Performance Analysis of Two 8-Bit Floating-Point-based Piecewise Uniform Quantizers for a Laplacian Data Source

Jelena R. Nikolic[1,*], Zoran H. Peric[1], Aleksandra Z. Jovanovic[1], Stefan S. Tomic[2], Sofija Z. Peric[1]
[1]*Faculty of Electronic Engineering, University of Nis,*
Aleksandra Medvedeva 14, 18000 Nis, Serbia
[2]*Department of Electromechanical Engineering and Technology, Abu Dhabi Polytechnic,*
Al Nasr Street, MBZ Z23, Abu Dhabi, UAE
[*]*jelena.nikolic@elfak.ni.ac.rs; zoran.peric@elfak.ni.ac.rs; aleksandra.jovanovic@elfak.ni.ac.rs;*
stefan.tomic@actvet.gov.ae; sofija.p@elfak.rs

***Abstract*—In this paper, we employ the analogy between the representation of the floating-point (FP) format and the representation level distribution of the piecewise uniform quantizer (PWUQ) to assess the performance of FP-based solutions more thoroughly. We present theoretical derivations to assess the performance of the FP format and the PWUQ determined by this format for input data from the Laplacian source. We also provide a performance comparison of two selected 8-bit FP-based PWUQs. Beyond the typical evaluation of the applied FP format, through the accuracy degradation caused by the application of both FP8 solutions in neural network compression, we also use objective quantization measures. This approach offers insights into the robustness of these 8-bit FP-based solutions with respect to changes in input variance, which can be important when the input variance changes. The results demonstrate that the allocation of bits to encode the exponent and mantissa in the FP8 format is important, as it can significantly impact overall performance.**

***Index Terms*—Accuracy; Floating-point arithmetic; Neural network compression; Quantization.**

I. INTRODUCTION

Quantization is a technique with enormous applications in recent years due to numerous benefits it provides, especially in the compression of neural networks (NNs) [1], [2]. Quantization performs mapping 32-bit floating-point (FP32) values to a smaller set of discrete values, thus approximating the original 32-bit values with lower-bit values. Accordingly, quantization decreases the memory footprint and the costs of NN deployment [1], [3], [4]. The reduced precision of the representation of numerical values is the cornerstone of speeding up training and inference in NN [5]. However, these benefits often come at the expense of reduced accuracy of compressed neural networks (NNs). The choice of the quantizer model for the compression of an NN is very important, as using an aggressive low-bit quantization model may lead to significant accuracy degradation [2], [3], [6], [7].

Among the numerous quantization models for compression of FP32 values, researchers have focussed special attention on the simplest uniform quantization corresponding to the fixed-point format [1], [8], [9]. Compression of values represented in the FP32 format into fixed-point format values has certain advantages because some edge devices only support integer arithmetic [4]. However, as highlighted in [1] and [4], 8-bit quantization based on the fixed-point Int8 format can result in significant accuracy degradation. This is not the case with piecewise uniform quantization (PWUQ) corresponding to the FP8 (8-bit floating-point) format, as concluded by empirical evaluations of the effectiveness of two different FP8 formats in a wide range of tasks, models, and data sets [10], [11]. The reason for this is that many NNs have a nonuniform distribution of parameters [3], [4], which makes it intuitively obvious that a nonuniform FP8 representation could potentially outperform the uniform representation defined by the Int8 format.

As new hardware has recently been released by NVIDIA Corp. [12], supporting the two FP8 formats examined in [10] and [11], the current focus of researchers has shifted towards these formats (see, e.g., [4], [5]) and their application in both the training and post-training phases. In the case of post-training quantization [6], which we address in this paper, quantization is applied to the pre-trained multilayer perceptron (MLP) weights.

The rest of the paper is structured as follows. In Section II, we provide the research framework and list the contributions of the paper. In Section III, we describe the FP format. In Section IV, we exploit the analogy between the FP format and PWUQ and derive formulas for the evaluation of the performance of PWUQ for Laplacian data, which reflect the performance of the corresponding FP-based solutions. In Section V, we evaluate the performance of the two FP8 formats using an objective measure from quantization and accuracy degradation, and analyse the robustness of these 8-bit FP-based solutions to changes in input variance. Finally, in Section VI we conclude the paper.

Manuscript received 8 October, 2024; accepted 7 January, 2025.
This research was supported in part by the European Union, within the program HORIZON-WIDERA-2023-ACCESS-02, under Grant No. 101160293, and in part by the Ministry of Science, Technological Development and Innovation of the Republic of Serbia under Grant No. 451-03-65/2024-03/200102.

