Generalized Threshold Optimization with Harmony Multi-Threshold Neurons for Accurate ANN-to-SNN Conversion

Wenhan Zhang[1], Zihan Huang[1], Tong Bu[1,2], Tiejun Huang[1], Zhaofei Yu[1,2]*

[1]School of Computer Science, Peking University
[2]Institute for Artificial Intelligence, Peking University
faizwh@stu.pku.edu.cn, hzh@stu.pku.edu.cn, putong30@pku.edu.cn, yuzf12@pku.edu.cn, tjhuang@pku.edu.cn

Abstract

Spiking Neural Networks (SNNs) are a promising paradigm designed to emulate the brain's energy efficient by incorporating the timing of spikes. Conversion is an efficient way to obtain high-performance SNNs from Artificial Neural Networks (ANNs). Existing conversion methods often face a trade-off between accuracy and time steps, which is largely caused by the incomplete release of residual membrane potentials. To minimize the conversion error, this paper proposed a harmonious mathematical property-based neuron, called Harmony Multi-Threshold Neurons (H-MT Neuron), which utilizes multiple spikes to minimize residual membrane potentials. The proposed neuron is further enhanced with an optional effective communication mechanism to achieve more accurate conversion. In addition, we propose a threshold optimization method applicable to a broader range cases of spiking neurons to to find the optimal neuron thresholds. Experiment results demonstrate that our method achieve superior accuracy on ImageNet benchmark datasets while significantly reducing the required time steps and energy consumption.

Code and Technical Appendix —
https://github.com/faizwh/H-MT.git

1 Introduction

Spiking Neural Networks (SNNs), regarded as the third generation of neural network models (Maass 1997), are distinguished by their biological plausibility and dynamic neuron behaviors (Gerstner et al. 2014), exhibiting significant potential to rival Artificial Neural Networks (ANNs). The two architectures differ in computational mechanisms: ANNs rely on continuous floating-point value transmission, whereas SNNs propagate sparse, binary spikes across layers. Such spike-based communication better emulates biological neuron dynamics and enhances energy efficiency (Merolla et al. 2014; Davies et al. 2018; DeBole et al. 2019; Pei et al. 2019).

However, the discrete nature of spikes introduces non-differentiability, posing challenges for training. Recent advancements in direct training have made progress in addressing this challenge via surrogate gradient methods (Neftci, Mostafa, and Zenke 2019). These methods approximate gradients of non-differentiable spike processes using smooth functions, enabling practical backward propagation and gradient descent, thereby improving SNN performance (Fang et al. 2021; Duan et al. 2022; Shi, Hao, and Yu 2024; Ding et al. 2025). Nevertheless, surrogate gradients remain approximations and may mislead the training (Gygax and Zenke 2025). Moreover, the temporal nature of SNNs leads to high computational costs due to the need for backpropagation through time(BPTT). Although several online training methods have been proposed to estimate gradients more efficient(Xiao et al. 2022; Bohnstingl et al. 2022; Meng et al. 2023; Zhu et al. 2024), they often results in sub-optimal accuracy. Consequently, direct training of SNNs remains a persistent challenge.

Meanwhile, another prominent spiking deep learning methodology is ANN-to-SNN conversion (Cao, Chen, and Khosla 2015; Han, Srinivasan, and Roy 2020; Li et al. 2021; Deng and Gu 2021; Bu et al. 2022a; Bu, Li, and Yu 2025; Zhao et al. 2025; Ding et al. 2021). This method converts pre-trained ANNs to SNNs by replacing nonlinear units with spiking neurons, such as traditional Integrate-and-Fire (IF) neuron with single threshold or its multi-threshold variants designed for higher precision within shorter inference timesteps. Although ANN-to-SNN conversion avoids complexities of direct training and produces high-performance SNNs with accuracy comparable to original ANNs, it typically requires more time steps for precision, resulting in higher latency and energy consumption compared to surrogate gradient based direct training.

In this article, we identify key inherent defect in many designs of multi-threshold neuron, due to which the output of these neurons can't be well-represented by specific mathematical expressions as IF neuron. We then further propose a novel neuron structure called H-MT neuron, with harmonious mathematical properties and without such defect, designed to minimize residual membrane potential and thereby reduce conversion error. The proposed neuron integrate two symmetric components to approximate the linear function $y = x$, along with a communication mechanism that helps alleviate unevenness errors(Bu et al. 2022b). By simulating $y = x$, we can keep everything in the form of spike, thus maintaining the energy efficiency of SNNs. Furthermore, we generalize the threshold optimization method from Huang

*Corresponding author

et al. to obtain the optimal threshold, broadening its applicability to a wider variety multi-threshold neuron architectures for simulating either ReLU or $y = x$ function.

Our main contributions are summarized as follows:

- We identify the key inherent defect in many designs of multi-threshold neuron, and further propose a novel multi-threshold neuron with nice mathematical properties and without the defect, capable of simulating both ReLU and $y = x$ functions. The design is further enhanced with a simple yet effective communication mechanism to reduce unevenness errors.
- We propose a general threshold iteration optimization method to determine the optimal threshold, enabling its application across a wide range of spiking neuron structures. This approach facilitates accurate threshold selection for converting both ReLU and $y = x$ functions.
- We evaluate our method on the ImageNet dataset. Compared with previous CNN-SNN conversion methods, especially with those solely based on modification of neurons, the proposed method achieves higher accuracy and decreased energy consumption.

2 Related Works

2.1 Neuron Design in SNNs

Spiking neurons serve as the fundamental computational units in SNNs. Researchers focus on both direct training and ANN-to-SNN conversion consistently investigate existing spiking neuron models and propose novel neuron structures to improve networks' performance and efficiency. Bu et al. conducted an in-depth analysis of the classical IF model as well as conversion errors in ANN-to-SNN, and put forward corresponding adjustment on neuron settings, such as initial membrane potential, inspiring following studies. Some previous works (Lv et al. 2024; Hao et al. 2024; Wang and Zhang 2023; Luo et al. 2024; Huang et al. 2024, 2025; Wang et al. 2025) modified traditional single-threshold IF neuron, for the sake of higher precision, to n-threshold neurons through essentially almost identical designs, which partition single threshold into n equal segments. With 0 taken into account, such neurons can output $n + 1$ distinct values per time-step.

Meanwhile, other studies (Li, Zhao, and Zeng 2022; Kim et al. 2018) introduced periodically varying thresholds during inference, but can cause unexpectedly long inference time for high accuracy. Moreover, Notably, distinct from the aforementioned efforts aiming at simulating ReLU or $y = x$, researchers (Jiang et al. 2024) proposed heuristic designs for composite neurons composite of smaller learnable-threshold sub-neurons, to emulate complex nonlinear modules widely used in Transformer architectures, e.g., GELU, e^x, LayerNorm and so on.

2.2 Neuron Threshold Policy for ANN-to-SNN conversion

Beyond neuron structure modifications, determining spike neuron thresholds is critical. Learning thresholds as trainable parameters is one represent that is popular in direct training. As for ANN-to-SNN, thresholds are traditionally set to the 99.9% quantile of the activation value of the replaced ANN nonlinear units, as recommended by Rueckauer et al.. which does show some excellence in practice despite being empirical. Researchers have also been continuously exploring threshold optimization policies grounded on more reliable methods and concrete theories. For instance, Li et al. attempted to find better thresholds via grid search, while Bu, Li, and Yu introduced a local threshold balancing algorithm that efficiently finds the optimal thresholds through training and performs fine-grained adjustment of the threshold value. Additionally, Huang et al. developed a threshold iteration optimization method to find the threshold minimizing the conversion error of IF neuron simulating ReLU, under the assumption that inputs obey normal distribution. However, the theorems have some limitations, such that they can only support a rough optimization on the multi-threshold neurons employed in this work. In this article, we extensively generalize these theorems by Huang et al. with rigorous proofs, extending their applicability to a broader range of spiking neuron designs for simulating either ReLU or $y = x$.

3 Preliminaries

3.1 The Integrate-and-Fire (IF) Neuron and Designs of Multi-Threshold Spiking Neuron

Similar to many previous works. We begin by foucsing on the Integrate-and-Fire (IF) Neuron, which is a classical spiking neuron model with one threshold and is used to replace ReLU neuron in ANN-to-SNN conversion. We use the "reset-by-subtraction" mechanism, and its dynamics can be expressed as follows:

$$\boldsymbol{m}^l[t] = \boldsymbol{v}^l[t-1] + \boldsymbol{I}^l[t] = \boldsymbol{v}^l[t-1] + f(\boldsymbol{x}^{l-1}[t]), \tag{1}$$

$$\boldsymbol{s}^l[t] = H(\boldsymbol{m}^l[t] - \boldsymbol{\theta}^l), \tag{2}$$

$$\boldsymbol{v}^l[t] = \boldsymbol{m}^l[t] - \boldsymbol{s}^l[t]\boldsymbol{\theta}^l, \tag{3}$$

$$\boldsymbol{x}^l[t] = \boldsymbol{s}^l[t]\boldsymbol{\theta}^l, \tag{4}$$

where $\boldsymbol{x}^{l-1}[t]$ is the initial input received by the l-th layer as well as the output of the $l-1$-th layer at time-step t, $\boldsymbol{I}^l[t]$ is the input ultimately fed into the neuron in the by l-th layer after some operation f, which can be affine transformation, while $\boldsymbol{m}^l[t]$ and $\boldsymbol{v}^l[t]$ are the membrane potential before and after firing at time-step t. Here θ^l is the threshold of the l-th layer and $H(\cdot)$ is the Heaviside step function, therefore the element of $\boldsymbol{s}^l[t]$, which is the spike vector of l-th layer, is either 1 if there is a spike or 0.

When using $\phi^l(T) = \frac{\sum_{t=1}^{T} x^l[t]}{T}$ as the average output of the l-th layer and $z^l = \frac{\sum_{t=1}^{T} I^l[t]}{T}$ as the average input fed into the spiking neuron at the l-th layer, we will have:

$$\boldsymbol{\phi}^l(T) = \boldsymbol{\theta^l}\frac{\sum_{t=1}^{T} s^l[t]}{T} = \boldsymbol{z}^l - \frac{\boldsymbol{v}^l[T] - \boldsymbol{v}^l[0]}{T}. \tag{5}$$

To estimate $\phi^l(T)$, Bu et al. put forward a strong assumption as shown in Equation (6) (here $u^l[t]$ is identical to $I^l[t]$).

$$\begin{cases} \begin{cases} \boldsymbol{u}^l\left[t\right] \leq 0, \forall t = 1, \cdots, T & \text{if } \boldsymbol{z}^l \leq 0 \\ \boldsymbol{u}^l\left[t\right] \in \left(0, \boldsymbol{\theta}^l\right), \forall t = 1, \cdots, T & \text{if } \boldsymbol{z}^l \in \left(0, \boldsymbol{\theta}^l\right) \\ \boldsymbol{u}^l\left[t\right] \geq \boldsymbol{\theta}^l, \forall t = 1, \cdots, T & \text{if } \boldsymbol{z}^l \geq \boldsymbol{\theta}^l \end{cases} \\ \boldsymbol{v}^l\left[0\right] \in \left[0, \boldsymbol{\theta}^l\right] \end{cases}. \tag{6}$$

With this assumption, the range of $v^l[t]$ can be limited to $[0, \theta^l)$ for $\forall t = 1, \ldots, T$, if $z^l \in (0, \theta^l)$, and a precise mathematical expression of the IF neuron's output can be derived to help the ANN-to-SNN conversion:

$$\boldsymbol{\phi}^l(T) = \frac{\boldsymbol{\theta}}{T} clip\left(\lfloor \frac{\boldsymbol{z}^l T + \boldsymbol{v}^l[0]}{\theta} \rfloor, 0, T \right). \tag{7}$$

To enhance computational accuracy, many previous works have made effort to design novel structures of Multi-Threshold Spiking Neuron to replace ReLU (Lv et al. 2024; Wang and Zhang 2023; Hao et al. 2024) as well as $y = x$ (Huang et al. 2024, 2025) , and have achieved some wonderful performance in comparison to the IF neuron.

Assuming there are n channels with different thresholds to fire spikes inside a neuron, just as IF neuron, these neuron can also derive an equation similar to Equation (5):

$$T\boldsymbol{\phi}^l(T) = \sum_{t=1}^{T}\sum_{i=1}^{n} \boldsymbol{\lambda}_i^l s_i^l[t] = \boldsymbol{z}^l T + \boldsymbol{v}^l[0] - \boldsymbol{v}^l[T]. \tag{8}$$

However, due to intrinsic defects of these neuron's structures, it's hard either to extend the strong assumption from (Bu et al. 2022b) or further derive similar mathematical expressions for these neurons, bringing trouble for more precise optimization. In short, the main inherent defect of these neuron designs is that they can't properly limit the range of neuron's membrane potential after firing spikes at per timestep as IF neuron does, even when under some strong assumption. A detailed analysis can be found at Appendix A.

3.2 Optimization Method for Threshold of Spiking Neuron

Besides the learning method in direct training as well as grid search (Li et al. 2021) and the local threshold balancing algorithm (Bu, Li, and Yu 2025) in ANN-to-SNN conversion, Huang et al. present a novel optimization method for the threshold of the IF neuron to lower the conversion error of simulating ReLU. Assume the input x fed into the neuron (here the x corresponds to z^l mentioned ahead) obey a normal distribution with a mean of μ and a variance of σ^2 (de G. Matthews et al. 2018). The error of conversion from ReLU to IF neuron can be expressed as a function of the threshold θ:

$$QE(\theta) = \int_{-\infty}^{\infty} (f(x,\theta) - \max(x,0))^2 \, e^{-\frac{(x-\mu)^2}{2\sigma^2}} \, dx, \tag{9}$$

$$f(x,\theta) = \frac{\theta}{N} clip(\lfloor \frac{Nx + \frac{\theta}{2}}{\theta} \rfloor, 0, N). \tag{10}$$

Then it can be proved that through the iteration $\theta_{new} = k_{\mu,\sigma,n}(\theta) \times \theta_{ori}$ (here θ_{new} and θ_{ori} means the threshold after and before being updated), the threshold will converge to the unique optimal solution of the error $QE(\theta)$. $k_{\mu,\sigma,n}(\theta)$ is as follows:

$$\begin{aligned} k_{\mu,\sigma,n}(\theta) = & \frac{\mu}{\theta} \frac{1 - \sum_{i=1}^{n} \frac{1}{n} erf(\frac{\frac{2i-1}{2n}\theta - \mu}{\sqrt{2}\sigma})}{1 - \sum_{i=1}^{n} \frac{2i-1}{n^2} erf(\frac{\frac{2i-1}{2n}\theta - \mu}{\sqrt{2}\sigma})} \\ & + \frac{\sigma}{\sqrt{\frac{\pi}{2}}\theta} \frac{1 - \sum_{i=1}^{n} \frac{1}{n} e^{-\frac{(\frac{2i-1}{2n}\theta - \mu)^2}{2\sigma^2}}}{1 - \sum_{i=1}^{n} \frac{2i-1}{n^2} erf(\frac{\frac{2i-1}{2n}\theta - \mu}{\sqrt{2}\sigma})}. \end{aligned} \tag{11}$$

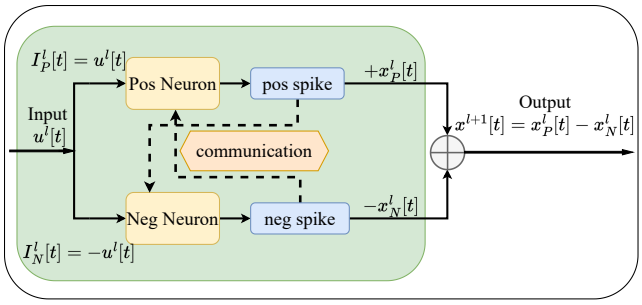


Figure 1: The structure of Harmony Multi-Threshold Spiking Neuron

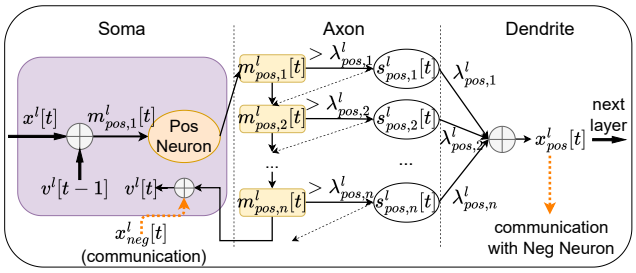


Figure 2: How spikes are emitted inside the Pos Neuron of H-MT. The black dot lines here mean if a spike is fired, the membrane potential should minus the corresponding threshold. Since the communication mechanism is optional, we mark with tangerine dashed lines.

4 Method

In this section, we propose a novel structure of multi-threshold neuron with harmonious mathematical properties as well as a generalized threshold optimization method, to achieve conversion of high accuracy with low latency.

4.1 Harmony Multi-Threshold Spiking Neuron

To better simulate $y = x$ (and ReLU) and to further optimize neuron's performance, it's necessary to have a good design of multi-threshold neuron able to overcome the defects mentioned in Section 3.1 and Appendix A. To properly limit the range of its membrane potential under some strong assumption, the neuron should have a better mechanism for emitting spikes as well as a proper structure. Here we present a new design called **Harmony Multi-Threshold Spiking Neuron** (in short **H-MT**), as it enjoys some good properties and is harmonious.

