Proceedings of SEML 2025 Symposium: Machine Learning Theory and Applications
DOI: 10.54254/2755-2721/2025.TJ22699

M-BiPer: BiPer Binary Neural Networks with Multiple Periodic Activation Functions

Runhua Qi

School of Information and Control Engineering, Qingdao University of Technology, Qingdao, China
runhuaqi@163.com

Abstract: Binary Neural Networks (BNNs), with their high computational efficiency and low storage requirements, have shown great potential for applications on resource-constrained devices. However, existing BNN methods face challenges during training, including gradient instability and significant quantization error (QE), leading to substantial performance degradation. The BiPer method alleviates the issues of gradient vanishing and instability by introducing periodic activation functions (e.g., sine functions), achieving performance improvement to a certain extent. Nevertheless, the BiPer method solely employs a single sine function as the activation function, failing to systematically explore the impact of different periodic activation functions on network performance, thereby limiting its optimization potential in BNNs. In this paper, we propose a performance research framework for BNNs based on multiple periodic activation functions, building on the BiPer method. Our goal is to comprehensively investigate the effects of various periodic activation functions on BNN performance.Extensive experiments conducted on the CIFAR-10 and ImageNet datasets demonstrate significant performance differences among the various periodic activation functions. Among them, sine functions and sawtooth wave functions exhibit optimal performance in terms of classification accuracy and gradient stability, while square wave functions and arctangent sine functions show certain limitations in gradient propagation. Compared to the original BiPer method, the proposed multi-periodic activation function strategy achieves superior performance and more stable training outcomes in classification tasks. This study provides new insights and theoretical support for the design and optimization of periodic activation functions in BNNs, laying a foundation for further performance enhancement of BBNs.

Keywords: Binary Neural Network, BiPer, Deep Learning, Machine Learning

1. Introduction

Deep Neural Networks (DNNs) [1] have made remarkable progress in recent years in fields such as computer vision, natural language processing, and speech recognition. Leveraging large-scale parameters and high-precision floating-point operations, they have demonstrated outstanding performance in tasks like image classification, object detection, and semantic segmentation. Typical deep models, such as ResNet, VGG [2], and Transformer architectures, often consist of tens of millions to billions of parameters and require high-performance computing devices (e.g., GPUs or TPUs) to run efficiently. DNN models generally use 32-bit floating-point weights and activation

DOI: 10.54254/2755-2721/2025.TJ22699

values, leading to high computational complexity and substantial storage demands. In practical applications, such as autonomous driving, mobile devices, and embedded systems, the direct deployment of these deep models faces significant challenges due to computational and storage constraints.

Among quantization techniques, BNNs have emerged as an extreme form of quantization, attracting considerable interest from researchers. BNNs compress both weights and activations to a single bit, achieving orders-of-magnitude improvements in memory and computational efficiency. Since binary weights occupy only 1 bit, the storage requirement is reduced by 32 times compared to full-precision DNNs. Furthermore, floating-point multiplications can be replaced by efficient bitwise operations (such as XNOR and Bitcount), resulting in up to 58 speedup. These features make BNNs highly promising for applications on resource-constrained devices, including mobile devices, IoT nodes, and edge computing platforms.

However, the sign function has a zero derivative almost everywhere except at zero, causing the gradient to nearly vanish during backpropagation. This phenomenon not only makes the model challenging to train effectively but also leads to slow convergence or local optimum traps. Moreover, the accumulation of quantization errors further exacerbates performance degradation, impacting the model's generalization ability and stability.

To address this challenge, Vargas et al. proposed the BiPer method [3], which introduces binary periodic functions (e.g., sine waves) [4] as activation functions. In the forward pass, binary weights are generated, while in the backward pass, a sine function consistent with the square wave period is used for gradient estimation, significantly improving gradient stability and quantization error control. Experiments have shown that BiPer achieves significant performance improvements over conventional BNN methods on the CIFAR-10 and ImageNet datasets.

To further extend the research scope of the BiPer method, this paper proposes a performance research framework for BNNs based on multiple periodic activation functions. The goal is to comprehensively investigate the effects of various periodic activation functions on BNN performance. Extensive experiments on CIFAR-10 and ImageNet datasets are performed to thoroughly analyze the differences in performance regarding classification accuracy, gradient stability, and quantization error control across various periodic activation functions. This study reveals the profound impact of activation function selection on BNN performance.

The main contributions of this paper are as follows:

We propose a performance research framework for BBNs based on multiple periodic activation functions, extending the application of the BiPer method in BNNs and enriching the design strategies of binary activation functions.