II. RESEARCH FRAMEWORK AND CONTRIBUTIONS

The effect of quantization error in compressed NNs is typically evaluated by assessing the accuracy degradation relative to the baseline NN model with weights in the FP32 format [2], [3], [6]. In this paper, we examine the potential of two FP8 formats from [12] by using not only the degradation of accuracy, but also the signal to quantization noise ratio (SQNR). Specifically, we estimate the error introduced in the representation of NN parameters by drawing an analogy between the distribution of FP values and the level distribution of PWUQ. Assuming a Laplacian probability density function (PDF) for the data to be quantized, which has been shown to effectively model weights in MLP [3], [13], we derive expressions for the mean-square error (MSE) distortion and the SQNR of the PWUQ. The number of PWUQ segments and the step sizes within each segment are determined by the allocation of bits to encode the exponent (e) and mantissa (m) in the FP format, as elaborated in Section IV. Our analysis focusses on two FP8 formats that differ in their bit allocations for the exponent and mantissa, with a particular emphasis on understanding how these bit allocations influence the performance of FP8 formats.

The contributions of this paper can be outlined as follows:

– A theoretical approach to evaluate error made in NN parameters representation using two FP8 formats is proposed;

– A comprehensive performance evaluation of two FP8-based solutions is provided, using both accuracy degradation and objective quantization measures;

– Our analysis examines whether the value of SQNR and the accuracy degradation are affected by the allocation of a given number of bits for encoding the exponent and mantissa;

– Our analysis also investigates the robustness of these two FP8-based solutions to changes in input variance, which can be crucial in layer-wise quantization [14], especially when data statistics change rapidly between layers [1].

III. FP FORMAT

The IEEE 754 standard defines a 32-bit single precision floating-point format (FP32) by which a real number y can be represented in binary form as in [15]–[18]

$$y = (sa_1a_2...a_eb_1b_2...b_m)_2, \tag{1}$$

where bit s is used to encode the sign, e = 8 bits ($a_1a_2\ldots a_e$) to encode the exponent E

$$E = (a_1a_2...a_e)_2 = \sum_{i=0}^{e-1} a_{e-i}2^i, \tag{2}$$

and m = 23 bits ($b_1b_2\ldots b_m$) to encode mantissa M

$$M = (b_1b_2...b_m)_2 = \sum_{i=0}^{m-1} b_{m-i}2^i. \tag{3}$$

Bit s is 0 for positive numbers and 1 for negative numbers.

In what follows, assume that we can use (1)–(3) to specify the FP format for the given e bits to encode the exponent E and m bits to encode the mantissa M. For practical implementation purposes, $1 + e + m$ should be the power of 2. From (2) and (3), we can conclude that exponent E takes integer values ranging from $E_{\min} = 0$ to $E_{\max} = 2^e$-1, while mantissa M takes integer values ranging from $M_{\min} = 0$ to $M_{\max} = 2^m$-1.

As with the FP32 format, where the decimal value of the exponent is biased by $b = 127$, for the arbitrary FP format, the exponent is also biased

$$E^* = E - b, \tag{4}$$

whereas the bias value is

$$b = \left(E_{\max} - 1\right)/2. \tag{5}$$

For the given value of e, the value of b determines from

$$b = 2^{e-1} - 1. \tag{6}$$

The biased exponent E^* then takes integer values ranging from $E^*_{\min}$ to $E^*_{\max}$, given by:

$$\begin{cases} E^*_{\min} = -b = 1 - 2^{e-1}, \\ E^*_{\max} = 2^e - 1 - b = 2^{e-1}, \end{cases} \tag{7}$$

where $E^*_{\min}$ is reserved for encoding 0, and $E^*_{\max}$ is reserved for encoding infinity and NaN (not a number) [15]. Therefore, integer values of E^* ranging from $E^*_{\min}$+1 to $E^*_{\max}$-1 are used to represent numbers.

According to the observed FP format, for the given values of e and m, the real number y determines from

$$y = (-1)^s 2^{E-b}\left(1 + M/2^m\right) = (-1)^s 2^{E^*}\left(1 + M/2^m\right). \tag{8}$$

This FP format, like the FP32 format, is symmetric about 0. For simplicity, we focus on the positive numbers represented in this FP format, as the symmetry around 0 ensures that the negative numbers can be treated similarly.

IV. PIECEWISE UNIFORM QUANTIZATION IN ACCORDANCE WITH FP FORMAT

A. Analogy between FP Format and Piecewise Uniform Quantization

For 2^e-2 values of the biased exponent, which take integer values in the range from $E^*_{\min}$+1 to $E^*_{\max}$-1, and for 2^m integer values of M from $M_{\min} = 0$ to $M_{\max} = 2^m$-1, we can calculate from (8) all real values of y represented by the observed FP format. These values are indeed piecewise uniformly distributed across the segments, such as the distribution of representation levels in PWUQ (Fig. 1). PWUQs have proven to be suitable for quantization of data having Laplacian PDF [19], [20].