As shown in Figure 1, inspired by (Jiang et al. 2024), H-MT is composed of two symmetric neurons, respectively called **Pos neuron** and **Neg neuron**. Both are based on the same structure **Basal Neuron** which can simulate ReLU, but receive opposite inputs. When received a transformed input $u^l[t]$, **H-MT** will fed $I_P^l[t] = u^l[t]$ to the Pos Neuron and fed $I_N^l[t] = -u^l[t]$ to the Neg Neuron. The spikes emitted by the two neuron are called **pos spikes** (defined as $x_P^l[t]$) and **neg spike** (defined as $x_N^l[t]$), and the output of H-MT is $x^l[t] = x_P^l[t] - x_N^l[t]$.

To be precise, the basic dynamics of Basal Neuron are as Equation (12) to (18). You may also referred to Figure 2, which describes how spikes are emitted inside the Pos Neuron of H-MT (since the Pos and Neg Neuron are symmetric).

$$\text{Thresholds}: \boldsymbol{\lambda}_1^l = \boldsymbol{\theta}^l, \boldsymbol{\lambda}_2^l = \frac{\boldsymbol{\theta}^l}{2}, \ldots, \boldsymbol{\lambda}_n^l = \frac{\boldsymbol{\theta}}{2^{n-1}} = \boldsymbol{\lambda}_{min}^l, \quad (12)$$

$$\boldsymbol{v}^l[0] = \frac{\boldsymbol{\lambda}_{min}^l}{2}, \quad (13)$$

$$\boldsymbol{m}_1^l[t] = \boldsymbol{v}^l[t-1] + \boldsymbol{I}^l[t], \quad (14)$$

$$s_i^l[t] = \begin{cases} 1, & \text{if } \boldsymbol{m}_i^l[t] >= \boldsymbol{\lambda}_i^l \\ 0, & \text{else} \end{cases}, \quad (15)$$

$$\boldsymbol{m}_{i+1}^l[t] = \boldsymbol{m}_i^l[t] - \boldsymbol{\lambda}_i^l s_i^l[t],\ i = 1, 2, \ldots, n-1, \quad (16)$$

$$\boldsymbol{x}^l[t] = \sum_{i=1}^{n} \boldsymbol{\lambda}_i^l s_i^l[t], \quad (17)$$

$$\begin{aligned} \boldsymbol{v}^l[t] &= \boldsymbol{m}_n^l[t] - \boldsymbol{\lambda}_n^l s_n^l[t] = \boldsymbol{m}_1^l[t] - \boldsymbol{x}^l[t] \\ &= \boldsymbol{v}^l[t-1] + \boldsymbol{I}^l[t] - \boldsymbol{x}^l[t]. \end{aligned} \quad (18)$$

At each time-step, there are 2^n possible output values for the H-MT neuron, from 0,λ_{min}^l to $(2^n-1)\lambda_{min}$. Therefore, we extend the strong assumption from (Bu et al. 2022b) as follows:

$$\forall t = 1, \ldots, T,$$
$$\begin{cases} \boldsymbol{u}^l[t] \le -(2^n-1)\boldsymbol{\lambda}_{min}^l, & \text{if } \boldsymbol{z}^l \le -(2^n-1)\boldsymbol{\lambda}_{min}^l \\ \boldsymbol{u}^l[t] \in (-(2^n-1)\boldsymbol{\lambda}_{min}^l, 0), & \text{if } \boldsymbol{z}^l \in (-(2^n-1)\boldsymbol{\lambda}_{min}^l, 0) \\ \boldsymbol{u}^l[t] \in [0, (2^n-1)\boldsymbol{\lambda}_{min}^l), & \text{if } \boldsymbol{z}^l \in [0, (2^n-1)\boldsymbol{\lambda}_{min}^l) \\ \boldsymbol{u}^l[t] \ge (2^n-1)\boldsymbol{\lambda}_{min}^l, & \text{if } \boldsymbol{z}^l \ge (2^n-1)\boldsymbol{\lambda}_{min}^l \end{cases} \cdot \quad (19)$$

Let $\boldsymbol{\phi}_{P/N}^l(T) = \frac{\sum_{t=1}^{T} \boldsymbol{x}_{P/N}^l[t]}{T}$. Similar to Equation (7), under this strong assumption, we can further derive the mathematical expression for H-MT's output, as given in Equation (20). This derivation is feasible because the Basal Neuron of H-MT is capable of restricting its membrane potential within either $[0, \boldsymbol{\lambda}_{min}^l)$, depending on the input.

$$\begin{aligned} \boldsymbol{\phi}^l(T) &= \boldsymbol{\phi}_P^l(T) - \boldsymbol{\phi}_N^l(T) \\ &= \frac{\boldsymbol{\lambda}_{min}^l}{T} clip(\lfloor \frac{\boldsymbol{z}^l T + \frac{\boldsymbol{\lambda}_{min}^l}{2}}{\boldsymbol{\lambda}_{min}^l} \rfloor, -(2^n-1)T, (2^n-1)T). \end{aligned} \quad (20)$$

Meanwhile, we are faced with another challenge called Unevenness Error, which is first proposed by Bu et al. and calls for more serious consideration. It descirbes the conversion error caused by the uneveness of input. A classical example of the unevenness error can be found in Appendix B for a better understanding. It can be even more complicated when it comes to multi-threshold spiking neurons, as the input and output become more complex, making it difficult to define "evenness" at least.

We notice that such errors are largely caused by specific inputs with extreme magnitude and the clustering of same-signed inputs, which can lead to extreme membrane potentials and further result in extra or missing spikes. To address this issue, we introduce a simple yet powerful **communication mechanism** to reduce unevenness error by preventing extreme membrane potentials. This mechanism, as defined in Equation (21), operates on the membrane potentials of the Pos Neuron and Neg Neuron ($v_P^l[t]$ and $v_N^l[t]$) after spike firing. This mechanism is executed after the neuron dynamic computation at each time-step.

$$\boldsymbol{v}_{P/N}^l[t] \leftarrow \boldsymbol{v}_{P/N}^l[t] + \boldsymbol{x}_{N/P}^l[t], \quad (21)$$

It can be easily proven that under the strong assumption Equation (19), H-MT with communication mechanism can still keep the mathematical expression Equation (20). Moreover, at each time-step, H-MT with communication mechanism will prevent the membrane potential of the two parts from being too extreme, and as following Lemma 4.1 shows:

Lemma 4.1. *H-MT with communication mechanism will have only one of its Pos and Neg neuron to fire spikes at every time-step, as for $\forall t = 1, \cdots, T$, we have:*

$$\boldsymbol{v}_P^l[t] + \boldsymbol{v}_N^l[t] = \boldsymbol{\lambda}_{min}^l. \quad (22)$$

Detailed analysis of H-MT's properties can be found at Appendix C. And the test cases demonstrating how communication mechanism suppresses unevenness error can be found at Appendix D.

4.2 Extension of the Optimization Method for Threshold

Since the mathematical expression of H-MT (Equation (20)) differs from that of IF model (Equation (7)), the optimization method of threshold from (Huang et al. 2025) doesn't apply. Therefore, we generalize the Huang et al.'s theorem and make it applicable for more multi-threshold neuron with proper structure that can derive similar mathematical expression.

Assumption 4.2. According to the (de G. Matthews et al. 2018), assume that the input x fed into the neuron (here the x corresponds to z^l mentioned ahead) follows a normal distribution with mean μ and variance σ^2.

By extending the error of conversion from ReLU to $y = x$, the following definitions and lemmas can be proposed to find the best threshold in such condition.

Definition 4.3. The quantization and clipping errors introduced when approximating the $y = x$ function can be formulated as:

$$QE(\theta, a, b) = \int_{-\infty}^{\infty} (f(x, \theta, a, b) - x)^2 \, e^{-\frac{(x-\mu)^2}{2\sigma^2}} \, dx, \quad (23)$$

$$f(x, \theta, a, b) = \frac{\theta}{n} clip(\lfloor \frac{nx + \frac{\theta}{2}}{\theta} \rfloor, a, b), a, b \in \mathbb{Z}, a < b, \quad (24)$$

where a, b are arbitrary integers not necessarily related to n, and that the $\mathbb{Z}$ here is the set of integer.

We can extend the error function QE to two auxiliary functions QE_1 and QE_2, which facilitate the calculation of the optimal threshold. These functions are defined as follows:

$$QE_1(\theta, k, a, b) = \int_{-\infty}^{\infty} (f_1(x, \theta, k, a, b) - x)^2 \, e^{-\frac{(x-\mu)^2}{2\sigma^2}} \, dx, \quad (25)$$

$$f_1(x, \theta, k, a, b) = k\frac{\theta}{n} clip(\lfloor \frac{nx + \frac{\theta}{2}}{\theta} \rfloor, a, b), a, b \in \mathbb{Z}, a < b, \quad (26)$$

$$QE_2(\theta, k, a, b) = \int_{-\infty}^{\infty} (f_2(x, \theta, k, a, b) - x)^2 \, e^{-\frac{(x-\mu)^2}{2\sigma^2}} \, dx, \quad (27)$$

$$f_2(x, \theta, k, a, b) = \frac{\theta}{n} clip(\lfloor \frac{nx + \frac{k\theta}{2}}{k\theta} \rfloor, a, b), a, b \in \mathbb{Z}, a < b. \quad (28)$$

Then we have following lemmas.

Algorithm 1: Threshold iteration method to find the best threshold for conversion from ReLU/$y = x$ to spiking neuron

Input: Pre-trained ANN Model F_{ANN}, Dataset $\boldsymbol{D}$.
Initialize: Set $\boldsymbol{\theta} \leftarrow 1$ (any positive initial value)
Run the model F_{ANN} on dataset $\boldsymbol{D}$ to statically compute the mean $\boldsymbol{\mu}$ and variance $\boldsymbol{\sigma}^2$ of pre-activations of each ReLU / $y = x$ separately.
repeat
    Compute $\boldsymbol{k_1}$ for current $\boldsymbol{\theta}$ based on $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}^2$ according to Eq (29)
    Update $\boldsymbol{\theta} \leftarrow \boldsymbol{k_1} \cdot \boldsymbol{\theta}$
until $1 - \epsilon < \boldsymbol{k_1} < 1 + \epsilon$, where ϵ tends to 0.
Output: Threshold $\boldsymbol{\theta}$

Lemma 4.4. *For any fixed integer a, b, n, any fixed $\theta > 0, \mu, \sigma > 0$, $QE_1(\theta, a, b)$ reaches the minimal value only when:*

$$\begin{aligned} k =& k_{\mu,\sigma,n,a,b}(\theta) \\ =& \frac{\mu}{\theta} \frac{n\left[a + b - \sum_{i=a+1}^{b} erf\left(\frac{\frac{2i-1}{2n}\theta - \mu}{\sqrt{2}\sigma}\right)\right]}{a^2 + b^2 - \sum_{i=a+1}^{b}(2i-1) erf\left(\frac{\frac{2i-1}{2n}\theta - \mu}{\sqrt{2}\sigma}\right)} \\ &+ \frac{\sigma}{\sqrt{\frac{\pi}{2}}\theta} \frac{n\sum_{i=a+1}^{b} e^{-\frac{\left(\frac{2i-1}{2n}\theta - \mu\right)^2}{2\sigma^2}}}{a^2 + b^2 - \sum_{i=a+1}^{b}(2i-1) erf\left(\frac{\frac{2i-1}{2n}\theta - \mu}{\sqrt{2}\sigma}\right)}. \end{aligned} \tag{29}$$

Lemma 4.5. *When $a \leq 0 \leq b,\ a < b$, for any $\theta > 0, \mu \in \mathbb{R}, \sigma > 0$, we will always have: $k_{\mu,\sigma,n,a,b}(\theta) > 0$.*

Lemma 4.6. *For any fixed integer a, b, n, any fixed $\theta > 0, \mu, \sigma > 0$, for all $k > 0$, $QE_2(\theta, a, b)$ reaches the minimal value only when: $k = 1$.*

Then for integers $a \leq 0 \leq b$ with $a < b$, the following inequality naturally holds:

$$QE(k_1\theta, a, b) \leq QE_2(k_1\theta, \frac{1}{k_1}, a, b) = QE_1(\theta, k_1, a, b) \leq QE(\theta, a, b). \tag{30}$$

where k_1 denotes $k_{\mu,\sigma,a,b}(\theta)$ for simplicty. Moreover, the equality holds if and only if $k_1 = 1$.

Lemma 4.7. *When $a \leq 0 \leq b,\ a < b$, there exists a unique $\theta_0 > 0$ such that $k_1 = 1$. When $0 < \theta < \theta_0$, $k_1 > 1$. When $\theta > \theta_0$, $0 < k_1 < 1$. We can further derive that for fixed a, b $QE(\theta, a, b)$ reaches the minimal when $\theta = \theta_0$.*

It's worth noticing that since $a \leq 0 \leq b,\ a < b$, and since a, b are arbitrary integers not necessarily related to n, all the generalized theorem above can be applied not only to optimization of the threshold of both H-MT and classical IF model, but also to the conversion from both $y = x$ and ReLU to spiking neuron.

Therefore, we can still optimize the threshold θ through iteration, as shown in Algorithm 1. The detailed proof of the theorems in this section can be found in Appendix E.

5 Experimental Results

In this section, we first evaluate the performance of our proposed method on ImageNet dataset (Deng et al. 2009) across different models, including ResNet18, ResNet34, and VGG-16bn, comparing our results with some previous ANN-to-SNN conversion methods. Then, we compute and analyze the energy consumption of the converted SNNs. Finally, we conduct comparative experiments to explore the influence of the number of thresholds in H-MT, check H-MT's performance of simulating identity mapping, and validate the effectiveness of generalized threshold optimization method as well as the communication mechanism.

5.1 Comparison with Previous ANN-to-SNN Conversion Methods

As shown in Table 1, we conducted CNN-to-SNN conversion with our H-MT on ImageNet (Deng et al. 2009) across ResNet18, ResNet34 and VGG-16bn, and compare the results with those of previous works. We denote the converted model as $model - n/M$, meaning that the multi-threshold neuron have n threshold and its threshold are chosen through method M. Here method **Opt** means our generalized threshold optimization method while method **999** means the traditional method of choosing the 99.9% quantile of ANN's activation value (denoted as $Ac_{99.9}$) as the threshold. To be precise, when using the **999** method, we normalize the minimum threshold of H-MT as $\lambda_{min} = Ac_{99.9}/(2^n - 1)$. The conversions are conducted basically through replacing ReLU with the Pos Neuron part of H-MT, since there are not many identity mappings in the 3 models involved. But we conduct experiments as well of H-MT substituting $y = x$ to check the neuron's performance with 16 inference timesteps, corresponding to $16^{\text{y=x}}$ in Table 1. H-MT here is equipped with communication mechanism.

From the results of experiments conducted with traditional threshold method, we can see that (Pos Neuron of) H-MT is of high precision due to its nicely designed structure with good mathematical property. Performance further improves with our generalized optimization method, highlighting the value of this approach. Moreover, the excellence of our method isn't fully demonstrated, even if DCGS from (Huang et al. 2025) outperforms our method, as they introduce a novel differential coding method with better precision to replace traditional rate coding, while we focus on modification of spiking neuron's structure. Besides, concrete data of experiments conducted on ImageNet (Deng et al. 2009) from other previous ANN-to-SNN conversion works based on design of spiking neuron structure is lacked for meaningful comparison.

5.2 Energy Estimation and Result Analysis

Accroding to (Horowitz 2014), the energy consumption ratio of the converted SNN relative to the ANN can be expressed by the following Equation,

$$\frac{E_{\text{SNN}}}{E_{\text{ANN}}} = \frac{MACs_{\text{SNN}} \times E_{\text{MAC}} + ACs_{\text{SNN}} \times E_{\text{AC}}}{MACs_{\text{ANN}} \times E_{\text{MAC}}}, \tag{31}$$

where $E_{\text{MAC}} = 4.6\text{pJ}$ and $E_{\text{AC}} = 0.9\text{pJ}$.

Since SNNs has almost no multiplication operations, we can regard $MACs_{\text{ANN}} >> MACs_{\text{SNN}}$. Thus, we only need

Method	Type	Arch.	Param.(M)	ANN Acc(%)	T	SNN Acc(%)
TS (Deng and Gu 2021)	CNN-to-SNN	VGG-16	138	72.40	64	70.97
SNM (Wang et al. 2022)	CNN-to-SNN	VGG-16	138	73.18	64	71.50
MMSE (Li et al. 2021)	CNN-to-SNN	ResNet-34	21.8	75.66	64	71.12
		VGG-16	138	75.36	64	70.69
QCFS (Bu et al. 2022b)	CNN-to-SNN	ResNet-34	21.8	74.32	64	72.35
		VGG-16	138	74.29	64	72.85
SRP (Hao et al. 2023)	CNN-to-SNN	ResNet-34	21.8	74.32	4, 64	66.71, 68.61
		VGG-16	138	74.29	4, 64	66.47, 69.43
GN (Lv et al. 2024)	CNN-to-SNN	ResNet-34	21.8	74.35	8, 32	73.57, 73.46
BSNN (Li, Zhao, and Zeng 2022)	CNN-to-SNN	ResNet-34	21.8	73.27	989	72.64
DCGS (Huang et al. 2025)	CNN-to-SNN	ResNet18	11.7	71.49	4, 8	70.07, 71.31
		ResNet34	21.8	76.42	4, 8	73.35, 76.04
		VGG16bn	138	73.25	4, 8	72.72, 73.17
H-MT **(Ours)**	CNN-to-SNN	ResNet18 -4/999	11.7	71.50	4, 16, $16^{y=x}$	67.66, 68.48, 66.78
		ResNet34 - 4/999	21.8	76.42	4, 16, $16^{y=x}$	71.91, 72.85,71.67
		VGG-16bn - 4/999	138	73.25	4, 16, $16^{y=x}$	72.19, 72.15, 71.78
		ResNet18 - 4/Opt	11.7	71.50	4, 16, $16^{y=x}$	69.02, 70.95, 67.61
		ResNet34 - 4/Opt	21.8	76.42	4, 16, $16^{y=x}$	73.25, 75.80, 74.86
		VGG-16bn - 4/Opt	138	73.25	4, 16, $16^{y=x}$	71.94, 72.92, 72.71
		ResNet18 - 8/Opt	11.7	71.50	**2**, **4**	**71.39**, **71.07**
		ResNet34 - 8/Opt	21.8	76.42	**2**, **4**	**76.276**, **76.168**
		VGG-16bn - 8/Opt	138	73.25	**2**, **4**	**73.242**, **72.452**

Table 1: Comparison between the proposed method and previous ANN-to-SNN conversion works on ImageNet dataset.

