3, 4]. Furthermore, user movement can dynamically alter the fit and seal and thus the acoustic conditions, necessitating continuous adjustment to maintain a consistent SPL frequency response and auditory experience [5, 6].

Traditional equalization methods on earbuds are static, relying on preset filters that fail to compensate for real-time changes in fit and environment [7]. Auto-equalization (auto-EQ) of the SPL frequency response seeks to address the challenge by continuously monitoring the audio output and adapting the frequency response in real-time, ensuring a stable and optimized listening experience regardless of external or anatomical variations and fit quality of the earbud tip in the ear canal.

A critical requirement for auto-EQ systems in wearables is computational complexity. In particular, wireless earbuds are constrained by limited processing power, battery capacity, and thermal considerations,

due to their small size. Algorithms for real-time SPL equalization must therefore be both computationally lean and power-efficient in order to not compromise device runtime or generate perceptible latency [8]. Efficient algorithm design also minimizes memory usage and reduces the load on the embedded digital signal processor (DSP), which typically is shared with other audio enhancement features such as active noise cancellation and microphone beam-forming.

A simple and computationally efficient approach to auto-EQ on earbuds is the use of fixed center frequencies and quality factors (Q-factors) for a given number of second-order biquadratic filters (biquads). A straightforward approach to calculate the optimal gain values of the individual biquads leverages the self-similarity of biquad filters on the log-magnitude scale and can be solved with a least-squares approach [9]. However, the self-similarity approximation does not hold well when the gain values range widely [10]. In earbuds, the equalization requirements to compensate for leaks and poor fit of the earbuds can reach upwards of 30 dB. As we will show in this paper, that range is high enough to introduce noticeable errors.

Improvements upon the method in [9] have been presented in [10, 11], where neural feed-forward networks with one or two hidden layers are used to compute the required gains values, rather than a least-squared approach. Further accuracy improvements have been made in [12], where a larger neural net with multiple hidden layers and differentiable filters [13] were combined, offering the benefit of using a spectral loss during training rather than just a parameter loss. Furthermore, the method presented in [12] incorporated variable center frequencies and Q-factors. Because this approach is significantly more computationally complex, we opted to only test the lighter-weight neural equalizer method presented [11] against the baseline method of [9] for use in an adaptive equalization for wireless earbuds.

In the following sections we will introduce the test set of EQ targets obtained from measurements on synthetic ears, describe the baseline method and our implementation of the neural equalizer. We then compare the results of the equalization with the two methods and show how a simple neural equalizer can improve upon the performance of the baseline method.

2 Methods

We compare two implementations of lightweight auto-EQ algorithms on a test problem of an adaptive EQ of in-ear headphones with variable fits of the earbud tip against a number of user-specific ear canals.

2.1 EQ target

As mentioned in the introduction and outlined in detail in previous work [2, 14], the ear canal geometries and fit and seal of the earbud can greatly affect the response at the eardrum. Differences in ear canal geometry mainly affect the SPL frequency response at the eardrum above 1 kHz, while an imperfect seal can result in leakage affecting the response mainly below 1.5 kHz. The variance in magnitude of the frequency response at the eardrum due to ear canal geometry differences is on the order of several dB, but the variance in response at the eardrum due to fit and seal differences can reach upwards of 30 dB. We note that for this work we use measurements at the eardrum of synthetic ears which is not feasible in a consumer product worn on real ears. In our previous work [14], we showed how a target curve at the eardrum can be translated to a personalized target curve at the near-field microphone of an earbud outfitted with a microphone between earbud speaker and eardrum, as is typically used for noise-canceling applications.

Test Ears In order to generate a test scenario for the different EQ applications, we manufactured 5 different ears with varying ear canal geometries. The test ear geometries were generated from the statistical model published by the University of Michigan Transportation Research Institute (UMTRI) [15]. In this model, the first principal component correlates well with the ear canal length. The five ear geometries used in this study were selected as the median ear (Ear 0), ears at negative and positive 3 standard deviations of the first principal component[1] (Ear 1, and Ear 2, respectively), and two geometries with ear canal lengths in between Ear 0 and Ears 1, and 2 respectively.

