Biologically Inspired Dynamic Thresholds for Spiking Neural Networks

Jianchuan Ding
Dalian University of Technology
`djc_dlut@mail.dlut.edu.cn`

Bo Dong
Princeton University
`bo.dong@princeton.edu`

Felix Heide
Princeton University
`fheide@princeton.edu`

Yufei Ding
UC Santa Barbara
`yufeiding@cs.ucsb.edu`

Yunduo Zhou
Dalian University of Technology
`1227627334@mail.dlut.edu.cn`

Baocai Yin
Dalian University of Technology
`ybc@dlut.edu.cn`

Xin Yang
Dalian University of Technology
`xinyang@dlut.edu.cn`

Abstract

The dynamic membrane potential threshold, as one of the essential properties of a biological neuron, is a spontaneous regulation mechanism that maintains neuronal homeostasis, *i.e.*, the constant overall spiking firing rate of a neuron. As such, the neuron firing rate is regulated by a dynamic spiking threshold, which has been extensively studied in biology. Existing work in the machine learning community does not employ bioplausible spiking threshold schemes. This work aims at bridging this gap by introducing a novel bioinspired dynamic energy-temporal threshold (BDETT) scheme for spiking neural networks (SNNs). The proposed BDETT scheme mirrors two bioplausible observations: a dynamic threshold has 1) a positive correlation with the average membrane potential and 2) a negative correlation with the preceding rate of depolarization. We validate the effectiveness of the proposed BDETT on robot obstacle avoidance and continuous control tasks under both normal conditions and various degraded conditions, including noisy observations, weights, and dynamic environments. We find that the BDETT outperforms existing static and heuristic threshold approaches by significant margins in all tested conditions, and we confirm that the proposed bioinspired dynamic threshold scheme offers bioplausible homeostasis to SNNs in complex real-world tasks.

1 Introduction

A spiking neural network (SNN) is a bioinspired neural network. Each spiking neuron is a mathematical model abstracted from the properties of a biological neuron. Spiking neurons communicate with each other through spike trains, mimicking the information transfer process of biological neurons [1, 2, 3]. Similar to how biological action potentials are all-or-none impulses, the spikes of SNNs are commonly binary voltage pulses. Leveraging this binary representation, specifically designed neuromorphic hardware [4, 5, 6], *e.g.*, TrueNorth [7] and Loihi [8], can run SNNs at extremely low power levels; they are 75 times more energy-efficient than their deep neural network counterparts on low-power GPU platforms [9]. As such, recently, SNNs have rapidly emerged as effective models for robotic control tasks, especially in mobile robots that demand low power consumption [10, 11].

However, existing SNNs suffer from poor generalizability, unlike their biological counterparts. Biologically, a neuron leverages a spontaneous regulation mechanism to maintain neuronal home-

Preprint. Under review.

ostasis [12]—the stable overall spiking firing rate or excitability within a network [13]—to robustly adapt to different external conditions and offer strong generalization. A dynamic threshold, one type of regulatory mechanism, plays an essential role in maintaining neuronal homeostasis by regulating the action potential firing rate; such thresholds are widely observed in different nervous systems [14, 15, 16, 17, 18, 19, 20, 21, 22, 23]. This threshold can be regarded as an adaptation to membrane potentials at short timescales [16], and it influences how the received signals of a neuron are encoded into a spike.

Even though different dynamic threshold schemes have been observed and extensively studied in neuroscience, only a handful of existing works investigate bioinspired dynamic threshold rules to improve the generalization of SNNs. Hao *et al.* [24] proposed a dynamic threshold method that relies on a heuristic dynamic scaling factor to gradually slow the growth of a threshold. Conversely, instead of controlling threshold growth, Shaban *et al.* [25] leveraged double exponential functions to manage the threshold decay. Kim *et al.* [26] used a predefined target firing count to adjust their threshold but did not define the optimal target firing count. No existing work has demonstrated that a bioinspired dynamic threshold scheme can achieve homeostasis in real-world tasks. More importantly, the existing work only validates the proposed dynamic threshold rules under ideal normal conditions without testing generalization to degraded conditions, which we argue is essential to validate whether homeostasis is achieved or not.

The direct use of bioplausible models in SNNs remains challenging, as most of these models are based on single cells in the nervous system and contain many optimized constants. In this work, we lift this limitation and introduce a novel bioinspired dynamic energy-temporal threshold (BDETT) scheme for SNNs; the scheme comprises two components: a dynamic energy threshold and a dynamic temporal threshold schema. The two components reflect the following two biological observations: in *vivo*, the dynamic threshold exhibits a positive correlation with the average membrane potential and a negative correlation with the preceding rate of depolarization (*i.e.*, the excitatory status) [16]. The dynamic energy threshold is inspired by a biological predictive model which can predict the occurrences of spikes based on the previous membrane potential in the inferior colliculus of a barn owl [16]. The proposed dynamic temporal threshold component is inspired by the fact that a monoexponential function can effectively present a negative correlation [17, 22]. Notably, we provide an analysis of the original biological models and propose layerwise statistical cues for SNNs to replace the constants in the two original biological models.

We integrate the proposed BDETT into two widely used SNN models: a spike response model (SRM) [27] and a leaky integrate-and-fire (LIF) model [28] (see Supplementary Material for details). The effectiveness of BDETT is validated with these two SNN models for both autonomous robotic obstacle avoidance and continuous control tasks under normal and various degraded conditions, *e.g.*, dynamic obstacles, noisy inputs, and weight uncertainty. Extensive experimental results validate that the SNNs equipped with the proposed BDETT offer the strongest generalization across all tested scenarios. More importantly, we quantitatively validate that BDETT can significantly increase the homeostasis of the host SNN. This is the first work to demonstrate that bioinspired dynamic threshold schemes can offer bioplausible homeostasis to SNNs in robotic real-world tasks under normal and degraded conditions, dramatically enhancing the generalizability and adaptability of the host SNNs.

In particular, we make the following contributions in this work:

- We introduce a bioinspired dynamic threshold scheme for SNNs that increases their generalizability.
- We devise a method that uses layerwise statistical cues of SNNs to set the parameters of our bioinspired threshold method.
- We validate that the proposed threshold scheme achieves bioplausible homeostasis, dramatically enhancing the generalizability across tasks, including obstacle avoidance and robotic control, and in normal and degraded conditions.

Scope We propose a novel approach to setting the parameters of our threshold scheme using layerwise statistical cues of an SNN. Although this is essential for the proposed method to be effective, implementing these statitical blocks directly in neuromorphic hardware may require extra engineering efforts, which is out of the scope of this work.

2 Background and Related Work

2.1 Spiking Neural Networks (SNNs)

Various models for spiking neurons have been described to mathematically describe the properties of a nervous neuron. Typically, three conditions are considered by these models: resting, depolarization, and hyperpolarization. When a neuron is resting, it maintains a constant membrane potential. The change in membrane potential can be either a decrease or an increase relative to the resting potential. An increase in the membrane potential is called depolarization, which enhances the ability of a cell to generate an action potential; it is excitatory. In contrast, hyperpolarization describes a reduction in the membrane potential, which makes the associated cell less likely to generate an action potential, and, as such, is inhibitory. All inputs and outputs of a spiking neuron model are sequences of spikes. A sequence of spikes is called a spike train and is defined as $s(t) = \Sigma_{t^{(f)} \in \mathcal{F}} \delta(t - t^{(f)})$, where $\mathcal{F}$ represents the set of times at which the individual spikes occur [29]. Typical spiking neuron models set the resting potential as 0. However, existing models achieve depolarization and hyperpolarization in substantially different ways. In the following, we briefly review two commonly used models: the spike response model (SRM) [27] and leaky integrate-and-fire (LIF) model [28]. More details about these two models are provided in Supplementary Note 1.

Spike Response Model (SRM) An SRM first converts an incoming spike train $s_i(t)$ into a spike response signal as $(\varepsilon * s_i)(t)$, where $\varepsilon(\cdot)$ is a spike response kernel. Then, the generated spike response signal is scaled by a synaptic weight w_i. Depolarization is achieved by summing all the scaled spike response signals: $\Sigma_i w_i(\varepsilon * s_i)(t)$. When incoming spike trains trigger a spike $s(t)$, the SRM models hyperpolarization by defining a refractory potential as $(\zeta * s)(t)$, where $\zeta(\cdot)$ is a refractory kernel.

Leaky Integrate-and-Fire (LIF) An LIF model is a simplified variant of an SRM. This scheme directly processes incoming spike trains and ignores the spike response kernel. Hyperpolarization is achieved by a simplified step decay function, $f_d(s(t)) = D$ for $s(t) = 0$; 0 for $s(t) = 1$.

2.2 Spiking Neural Networks for Robot Control

Biological neural circuits have an impressive ability to avoid obstacles robustly in complex dynamical environments, *e.g.*, as in dragonfly flight trajectories. Inspired by this observation, recently, researchers have explored SNNs for obstacle avoidance [30, 31, 32, 33]. For example, Tang *et al.* [33] devised an SNN to mimic a neurophysiologically plausible connectome of the brain's navigational system without assuming all-to-all connectivity. Following the path, Tang *et al.* [9] proposed a spiking deep deterministic policy gradient (SDDPG) method to train a LIF-based spiking actor-network (SAN) for mapless navigation. They show that SNNs can robustly control a robot in mapping tasks while being able to explore an unknown environment. SNNs have also been proposed for continuous robot control tasks. Patel *et al.* [34] proposed to combine SNNs with a Deep Q-network algorithm, improving the robustness to occlusion in the input image. Tang *et al.* [35] proposed a population-coded spiking actor network (PopSAN) to solve high-dimensional continuous control problems, trained using deep reinforcement learning algorithms. Recently, modern neuromorphic hardware has made it possible to deploy SNNs on neuromorphic processors in ultra-low power envelopes [9, 36, 37, 38]. Compared to existing convolutional deep policy networks [39] on the mobile-GPUs such as the Nvidia Jetson TX2, SAN and PopSAN on Loihi neuromorphic processor consume 75 and 140 times less energy per inferences, respectively. All SNN-based models discussed above only consider static spiking thresholds. More importantly, experiments show that they suffer from poor generalization and fail in realistic degraded conditions. In this work, we use both SAN and PopSAN as testbeds and baseline methods to validate the effectiveness of the proposed bioinspired dynamic threshold scheme, BDETT.

3 Bio-Plausible Dynamic Energy-Temporal Threshold (BDETT)

Motivated by the behavior of spiking threshold dynamics in biological nervous systems, we propose a model with dynamic thresholds that exhibit positive and negative correlations with the average membrane potential and the preceding rate of depolarization, respectively. To achieve this behavior in the proposed scheme, given the i-th neuron in the l-th layer at timestamp $t + 1$, we define a bioplausible dynamic threshold $\Theta_i^l(t + 1)$ as

$$\Theta_i^l(t+1) = \frac{1}{2}(\mathrm{E}_i^l(t) + \mathrm{T}_i^l(t+1)), \tag{1}$$

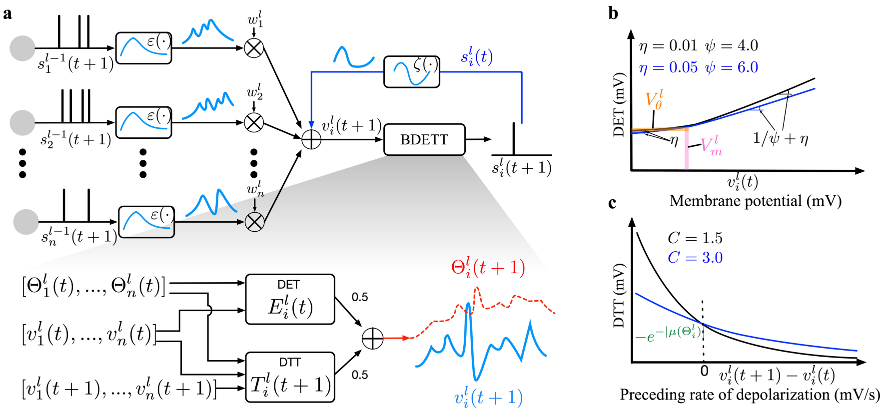

Figure 1: **An illustration of the proposed BDETT scheme**. a. We demonstrate the intuitive idea of our BDETT scheme for the i-th neuron in the l-th layer at timestamp $t+1$ from the perspective of an SRM-based SNN model; $\Theta_i^l(t)$ and $v_i^l(t)$ are the dynamic threshold and postsynaptic membrane potential of the i-th neuron in the l-th layer at timestamp t, respectively. b & c. Two example DET and DTT graphs, respectively.

where $E_i^l(t)$ is the dynamic energy threshold (DET) of the neuron for ensuring a positive correlation, and $T_i^l(t+1)$ is the dynamic temporal threshold (DTT), which ensures a negative correlation; see Figure 4a. Note that each neuron has a different dynamic threshold at timestamp $t+1$ based on the proposed DET and DTT, which we describe below.

