Rev. Roum. Sci. Techn.– Électrotechn. et Énerg.
Vol. **64**, *4*, pp. 403–408, Bucarest, 2019

QUASILOGARITHMIC QUANTIZER FOR LAPLACIAN SOURCE: SUPPORT REGION UBIQUITOUS OPTIMIZATION TASK

ZORAN PERIĆ[1], DANIJELA ALEKSIĆ[2]

Key words: Support region, Clipping, Quasilogarithmic quantizer parametrization, Curve overlapping.

Signal coding, as the process of obtaining a suitable digital representation of signals, is an important aspect of modern telecommunication environment. To attain desirable signal quality, keeping the relevant signal attributes, while diminishing the signal distortion, different quantizers, their parametrization and coding methods, are suggested. In this paper we address the iterative method for the support region threshold determination, as the segment of good parametrization for robust μ-law quasilogarithmic quantizer, designed for the Laplacian source of unit variance. The simple solution that we propose, overcomes the shortcomings and impediments that can be influenced by the unpredictable nature of speech and audio signals, approximated by the Laplacian probability density function. With the increasing widespread quantization deployment in many contemporary applications, the effectiveness of the proposed robust quasilogarithmic quantizer solution can be recognized as the part of the ubiquitous optimization task. A growing interest in deep neural network (DNN) is directed towards to the efficient inferencing using the compressed models.

1. INTRODUCTION

We are constantly interested in the objects in the environment that use signals to carry information about. Despite the high degree of complexity of these signals, it is obvious aspiration, to develop new and expand existing methods and techniques that enable valid signal transmission and recognition.

Speech is indeed the simplest and the most efficient mean of human communication. Legacy voice is mature, robust and high performing. Voice service continues to be globally forecasted as the challenge that traditionally has a relatively predictable and small use of bandwidth, but requires real time transmission.

Nowadays, different applications utilize many modules which are related to signal processing, including signal analysis and synthesis of natural or artificial speech signals, additional signal enhancement, signal coding and compression. The Internet of Things (IoT) aims to leverage on devices with the ability to communicate with corresponding devices, using Internet. With the revolution of telecommunications, the number of connected devices has already exceeded the world’s population and is increasing exponentially, with an optimistic further growth trend [1].

Correspondingly, the existing challenges, associated with network traffic handling and capacity limitation, numerous advanced solutions has been offered to support the successful deployment of massive IoT and 5G [2–5]. Surely, these state-of-the-art solutions provide more efficiency to manage the data having the users, individual types of accesses or services into focus. Apart from scaling existing network infrastructures, the increasing demands of user applications have pulled up the explosive data traffic. As communication services evolve, a variety of novel technologies are laboring to improve the communication system performances, bringing about larger data transmission, with the optimized data rate [6].

In overall digital signal processing, quantization of any kind of signal, has undeniably important role. The main hurdles to the speech quantization is primarily, the unpredictable nature of speech signals. Well-designed quantizer reduces the potential influence of unpredictable statistical characteristics of the speech signals to the speech quality, thereby, contributing to increased robustness.

As well known, the data compression models can assume a source of information with known input statistical properties. We assume here Laplacian probability density function, which is widely accepted as a good approximation, for the distinctive attributes of speech and audio signals [7]. Hence, tuning for some of predominant quantizer’s parameter for Laplacian source is facilitated, according to the fact, that many attributes, as the compression function, can be expressed in closed form.

Laplacian distribution can be used for analysis of symmetric bell-shaped data, such as activation tensors used in deep learning models [8]. Having the low-precision scheme that involves a floating-point to integer conversion [8] at the forefront, there is an emphasis on the optimization from the standpoint of the analytical clipping.

