SinReQ: Generalized Sinusoidal Regularization for Low-Bitwidth Deep Quantized Training

Ahmed T. Elthakeb 1 Prannoy Pilligundla 1 Hadi Esmaeilzadeh 1

Abstract
Deep quantization of neural networks (below eight bits) offers significant promise in reducing their compute and storage cost. Albeit alluring, without special techniques for training and optimization, deep quantization results in significant accuracy loss. To further mitigate this loss, we propose a novel sinusoidal regularization, called SinReQ, for deep quantized training. SinReQ adds a periodic term to the original objective function of the underlying training algorithm. SinReQ exploits the periodicity, differentiability, and the desired convexity profile in sinusoidal functions to automatically propel weights towards values that are inherently closer to quantization levels. Since, this technique does not require invasive changes to the training procedure, SinReQ can harmoniously enhance quantized training algorithms. SinReQ offers generality and flexibility as it is not limited to a certain bitwidth or a uniform assignment of bitwidths across layers. We carry out experimentation using the CIFAR-10, ResNet-20, SVHN DNNs with three to five bits for quantization and show the versatility of SinReQ in enhancing multiple quantized training algorithms, DoReFa (Zhou et al., 2016) and WRPN (Mishra et al., 2018). Averaging across all the bit configurations shows that SinReQ closes the accuracy gap between these two techniques and the full-precision runs by 35.7% and 37.1%, respectively. That is improving the absolute accuracy of DoReFa and WRPN up to 5.3% and 2.6%, respectively.

1. Introduction
Despite the success of DNNs in various domains (LeCun et al., 2015; Hauswald et al., 2015; Krizhevsky et al., 2012), their compute efficiency hinders effective deployment in resource-limited platforms (Sze et al., 2017). Quantization, in general, and deep quantization, in particular, aim to not only reduce the compute requirements of DNNs but also significantly reduce their memory footprint (Zhou et al., 2016; Mishra et al., 2018; Sharma et al., 2018; Judd et al., 2016). Nevertheless, without specialized training and optimization training algorithms, quantization can diminish the accuracy. As such, several techniques have been proposed that aim to train DNNs in quantized mode with as low as possible loss in accuracy (Courbariaux et al., 2015; Gupta et al., 2015; Hubara et al., 2017a; Zhou et al., 2017). However, eliminating the loss has proven to be illusive.

This paper aims to provide a new mechanism that enhances these techniques and significantly closes the remaining gap between deeply quantized and full precision networks. As such, we propose a sinusoidal regularization technique, a differentiable loss, that naturally pushes the weight values toward the quantization levels exploiting the inherent periodicity of sinusoidal functions. As such, quantized training algorithms (Zhou et al., 2016; Mishra et al., 2018) that still use some form of backpropagation (Rumelhart et al., 1986) can effectively utilize the proposed mechanism to further enhance their performance in accuracy recovery. SinReQ offers generality and can be used with different bitwidths by setting the periodicity of the regularizer according to the desired bitwidth. Moreover, the proposed technique is flexible and a dedicated sinusoidal term for each layer with different periods can enable heterogenous quantization across the layers. The SinReQ regularization can also be applied for training a model from scratch, or for fine-tuning a pretrained model. We evaluate SinReQ using different bitwidth assignments across for different DNNs (CIFAR-10, ResNet-20, and SVHN). To show the versatility of SinReQ, it is used with two different quantized training algorithms, DoReFa (Zhou et al., 2016) and WRPN (Mishra et al., 2018). Over all the bitwidth assignments, the proposed regularization, on average, improves the top-1 accuracy of DoReFa and WRPN by 2.8% and 2.1%, respectively. That is, closing the gap between the quantized network and a full-precision network by 37.1% in the case of DoReFa and 36.5% in the case of WRPN.
2. Related Work

SinReQ, which is a regularization technique, is complementary to the previously proposed quantized training (Courbariaux et al., 2015; Gupta et al., 2015; Hubara et al., 2017a; Zhou et al., 2017) and binarization (Hubara et al., 2016; Rastegari et al., 2016) algorithms and can potentially augment their training procedure. Additionally, there is a line of research that aims to incorporate the distance between the quantized levels and the full-precision weights in the training loss (Jung et al., 2018; Hou et al., 2017; Hou & Kwok, 2018; Choi et al., 2018b; Zhang et al., 2018). In contrast, SinReQ utilizes the periodic nature of sinusoidal function to push the weight values to the quantization levels possible by the allocated bitwidth to each layer.