Dynamic Energy Threshold (DET) Positive correlations between dynamic thresholds and average membrane potentials have been observed in several areas of diverse biological nervous systems, such as the visual cortex and auditory midbrain [17, 21, 22]. With sufficient voltage measurements at spike onsets, one can fit a model to directly predict the voltage of a threshold [40]. However, the fitted biological model is only meaningful to a specific nervous system, and stimulus or measurement uncertainty can significantly impact the model accuracy. Fontaine *et al.* [16] proposed a biological predictive approach to assess the occurrence of spikes based on the previous membrane potential; this method does not rely on voltage measurements at spike onsets. Even though the model was based on a barn owl's inferior colliculus, it exhibits great generality in terms of threshold variability statistics with other nervous systems (*e.g.*, cortical neurons) [16]. The proposed dynamic energy threshold is inspired by this biological predictive model but includes several changes that are critical for the model to be effective in SNNs. For the i-th neuron in the l-th layer at timestamp t, we define

$$\mathrm{E}_i^l(t) = \eta(v_i^l(t) - V_m^l(t)) + V_\theta^l(t) + ln(1 + e^{\frac{v_i^l(t) - V_m^l(t)}{\psi}}), \tag{2}$$

$$V_m^l(t) = \mu(v_i^l(t)) - 0.2(\max(v_i^l(t)) - \min(v_i^l(t))) \quad \text{for } i = 1, 2, ..., n^l, \tag{3}$$

$$V_\theta^l(t) = \mu(\Theta_i^l(t)) - 0.2(\max(\Theta_i^l(t)) - \min(\Theta_i^l(t))) \quad \text{for } i = 1, 2, ..., n^l, \tag{4}$$

where $v_i^l(t)$ is the neuron postsynaptic membrane potential at timestamp t; μ is the mean operator; n^l is the total number of neurons in the l-th layer; and η and ψ are two hyperparameters, which are set empirically. Figure 4b shows two example graphs for Eq. 2; η controls the shallow slope, and $\frac{1}{\psi} + \eta$ defines the slope of the steep part.

Intuitively, $V_\theta^l(t)$ and $V_m^l(t)$ define a critical region. When the membrane potential $v_i^l(t)$ is smaller than $V_m^l(t)$, the function has a shallower slope, and the threshold value is dominated by $V_\theta^l(t)$. In the opposite case, the energy threshold has a higher rate of increase to inhibit a high spiking firing rate. In the biological predictive model proposed by Fontaine et al. [16], $V_m^l(t)$ and $V_\theta^l(t)$ are the constants to be optimized during the model fitting process. However, we find that directly adopting these two fitted constants in an SNN does not result in generalization. Refer to experiment in the main paper which shows this To tackle this challenge, we leverage the statistical cues of SNN layers to adjust these two important parameters, as defined in Eqs. 3 and 4. Specifically, we model $V_m^l(t)$ as the mean of the membrane potentials of the neurons in the layer l. The mean value is shifted by a bias, $0.2(\max(v_i^l(t)) - \min(v_i^l(t)))$, which is based on the range of the potentials; see Eq. 3. The motivation behind this formulation is that we aim to couple the DET and the potentials of all other neurons in the same layer. Furthermore, we leverage the bias term to adjust the DET sensitivity to the layerwise potential range. Here, $V_\theta^l(t)$ is modeled based on similar insights, where we use threshold potentials (*i.e.*, $\Theta_i^l(t)$) instead of membrane potentials; see Eq. 4. We note that the performance of the proposed BDETT is not sensitive to the constant value 0.2; see Supplementary Note 7 for details.

Dynamic Temporal Threshold (DTT) We propose a DTT scheme to address the observed negative correlation between the spiking threshold and the preceding rate of depolarization. Azouz *et al.* [17, 22] discovered that a monoexponential function $y = a + be^{-V/C}$ can effectively capture the negative correlation of a biological neuron, where $V = dV_m/dt$; C is a decay constant; and a, b, and C are parameters to optimize. The authors applied this function to 42 cortical neurons and found significant correlations in 92% of the trials [17]. We propose a variant of this mechanism. In particular, we replace the constant a with an exponential decay function, and we base the decay rate on the mean of the dynamic thresholds of all neurons in the l-th layer at the previous timestamp t; b is set to 1. Additionally, we empirically set the delay constant C. Mathematically, for the i-th neuron in the l-th layer, the DTT at timestamp $t+1$ is defined as

$$\mathrm{T}_i^l(t+1) = a + e^{\frac{-(v_i^l(t+1)-v_i^l(t))}{C}}, \tag{5}$$

$$a = -e^{-|\mu(\Theta_i^l(t))|} \quad \text{for } i = 1, 2, ..., n^l. \tag{6}$$

Figure 4c shows two example graphs for Eq. 5. These plots highlight that higher depolarization (*i.e.*, $v_i^l(t+1) - v_i^l(t) > 0$) leads to a lower temporal threshold, while higher hyperpolarization (*i.e.*, $v_i^l(t+1) - v_i^l(t) < 0$) significantly increases the temporal threshold. We propose modeling a similar to how $V_m^l(t)$ is modeled in the DET, that is, by coupling the DTT value and the layerwise dynamic thresholds at the previous timestamp t (*i.e.*, $\Theta_i^l(t)$). The delay constant C adjusts the sensitivity of the DTT to changes in the temporal potential of a neuron. As shown in Figure 4c, a lower C value results in a substantially faster drop in the DTT value (*i.e.*, the black curve) than that provided by a higher C value (*i.e.*, the blue curve).

Interaction of DET and DTT A critical difference between DET and DTT lies in the drivers of the two threshold schemes. DET leverages the magnitude of the membrane potential to estimate a threshold, while the DTT based on the preceding rate of depolarization. Therefore, they may be counteracting or helping each other to achieve an optimal threshold. One example is that when noise causes low potential fluctuations, the overall threshold should increase to suppress the noise. In this case, the DET increases as the noise increases the membrane potential. However, DTT remains at a relatively constant threshold (*i.e.*, $a + 1$) as the preceding rate of depolarization caused by the noise is close to 0. When a neuron experiences a fast membrane potential drop, *e.g.*, during the relative refractory period, we expect the overall threshold to increase. In this scenario, even though DET decreases with the reduced membrane potential, DTT increases faster. Hence, the proposed method increases the overall threshold in this case.

4 Experiments

We assess the effectiveness of BDETT on two different tasks: robot obstacle avoidance and robotic continuous control. In the robot obstacle avoidance task, a robot aims to reach a randomly chosen destination without touching any obstacle within 1000 steps, counted as a "pass". For this task, we assess methods by measuring success rate (SR), the percentage of successful passes out of 200 trials. As continuous control tasks, we evaluate the HalfCheetah-v3 and Ant-v3 control outputs (see Figure 3a) from the OpenAI gym [41]. In these two continuous control tasks, an agent relies on a learned SNN-based control policy to decide the next action based on the current observation (*i.e.*, state), and each action is associated with a reward; see Figure 3a. We assess control policies with the total sum of the rewards. Note that the Ant-v3 control task is more challenging than HalfCheetah-v3, with significantly large state and action spaces.

In addition to evaluating the control output, *i.e.*, SR and total reward, we also measure the homeostasis of the host SNNs. In particular, we use three statistical metrics, FR_m, FR_{std}^m, and FR_{std}^s, to quantify the homeostasis of an SNN; these metrics are based on the neuron firing rate. FR_m is the mean neuron firing rate of an SNN across all P trials; FR_{std}^m is the average of P standard deviations, and each of them is the standard deviation of the neuron firing rates of an SNN during a single trial; FR_{std}^s denotes the standard deviation of the P standard deviations. FR_{std}^s represents the standard deviation across all P trials, while FR_{std}^m denotes the mean of these standard deviations. Details on these three metrics can be found in Supplementary Note 2.

Experimental Setup For robot obstacle avoidance tasks, the we use variants of the spiking actor network (SAN) [9] as host SNN. The original SAN uses LIF as its neuron model, but it resets the membrane potentials of all neurons to zero for each robot state. The resting operation is contradictory to the leaky function of LIF. Therefore, we modify the SAN by removing the resting operation, which is dubbed SAN-NR. To validate the effectiveness of the proposed BDETT, we integrate it into both LIF-based and SRM-based SAN-NR models and compare them with their original static threshold and two heuristic dynamic threshold schemes, DT1 [24] and DT2 [26]. See Supplementary Note 2 for details on the DT1 and DT2 schemes. We set the batch size to 256 and the learning rate to 0.00001 for both the actor and critic networks during the training process. In addition, we use the following hyperparameter settings for the proposed BDETT: $\eta = 0.01$ and $\psi = 4.0$ for the DET and $C = 3.0$ for the DTT. For estimating homeostasis, we set $P = 200$. See Supplementary Notes 4 for further training details.

For robot continuous control, we adopt the population-coded SAN (PopSAN) [35] as our baseline model; it is a modified version of SAN [9] with a specifically designed encoder and decoder for accommodating high-dimensional control tasks. Note that PopSAN does not rest the membrane potentials as the encoder leverages soft-reset IF neurons. Hence, PopSAN is the counterpart of the SAN-NR used in the obstacle avoidance tasks. We integrate BDETT into both LIF- and SRM-based PopSAN models and compare them with their original static threshold schemes and the two heuristic dynamic schemes, DT1 and DT2. Following the evaluation settings of PopSAN [35], we train ten models corresponding to ten random seeds, and the best-performing model is used for our assessment conducted under different degraded conditions. In particular, the best-performing model is evaluated ten times under each experimental condition, and the mean reward of the ten evaluations represents the model performance. Each evaluation consists of ten episodes, and each episode lasts for a maximum of 1000 execution steps. Hence, the P value used for estimating homeostasis is set to 100, *i.e.*, 10 episodes $\times$ 10 evaluations. PopSAN and its variants are trained by using the twin-delayed deep deterministic policy gradient off-policy algorithm [42]. The hyperparameter settings of BDETT is the same as the ones used for obstacle avoidance tasks, except the ψ for the DET is set to 6.0. Following the training protocol of PopSAN [35], we set the batch size to 100 and the learning rate to 0.0001 for both the actor and critic networks. The reward discount factor is set to 0.99, and the maximum length of the replay buffer is set to 1 million. See Supplementary Notes 5 for more training details related.

For the SRM-based baseline methods, the spike response kernel and refractory kernel of the SRM are adopted from [27, 29], and they are defined as $\varepsilon(t) = te^{1-t}$ and $\zeta(t) = -2\Theta(t)e^{-t}$, respectively. For all tasks, each dimension of a robot state is encoded into a spike train with T timesteps. All experimental results are obtained with $T = 5$. For a demonstration of the generalization provided by the BDETT, we provide the experimental results obtained with $T = 25$ in Supplementary Notes 4, 5, 6 for the obstacle avoidance, HalfCheetah-v3, and Ant-v3 tasks, respectively.

4.1 Robot Obstacle Avoidance with BDETT

We evaluate the proposed method for robot obstacle avoidance tasks with one standard condition, *i.e.*, static obstacles, and three specifically designed adverse conditions: dynamic obstacles, degraded inputs, and weight uncertainty. For the dynamic obstacle experiments, we introduce 11 dynamically moving cylinders in a static testing environment, and each repeatedly wanders between two points; see Figure 2a. The wandering distance and speed are designed to provide sufficient space and time to allow possible passes. The robot utilizes a Robo Peak light detection and ranging (RPLIDAR) system as its sensing device to detect obstacles, offering a field of view of 180 degrees with 18 range measurements, as shown in Figure 2b.

In our degraded input scenario, we disturb the obtained range measurements in three different ways: “0.2": We set the range of the 3rd, 9th, and 15th lasers to 0.2 m, always reporting obstacles even when none occur; “6.0": This is similar to the “0.2" setting, but we set the three lasers’ ranges to 6.0 m, which is the average visible range in the test environment and means that the three lasers cannot perceive any obstacles; “GN": We add Gaussian noise [43] to each of the 18 range measurements. The three proposed degraded input settings are illustrated in Figure 2c.