Analytical clipping for integer quantization (ACIQ) entails the optimized clipping value assessment. The main trade off in clipping process is, making the dynamic range large enough to accommodate the activation tensor values, while keeping this dynamic range small enough to minimize the quantization effects. Despite a possible degradation in activation tensors prediction, the proposed clipping approach is particularly beneficial for implementing in models with the extreme memory requirements, such as mobile devices and embedded systems [9].

The clipping process formulation reminiscent of the support region determination, while the same objective is to make the mean-square quantization error (MSE) negligible [8]. By normalizing the inputs in neural network (NN), having a zero mean and unit variance, one can effectively obtain more appropriate learning of the optimal outcomes, while avoiding overshooting and optimal outcomes mismatch [10, 11].

We attempt to tackle the possible clipping value by suggesting simple method for its offline calculation, which can be stored for retrieval. Our anticipation is that, such a precalculated value can be leveraged in deterministic quantization process [9]. It is worth emphasizing that

[1] Faculty of Electronic Engineering, Department of Telecommunications, University of Niš, E-mail: zoran.peric@elfak.ni.ac.rs.
[2] Telekom Srbija, Department of mobile network Niš, E-mail: danijelaal@telekom.rs.

optimal clipping is a part of ubiquitous optimization task for quantized neural network (QNN), that alleviate performance degradation influenced by the quantization.

In this paper, we focus particularly on quasilogarithmic quantizer and its clipping value is henceforth referred to as support region. Subsequently, the precise goal is to determine simple formula for the support region of quasilogarithmic quantizer, having in mind the influence of the accuracy of its determination on the total distortion, introduced in the quantization process. Beside the foregoing effort, of particular interest is to offer a simple manner for determining the support region threshold, as a part of overall quantizer parametrization task.

The rest of the paper is organized as follows. Section 2 recalls some basic facts about importance of the support region determination and offers an iterative method for its calculation, for quasilogarithmic quantizer, designed for the Laplacian source and an arbitrary compression factor. The achieved numerical results are the topic discussed in Section 3. Finally, Section 4 concludes the paper by summarizing the contribution achieved in this paper. Additionally, in Section 4, we offer motivation for our simple solution, by defining the use cases and possible further challenges.

2. SUPPORT REGION OF THE QUASILOGARITHMIC QUANTIZER DESIGNED FOR THE LAPLACIAN PDF

Signal coding refers to the process of extracting and transforming of the raw analog information from the signal, into digitally coded representations, to facilitate the access to descriptive and invariant attributes of this signal [12]. Signal decoding, as the reciprocal process of the signal coding, has the main task to obtain, close to an original signal, using the extracted attributes, to recover actual analog signal on the receiver end. Furthermore, signal companding method, based on A-law or μ-law tries to represent any signal, using a few bits whenever possible, maintaining at the same time, the required amount of fidelity and signal quality. Following the main aspect of data coding and compression, as the shrinkage on the number of bits, necessary to deliver original analog signal, it is obvious, that companding quantizers have been receiving much research interest, especially if they are well suited for less design complexity and implementation solution.

For the fixed-length or variable-length codewords, one way to overcome the coding efficiency challenge is to use a combination of two or more coders of any type, as a hybrid quantizer solution [13]. Without losing of generality, we can assume that nonuniform quantization allows better utilization of the quantization bits [14]. Underlying signal fidelity conditions are measured by an acceptable total distortion, formed from the granular and the overload distortion, that behave different according to various quantizer parametrization [12].

One heuristic solution for the support region of the product polar companded quantizer has already been shown in [15], while [16] deals with the support region estimation of semilogarithmic quantizer for Laplacian source.

A new scheme for wideband speech signal coding, based on modified G.711.1 standard solution, with a lower processing delay and higher quality of reconstructed signal has been suggested in [17]. Additionally, the support range values of quasilogarithmic quantizers, used as G.711 coders in both encoded branches, have been proposed.