Model Config	Acc/Energy	Time-step T	
		2	4
ResNet34 - 8/Opt, Param:21.8M, Acc:76.42%	Acc	**76.276**	**76.168**
	Energy ratio	**0.2986**	**0.4703**
ResNet34 - 8/999, Param:21.8M, Acc:76.42%	Acc	72.908	72.908
	Energy ratio	0.5435	1.0861
ResNet34 - 8/999_2, Param:21.8M, Acc:76.42%	Acc	75.702	75.684
	Energy ratio	0.4622	0.9228

Table 2: Accuracy and energy ratio of H-MT(Ours) with $n = 8$ thresholds, of ResNet34 on ImageNet Dataset. Thresholds here are chosen through the generalized threshold optimization method, traditional policy and empirical modification of the traditional policy.

to estimate $\frac{ACs_{\text{SNN}}}{MACs_{\text{ANN}}}$, which in H-MT Neuron can be calculated by

$$\frac{ACs_{\text{SNN}}}{MACs_{\text{ANN}}} = \frac{\sum_{l=1}^{L}\sum_{i=1}^{N}\sum_{j=1}^{n}(s^l_{i,P,j}[t] + s^l_{i,N,j}[t])c^l_i}{\sum_{l=1}^{L}\sum_{i=1}^{N} c^l_i}, \quad (32)$$

where c^l_i represents the number of outgoing connections from the i-th neuron in layer l to the neurons in the next layer. Table 2 presents partial results of energy consumption ratio, and the detailed supplement results can be found in Appendix F.

As shown in Table 2, with $n = 8$ thresholds on ResNet34, the generalized optimization method (Opt) achieve less than 0.15% accuracy loss with low energy ratio within very low latency ($T = 2$), while the traditional policy (999) causes higher energy consumption and still remarkable accuracy loss. We assume this may be because that the threshold chosen by traditional policy isn't large enough, thus firing spikes that are of large number but insufficient to generate desired output value.

Therefore, we attempt another empirical modification of the traditional policy, denoted as 999_2, i.e. using twice the threshold obtained by traditional methods as the actual threshold. As the experiment results show, method 999 achieves higher accuracy and a little lower energy ratio in comparison with traditional policy, but clearly doesn't outperform the generalized optimization method on these two metrics.

5.3 Influence of the Number of Thresholds

As Table 3 shows, the effect of number of threshold is remarkable. There is evident accuracy gap between H-MT with $n = 3$ and $n = 4$ thresholds, which is understandable, since for $n = 3$ and $n = 4$, there are respectively 2^3

Model Config\Acc(%)	Time-step T			
	2	4	8	16
ResNet34 - 4/Opt, Param:21.8M, Acc:76.42%	71.73	73.25	75.02	75.80
ResNet18 - 4/Opt, Param:11.7M, Acc:71.50%	67.78	69.02	70.35	70.95
VGG16bn - 4/Opt, Param:138M, Acc:73.25%	70.78	71.94	73.03	72.92
ResNet34 - 3/Opt, Param:21.8M, Acc:76.42%	64.33	68.33	71.6	73.91
ResNet18 - 3/Opt, Param:11.7M, Acc:71.50%	61.33	65.38	67.95	69.43
VGG16bn - 3/Opt, Param:138M, Acc:73.25%	67.97	70.37	71.65	72.62

Table 3: Accuracy of H-MT(Ours) with $n = 4$ thresholds, and Accuracy of H-MT(Ours) with $n = 3$ thresholds, of different converted models on ImageNet Dataset. Thresholds here are chosen through the generalized threshold optimization method

and 2^4 possible output values at per time-step. It's also interesting that there is still notable accuracy gap between $n = 3$ with $T = 16$ and $n = 4$ with $T = 8$. Anyway, such results again emphasizes the high demands of ANN-to-SNN conversion on the precision of neurons. Besides, result from Table 2 validates H-MT's excellence with a bigger number of thresholds.

5.4 Effectiveness of H-MT's Communication Mechanism

As the results with $16^{\mathrm{y=x}}$ in Table 1 show, although there are very few identity modules directly used only at the very beginning of ResNet18/34 and VGG-16bn, replacing them with H-MT across these models can cause some performance decline. To magnify the effect of communication mechanism, we conduct conversion with H-MT on ViT-base framework. In order to prevent great accuracy loss, we will not replace all the identity modules. Instead, we follow the policy of (Huang et al. 2025). Before fed into convolution or linear layer, input will first be fed into spiking neuron simulating identity mapping and be turned into spiking form.

Vit-base	Time-step $T = 16$	
Method	Accuracy(%)	
H-MT - 4/Opt	w/ communication	74.942
	w/o communication	69.214
H-MT - 4/999	w/ communication	73.878
	w/o communication	63.482

Table 4: Accuracy of H-MT(Ours) with $n = 4$ thresholds with or without communication mechanism for simplified conversion of ViT-base. Thresholds here are chosen through either the generalized threshold optimization method or the traditional policy for comparison.

As shown in Table 4, the effect of communication is significant. There will be a great drop of accuracy without communication mechanism no matter what threshold policy applied. Besides, H-MT with the generalized optimization method outperforms H-MT with traditional policy a lot when communication mechanism is not in use. Therefore, both communication mechanism and the generalized optimization method are excellent.

5.5 Effectiveness of Generalized Threshold Optimization Method

As shown in Table 1, 3 and 4, H-MT achieves better conversion with the generalized threshold optimization method, especially when simulating ReLU or $y = x$ across ResNet18, ResNet34 and ViT-base. For example, ResNet34 after such conversion can keep an accuracy of 75.80% through 16 inference time-steps. Besides, results from Table 2 further validate the effectiveness of the generalized optimization method in enhancing accuracy and reducing energy ratio.

However, the generalized threshold optimization method doesn't overall outperform traditional method. When conducting conversion on VGG-16bn, we find out that with less inference time-steps, the traditional method has a little advantage, but is defeated with more inference time-steps.

From our point of view, such results are largely because the assumptions involved (Equation (19) and Assumption 4.2) are too strong and not so universal, but the generalized optimization method for threshold of spiking neuron is indeed of value and does provide a promising direction for future research of ANN-to-SNN conversion.

6 Conclusion

This article firstly introduces a design of a novel multi-threshold spiking neuron structure called Harmony Multi-threshold Spiking Neuron (H-MT) that simulates $y = x$. H-MT is an extend version of classical IF neuron, containing hybrid structure of a Pos Neuron and a Neg neuron, as well as a nova communication mechanism. It is of harmonious mathematical property which many multi-threshold spiking neuron don't enjoy and its structure is of potential to simulate ReLU with higher precision. To adapt to H-MT, it then includes a generalized optimization method for threshold of spiking neurons simulating $y = x$ or ReLU, which does further enhance the performance.

The methods proposed in this article of course have plenty of room for improvement. H-MT is not so of advantage in efficiency and energy consumption in exchange for its mathematical property, and its performance of simulating $y = x$ is in urgent need of further advancing. Meanwhile, the generalized optimization method may require lots of iterations to get an ideal threshold, and isn't generalized enough based on the assumptions. We do hope this article can provide guidance and inspiration for future research, thereby better addressing these challenges.

Acknowledgments

We would like to thank Yuanhong Tang for helpful discussion. This work was supported by the National Natural Science Foundation of China (U24B20140, 62422601 and 62302016), Beijing Municipal Science and Technology Program (Z241100004224004), Beijing Nova Program (20230484362, 20240484703), and State Key Laboratory of General Artificial Intelligence.

References

Bohnstingl, T.; Woźniak, S.; Pantazi, A.; and Eleftheriou, E. 2022. Online Spatio-Temporal Learning in Deep Neural Networks. *IEEE Transactions on Neural Networks and Learning Systems*.

Bu, T.; Ding, J.; Yu, Z.; and Huang, T. 2022a. Optimized Potential Initialization for Low-Latency Spiking Neural Networks. In *AAAI Conference on Artificial Intelligence*.

Bu, T.; Fang, W.; Ding, J.; DAI, P.; Yu, Z.; and Huang, T. 2022b. Optimal ANN-SNN Conversion for High-Accuracy and Ultra-Low-Latency Spiking Neural Networks. In *International Conference on Learning Representations*.

Bu, T.; Li, M.; and Yu, Z. 2025. Inference-Scale Complexity in ANN-SNN Conversion for High-Performance and Low-Power Applications. In *Computer Vision and Pattern Recognition Conference*.

Cao, Y.; Chen, Y.; and Khosla, D. 2015. Spiking Deep Convolutional Neural Networks for Energy-Efficient Object Recognition. *International Journal of Computer Vision*.

Davies, M.; Srinivasa, N.; Lin, T.-H.; Chinya, G.; Cao, Y.; Choday, S. H.; Dimou, G.; Joshi, P.; Imam, N.; Jain, S.; Liao, Y.; Lin, C.-K.; Lines, A.; Liu, R.; Mathaikutty, D.; McCoy, S.; Paul, A.; Tse, J.; Venkataramanan, G.; Weng, Y.-H.; Wild, A.; Yang, Y.; and Wang, H. 2018. Loihi: a Neuromorphic Manycore Processor with On-Chip Learning. *IEEE Micro*.

de G. Matthews, A. G.; Hron, J.; Rowland, M.; Turner, R. E.; and Ghahramani, Z. 2018. Gaussian Process Behaviour in Wide Deep Neural Networks. In *International Conference on Learning Representations*.

DeBole, M. V.; Taba, B.; Amir, A.; Akopyan, F.; Andreopoulos, A.; Risk, W. P.; Kusnitz, J.; Ortega Otero, C.; Nayak, T. K.; Appuswamy, R.; Carlson, P. J.; Cassidy, A. S.; Datta, P.; Esser, S. K.; Garreau, G. J.; Holland, K. L.; Lekuch, S.; Mastro, M.; McKinstry, J.; di Nolfo, C.; Paulovicks, B.; Sawada, J.; Schleupen, K.; Shaw, B. G.; Klamo, J. L.; Flickner, M. D.; Arthur, J. V.; and Modha, D. S. 2019. TrueNorth: Accelerating From Zero to 64 Million Neurons in 10 Years. *Computer*.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A Large-Scale Hierarchical Image Database. In *Computer Vision and Pattern Recognition*.

Deng, S.; and Gu, S. 2021. Optimal Conversion of Conventional Artificial Neural Networks to Spiking Neural Networks. In *International Conference on Learning Representations*.

Ding, J.; Yu, Z.; Tian, Y.; and Huang, T. 2021. Optimal ANN-SNN Conversion for Fast and Accurate Inference in Deep Spiking Neural Networks. In *International Joint Conference on Artificial Intelligence*.

Ding, J.; Zhang, J.; Huang, T.; Liu, J. K.; and Yu, Z. 2025. Assisting Training of Deep Spiking Neural Networks With Parameter Initialization. *IEEE Transactions on Neural Networks and Learning Systems*.

Duan, C.; Ding, J.; Chen, S.; Yu, Z.; and Huang, T. 2022. Temporal Effective Batch Normalization in Spiking Neural Networks. In *Advances in Neural Information Processing Systems*.

Fang, W.; Yu, Z.; Chen, Y.; Huang, T.; Masquelier, T.; and Tian, Y. 2021. Deep Residual Learning in Spiking Neural Networks. In *Advances in Neural Information Processing Systems*.

Gerstner, W.; Kistler, W. M.; Naud, R.; and Paninski, L. 2014. *Neuronal Dynamics: From Single Neurons to Networks and Models of Cognition*. Cambridge University Press.

Gygax, J.; and Zenke, F. 2025. Elucidating the Theoretical Underpinnings of Surrogate Gradient Learning in Spiking Neural Networks. *Neural Computation*.

Han, B.; Srinivasan, G.; and Roy, K. 2020. RMP-SNN: Residual Membrane Potential Neuron for Enabling Deeper High-Accuracy and Low-Latency Spiking Neural Network. In *Computer Vision and Pattern Recognition*.

Hao, Z.; Bu, T.; Ding, J.; Huang, T.; and Yu, Z. 2023. Reducing ANN-SNN Conversion Error through Residual Membrane Potential. In *AAAI Conference on Artificial Intelligence*.

Hao, Z.; Shi, X.; Liu, Y.; Yu, Z.; and Huang, T. 2024. LM-HT SNN: Enhancing the Performance of SNN to ANN Counterpart through Learnable Multi-hierarchical Threshold Model. In *Advances in Neural Information Processing Systems*.

Horowitz, M. 2014. 1.1 Computing's Energy Problem (and What We Can Do About It). In *IEEE International Solid-State Circuits Conference Digest of Technical Papers*.

Huang, Z.; Fang, W.; Bu, T.; Xue, P.; Hao, Z.; Liu, W.; Tang, Y.; Yu, Z.; and Huang, T. 2025. Differential Coding for Training-Free ANN-to-SNN Conversion. In *International Conference on Machine Learning*.

Huang, Z.; Shi, X.; Hao, Z.; Bu, T.; Ding, J.; Yu, Z.; and Huang, T. 2024. Towards High-performance Spiking Transformers from ANN to SNN Conversion. In *ACM International Conference on Multimedia*.

Jiang, Y.; Hu, K.; Zhang, T.; Gao, H.; Liu, Y.; Fang, Y.; and Chen, F. 2024. Spatio-Temporal Approximation: A Training-Free SNN Conversion for Transformers. In *International Conference on Learning Representations*.

Kim, J.; Kim, H.; Huh, S.; Lee, J.; and Choi, K. 2018. Deep Neural Networks with Weighted Spikes. *Neurocomputing*.

Li, Y.; Deng, S.; Dong, X.; Gong, R.; and Gu, S. 2021. A Free Lunch from ANN: Towards Efficient, Accurate Spiking Neural Networks Calibration. In *International Conference on Machine Learning*.

Li, Y.; Zhao, D.; and Zeng, Y. 2022. BSNN: Towards Faster and Better Conversion of Artificial Neural Networks to Spiking Neural Networks with Bistable Neurons. *Frontiers in Neuroscience*.

Luo, X.; Yao, M.; Chou, Y.; Xu, B.; and Li, G. 2024. Integer-Valued Training and Spike-Driven Inference Spiking Neural Network for High-Performance and Energy-Efficient Object Detection. In *European Conference on Computer Vision*.

Lv, L.; Fang, W.; Yuan, L.; and Tian, Y. 2024. Optimal ANN-SNN Conversion with Group Neurons. In *International Conference on Acoustics, Speech and Signal Processing*.

Maass, W. 1997. Networks of Spiking Neurons: The Third Generation of Neural Network models. *Neural Networks*.

Meng, Q.; Xiao, M.; Yan, S.; Wang, Y.; Lin, Z.; and Luo, Z.-Q. 2023. Towards Memory-and-Time-Efficient Backpropagation for Training Spiking Neural Networks. In *International Conference on Computer Vision*.

Merolla, P. A.; Arthur, J. V.; Alvarez-Icaza, R.; Cassidy, A. S.; Sawada, J.; Akopyan, F.; Jackson, B. L.; Imam, N.; Guo, C.; Nakamura, Y.; et al. 2014. A Million Spiking-neuron Integrated Circuit with a Scalable Communication Network and Interface. *Science*.

Neftci, E. O.; Mostafa, H.; and Zenke, F. 2019. Surrogate Gradient Learning in Spiking Neural Networks: Bringing the Power of Gradient-based Optimization to Spiking Neural Networks. *IEEE Signal Processing Magazine*.

Pei, J.; Deng, L.; Song, S.; Zhao, M.; Zhang, Y.; Wu, S.; Wang, G.; Zou, Z.; Wu, Z.; He, W.; et al. 2019. Towards Artificial General Intelligence with Hybrid Tianjic Chip Architecture. *Nature*.

Rueckauer, B.; Lungu, I.-A.; Hu, Y.; and Pfeiffer, M. 2016. Theory and Tools for the Conversion of Analog to Spiking Convolutional Neural Networks. *arXiv preprint arXiv:1612.04052*.

Shi, X.; Hao, Z.; and Yu, Z. 2024. Spikingresformer: Bridging Resnet and Vision Transformer in Spiking Neural Networks. In *Computer Vision and Pattern Recognition*.

Wang, X.; and Zhang, Y. 2023. MT-SNN: Enhance Spiking Neural Network with Multiple Thresholds. *arXiv preprint arXiv:2303.11127*.

Wang, Y.; Zhang, M.; Chen, Y.; and Qu, H. 2022. Signed Neuron with Memory: Towards Simple, Accurate and High-Efficient ANN-SNN Conversion. In *International Joint Conference on Artificial Intelligence*.

Wang, Z.; Fang, Y.; Cao, J.; Ren, H.; and Xu, R. 2025. Adaptive calibration: A unified conversion framework of spiking neural networks. In *AAAI Conference on Artificial Intelligence*.