The ear geometries were adapted in CAD software to accept a Brüel and Kjær (B&K) microphone type 4620 at the location of the eardrum. Finally, the geometry

[1] The first principal component of the ear geometries correlates well with the lateral position of the ear canal entrance, and thus also with ear canal length [15]

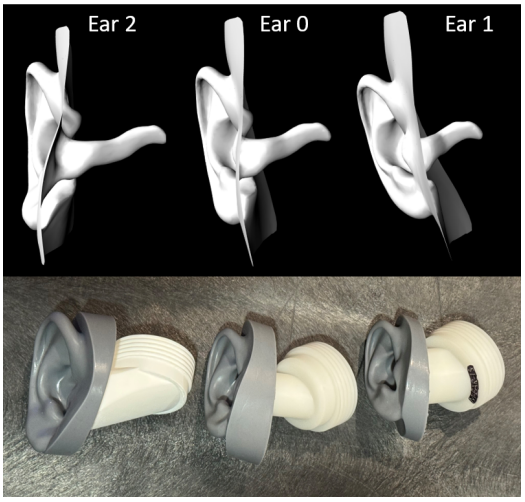


Fig. 1: Synthetic ear geometries and final manufactured products.

was split into two parts: The outer part of the ear up to approximately 10 mm inside the anterior notch of the ear was manufactured with a soft silicone of durometer Shore 5A; The inner part of the ear canal was 3D printed with an ABS-like material, which can be considered rigid for all intents and purposes.
The ear geometries and final manufactured ears of the mean and two extremal ears are shown in Figure 1.

Measurements An earbud was inserted into the synthetic ears and the transfer function from the voltage at the transducer leads to the pressure at eardrum reference point (DRP) was recorded with the B&K 4620 microphone. The input signal was a 30 second long random pink noise signal. Insert position and fit were varied and an attempt was made to include a wide variety of fits ranging from perfect seal to very loose fits. In total, 214 usable recordings were made, and from these recordings we obtained 214 EQ magnitude targets, defined as

$$EQ_i^*(f) = DRP^*(f) - DRP_i(f), \qquad (1)$$

where i is the measurement index, $DRP(f)$ is the magnitude of the frequency response of the transfer function from voltage at transducer terminals to pressure at DRP in decibels, and DRP^* is a chosen (fixed) target frequency response. The set of all target equalization curves is shown in Figure 2. Four arbitrarily selected curves are highlighted in color to better show differences between individual curves.

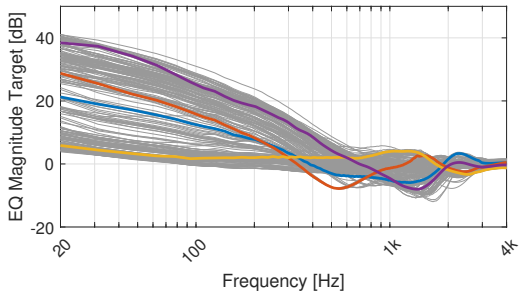


Fig. 2: EQ targets to test EQ algorithms. Four individual targets are highlighted in color as arbitrary examples to facilitate distinction between individual targets.

2.2 Baseline Method

As our baseline method, we implemented a least-squares method that leverages the self-similarity of the peaking filters with respect to gain changes on the log-magnitude scale [9]. The main difference to the implementation of the baseline algorithm in [12] is that we implemented an algorithm with six peaking biquad filters only. No shelving filter is implemented in the algorithm for this paper.

As stated in [12], the baseline algorithm requires an *a-priori* selection of center frequencies (fc_i) and quality factors (q_i). The biquad gains (g_i) are then calculated optimally with respect to the choice of fc_i's and q_i's in a least-squares sense: Given a desired magnitude response x_i^* of the EQ filter at N frequencies, the root-mean-squared error (RMSE) is defined as

$$RMSE(\mathbf{x}^*, \hat{\mathbf{x}}) = \sqrt{\frac{1}{N}||\mathbf{x}^* - \hat{\mathbf{x}}||_2^2} \qquad (2)$$

where $\mathbf{x}^*$ and $\hat{\mathbf{x}}$ are vectors of EQ target gains x_i^* and EQ estimated gains $\hat{x}_i$ at the chosen fc_i.