For a quasilogarithmic quantizer Q_μ^N designed for a variance σ_p^2, where μ is the compression factor and N is the number of quantization levels, compression is performed using the μ- law compressor function:

$$c_\mu(x): [-x_{\max}^{\sigma_p}, x_{\max}^{\sigma_p}] \rightarrow [-x_{\max}^{\sigma_p}, x_{\max}^{\sigma_p}] \ [12, 18]$$

$$c_\mu(x) = \frac{x_{\max}^{\sigma_p}}{\ln(1+\mu)} \ln\left(1+\mu\frac{|x|}{x_{\max}^{\sigma_p}}\right)\mathrm{sgn}(x), \quad (1)$$

for $|x| \le x_{\max}^{\sigma_p}$, wherein $x_{\max}^{\sigma_p}$ stands for the μ-law companding quantizer's support region threshold.

We assume that quasilogarithmic quantizer is designed for the unit variance $\sigma_p^2 = 1$, while the value of the parameter μ is arbitrary. Accordingly, in what follows we use $x_{\max}$ instead of $x_{\max}^{\sigma_p}$ to simplify the notation.

Optimization of the support region threshold for the μ-law companding quantizer, addressed in [18], has been performed under the constraint, that compression factor μ has a large enough value.

For a given quantizer model wherein $[-x_{\max}, x_{\max}]$ defines the μ-law companding quantizer's support region, the expression for the total distortion is given by [12, 18]:

$$D(Q_\mu) = C\left[\frac{x_{\max}^2}{\mu^2} + \frac{\sqrt{2}x_{\max}}{\mu} + 1\right] + \exp\left(-\sqrt{2}x_{\max}\right), \quad (2)$$

where $C = \ln^2(\mu+1)/(3N^2)$ is a constant.

By minimizing distortion, that is by setting the first derivate of the so obtained distortion $D(Q_\mu)$ with respect to $x_{\max}$ equal to zero:

$$\frac{\partial \mathrm{D}(Q_\mu)}{\partial x_{\max}} = \frac{2Cx_{\max}}{\mu^2} + \frac{C\sqrt{2}}{\mu} - \sqrt{2}\exp\left(-\sqrt{2}x_{\max}\right) = 0 \,. \quad (3)$$

We derive the following expression for the optimal support region threshold of the considered quasilogarithmic quantizer:

$$x_{\max} = \frac{1}{\sqrt{2}} \ln\left[\frac{1}{C\left(\frac{\sqrt{2}x_{\max}}{\mu^2} + \frac{1}{\mu}\right)}\right]. \quad (4)$$

Obviously, in order to determine $x_{\max}$, the application of an iterative numerical method is required:

$$x_{\max}^{(i)} = \frac{1}{\sqrt{2}} \ln\left(\frac{\mu}{C\left(1 + \frac{\sqrt{2}x_{\max}^{(i-1)}}{\mu}\right)}\right). \quad (5)$$

The performances of optimal companding quantizer for memoryless Laplacian source has been ascertained [19], by applying the simple compression algorithm with an additional bit rate improvement. However, robustness of the optimal companding quantizer on the changing variance of the input signal is much smaller compared to the companding quantizer we consider here.

It is worth mentioning that in [19], three analytical estimates of the optimal maximal signal amplitude have been derived. Accepting that an optimal distortion estimation task it fulfilled by minimizing the MSE, one of estimates has been pointed out as suitable for practical applications [19].

To accelerate the estimation of $x_{\max}$, we initialize the iterative algorithm using the support region threshold $t_{\max}{}^{(3)}$ obtained in [19]:

Step1. The initialization of the algorithm:

$$t_{\max}{}^{(3)} = \frac{3}{\sqrt{2}} \ln[N+1],$$

$$x_{\max}{}^{(1)}\Big|_{x_{\max}{}^{(0)}=t_{\max}{}^{(3)}} = \frac{1}{\sqrt{2}} \ln\left(\frac{\frac{\mu^2}{C}}{\mu + 3\ln(N+1)}\right)$$

$$= \frac{1}{\sqrt{2}} \ln\left(\frac{\mu}{C}\right) + \frac{1}{\sqrt{2}} \ln\left(\frac{\mu}{\mu + 3\ln(N+1)}\right). \tag{6}$$