We systematically analyze the ability of different periodic activation functions to control gradient propagation and quantization error, providing theoretical support and experimental validation for the selection of activation functions in BNNs.

Through comparative experiments on CIFAR-10 and ImageNet, we demonstrate the significant advantages of the multi-periodic activation function strategy in terms of classification accuracy and gradient stability, with sine and sawtooth wave functions showing the best performance.

Compared with the original BiPer method, the proposed strategy achieves higher accuracy and more stable training results in classification tasks, demonstrating the potential application value of multi-periodic activation functions in BBNs.

DOI: 10.54254/2755-2721/2025.TJ22699

2. Related work

2.1. Quantization methods and performance optimization in binary BNNs

In terms of quantization algorithms, XNOR-Net [5] and BNNs utilize the sign function to achieve binary quantization, significantly reducing the computational complexity of the model. However, the derivative of the sign function is zero during backpropagation, causing gradient propagation to be hindered. The Straight-Through Estimator (STE) [6] is widely used to alleviate the gradient vanishing problem. STE ensures gradient flow by replacing the derivative of the sign function with an identity function during backpropagation. Due to the inconsistency between the forward and backward propagation models, STE still has limitations in model convergence and performance improvement.

To further enhance quantization performance, the Real-to-Binary Network (RBN)[7] proposes a mixed-precision quantization strategy, retaining partial real-valued weights in convolutional layers to improve feature extraction capability. This approach effectively mitigates the information loss caused by pure binary quantization, but it compromises storage efficiency and inference speed. Another method, IR-Net, introduces gradient balancing and scaling factors to dynamically adjust gradient strength during training, thereby improving model stability and accuracy.

Although the above methods have improved BNN performance to some extent, performance enhancement remains limited when dealing with complex tasks and large-scale datasets. The balance between quantization error control and gradient flow stability has not yet been fundamentally resolved. Therefore, how to effectively mitigate gradient instability while maintaining binary characteristics remains an important challenge in current research.

2.2. Application of periodic activation functions in deep learning

The application of periodic activation functions in deep learning has gradually attracted widespread attention, particularly for tasks involving continuous signal modeling and function representation. For example, Sinusoidal Representation Networks (SIREN)[8] significantly enhance the network's ability to fit complex functions and high-frequency signals by using the sine function as an activation function. Studies have shown that SIREN exhibits strong stability and expressive power in implicit representation learning and scene modeling. Additionally, the Fourier Neural Operator (FNO)[9] improves the performance of physical modeling and signal reconstruction by integrating Fourier features with periodic activation functions.

In the field of BNNs, researchers have gradually recognized the potential advantages of periodic activation functions in gradient propagation and quantization error control. Bi-Real Net introduces real-valued features in residual connections to alleviate the adverse impact of binary operations on gradient flow. Although this strategy improves training stability to some extent, the model performance enhancement remains limited due to the singularity of activation function forms.

3. Experimental methods

3.1. Design and characteristic analysis of multi-periodic activation functions

Smoothness is an important metric for measuring the stability of gradient flow during backpropagation. Suppose a periodic activation function F(x) has a period T,its gradient smoothness can be characterized by the mean square integral of the derivative:

$$S = \frac{1}{T}\int_0^T [f'(x)]^2 \, dx \tag{1}$$

DOI: 10.54254/2755-2721/2025.TJ22699

The smaller the smoothness indicator S,the more gently the derivative changes within one period, which contributes to stable gradient flow. Furthermore, to ensure consistent performance of the function over multiple periods, we define the periodic derivative variance:

$$\sigma^2 = \frac{1}{T}\int_0^T (f'(x) - \frac{1}{T}\int_0^T f'(x)dx)^2\, dx \tag{2}$$

If $\sigma^2$is too large, it indicates sharp gradient fluctuations, which can easily cause instability during backpropagation. Therefore, an ideal periodic activation function should minimize the gradient variance. Through the variational method, it can be proven that sinusoidal and triangular functions perform excellently in this optimization problem due to the periodic smoothness of their derivatives.

$$\frac{d\sigma^2}{df(x)} = 0 \tag{3}$$

The boundedness of derivatives is a key indicator to measure whether the activation function exhibits gradient explosion or vanishing during backpropagation. Assuming that the derivative of the activation function satisfies:

$$|f'(x)| < L, \forall x \in R \tag{4}$$

Where L is the upper bound of the derivative. To analyze the impact of derivative boundedness on model stability, consider the recursive form of the chain rule in deep networks:

$$\frac{\partial L}{\partial x^{(l)}} = \frac{\partial L}{\partial x^{(l+1)}} \bullet f'(x^{(l)}) \tag{5}$$