In the weight uncertainty experiments, as illustrated in Figure 2d, the learned synaptic weights of the host SNNs are also disturbed in three different ways. “8-bit Loihi weight": Neuromorphic hardware (*e.g.*, Loihi) achieves computing efficiency by sacrificing the weight precision. Therefore, when deploying an SNN on neuromorphic hardware, one needs to scale and round up the learned floating-point synaptic weights to low-precision weights. “GN weight": We add Gaussian noise to all weights. “30% zero weight": Among the synaptic weights between every two adjacent layers, we randomly set 30% of them to 0. To reduce the impact of the randomness introduced in the “GN

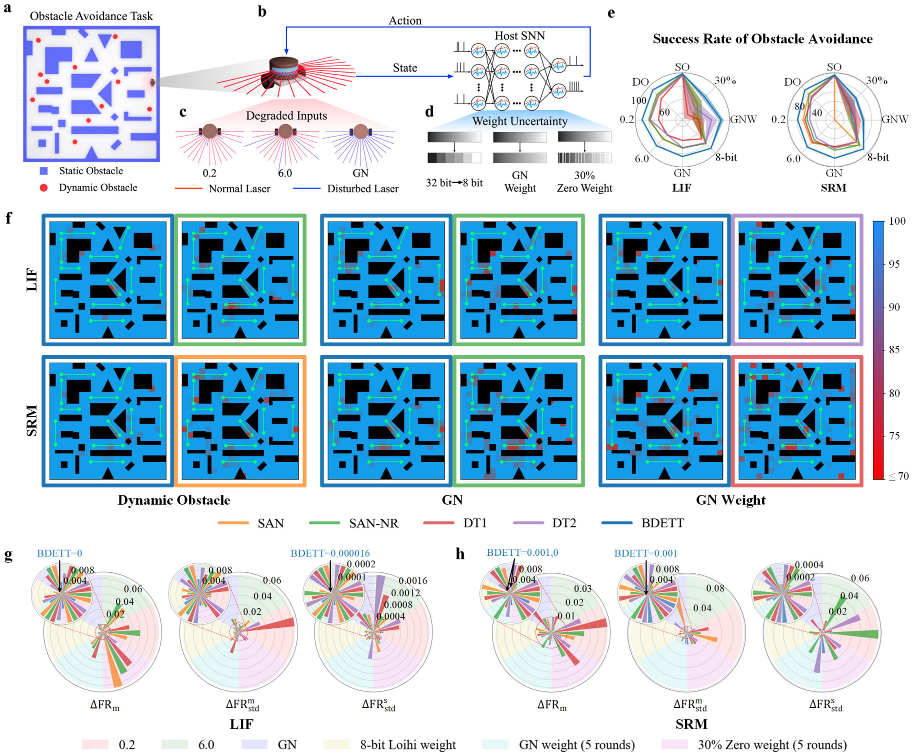

Figure 2: Proposed method for robot obstacle avoidance. a. The static and dynamic testing environments of the obstacle avoidance tasks. b. The control loop of a robot. c. The three specifically designed degraded input conditions. d. A demonstration of the three weight uncertainty experimental settings. e. The SRs of obstacle avoidance under all experimental settings. 'SO' and 'DO' indicate the testing environments with static and dynamic obstacles, respectively; '30%' and 'GNW' denote the "30% zero weight" and "GN weight" conditions. f. The heatmaps yielded by the best and runner-up performers under three different conditions indicate the areas with lower SRs (*i.e.*, thoes shown in red). g & h show the LIF- and SRM-based SNNs' homeostasis changes with respect to the base condition (*i.e.*, DO) in terms of three metrics. e-h use the same color codes as shown in f.

weight" and "30% zero weight" experiments, we report the average success rates (SRs) of 5-round tests.

Success Rate The SRs of the competing LIF- and SRM-based approaches across all experimental settings are reported in Figure 2e. For the "GN weight" and "30% zero weight" experiments, the standard deviations of the 5-round SRs are also reported. The proposed BDETT achieves the highest SRs in all experiments, demonstrating its effectiveness. Notably, under dynamic obstacle conditions, the BDETT outperforms the runners-up by significant margins (9% versus the LIF and 12% versus the SRM). Under degraded input conditions, the BDETT yields at least 10% more successful passes than other competing methods. In the weight uncertainty experiments, our BDETT increases the SRs of the baseline SAN-NR model by at least 10.5%, 24.6%, and 15.6% under "8-bit Loihi weight", "GN weight", and "30% zero weight" settings, respectively. The success rates of the best and runner-up performers under three different adverse conditions are also qualitatively illustrated in Figure 2f (see Supplementary Figure 3 for additional results). We observe that our BDETT can help the robots effectively avoid both static and dynamic obstacles under all three adverse conditions; see Supplementary Tables 2, 3, and 4 for details.

Homeostatic Evaluation When an SNN is in homeostasis, all neurons are expected to have similar and sparse firing patterns under different conditions [44, 45]. Therefore, when transferring from one condition to another, the SNNs with stronger homeostasis are expected to induce fewer changes in all three metrics. The changes induced in all successful trials involving the LIF- and SRM-based host SNNs under different experimental settings are illustrated in Figures 2g and h, respectively. The changes (*i.e.*, in ΔFR_m, ΔFR^m_{std}, and ΔFR^s_{std}) are estimated with respect to the corresponding homeostasis achieved in the dynamic obstacle experiments, *i.e.*, under the base condition. The proposed BDETT scheme yields minimal changes in all three metrics when transferring from the base condition to all other experimental settings, except for the $\Delta\mathrm{FR}^{s}_{std}$ estimated based on the SRM-based 8-bit Loihi weight experiment. The figures highlight that the proposed BDETT significantly improves on the baseline SAN-NR model, as evidenced by the remarkable drops in these three statistical metrics. For example, as shown in the "6.0" section of the $\Delta\mathrm{FR}_m$ in Figure 2g, our dynamic threshold scheme reduces the $\Delta\mathrm{FR}_m$ from 0.043 to 0.001. In the "0.2" section of the $\Delta\mathrm{FR}^{s}_{std}$ in Figure 2h, the $\Delta\mathrm{FR}^{s}_{std}$ is decreased to 1.7% of its original value (from 0.0058 to 0.0001). We also witness that the DT1 and DT2 schemes significantly weaken the baseline model's homeostasis, as shown in Figure 2g in the "0.2" section of the $\Delta\mathrm{FR}^{m}_{std}$ and the "6.0" section of the $\Delta\mathrm{FR}^{s}_{std}$.

The goal of homeostasis to enhance the host SNN generalization. Therefore, we expect SNNs with stronger homeostasis (*i.e.*, smaller $\Delta\mathrm{FR}_m$, $\Delta\mathrm{FR}^{m}_{std}$, and $\Delta\mathrm{FR}^{s}_{std}$ values) to outperform those with weaker homeostasis. Our experimental results confirm this, validating that the strong homeostasis provided by our BDETT can improve the generalization capabilities of SNNs to different degraded conditions. We argue that this is a highly desired capability not only for mobile robotics but also for broader machine learning. See Supplementary Note 4 for more experimental results and analysis.

4.2 Continuous Robot Control with BDETT

For the HalfCheetah-v3 and Ant-v3 tasks, similar to the robot obstacle avoidance tasks, we evaluate on one standard and two specifically designed degraded inputs and weight uncertainty adverse conditions to demonstrate the strong generalization enabled by our BDETT. In this context, for the degraded input conditions, we disturb the observations of these two control tasks in three ways. "Random joint position": For each episode, one of the joint positions is randomly selected, and its original position is replaced by a random number sampled from a Gaussian distribution $\mathcal{N}(0, 0.1)$. "Random joint velocity": We randomly select one of the joint velocities in each episode and change its observed velocity to a random number sampled from a Gaussian distribution $\mathcal{N}(0, 10.0)$. "GN": In each episode, we add Gaussian noise sampled from the distribution $\mathcal{N}(0, 1.0)$ to each dimension of a state; see Figure 3b. The weight uncertainty conditions of the control tasks are the same as those used in the robot obstacle avoidance tasks, as illustrated in Figure 3c.

Rewards As shown in Figures 3d and e, under all experimental settings, the proposed BDETT offers the host SNNs the highest rewards, significantly improving upon the rewards of the baseline PopSAN model by at least 438 (*i.e.*, the SRM-based PopSAN model under the "GN" setting) for the HalfCheetah-v3 tasks and 213 (*i.e.*, the LIF-based PopSAN model under the "Random joint velocity" setting) for the Ant-v3 tasks. Notably, under weight uncertainty conditions with a HalfCheetah-v3 agent, even with low-precision 8-bit weights, the proposed BDETT helps the SRM-based host SNN achieve a higher reward than that obtained with high-precision floating-point weights (11767 vs. 11268); see Supplementary Tables 7 and 9. With an Ant-v3 agent, the proposed BDETT helps both the LIF- and SRM-based host SNNs achieve higher rewards, even with low-precision weights, *i.e.*, 5570 vs. 5526 and 5648 vs. 5643, respectively. See Supplementary Notes 5 and 6 for additional experimental results and analysis related to the HalfCheetah-v3 and Ant-v3 tasks, respectively.

Homeostatic Evaluation In Figures 3f-i, we show the changes induced in these three metrics when shifting from normal conditions (*i.e.*, the base conditions) to all other experimental settings. The proposed BDETT offers the strongest homeostasis to the host SNNs among all competing approaches for both the HalfCheetah-v3 and Ant-v3 control tasks.

In particular, for the HalfCheetah-v3 control task, as shown in the "30% zero weight" section of the $\Delta\mathrm{FR}_m$ in Figure 3f, our dynamic threshold scheme reduces the $\Delta\mathrm{FR}_m$ of the baseline PopSAN model from 0.069 to 0.006. In the "GN weight" section of the $\Delta\mathrm{FR}^{s}_{std}$ in Figure 3g, the proposed BDETT decreases the $\Delta\mathrm{FR}^{s}_{std}$ of the SRM-based PopSAN to 8.3% of its original value (from 0.0012 to 0.0001); see Supplementary Table 10 for details. For the Ant-v3 control task, as shown in the "GN weight" section of the $\Delta\mathrm{FR}_m$ in Figure 3h, our dynamic threshold scheme reduces the $\Delta\mathrm{FR}_m$ of the LIF-based baseline model from 0.041 to 0.003. In the "Random joint position" section of the $\Delta\mathrm{FR}^{s}_{std}$ in Figure 3h, the $\Delta\mathrm{FR}^{s}_{std}$ of the LIF-based baseline model is decreased to 10% of its original value (from 0.0010 to 0.0001); see Supplementary Table 15 for details. As in the obstacle avoidance tasks, the DT1 and DT2 schemes significantly decrease the homeostasis of both the LIF- and SRM-based baseline models in both continuous control tasks. Some extreme cases are shown in the "Random joint velocity" section of the $\Delta\mathrm{FR}^{m}_{std}$ in Figure 3f, and the "GN weight" section of the $\Delta\mathrm{FR}_m$ in Figure 3i.

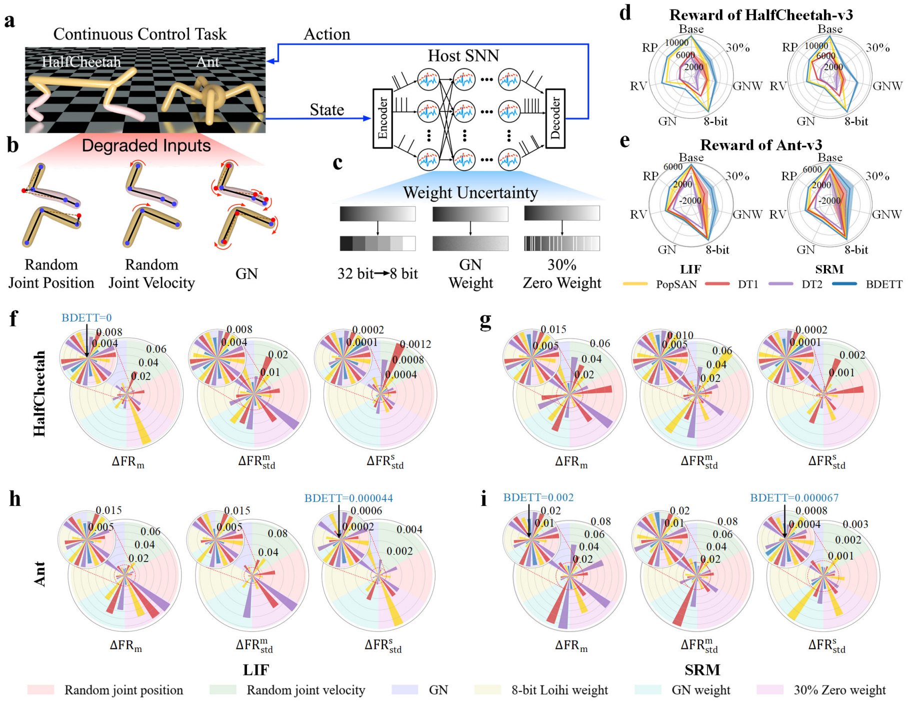

Figure 3: Proposed method for continuous robot control. a. The control loops of HalfCheetah-v3 and Ant-v3. b. Examples of the three specifically designed degraded input conditions, where the red dots and curved arrows indicate the disturbed joint positions and velocities, respectively. c. The three specifically designed weight uncertainty conditions. d & e. The rewards of the HalfCheetah-v3 and Ant-v3 tasks across all experimental conditions, respectively. 'Base' indicates the normal base condition; 'RP' and 'RV' denote 'Random joint position' and 'Random joint velocity'. f & g. The LIF- and SRM-based SNNs' homeostasis changes with respect to the 'Base' condition in the HalfCheetah-v3 tasks. h & i. The LIF- and SRM-based SNNs' homeostasis changes with respect to the 'Base' condition in the Ant-v3 tasks. **d-i** use the same color codes shown in e.