We skip the derivation of the frequency response of a peaking biquad filter here and refer to [12] or any digital signal processing handbook. Instead we define the magnitude of the frequency response of a peaking biquad H with center frequency fc, Q-factor q and gain in dB g as $|H(fc,q,g,f)|$. The self-similarity principle exploits the fact that

$$log|H(fc,q,g,f)| \approx g\, log|H(fc,q,1,f)| \qquad (3)$$

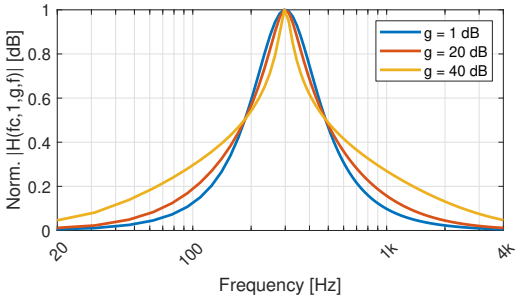


Fig. 3: Comparison of three normalized amplitude responses of biquad peaking filters with different gains. All filters have center frequencies of fc = 300 Hz, and quality factors of q = 1.

With this definition we can now define a basis matrix $\mathbf{B}$, containing the unit gains of each of the $j = 1, \cdots, N$ biquad peak filters at each of the fc_i ($i = 1, \cdots, N$):

$$\mathbf{B} = \begin{bmatrix} 20log|H(fc_1, q_1, 1, [fc_1, \cdots, fc_N])| \\ \vdots \\ 20log|H(fc_N, q_N, 1, [fc_1, \cdots, fc_N])| \end{bmatrix}^T \quad (4)$$

Finally, we can now calculate the baseline ($.^{(B)}$) optimal gains $\hat{g}_i^{(B)}$ that minimize the RMSE in equation (2):

$$\hat{\mathbf{g}}^{(\mathbf{B})} = \left(\mathbf{B}^T\mathbf{B}\right)^{-1}\mathbf{B}^T\mathbf{x}^* \quad (5)$$

2.3 Neural Equalizer

As shown in more detail in [7], the self-similarity for peaking biquad filters in equation (3) is not perfect. An example is given in Figure 3, where the normalized magnitude of two biquad peaking filters are plotted for an fc of 300 Hz, a q of 1, and gains g_1 of 1 dB, g_2 of 20 dB, and g_3 of 40 dB. Evidently, the frequency responses of the normalized filters are not exactly equal at all frequencies.

In order to better compensate for the inconsistencies when scaling the biquad filter to different gains, we employ a neural EQ approach taken from [10, 11], but we relax the problem to using only 6 peaking biquad filters, which results in a very small and efficient network. The detailed implementation of this network will be explained in more detail in the following paragraphs.

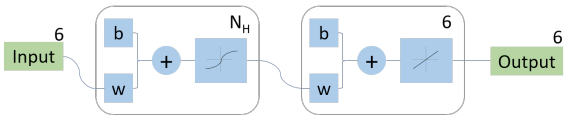


Fig. 4: Graph of the feed-forward neural net with N_H hidden neurons. w are the weights, and b are the biases.

Network Structure The structure of the neural network is that of a simple feed-forward network that can be generated by the Matlab function `fitnet`. The feed-forward neural net is a function-fitting neural network that can be trained to generalize the relation of the input data to the output data. The network has six inputs, six outputs, and a fully connected hidden layer with N_H neurons and a `tanh` activation, as well as a linear projection output layer that is commonly used in regression networks. A graph of the network can be seen in Figure 4.

