A. Detailed Experimental Setups

A.1. Finite Difference Scaling and Clipping in Aggregated Gradient Computation

Experimentally, it has been revealed that neurons' membrane potentials $\boldsymbol{u}$ mostly transfer to their closest neighbors during training, making the finite differences involving farther neighbors less important and sometimes even misleading. Furthermore, the third step in the NA pipeline computes the first-order approximation to the loss change with respect to the PSC, which becomes less accurate when there exists a large distance between the neuron's present $\boldsymbol{a}$ and its neighbor's PSC $\boldsymbol{a}_p$. Based on the two reasons above, instead of using the finite difference $f_{d(\boldsymbol{u},\boldsymbol{u}_p)}L$ defined in (14), we employed the following scaled finite difference $\widetilde{f}_{d(\boldsymbol{u},\boldsymbol{u}_p)}L$ for aggregated gradient computation:

$$\begin{aligned}\widetilde{f}_{d(\boldsymbol{u},\boldsymbol{u}_p)}L = \boldsymbol{e} \cdot (\boldsymbol{a}_p - \boldsymbol{a}) \\ \cdot\, clip\left(\frac{1}{\mathrm{d_{MP}}(\boldsymbol{u},\boldsymbol{u}_p)^3}, -b, b\right),\end{aligned} \tag{17}$$

In order to avoid value explosions when $\mathrm{d_{MP}}(\boldsymbol{u},\boldsymbol{u}_p)$ approaches zero, we clipped $\frac{1}{\mathrm{d_{MP}}(\boldsymbol{u},\boldsymbol{u}_p)^3}$ within $[-b, b]$. We recommend to set the hyperparameter $b \in [2, 20]$ for stable performance, which was set to 10 in all our experiments.

A.2. Training setups for different datasets

The proposed NA algorithm was run on a single Nvidia Titan Xp GPU to train SNNs based on three different datasets. The specific settings used are described below.

A.2.1. MNIST

The MNIST dataset (LeCun, 1998) contains 60,000 training images and 10,000 testing images. We set the batch size to 64, the number of training epochs to 200, and the learning rate to 0.0005 for the adopted AdamW optimizer (Loshchilov & Hutter, 2017). The images were converted to continuous-valued multi-channel currents applied as the inputs to the SNN under training. Moreover, data augmentations using RandomCrop and RandomRotation were applied to improve performance (Shorten & Khoshgoftaar, 2019).

A.2.2. N-MNIST

The N-MNIST (Orchard et al., 2015) is the neuromorphic version of the MNIST dataset (LeCun, 1998) and also has 60,000 training images and 10,000 testing images. We trained an SNN using the NA algorithm for 100 epochs with a batch size of 50. The AdamW optimizer (Loshchilov & Hutter, 2017) with a learning rate of 0.0005 was applied. No data augmentation was used.

A.2.3. CIFAR10

The CIFAR10 dataset (Krizhevsky et al., 2009) contains 50,000 training images and 10,000 test images. We trained our SNN using NA for 600 epochs with a batch size of 50 and a learning rate 0.0005 for the AdamW optimizer (Loshchilov & Hutter, 2017). The input image coding strategy used for the MNIST dataset was adopted. Moreover, data augmentations including RandomCrop, ColorJitter, and RandomHorizontalFlip (Shorten & Khoshgoftaar, 2019) were applied. The convolutional layers were initialized using the kaiming uniform initializer (He et al., 2015), and the linear layers were initialized using the kaiming normal initializer (He et al., 2015).