These experimental results obtained for the two continuous control tasks support the observations obtained in the obstacle avoidance tasks. More importantly, we witness that the strong homeostasis provided by our BDETT improves generalization to severely degraded conditions.

4.3 BDETT without Statistical Parameter Adjustment

We found it essential to replace the constants in the two biological models we base our approach on with layerwise statistical cues. Here, we report the performance of the BDETT with the original constants of the fitted biological models, demonstrating the effectiveness of the proposed layerwise statistical parameter settings. In particular, we first use the corresponding constants of the fitted adaptive threshold model [16] and replaced the $V_m^l(t)$ and $V_\theta^l(t)$, *i.e.*, Eq. 3 and Eq. 4, with 3 and 7, respectively. These two constants are obtained by shifting the originally fitted constants -67 and -63 by 70 to compensate for the difference of the rest potentials; -70 mV in the original model but 0 mV for LIF and SRM models. Furthermore, we use the original fitted parameters in our DTT, and Eq. 5 becomes $\mathrm{T}_i^l(t+1) = 1.0 + 10e^{\frac{-(v_i^l(t+1)-v_i^l(t))}{3}}$. For obstacle avoidance, with the originally fitted constants, the LIF-based policy cannot produce any successful pass even under the standard testing condition; SR drops from 92.5% to 0%. For the HalfCheetah-v3 and Ant-v3 tasks, with the originally fitted constants, the rewards achieved by a LIF-based policy dropped from 11064 to -35 and 5276 to -9, respectively. Note that an untrained BDETT-based policy achieves -124 and -73 rewards for these two continuous control tasks. These experimental results validate that the proposed statistical cues are essential to the proposed method.

5 Conclusion

This work introduces a novel biologically inspired BDETT scheme to SNNs that significantly improves generalization, and as such, fills a gap between biological research and machine learning. Dynamic threshold behavior plays an essential role in maintaining a neuronal homeostasis in biological nervous systems. Motivated by this observation, we propose a dynamic threshold scheme to achieve homeostasis in artificial SNNs. We assess the proposed approach in real-world tasks under normal and severely degraded conditions to validate its generalization capabilities. We find that the proposed dynamic threshold achieves strong homeostasis along with generalization to diverse degraded conditions. This finding is a step toward employing bioplausible SNNs in real-world applications. As future work, we plan to implement the proposed scheme on neuromorphic hardware to broadly deploy BDETT in future robotic platforms.

References

[1] Manuel Valero, Ipshita Zutshi, Euisik Yoon, and György Buzsáki. Probing subthreshold dynamics of hippocampal neurons by pulsed optogenetics. *Science*, 375(6580):570–574, 2022.

[2] Ai Nakashima, Naoki Ihara, Mayo Shigeta, Hiroshi Kiyonari, Yuji Ikegaya, and Haruki Takeuchi. Structured spike series specify gene expression patterns for olfactory circuit formation. *Science*, 365(6448):eaaw5030, 2019.

[3] Cheng yu T. Li, Mu ming Poo, and Yang Dan. Burst spiking of a single cortical neuron modifies global brain state. *Science*, 324(5927):643–646, 2009.

[4] Paul A. Merolla, John V. Arthur, Rodrigo Alvarez-Icaza, Andrew S. Cassidy, Jun Sawada, Filipp Akopyan, Bryan L. Jackson, Nabil Imam, Chen Guo, Yutaka Nakamura, Bernard Brezzo, Ivan Vo, Steven K. Esser, Rathinakumar Appuswamy, Brian Taba, Arnon Amir, Myron D. Flickner, William P. Risk, Rajit Manohar, and Dharmendra S. Modha. A million spiking-neuron integrated circuit with a scalable communication network and interface. *Science*, 345(6197):668–673, 2014.

[5] Youhui Zhang, Peng Qu, Yu Ji, Weihao Zhang, Guangrong Gao, Guanrui Wang, Sen Song, Guoqi Li, Wenguang Chen, Weimin Zheng, Feng Chen, Jing Pei, Rong Zhao, Mingguo Zhao, and Luping Shi. A system hierarchy for brain-inspired computing. *Nature*, 586(7829):378–384, 2020.

[6] J. Feldmann, N. Youngblood, C. D. Wright, H. Bhaskaran, and W. H. P. Pernice. All-optical spiking neurosynaptic networks with self-learning capabilities. *Nature*, 569(7755):208–214, 2019.

[7] Michael V. DeBole, Brian Taba, Arnon Amir, Filipp Akopyan, Alexander Andreopoulos, William P. Risk, Jeff Kusnitz, Carlos Ortega Otero, Tapan K. Nayak, Rathinakumar Appuswamy, Peter J. Carlson, Andrew S. Cassidy, Pallab Datta, Steven K. Esser, Guillaume J. Garreau, Kevin L. Holland, Scott Lekuch, Michael Mastro, Jeff McKinstry, Carmelo di Nolfo, Brent Paulovicks, Jun Sawada, Kai Schleupen, Benjamin G. Shaw, Jennifer L. Klamo, Myron D. Flickner, John V. Arthur, and Dharmendra S. Modha. Truenorth: Accelerating from zero to 64 million neurons in 10 years. *Computer*, 52(5):20–29, 2019.

[8] Mike Davies, Narayan Srinivasa, Tsung-Han Lin, Gautham Chinya, Yongqiang Cao, Sri Harsha Choday, Georgios Dimou, Prasad Joshi, Nabil Imam, Shweta Jain, Yuyun Liao, Chit-Kwan Lin, Andrew Lines, Ruokun Liu, Deepak Mathaikutty, Steven McCoy, Arnab Paul, Jonathan Tse, Guruguhanathan Venkataramanan, Yi-Hsin Weng, Andreas Wild, Yoonseok Yang, and Hong Wang. Loihi: A neuromorphic manycore processor with on-chip learning. *IEEE Micro*, 38(1):82–99, 2018.

[9] Guangzhi Tang, Neelesh Kumar, and Konstantinos P Michmizos. Reinforcement co-learning of deep and spiking neural networks for energy-efficient mapless navigation with neuromorphic hardware. *arXiv preprint arXiv:2003.01157*, 2020.

[10] Travis DeWolf. Spiking neural networks take control. *Science Robotics*, 6(58):eabk3268, 2021.

[11] Ignacio Abadía, Francisco Naveros, Eduardo Ros, Richard R. Carrillo, and Niceto R. Luque. A cerebellar-based solution to the nondeterministic time delay problem in robotic control. *Science Robotics*, 6(58):eabf2756, 2021.

[12] Kevin B. Koronowski and Paolo Sassone-Corsi. Communicating clocks shape circadian homeostasis. *Science*, 371(6530):eabd0951, 2021.

[13] Gina G Turrigiano and Sacha B Nelson. Homeostatic plasticity in the developing nervous system. *Nature reviews neuroscience*, 5(2):97–107, 2004.

[14] Wei Zhang and David J. Linden. The other side of the engram: experience-driven changes in neuronal intrinsic excitability. *Nature Reviews Neuroscience*, 4(11):885–900, 2003.

[15] Leon N Cooper and Mark F. Bear. The bcm theory of synapse modification at 30: interaction of theory with experiment. *Nature Reviews Neuroscience*, 13(11):798–810, 2012.

[16] Bertrand Fontaine, José Luis Peña, and Romain Brette. Spike-threshold adaptation predicted by membrane potential dynamics in vivo. *PLOS Computational Biology*, 10(4), 2014.

[17] Rony Azouz and Charles M. Gray. Dynamic spike threshold reveals a mechanism for synaptic coincidence detection in cortical neurons in vivo. *Proceedings of the National Academy of Sciences of the United States of America*, 97(14):8110–8115, 2000.

[18] Luk Chong Yeung, Harel Z Shouval, Brian S Blais, and Leon N Cooper. Synaptic homeostasis and input selectivity follow from a calcium-dependent plasticity model. *Proceedings of the National Academy of Sciences*, 101(41):14943–14948, 2004.

[19] Qian-Quan Sun. Experience-dependent intrinsic plasticity in interneurons of barrel cortex layer iv. *Journal of Neurophysiology*, 102(5):2955–2973, 2009.

[20] Karine Pozo and Yukiko Goda. Unraveling mechanisms of homeostatic synaptic plasticity. *Neuron*, 66(3):337–351, 2010.

[21] Jose Luis Pena and Masakazu Konishi. From postsynaptic potentials to spikes in the genesis of auditory spatial receptive fields. *Journal of Neuroscience*, 22(13):5652–5658, 2002.

[22] Rony Azouz and Charles M Gray. Adaptive coincidence detection and dynamic gain control in visual cortical neurons in vivo. *Neuron*, 37(3):513–523, 2003.

[23] Guo Zhang, Ke Yu, Tao Wang, Ting-Ting Chen, Wang-Ding Yuan, Fan Yang, Zi-Wei Le, Shi-Qi Guo, Ying-Yu Xue, Song-An Chen, Zhe Yang, Feng Liu, Elizabeth C. Cropper, Klaudiusz R. Weiss, and Jian Jing. Synaptic mechanisms for motor variability in a feedforward network. *Science Advances*, 6(25):eaba4856, 2020.

[24] Yunzhe Hao, Xuhui Huang, Meng Dong, and Bo Xu. A biologically plausible supervised learning method for spiking neural networks using the symmetric stdp rule. *Neural Networks*, 121:387–395, 2020.

[25] Ahmed Shaban, Sai Sukruth Bezugam, and Manan Suri. An adaptive threshold neuron for recurrent spiking neural networks with nanodevice hardware implementation. *Nature Communications*, 12(1):4234, 2021.

[26] Taeyoon Kim, Suman Hu, Jaewook Kim, Joon Young Kwak, Jongkil Park, Suyoun Lee, Inho Kim, Jong-Keuk Park, and YeonJoo Jeong. Spiking neural network (snn) with memristor synapses having non-linear weight update. *Frontiers in computational neuroscience*, 15:22, 2021.

[27] Wulfram Gerstner. Time structure of the activity in neural network models. *Physical Review E*, 51(1):738–758, 1995.

[28] Wulfram Gerstner and Werner M. Kistler. *Spiking Neuron Models: Single Neurons, Populations, Plasticity*. 2002.

[29] Sumit Bam Shrestha and Garrick Orchard. Slayer: Spike layer error reassignment in time. In *Advances in Neural Information Processing Systems*, volume 31, pages 1412–1421, 2018.

[30] Moritz B Milde, Hermann Blum, Alexander Dietmüller, Dora Sumislawska, Jörg Conradt, Giacomo Indiveri, and Yulia Sandamirskaya. Obstacle avoidance and target acquisition for robot navigation using a mixed signal analog/digital neuromorphic processing system. *Frontiers in neurorobotics*, 11:28, 2017.

[31] Amarnath Mahadevuni and Peng Li. Navigating mobile robots to target in near shortest time using reinforcement learning with spiking neural networks. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 2243–2250. IEEE, 2017.

[32] Zhenshan Bing, Ivan Baumann, Zhuangyi Jiang, Kai Huang, Caixia Cai, and Alois Knoll. Supervised learning in snn via reward-modulated spike-timing-dependent plasticity for a target reaching vehicle. *Frontiers in neurorobotics*, 13:18, 2019.

[33] Guangzhi Tang and Konstantinos P Michmizos. Gridbot: an autonomous robot controlled by a spiking neural network mimicking the brain's navigational system. In *Proceedings of the International Conference on Neuromorphic Systems*, pages 1–8, 2018.