Training algorithms for quantized neural networks. There have been several techniques (Zhou et al., 2016; Zhu et al., 2017b; Mishra et al., 2018) that train a neural network in a quantized domain after the bitwidth of the layers is determined manually. DoReFa quantizes weights, activations and gradients of neural networks using different bitwidths. They suggest maintaining a high-precision floating point copy of the weights while feeding quantized weights into backprop. WRPN introduces a scheme to train networks from scratch using reduced-precision activations by decreasing the precision of both activations and weights and increasing the number of filter maps in a layer. (Zhu et al., 2017b) performs the training phase of the network in full precision, but for inference uses ternary weight assignments. For this assignment, the weights are quantized using two scaling factors which are learned during training phase. SinReQ is a complimentary method that can potentially enhance these algorithms. The paper demonstrates this feature concretely in the context of DoReFa and WRPN training algorithms.

Binarized and ternarized neural networks. Extensive work, (Hubara et al., 2017b; Rastegari et al., 2016; Li & Liu, 2016) focus on binarized neural networks, which impose accuracy loss but reduce the bitwidth to lowest possible level. In BinaryNet (Hubara et al., 2016), an extreme case, a method is proposed for training binarized neural networks which reduce memory size, accesses and computation intensity at the cost of accuracy. XNOR-Net (Rastegari et al., 2016) leverages binary operations (such as XNOR) to approximate convolution in binarized neural networks. On the other side, in a weight ternarized network, zero is used as an additional quantized value. (Li & Liu, 2016) introduces ternary-weight networks, in which the weights are quantized to -1, 0, +1 values by minimizing the Euclidian distance between full-precision weights and their ternary assigned values. In (Zhu et al., 2017a) different scaling factors are introduced to the ternarized weights. The scaling parameters are learned by gradient descent. None of these techniques propose sinusoidal regularization to make the weight values more quantization friendly as training progresses.

Loss-aware weight quantization. Recent works pursued loss-aware minimization approaches for quantization. (Hou et al., 2017; Hou & Kwok, 2018) developed approximate solutions using proximal Newton algorithm to minimize the loss function directly under the constraints of low bitwidth weights. (Choi et al., 2018b) proposed to learn the quantization of DNNs through regularization by introducing a learnable regularization coefficient to find low bitwidth models efficiently in training. (Zhang et al., 2018) proposed an adaptive technique to jointly train a quantized, bit-operation-compatible DNN and its associated quantizers, as opposed to using fixed, handcrafted quantization schemes such as uniform or logarithmic quantization. Although these techniques use regularization to guide the process of quantized training, they don not explore the use of periodic differentiable trigonometric functions.

3. Sinusoidal Regularization for Automatic Quantization during Training

Our proposed method SinReQ exploits weight regularization in order to automatically quantize a neural network while training. To that end, Sections 3.1 to 3.2 describe the role of regularization in neural networks and then Section 3.3 explains SinReQ in more detail.

3.1. Loss Landscape of Neural Networks

Neural networks’ loss Landscapes are known to be highly non-convex and generally very poorly understood. It has been empirically verified that loss surfaces for large neural networks have many local minima that are essentially equivalent in terms of test error (Choromanska et al., 2015). (Li et al., 2018). Moreover, converging to one of the many good local minima proves to be more useful as compared to struggling to find the global minimum of the accuracy loss on the training set (which often leads to overfitting). This opens up and encourages a possibility of adding extra custom objectives to optimize for during the training process, in addition to the original objective (i.e., to minimize the accuracy loss). The added custom objective could be with the purpose of increasing generalization performance or imposing some preference on the weights values. Regularization is one of the major techniques that makes use of such facts as discussed in the following subsection.

3.2. Regularization in Neural Networks

Neural networks often suffer from redundancy of parameterization and consequently they commonly tend to overfit. Regularization is one of the commonly used techniques to enhance generalization performance of neural networks. Regularization effectively constrains weight parameters by
adding a term (regularizer) to the objective function that captures the desired constraint in a soft way. This is achieved by imposing some sort of preference on weight updates during the optimization process. As a result, regularization seamlessly leads to unconditionally constrained optimization problem instead of explicitly constrained which, in most cases, is much more difficult to solve.