The number of segments of our PWUQ in the positive part of the values, following the analogy between the FP format and the PWUQ, is then $S = 2^e$-2, and the number of representation levels in each segment is 2^m. The levels of PWUQ representation in the positive part of the values for the i^{th} segment (i = 1, 2, …, 2^e-2), can be determined from (8) as

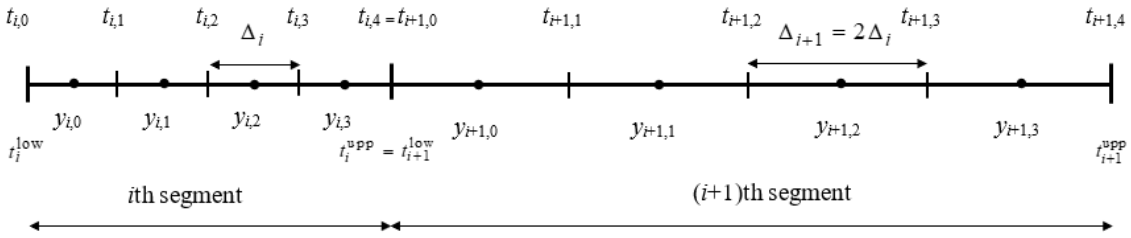

Fig. 1. Two consecutive segments of PWUQ for $m = 2$.

$$y_{i,j} = (-1)^0 2^{i-b}\left(1+ j/2^m\right) = \\ = 2^{i-b-m}\left(2^m + j\right),\ i = 1,2,...,S,\ j = 0,1,...,2^m - 1. \quad (9)$$

Due to the uniform distribution of representation levels and decision thresholds within each segment, we determine the uniform step size in the i^{th} segment from (9) as

$$\Delta_i = 2^{i-b-m},\ i = 1,2,...,S, \quad (10)$$

and decision thresholds from

$$\left\lfloor \begin{array}{l} t_{1,0} = 0 \\ t_{i,j+1} = y_{i,j} + \dfrac{\Delta_i}{2},\ i = 1,2,...,S,\ j = 0,1,...,2^m - 1 \end{array} \right\rfloor, \quad (11)$$

that is, from

$$\left\lfloor \begin{array}{l} t_{1,0} = 0 \\ t_{i,j} = 2^{i-b-m}\left(2^m + \dfrac{2j-1}{2}\right), i = 1,2,...,S,\ j = 1,2,...,2^m \end{array} \right\rfloor. \quad (12)$$

As it holds

$$\left\lfloor \begin{array}{l} t_{i+1,0} = t_{i+1}^{\text{low}} \\ t_{i,2^m} = t_i^{\text{upp}} \\ t_{i+1,0} = t_{i,2^m},\ i = 1,2,...,S-1 \end{array} \right\rfloor, \quad (13)$$

from (13), we can conclude that $t_i^{\text{upp}} = t_{i+1}^{\text{low}}$ (Fig. 1) and that the following holds for the boundaries of the i^{th} segment, denoted by $t_i^{\text{low}} = t_{i,0}$ and $t_i^{\text{upp}} = t_{i,2^m}$

$$\frac{t_i^{\text{upp}}}{t_i^{\text{low}}} = \frac{t_{i+1}^{\text{low}}}{t_i^{\text{low}}} = 2. \quad (14)$$

In other words, the boundaries of the segments follow a geometric progression with a common ratio of exactly 2. To assess the observed FP format and PWUQ, as specified in the following, we also specify the highest value of the representation level of PWUQ, i.e., the highest number that can be represented by the considered FP format. This value is calculated from (8) for $i = E_{\max}\text{-}1 = 2^e\text{-}2$ and $j = M_{\max} = 2^m\text{-}1$ as

$$y_{\max} = y_{E_{\max}-1,M_{\max}} = 2^{2^{(e-1)}}\left(1-2^{-(m+1)}\right) \approx 2^{2^{(e-1)}}. \quad (15)$$

For the calculated $y_{\max}$, from (11), we can determine $t_s^{\text{upp}} = t_{S,2^m} = y_{\max} + \Delta_S/2$. With this, we have determined the support region of PWUQ denoted with $\lfloor -t_s^{\text{upp}}, t_s^{\text{upp}} \rfloor$.