[34] Devdhar Patel, Hananel Hazan, Daniel J Saunders, Hava T Siegelmann, and Robert Kozma. Improved robustness of reinforcement learning policies upon conversion to spiking neuronal network platforms applied to atari breakout game. *Neural Networks*, 120:108–115, 2019.

[35] Guangzhi Tang, Neelesh Kumar, Raymond Yoo, and Konstantinos P Michmizos. Deep reinforcement learning with population-coded spiking neural network for continuous control. *arXiv preprint arXiv:2010.09635*, 2020.

[36] Guangzhi Tang, Arpit Shah, and Konstantinos P Michmizos. Spiking neural network on neuromorphic hardware for energy-efficient unidimensional slam. *arXiv preprint arXiv:1903.02504*, 2019.

[37] Kate D. Fischl, Kaitlin Fair, Wei-Yu Tsai, Jack Sampson, and Andreas Andreou. Path planning on the truenorth neurosynaptic system. In *2017 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1–4, 2017.

[38] Hermann Blum, Alexander Dietmüller, Moritz Milde, Jörg Conradt, Giacomo Indiveri, and Yulia Sandamirskaya. A neuromorphic controller for a robotic vehicle equipped with a dynamic vision sensor. *Robotics Science and Systems, RSS 2017*, 2017.

[39] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

[40] Anton V. Chizhov, Elena Yu. Smirnova, K. Kh. Kim, and Aleksey V. Zaitsev. A simple markov model of sodium channels with a dynamic threshold. *J. Comput. Neurosci.*, 37(1):181–191, 2014.

[41] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

[42] Scott Fujimoto, Herke Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In *International Conference on Machine Learning*, pages 1587–1596. PMLR, 2018.

[43] Jinyoung Choi, Kyungsik Park, Minsu Kim, and Sangok Seok. Deep reinforcement learning of navigation in a complex and crowded environment with a limited field of view. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 5993–6000. IEEE, 2019.

[44] Andreea Lazar, Gordon Pipa, and Jochen Triesch. 2007 special issue: Fading memory and time series prediction in recurrent networks with different forms of plasticity. *Neural Networks*, 20(3):312–322, 2007.

[45] Friedemann Zenke, Guillaume Hennequin, and Wulfram Gerstner. Synaptic plasticity in neural networks needs homeostasis with a fast rate detector. *PLOS Computational Biology*, 9(11), 2013.

[46] A.L. Hodgkin and A.F. Huxley. A quantitative description of membrane current and its application to conduction and excitation in nerve. *Journal of Physiology*, 117:500–544, 1952.

[47] Andrew Y. Y. Tan, Yuzhi Chen, Benjamin Scholl, Eyal Seidemann, and Nicholas J. Priebe. Sensory stimulation shifts visual cortex from synchronous to asynchronous states. *Nature*, 509(7499):226–229, 2014.

[48] David A. McCormick, Yousheng Shu, and Yuguo Yu. Hodgkin and huxley model —still standing? *Nature*, 445(7123):E1–E2, 2007.

[49] W Tackmann and HJ Lehmann. Refractory period in human sensory nerve fibres. *European neurology*, 12(5-6):277–292, 1974.

[50] Eric R. Kandel, James H. Schwartz, and Thomas M. Jessell, editors. *Principles of Neural Science*. Elsevier, New York, third edition, 1991.

[51] Jonathan Platkiewicz and Romain Brette. A threshold equation for action potential initiation. *PLoS computational biology*, 6(7):e1000850, 2010.

[52] Björn Naundorf, Fred Wolf, and Maxim Volgushev. Unique features of action potential initiation in cortical neurons. *Nature*, 440(7087):1060–1063, 2006.

[53] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *arXiv preprint arXiv:1912.01703*, 2019.

A Supplementary Material

Supplementary Note 1: Spiking Neural Networks

Spiking Neural Network (SNN)

Various models for spiking neurons mathematically describe the properties of a cell in the nervous system with varying degrees of detail. Normally, three conditions are considered by these models: resting, depolarization, and hyperpolarization. When a neuron is resting, it maintains a constant membrane potential. The change in membrane potential can be either a decrease or an increase relative to the resting potential. An increase in the membrane potential is called depolarization, which enhances a cell's ability to generate an action potential; it is excitatory. In contrast, hyperpolarization describes a reduction in the membrane potential, which makes the associated cell less likely to generate an action potential, and, as such, is inhibitory. All inputs and outputs of a spiking neuron model are sequences of spikes.

A sequence of spikes is called a spike train and is defined as $s(t) = \Sigma_{t^{(f)} \in \mathcal{F}} \delta(t - t^{(f)})$, where $\mathcal{F}$ represents the set of times at which the individual spikes occur [29]. Typical spiking neuron models set the resting potential to 0. However, existing models achieve depolarization and hyperpolarization in substantially different ways. In the following, we review two commonly used models: the spike response model (SRM) [27] and leaky integrate-and-fire (LIF) model [28].

Spike Response Model (SRM) An SRM first converts an incoming spike train $s_i(t)$ into a spike response signal as $(\varepsilon * s_i)(t)$, where $\varepsilon(\cdot)$ is a spike response kernel. Then, the generated spike response signal is scaled by a synaptic weight w_i. Depolarization is achieved by summing all the scaled spike response signals: $\Sigma_i w_i (\varepsilon * s_i)(t)$. When incoming spike trains trigger a spike $s(t)$, the SRM models hyperpolarization by defining a refractory potential as $(\zeta * s)(t)$, where $\zeta(\cdot)$ is a refractory kernel. With an SRM, a feedforward SNN architecture with n_l layers can be defined. Given N^l incoming spike trains at layer l, $s_i^l(t)$, the forward propagation process of the network is mathematically defined as follows [27, 29]:

$$v_i^{l+1}(t) = \sum_{j=1}^{N^l} w_{ij} (\varepsilon * s_j^l)(t) + (\zeta * s_i^{l+1})(t-1), \tag{7}$$

$$s_i^{l+1}(t) = f_s(v_i^{l+1}(t)), \tag{8}$$

$$f_s(v) : v \to s, s(t) := s(t) + \delta(t - t^{(f+1)}), \tag{9}$$

$$t^{f+1} = \min\{t : v(t) = \Theta, t > t^{(f)}\}, \tag{10}$$

where $f_s(\cdot)$ is a spike function and Θ is the membrane potential threshold, which is static and the same for all neurons in the network. This static threshold is the one that we replace with the proposed bioplausible dynamic threshold.

Leaky Integrate-and-Fire (LIF) An LIF model is a simplified variant of an SRM. This scheme directly processes incoming spike trains and ignores the spike response kernel. Hyperpolarization is achieved by a simple decay function $f_d(\cdot)$. The forward propagation process of the network can be defined as:

$$v_i^{l+1}(t) = \sum_{j=1}^{N^l} w_{ij} s_j^l(t) + v_i^{l+1}(t-1) f_d(s_i^{l+1}(t-1)) + b_i^{l+1}, \tag{11}$$

$$s_i^{l+1}(t) = f_s(v_i^{l+1}(t)), \tag{12}$$

$$f_d(s(t)) = \begin{cases} D & s(t) = 0 \\ 0 & s(t) = 1, \end{cases} \tag{13}$$

where b_i^{l+1} is an adjustable bias that is learned to mimic a dynamic threshold behavior. However, the biases of this model are static during forwarding propagation. In contrast, the proposed dynamic threshold is dynamic and automatically adapts to membrane potentials.

Supplementary Note 2: Related Mathematical Definitions

In this section, we provide mathematical definitions for DT1, DT2, and Loihi weight transferring. In addition, we formally define the proposed homeostasis metrics.

DT1 Hao *et al.* [24] noted that neurons do not (or barely) fire when their thresholds are too large, which can negatively affect model performance. Therefore, they proposed DT1 to slow threshold growth. For the i-th neuron in the l-th layer, DT1 at timestamp t is mathematically defined as:

$$\Theta^l_{DT1,i}(t) = \Theta_{const} + (-\Theta^l_{DT1,i}(t-1) + \frac{\Theta_{initial}}{|2\Theta^l_{DT1,i}(t-1) - \Theta_{initial}|} \sum_{i=1}^{n^l} s^l_i(t)), \tag{14}$$

where Θ_{const} and $\Theta_{initial}$ are two hyperparameters; the dynamic threshold is mainly controlled by a dynamic scaling factor $\frac{\Theta_{initial}}{|2\Theta^l_{DT1,i}(t-1)-\Theta_{initial}|}$; n^l is the total number of neurons in the l-th layer. For fair comparisons with other competing approaches, we apply grid search to find the optimal values of $\Theta_{initial}$ and Θ_{const} and ensure that the host SNNs of DT1 offer similar success rates (SRs) in the static obstacle avoidance task to those of other approaches. Based on the grid search, $\Theta_{initial}$ is set to 10.0; 0.5 and 0.2 are the optimal values of Θ_{const} for the LIF-based and SRM-based host SNNs, respectively.

DT2 Inspired by the observed homeostasis in biology, Kim *et al.* [26] proposed DT2 to maintain neurons' firing rates at a predefined constant target frequency. Mathematically, for the i-th neuron in the l-th layer, DT2 at timestamp t is defined as:

$$\Theta^l_{DT2,i}(t) = \Theta^l_{DT2,i}(t-1) + (\sum_{i=1}^{n^l} s^l_i(t)) - f^l_{target}) \times \Theta^l_{DT2,i}(t-1) \times \gamma, \tag{15}$$

where f^l_{target} is the predefined constant target frequency; γ is a homeostasis factor that determines the threshold changing rate. Based on our grid search, we set f^l_{target} to 85 (*i.e.*, 1/3 of 256) for the three 256-neuron layers and γ to 0.004 to achieve the same static obstacle avoidance performance as that of other competing SNNs.

8-Bit Loihi Weights In our weight uncertainty (WU) experiments, we scale and round up the learned floating-point synaptic weights to low-precision 8-bit weights. The weight scaling process is defined as:

$$r^l = \frac{w^{\text{Loihi}}_{\text{max}}}{w^l_{\text{max}}}, \tag{16}$$

$$w^{(l)(\text{Loihi})}_{ij} = \text{round}(r^l w^l_{ij}), \tag{17}$$

$$\Theta^{(l)(\text{Loihi})}_i(t) = \text{round}(r^l \Theta^l_i(t)), \tag{18}$$

$$v^l_i(t) = v^{(l)(\text{Loihi})}_i(t)/r^l, \tag{19}$$

where r^l is the rescaling ratio of layer l; w^{Loihi}_{max} is the maximum weight that Loihi supports; w^l_{max} is the maximum weight of the l-th layer of the host SNN; w^l_{ij} is the synaptic weight between the i-th neuron in the l-th layer and the j-th neuron in the $(l-1)$-th layer, and $w^{(l)(\text{Loihi})}_{ij}$ is the corresponding rescaled weight on Loihi; $\Theta^l_i(t)$ and $\Theta^{(l)(\text{Loihi})}_i(t)$ are the original membrane threshold and the corresponding threshold for Loihi of the i-th neuron in the l-th layer at timestamp t, respectively; round(x) is a rounding function that returns the rounded version of x. Notably, to estimate $\Theta^l_i(t)$, we need to know the original membrane potentials. However, all the membrane potentials on Loihi are rescaled. Therefore, to obtain the original membrane potentials, we need to reverse the process defined in Eq 19.

Homeostasis Metrics We leverage three statistical metrics to quantify the homeostasis of an SNN. Mathematically, they are defined as follows:

$$\text{FR}_m = \mu(\text{FR}^p_m) \quad \text{for} \quad p = 1, 2, ..., P, \tag{20}$$

$$\text{FR}^m_{std} = \mu(\text{FR}^p_{std}) \quad \text{for} \quad p = 1, 2, ..., P, \tag{21}$$

$$\text{FR}^s_{std} = \sigma(\text{FR}^p_{std}) \quad \text{for} \quad p = 1, 2, ..., P, \tag{22}$$

$$\text{FR}^p_m = \mu(f^{l,p}_i) \quad \text{for } i = 1, 2, ..., N^l \quad l = 1, 2, ..., L, \tag{23}$$

$$\text{FR}^p_{std} = \sigma(f^{l,p}_i) \quad \text{for } i = 1, 2, ..., N^l \quad l = 1, 2, ..., L, \tag{24}$$

$$f^{l,p}_i = \frac{\sum_{t^p=1}^{T^p} s^l_i(t^p)}{T^p}, \tag{25}$$

where, T^p is the time taken for the p-th trial and $f^{l,p}_i$ is the firing rate of the i-th neuron in the l-th layer during the p-th trial. FR^p_m denotes the mean firing rate of all neurons of an SNN during the p-th trial, and FR^p_{std} is the standard deviation of all neuron firing rates for an SNN during the p-th trial. The definitions of FR_m, FR^m_{std}, and FR^s_{std} are defined in the main paper.