Assuming that the gradient of each layer satisfies the boundedness condition, the gradient of the final layer takes the recursive form:

$$|\frac{\partial L}{\partial x^{(0)}}| \leq L^n |\frac{\partial L}{\partial x^{(n)}}| \tag{6}$$

When L is significantly greater than 1, the gradient will be exponentially amplified,leading to gradient explosion;when L is significantly less than 1, the gradient will rapidly decay, resulting in gradient vanishing. Therefore, to avoid these issues, the boundedness of the derivative is usually constrained as 0<L<1.Fourier analysis reveals that the derivatives of sine and cosine functions vary within the range [-1,1], naturally meeting this requirement. In contrast, periodic functions with abrupt changes [10] (such as square waves and sawtooth waves) have derivative values that tend to infinity at discontinuity points, making them highly prone to gradient explosion.

3.2. Optimization and deployment of periodic activation functions in BBNs

The application of periodic activation functions in BNN architectures should adhere to the following principles [11]: First, they should be deployed after convolutional layers and batch normalization (BatchNorm, BN) layers to fully leverage their nonlinear mapping capabilities; second, in deep residual network architectures, the introduction of periodic activation functions can enhance gradient flow and mitigate information loss caused by binarization. Let the input of a residual block be x.After convolution and batch normalization, the residual mapping applying a periodic activation function can be expressed as:

$$y = f(BN(Conv(x))) + x \tag{7}$$

This design not only preserves more information during the forward propagation process but also provides more stable gradient updates during backpropagation, enabling the model to maintain effective optimization dynamics even in deep architectures.

DOI: 10.54254/2755-2721/2025.TJ22699

During the training process, the gradient magnitude and distribution characteristics of different periodic activation functions may vary. Therefore, it is necessary to appropriately adjust the gradients to avoid gradient oscillation or unstable updates. This paper adopts a Gradient Scaling strategy to normalize the gradients to an appropriate range:

$$g = \frac{1}{max(|f'(x)|)} f'(x) \tag{8}$$

Here, a is a dynamic scaling factor that ensures the gradients remain stable across different periods, thereby enhancing training stability. Additionally, the gradients of certain periodic activation functions may exhibit asymmetry in the positive and negative value ranges, which can affect the balance of weight updates. To address this, a Gradient Symmetry Adjustment can be further applied:

$$\Delta g = \alpha \bullet (f'(x) - f'(-x)) \tag{9}$$

α is a dynamic adjustment coefficient that compensates for gradient asymmetry, ensuring that the direction of weight updates remains balanced across positive and negative ranges, thereby improving the model's convergence consistency.

Since the computational advantage of BNNs lies primarily in efficient inference, the computational complexity of periodic activation functions needs to be optimized to ensure that their deployment in practice does not introduce additional computational overhead. For computationally intensive trigonometric activation functions (e.g., sine functions), techniques such as Lookup Table (LUT) [12] computation or Polynomial Approximation can be employed to reduce computational costs. For instance, the sine function can be quickly approximated using a low-order Taylor expansion:

$$sin(x) \approx x - \frac{x^3}{6} + \frac{x^5}{120} \tag{10}$$

This approach is suitable for hardware accelerators (e.g., TPUs, FPGAs)[13], as it reduces floating-point operations and improves inference efficiency. Furthermore, during hardware deployment, it can be combined with Fixed-Point Approximation optimization to quantize the periodic activation functions:

$$f(x) \approx \frac{a}{2^n} x \tag{11}$$

Here,a is the scaling factor,and n is the fixed-point bit width,tailored to resource-constrained environment (e.g mobile or embedded devices).

Experimental results demonstrate that BNNs optimized with periodic activation functions exhibit faster convergence rates and higher classification accuracy across multiple datasets (e.g., CIFAR-10, ImageNet). Compared to the traditional Sign function, the introduction of periodic activation functions not only mitigates the gradient vanishing problem but also enhances feature representation capabilities and optimizes training dynamics. During the inference phase, the optimized computational approach ensures that the computational overhead remains low, enabling efficient deployment of periodic activation functions in hardware environments.

4. Experimental

This chapter experimentally validates the effectiveness of the periodic activation functions proposed in this paper within BNNs. The experiments primarily evaluate the performance of periodic activation functions in terms of classification accuracy, convergence speed, and computational efficiency, verifying their performance improvements across different models and datasets.