B. Derivation of Formulas for Performance Analysis of Piecewise Uniform Quantization for Laplacian Data Source

To estimate the error introduced by representing numbers in the considered FP format, we evaluate the performance of the corresponding PWUQ using objective measures from quantization, such as distortion and SQNR [21], [22]. The distortion of PWUQ in question, for given values of e and m, and the assumed PDF of the data to be quantized, $p(x, \sigma)$, is the sum of the granular mean squared error (MSE) distortion and the MSE overload distortion

$$D^{(1,e,m)}(\sigma) = D_g^{(1,e,m)}(\sigma) + D_o^{(1,e,m)}(\sigma), \quad (16)$$

respectively, specified by asymptotic formulas:

$$D_g^{(1,e,m)}(\sigma) = 2\sum_{i=1}^{S}\frac{\Delta_i^2}{12}P_i(\sigma), \quad (17)$$

$$D_o^{(1,e,m)}(\sigma) = 2\int_{y_{\max}}^{+\infty}(x - y_{\max})^2\, p(x,\sigma)dx. \quad (18)$$

Asymptotic formulas, such as those given in (17) and (18), have been shown to be highly accurate for the high-resolution case [21], [22], which is the case we will analyse in the following. In (17) and (18), 2 arises from the symmetry of the PWUQ and the corresponding FP format, S denotes the number of segments in the positive part of the values, ($S = 2^e\text{-}2$), while Δ_i and $y_{\max}$ are given by (10) and (15), respectively. $P_i(\sigma)$ denotes the probability that the input sample x, with variance σ^2 and PDF, $p(x, \sigma)$, belongs to the i^{th} segment denoted with $\lfloor t_i^{\text{low}}, t_i^{\text{upp}} \rfloor$

$$P_i(\sigma) = \int_{t_i^{\text{low}}}^{t_i^{\text{upp}}} p(x,\sigma)dx. \quad (19)$$

Since our research was driven by the initiative to analyse the performance of two FP8 formats from [12] applied in post-training quantization, we assume here the Laplacian PDF

$$p(x,\sigma) = \frac{1}{\sqrt{2}\sigma}\exp\left\{-\frac{\sqrt{2}\,|x|}{\sigma}\right\}, \quad (20)$$

which models well the PDF of weights in neural networks [3]. By substituting (20) into (19), we obtain

$$P_i(\sigma)=\frac{1}{2}\left(\exp\left\{-\frac{\sqrt{2}t_i^{\text{low}}}{\sigma}\right\}-\exp\left\{-\frac{\sqrt{2}t_i^{\text{upp}}}{\sigma}\right\}\right), \quad (21)$$

which is further transformed by (14) into

$$P_i(\sigma)=\exp\left\{-\frac{3\sqrt{2}t_i^{\text{low}}}{2\sigma}\right\}\sinh\left(\frac{\sqrt{2}t_i^{\text{low}}}{2\sigma}\right)=$$

$$=\exp\left\{-3\times\left(1-\frac{1}{2^{m+1}}\right)\frac{2^{i-b-\frac{1}{2}}}{\sigma}\right\}\sinh\left(\left(1-\frac{1}{2^{m+1}}\right)\frac{2^{i-b-\frac{1}{2}}}{\sigma}\right). \quad (22)$$

Using (15)–(18), (20) and (22), and assuming as in (15) that it holds $1-2^{-(m+1)}\approx 1$, we determine the MSE distortion

$$D^{(1,e,m)}(\sigma)\approx\sigma^2\exp\left\{-\frac{2^{2^{(e-1)}+\frac{1}{2}}}{\sigma}\right\}+$$

$$+\sum_{k=1-b}^{S-b}\frac{2^{2(k-m)}}{6}\exp\left\{-3\times\frac{2^{k-\frac{1}{2}}}{\sigma}\right\}\sinh\left(\frac{2^{k-\frac{1}{2}}}{\sigma}\right). \quad (23)$$

Further substituting (23) into the expression of SQNR [21]

$$\text{SQNR}^{(1,e,m)}[\text{dB}]=10\log_{10}(\sigma^2/D^{(1,e,m)}(\sigma)), \quad (24)$$

where the variance of the signal to be quantized is σ^2, and in dB is defined by:

$$v[\text{dB}]=10\log_{10}(\sigma^2/\sigma_{\text{ref}}^2), \quad (25)$$

$$\sigma=\sigma_{\text{ref}}\,10^{v/20}, \quad (26)$$

we can calculate $\text{SQNR}^{(1,e,m)}$ [dB] for $\sigma^2_{\text{ref}}=1$ (a common choice for reference variance, σ^2_{ref}, is the unit variance).