**Conventional regularization: weight decay.** The most commonly used regularization technique is known as weight decay, which aims to reduce the network complexity by limiting the growth of the weights. It is realized by adding a term to the objective function that penalizes large weight values

\[
E(w) = E_o(w) + \frac{1}{2} \lambda \sum_i w_i^2 \tag{1}
\]

where \(E_o\) is the original loss measure, and \(\lambda\) is a parameter governing how strongly large weights are penalized. \(w_i\) is a vector of all weights for layer \(i\) of the network while summation of \(i\) is over all the layers in the network.

**Periodic regularization: SinReQ.** In this work, we propose a new type of regularization that is friendly to quantization. The proposed regularization is based on a periodic function (sinusoidal) that provides a smooth and differentiable loss to the original objective, Figure 1 (a). The periodic regularizer has a periodic pattern of minima that correspond to the desired quantization levels. Such correspondence is achieved by matching the period to the quantization step based on a particular number of bits for a given layer.

\[
E(w) = E_o(w) + \frac{1}{2} \lambda \sum_i w_i^2 + \frac{1}{2} \lambda q \sum_i \sin^2\left(\frac{\pi w_i}{2^{q\text{bits}} - 1}\right) \tag{2}
\]

where \(E_o\) is the original loss measure, and \(\lambda_q\) is SinReQ regularization strength that is a parameter governing how strongly weight quantization errors are penalized. For the sake of simplicity and clarity, Figure 1 (b) and (c) depict a geometrical sketch for a hypothetical loss surface (original objective function to be minimized) and an extra regularization term in 2-D weight space for weight decay and SinReQ respectively. \(w_{opt}\) is the optimal point just for the loss function alone.

**Algorithm 1 SinReQ implementation on LeNet**

1: \(q\text{bits} \leftarrow \text{number of quantization bits; } q\text{bits} \in \{1,2,3,...\}\)
2: \(\lambda_q \leftarrow \text{regularization strength}\)
   ▷ Set the quantization step based on the used quantization technique
   ▷ for DoReFa quantization
3: \(\text{step} \leftarrow 1/(2^{\text{qbits}} - 0.5), \Delta \leftarrow \text{step}/2\)
   ▷ for WRPN quantization
4: \(\text{step} \leftarrow 1/(2^{\text{qbits}} - 1.0), \Delta \leftarrow 0\)
   ▷ For each layer in the network, calculate the sinreq loss
   ▷ Layer conv1
5: \(\text{kernel} \leftarrow \text{conv1.float}\_weight\)
6: \(\text{sinreq}_\text{conv1} \leftarrow \text{reduce}\_\text{mean}(\sin^2(\pi \times (\text{kernel} + \Delta))/\text{step})\)
   ▷ Layer conv2
7: \(\text{kernel} \leftarrow \text{conv2.float}\_weight\)
8: \(\text{sinreq}_\text{conv2} \leftarrow \text{reduce}\_\text{mean}(\sin^2(\pi \times (\text{kernel} + \Delta))/\text{step})\)
   ▷ Layer fcl
9: \(\text{kernel} \leftarrow \text{fc1.float}\_weight\)
10: \(\text{sinreq}_\text{fc1} \leftarrow \text{reduce}\_\text{mean}(\sin^2(\pi \times (\text{kernel} + \Delta))/\text{step})\)
   ▷ Layer fc2
11: \(\text{kernel} \leftarrow \text{fc2.float}\_weight\)
12: \(\text{sinreq}_\text{fc2} \leftarrow \text{reduce}\_\text{mean}(\sin^2(\pi \times (\text{kernel} + \Delta))/\text{step})\)
   ▷ Sum over all layers
13: \(\text{sinreq}\_\text{loss} = \text{sinreq}_\text{conv1} + \text{sinreq}_\text{conv2} + \text{sinreq}_\text{fc1} + \text{sinreq}_\text{fc2}\)
   ▷ Calculate the overall loss
14: \(\text{LOSS} = \text{original}\_\text{loss} + \lambda_q \times \text{sinreq}\_\text{loss}\)

function to be minimized) and an extra regularization term in 2-D weight space. For weight decay regularization, in Figure 1 (b), the faded circular contours show that as we get closer to the origin, the regularization loss is minimized. \(w_{opt}\) is the optimum just for the loss function alone and the overall optimum solution is achieved by striking a balance between the original loss term and the regularization loss term.