Step2. Computation of new threshold value $x_{\max}{}^{(2)}$ using (5):

$$x_{\max}{}^{(2)} = \frac{1}{\sqrt{2}} \ln\left(\frac{\frac{\mu^2}{C}}{\mu + \ln\left(\frac{\frac{\mu^2}{C}}{\mu + 3\ln(N+1)}\right)}\right). \tag{7}$$

Step3. Interruption of the iteration method with the estimation of relative error defined as:

$$\delta[\%] = \left| \frac{x_{\max}{}^{(2)} - x_{\max}{}^{(1)}}{x_{\max}{}^{(2)}} \right| \times 100. \tag{8}$$

3. THE NUMERICAL RESULTS

In this section, we discuss the appropriateness of the suggested support region determination solution. Also, our objective is to ascertain and summarize the influence of $x_{\max}$ on the performances of quasilogarithmic quantizer.

The particular idea of determining such a simplified asymptotic formula originate from the fact that $t_{\max}{}^{(3)}$ estimation given in [19], is close to an optimal. Because of the prominence of this good initial estimation, the outcome of further finely tuned, iteratively calculated value for $x_{\max}$, is expectedly good.

In Figs. 1–3 overall distortion is plotted for a large, medium and small compression values μ (μ = 255, μ = 100, μ = 10), as the function of $x_{\max}{}^{(0)}$, $x_{\max}{}^{(1)}$ and $x_{\max}{}^{(2)}$. The marked points on the curve are corresponding values for the $x_{\max}{}^{(0)}$, $x_{\max}{}^{(1)}$ and $x_{\max}{}^{(2)}$, calculated for the range of quantization levels N (N=16, 32, 64, 128, 256, 512, 1024).

In what follows we analyze the curve/curve overlapping and discuss the intersection and partial overlapping along the finite part of their length. Note that curve overlapping is an illustrative indicator and hence, the convergence of the iterative process is better, when the overlapping is higher.

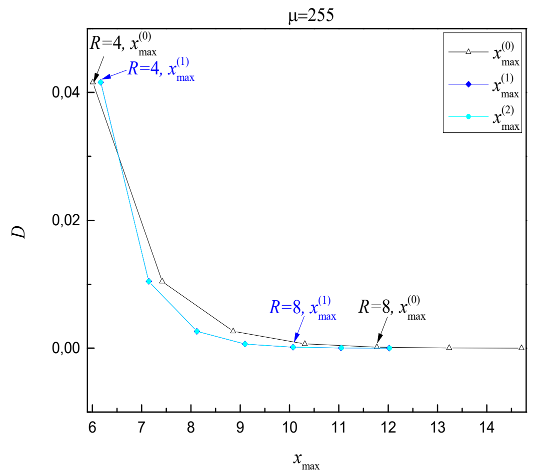

Fig. 1 – Distortion dependence on the support region values $x_{\max}{}^{(0)}$, $x_{\max}{}^{(1)}$ and $x_{\max}{}^{(2)}$ for μ = 255 and R = 4-10 bit/sample.

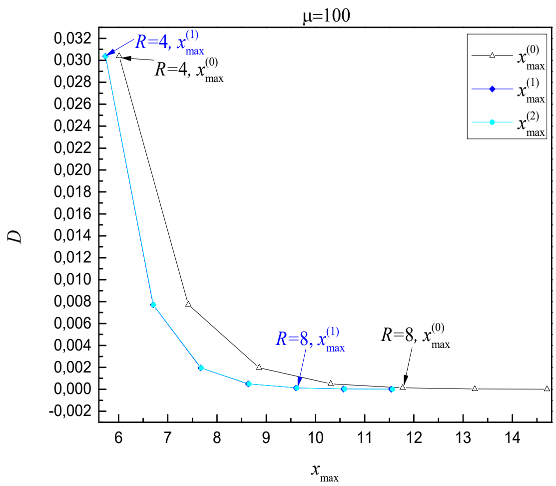

Fig. 2 – Distortion dependence on the support region values $x_{\max}{}^{(0)}$, $x_{\max}{}^{(1)}$ and $x_{\max}{}^{(2)}$ for μ = 100 and R = 4-10 bit/sample.