C. Two Different Solutions of Piecewise Uniform Quantizers Specified in Accordance with the FP8 Format

Let us assume the FP8 format for the triple of numbers (1, 5, 2) from [12], whereas 1 bit is used to encode the sign, e = 5 bits and m = 2 bits are used to encode the exponent and mantissa, respectively. Given that $S = 2^e-2 = 30$, for the first PWUQ solution specified according to FP8, by substituting the triple of numbers (1, 5, 2) into (6), (9), (10), and (12), we determine the value of bias

$$b^{(1,5,2)}=2^{5-1}-1=15, \quad (27)$$

the uniform step size in the i^{th} segment

$$\Delta_i^{(1,5,2)}=2^{i-15-2}=2^{i-17},\ i=1,2,...,30, \quad (28)$$

the representation levels in the positive part of values for the i^{th} segment

$$y_{i,j}^{(1,5,2)}=2^{i-17}\left(4+j\right),\ i=1,2,...,30,\ j=0,1,2,3, \quad (29)$$

the decision thresholds of the i^{th} segment

$$\left[\begin{array}{l} t_{1,0}^{(1,5,2)}=0 \\ t_{i,j}^{(1,5,2)}=2^{i-17}\left(4+\frac{2j-1}{2}\right),\ i=1,2,...,30,\ j=1,2,3,4 \end{array}\right]. \quad (30)$$

Applying (23) for the triple of numbers (1, 5, 2), we obtain

$$D^{(1,5,2)}(\sigma)\approx\sigma^2\exp\left\{-2^{\frac{33}{2}}/\sigma\right\}+$$

$$+\sum_{k=-14}^{15}\frac{2^{2k}}{96}\exp\left\{-3\times2^{k-\frac{1}{2}}/\sigma\right\}\sinh\left(2^{k-\frac{1}{2}}/\sigma\right). \quad (31)$$

Let us further assume the FP8 format for the second triple of numbers (1, 4, 3) from [12], where 1 bit is used to encode the sign, e = 4 bits and m = 3 bits are used to encode the exponent and mantissa, respectively. Given that $S = 2^e - 2 = 14$, for the second solution of PWUQ specified according to FP8, by replacing the triple of numbers (1, 4, 3) in (6), (9), (10), and (12), we determine the bias value

$$b^{(1,4,3)}=2^{4-1}-1=7, \quad (32)$$

the uniform step size in the i^{th} segment

$$\Delta_i^{(1,4,3)}=2^{i-7-3}=2^{i-10},\ i=1,2,...,14, \quad (33)$$

the representation levels in the positive part of values for the i^{th} segment

$$y_{i,j}^{(1,4,3)}=2^{i-10}\left(8+j\right),i=1,2,...,14,\ j=0,1,...,7, \quad (34)$$

the decision thresholds and the boundaries of the i^{th} segment

$$\left[\begin{array}{l} t_{1,0}^{(1,4,3)}=0 \\ t_{i,j}^{(1,4,3)}=2^{i-10}\left(8+\frac{2j-1}{2}\right),i=1,2,...,14,\ j=1,2,...,8 \end{array}\right]. \quad (35)$$

As in the previous solution, by replacing the specified triple (1, 4, 3) in (23), we obtain the following

$$D^{(1,4,3)}(\sigma)\approx\sigma^2\exp\left\{-2^{\frac{17}{2}}/\sigma\right\}+$$

$$+\sum_{k=-6}^{7}\frac{2^{2k}}{384}\exp\left\{-3\times2^{k-\frac{1}{2}}/\sigma\right\}\sinh\left(2^{k-\frac{1}{2}}/\sigma\right). \quad (36)$$

Eventually, for both FP8 solutions, applying (24)–(26), (31) and (36), for the variance given of the signal to be quantized, σ^2, we determine the values of $\text{SQNR}^{(1,e,m)}$ [dB].

V. NUMERICAL RESULTS

To evaluate the performance of the selected FP8 formats, we use the analogy with PWUQ, which enables us to determine the MSE distortion introduced by this format in the representation of the data. Since the Laplacian PDF models a large number of phenomena, such as the weight distribution in neural networks [3], [13], we assume that the data have the

Laplacian PDF. In other words, we evaluate the performance of the FP8 format by analysing the logarithmic ratio of the variance of the input data, σ^2, and the MSE introduced by this format, i.e., by determining the SQNR of the corresponding PWUQ from (23) and (24). Since the distortions of PWUQs corresponding to the formats FP8 with parameters $e = 5$, $m = 2$, and $e = 4$, $m = 3$ are given by (31) and (36), respectively, we substitute (31) and (36) into (24) for different variance values to determine the SQNR values for the observed formats FP8 (1, 5, 2) and FP8 (1, 4, 3). Our results are shown in Table I and Fig. 2. We can conclude that both analysed FP8 formats are robust to variance changes, with the FP8 (1, 4, 3) format achieving a higher SQNR. The robustness of the considered 8-bit FP formats results from the fact that the uniform step size per segment changes according to the geometric progression as follows: Δ_1, $2\Delta_1$, $2^2\Delta_1$, $2^3\Delta_1$, …, $2^{S-1}\Delta_1$. This is also a characteristic of robust piecewise linear companding quantizers used in speech signal coding [19].