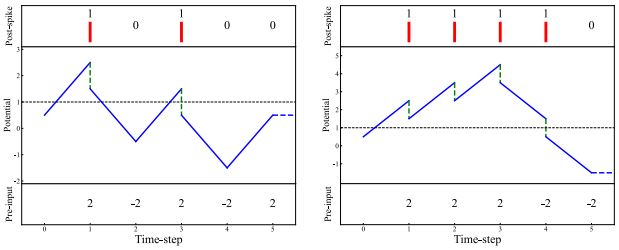
Xiao, M.; Meng, Q.; Zhang, Z.; He, D.; and Lin, Z. 2022. Online Training Through Time for Spiking Neural Networks. In *Advances in Neural Information Processing Systems*.

Zhao, L.; Huang, Z.; Ding, J.; and Yu, Z. 2025. TTFS-Former: A TTFS-Based Lossless Conversion of Spiking Transformer. In *International Conference on Machine Learning*.

Zhu, Y.; Ding, J.; Huang, T.; Xie, X.; and Yu, Z. 2024. Online Stabilization of Spiking Neural Networks. In *International Conference on Learning Representations*.

A Classical Example of Unevenness Error

Here is a classical case from (Bu et al. 2022b).



(a) Even case (b) Uneven case

Figure 3: The threshold is $\theta = 1$. For the even case, the input is $[2, -2, 2, -2, 2]$ and the average output is $\frac{2}{5}$ as there are 2 spikes, while for the uneven case input is $[2, 2, 2, -2, -2]$ and the average output is $\frac{4}{5}$ as there are 4 spikes.

Let's assume the total inference time-steps $T = 5$, the threshold is $\theta = 1$ and the initial membrane potential $v[0] = \frac{1}{2}$. The even input fed into IF neuron is $[2, -2, 2, -2, 2]$, while the uneven case is $[2, 2, 2, -2, -2]$. We can see that for both cases the average input is $\frac{2}{5}$, and we expect the average output of IF neuron is $\frac{2}{5}$ just as ReLU. As 3 shows, for the even case, the average output is luckily $\frac{2}{5}$, while for the unevenness case, the average output is $\frac{4}{5}$ as positive inputs assemble at the very beginning.

B Analysis of Structure of Multi-Threshold Neuron in Previous Works

Since there are various design of multi-threshold spiking neuron, we will only focus on one kind of structure. It is a simple extension of the IF neuron to simulate $y = x$, as well as an polished version of the combination of the multi-threshold neuron adopted in (Huang et al. 2025, 2024).

First, it has $2n$ thresholds:

$$\begin{aligned} &\boldsymbol{\lambda}_1^l = \boldsymbol{\theta}^l, \boldsymbol{\lambda}_2^l = \frac{\boldsymbol{\theta}^l}{2}, \dots, \boldsymbol{\lambda}_n^l = \frac{\boldsymbol{\theta}^l}{2^{n-1}}, \\ &\boldsymbol{\lambda}_{n+1}^l = -\boldsymbol{\theta}^l, \boldsymbol{\lambda}_{n+2}^l = -\frac{\boldsymbol{\theta}^l}{2}, \dots, \boldsymbol{\lambda}_{2n}^l = -\frac{\boldsymbol{\theta}^l}{2^{n-1}}, \end{aligned} \tag{33}$$

Its dynamics is as follows.Clearly this design of multi-threshold neuron share many similarities with the IF neuron, except the mechanism for emitting spikes, which won't affect the analysis.

$$\boldsymbol{m}^l[t] = \boldsymbol{v}^l[t-1] + \boldsymbol{u}^l[t] = \boldsymbol{v}^l[t-1] + f(\boldsymbol{x}^{l-1}[t]) \tag{34}$$

$$\boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{l}}[t] = MTH_{\theta,n}(\boldsymbol{m}^l[t], i) \tag{35}$$

$$\boldsymbol{x}^l[t] = \sum_{i=1}^{2n} \boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{l}}[t]\boldsymbol{\lambda}_i^l \tag{36}$$

$$\boldsymbol{v}^l[t] = \boldsymbol{m}^l[t] - \boldsymbol{x}^l[t], \tag{37}$$

$$MTH_{\boldsymbol{\theta},n}(\boldsymbol{x}, i) = \begin{cases} 1, & \text{if } i = \arg\min_p |\boldsymbol{x} - \boldsymbol{\lambda}_p| \\ 0, & \text{else} \end{cases} \tag{38}$$

We can derive almost the same equation as Equation (5), where n should be $2n$. And the extended strong assumption is:

$$\begin{cases} \boldsymbol{u}^l[t] \le -\boldsymbol{\theta}^l, \ \forall t = 1, \dots, T, & \text{if } \boldsymbol{z}^l \le -\boldsymbol{\theta}^l \\ \boldsymbol{u}^l[t] \in (-\boldsymbol{\theta}^l, 0), \ \forall t = 1, \dots, T, & \text{if } \boldsymbol{z}^l \in (-\boldsymbol{\theta}^l, 0) \\ \boldsymbol{u}^l[t] \in [0, \boldsymbol{\theta}^l), \ \forall t = 1, \dots, T, & \text{if } \boldsymbol{z}^l \in [0, \boldsymbol{\theta}^l) \\ \boldsymbol{u}^l[t] \ge \boldsymbol{\theta}^l, \ \forall t = 1, \dots, T, & \text{if } \boldsymbol{z}^l \ge \boldsymbol{\theta}^l \end{cases} \tag{39}$$

By dividing $\boldsymbol{\lambda}_{max}$, we have Equation (40). However, we find it hard to define whether $\frac{\boldsymbol{v}^l[T]}{\boldsymbol{\lambda}_{max}^l}$ is positive or negative though its range is limited to $(-1, 1)$. Besides, $\sum_{t=1}^{T}\sum_{i=1}^{2n} \boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{l}}[t]\frac{\boldsymbol{\lambda}_i^l}{\boldsymbol{\lambda}_{max}^l}$ isn't necessarily be an integer. These preventing us to further derive mathematical expressions similar to Equation (7) with whether a floor or ceiling function .

$$\frac{T\boldsymbol{\phi}^l(T)}{\boldsymbol{\lambda}_{max}^l} = \sum_{t=1}^{T}\sum_{i=1}^{2n} \boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{l}}[t]\frac{\boldsymbol{\lambda}_i^l}{\boldsymbol{\lambda}_{max}^l} = \frac{\boldsymbol{z}^l T + \boldsymbol{v}^l[0]}{\boldsymbol{\lambda}_{max}^l} - \frac{\boldsymbol{v}^l[T]}{\boldsymbol{\lambda}_{max}^l} \tag{40}$$

So naturally, let's try dividing $\boldsymbol{\lambda}_{min}$, and we have Equation (41). Now $\sum_{t=1}^{T}\sum_{i=1}^{2n} \boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{l}}[t]\frac{\boldsymbol{\lambda}_i^l}{\boldsymbol{\lambda}_{min}^l}$ is an integer but the range of $\frac{\boldsymbol{v}^l[T]}{\boldsymbol{\lambda}_{min}^l}$ is frustratingly not very under control.

$$\frac{T\boldsymbol{\phi}^l(T)}{\boldsymbol{\lambda}_{min}^l} = \sum_{t=1}^{T}\sum_{i=1}^{2n} \boldsymbol{s}_{\boldsymbol{i}}^{\boldsymbol{l}}[t]\frac{\boldsymbol{\lambda}_i^l}{\boldsymbol{\lambda}_{min}^l} = \frac{\boldsymbol{z}^l T + \boldsymbol{v}^l[0]}{\boldsymbol{\lambda}_{min}^l} - \frac{\boldsymbol{v}^l[T]}{\boldsymbol{\lambda}_{min}^l} \tag{41}$$

Besides, even we only use the positive part of such multi-threshold neuron to simulate ReLU, we will still encounter with similar problems, let alone many other design of multi-threshold spiking neurons.

C Mathematical Properties of Harmony Multi-Threshold Spiking Neuron

C.1 Derivation of H-MT's Mathematical Expression

Under the strong assumption (Equation (19)), it can be easily found that when the input z^l is non-positive, only the Neg Neuron can emit spikes through all inference time-steps, and that when the input z^l is positive, only the Pos Neuron will. Due to the symmetry between the Pos and Neg Neuron, we can first study the dynamics of Pos Neuron, then similarly have that of Neg neuron, and finally have H-MT's Mathematical Expression.

Similar to Equation (41), for Pos Neuron, we have:

$$\frac{T\boldsymbol{\phi}_P^l(T)}{\boldsymbol{\lambda}_{min}^l} = \sum_{t=1}^{T}\sum_{i=1}^{n} \boldsymbol{s}_{P,i}^{l}[t]\frac{\boldsymbol{\lambda}_i^l}{\boldsymbol{\lambda}_{min}^l} = \frac{\boldsymbol{z}^l T + \boldsymbol{v}_P^l[0]}{\boldsymbol{\lambda}_{min}^l} - \frac{\boldsymbol{v}_P^l[T]}{\boldsymbol{\lambda}_{min}^l} \tag{42}$$

Here $\frac{T\boldsymbol{\phi}_{pos}^l(T)}{\boldsymbol{\lambda}_{min}^l} = \sum_{t=1}^{T}\sum_{i=1}^{n} \boldsymbol{s}_{P,i}^{l}[t]\frac{\boldsymbol{\lambda}_i^l}{\boldsymbol{\lambda}_{min}^l}$ is an integer. As the Pos Neuron can emit at most n spikes per time-step, the maximum spike sum of it at single time-step is:

$$\max \boldsymbol{x}_P^l[t] = \sum_{i=1}^{n} \boldsymbol{\lambda}_i^l = \boldsymbol{\lambda}_{min}^l \sum_{i=1}^{n} 2^{i-1} = \boldsymbol{\lambda}_{min}^l(2^n - 1) \tag{43}$$

So when $\boldsymbol{z}^l \ge (2^n - 1)\boldsymbol{\lambda}_{min}^l$, under the assumption, H-MT will fire n spikes at every time-step and thus we have $\boldsymbol{\phi}_P^l(T) = \boldsymbol{\lambda}_{min}^l(2^n - 1)$. Similarly, when $\boldsymbol{z}^l \le 0$, under the

assumption, H-MT will fire no spike at every time-step and thus we have $\boldsymbol{\phi}_P^l(T) = 0$.

When $\boldsymbol{z}^l \in (0, (2^n - 1)\boldsymbol{\lambda}_{min}^l)$, since the thershold sequence $\{\boldsymbol{\lambda}_i\}_{i=1}^n$ is a geometric sequence with a common ratio of $\frac{1}{2}$, therefore for $\forall t = 1, 2, \ldots, T$, the membrane potential is fully released after firing and we have $\boldsymbol{v}_P^l[t] \in [0, \boldsymbol{\lambda}_{min})$ as well as $\frac{\boldsymbol{v}_P^l[t]}{\boldsymbol{\lambda}_{min}} \in [0, 1)$, which can be proved through simple mathematical induction starting with $\boldsymbol{v}_P^l[0] = \frac{\boldsymbol{\lambda}_{min}^l}{2}$.

We can now naturally further derive Equation (44) from Equation (42), since the range of $\frac{\boldsymbol{v}_{pos}^l[t]}{\boldsymbol{\lambda}_{min}}$ is under control.

$$\boldsymbol{\phi}_P^l(T) = \frac{\boldsymbol{\lambda}_{min}^l}{T} clip(\lfloor \frac{\boldsymbol{z}^l T + \frac{\boldsymbol{\lambda}_{min}^l}{2}}{\boldsymbol{\lambda}_{min}^l} \rfloor, 0, (2^n - 1)T) \tag{44}$$

With the symmetry between the Pos and Neg Neuron of H-MT, we naturally derive the mathematical expression of the Neg Neuron under the assumption as shown in Equation (45):

$$\boldsymbol{\phi}_N^l(T) = \frac{\boldsymbol{\lambda}_{min}^l}{T} clip(\lfloor \frac{(-\boldsymbol{z}^l) T + \frac{\boldsymbol{\lambda}_{min}^l}{2}}{\boldsymbol{\lambda}_{min}^l} \rfloor, 0, (2^n - 1)T) \tag{45}$$

We can transform the above expressions to a similar form of the expression used in (Huang et al. 2025) for comparison, as shown in Equation (46), where $N = 2^{n-1}T$ and $\boldsymbol{\theta} = 2^{n-1}\boldsymbol{\lambda}_{min}$. Meanwhile, we can derive a rough form of H-MT's mathematical expression under the assumption, as shown in Equation (48).

$$\boldsymbol{\phi}_P^l(T) = \frac{\boldsymbol{\theta}^l}{N} clip(\lfloor \frac{\boldsymbol{z}^l N + \frac{\boldsymbol{\theta}^l}{2}}{\boldsymbol{\theta}^l} \rfloor, 0, (2 - \frac{1}{2^{n-1}})N) \tag{46}$$

$$\boldsymbol{\phi}_N^l(T) = \frac{\boldsymbol{\theta}^l}{N} clip(\lfloor \frac{(-\boldsymbol{z}^l) T + \frac{\boldsymbol{\theta}^l}{2}}{\boldsymbol{\theta}^l} \rfloor, 0, (2 - \frac{1}{2^{n-1}})N) \tag{47}$$

$$\begin{aligned} \boldsymbol{\phi}^l(T) &= \boldsymbol{\phi}_P^l(T) - \boldsymbol{\phi}_N^l(T) \\ &= \begin{cases} \frac{\boldsymbol{\lambda}_{min}^l}{T} clip(\lfloor \frac{\boldsymbol{z}^l T + \frac{\boldsymbol{\lambda}_{min}^l}{2}}{\boldsymbol{\lambda}_{min}^l} \rfloor, 0, (2^n - 1)T), \\ \qquad \text{if } \boldsymbol{z}^l \geq 0 \\ -\frac{\boldsymbol{\lambda}_{min}^l}{T} clip(\lfloor \frac{(-\boldsymbol{z}^l) T + \frac{\boldsymbol{\lambda}_{min}^l}{2}}{\boldsymbol{\lambda}_{min}^l} \rfloor, 0, (2^n - 1)T), \\ \qquad \text{if } \boldsymbol{z}^l < 0 \end{cases} \end{aligned} \tag{48}$$

Before finally deriving the mathematical expression of H-MT, we need to introduce a simple Lemma.

Lemma C.1. *$f(x) = \lfloor x + \frac{1}{2} \rfloor$ is an odd function.*

Proof. Assume $x = \lfloor x \rfloor + q \geq 0$, where q denotes the fractional part. We analyze two cases based on the value of q:

1. **Case 1:** $0 \leq q < \frac{1}{2}$ By definition of f, we have:

$$f(x) = \left\lfloor \lfloor x \rfloor + q + \frac{1}{2} \right\rfloor = \lfloor x \rfloor$$

Since $x = \lfloor x \rfloor + q$, it follows:

$$-x = -\lfloor x \rfloor - q = \lfloor -x \rfloor + 1 + (-q)$$

Substituting into $f(-x)$:

$$f(-x) = \left\lfloor \lfloor -x \rfloor + 1 + (-q) + \frac{1}{2} \right\rfloor = \left\lfloor \lfloor -x \rfloor - q + \frac{3}{2} \right\rfloor \tag{49}$$

Since $0 \leq q \leq \frac{1}{2}$, we have $\frac{3}{2} - q \geq 1$, so the floor operation yields:

$$f(-x) = \lfloor -x \rfloor + 1 = -\lfloor x \rfloor = -f(x)$$

2. **Case 2:** $\frac{1}{2} \leq q < 1$ Similarly, by definition of f:

$$f(x) = \left\lfloor \lfloor x \rfloor + q + \frac{1}{2} \right\rfloor = \lfloor x \rfloor + 1$$

For $-x$, we again use $-x = \lfloor -x \rfloor + 1 + (-q)$. Substituting into $f(-x)$:

$$f(-x) = \left\lfloor \lfloor -x \rfloor + 1 + (-q) + \frac{1}{2} \right\rfloor = \left\lfloor \lfloor -x \rfloor - q + \frac{3}{2} \right\rfloor \tag{50}$$

Since $\frac{1}{2} < q < 1$, we have $\frac{3}{2} - q < 1$, so the floor operation yields:

$$f(-x) = \lfloor -x \rfloor = -(\lfloor x \rfloor + 1) = -f(x)$$

In both cases, $f(-x) = -f(x)$ for $x \geq 0$. Thus, f is an odd function. □

With Lemma C.1, we can now derive the final form of H-MT's mathematical expression, i.e. Equation (20), which is symmetric and harmonious, from the rough form, i.e. Equation (48).

C.2 Another Property of H-MT extended from (Bu et al. 2022b)

Since we adopt the setting from (Bu et al. 2022b) of having the initial membrane potential as half of the threshold, though we set $v^l[0] = \lambda_{min}^l/2$ instead of letting $v^l[0] = \theta^l/2 = \lambda_{max}^l/2$, we can extend Theorem 2 of (Bu et al. 2022b) to Lemma C.2.

Lemma C.2 reflects that under the special uniform distribution assumptions (and corresponding strong assumptions), the expected conversion error between the H-MT neuron and $y = x$ is zero.