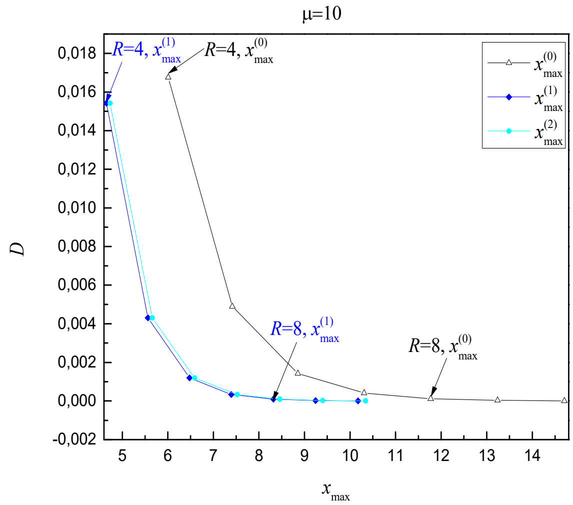

Fig. 3 – Distortion dependence on the support region values $x_{\max}{}^{(0)}$, $x_{\max}{}^{(1)}$ and $x_{\max}{}^{(2)}$ for μ = 10 and R = 4-10 bit/sample.

The above Fig. 1, for μ =255, depicts curves for $x_{max}^{(2)}$ and $x_{max}^{(1)}$ that intersect $x_{max}^{(0)}$ curve for R between 4 bit/sample and 5 bit/sample. Used curve's colors, help distinguish the curves involved in an intersection, while blending of the colors is sufficient to discern almost completely overlapping for $x_{max}^{(2)}$ and $x_{max}^{(1)}$ curves while μ = 255.

One can notice a good overlapping of $x_{max}^{(2)}$ and $x_{max}^{(1)}$ curves for μ = 100. It is also noticeable that for μ = 10 partial overlapping for $x_{max}^{(2)}$ and $x_{max}^{(1)}$ along the finite part of their length is achieved for R ranging from 7 bit/sample to 10 bit/sample.

Namely, if the initial value of $\overset{\equiv}{x}_{max}^{(0)}$ for the suggested iterative method is equal to $x_{max}^{a} = 1/\sqrt{2}\cdot\ln(\mu/C)$, which is obtained from an approximate closed-form formula for the support region threshold of the quasilogarithmic quantizer given in [18], the threshold value $\overset{\equiv}{x}_{max}^{(1)}$ can be calculated as:

$$\overset{\equiv}{x}_{max}^{(1)}\Big|_{\overset{\equiv}{x}_{max}^{(0)}=x_{max}^{a}} = \frac{1}{\sqrt{2}}\ln\left(\frac{\frac{\mu}{C}}{1+\frac{1}{\mu}\ln\left(\frac{\mu}{C}\right)}\right) = $$

$$= \frac{1}{\sqrt{2}}\ln\left(\frac{\mu}{C}\right)+\frac{1}{\sqrt{2}}\ln\left(\frac{\mu}{\mu+\ln\left(\frac{\mu}{C}\right)}\right), \qquad (9)$$

A new threshold value $\overset{\equiv}{x}_{max}^{(2)}$ can be calculated as:

$$\overset{\equiv}{x}_{max}^{(2)} = \frac{1}{\sqrt{2}}\ln\left(\frac{\frac{\mu}{C}}{1+\frac{1}{\mu}\ln\left(\frac{\frac{\mu}{C}}{1+\frac{1}{\mu}\ln\left(\frac{\mu}{C}\right)}\right)}\right). \qquad (10)$$

As listed in Tables 1–3 according to (6), (7), (8), (9) and (10), both iterative methods exhibit very fast convergence.