TABLE I. SQNR FOR INT8 [8] AND TWO FP8 FORMATS (FP8 (1, 5, 2) AND FP8 (1, 4, 3)).

σ^2	SQNR		
	INT8	**FP8 (1, 5, 2)**	**FP8 (1, 4, 3)**
0,1	0.7918	24.9400	31.2444
0,3	5.5630	24.9402	31.2455
0,5	7.7815	24.9402	31.2454
1	10.7918	24.9405	31.2460
3	15.5630	24.9408	31.2461
5	17.7815	24.9402	31.2457
10	20.7918	24.9406	31.2459

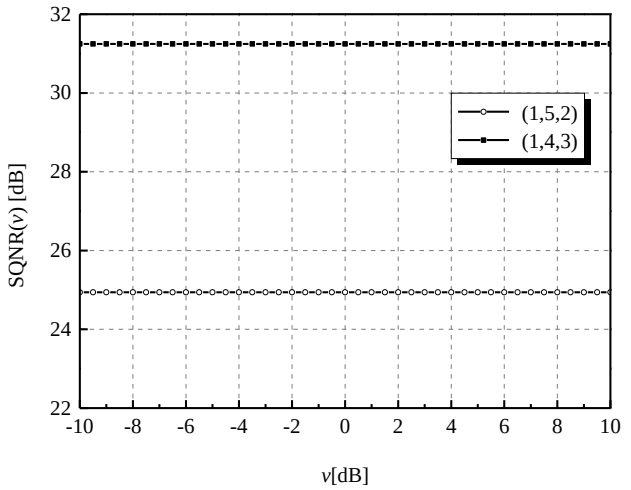

Fig. 2. Dependence of SQNR on v [dB] for two FP8 solutions.

With the FP8 (1, 5, 2) format, an SQNR of approximately 24.94 dB is achieved throughout the observed variance range, while with the FP8 (1, 4, 3) format, an SQNR of approximately 31.24 dB is achieved in the same variance range. These results also demonstrate that the value of SQNR is influenced by the way the allocation of a given number of bits is performed to encode for the exponent and mantissa. When one bit less is used for coding the exponent, while the total number of bits remains 8, we observe a higher SQNR value for 6.3 dB. This is because the number of bits defined to encode the exponent e dominantly determines the highest value that can be represented by the FP format (see (15)), as well as the uniform step size, Δ_i, i.e., the resolution of the format. The higher the value of e, the higher the values that can be represented by this format. Consequently, for a given total number of bits (8 bits in the FP8 format), the higher the value of e, the lower the resolution of the format, meaning the larger the uniform step size. Specifically, for $e = 4$ from (15), we calculate $y_{\max} = 240$, whereas for $e = 5$, $y_{\max} = 57344$. For the range of variances, we observe in dB scale v[dB] $\in$ [−10 dB, 10 dB], and the Laplacian PDF (see Fig. 3), the values to be represented in the FP format are much smaller than $y_{\max} = 57344$. Therefore, these values are better represented in the format with the smaller $y_{\max}$ but with higher resolution. This choice of $e = 4$ results in less error when presenting the data and achieves a higher SQNR.

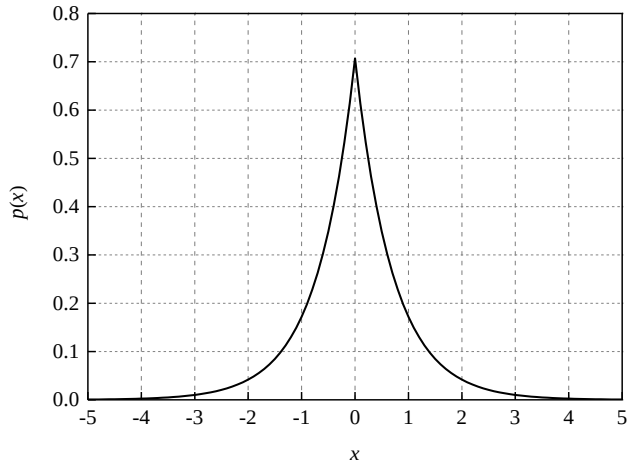

Fig. 3. Laplacian PDF of zero mean and unit variance.