Lemma C.2. *If the probability distribution of the random variable x satisfies one of the following conditions:*

1. *On each small interval $[m_t, m_{t+1}]$, x is uniformly distributed with probability density p_t, where $m_0 = 0$, $m_{Z+1} = \frac{Z}{T}\theta$, $m_t = \frac{(t-\frac{1}{2})\theta}{T}$ for $t = 1, 2, \ldots, Z$, and $p_0 = p_Z$; simultaneously, on each small interval $[w_{t+1}, w_t]$, x is uniformly distributed with probability density q_t, where $w_0 = 0$, $w_{S+1} = -\frac{S}{T}\theta$, $w_t = \frac{-(t-\frac{1}{2})\theta}{T}$ for $t = 1, 2, \ldots, S$, and $q_0 = q_S$. Here, T, Z, S are unrelated positive integers.*
2. *On each small interval $[m_t, m_{t+1}]$, x is uniformly distributed with probability density p_t, where $m_0 = 0$, $m_{Z+1} = \frac{Z}{T}\theta$, $m_t = \frac{(t-\frac{1}{2})\theta}{T}$ for $t = 1, 2, \ldots, Z$; simultaneously, on each small interval $[w_{t+1}, w_t]$, x is uniformly distributed with probability density q_t, where $w_0 = 0$, $w_{S+1} = -\frac{S}{T}\theta$, $w_t = \frac{-(t-\frac{1}{2})\theta}{T}$ for $t = 1, 2, \ldots, S$. Additionally, $q_0 = p_0$ and $q_S = p_Z$. Here, T, Z, S are unrelated positive integers.*

Then, the following holds:

$$\mathbb{E}_x\left(x-\frac{\theta}{T}\left\lfloor\frac{Tx}{\theta}+\frac{1}{2}\right\rfloor\right)=0$$

D Analysis of Communication Mechanism's Effect and Test Cases

D.1 Analysis of the Mechanism's Effects

It's clear that communication will compensate the membrane potential of Pos or Neg neuron when the other one fires spikes at one time-step, thus preventing the membrane potetial of both Pos and Neg neuron being too extreme for long time. But there is another effect that we can further derive, as Lemma D.1 shows.

Lemma D.1. *H-MT with communication mechanism will have only one of its Pos and Neg neuron to fire spikes at every time-step, as for* $\forall t=1,\cdots,T$ *we have:*

$$\boldsymbol{v}_P^l[t]+\boldsymbol{v}_N^l[t]=\boldsymbol{\lambda}_{min}^l \tag{51}$$

Proof. First, we have the initial setting:

$$\boldsymbol{v}_P^l[0]+\boldsymbol{v}_N^l[0]=\frac{\boldsymbol{\lambda}_{min}^l}{2}+\frac{\boldsymbol{\lambda}_{min}^l}{2}=\boldsymbol{\lambda}_{min}^l \tag{52}$$

Then with Equation (18), and (21), we have:

$$\begin{aligned}\boldsymbol{v}_P^l[1]+\boldsymbol{v}_N^l[1]=&\boldsymbol{v}_P^l[0]+\boldsymbol{u}^l[1]-\boldsymbol{x}_P^l[1]+\boldsymbol{x}_N^l[1]\\&+\boldsymbol{v}_N^l[0]+(-\boldsymbol{u}^l[1])-\boldsymbol{x}_N^l[1]+\boldsymbol{x}_P^l[1]\\=&\boldsymbol{v}_P^l[0]+\boldsymbol{v}_N^l[0]\\=&\boldsymbol{\lambda}_{min}^l\end{aligned} \tag{53}$$

Naturally, through simple recursion, we have:

$$\boldsymbol{v}_P^l[t]+\boldsymbol{v}_N^l[t]=\boldsymbol{v}_P^l[t-1]+\boldsymbol{v}_N^l[t-1]=\boldsymbol{\lambda}_{min}^l \tag{54}$$

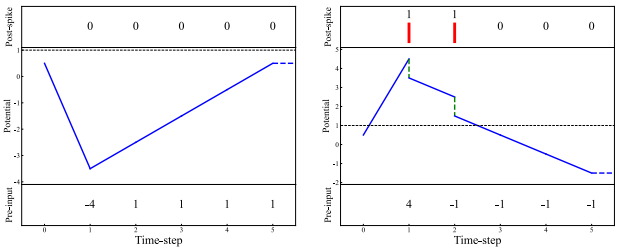
$$\boldsymbol{m}_P^l[t+1]+\boldsymbol{m}_N^l[t+1]=\boldsymbol{v}_P^l[t]+\boldsymbol{u}^l[t]+\boldsymbol{v}_N^l[t]+(-\boldsymbol{u}^l[t])=\boldsymbol{\lambda}_{min}^l \tag{55}$$

Since the Pos and Neg neuron are symmetric, let's assume at time-step t, the pos neuron is about to fire spikes. Then we have $\boldsymbol{m}_P^l[t]\geq\boldsymbol{\lambda}_{min}^l$. With Equation (55), we further derive that $\boldsymbol{m}_N^l[t]\leq\boldsymbol{0}$, which means the Neg neuron will not fire spikes at this time-step. □

Since there is up to only one neuron can fire spikes at each time-step, under the strong assumption, when z^l is nonnegative, all input $u^l[t]$ will be nonnegative, only Pos Neuron can fire spikes at each time-step and Neg neuron will keep silent though the communication mechanism will compensate Neg neuron's membrane potential, while when z^l is negative, things will be symmetric. Therefore, under the strong assumption, H-MT with communication mechanism can still keep its mathematical expression as Equation (20) describes.

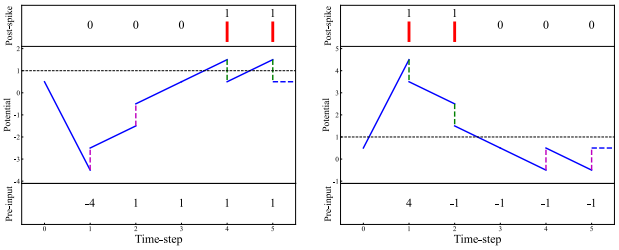
D.2 Test Cases

Here are some cases which shows the effect of the communication mechanism of H-MT, and where inputs are clearly uneven though unevenness is not so easy to define. In following cases, the membrane potential decrease caused by spike firing will be represented by green dashed lines, while the membrane potential increase induced by the communication mechanism will be denoted by magenta dashed lines.



(a) Pos Neuron (b) Neg Neuron

Figure 4: This is H-MT without communication and with only one threshold $\theta=1$. The input is $[-4,1,1,1,1]$. The pos neuron outputs 0 spikes while the neg neuron outputs 2 spikes. Therefore, the final average output is $-2/5$ instead of expected 0.



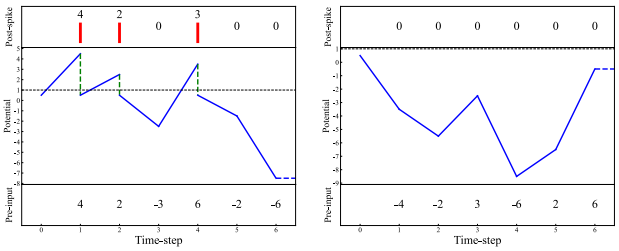
(a) Pos Neuron (b) Neg Neuron

Figure 5: This is H-MT with communication and with only one threshold $\theta=1$. The input is $[-4,1,1,1,1]$. The pos neuron outputs 2 spikes while the neg neuron outputs 2 spikes. Therefore, the final average output is 0 as expected.

For the first case, we use H-MT with only one threshold $\theta=1$, and the input is $[-4,1,1,1,1]$, which is obviously very uneven, as the extreme negative input is at the very beginning while the positive inputs assemble afterwards. The ideal output is obviously 0. However, the average output of H-MT without communication mechanism is $2/5$ while the average output of H-MT without communication mechanism is 0, as Figure 4 and 5 show.

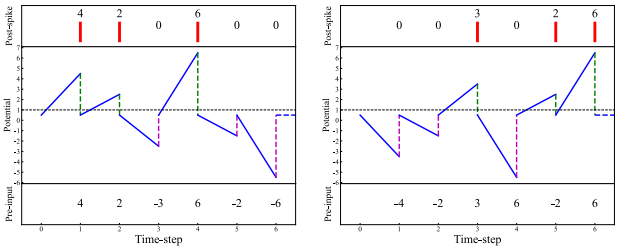
It can be seen that the extreme negative input -4 prevents the Pos Neuron to output spikes but causes the Neg Neuron to output spikes at the early stage. Without the mechanism, Pos Neuron will not output any spike. But with the mechanism, the membrane potential of the Pos Neuron can increase when the Neg Neuron outputs spikes, thus having the Pos Neuron output 2 spikes at the last 2 steps and decreasing the unevenness error.

For the second case, we use H-MT with with 3 threshold $[\lambda_1,\lambda_2,\ \lambda_3]=[4,2,1]$ (so the maximum output at single time-step is 7), and the input is $[4,2,-3,6,-2,-6]$, which is also hard to be called even. The ideal output is $1/6$. However, the average output of H-MT without communication mechanism is $9/6$ while the average output of H-MT with-



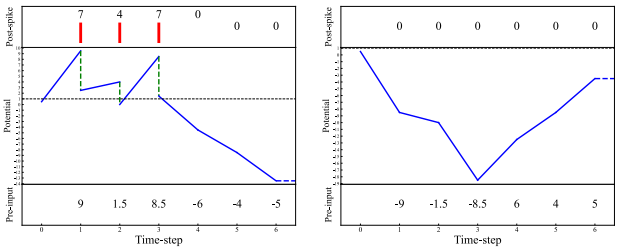
(a) Pos Neuron (b) Neg Neuron

Figure 6: This is H-MT without communication and with 3 threshold $[\lambda_1, \lambda_2, \lambda_3] = [4, 2, 1]$ (so the maximum output at single time-step is 7). The input is $[4, 2, -3, 6, -2, -6]$. The pos neuron outputs 9 in total while the neg neuron outputs 0. Therefore, the final average output is $9/6$ instead of expected $1/6$.



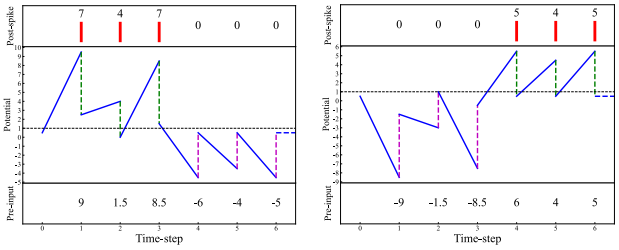
(a) Pos Neuron (b) Neg Neuron

Figure 7: This is H-MT without communication and with 3 threshold $[\lambda_1, \lambda_2, \lambda_3] = [4, 2, 1]$ (so the maximum output at single time-step is 7). The input is $[4, 2, -3, 6, -2, -6]$. The pos neuron outputs 12 in total while the neg neuron outputs 11. Therefore, the final average output is expected $1/6$. More communication occurs in this case.



(a) Pos Neuron (b) Neg Neuron

Figure 8: This is H-MT without communication and with 3 threshold $[\lambda_1, \lambda_2, \lambda_3] = [4, 2, 1]$ (so the maximum output at single time-step is 7). The input is $[9, , 1.5, 8.5, -6, -4, -5]$. The pos neuron outputs 18 in total while the neg neuron outputs 0. Therefore, the final average output is $18/6$ instead of expected $4/6$.



(a) Pos Neuron (b) Neg Neuron

Figure 9: This is H-MT without communication and with 3 threshold $[\lambda_1, \lambda_2, \lambda_3] = [4, 2, 1]$ (so the maximum output at single time-step is 7). The input is $[4, 2, -3, 6, -2, -6]$. The pos neuron outputs 18 in total while the neg neuron outputs 14. Therefore, the final average output is expected $4/6$. Much communication occurs in this case as well.

out communication mechanism is $1/6$, as Figure 6 and 7 show.

It can be seen that though the inputs are not so extreme, without the mechanism, Neg Neuron will keep silent all the time, but with the mechanism the Neg Neuron can better output the information it receives and even communicate with the Pos Neuron, leading to better average output. This can be again validated in the third case shown in Figure 6 and 7, whose input is even more complex and uneven.

However, when we apply the classical even and uneven case from (Bu et al. 2022b) to H-MT with only one threshold $\theta = 1$, something unexpected happens. For the uneven case, communication mechanism again decreases the unevenness error, but for the even case, the mechanism causes unevenness error as it causes the Neg Neuron to output spikes that should not exist, as shown in Figure 10 to 13 However, since the input and output is very complex and such even input is extremely rare when multi-threshold spiking neuron is in use, communication mechanism is still of value and may need some further modification.

To conclude, communication mechanism prevents the membrane potential of either the Pos Neuron or the Neg neuron to be extreme for long time when some uneven input occurs, thus leading to a better average output in general. Moreover, as the expected average output in the cases shown above are all nonnegative, the structure of H-MT with communication mechanism is of potential to better simulate ReLU, but of course more modification is needed to prevent the neuron from having negative average output and limit extra computational consumption.

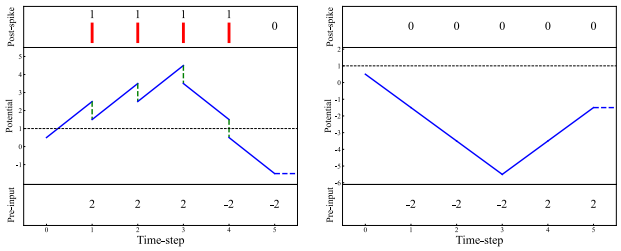
E Proof of Theorems in Section 4.2

E.1 Proof of Lemma 4.4

Before the proof, we need first introduce Lemma E.1 from (Huang et al. 2025).

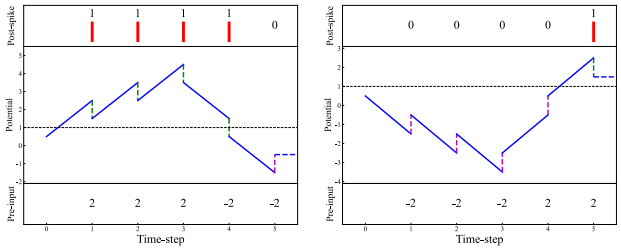
Lemma E.1.

$$\begin{aligned}\int_a^b (c-x)^2 e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx \; =& \sqrt{\tfrac{\pi}{2}}\sigma\left(\sigma^2+\mu^2-2c\mu+c^2\right) \\ & * \left(erf\left(\frac{(b-\mu)}{\sqrt{2}\sigma}\right) - erf\left(\frac{(a-\mu)}{\sqrt{2}\sigma}\right)\right) \\ & + \left(-\sigma^2\left(b+\mu-2c\right)\right) e^{-\frac{(b-\mu)^2}{2\sigma^2}} \\ & + \left(\sigma^2\left(a+\mu-2c\right)\right) e^{-\frac{(a-\mu)^2}{2\sigma^2}}\end{aligned} \tag{56}$$



(a) Pos Neuron (b) Neg Neuron

Figure 10: When we apply the classical uneven case from (Bu et al. 2022b) to H-MT with only one threshold $\theta = 1$ and without communication mechanism, the pos neuron still outputs 4 spikes while the neg neuron keeps silent. Therefore, the final average output is $4/5$ instead of expected $2/5$.



(a) Pos Neuron (b) Neg Neuron

Figure 11: When we apply the classical uneven case from (Bu et al. 2022b) to H-MT with only one threshold $\theta = 1$ and with communication mechanism, the pos neuron still outputs 4 spikes while the neg neuron outputs 1 spike. Therefore, the final average output is $3/5$ instead of expected $2/5$. The unevenness error still exist but gets decreased a bit.

Then we have:

Lemma E.2. ***(Repeated from Lemma 4.4)***

$$QE_1(\theta,k,a,b)=\int_{-\infty}^{\infty}\left(f_1(x,\theta,k,a,b)-x\right)^2 e^{-\frac{(x-\mu)^2}{2\sigma^2}}\,dx \tag{57}$$

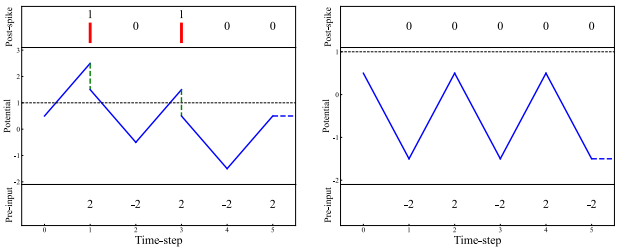
$$f_1(x,\theta,k,a,b)=k\frac{\theta}{n}clip(\lfloor\frac{nx+\frac{\theta}{2}}{\theta}\rfloor,\ a,\ b\ ),a,b\in\mathbb{Z},a<b \tag{58}$$

For any fixed integer a,b,n, any fixed $\theta>0,\mu,\sigma>0$, $QE_1(\theta,a,b)$ reaches the minimal value only when:

$$\begin{aligned}k=k_1=&\frac{\mu}{\theta}\frac{n\left[a+b-\sum_{i=a+1}^{b}erf\left(\frac{\frac{2i-1}{2n}\theta-\mu}{\sqrt{2}\sigma}\right)\right]}{a^2+b^2-\sum_{i=a+1}^{b}(2i-1)erf\left(\frac{\frac{2i-1}{2n}\theta-\mu}{\sqrt{2}\sigma}\right)}\\&+\frac{\sigma}{\sqrt{\frac{\pi}{2}}\theta}\frac{n\sum_{i=a+1}^{b}e^{-\frac{\left(\frac{2i-1}{2n}\theta-\mu\right)^2}{2\sigma^2}}}{a^2+b^2-\sum_{i=a+1}^{b}(2i-1)erf\left(\frac{\frac{2i-1}{2n}\theta-\mu}{\sqrt{2}\sigma}\right)}\end{aligned} \tag{59}$$

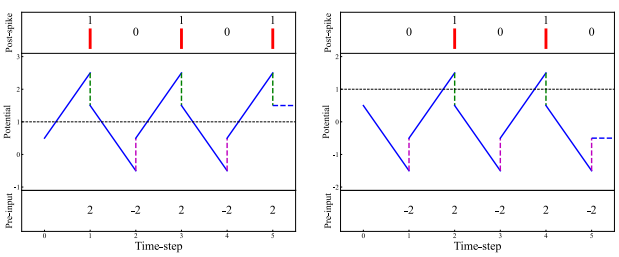
$erf(x)$ is a function monotonically increasing in x, to be precise:

$$\begin{aligned}&erf(x)=\frac{2}{\sqrt{\pi}}\int_0^x e^{-x^2}dx,\\&erf(-x)=-erf(x),erf(\infty)=1\end{aligned} \tag{60}$$



(a) Pos Neuron (b) Neg Neuron

Figure 12: When we apply the classical even case from (Bu et al. 2022b) to H-MT with only one threshold $\theta = 1$ and without communication mechanism, the pos neuron still outputs 2 spikes while the neg neuron keeps silent. Therefore, the final average output is expected $2/5$.