Notice that the initial value of $t_{max}^{(3)}$ is not influenced with the compession factor μ. Namely, that is confirmed by the same initial values of $x_{max}^{(0)}$ regardless to μ.

Comparing $x_{max}^{(2)}$ and $\overset{\equiv}{x}_{max}^{(2)}$ for μ =255 and μ =100 (according to Table 1 and Table 2), one can notice match up to third decimal place, while for μ =10 (Table 3) the matching is achieved up to the second decimal place.

While essentially the support region threshold value is predominantly influenced by $x_{max}^{a} = 1/\sqrt{2}\cdot\ln(\mu/C)$, and $\overset{\equiv}{x}_{max}^{(0)}$, as an initial guess, is reasonably close to the x_{max}^{a}, an iterative process converges very fast. For the simplicity purpose, we assumed that the x_{max} finding can be accomplished after only one iteration.

Table 1

Iterative values of Support region threshold for two different initialization (μ = 255)

μ = 255	$x_{max}^{(0)}$	$x_{max}^{(1)}$	$x_{max}^{(2)}$	$\overset{\equiv}{x}_{max}^{(0)}$	$\overset{\equiv}{x}_{max}^{(1)}$	$\overset{\equiv}{x}_{max}^{(2)}$
R = 4	6.0102	6.1705	6.1699	6.19369	6.1698	6.1699
R = 5	7.4172	7.1454	7.1465	7.17395	7.1464	7.1465
R = 6	8.8552	8.1203	8.1231	8.15420	8.1229	8.1231
R = 7	10.3092	9.0952	9.0997	9.13446	9.0995	9.0997
R = 8	11.7714	10.0700	10.0763	10.11472	10.0761	10.0763
R = 9	13.2376	11.0449	11.0529	11.09498	11.0528	11.0529
R = 10	14.7059	12.0198	12.0296	12.07524	12.0294	12.0296

Table 2

Iterative values of support region threshold for two different initialization (μ =100)

μ = 100	$x_{max}^{(0)}$	$x_{max}^{(1)}$	$x_{max}^{(2)}$	$\overset{\equiv}{x}_{max}^{(0)}$	$\overset{\equiv}{x}_{max}^{(1)}$	$\overset{\equiv}{x}_{max}^{(2)}$
R = 4	6.0102	5.7337	5.7363	5.7914	5.7357	5.7363
R = 5	7.4172	6.7011	6.7076	6.7717	6.7070	6.7076
R = 6	8.8552	7.6685	7.6791	7.7519	7.6784	7.6790
R = 7	10.3092	8.6359	8.6507	8.7322	8.6498	8.6506
R = 8	11.7714	9.6036	9.6224	9.7124	9.6214	9.6222
R = 9	13.2376	10.5714	10.5942	10.6927	10.5931	10.5940
R = 10	14.7059	11.5394	11.5660	11.6730	11.5649	11.5658

Table 3

Iterative values of support region threshold for two different initialization (μ=10)

μ = 10	$x_{max}^{(0)}$	$x_{max}^{(1)}$	$x_{max}^{(2)}$	$\overset{\equiv}{x}_{max}^{(0)}$	$\overset{\equiv}{x}_{max}^{(1)}$	$\overset{\equiv}{x}_{max}^{(2)}$
R = 4	6.0102	4.6542	4.7316	5.0892	4.7058	4.7285
R = 5	7.4172	5.5622	5.6591	6.0694	5.6313	5.6552
R = 6	8.8552	6.4756	6.5900	7.0497	6.5606	6.5856
R = 7	10.3092	7.3940	7.5239	8.0299	7.4934	7.5190
R = 8	11.7714	8.3172	8.4604	9.0102	8.4292	8.4552
R = 9	13.2376	9.2445	9.3993	9.9905	9.3676	9.3939
R = 10	14.7059	10.1754	10.3403	10.9707	10.3084	10.3349