Finally, let us compare the FP8 format with the int8 format corresponding to uniform quantization [8]. To this end, let us determine the MSE distortion and SQNR of the int8 format given by (17), (18) and (24) for $\Delta_i = \Delta = 1$ as follows:

$$D^{\text{int8}}(\sigma) = \frac{\Delta^2}{12}\left(1 - \exp\left\{-\frac{\sqrt{2}}{\sigma}x_{\max}\right\}\right) + \sigma^2 \exp\left\{-\frac{\sqrt{2}}{\sigma}x_{\max}\right\}, \quad (37)$$

$$\text{SQNR}^{\text{int8}}[\text{dB}] = 10\log_{10}\left(\sigma^2 / D^{\text{int8}}(\sigma)\right), \quad (38)$$

where $x_{\max} = 127$ is the highest value that can be represented in int8 format. From Table I, we can conclude that the considered FP8 formats provide significantly higher robustness to variance changes, and achieve a substantially higher SQNR compared to the int8 format.

In addition to evaluating the performance of the two FP8 formats using SQNR, we assess their performance based on the accuracy degradation caused by representing the FP32 weight values in the pretrained MLP using the FP8 formats considered. We use the pre-trained MLP model from [14] to ensure that the same weights are subjected to quantization. Briefly, the MLP model consists of three fully-connected layers and has been pre-trained on the MNIST data set. The accuracy of the pre-trained model, referred to as the baseline due to its use of FP32 weights, is 98.1 %. Unlike the work in [14], where uniform quantization was applied to weights that were normalised before quantization, in this paper, we deal with PWUQ and do not perform weight normalisation because the robustness of the PWUQ model allows it. Applying the first PWUQ model determined by the FP8 (1, 5, 2) format, we determined that the accuracy amounts to 97.76 %. The accuracy of the MLP with the second applied PWUQ model, corresponding to the FP8 (1, 4, 3) format, is

slightly higher and amounts to 97.93 %. Therefore, the accuracy degradation compared to the baseline model is 0.34 % and 0.17 %, respectively. Consistent with the SQNR-based performance evaluation, FP8 (1, 4, 3) outperformed FP8 (1, 5, 2), resulting in less accuracy degradation. Furthermore, both FP8 models significantly outperformed the int8 model, since the accuracy of the MLP with the applied int8 model amounts to 9.8 %. As NVIDIA Corp. [12] recently launched new hardware supporting the two FP8 formats analysed in detail in this paper, we believe that our theoretical framework presented for arbitrary FP formats could significantly contribute to the development and practical application of emerging FP formats.

VI. CONCLUSIONS

This paper utilised the analogy between the FP format and piecewise uniform quantization to comprehensively evaluate the performance of FP-based solutions, using both accuracy degradation and SQNR as metrics. The results have shown that two 8-bit FP-based solutions can indeed have different performances when different numbers of bits are allocated for encoding the exponent and the mantissa. In addition, for the two selected 8-bit FP-based PWUQ models, we have shown that similar robustness to variance changes is achieved by both models, whereas higher SQNR values are achieved by assigning more bits for encoding the mantissa, i.e., fewer bits for encoding the exponent. We have concluded that for the wide range of variances observed in dB scale v[dB] $\in$ [−10 dB, 10 dB] and Laplacian PDF, it is better to present the values in FP8 format with a smaller $y_{\max}$ value, but a higher resolution. This choice results in less error when presenting the data and provides a higher SQNR and a smaller accuracy degradation of the pre-trained MLP model. This observation highlights a potential practical application of our results: estimating and comparing SQNRs across different FP formats can guide the selection of FP parameters (e and m) to represent NN parameters when these parameters adhere to a Laplacian probability density function. Future work may explore extending these insights to other FP formats and broader data distributions, further validating the generality and applicability of the proposed approach.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest.

REFERENCES

[1] S. Kim, A. Gholami, Z. Yao, M. W. Mahoney, and K. Keutzer, "I-BERT: Integer-only BERT quantization", in *Proc. of the 38th International Conference on Machine Learning (PMLR)*, 2021, pp. 5506–5518, vol. 139.

[2] B. Zhang, T. Wang, S. Xu, and D. Doermann, *Neural Networks with Model Compression*. Singapore, CA: Springer Singapore, 2024. DOI: 10.1007/978-981-99-5068-3.

[3] R. Banner, Y. Nahshan, and D. Soudry, "Post-training 4-bit quantization of convolutional networks for rapid-deployment", in *Proc. of the 33rd International Conference on Neural Information Processing Systems* (*NeurIPS 2019*), 2019, art. no. 714, pp. 7950–7958.