Training Data The first attempt at solving the problem of finding optimal parameters for the six biquads was performed with the initial guess of equally distributing the center frequencies in the log-scale between 47 Hz and 2.5 kHz. The lower limit of the center frequencies is given by hardware restrictions of the earbuds, and the upper limit was chosen due to the fact that EQ targets in the test set above 2.5 kHz are already very small (see Figure 2). The initial guess for the Q-factors was estimated to be 0.8 for all biquads. With these assumptions we generated 5000 EQ filters

$$H_{EQ}(f) = \prod_{k=1}^{6} H(fc_k, q_k, g_k, f) \quad (6)$$

with uniformly chosen random gain levels g_k between the upper and lower boundaries g_{min} and g_{max} respectively for the training of the neural EQ algorithm. The training inputs consist of the db-gain of the EQ filters

$$EQ(f) = 20log|H_{EQ}(f)| \quad (7)$$

evaluated at fc's, i.e. $EQ(fc_1, ...fc_6)$ and the training output consists of the respective sampled gain values g_k in dB. The values of fc_k, q_k, and g_{min} and g_{max} are shown in Table 1.

Hidden Layer Size With the initial generated training data set and the parameters in Table 1, we ran a

Table 1: Parameters for the generation of training data for the neural EQ algorithm with the initial guesses of f_c's and Q-factors.

Biquad k	fc [Hz]	q [-]	g_{min} [dB]	g_{max} [dB]
1	47	0.8	-10	40
2	104	0.8	-10	30
3	230	0.8	-15	20
4	510	0.8	-15	20
5	1129	0.8	-15	15
6	2500	0.8	-15	15

study over different hidden layer sizes to determine optimal trade-off between number of learnable parameters (weights and biases) versus accuracy. A fully connected layer with n_i inputs and n_o outputs has $n_i \times n_o$ weights and n_o biases, so $(n_i+1)\times n_o$ learnable variables. For the network shown in Figure 4 with $n_i = 6$, can calculate the total number of learnable variables N_L as follows:

$$N_L = (6+1)N_H + (N_H+1)6 = 13N_H + 6 \quad (8)$$

This means N_L scales linearly with N_H.

The tested numbers of neurons in the single hidden layer were $2,4,\ldots,12,16,20$. The training was performed for 2000 epochs using the `trainlm` method of Matlab's `fitnet`, and the training data split into a 70-15-15 training-, validation-, and test-ratio. We had also explored the use of multiple hidden layers, but it did not improve the performance significantly enough to warrant the increased computational cost.

The performance of individual nets was tested using the EQ targets shown in Figure 2. The resulting RMSE over the test EQ filters for a selection of different hidden layer sizes is shown in Figure 5. We included the performance of the baseline method, and its drawbacks become evident at frequencies below 1 kHz, where the ranges of the gains g_k are largest and the approximation of (3) has the most significant error.

The neural EQ algorithm minimized the RMSE as stated in (2) and the error between fc's cannot be controlled directly. This drawback of the simple feed-forward neural net that only takes the parameter loss (2) into its cost function is well documented in [12], but the investigation of more sophisticated networks that can take the entire spectral loss into account is not the scope of this work, as it would likely be prohibitive to

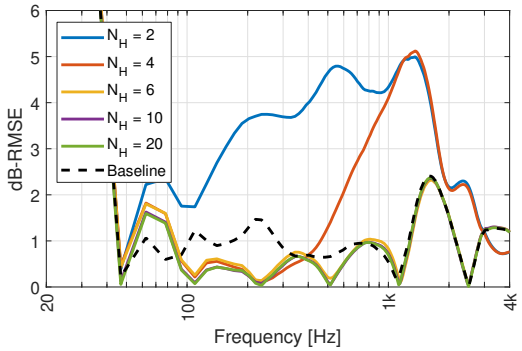


Fig. 5: RMSE's of the neural EQ algorithm for a selection of different hidden layer sizes. The performance of the baseline method is shown for comparison, and highlights the drawback of the method, which is particularly apparent at frequencies below 1 kHz, where the biquad gains g_k have a larger variance.