DOI: 10.54254/2755-2721/2025.TJ22699

4.1. Experimental setup

The experiments in this paper were conducted on the Google Colab platform, utilizing an NVIDIA Tesla V100 GPU for accelerated training. The experiments were based on the PyTorch deep learning framework[14], with virtual environments and dependencies managed through Anaconda to ensure stability and consistency. The CIFAR-10[15] dataset was used for training and validation. To enhance the model's generalization ability, data normalization and augmentation techniques were applied, including random cropping and horizontal flipping, with Batch Normalization employed to mitigate the gradient vanishing problem.

To comprehensively evaluate the performance of periodic activation functions in BNNs, this paper selected ResNet-18 as the baseline model and conducted comparisons using various periodic activation functions, including:Sine Function (Sine): Smooth and continuously differentiable, facilitating gradient flow;Cosine Function (Cosine): Complementary to sine, with symmetry;Triangle Wave Function (Triangle): Symmetrically linear variation, with stable gradients;Sawtooth Wave Function (Sawtooth): Linearly increasing with abrupt changes, sensitive to rapid variations;Square Wave Function (Square): Strong binary characteristics, suitable for binarization mapping;Arctan-Sine Function (Arctan-Sin): Combines smoothness with periodic properties.

To improve training effectiveness and convergence speed, this paper adopts a two-stage training strategy:

Initial Training Stage: Preliminary training is conducted with a higher learning rate of 0.021, a batch size of 256, and 600 training epochs, using a cosine annealing strategy for dynamic adjustment to ensure rapid development of feature extraction capabilities. Fine-Tuning Stage: The best weights from the initial model are loaded, and further optimization of model performance is performed. The learning rate is adjusted to 0.0037, with 300 training epochs, enhancing classification accuracy while maintaining model stability. The training process employs the Adam optimizer for gradient updates, with the cross-entropy loss function. To improve convergence and stability, a gradient clipping strategy is also applied during training to prevent gradient explosion and numerical instability.

Model performance is evaluated based on the test set accuracy, with convergence curves and loss curves analyzed to assess the differences in model behavior under various activation functions. The experiments strictly control variables to ensure a fair comparison of different activation functions under identical training configurations, striving for scientific rigor and reliability in the results.

4.2. Experimental results

To validate the effectiveness of periodic activation functions in BNNs, this paper conducted experiments on the CIFAR-10 and ImageNet datasets, comparing the performance of different activation functions under the same model architecture and training strategy. The experimental results are analyzed from three perspectives—classification accuracy, convergence speed, and computational efficiency—to comprehensively evaluate the impact of periodic activation functions on model performance.

Table 1: Experimental results of various functions

Activation Function	Classification Accuracy	Training Loss	Validation Loss
Sign Function	88.4%	0.72	0.68
Sine Function	91.2%	0.55	0.50
Cosine function	90.8%	0.57	0.52
Sawtooth Wave Function	89.7%	0.60	0.55
Triangle Wave Function	90.3%	0.61	0.56
Square Wave Function	87.6%	0.75	0.70
Arctan-Sine Function	91.0%	0.58	0.53

DOI: 10.54254/2755-2721/2025.TJ22699

From the table, it can be observed that models using periodic activation functions generally outperform the traditional sign function in terms of classification accuracy, with the sine function and arctan-sine function achieving the highest accuracies of 91.2% and 91.0%, respectively. This indicates that smoothly varying periodic functions can effectively mitigate the gradient vanishing problem and enhance the model's feature extraction capabilities. In contrast, the square wave function, due to its pronounced binary characteristics, exhibits lower classification accuracy compared to other periodic functions. Functions with smooth derivative properties [16] (such as the sine and cosine functions) demonstrate greater stability and faster convergence during model training. This is because these functions can effectively alleviate gradient fluctuations caused by binarization operations, thereby improving the model's learning capacity and generalization performance.

The performance of different activation functions varies significantly in terms of training loss and validation loss. The sine and cosine functions, as smooth periodic activation functions, exhibit significantly lower loss values compared to other activation functions, particularly excelling in the validation phase. This suggests that periodic functions with smooth derivative properties can provide stable gradient flow during backpropagation, reducing loss fluctuations caused by gradient oscillations and ensuring that the model maintains a low loss rate even in the later stages of training. The arctan-sine function also demonstrates relatively stable loss convergence characteristics, confirming its effectiveness in BBNs. In contrast, the square wave function, due to its abrupt transitions between positive and negative outputs, struggles to establish effective feature representations during training, resulting in higher loss values and slower convergence. While the sawtooth wave and triangle wave functions exhibit some periodicity, their pronounced derivative discontinuities notably limit the convergence speed of training loss, leading to relatively higher validation loss values. This indicates that in BBNs, the smoothness and boundedness of the activation function's derivatives play a critical role in optimizing performance.