(a) Pos Neuron (b) Neg Neuron

Figure 13: When we apply the classical even case from (Bu et al. 2022b) to H-MT with only one threshold $\theta = 1$ and with communication mechanism, the pos neuron outputs 3 spikes while the neg neuron outputs 2 spikes. Therefore, the final average output is $1/5$ rather than expected $2/5$. The unevenness error unexpectedly occurs.

Then we have the proof.

Proof.

$$\begin{aligned}&QE_1(\theta,k,a,b)\\&=\int_{-\infty}^{\infty}\left(f_1(x,\theta,k,a,b)-x\right)^2 e^{-\frac{(x-\mu)^2}{2\sigma^2}}\,dx\\&=\int_{-\infty}^{\infty}(\frac{k\theta}{n}clip(\lfloor\frac{nx+\frac{\theta}{2}}{\theta}\rfloor,a,b)-x)^2e^{-\frac{(x-\mu)^2}{2\sigma^2}}\,dx\\&=\int_{-\infty}^{\frac{(2a-1)\theta}{2n}}(k\frac{\theta}{n}a-x)^2\,e^{-\frac{(x-\mu)^2}{2\sigma^2}}dx\\&\quad+\int_{\frac{(2a-1)\theta}{2n}}^{\frac{(2a+1)\theta}{2n}}(k\frac{\theta}{n}a-x)^2\,e^{-\frac{(x-\mu)^2}{2\sigma^2}}dx\\&\quad+\int_{\frac{(2a+1)\theta}{2n}}^{\frac{(2a+3)\theta}{2n}}(k\frac{\theta}{n}(a+1)-x)^2\,e^{-\frac{(x-\mu)^2}{2\sigma^2}}dx+\dots\\&\quad+\int_{\frac{(2b-3)\theta}{2n}}^{\frac{(2b-1)\theta}{2n}}(k\frac{\theta}{n}(b-1)-x)^2\,e^{-\frac{(x-\mu)^2}{2\sigma^2}}dx\\&\quad+\int_{\frac{(2b-1)\theta}{2n}}^{\infty}(k\frac{\theta}{n}b-x)^2\,e^{-\frac{(x-\mu)^2}{2\sigma^2}}dx\end{aligned} \tag{61}$$

$$
\begin{aligned}
&= \int_{-\infty}^{\frac{(2a-1)\theta}{2n}} (k\frac{\theta}{n}a - x)^2 \, e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx \\
&\quad + \sum_{i=a}^{b-1} \int_{\frac{(2i-1)\theta}{2n}}^{\frac{(2i+1)\theta}{2n}} (k\frac{\theta}{n}i - x)^2 \, e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx \\
&\quad + \int_{\frac{(2b-1)\theta}{2n}}^{\infty} (k\frac{\theta}{n}b - x)^2 \, e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx \\
&= \sqrt{\frac{\pi}{2}}\sigma(\sigma^2 + \mu^2 - 2k(\frac{\theta a}{n}\mu) + k^2(\frac{\theta a}{n})^2) \\
&\qquad * (erf(\frac{\frac{2a-1}{2n}\theta - \mu}{\sqrt{2}\sigma}) + 1) \\
&\quad + (-\sigma^2(\frac{2a-1}{2n}\theta + \mu - 2k\frac{\theta a}{n})e^{-\frac{(\frac{2a-1}{2n}\theta - \mu)^2}{2\sigma^2}}) \\
&\quad + \sum_{i=a}^{b-1}[\sqrt{\frac{\pi}{2}}\sigma(\sigma^2 + \mu^2 - 2k(\frac{\theta i}{n}\mu) + k^2(\frac{\theta i}{n})^2) \\
&\qquad * (erf(\frac{\frac{2i+1}{2n}\theta - \mu}{\sqrt{2}\sigma}) - erf(\frac{\frac{2i-1}{2n}\theta - \mu}{\sqrt{2}\sigma})) \\
&\quad + (-\sigma^2(\frac{2i+1}{2n}\theta + \mu - 2k\frac{\theta i}{n})e^{-\frac{(\frac{2i+1}{2n}\theta - \mu)^2}{2\sigma^2}}) \\
&\quad + (\sigma^2(\frac{2i-1}{2n}\theta + \mu - 2k\frac{\theta i}{n})e^{-\frac{(\frac{2i-1}{2n}\theta - \mu)^2}{2\sigma^2}})] \\
&\quad + \sqrt{\frac{\pi}{2}}\sigma(\sigma^2 + \mu^2 - 2k(\frac{\theta b}{n}\mu) + k^2(\frac{\theta b}{n})^2) \\
&\qquad * (1 - erf(\frac{\frac{2b-1}{2n}\theta - \mu}{\sqrt{2}\sigma})) \\
&\quad + (\sigma^2(\frac{2b-1}{2n}\theta + \mu - 2k\frac{\theta b}{n})e^{-\frac{(\frac{2b-1}{2n}\theta - \mu)^2}{2\sigma^2}})
\end{aligned}
$$

From Equation (E.2), we have that $QE_1(\theta, k, a, b)$ is a convex function of k. To be more concrete, it is a quadratic function in k with positive binomial coefficients, considering the properties of $erf(x)$, as shown in Equation (60).

Therefore, differentiating $QE_1(\theta, k, a, b)$ with respect to k, we obtain:

$$
\begin{aligned}
\frac{\partial QE_1}{\partial k} =& 2k\sigma\sqrt{\frac{\pi}{2}}(\frac{\theta}{n})^2 \, [a^2(erf(\frac{\frac{2a-1}{2n}\theta - \mu}{\sqrt{2}\sigma}) + 1) \\
&+ \sum_{i=a}^{b-1} i^2(erf(\frac{\frac{2i+1}{2n}\theta - \mu}{\sqrt{2}\sigma}) - erf(\frac{\frac{2i-1}{2n}\theta - \mu}{\sqrt{2}\sigma})) \\
&+ b^2(1 - erf(\frac{\frac{2b-1}{2n}\theta - \mu}{\sqrt{2}\sigma}))] \\
&- 2[\sqrt{\frac{\pi}{2}}\sigma\frac{\theta\mu}{n}(a(erf(\frac{\frac{2a-1}{2n}\theta - \mu}{\sqrt{2}\sigma}) + 1) \\
&+ \sum_{i=a}^{b-1} i(erf(\frac{\frac{2i+1}{2n}\theta - \mu}{\sqrt{2}\sigma}) - erf(\frac{\frac{2i-1}{2n}\theta - \mu}{\sqrt{2}\sigma})) \\
&+ b(1 - erf(\frac{\frac{2b-1}{2n}\theta - \mu}{\sqrt{2}\sigma}))) \\
&+ \sigma^2\frac{\theta}{n}(-a\, e^{-\frac{(\frac{2a-1}{2n}\theta - \mu)^2}{2\sigma^2}} \\
&+ \sum_{i=a}^{b-1} i(-e^{-\frac{(\frac{2i+1}{2n}\theta - \mu)^2}{2\sigma^2}} + e^{-\frac{(\frac{2i-1}{2n}\theta - \mu)^2}{2\sigma^2}}) \\
&+ b\, e^{-\frac{(\frac{2b-1}{2n}\theta - \mu)^2}{2\sigma^2}})]
\end{aligned} \tag{62}
$$

$$
\begin{aligned}
=& 2k\sigma\sqrt{\frac{\pi}{2}}(a^2 + b^2 - \sum_{i=a+1}^{b}(2i-1)erf(\frac{\frac{2i-1}{2n}\theta - \mu}{\sqrt{2}\sigma})) \\
&- 2[\sqrt{\frac{\pi}{2}}\sigma\frac{\theta\mu}{n}(a + b - \sum_{i=a+1}^{b} erf(\frac{\frac{2i-1}{2n}\theta - \mu}{\sqrt{2}\sigma})) \\
&+ \sigma^2\frac{\theta}{n}(\sum_{i=a+1}^{b} e^{-\frac{(\frac{2i-1}{2n}\theta - \mu)^2}{2\sigma^2}})]
\end{aligned}
$$

From $\frac{\partial QE_1}{\partial k} = 0$, $QE_1(\theta, k, a, b)$ attains its minimum value when:

$$
\begin{aligned}
k = k_1 =& \frac{\mu}{\theta}\frac{n[a + b - \sum_{i=a+1}^{b} erf(\frac{\frac{2i-1}{2n}\theta - \mu}{\sqrt{2}\sigma})]}{a^2 + b^2 - \sum_{i=a+1}^{b}(2i-1)erf(\frac{\frac{2i-1}{2n}\theta - \mu}{\sqrt{2}\sigma})} \\
&+ \frac{\sigma}{\sqrt{\frac{\pi}{2}}\theta}\frac{n\sum_{i=a+1}^{b} e^{-\frac{(\frac{2i-1}{2n}\theta - \mu)^2}{2\sigma^2}}}{a^2 + b^2 - \sum_{i=a+1}^{b}(2i-1)erf(\frac{\frac{2i-1}{2n}\theta - \mu}{\sqrt{2}\sigma})}
\end{aligned} \tag{63}
$$

□

E.2 Proof of Lemma 4.5

Lemma E.3. ***(Repeated from Lemma 4.5)***

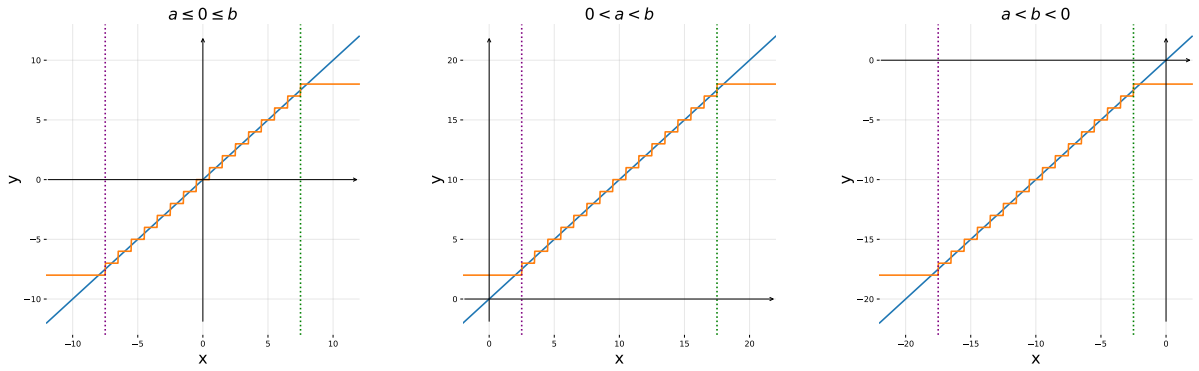
When $a \leq 0 \leq b$, $a < b$, for any $\theta > 0, \mu \in \mathbb{R}, \sigma > 0$, we will always have: $k_1 > 0$.

Proof. For $f(x, \theta, a, b) = \frac{\theta}{n} clip(\lfloor \frac{nx + \frac{\theta}{2}}{\theta} \rfloor, a, b), a, b \in \mathbb{Z}$, we have:

1. When $x \leq \frac{2a+1}{2n}\theta$, $f(x, \theta, a, b) = a$.
2. When $x \geq \frac{2b-1}{2n}\theta$, $f(x, \theta, a, b) = b$.

We now discuss the following three cases, which can be easily understood with Figure 14

1. When $a \leq 0 \leq b$: - For $x < \frac{2a+1}{2n}\theta$, - For $\frac{2a+1}{2n}\theta \leq x \leq \frac{2b-1}{2n}\theta$, and - For $x > \frac{2b-1}{2n}\theta$, the inequality $|x - f(x, \theta, a, b)| \leq |x - kf(x, \theta, a, b)|$ holds for all $k \leq 0$ but does not always hold with equality. Thus, $QE_1(\theta, 1, a, b) < QE_1(\theta, k, a, b)$ for all $k \leq 0$, which necessarily implies $k_1 > 0$ for all $\theta > 0, \mu \in \mathbb{R}, \sigma > 0, n \in \mathbb{N}$.
2. When $0 < a < b$: - For $x < \frac{2a+1}{2n}\theta$, the inequality $|x - f(x, \theta, a, b)| \leq |x - kf(x, \theta, a, b)|$ does not always hold for $k \leq 0$; - For $\frac{2a+1}{2n}\theta \leq x \leq \frac{2b-1}{2n}\theta$ and $x > \frac{2b-1}{2n}\theta$, the inequality holds for all $k \leq 0$ but does not always hold with equality. In extreme cases (e.g., μ is a negative number whose absolute value is extremely large relative to σ, and the probability distribution of x is concentrated around $x = \mu$), the k_1 calculated by Lemma 4.4 may be non-positive.
3. When $a < b < 0$: For $x > \frac{2b-1}{2n}\theta$, the inequality $|x - f(x, \theta, a, b)| \leq |x - kf(x, \theta, a, b)|$ does not always hold for $k \leq 0$; - For $\frac{2a+1}{2n}\theta \leq x \leq \frac{2b-1}{2n}\theta$ and $x < \frac{2a+1}{2n}\theta$, the inequality holds for all $k \leq 0$ but does not always hold with equality. In extreme cases (e.g., μ is a positive number whose absolute value is extremely large relative to σ, and the probability distribution of x is concentrated around $x = \mu$), the k_1 calculated by Lemma 4.4 may be non-positive.



(a) $a \leq 0 \leq b$ (b) $0 < a < b$ (c) $a < b < 0$

Figure 14: Three cases of Lemma 4.5

To conclude, When $a \leq 0 \leq b,\ a < b$, for any $\theta > 0, \mu \in \mathbb{R}, \sigma > 0$, we will always have: $k_1 > 0$. □

E.3 Proof of Lemma 4.6

Lemma E.4. ***(Repeated from Lemma 4.6)***

For any fixed integer a, b, n, any fixed $\theta > 0, \mu, \sigma > 0$, for all $k > 0$, $QE_2(\theta, a, b)$ reaches the minimal value only when: $k = 1$.

Then when $a \leq 0 \leq b,\ a < b$, we naturally have (equality holds if and only if when $k_1 = 1$) :

$$QE(k_1\theta, a, b) \leq QE_2(k_1\theta, \frac{1}{k_1}, a, b) = QE_1(\theta, k_1, a, b) \leq QE(\theta, a, b) \quad (64)$$

Proof.

$$\begin{aligned}
&QE_2(\theta, k, a, b)\\
=&\int_{-\infty}^{\infty} (f_2(x, \theta, k, a, b) - x)^2\, e^{-\frac{(x-\mu)^2}{2\sigma^2}}\, dx\\
=&\int_{-\infty}^{\infty} (\frac{\theta}{n} clip(\lfloor \frac{nx + k\frac{\theta}{2}}{k\theta} \rfloor,\ a,\ b\ ) - x)^2\, e^{-\frac{(x-\mu)^2}{2\sigma^2}}\, dx\\
=&\int_{-\infty}^{\frac{(2a-1)k\theta}{2n}} (\frac{\theta}{n}a - x)^2\, e^{-\frac{(x-\mu)^2}{2\sigma^2}}\, dx\\
&+\sum_{i=a}^{b-1} \int_{\frac{(2i-1)k\theta}{2n}}^{\frac{(2i+1)k\theta}{2n}} (\frac{\theta}{n}i - x)^2\, e^{-\frac{(x-\mu)^2}{2\sigma^2}}\, dx\\
&+\int_{\frac{(2b-1)k\theta}{2n}}^{\infty} (\frac{\theta}{n}b - x)^2\, e^{-\frac{(x-\mu)^2}{2\sigma^2}}\, dx
\end{aligned} \quad (65)$$

where:

$$\begin{aligned}
&\int_{-\infty}^{\frac{(2a-1)k\theta}{2n}} (\frac{\theta a}{n} - x)^2\, e^{-\frac{(x-\mu)^2}{2\sigma^2}}\, dx\\
=&\sqrt{\frac{\pi}{2}}\sigma(\sigma^2 + \mu^2 - 2\mu(\frac{\theta a}{n}) + (\frac{\theta a}{n})^2)\\
&*(erf(\frac{\frac{(2a-1)\theta}{2n}k - \mu}{\sqrt{2}\sigma}) + 1)\\
&+(-\sigma^2(\frac{(2a-1)\theta}{2n}k + \mu - \frac{2\theta a}{n}))\\
&*e^{-\frac{(\frac{(2a-1)\theta}{2n}k - \mu)^2}{2\sigma^2}}
\end{aligned} \quad (66)$$

$$\begin{aligned}
&\int_{\frac{(2i-1)k\theta}{2n}}^{\frac{(2i+1)k\theta}{2n}} (\frac{\theta i}{n} - x)^2\, e^{-\frac{(x-\mu)^2}{2\sigma^2}}\, dx\\
=&\sqrt{\frac{\pi}{2}}\sigma(\sigma^2 + \mu^2 - 2\mu(\frac{\theta i}{n}) + (\frac{\theta i}{n})^2)\\
&*(erf(\frac{\frac{(2i+1)\theta}{2n}k - \mu}{\sqrt{2}\sigma}) - erf(\frac{\frac{(2i-1)\theta}{2n}k - \mu}{\sqrt{2}\sigma}))\\
&+(-\sigma^2(\frac{(2i+1)\theta}{2n}k + \mu - \frac{2\theta i}{n}))*e^{-\frac{(\frac{(2i+1)\theta}{2n}k - \mu)^2}{2\sigma^2}}\\
&+(\sigma^2(\frac{(2i-1)\theta}{2n}k + \mu - \frac{2\theta i}{n}))*e^{-\frac{(\frac{(2i-1)\theta}{2n}k - \mu)^2}{2\sigma^2}}
\end{aligned} \quad (67)$$

$$\begin{aligned}
&\int_{\frac{(2b-1)k\theta}{2n}}^{\infty} (\frac{\theta b}{n} - x)^2\, e^{-\frac{(x-\mu)^2}{2\sigma^2}}\, dx\\
=&\sqrt{\frac{\pi}{2}}\sigma(\sigma^2 + \mu^2 - 2\mu(\frac{\theta b}{n}) + (\frac{\theta b}{n})^2)\\
&*(1 - erf(\frac{\frac{(2b-1)\theta}{2n}k - \mu}{\sqrt{2}\sigma}))\\
&+(\sigma^2(\frac{(2b-1)\theta}{2n}k + \mu - \frac{2\theta b}{n}))\\
&*e^{-\frac{(\frac{(2b-1)\theta}{2n}k - \mu)^2}{2\sigma^2}}
\end{aligned} \quad (68)$$