Supplementary Note 3: Biological Concepts

A typical biological neuron has four morphological regions: a cell body, dendrites, an axon, and synaptic terminals. Inside the cell body (*i.e.*, the soma), a nuclear envelope contains the cell's genes. The short tree-like dendrites branch out from the cell body, and they are the main apparatus for receiving incoming signals from other neurons. The long tubular axon covered by the myelin sheath extends some distance from the cell body and carries action potential to other neurons through synaptic terminals. A typical nerve neuron is illustrated in Figure 4a.

In a nerve cell that is at rest, the extracellular surface of the cell membrane has an excess positive charge, while the cytoplasmic side has an excess negative charge. The cell membrane maintains the separation of charge as a barrier against the diffusion of ions; see Figure 4b. The electrical potential difference across the membrane is defined as the membrane potential, which has three different statuses: resting, depolarization, and hyperpolarization. At rest, no net charge movement across the membrane occurs, and the resting membrane potential is maintained. By convention, the potential outside the cell is defined as zero, and hence, the resting potential is a negative value. A net flow of cations or anions into or out of a cell disturbs the resting membrane, causing depolarization or hyperpolarization, respectively. Depolarization indicates less negative membrane potential, while hyperpolarization signifies more negative potential; see Figure 4c.

Dynamic Thresholds in Biological Neurons

The Hodgkin-Huxley model [46] has served as an archetype for compartmental models of the electrophysiology of biological membranes (see Figure 4a). Many numerical methods leverage the Hodgkin-Huxley model as their testbeds, which can be applied to more complex models [17, 47, 48]. We use the Hodgkin-Huxley model to introduce the concept of threshold in biological neurons. Based on the Hodgkin-Huxley model, an action potential is produced when the membrane potential is higher than a particular threshold; this involves the following sequence of processes. First, when the membrane potential is higher than a threshold, the associated depolarization opens sodium (Na^+) channels, resulting in an inward Na^+ current. By discharging the membrane capacitance, the inward current causes further depolarization and the opening of more Na^+ channels, resulting in a further increase in the inward current. Second, under prolonged depolarization, the voltage-gated Na^+ channels become inactive. Furthermore, after some delay, the voltage-gated potassium (K^+) channels begin to open, causing an outward K^+ current that tends to repolarize the membrane (see Figure 4b). The second process underlies the absolute refractory period [49], a period during which no action potential can be elicited. After that, with some K^+ channels being closed and some Na^+ channels recovering from inactivation, the membrane enters a relative refractory period [49] (see Figure 4c). During this period, it is possible to trigger an action potential, but this requires a higher threshold.

Note that a small subthreshold depolarization cannot trigger an action potential, as it not only increases the inward Na^+ current but also increases the outward K^+ current [46]. Only at a specific membrane potential value does the net ionic current become inward, depositing a net positive charge on the inside of the membrane capacitance. This specific value is the potential (or spike) threshold [50].

The threshold changes dynamically, widely observed in the different nervous systems [14, 15, 16, 17, 18, 19, 20, 21, 22]. A thread of studies leverage the Hodgkin-Huxley model to verify the observed threshold dynamics [51, 17]. However, not all spike initiation dynamics of biological neurons can be accurately described by the Hodgkin-Huxley model [52].

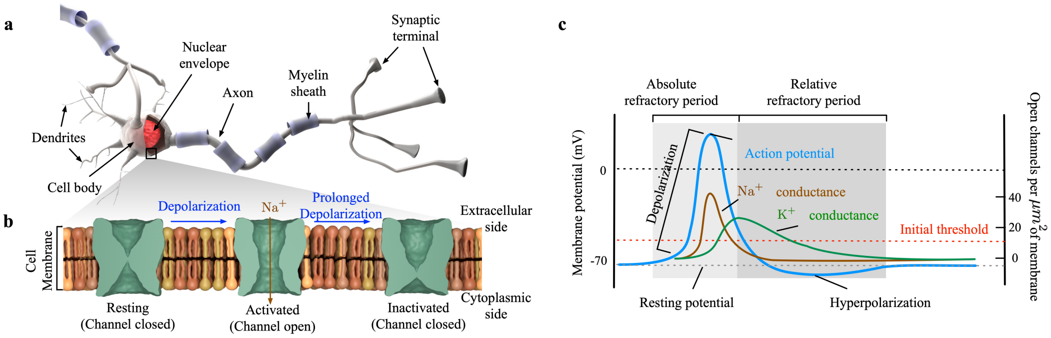

Figure 4: a. The structure of a neuron. b. We demonstrate a sodium (Na^+) voltage-gated channel under resting, activated, and inactivated (refractory) states. The Na^+ channel enters an inactivated state after depolarization and returns to a resting state only after the membrane potential is restored to its resting potential. c. The process of action potential generation, which is based on the Hodgkin-Huxley model [46], involves the sequential opening of voltage-gated Na^+ and K^+ channels. The Na^+ and K^+ conductance curves are adapted from the Hodgkin-Kuxley model [46].

Supplementary Note 4: Additional Details on Obstacle Avoidance Experiments

Experimental Setup

In the obstacle avoidance experiments, our evaluation baseline model and test environment are modified variants of the spiking actor network (SAN) [9] and its original simulated test environment, respectively. The SAN is a part of the spiking deep deterministic policy gradient (SDDPG) framework [9], which is a fully connected four-layer SNN (*i.e.*, three 256-neuron hidden layers and one two-neuron output layer). This network maps a state s of a robot to a control action a. Specifically, a state $s = \{G_{dis}, G_{dir}, \nu, \omega, L\}$ is encoded into 24 Poisson spike trains as inputs of the SAN, and each spike train has T timesteps. G_{dis} and G_{dir} are the relative 1-D distances from the robot to the goal and a 2-D direction (*i.e.*, right and left directions), respectively; ν and ω are the robot's 1-D linear and 2-D angular velocities (*i.e.*, rightward and leftward angular velocities); L denotes the distance measurements obtained from a Robo Peak light detection and ranging (RPLIDAR) laser range scanner (range: 0.2-40 m), which has a field of view of 180 degrees with 18 range measurements, each with a 10-degree resolution. The two output spike trains are decoded to control the robot via an action $a = \{\nu_L, \nu_R\}$, where ν_L and ν_R are the left and right wheel speeds of the differential-drive mobile robot, respectively [9].

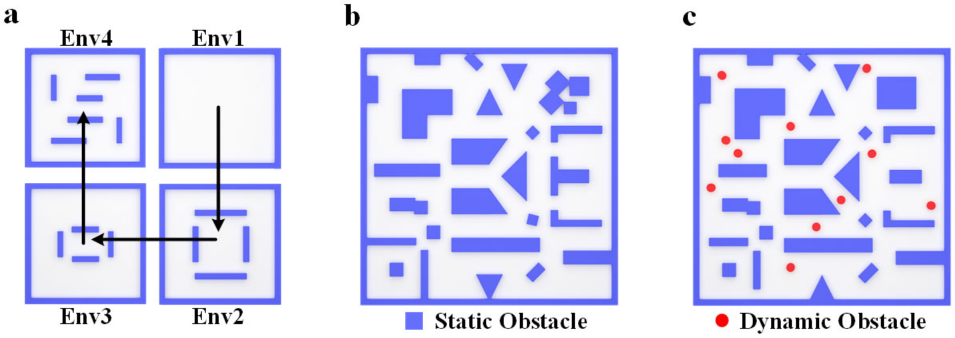

Figure 5: Illustrations of the training, static testing and dynamic testing environments. a. The training environments of the obstacle avoidance tasks. The training processes of all competing SNNs start from Env1 and end with Env4. b. Static testing environment. c. Dynamic testing environment. In addition to static obstacles, 11 dynamic obstacles are inserted.

Training

The SAN and its modified versions are trained with the original SDDPG framework. The training environments consist of four different maps, as shown in Figure 5a. In particular, during the training process, we set 100, 200, 300, and 400 start-goal pairs in the Env1, Env2, Env3, and Env4, respectively. The training starts from Env1 and ends with Env4. Following the training protocol described by Tang *et al.* [9], the hyperparameters related to training are set as follows: $D = 0.75$ for the LIF; $\eta = 0.01$ and $\psi = 4.0$ for the dynamic energy threshold (DET); $C = 3.0$ for the dynamic temporal threshold (DTT); $\tau_s = \tau_r = 1.0$ for the SRM; collision reward $= -20$; goal reward $= 30$; step reward $= 15$; goal l_2 distance threshold $= 0.5$ m; obstacle l_2 distance threshold $= 0.35$ m; ϵ ranges for Env1 to Env4 of $(0.9, 0.1)$, $(0.6, 0.1)$, $(0.6, 0.1)$, and $(0.6, 0.1)$, respectively; and corresponding ϵ-decays of 0.999 for the four environments. During the training procedure, we set the batch size to 256 and the learning rates to 0.00001 for both the actor and critic networks. We use PyTorch [53] to train and test all competing SNNs with an i7-7700 CPU and an NVIDIA GTX 1080Ti GPU. We direct the readers to the SAN algorithm [9] for details.

Assessment—Success Rate

We evaluate the obstacle avoidance capabilities of the proposed method by using SR as a metric. The SR is the percentage of successful passes out of 200 trials. A successful pass is a trial in which the robot can reach its destination without touching any static or dynamic obstacle within 1000 steps. In addition to the SR, we also report the overtime percentage (OTP), the percentage of overtime trials out of the total trials (*i.e.*, 200 trials), where overtime is defined as a trial in which the robot cannot reach the goal within 1000 steps but does not touch any obstacle.

We use the Gazebo simulator to construct a $20 \times 20\ m^2$ static test environment (see Figure 5b) and adopt the randomly sampled 200 start-goal location pairs used for testing the SAN [9]. For fairness, we apply grid searches on all tunable hyperparameters to ensure that the SRs of all competing approaches are relatively the same (*i.e.*, within $\pm 2\%$) when testing in the static testing environment. The quantitative experimental results obtained by all competing dynamic threshold methods in the static obstacle avoidance tasks are shown in Table 1. Due to space limits, we only report the experimental results based on $T = 5$ in the main manuscript. Here, we offer the quantitative performance with both $T = 5$ and $T = 25$.

Based on Table 1, compared to the SRs obtained with $T = 5$, the SRs obtained under $T = 25$ only change slightly (*i.e.*, $\pm 0.5\%$), indicating that all competing SNNs are not sensitive to the T value in static obstacle avoidance tasks. The observations related to the T value also hold in the dynamic obstacle experiments (see Table 2). However, we observe that the SRs decrease as the T value increases for most degraded input and weight uncertainty experiments.

Tables 2, 3, and 4 show the quantitative performance of all competing approaches in dynamic obstacle, degraded inputs, and weight uncertainty experiments, respectively. The SRs are also shown in Figures 7a and b. In terms of the OTPs, we witness high levels of overtime trials in the "0.2" section of the degraded inputs condition and the "GN weight" section of the weight uncertainty condition. As discussed in the main manuscript, with the "0.2" setup, the three disturbed lasers generate more spikes than they are supposed to, making the robot more cautious. Thus, the robot's speed slows, leading to more overtime trials. As expected, adding Gaussian noise to the learned weights reduces the effectiveness of the avoidance policy. However, our approach faces the most negligible impact, offering the best SRs under all experimental results. In the following, we provide a more detailed analysis for each degraded condition.

Dynamic Obstacles As discussed, we introduce 11 dynamically moving cylinders to the static testing environment; see Figure 5c. Table 2 shows the corresponding experimental results obtained under this condition. Our approach delivers the highest SRs with both the LIF and SRM neuron models. Notably, under both the $T = 5$ and $T = 25$ settings, the proposed approach outperforms the runners-up by significant margins (by at least 9% over the LIF model and 12% over the SRM). The results demonstrate that the proposed bioplausible dynamic threshold scheme provides substantial environmental adaptability to the host SNNs. Since all the synaptic weights and static thresholds of both the SAN and the SAN with no resting operation (SAN-NR) are learned from the environments with static obstacles only, we expect that they cannot adapt well to an environment with dynamic objects. Surprisingly, compared to the static threshold scheme, the two heuristic dynamic threshold schemes, DT1 and DT2, obtain lower SRs. The threshold dynamics provided by DT1 rely on two hyperparameters, the constant potential and initial potential. DT2 requires a target firing count to be set. These hyperparameters are justified during the training process but fixed during testing. We believe that these hyperparameters dramatically impact the adaptability of the tested heuristic dynamic threshold schemes. In contrast, the dynamics offered by the proposed bioinspired dynamic energy-temporal threshold (BDETT) scheme are dynamically based on layerwise statistical cues.