In a similar vein, Figure 1 (c) shows a representation of the proposed regularization. A periodic pattern of minima pockets are seen surrounding the original optimum point. The objective of the optimization problem is to find the...
Table 1. Summary of results comparing state-of-the-art methods DoReFa, and WRPN with and without SinReQ for different neural networks

<table>
<thead>
<tr>
<th>Network</th>
<th>Weights Bitwidth</th>
<th>Top1 Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>DoReFa</td>
</tr>
<tr>
<td>CIFAR10</td>
<td>3 bits</td>
<td>58.80</td>
</tr>
<tr>
<td></td>
<td>4 bits</td>
<td>65.54</td>
</tr>
<tr>
<td></td>
<td>5 bits</td>
<td>73.12</td>
</tr>
<tr>
<td>SVHN</td>
<td>3 bits</td>
<td>88.40</td>
</tr>
<tr>
<td></td>
<td>4 bits</td>
<td>93.81</td>
</tr>
<tr>
<td></td>
<td>5 bits</td>
<td>96.76</td>
</tr>
<tr>
<td>ResNet-20 on CIFAR10</td>
<td>3 bits</td>
<td>87.08</td>
</tr>
<tr>
<td></td>
<td>4 bits</td>
<td>90.16</td>
</tr>
<tr>
<td></td>
<td>5 bits</td>
<td>90.58</td>
</tr>
</tbody>
</table>

**Experimental Setup.** We implemented our technique inside Distiller (Zmora et al., 2018), an open source framework for compression by Intel Nervana. The reported accuracies for DoReFa and WRPN are with the built-in implementations in Distiller, which may not exactly match the accuracies reported in their respective papers. However, an independent implementation from a major company provides an unbiased foundation for the comparisons.

**Semi-quantized weight distributions.** Figure 2 shows the evolution of weights distributions over fine-tuning epochs for different layers of (a) CIFAR10 and (b) SVHN networks at different bitwidths (3, 4, and 5 bits). The high-precision weights form clusters and gradually converge around the quantization centroids as regularization loss is minimized along with the main accuracy loss. The rate of convergence to the target quantization levels depends on (i) the number of fine-tuning epochs, (2) the regularization strength (λ_q). It is worth noting that λ_q is a hyper-parameter that controls the tradeoff between the accuracy loss and the regularization loss. Fixed value can be presumed ahead of training or fine-tuning, however careful setting of such parameter can yield optimum results. (Choi et al., 2018b) considers the regularization coefficient as a learnable parameter.

**Arbitrary-bitwidth quantization.** Considering the following sinusoidal regularizer, with step_q denoting the quantization step, and Δ is an offset.

\[
R(W) = \lambda_q \sum_i \sin^2 \left( \frac{\pi w_i + \Delta}{\text{step}_q} \right)
\]  

SinReQ provides generality in two aspects. First, the flexibility to adapt for arbitrary number of bits. The parameter step_q controls the periodicity of the sinusoidal function. Thus, for any arbitrary bitwidth (qbits), step_q can be tuned to match the respective quantization step. For uniform quantization: step_q = 2^qbits - 1

The second aspect of generality is the seamless accommodation for different quantization styles. There are two styles of uniform quantization: mid-tread and mid-rise. In mid-tread, zero is considered as a quantization level, while in mid-rise, quantization levels are shifted by half a step such that zero is not included. Ternary quantization, using \{-1, 0, 1\}, is an example of the former, while binary quantization, using \{-1, 1\}, is an example of the latter. Figure 2 (a) shows the second conv layer of CIFAR10 at 3 bits, top row: mid-rise quantization, and bottom row: mid-tread quantization.

**Layer-wise optimization.** As different layers have different levels of sensitivity to the quantization bitwidth (Elthakeb et al., 2018), enabling layer-wise quantization opens the possibility for heterogenous bitwidth quantization and consequently more optimized quantized networks. This can be achieved by adding a custom regularizer (as shown in

Figure 2. Evolution of weight distributions over training epochs (with the proposed regularization) at different layers and bitwidths for CIFAR10 and SVHN. (a) CIFAR10, second convolution layer with 3 bits, top row: mid-rise type of quantization (shifting by half a step to exclude zero as a quantization level); bottom row: mid-tread type of quantization (zero is included as a quantization level). (b) SVHN, top row: first convolution layer with 4 bits; bottom row: first fully connected layer with 5 bits.

4. Evaluation: SinReQ in Action

To demonstrate the effectiveness of our proposed sinusoidal regularization, we evaluated it on three neural networks (CIFAR10, SVHN, and ResNet-20) Here, we focus on fine-tuning from a pretrained models as compared to training from scratch.