Combining these results, we obtain:

$$\begin{aligned}
&QE_2(\theta, k, a, b)\\
=&\sqrt{\frac{\pi}{2}}\sigma(\sigma^2 + \mu^2 - 2\mu(\frac{\theta a}{n}) + (\frac{\theta a}{n})^2)\\
&+\sqrt{\frac{\pi}{2}}\sigma(\sigma^2 + \mu^2 - 2\mu(\frac{\theta b}{n}) + (\frac{\theta b}{n})^2)\\
&+\sum_{i=a+1}^{b} erf(\frac{\frac{(2i-1)\theta}{2n}k - \mu}{\sqrt{2}\sigma})\sqrt{\frac{\pi}{2}}\sigma[(\sigma^2 + \mu^2 - 2\mu\frac{\theta(i-1)}{n}\\
&+(\frac{\theta(i-1)}{n})^2) - (\sigma^2 + \mu^2 - 2\mu\frac{\theta i}{n} + (\frac{\theta i}{n})^2)]\\
&+\sum_{i=a+1}^{b} \sigma^2(\frac{(2i-1)\theta}{2n}k + \mu - 2\frac{\theta i}{n})e^{-\frac{(\frac{(2i-1)\theta}{2n}k - \mu)^2}{2\sigma^2}}
\end{aligned} \quad (69)$$

$$
\begin{aligned}
&- \sigma^2(\frac{(2i-1)\theta}{2n}k + \mu - 2\frac{\theta(i-1)}{n})e^{-\frac{(\frac{(2i-1)\theta}{2n}k-\mu)^2}{2\sigma^2}} \\
=&\sqrt{\frac{\pi}{2}}\sigma(\sigma^2+\mu^2-2\mu(\frac{\theta a}{n})+(\frac{\theta a}{n})^2) \\
&+\sqrt{\frac{\pi}{2}}\sigma(\sigma^2+\mu^2-2\mu(\frac{\theta b}{n})+(\frac{\theta b}{n})^2) \\
&+\sum_{i=a+1}^{b} erf(\frac{\frac{(2i-1)\theta}{2n}k-\mu}{\sqrt{2}\sigma})\sqrt{\frac{\pi}{2}}\sigma[2\mu\frac{\theta}{n}+(\frac{\theta}{n})^2(-(2i-1))] \\
&+\sum_{i=a+1}^{b} \sigma^2(-\frac{2\theta}{n})e^{-\frac{(\frac{(2i-1)\theta}{2n}k-\mu)^2}{2\sigma^2}} \\
=&\sqrt{\frac{\pi}{2}}\sigma(2(\sigma^2+\mu^2)-2\mu(\frac{\theta(a+b)}{n})+(\frac{\theta}{n})^2(a^2+b^2)) \\
&+\sum_{i=a+1}^{b} \frac{2\theta}{n}\sigma[\sqrt{\frac{\pi}{2}}(\mu+(-\frac{(2i-1)\theta}{2n}))erf(\frac{\frac{(2i-1)\theta}{2n}k-\mu}{\sqrt{2}\sigma}) \\
&- \sigma e^{-\frac{(\frac{(2i-1)\theta}{2n}k-\mu)^2}{2\sigma^2}}]
\end{aligned}
$$

Extract the terms involving k, omitting the preceding positive coefficient $\frac{2\theta}{n}\sigma$, we have:

$$
\begin{aligned}
G(k) = \sum_{i=a+1}^{b} &(\sqrt{\frac{\pi}{2}}(\mu+(-\frac{(2i-1)\theta}{2n}))erf(\frac{\frac{(2i-1)\theta}{2n}k-\mu}{\sqrt{2}\sigma}) \\
&- \sigma e^{-\frac{(\frac{(2i-1)\theta}{2n}k-\mu)^2}{2\sigma^2}})
\end{aligned} \tag{70}
$$

Differentiating $G(k)$, we obtain:

$$
\begin{aligned}
\frac{\partial G}{\partial k} =& \sum_{i=a+1}^{b} \sqrt{\frac{\pi}{2}}(\mu+\frac{-(2i-1)\theta}{2n})\frac{2}{\sqrt{\pi}}e^{-\frac{(\frac{(2i-1)\theta}{2n}k-\mu)^2}{2\sigma^2}} * \frac{\frac{2i-1}{2n}\theta}{\sqrt{2}\sigma} \\
&+(-\sigma)e^{-\frac{(\frac{(2i-1)\theta}{2n}k-\mu)^2}{2\sigma^2}}(-2\frac{\frac{(2i-1)\theta}{2n}k-\mu}{\sqrt{2}\sigma}) * \frac{\frac{2i-1}{2n}\theta}{\sqrt{2}\sigma} \\
=& \sum_{i=a+1}^{b} (\frac{\frac{2i-1}{2n}\theta}{\sqrt{2}\sigma})(\sqrt{2}(\mu e^{-\frac{(\frac{(2i-1)\theta}{2n}k-\mu)^2}{2\sigma^2}} - \frac{2i-1}{2n}\theta e^{-\frac{(\frac{(2i-1)\theta}{2n}k-\mu)^2}{2\sigma^2}}) \\
&+\sqrt{2}(\frac{2i-1}{2n}\theta k-\mu)e^{-\frac{(\frac{(2i-1)\theta}{2n}k-\mu)^2}{2\sigma^2}}) \\
=& \sum_{i=a+1}^{b} (\frac{\frac{2i-1}{2n}\theta}{\sqrt{2}\sigma}) * \sqrt{2} * (\frac{2i-1}{2n})\theta(k-1)e^{-\frac{(\frac{(2i-1)\theta}{2n}k-\mu)^2}{2\sigma^2}} \\
=& \sum_{i=a+1}^{b} \frac{(2i-1)^2\theta^2}{4n^2\sigma}\theta^2(k-1)e^{-\frac{(\frac{(2i-1)\theta}{2n}k-\mu)^2}{2\sigma^2}}
\end{aligned} \tag{71}
$$

Therefore, we can find out that $\frac{\partial G}{\partial k} < 0$ when $0 < k < 1$, $\frac{\partial G}{\partial k} > 0$ for $k > 1$. Thus, $G(k)$ attains its global minimum at $k = 1$, and consequently, $QE_2(\theta, k, a, b)$ also achieves its minimum at $k = 1$.

Combining Lemma 4.4 and 4.5, we further obtain that when $a \le 0 \le b,\ a < b$, (equality holds if and only if when $k_1 = 1$):

$$
QE(k_1\theta, a, b) \le QE_2(k_1\theta, \frac{1}{k_1}, a, b) = QE_1(\theta, k_1, a, b) \le QE(\theta, a, b) \tag{72}
$$

□

E.4 Proof of Lemma 4.7

Lemma E.5. ***(Repeated from Lemma 4.7)***

When $a \le 0 \le b,\ a < b$, there exists a unique $\theta_0 > 0$ such that $k_1 = 1$. When $0 < \theta < \theta_0$, $k_1 > 1$. When $\theta > \theta_0$, $0 < k_1 < 1$. We can further derive that for fixed a, b $QE(\theta, a, b)$ reaches the minimal when $\theta = \theta_0$.

Proof. First, let's study when $k_1 = 1$. It's easy to notice that the denominator of k_1 corresponds to the coefficient of the quadratic term in $QE_1(\theta, k, a, b)$ with respect to k, which is strictly positive (considering the properties of the erf function, $\sigma > 0$, and $\theta > 0$ according to Lemma E.2). Furthermore, since $k_1 > 0$, we take the numerator of k_1 (divided by the coefficient n for simplicity) as:

$$
s(\theta) = \sqrt{\frac{\pi}{2}}\mu[a+b-\sum_{i=a+1}^{b} erf(\frac{\frac{2i-1}{2n}\theta-\mu}{\sqrt{2}\sigma}))] + \sigma\sum_{i=a+1}^{b} e^{-\frac{(\frac{2i-1}{2n}\theta-\mu)^2}{2\sigma^2}} \tag{73}
$$

Then we have $s(\theta) > 0,\ for\ \forall\theta > 0 (\because k_1 > 0)$ and can further derive:

$$
\begin{aligned}
s(0) =& \lim_{\theta\to 0^+} s(\theta) \\
=&\sqrt{\frac{\pi}{2}}\mu[a+b-\sum_{i=a+1}^{b} erf(\frac{-\mu}{\sqrt{2}\sigma}))] + \sigma\sum_{i=a+1}^{b} e^{-\frac{\mu^2}{2\sigma^2}} \\
\ge& 0
\end{aligned} \tag{74}
$$

Then denote the numerator minus the denominator of k_1 (both divided by the coefficient n for simplicity) as:

$$
\begin{aligned}
h(\theta) =& \sqrt{\frac{\pi}{2}}\mu[a+b-\sum_{i=a+1}^{b} erf(\frac{\frac{2i-1}{2n}\theta-\mu}{\sqrt{2}\sigma}))] + \sigma\sum_{i=a+1}^{b} e^{-\frac{(\frac{2i-1}{2n}\theta-\mu)^2}{2\sigma^2}} \\
&- \frac{\theta}{n}(a^2+b^2-\sum_{i=a+1}^{b}(2i-1)erf(\frac{\frac{2i-1}{2n}\theta-\mu}{\sqrt{2}\sigma}))
\end{aligned} \tag{75}
$$

We immediately arrive at $h(0) = s(0) \ge 0$. To be more concrete, we have $h(0) = s(0) > 0$. Here's a short proof. Since we have:

$$
\begin{aligned}
h(0) =& s(0) \\
=&\sqrt{\frac{\pi}{2}}\mu\left[a+b-\sum_{i=a+1}^{b}\text{erf}\left(\frac{-\mu}{\sqrt{2}\sigma}\right)\right] \\
&+\sigma\sum_{i=a+1}^{b} e^{-\frac{\mu^2}{2\sigma^2}} \\
=&\sigma\left(\sqrt{\pi}t\left[a+b+(b-a)\text{erf}(t)\right]+(b-a)e^{-t^2}\right) \\
&(\text{where } t = \frac{\mu}{\sqrt{2}\sigma})
\end{aligned} \tag{76}
$$

Let's denote $z(t) = \sqrt{\pi}t[a+b+(b-a)erf(t)] + (b-a)e^{-t^2}, t \in R$. Then differentiating $z(t)$, we obtain:

$$
\begin{aligned}
\frac{\partial z}{\partial t} =& \sqrt{\pi}(a+b+(b-a)erf(t)) \\
&+\sqrt{\pi}t((b-a)\frac{2}{\sqrt{\pi}}e^{-t^2}) \\
&+(b-a)e^{-t^2}(-2t) \\
=&\sqrt{\pi}(a+b+(b-a)erf(t))
\end{aligned} \tag{77}
$$

Since the error function $\text{erf}(t)$ is monotonically increasing on $\mathbb{R}$, we define $\text{erf}(t^*) = \frac{-(a+b)}{b-a}$ (where $b-a \ge |a+b| \ge 0,\ b-a > 0$, **since** $a \le 0 \le b,\ a < b$).Therefore, when $t < t^*$, $\frac{\partial z}{\partial t} < 0$, and when $t > t^*$, $\frac{\partial z}{\partial t} > 0$. Consequently, we obtain $z(t) > 0, \forall t \in \mathbb{R}$:

$$
\begin{aligned}
\min_t z(t) =& z(t^*) \\
=&\sqrt{\pi}t^*[a+b+(b-a)erf(t^*)] \\
&+(b-a)e^{-(t^*)^2} \\
=&(b-a)e^{-(t^*)^2} > 0
\end{aligned} \tag{78}
$$

It worth noticing that $z(t) > 0, \forall t \in \mathbb{R}$ still holds when $b-a = |a+b|$, which means $a = 0$, i.e. $t^* = -\inf$, or $b = 0$, i.e. $t^* = +\inf$. Therefore, we have $h(0) > 0$, for $\forall \mu \in \mathbb{R}, \sigma > 0$, when $a \leq 0 \leq b, a < b$.

Then consider consider the value of the function $h(\theta)$ as $\theta \to \infty$.:

$$\begin{aligned} h(+\infty) = \lim_{\theta\to+\infty} \sqrt{\frac{\pi}{2}}\mu[a+b-\sum_{i=a+1}^{b} erf(\frac{\frac{2i-1}{2n}\theta-\mu}{\sqrt{2}\sigma}))] \\ + \sigma \sum_{i=a+1}^{b} (e^{-\frac{(\frac{2i-1}{2n}\theta-\mu)^2}{2\sigma^2}} \\ - \frac{\theta}{n}(a^2+b^2-\sum_{i=a+1}^{b}(2i-1)erf(\frac{\frac{2i-1}{2n}\theta-\mu}{\sqrt{2}\sigma}))) \end{aligned} \tag{79}$$

It's easy to derive that:

$$\begin{aligned} &\lim_{\theta\to+\infty} \sqrt{\frac{\pi}{2}}\mu[a+b-\sum_{i=a+1}^{b} erf(\frac{\frac{2i-1}{2n}\theta-\mu}{\sqrt{2}\sigma}))] \\ &= \sqrt{\frac{\pi}{2}}\mu[a+b-\sum_{i=a+1}^{0} erf(-\infty) - \sum_{i=1}^{b} erf(\infty)] \\ &= \sqrt{\frac{\pi}{2}}\mu[a+b-(-a)*(-1)-b*1] \\ &= 0 \end{aligned} \tag{80}$$

$$\lim_{\theta\to+\infty} \sigma \sum_{i=a+1}^{b} e^{-\frac{(\frac{2i-1}{2n}\theta-\mu)^2}{2\sigma^2}} = \sigma \sum_{i=a+1}^{b} e^{-\infty} = 0 \tag{81}$$

Then consider:

$$\begin{aligned} &\theta(a^2+b^2-\sum_{i=a+1}^{b}(2i-1)erf(\frac{\frac{2i-1}{2n}\theta-\mu}{\sqrt{2}\sigma})) \\ =&\theta(a^2-\sum_{i=a+1}^{0}(2i-1)erf(\frac{\frac{2i-1}{2n}\theta-\mu}{\sqrt{2}\sigma})+b^2 \\ &-\sum_{i=1}^{b}(2i-1)erf(\frac{\frac{2i-1}{2n}\theta-\mu}{\sqrt{2}\sigma})) \\ =&\theta(a^2-\sum_{j=0}^{-(a+1)}(2(-j)-1)erf(\frac{\frac{2(-j)-1}{2n}\theta-\mu}{\sqrt{2}\sigma})+b^2 \\ &-\sum_{i=1}^{b}(2i-1)erf(\frac{\frac{2i-1}{2n}\theta-\mu}{\sqrt{2}\sigma})) \\ =&\theta(a^2-\sum_{j=0}^{-(a+1)}(2j+1)erf(\frac{\frac{2j+1}{2n}\theta+\mu}{\sqrt{2}\sigma})+b^2 \\ &-\sum_{i=1}^{b}(2i-1)erf(\frac{\frac{2i-1}{2n}\theta-\mu}{\sqrt{2}\sigma})) \\ =&\theta(a^2-\sum_{j=1}^{-a}(2j-1)erf(\frac{\frac{2j-1}{2n}\theta+\mu}{\sqrt{2}\sigma})+b^2 \\ &-\sum_{i=1}^{b}(2i-1)erf(\frac{\frac{2i-1}{2n}\theta-\mu}{\sqrt{2}\sigma})) \end{aligned} \tag{82}$$