Degraded Inputs In this experiment, as discussed in our main manuscript, in addition to the presence of dynamic obstacles, we disturb the obtained range measurements in three different ways: a) "0.2": we set the 3rd, 9th, and 15th laser ranges to 0.2 m. In this case, the three modified measurements always report obstacles in their perception fields even when none are present; b) "6.0": the ranges of the same three lasers are set to 6.0 m, which is the average visible range in the test environment. This means that the three lasers cannot perceive any object; c) "GN": we add Gaussian noise (*i.e.*, $clip(s_{input} + \mathcal{N}(0, 1.0), 0.2, 6.0)$, as suggested in a study regarding long short-term memory with a local map critic (LSTM-LMC) [43]) to each of the 18 range measurements. The experimental results obtained under these settings are shown in Table 3.

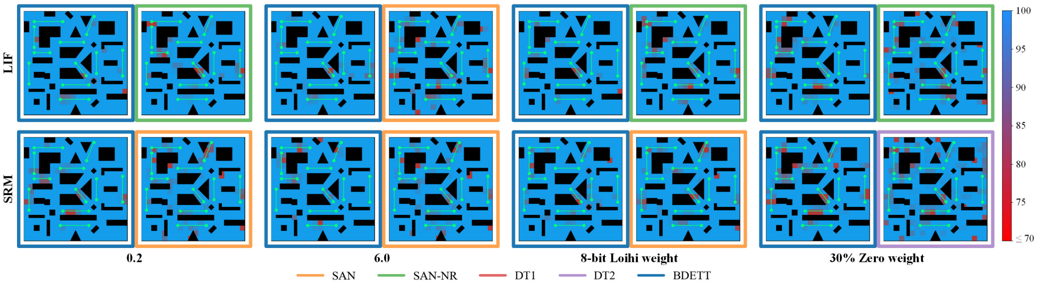

Figure 6: The success rate heatmaps. The heatmaps yielded by the best and runner-up performers under four different conditions indicate the areas with lower success rates (*i.e.*, those shown in red).

In all degraded input experiments, the SRs offered by our BDETT scheme still remain the highest and outperform the runners-up by at least 10%. This reflects that the proposed dynamic threshold scheme provides the host SNNs with strong adaptability to all designed degraded inputs, which is highly desired and appreciated in mobile robot applications. In the "0.2" setup, the three disturbed lasers generate more spikes than they are supposed to, making the robot more cautious. All host SNNs obtain lower SRs than those obtained under the dynamic

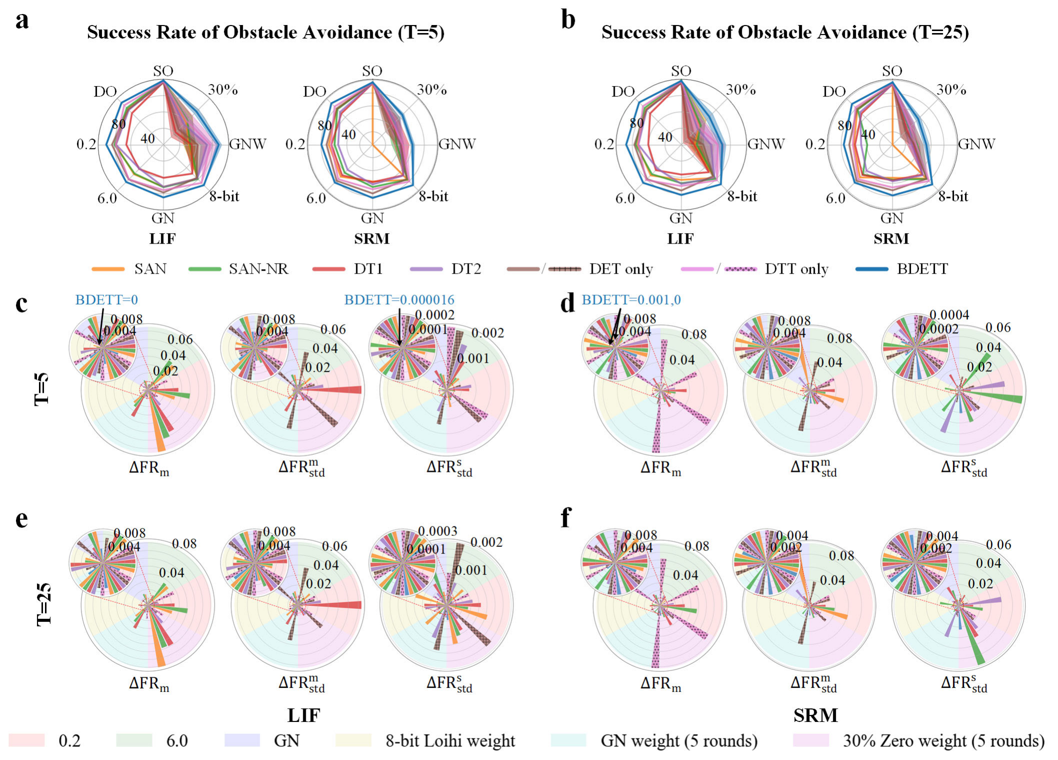

Figure 7: The experimental results obtained for the robot obstacle avoidance tasks. a & b. The SRs of obstacle avoidance under normal and different degraded conditions under the $T = 5$ and $T = 25$ settings, respectively. 'SO' denotes the static obstacle condition; 'DO' represents the dynamic obstacle condition; '0.2', '6.0', and 'GN' are the three degraded input conditions; '8-bit', 'GNW', and '30%' denotes the 8-bit Loihi weights, GN weights, and 30% zero weights, respectively. c & d. Homeostasis measurements obtained with the $T = 5$ setting by the LIF- and SRM-based host SNNs. e & f. The homeostasis results obtained with the $T = 25$ setup by the LIF- and SRM-based host SNNs, respectively.

obstacle settings. However, the SRM-based host SNNs are significantly impacted. We believe the reason for this is that the spike response and refractory kernels of the SRM amplify the intense spikes triggered by the three modified measurements. In the "6.0" experiments, the three modified measurements trigger fewer spikes, and the robot becomes more relaxed due to the 'hyperopia' effect. Therefore, we expect more failed passes than in the "0.2" setup. This is true for the LIF-based SNNs but not for the SRM-based SNNs. Since the three farsighted lasers reduce the signal amplification effects caused by the two kernels, the SRM-based SNNs perform better here than in the "0.2" experiments. Under the "GN" condition, the SRs of all competing host SNNs decrease, but our approach is the least affected and induces the lowest SR drops. Remarkably, under all degraded input experiments, the proposed BDETT improves upon the SRs of both the LIF-based and SRM-based baseline models (*i.e.*, SAN-NR) by at least 10% and 17%, respectively. The success rates of the best and runner-up performers under "0.2" and "6.0" conditions are also qualitatively illustrated in Figure 6.

Weight Uncertainty Neuromorphic hardware (*e.g.*, Loihi) achieves computing efficiency by sacrificing the weight precision, and an 8-bit integer normally yields the highest precision. Therefore, when deploying an SNN on neuromorphic hardware, one needs to scale and round up the learned floating-point synaptic weights to low-precision weights. We mimic this scenario by mapping the learned weights to Intel's Loihi 8-bit integer weights. The mapping equations are provided in Supplementary Note 2. In addition, we design two extra weight pollution experiments. "GN weight" involves adding Gaussian noise (*i.e.*, $w_{ij} + \mathcal{N}(0, 0.05)$) to all synaptic weights; under "30% zero weight", we reandomly set 30% of the synaptic weights between every two adjacent layers to 0. To reduce the impact caused by the randomness introduced in the two additional experiments, we report the average SRs of 5-round tests.

As shown in Table 4, the proposed BDETT can effectively reduce the impact caused by degraded synaptic weights and deliver the best SRs under all experimental settings. Low-precision weight convergence slightly reduces the SRs of all competing host SNNs slightly. We observe that the effectiveness of the SRM-based SNNs

Table 1: Quantitative performance of obstacle avoidance with static obstacles.

	LIF ($T = 5$)		**SRM** ($T = 5$)		**LIF** ($T = 25$)		**SRM** ($T = 25$)	
Name	SR↑	OTP	SR↑	OTP	SR↑	OTP	SR↑	OTP
SAN	98%	0.0%	**96.5%**	0.0%	98%	0.0%	96%	0.0%
SAN-NR	98%	0.0%	95.5%	0.0%	**98.5%**	0.0%	95.5%	0.0%
DT1 [24]	96.5%	0.0%	95%	0.0%	96%	0.0%	94.5%	0.5%
DT2 [26]	97%	0.0%	95%	0.0%	97%	0.0%	94%	0.0%
DET only	96%	0.0%	95.5%	0.0%	95.5%	0.0%	95%	0.0%
DTT only	97%	0.0%	95.5%	0.0%	97%	0.0%	95%	0.0%
BDETT	**98.5%**	0.0%	**96.5%**	0.0%	98%	0.0%	**97%**	0.0%

Table 2: Quantitative performance of obstacle avoidance with dynamic obstacles.

	LIF ($T = 5$)		**SRM** ($T = 5$)		**LIF** ($T = 25$)		**SRM** ($T = 25$)	
Name	SR↑	OTP	SR↑	OTP	SR↑	OTP	SR↑	OTP
SAN	81.5%	0.0%	78.5%	0.0%	81%	0.0%	77.5%	0.0%
SAN-NR	83.5%	0.0%	77.5%	0.5%	83.5%	0.0%	77%	1.0%
DT1 [24]	74.5%	0.0%	68.5%	0.0%	74%	0.5%	68.5%	0.5%
DT2 [26]	80%	0.0%	71.5%	0.5%	80%	0.0%	71.5%	0.0%
DET only	81%	0.0%	78.5%	0.5%	80.5%	0.0%	78.5%	1.0%
DTT only	88%	0.0%	83.5%	0.0%	86.5%	0.0%	82%	0.0%
BDETT	**92.5%**	0.0%	**90.5%**	0.0%	**93%**	0.0%	**89.5%**	0.5%

is dramatically impacted by the “GN weight" and “30% zero weight" pollution settings, especially the SAN. This means that SRM-based models are more sensitive to weight changes than LIF-based SNNs. Again, under the three degraded conditions, the proposed BDETT increases the SRs of the baseline model SAN-NR by at least 9.5%, 14.2%, and 14.8%. The success rates of the best and runner-up performers under “8-bit Loihi weight” and “30% zero weight” conditions are also qualitatively illustrated in Figure 6.

Table 3: Quantitative performance of obstacle avoidance under degraded input conditions.

		LIF ($T = 5$)		**SRM** ($T = 5$)		**LIF** ($T = 25$)		**SRM** ($T = 25$)	
Type	**Name**	SR↑	OTP	SR↑	OTP	SR↑	OTP	SR↑	OTP
0.2	SAN	78.5%	1.5%	68%	1.0%	74%	2.5%	60.5%	1.0%
	SAN-NR	80%	2.5%	59%	3.0%	76%	3.0%	39.5%	4.5%
	DT1 [24]	65.5%	4.0%	64%	3.5%	60.5%	3.5%	58.5%	5.0%
	DT2 [26]	78%	3.0%	53.5%	3.5%	72.5%	4.0%	49%	3.5%
	DET only	83%	2.0%	71.5%	3.0%	77.5%	3.5%	67.5%	3.0%
	DTT only	78.5%	3.5%	64.5%	1.0%	72.5%	3.5%	62%	2.5%
	BDETT	**90%**	2.5%	**79.5%**	3.5%	**87.5%**	3.5%	**76%**	4.5%
6.0	SAN	71%	0.0%	70%	0.0%	73%	0.0%	72%	0.0%
	SAN-NR	70%	0.0%	61.5%	0.0%	71%	0.0%	65.5%	0.0%
	DT1 [24]	62%	0.0%	67%	0.0%	64%	0.0%	66.5%	0.0%
	DT2 [26]	61.5%	0.0%	55%	0.5%	61.5%	0.0%	57.5%	0.0%
	DET only	80%	0.0%	79%	0.0%	79.5%	0.0%	79.5%	0.0%
	DTT only	80%	0.0%	75.5%	0.0%	81%	0.0%	76%	0.0%
	BDETT	**84.5%**	0.0%	**83%**	0.0%	**86%**	0.0%	**83.5%**	0.0%
GN	SAN	71.5%	0.0%	57%	0.0%	63%	0.0%	51.5%	0.5%
	SAN-NR	72%	0.0%	65.5%	1.0%	67%	0.5%	54.5%	2.0%
	DT1 [24]	60.5%	0.5%	58%	0.0%	56.5%	0.5%	55.5%	0.0%
	DT2 [26]	71.5%	1.5%	61.5%	0.0%	68%	1.0%	57%	1.5%
	DET only	78.5%	1.0%	75.5%	0.5%	76%	2.0%	71%	0.5%
	DTT only	75.5%	0.0%	69%	0.0%	70.5%	0.0%	66.5%	0.5%
	BDETT	**84.5%**	0.0%	**82.5%**	0.0%	**81.5%**	0.0%	**79%**	0.5%

Table 4: Quantitative performance of obstacle avoidance under weight uncertainty conditions.