Best solution that is the closest to one of those minima pockets where weight values are nearly matching the desired quantization levels, hence the name quantization-friendly. Algorithm 1 details the implementation procedure of SinReQ regularization using LeNet as an example.

```python
R(W) = \lambda_q \sum_i \sin^2 \left( \frac{\pi w_i + \Delta}{\text{step}_q} \right)
```

```latex
\begin{align}
R(W) = \lambda_q \sum_i \sin^2 \left( \frac{\pi w_i + \Delta}{\text{step}_q} \right)
\end{align}
```
1.6% respectively. While the accuracy is maximized. This demonstrates a valid-
we add the regularization losses of all layers to the main
their training. While this technique consistently improves the accuracy, SinReQ does not require changes to
across all layers and in the range of 0.5 − 10. We reckon that
careful setting of \( \lambda_q \) across the layers and during the training
epochs is essential for optimum results (Choi et al., 2018b).

5. Conclusion

Deep quantization of DNNs promises to be a powerful tech-
vice of loss in accuracy that needs to be remedied. This paper provided a new approach in using sinusoidal regulariza-
tions terms to push the weight values closer to the quantized
accuracy improvement. The convergence behavior, however,
other quantized training algorithms by improving the quality
methods respectively. That is averaging to around 2% and
SinReQ does not require changes to
base training algorithm or the neural network topology.

Comparison to existing methods. We assess the efficacy of
training quantized networks, DoReFa, and WRPN. Table 1 summarizes
and SinReQ. Results show that integrating
within the training algorithm achieves up to 5.3%,
and WRPN methods respectively. That is averaging to around 2% and
6% respectively.

Convergence analysis. Figure 3 (a), and (b) show the conver-
gnosis loss over fine-tuning epochs for (a) CIFAR10, (b) SVHN.
Comparing convergence behavior with and without SinReQ during
accuracy loss and pass the entire collective loss to the
gradient-descent optimizer.

Comparison to existing methods. We assess the efficacy of
SinReQ on boosting the performance of existing methods for
layers and during the training
epochs is essential for optimum results (Choi et al., 2018b).

5. Conclusion

Deep quantization of DNNs promises to be a powerful tech-
vice of loss in accuracy that needs to be remedied. This paper provided a new approach in using sinusoidal regulariza-
tions terms to push the weight values closer to the quantized
levels. This mathematical approach is versatile and augments
accuracy improvement. The convergence behavior with and without
SinReQ for the case of training from scratch for VGG-11. It can be noticed
that, at the onset of training, the accuracy in the presence of
SinReQ is behind that without SinReQ. This can be explained
as a result of optimizing for an extra objective in case of
SinReQ as compared to without. Shortly thereafter, the
regularization effect kicks in and eventually achieves ∼6% accuracy improvement. The convergence behavior, however,
is primarily controlled by the regularization strength
(\( \lambda_q \)). As briefly mentioned in section 3.2, \( \lambda_q \in [0, \infty) \) is a
hyperparameter that weights the relative contribution of the
proposed regularization objective to the standard accuracy
objective. In the context of neural networks, it is sometimes
desirable to use a separate setting of \( \lambda_q \) for each layer of the
network. Throughout our experiments, \( \lambda_q \) is set the same

References

5th International Conference on Learning Representations,
ICLR 2017, Toulon, France, April 24-26, 2017, Conference
Track Proceedings, 2017. OpenReview.net. URL

Choi, J., Wang, Z., Venkataramani, S., Chuang, P. I.-J., Srinivasan, V.,
and Gopalakrishnan, K. Pact: Parameterized clipping activation for quantized neural networks. CoRR,
abs/1805.06085, 2018a.

Choi, Y., El-Khamy, M., and Lee, J. Learning low
precision deep neural networks through regular-
ization. CoRR, abs/1809.00095, 2018b. URL

Choromanska, A., Henaff, M., Mathieu, M., Arous, G. B.,
and LeCun, Y. The Loss Surfaces of Multilayer Networks.
In Artificial Intelligence and Statistics, 2015.

Courbariaux, M., Bengio, Y., and David, J. Binaryconnect:
Training deep neural networks with binary weights
during propagations. In Cortes, C., Lawrence, N. D., Lee,
D. D., Sugiyama, M., and Garnett, R. (eds.), Advances in
Neural Information Processing Systems 28: Annual
Conference on Neural Information Processing Systems
2015, December 7-12, 2015, Montreal, Quebec, Canada, pp.
3123–3131, 2015. URL

Elhakeb, A. T., Pilligundla, P., Yazdanbakhsh, A.,
Miresghallah, F., and Esmaeilzadeh, H. Releq: A
reinforcement learning approach for deep quantization