As for $\theta(a^2-\sum_{j=1}^{-a}(2j-1)erf(\frac{\frac{2j-1}{2n}\theta+\mu}{\sqrt{2}\sigma}))$, we notice that:

$$\lim_{\theta\to+\infty} \theta = +\infty \tag{83}$$

$$\lim_{\theta\to+\infty} a^2-\sum_{j=1}^{-a}(2j-1)erf(\frac{\frac{2j-1}{2n}\theta+\mu}{\sqrt{2}\sigma}) \tag{84}$$

$$= a^2-\sum_{j=1}^{-a}(2j-1)erf(+\infty) = 0$$

$$a^2-\sum_{j=1}^{-a}(2j-1)erf(\frac{\frac{2j-1}{2n}\theta+\mu}{\sqrt{2}\sigma}) > a^2-\sum_{j=1}^{-a}(2j-1)*1 = 0 \tag{85}$$

$$\begin{aligned} &\because \sum_{j=1}^{-a}(2j-1)erf(\frac{\frac{2j-1}{2n}\theta+\mu}{\sqrt{2}\sigma}) \in \\ &[\sum_{j=1}^{-a}(2j-1)erf(\frac{\frac{1}{2n}\theta+\mu}{\sqrt{2}\sigma}), \sum_{j=1}^{-a}(2j-1)erf(\frac{\frac{2(-a)-1}{2n}\theta+\mu}{\sqrt{2}\sigma})] \\ &\therefore \lim_{\theta\to+\infty}\theta(a^2-\sum_{j=1}^{-a}(2j-1)erf(\frac{\frac{2j-1}{2n}\theta+\mu}{\sqrt{2}\sigma})) \in \\ &\quad [\lim_{\theta\to+\infty}\theta(a^2-\sum_{j=1}^{-a}(2j-1)erf(\frac{\frac{2(-a)-1}{2n}\theta+\mu}{\sqrt{2}\sigma})), \\ &\quad \lim_{\theta\to+\infty}\theta(a^2-\sum_{j=1}^{-a}(2j-1)erf(\frac{\frac{1}{2n}\theta+\mu}{\sqrt{2}\sigma}))] \end{aligned} \tag{86}$$

For $\forall k > 0$, we have:

$$\begin{aligned} &\lim_{\theta\to+\infty}\theta(a^2-\sum_{j=1}^{-a}(2j-1)erf(\frac{k\theta+\mu}{\sqrt{2}\sigma})) \\ &= \lim_{p\to0^+}\frac{a^2-\sum_{j=1}^{-a}(2j-1)erf(\frac{k\frac{1}{p}+\mu}{\sqrt{2}\sigma})}{p} \\ &\text{(by L'Hospital's Rule)} \\ &= \lim_{p\to0^+}\frac{-a^2\sqrt{\frac{2}{\pi}}e^{-\frac{(k\frac{1}{p}+\mu)^2}{2\sigma^2}}\frac{k}{\sqrt{2}\sigma}(-\frac{1}{p^2})}{1} \\ &= \lim_{p\to0^+}\frac{e^{-\frac{(k\frac{1}{p}+\mu)^2}{2\sigma^2}}}{p^2}*\frac{a^2k\sqrt{2}}{\sigma\sqrt{\pi}} \\ &= \lim_{p\to0^+}\frac{\frac{1}{p^2}}{e^{\frac{(k\frac{1}{p}+\mu)^2}{2\sigma^2}}}*\frac{a^2k\sqrt{2}}{\sigma\sqrt{\pi}} \\ &= \lim_{t\to+\infty}\frac{t^2}{e^{\frac{(kt+\mu)^2}{2\sigma^2}}}*\frac{a^2k\sqrt{2}}{\sigma\sqrt{\pi}} \\ &\text{(by L'Hospital's Rule)} \\ &= \lim_{t\to+\infty}\frac{2t}{e^{\frac{(kt+\mu)^2}{2\sigma^2}}*\frac{(kt+\mu)k}{\sigma^2}}*\frac{a^2k\sqrt{2}}{\sigma\sqrt{\pi}} \\ &= \lim_{t\to+\infty}\frac{2\sigma^2}{e^{\frac{(kt+\mu)^2}{2\sigma^2}}*k^2}*\frac{a^2k\sqrt{2}}{\sigma\sqrt{\pi}} \\ &=0 \end{aligned} \tag{87}$$

Then we arrive at:

$$\begin{aligned} &\lim_{\theta\to+\infty}\theta(a^2-\sum_{j=1}^{-a}(2j-1)erf(\frac{\frac{1}{2n}\theta+\mu}{\sqrt{2}\sigma})) \\ &= \lim_{\theta\to+\infty}\theta(a^2-\sum_{j=1}^{-a}(2j-1)erf(\frac{\frac{-2a-1}{2n}\theta+\mu}{\sqrt{2}\sigma})) = 0 \end{aligned} \tag{88}$$

$$\therefore \lim_{\theta\to+\infty} \theta(a^2 - \sum_{j=1}^{-a}(2j-1)erf(\frac{\frac{2j-1}{2n}\theta+\mu}{\sqrt{2}\sigma})) = 0 \tag{89}$$

Similarly, we have:

$$\lim_{\theta\to+\infty} \theta(b^2 - \sum_{i=1}^{b}(2i-1)erf(\frac{\frac{2i-1}{2n}\theta-\mu}{\sqrt{2}\sigma})) = 0 \tag{90}$$

Therefore:

$$\lim_{\theta\to+\infty} \theta(a^2+b^2 - \sum_{i=a+1}^{b}(2i-1)erf(\frac{\frac{2i-1}{2n}\theta-\mu}{\sqrt{2}\sigma})) = 0 \tag{91}$$

$$\begin{aligned} h(+\infty) =& \lim_{\theta\to+\infty} \sqrt{\frac{\pi}{2}}\mu[a+b-\sum_{i=a+1}^{b} erf(\frac{\frac{2i-1}{2n}\theta-\mu}{\sqrt{2}\sigma}))] \\ &+ \sigma\sum_{i=a+1}^{b} e^{-\frac{(\frac{2i-1}{2n}\theta-\mu)^2}{2\sigma^2}} \\ &- \frac{\theta}{n}(a^2+b^2-\sum_{i=a+1}^{b}(2i-1)erf(\frac{\frac{2i-1}{2n}\theta-\mu}{\sqrt{2}\sigma})) \\ =&0 \end{aligned} \tag{92}$$

Now we have obtained $h(0) > 0$ and $h(+\infty) = 0$. We further examine the first-order derivative of $h(\theta)$ with respect to θ:

$$\begin{aligned} \frac{\partial h}{\partial\theta} =& \sqrt{\frac{\pi}{2}}\mu(-\sum_{i=a+1}^{b}\frac{2}{\sqrt{pi}}e^{-\frac{(\frac{2i-1}{2n}\theta-\mu)^2}{2\sigma^2}} * \frac{2i-1}{2n\sqrt{2}\sigma}) \\ &+ \sigma\sum_{i=a+1}^{b} e^{-\frac{(\frac{2i-1}{2n}\theta-\mu)^2}{2\sigma^2}} * \frac{-2(\frac{2i-1}{2n}\theta-\mu)\frac{2i-1}{2n}}{2\sigma^2} \\ &- \frac{a^2+b^2}{n}\sqrt{\frac{\pi}{2}} + \frac{\sqrt{\frac{\pi}{2}}}{n}\sum_{i=a+1}^{b}(2i-1)erf(\frac{\frac{2i-1}{2n}\theta-\mu}{\sqrt{2}\sigma}) \\ &+ \frac{\sqrt{\frac{\pi}{2}}}{n}\sum_{i=a+1}^{b}\theta(2i-1)\frac{2}{\sqrt{\pi}}e^{-\frac{(\frac{2i-1}{2n}\theta-\mu)^2}{2\sigma^2}}\frac{\frac{2i-1}{2n}}{\sqrt{2}\sigma} \\ =& -\frac{a^2+b^2}{n}\sqrt{\frac{\pi}{2}} + \frac{\sqrt{\frac{\pi}{2}}}{n}\sum_{i=a+1}^{b}(2i-1)erf(\frac{\frac{2i-1}{2n}\theta-\mu}{\sqrt{2}\sigma}) \\ &+ \sum_{i=a+1}^{b} e^{-\frac{(\frac{2i-1}{2n}\theta-\mu)^2}{2\sigma^2}} * \frac{2i-1}{2n\sigma}(-\mu+\frac{2i-1}{n}\theta-(\frac{2i-1}{2n}\theta-\mu)) \\ =& \frac{1}{n}[-(a^2+b^2)\sqrt{\frac{\pi}{2}} + \sqrt{\frac{\pi}{2}}\sum_{i=a+1}^{b}(2i-1)erf(\frac{\frac{2i-1}{2n}\theta-\mu}{\sqrt{2}\sigma}) \\ &+ \theta\sum_{i=a+1}^{b}\frac{(2i-1)^2}{4n\sigma}e^{-\frac{(\frac{2i-1}{2n}\theta-\mu)^2}{2\sigma^2}}] \end{aligned} \tag{93}$$

Taking the second derivative, we obtain:

$$\begin{aligned} \frac{\partial^2 h}{\partial\theta^2} =& \frac{1}{n}[\sqrt{\frac{\pi}{2}}\sum_{i=a+1}^{b}(2i-1)\frac{2}{\sqrt{\pi}}e^{-\frac{(\frac{2i-1}{2n}\theta-\mu)^2}{2\sigma^2}} * \frac{2i-1}{2n\sqrt{2}\sigma} \\ &+ \sum_{i=a+1}^{b}\frac{(2i-1)^2}{4n\sigma}e^{-\frac{(\frac{2i-1}{2n}\theta-\mu)^2}{2\sigma^2}} \\ &+ \theta\sum_{i=a+1}^{b}\frac{(2i-1)^2}{4n\sigma}e^{-\frac{(\frac{2i-1}{2n}\theta-\mu)^2}{2\sigma^2}} * \frac{-2(\frac{2i-1}{2n}\theta-\mu)}{2\sigma^2}\frac{2i-1}{2n}] \\ =& \frac{1}{4n^2\sigma}\sum_{i=a+1}^{b}(2i-1)^2 e^{-\frac{(\frac{2i-1}{2n}\theta-\mu)^2}{2\sigma^2}}(3-\theta(\frac{\frac{2i-1}{2n}\theta-\mu}{\sigma^2})(\frac{2i-1}{2n})) \end{aligned} \tag{94}$$

$$\frac{\partial^2 h}{\partial\theta^2}(0) = \frac{1}{4n^2\sigma}\sum_{i=a+1}^{b}(2i-1)^2 * 3e^{-\frac{\mu^2}{2\sigma^2}} > 0 \tag{95}$$

$$\begin{aligned} \frac{\partial^2 h}{\partial\theta^2}(+\infty) =& \lim_{\theta\to+\infty}\frac{1}{4n^2\sigma}\sum_{i=a+1}^{b}(2i-1)^2 \\ &* (3e^{-\frac{(\frac{2i-1}{2n}\theta-\mu)^2}{2\sigma^2}} - \theta(\frac{\frac{2i-1}{2n}\theta-\mu}{\sigma^2})e^{-\frac{(\frac{2i-1}{2n}\theta-\mu)^2}{2\sigma^2}}) \end{aligned} \tag{96}$$

Since $e^{-\frac{\left(\frac{2i-1}{2n}\theta-\mu\right)^2}{2\sigma^2}} > 0$, as $\theta \to +\infty$, this term will eventually decrease and tend to 0. Meanwhile, the expression $3 - \theta\frac{\frac{2i-1}{2n}\theta-\mu}{\sigma^2}\frac{2i-1}{2n}$ equals 3 when $\theta = 0$, and as $\theta \to +\infty$, it will eventually decrease and tend to $-\infty$.

Therefore, with $\frac{\partial^2 h}{\partial\theta^2}(0) > 0$ by 95, $\frac{\partial^2 h}{\partial\theta^2}$ decreases to 0 as θ increases, then further decreases to negative values, and remains negative thereafter. Ultimately we have $\frac{\partial^2 h}{\partial\theta^2} \to 0^-$ as $\theta \to +\infty$.

Thus, $\frac{\partial h}{\partial\theta}$ first increases and then decreases. Additionally:

$$\begin{aligned} \frac{\partial h}{\partial\theta}(0) =& \frac{1}{n}[-(a^2+b^2)\sqrt{\frac{\pi}{2}} + \sqrt{\frac{\pi}{2}}\sum_{i=a+1}^{b}(2i-1)erf(\frac{-\mu}{\sqrt{2}\sigma}) \\ &+ 0 * \sum_{i=a+1}^{b}\frac{(2i-1)^2}{4n\sigma}e^{-\frac{\mu^2}{2\sigma^2}}] \\ =& \frac{1}{n}[\sqrt{\frac{\pi}{2}}(-a^2 + \sum_{i=a+1}^{0}(2i-1)erf(\frac{-\mu}{\sqrt{2}\sigma}) \\ &- b^2 + \sum_{i=1}^{b}(2i-1)erf(\frac{-\mu}{\sqrt{2}\sigma})] \\ =& \frac{1}{n}[\sqrt{\frac{\pi}{2}}(-a^2 + \sum_{j=1}^{-a}(2j-1)erf(\frac{\mu}{\sqrt{2}\sigma}) \\ &- b^2 + \sum_{i=1}^{b}(2i-1)erf(\frac{-\mu}{\sqrt{2}\sigma})] \\ <& \frac{1}{n}[\sqrt{\frac{\pi}{2}}(-a^2 + \sum_{j=1}^{-a}(2j-1)*1 - b^2 + \sum_{i=1}^{b}(2i-1)*1] = 0 \end{aligned} \tag{97}$$

$$\begin{aligned} \frac{\partial h}{\partial\theta}(+\infty) =& \lim_{\theta\to+\infty}\frac{1}{n}[-(a^2+b^2)\sqrt{\frac{\pi}{2}} \\ &+ \sqrt{\frac{\pi}{2}}\sum_{i=a+1}^{b}(2i-1)erf(\frac{\frac{2i-1}{2n}\theta-\mu}{\sqrt{2}\sigma}) \\ &+ \theta\sum_{i=a+1}^{b}\frac{(2i-1)^2}{4n\sigma}e^{-\frac{(\frac{2i-1}{2n}\theta-\mu)^2}{2\sigma^2}}] \\ =& \lim_{\theta\to+\infty}\frac{1}{n}[\sqrt{\frac{\pi}{2}}(-a^2 + \sum_{j=1}^{-a}(2j-1)erf(\frac{\frac{2j-1}{2n}\theta+\mu}{\sqrt{2}\sigma}) \\ &- b^2 + \sum_{i=1}^{b}(2i-1)erf(\frac{\frac{2i-1}{2n}\theta-\mu}{\sqrt{2}\sigma})) \\ &+ \theta\sum_{i=a+1}^{b}\frac{(2i-1)^2}{4n\sigma}e^{-\frac{(\frac{2i-1}{2n}\theta-\mu)^2}{2\sigma^2}}] \\ =&0 \end{aligned} \tag{98}$$

Therefore, with $\frac{\partial h}{\partial\theta}(0) < 0$, the first-order derivative $\frac{\partial h}{\partial\theta}$ first increases, then decreases as $t \to +\infty$ with $\frac{\partial h}{\partial\theta}(+\infty) = 0$. In other words, $\frac{\partial h}{\partial\theta}$ increases from a negative value to a positive value, then decreases and approaches 0 as $t \to +\infty$.

Consequently, $h(\theta)$ starts from $h(0) > 0$, first decreases then increases, reaching $h(+\infty) = 0$. Thus, the equation $h(\theta) = 0$ has a unique solution $\theta_0 > 0$.

Further, we have:

1. When $0 < \theta < \theta_0$, $f(\theta) > 0$ and $k_1 > 1$.
2. When $\theta = \theta_0$, $f(\theta) = 0$ and $k_1 = 1$.
3. When $\theta > \theta_0$, $f(\theta) < 0$ and $k_1 < 1$.

Suppose the optimal solution θ^* of $QE(\theta, a, b)$ satisfies $\theta^* \neq \theta_0$. Then there exists $k_1 \neq 1$ such that $QE(\theta^*, a, b) >$

Model & Config	Time-step T	
Acc/Energy	2	4
ResNet18 - 8/Opt, Param:11.7M, Acc:71.50%		
Acc	71.39	71.07
Energy ratio	0.2901	0.4473
ResNet18 - 8/999, Param:11.7M, Acc:71.50%		
Acc	68.422	68.412
Energy ratio	0.5279	1.0549
ResNet18 - 8/999_2, Param:11.7M, Acc:71.50%		
Acc	70.61	70.588
Energy ratio	0.452	0.9026
VGG16bn - 8/Opt, Param:138M, Acc:73.25%		
Acc	73.242	72.452
Energy ratio	0.2428	0.3237
VGG16bn - 8/999, Param:138M, Acc:73.25%		
Acc	72.15	72.132
Energy ratio	0.4734	0.9463
VGG16bn - 8/999_2, Param:138M, Acc:73.25%		
Acc	73.208	73.206
Energy ratio	0.4009	0.8009

Table 5: Accuracy and energy ratio of H-MT(Ours) with $n = 8$ thresholds, of ResNet18 and VGG16bn on ImageNet Dataset for supplement. Thresholds here are chosen through the generalized threshold optimization method, traditional policy and empirical modification of the traditional policy.

$QE(k_1\theta^*, a, b)$, which contradicts the fact that θ^* is the optimal solution of $QE(\theta, a, b)$. Thus, $\theta^* = \theta_0$.

In summary, when $a \leq 0 \leq b$, $a < b$, $a \in \mathbb{Z}$, and $b \in \mathbb{Z}$:

1. There exists a unique θ_0 such that the computed $k_1 = 1$.
2. When $0 < \theta < \theta_0$, $k_1 > 1$, and when $\theta > \theta_0$, $k_1 < 1$.
3. Moreover, $QE(\theta, a, b)$ has a unique optimal solution $\theta^* = \theta_0$.
4. θ will finally converge to the optimal θ_0. Through iteration $\theta \leftarrow \theta * k_1$, $QE(\theta)$ gets decreased. As obviously we have $QE(\theta) \geq QE(\theta_0)$, $QE(\theta)$ (seen as a sequence without serial numbers) will reach a limit. Suppose when $QE(\theta)$ reaches its limit, θ equals to a special value $\theta^{'}$ satisfying $QE(k_1\theta^{'}) = QE(\theta^{'})$. As analyzed before, such $\theta^{'}$ can only be θ_0. Therefore, θ will finally converge to the optimal θ_0.

□

F Supplement experiment of Section 5.2

Please refer to Table 5.