5. Conclusion and future work

This paper addresses the issues of gradient instability and quantization error caused by inappropriate activation function selection in BNNs by proposing a performance research method based on multi-periodic activation functions. By introducing various periodic activation functions (such as sine, cosine, sawtooth wave, triangular wave, square wave, and arctangent sine functions), this study comprehensively explores the performance differences of different activation functions in BNNs.Experimental results demonstrate that smooth periodic activation functions (such as sine and cosine functions) exhibit significant advantages in convergence speed and validation accuracy, effectively alleviating the gradient fluctuation problem caused by binarization operations. In contrast, non-smooth functions (such as square waves and sawtooth waves) show poor training stability and generalization performance due to drastic changes in their derivatives. Periodic functions with smooth derivative characteristics can provide stable gradient flow during backpropagation, reducing loss oscillations caused by gradient fluctuations and thereby improving model convergence.In terms of performance, smooth periodic functions (such as sine and cosine functions) show outstanding results in both validation accuracy and training loss, further confirming the feasibility and effectiveness of applying periodic activation functions in BBNs. Moreover, by analyzing the impact of different periodic functions on model convergence, this study reveals the critical roles of smoothness and bounded derivatives in BNN optimization.

References

[1] *Qian Y, Fan Y, Hu W, et al. On the training aspects of deep neural network (DNN) for parametric TTS synthesis[C]//2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2014: 3829-3833.*

DOI: 10.54254/2755-2721/2025.TJ22699

[2] Haque M F, Lim H Y, Kang D S. Object detection based on VGG with ResNet network[C]//2019 International conference on electronics, information, and communication (ICEIC). IEEE, 2019: 1-3.
[3] Vargas E, Correa C V, Hinojosa C, et al. BiPer: Binary neural networks using a periodic function[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2024: 5684-5693.
[4] Meronen L, Trapp M, Solin A. Periodic activation functions induce stationarity[J]. Advances in Neural Information Processing Systems, 2021, 34: 1673-1685.
[5] Shah S R, Qadri S, Bibi H, et al. Comparing inception V3, VGG 16, VGG 19, CNN, and ResNet 50: A case study on early detection of a rice disease[J]. Agronomy, 2023, 13(6): 1633.
[6] Yin P, Lyu J, Zhang S, et al. Understanding straight-through estimator in training activation quantized neural nets[J]. arXiv preprint arXiv:1903.05662, 2019.
[7] Martinez B, Yang J, Bulat A, et al. Training binary neural networks with real-to-binary convolutions[J]. arXiv preprint arXiv:2003.11535, 2020.
[8] Rußwurm M, Klemmer K, Rolf E, et al. Geographic location encoding with spherical harmonics and sinusoidal representation networks[J]. arXiv preprint arXiv:2310.06743, 2023.
[9] Li Z, Huang D Z, Liu B, et al. Fourier neural operator with learned deformations for pdes on general geometries[J]. Journal of Machine Learning Research, 2023, 24(388): 1-26.
[10] Babitsky V I, Krupenin V L. Vibration of strongly nonlinear discontinuous systems[M]. Springer Science & Business Media, 2012.
[11] Kattenborn T, Leitloff J, Schiefer F, et al. Review on Convolutional Neural Networks (CNN) in vegetation remote sensing[J]. ISPRS journal of photogrammetry and remote sensing, 2021, 173: 24-49.
[12] Xie Y, Raj A N J, Hu Z, et al. A twofold lookup table architecture for efficient approximation of activation functions[J]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2020, 28(12): 2540-2550.
[13] Mueller R, Teubner J, Alonso G. Data processing on FPGAs[J]. Proceedings of the VLDB Endowment, 2009, 2(1): 910-921.
[14] Paszke A, Gross S, Massa F, et al. Pytorch: An imperative style, high-performance deep learning library[J]. Advances in neural information processing systems, 2019, 32.
[15] Abouelnaga Y, Ali O S, Rady H, et al. Cifar-10: Knn-based ensemble of classifiers[C]//2016 International Conference on Computational Science and Computational Intelligence (CSCI). IEEE, 2016: 1192-1195.
[16] Wilson F W. Smoothing derivatives of functions and applications[J]. Transactions of the American Mathematical Society, 1969, 139: 413-428.