Table 4

Relative error for two different initialization

Bit/sample	δ [%] μ = 255	δ [%] μ = 100	δ [%] μ = 10
R = 4	0.0099	0.0445	1.6354
R = 5	0.0143	0.0971	1.7119
R = 6	0.0340	0.1384	1.7364
R = 7	0.0496	0.1706	1.7254
R = 8	0.0624	0.1958	1.6925
R = 9	0.0729	0.2155	1.6470
R = 10	0.0815	0.2309	1.5950

In order to account for the assumption validation, the relative error of estimating the support region threshold is calculated for μ=255, μ=100, μ=10 and a different number of quantization level N (N=16, 32, 64, 128, 256, 512, 1024).

By calculating $x_{\max}^{(1)}$ the first degree dependence on μ is achieved. An apparently good overlapping of $x_{\max}^{(2)}$ and $x_{\max}^{(1)}$ curves is visible for μ=255 and μ=100, while R range from 4 bit/sample to 10 bit/sample, so we can accept that only one iteration is sufficient for the exact $x_{\max}$ determination. Since δ approximately amounts to 0.2 %, in the worst case, we have confirmed the correctness of the introduced assumption.

In Table 4, observe that for μ=10, while R range from 4 bit/sample to 10 bit/sample, δ approximately amounts to 1.7 %. Accepting that δ is within the confines of acceptable error, even in this case $x_{\max}^{(1)}$ can be approved as a good estimation for $x_{\max}$.

It should be noted that irrespective of what μ value is selected (μ = 10, 100 or 255), the coding efficiency of suggested quantizer is improved by selecting $x_{\max}^{(1)}$, while $D(Q_\mu^{N=16})$ and $D(Q_\mu^{N=256})$ are obviously less then MSE for 4 bit and 8 bit quantization, shown in [8].

4. SUMMARY AND CONCLUSION

It is a real challenge to predict what direction signal processing will take in the future, with respect to the fact, that this field has occupied an attention over the past few decades. Although signal processing methods, developed within the framework of traditional Public Switched Telephone Network (PSTN), are relatively mature, developing a signal processing representation model, remains a challenging problem for the IoT and NNs.

When it comes to solving quantization challenge through innovation, one can agree that appropriate quantizer solution, can be used as a part of new model or new system. In parallel of looking for the new solutions, one could directly look for an optimized existing solution.

We have proposed apparently simple solution for the quasilogarithmic quantizer support region determination, on a way that is not very intricate in itself. Since, in general case, the support region is a very important part of quantizer parametrization task, its determination is desirable to be computationally not demanding, which has been gained with the proposal of this paper. Although outputting approximate values, suggested asymptotic formula has achieved quite accurate results after only one iteration.

Future network solutions will involve many more components or connected networks, while analyzing them requires a diverse set of perspectives. Numerous papers have demonstrated that by quantizing predominant network components, during and after training phase, more beneficial and efficient DNNs could be modeled. It should be noted that possible clipping values have been offline precalculated, and stored for fast retrieval.

No matter if the offered optimized solution can be recognized as the most important advantage, that offers easier adaptation into increasingly heterogeneous network structure, we have to seamlessly look for the simple and elegant solution.

Having in mind the significant overlapping in use of quantization techniques, for the numerous contemporary and novel network solutions, from different perspectives we encourage and vigorously motivate further quantizer parameters improvement.