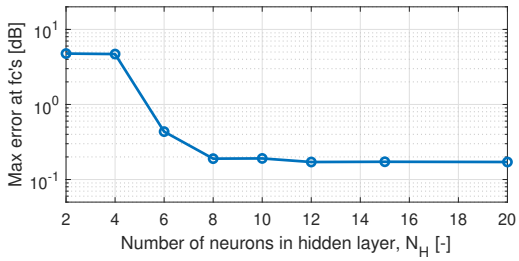


Fig. 6: Max RMSE at fc's for the neural EQ with N_H neurons in the hidden layer.

implement such networks for real-time equalization on earbuds with limited storage and computation power. Therefore, we limit the analysis of the effect of hidden layer size on the EQ filter error at the fc's only.

The maximum error at fc's for different N_H is plotted in Figure 6. From this result, one can see that the error is high for small N_H, and it levels out at or below 0.19 for $N_H >= 8$. This result shows that the network can indeed produce small error at fc's with a very small and simple network. For the remainder of the paper, we will operate with $N_H = 10$, which provides a small margin of error against uncertainty compared to the absolutely minimal case of $N_H = 8$.

Manual Optimization q's The results in Figure 5 show that the RMSE between fc's does not have equal local maxima. This suggests that a better choice of the

Table 2: Manually optimized and q's along with the log-spaced center frequencies fc.

Biquad k	$\hat{fc}$ [Hz]	$\hat{q}$ [-]
1	47	0.4
2	104	1.5
3	230	0.8
4	510	0.5
5	1129	1.0
6	2500	1.0

fixed fc's and q's than the initial guess shown in Table 1 might exist to reduce the maximum error in the band of interest. These optimizations can be done to tailor the equalizer to a particular task.

We found that the objective of reducing the maximal $RMSE(f)$ by varying fc's and q's is non-convex and has many local minima. We therefore resorted to a manual optimization of the q values only and left the center-frequencies untouched. This heuristic approach yielded the parameters shown in Table 2.

A more serious attempt at finding the globally optimal parameters could yield a better global performance of the neural equalizer, i.e. it is possible to reduce the maximum RMSE value in the band of interest, but it would not change the findings of this paper in a significant manner. The differences between the tested methods would still be apparent. The interested reader could attempt an implementation of a more sophisticated optimization method, such as genetic algorithms, particle swarm, or simulated annealing to optimize fc's along with the q's.

2.4 IIRNet

The baseline method and the neural equalizer both require the a-priori selection of center-frequencies and quality factors for the peaking filters. A filter-design approach that can optimize those parameters along with the gains has been proposed in [16], where a neural net (IIRNet, [17]) has been trained by sampling random polynomials to map a target magnitude response directly to the filter coefficients of a biquad cascade. As such IIRNet does not optimize parameters of peaking or shelving filters, but it directly optimizes the coefficients of any biquadratic filter. This is a powerful approach for the direct design of filter cascades, however it comes at some cost.

As implemented in the published version, IIRnet is composed of 2 fully connected linear layers, each followed by layer normalization, and LReLU activation. The authors experimented with the number of neurons in the hidden layers, varying it from a low value of 64 to a high value of 4096. The downloadable pre-trained models that are used in this work have 2048 neurons in each hidden layer, resulting in a total of 5.35 million parameters that need to be stored. For comparison, the neural equalizer presented above only requires storage of 136 parameters for a hidden layer size of $N_H = 10$.

The loss function $\mathscr{L}$ that IIRnet has been trained on is calculated as the mean squared error of the log of the magnitude responses X of the estimated and target magnitude responses $\hat{X}$ over a set of F linearly spaced frequencies

$$\mathscr{L} = \frac{1}{F} \left\| \log(\hat{X}) - \log(X) \right\|_2^2. \tag{9}$$

Using the log of the magnitude response $\log(X)$ is akin to taking the dB-error between filter target and estimate in (7). Where IIRNet and the neural equalizer differ more substantially is the frequency sampling: IIRNet samples the frequencies linearly, while the neural equalizer essentially has a log-spacing of the center frequencies, where the RMSE loss function (2) is calculated. This small but crucial detail focuses the IIRNet biquads onto the higher octaves, while leaving only few biquads for the lower octaves.