[4] J. Li, T. Zhang, I. E.-H. Yen, and D. Xu, "FP8-BERT: Post-training quantization for transformer", 2023. DOI: 10.48550/arXiv.2312.05725.

[5] P. Micikevicius *et al.*, "FP8 formats for deep learning", 2022. DOI: 10.48550/arXiv.2209.05433.

[6] L. Zhao, Z. Dong, and K. Keutzer, "Analysis of quantization on MLP-based vision models", 2022. DOI: 10.48550/arXiv.2209.06383.

[7] K. Yuan *et al.*, "Fully-fused multi-layer perceptrons on Intel Data Center GPUs", 2024. DOI: 10.48550/arXiv.2403.17607.

[8] T. Dettmers, M. Lewis, Y. Belkada, and L. Zettlemoyer, "LLM.int8(): 8-bit matrix multiplication for transformers at scale", in *Proc. of the 36th Conference on Advances in Neural Information Processing Systems*, 2022, pp. 30318–30332.

[9] P. Peng, M. You, W. Xu, and J. Li, "Fully integer-based quantization for mobile convolutional neural network inference", *Neurocomputing*, vol. 432, pp. 194–205, 2021. DOI: 10.1016/j.neucom.2020.12.035.

[10] X. Sun *et al.*, "Hybrid 8-bit floating point (HFP8) training and inference for deep neural networks", in *Proc. of 33rd Conference on Neural Information Processing Systems* (*NeurIPS 2019*), 2019, pp. 4900–4909.

[11] N. Wang, J. Choi, D. Brand, C.-Y. Chen, and K. Gopalakrishnan, "Training deep neural networks with 8-bit floating point numbers", in *Proc. of 32nd Conference on Neural Information Processing Systems* (*NeurIPS 2018*), 2018, pp. 7686–7695.

[12] "NVIDIA H100 tensor core GPU architecture overview", NVIDIA Corp., 2023. [Online]. Available: https://resources.nvidia.com/en-us-tensor-core

[13] Z. H. Peric, B. D. Denic, M. S. Savic, N. J. Vucic, and N. B. Simic, "Binary quantization analysis of neural networks weights on MNIST dataset", *Elektronika ir Elektrotechnika*, vol. 27, no. 4, pp. 55–61, 2021. DOI: 10.5755/j02.eie.28881.

[14] S. Tomić, J. Nikolić. Z. Perić, and D. Aleksić, "Performance of post-training two-bits uniform and layer-wise uniform quantization for MNIST dataset from the perspective of support region choice", *Mathematical Problems in Engineering*, vol. 2022, art. ID 1463094, 2022. DOI: 10.1155/2022/1463094.

[15] M. Dinčić, Z. Perić, M. Savić, M. Milojković, and N. Vučić, "SQNR analysis and classification accuracy of the 24-bit floating point representation of the Laplacian data Source applied for quantization of weights of a multilayer perceptron", in *Proc. of XV International Conference on Systems, Automatic Control and Measurements (SAUM)*, 2021.

[16] *IEEE Standard for Binary Floating-Point Arithmetic*, ANSI/IEEE Standard 754-1985, pp. 1–20, 1985.

[17] *IEEE Standard for Floating-Point Arithmetic*, IEEE Standard 754-2008, pp. 1–70, 2008.

[18] *IEEE Standard for Floating-Point Arithmetic*, IEEE Standard 754-2019 (Revision of IEEE 754-2008), pp. 1–84, 2019.

[19] J. Nikolić, Z. Perić, A. Jovanović, and D. Antić, "Design of forward adaptive piecewise uniform scalar quantizer with optimized reproduction level distribution per segments", *Elektronika ir Elektrotechnika*, vol. 119, no. 3, pp. 19–22, 2012. DOI: 10.5755/j01.eee.119.3.1356.

[20] J. Nikolić, D. Aleksić, Z. Perić, and M. Dinčić, "Iterative algorithm for parameterization of two-region piecewise uniform quantizer for the Laplacian source", *Mathematics*, vol. 9, no. 23, p. 3091, 2021. DOI: 10.3390/math9233091.

[21] S. Na, "Asymptotic formulas for mismatched fixed-rate minimum MSE Laplacian quantizers", *IEEE Signal Processing Letters*, vol. 15, pp. 13–16, 2008. DOI: 10.1109/LSP.2007.910240.

[22] S. Na and D. L. Neuhoff, "Asymptotic MSE distortion of mismatched uniform scalar quantization", *IEEE Transactions on Information Theory*, vol. 58, no. 5, pp. 3169–3181, 2012. DOI: 10.1109/TIT.2011.2179843.