Received on January 23, 2019

REFERENCES

1. Understanding the Internet of Things (IoT), GSMA association, white paper, *Available: https://www.gsma.com/iot/wp-content/uploads/2014/ 08/cl_iot_wp_07_14.pdf* (2014).
2. S. Chen, J. Zhao, *The Requirements, Challenges, and Technologies for 5G of Terrestrial Mobile Telecommunication*, IEEE Communications Magazine, **52**, *5*, pp. 36–43 (2014).
3. K. E. Skouby, P. Lynggaard, *Smart Home and Smart City Solutions Enabled by 5G, IoT, AAI and CoT Services*, in Int. Conference on Contemporary Computing and Informatics, pp. 874–878, Mysore, India, 2014.
4. GSMA Intelligence Understanding 5G: *Perspectives on Future Technological Advancements in Mobile*, White paper, pp. 6–13 (2014).
5. M. Agiwal, A. Roy, N. Saxena, *Towards Connected Living: 5G Enabled Internet of Things (IoT)*, IETE Technical Review, pp. 1–13 (2018).
6. O. Said, A. Elnashar, O. Elshakankiry, *Optimized Mechanism for Minimizing Data Overhead in IoT Environments: Design, Simulation and Performance Evaluation*, International Journal of Applied Engineering Research, **12** (2017).
7. L. Hanzo, C. A. Somerville, J. Woodard, *Voice and Audio Compression for Wireless Communications*, John Wiley & Sons - IEEE Press, pp/ 11– 28 (2007).
8. R. Banner, Y. Nahshan, E. Hoffer, D. Soudry, *ACIQ: Analytical Clipping for Integer Quantization of Neural Networks*, *arXiv:1810.05723* (2018).
9. Y. Guo, *A Survey on Methods and Theories of Quantized Neural Networks*, *arXiv:1808.04752* (2018).
10. E. Hoffer, R. Banner, I. Golan, D. Soundry, *Norm matters: efficient and accurate normalization schemes in deep networks*, arXiv preprint *arXiv:1803.01814* (2018).
11. S. Wu, G. Li, L. Deng, *et al.*, *L1-norm batch normalization for efficient training of deep neural networks*, arXiv preprint *arXiv:1802.09769* (2018).
12. N.S. Jayant, P. Noll, *Digital Coding of Waveforms*, Prentice Hall, New Jersey, 1984.
13. Z. Perić, J. Nikolić, D. Pokrajac, *Hybrid Scalar Quantizer for the Laplacian Source*,WSEAS Transactions on Communications, **6**, pp. 60–65 (2007).
14. Z. Perić, J. Lukić, J. Nikolić, D. Denić, *Support Design of Nonuniform dead-zone quantizer with low number of quantization levels for the Laplacian source,* Rev. Roum. Sci. Techn. – Électrotechn. et Énerg., **58**, *1*, pp. 93–100, (2013).
15. Z. Perić, M. Petković, J. Nikolić, S. Jovanović, *Support region estimation of the product polar companded quantizer for Gaussian source*, Signal Processing, **143**, pp. 140–145 (2018).
16. Z. Perić, S. Suzić, T. Delić, N. Simić, *Support region of Semilogarithmic Quantizer for Laplacian source*, Elektronika IR Elektrotechnika, **24**, *4* (2018).
17. M. Tančić, Z. Perić, N. Simić, *Modified wideband speech coding system with embedded G.711 coders*, Rev. Roum. Sci. Techn. – Électrotechn. et Énerg., **62**, *3*, pp. 299–304 (2017).

18. D. Aleksić, Z. Perić, J. Nikolić, *Support Region Determination of the Quasilogarithmic Quantizer for Laplacian Source*, Przeglad Elektrotechniczny, **88**, *7A*, pp. 130–132, 2012.

19. Z. Perić, M. Petković, M. Dinčić, *Simple Compression Algorithm for Memoryless Laplacian Source Based on the Optimal Companding Technique*, Informatica, **20**, *1*, pp. 99–114 (2009).