The pre-trained models of IIRNet allow for a filter order of 4, 8, 16, 32, or 64, i.e for cascades of 2, 4, 8, 16, or 32 biquads. In this work, we used a filter order of 16 (8 biquads) to be comparable with the limitation of 6 peaking filters used in the baseline method and neural equalizer.

3 Results

In Figure 7 we plot the same sample of target EQ curves as in Figure 2, and we compare them with the predictions from the neural equalizer, baseline method, and IIRNet with default settings and filter order 16. Where the EQ gains are large, there is a clear difference between the baseline method and the neural equalizer. The baseline method is not able to match the target at the second and third center frequency for the purple example curve. The neural equalizer matches the target with better accuracy at the center frequencies.

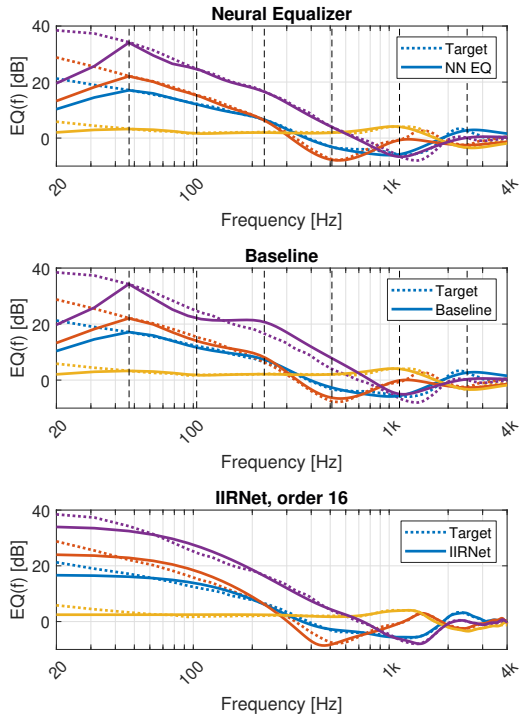


Fig. 7: Examples of EQ targets and the predictions of the 6-band EQ's using optimized fc's and q's for the neural equalizer and baseline method. Dashed vertical lines correspond to fc's.

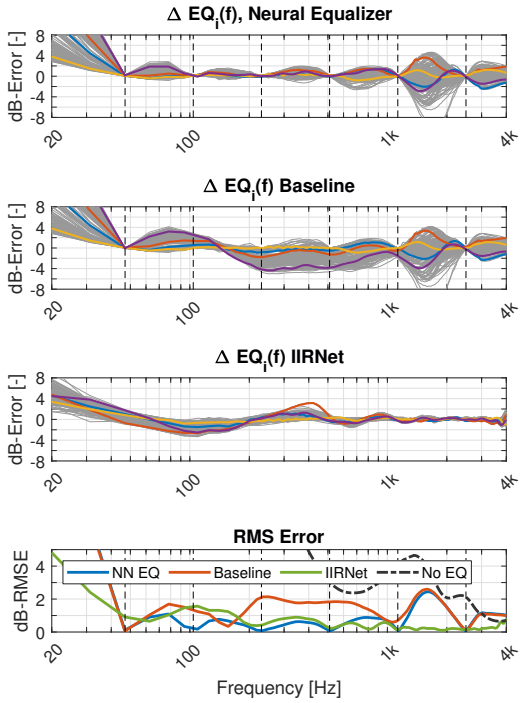


Fig. 8: Performance of the neural equalizer network with $N_H = 10$ and optimized fc's and q's compared to baseline method and the RMSE without EQ. Dashed vertical lines correspond to fc's. The RMS error for the "No EQ" case is not fully shown, but it grows to 16.7 dB at 100 Hz and 25.1 dB at 20 Hz. It is essentially the RMS value of the curves shown in Figure 2.

As mentioned in the previous section, the IIRNet method is based on linear frequency sampling and thus has more poles at higher octaves than the other two methods, which use log-spaced center frequencies by design. The result is that IIRNet matches the target at higher frequencies extremely well, struggles with accuracy at lower frequencies.

The differences $\Delta EQ_i(f)$ between all the EQ target curves $EQ_i^*(f)$ and the predicted $\hat{EQ}_i(f)$ for the different equalizer methods are shown in Figure 8. The colored lines in the top three graphs of Figure 8 correspond to the examples shown in Figures 2 and 7. The fourth plot in the figure shows the RMSE over all target curves for each of the three methods, along with the RMSE of an earbud without equalization. The baseline method and neural equalizer method used the fc's and q's shown in Table 2.

The results show that the approximation in Equation (3), upon which the baseline method is founded, can lead to significant errors when the gain parameters g_k are much larger than 1. In the set of the four example cases, this error is most apparent in the purple curves of Figures 7 and 8. The RMSE plots in Figure 8 show that the IIRNet method outperforms the other two methods above 500 Hz, but it underperforms below that frequency compared to the neural equalizer. An adapted version of the IIRNet that is based on log-spaced frequency sampling could balance the accuracy at lower and higher octaves better. The question still remains whether a neural net of the size of IIRnet offers a feasible approach that could be implemented in a wireless earbud.

The results also show that although the RMSE has been reduced drastically overall by either EQ approach compared to the non-equalized solution, when the EQ target maxima and/or minima are located between the chosen fc's there can be some non-zero error between

$\Delta EQ_i^*(f)$ at the frequencies between fc's. Neither the baseline nor the neural equalizer have enough control over the EQ response between fc's, but with six biquads, we have managed to keep the max RMSE of the neural equalizer below 1 dB between 47 Hz and 1.1 kHz. At frequencies above 1.1 kHz the target has some narrow-band variations that cannot be capture with a single biquad peaking filter, and the RMSE in this band grows to 2.5 dB at 1.6 kHz. A more sophisticated choice of fc's, q's can reduce this large error peak, but it likely would come at a cost of an increased error in another frequency band. This underlines the challenge to pick optimal fc's and q's for the problem at hand with the equalizer methods that estimate optimal gain values only.

4 Conclusion

This study shows how a small neural network can improve the EQ performance over the least-squares baseline method for a case of auto-equalization in wireless earbuds, where EQ performance must be balanced with storage and computation requirements. The simple feed-forward neural net is able to learn the non-linear relationship between gain values of peak biquad filters and the frequency response shape, where the baseline method assumes perfect similarity in Equation (3).

Fixed center frequencies and Q-factors still expose some potential for errors in the predicted filter response between the chosen center frequencies. This is particularly true when the EQ target curve has an extremum that lies between a pair of center frequencies. It is therefore recommended to choose fc's and q's carefully for the chosen task at hand. We have used measurements of earbuds in synthetic ears of different shapes and sizes to generate a realistic test-set. For this test-set and under the limitation of only using six peaking biquad filters, we have found that the neural equalizer can maintain an RMS error of less than 1 dB in the band between 47 Hz and 1.1 kHz, where the baseline method has an RMSE of up to more than 2 dB.

Compared to the published version of IIRNet, a fully parametric equalizer, the neural method was able to achieve better performance at lower octave-bands, but at higher octave-bands the IIRnet method achieved remarkably high accuracy.

In order to improve further upon the EQ accuracy of the neural equalizer, two logical avenues readily present themselves: 1) Use a higher number of biquad filters and 2) Optimize the choice of fixed fc's to better balance the RMSE at bands of high RMSE with bands of lower RMSE. The former options will increase the demand on storage and compute power on the device, and the trade-off between the resource cost and accuracy must be weighed for each application. A modified version of IIRNet, tailored to the problem at hand with log-spaced frequency sample, offers promise to achieve very high accuracy across all octaves in the frequency range of interest. The relatively high computational cost of IIRNet compared to the much leaner neural equalizer might still make it unfeasible for a real-time application in resource-limited hardware.

5 Acknowledgments

Samsung Electronics and Samsung Research America supported this work. The authors would like to thank the entire staff of Samsung's US Audio Lab who helped with all aspects of this research, offered insightful suggestions, and contributed to this work.

References

[1] Blauert, J., *Spatial Hearing*, The MIT Press, 1996, ISBN 9780262268684, doi:10.7551/mitpress/6391.001.0001.

[2] Bezzola, A., "In-Ear Headphones on Ear Canal Simulator vs Real Human Ear Geometries: Quantifying the Differences with Simulations," in *Audio Engineering Society Convention 155*, Audio Engineering Society, 2023.

[3] Stinson, M. R. and Lawton, B. W., "Specification of the geometry of the human ear canal for the prediction of sound-pressure level distribution," *Journal of the Acoustical Society of America*, 85, pp. 2492–2503, 1989, ISSN NA, doi:10.1121/1.397744.

[4] Toole, F., *The Acoustics and Psychoacoustics of Loudspeakers and Rooms*, Focal Press, 2 edition, 2013, ISBN 978-0-240-52009-4.

[5] Celestinos, A., Mcmullin, E., Banka, R., and Brunet, P., "Personalized and Self-Adapting Headphone Equalization Using Near-Field Response," in *147th Audio Engineering Society Convention*, Audio Engineering Society, 2019.

[6] 11904-2:2021, I., "Acoustics - Determination of sound immission from sound sources placed close to the ear - Part 2: Technique using a manikin," Technical report, ISO, 2021.

[7] Liski, J., Välimäki, V., Vesa, S., and Vaananen, R., "Real-time adaptive equalization for headphone listening," in *2017 25th European Signal Processing Conference (EUSIPCO)*, pp. 608–612, IEEE, 2017, ISBN 978-0-9928626-7-1, doi:10.23919/EUSIPCO.2017.8081279.

[8] Kamaris, G., Zachos, P., and Mourjopoulos, J., "Low Filter Order Digital Equalization for Mobile Device Earphones," *AES: Journal of the Audio Engineering Society*, 69, pp. 297–308, 2021, ISSN 15494950, doi:10.17743/jaes.2021.0003.

[9] Abel, J. S. and Berners, D. P., "Filter Design Using Second-Order Peaking and Shelving Sections," in *Proceedings of the International Computer Music Association Conferences*, 2004.

[10] Rämö, J. and Välimäki, V., "Neural third-octave graphic equalizer," in *Proceedings of the 22nd International Conference on Digital Audio Effects (DAFx2019)*, Proceedings of the International Conference on Digital Audio Effects, 2019, ISSN 2413-6689.

[11] Välimäki, V. and Rämö, J., "Neurally Controlled Graphic Equalizer," *IEEE/ACM Transactions on Audio Speech and Language Processing*, 27, pp. 2140–2149, 2019, ISSN 23299304, doi:10.1109/TASLP.2019.2935809.

[12] Nercessian, S., "Neural Parametric Equalizer Matching Using Differentiable Biquads," in *Proceedings of the 23rd International Conference on Digital Audio Effects (DAFx2020)*, pp. 265–272, 2020.

[13] Engel, J., Hantrakul, L., Gu, C., and Roberts, A., "DDSP: Differentiable Signal Processing," in *Proceedings of the International Conference on Learning Representation*, 2020.

[14] Bezzola, A., Celestinos, A., and Blanes, E. S., "Personalized Equalization of Sound Pressure at Eardrum With Insert Earbuds," 2024.

[15] Reed, M. P., Vallier, T. R., and Bonifas, A. C., "Development of a parametric model of adult human ear geometry," *International Journal of Industrial Ergonomics*, 107, p. 103738, 2025, ISSN 01698141, doi:10.1016/j.ergon.2025.103738.

[16] Colonel, J. T., Steinmetz, C. J., Michelen, M., and Reiss, J. D., "Direct Design of Biquad Filter Cascades With Deep Learning by Sampling Random Polynomials," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, volume 2022-May, pp. 3104–3108, Institute of Electrical and Electronics Engineers Inc., 2022, ISBN 9781665405409, ISSN 15206149, doi:10.1109/ICASSP43922.2022.9747660.

[17] Steinmetz, C. and Colonel, J. T., "https://github.com/csteinmetz1/IIRNet," 2022.