		LIF ($T = 5$)		**SRM** ($T = 5$)		**LIF** ($T = 25$)		**SRM** ($T = 25$)	
Type	**Name**	SR↑	OTP	SR↑	OTP	SR↑	OTP	SR↑	OTP
8-bit Loihi weight	SAN	78.5%	0.0%	77%	0.0%	77.5%	0.0%	74.5%	0.0%
	SAN-NR	79.5%	0.0%	76.5%	0.5%	79%	0.0%	75.5%	0.0%
	DT1 [24]	70%	0.0%	67%	0.0%	67.5%	0.5%	65.5%	0.0%
	DT2 [26]	78.5%	0.0%	67.5%	1.0%	77%	0.0%	67%	0.5%
	DET only	77.5%	0.0%	75%	0.0%	74.5%	0.5%	75%	0.0%
	DTT only	86%	0.0%	80.5%	0.0%	81.5%	0.0%	80%	0.0%
	BDETT	**90%**	0.0%	**88.5%**	0.0%	**88.5%**	0.0%	**87.5%**	0.0%
GN weight (5 rounds)	SAN	51.3% (σ-6.8)	1.2%	0% (σ-0)	20.3%	36.2% (σ-7.3)	3.6%	0% (σ-0)	18.4%
	SAN-NR	52.5% (σ-7.1)	1.6%	37.2% (σ-7.6)	2.4%	39.0% (σ-7.8)	3.1%	38.4% (σ-8.9)	2.1%
	DT1 [24]	54.6% (σ-7.9)	2.3%	44.9% (σ-11.4)	3.0%	35.7% (σ-9.2)	3.3%	32.4% (σ-9.6)	4.7%
	DT2 [26]	73.2% (σ-7.4)	2.1%	43.6% (σ-4.4)	2.4%	56.2% (σ-9.3)	2.0%	30.3% (σ-5.7)	2.9%
	DET only	61.8% (σ-12.0)	1.8%	43.3% (σ-4.6)	2.5%	47.0% (σ-11.3)	2.3%	34.1% (σ-5.4)	2.6%
	DTT only	77.1% (σ-8.8)	1.5%	46.4% (σ-8.0)	1.6%	64.7% (σ-8.0)	1.8%	36.8% (σ-10.2)	1.4%
	BDETT	**87.7%** (σ-3.3)	0.8%	**61.8%** (σ-2.9)	1.3%	**70.1%** (σ-4.2)	0.4%	**52.6%** (σ-4.0)	2.5%
30% Zero weight (5 rounds)	SAN	59.3% (σ-10.5)	0.0%	0% (σ-0)	17.7%	51.2% (σ-11.3)	0.0%	0% (σ-0)	19.4%
	SAN-NR	61.6% (σ-7.5)	0.0%	46.5% (σ-12.4)	0.0%	53.6% (σ-6.7)	0.0%	36.5% (σ-9.9)	0.2%
	DT1 [24]	41.2% (σ-7.7)	0.7%	44.3% (σ-11.7)	0.0%	32.2% (σ-9.1)	1.3%	31.7% (σ-12.6)	0.6%
	DT2 [26]	55.6% (σ-9.3)	0.3%	49.1% (σ-10.8)	0.8%	48.0% (σ-10.5)	0.0%	37.8% (σ-10.2)	1.1%
	DET only	46.2% (σ-8.5)	0.0%	39.8% (σ-11.5)	1.4%	33.6% (σ-8.5)	0.8%	29.3% (σ-12.3)	2.5%
	DTT only	60.6% (σ-9.5)	0.0%	45.4% (σ-7.4)	0.5%	50.3% (σ-8.7)	0.4%	38.8% (σ-8.0)	1.3%
	BDETT	**77.2%** (σ-3.6)	0.0%	**65.2%** (σ-2.7)	0.3%	**68.4%** (σ-5.2)	0.0%	**56.5%** (σ-4.3)	0.7%

Table 5: The raw homeostasis measurements of successful trials and the corresponding changes with respect to the baseline condition in obstacle avoidance tasks with the $T = 5$ setting.

		LIF ($T = 5$)			**SRM** ($T = 5$)		
Type	**Name**	$\mathrm{FR}_m(\Delta)$	$\mathrm{FR}^m_{std}(\Delta)$	$\mathrm{FR}^s_{std}(\Delta)$	$\mathrm{FR}_m(\Delta)$	$\mathrm{FR}^m_{std}(\Delta)$	$\mathrm{FR}^s_{std}(\Delta)$
Dynamic obstacle (baseline condition)	SAN	0.523	0.325	0.000891	0.278	0.301	0.000596
	SAN-NR	0.515	0.330	0.001029	0.487	0.309	0.006853
	DT1 [24]	0.443	0.325	0.001738	0.482	0.310	0.002942
	DT2 [26]	0.400	0.345	0.002136	0.418	0.309	0.005873
	DET only	0.508	0.336	0.001024	0.380	0.276	0.002564
	DTT only	0.456	0.320	0.000902	0.475	0.294	0.002284
	BDETT	0.439	0.312	0.000916	0.501	0.298	0.001759
0.2	SAN	0.556 (0.033)	0.329 (0.004)	0.000750 (0.000141)	0.293 (0.015)	0.247 (0.054)	0.000746 (0.000150)
	SAN-NR	0.565 (0.050)	0.326 (0.004)	0.000738 (0.000291)	0.508 (0.021)	0.333 (0.024)	0.001069 (0.005784)
	DT1 [24]	0.479 (0.036)	0.387 (0.062)	0.000973 (0.000765)	0.518 (0.036)	0.348 (0.038)	0.003993 (0.001051)
	DT2 [26]	0.412 (0.012)	0.370 (0.025)	0.001293 (0.000843)	0.438 (0.020)	0.325 (0.016)	0.001709 (0.004164)
	DET only	0.495 (0.013)	0.353 (0.017)	0.001523 (0.000499)	0.380 (0.000)	0.244 (0.032)	0.003969 (0.001405)
	DTT only	0.481 (0.025)	0.335 (0.015)	0.000768 (0.000134)	0.420 (0.055)	0.335 (0.041)	0.002040 (0.000244)
	BDETT	0.444 **(0.005)**	0.315 **(0.003)**	0.000851 **(0.000065)**	0.494 **(0.007)**	0.310 **(0.012)**	0.001884 **(0.000125)**
6.0	SAN	0.564 (0.041)	0.342 (0.017)	0.001454 (0.000563)	0.275 (0.003)	0.306 (0.005)	0.000747 (0.000151)
	SAN-NR	0.558 (0.043)	0.339 (0.009)	0.001548 (0.000519)	0.483 (0.004)	0.306 (0.003)	0.002533 (0.004320)
	DT1 [24]	0.432 (0.011)	0.318 (0.007)	0.002862 (0.001124)	0.471 (0.011)	0.315 (0.005)	0.004224 (0.001282)
	DT2 [26]	0.407 (0.007)	0.354 (0.009)	0.003785 (0.001649)	0.408 (0.010)	0.320 (0.011)	0.005417 (0.000456)
	DET only	0.515 (0.007)	0.374 (0.038)	0.003088 (0.002064)	0.377 (0.003)	0.230 (0.046)	0.003889 (0.001325)
	DTT only	0.450 (0.006)	0.325 (0.005)	0.003057 (0.002155)	0.403 (0.072)	0.290 (0.004)	0.002855 (0.000571)
	BDETT	0.440 **(0.001)**	0.317 **(0.005)**	0.000960 **(0.000044)**	0.501 **(0.000)**	0.300 **(0.002)**	0.001870 **(0.000111)**
GN	SAN	0.534 (0.011)	0.310 (0.015)	0.001416 (0.000525)	0.287 (0.009)	0.204 (0.097)	0.000727 (0.000131)
	SAN-NR	0.527 (0.012)	0.314 (0.016)	0.001622 (0.000593)	0.497 (0.010)	0.298 (0.011)	0.007395 (0.000542)
	DT1 [24]	0.451 (0.008)	0.319 (0.006)	0.000982 (0.000756)	0.502 (0.020)	0.302 (0.008)	0.004634 (0.001692)
	DT2 [26]	0.405 (0.005)	0.337 (0.008)	0.001654 (0.000482)	0.423 (0.005)	0.283 (0.026)	0.005578 (0.000295)
	DET only	0.502 (0.006)	0.361 (0.025)	0.001228 (0.000204)	0.390 (0.010)	0.283 (0.007)	0.002454 (0.000110)
	DTT only	0.445 (0.011)	0.315 (0.005)	0.001453 (0.000551)	0.390 (0.085)	0.308 (0.014)	0.001968 (0.000316)
	BDETT	0.443 **(0.004)**	0.307 **(0.005)**	0.000880 **(0.000036)**	0.500 **(0.001)**	0.301 **(0.003)**	0.001886 **(0.000127)**
8-bit Loihi weight	SAN	0.520 (0.003)	0.319 (0.006)	0.000988 (0.000097)	0.288 (0.010)	0.314 (0.013)	0.000731 **(0.000135)**
	SAN-NR	0.513 (0.002)	0.334 **(0.004)**	0.001286 (0.000257)	0.479 (0.008)	0.315 (0.006)	0.005527 (0.001326)
	DT1 [24]	0.437 (0.006)	0.319 (0.006)	0.001589 (0.000149)	0.473 (0.009)	0.325 (0.015)	0.002084 (0.000858)
	DT2 [26]	0.407 (0.007)	0.340 (0.005)	0.001853 (0.000283)	0.425 (0.007)	0.316 (0.007)	0.005638 (0.000235)
	DET only	0.505 (0.003)	0.341 (0.005)	0.000868 (0.000156)	0.385 (0.005)	0.285 (0.009)	0.002185 (0.000379)
	DTT only	0.446 (0.010)	0.325 (0.005)	0.000787 (0.000115)	0.460 (0.015)	0.301 (0.007)	0.002436 (0.000152)
	BDETT	0.439 **(0.000)**	0.308 **(0.004)**	0.000932 **(0.000016)**	0.500 **(0.001)**	0.293 **(0.005)**	0.001570 (0.000189)
GN weight (5 rounds)	SAN	0.501 (0.022)	0.338 (0.013)	0.001192 (0.000301)	-	-	-
	SAN-NR	0.490 (0.025)	0.338 (0.008)	0.001281 (0.000252)	0.498 (0.011)	0.326 (0.017)	0.004461 (0.002392)
	DT1 [24]	0.407 (0.036)	0.349 (0.024)	0.001902 (0.000164)	0.487 (0.005)	0.319 (0.009)	0.003573 (0.000631)
	DT2 [26]	0.391 (0.009)	0.335 (0.010)	0.001356 (0.000780)	0.410 (0.008)	0.291 (0.018)	0.010002 (0.004129)
	DET only	0.516 (0.008)	0.375 (0.039)	0.002225 (0.001201)	0.399 (0.019)	0.212 (0.064)	0.003675 (0.001111)
	DTT only	0.467 (0.011)	0.327 (0.007)	0.001244 (0.000342)	0.387 (0.088)	0.301 (0.007)	0.002387 (0.000103)
	BDETT	0.444 **(0.005)**	0.318 **(0.006)**	0.001013 **(0.000097)**	0.498 **(0.003)**	0.299 **(0.001)**	0.001602 **(0.000157)**
30% Zero weight (5 rounds)	SAN	0.448 (0.075)	0.321 **(0.004)**	0.001490 (0.000599)	-	-	-
	SAN-NR	0.454 (0.061)	0.335 (0.005)	0.001399 (0.000370)	0.470 (0.017)	0.331 (0.022)	0.009941 (0.003088)
	DT1 [24]	0.387 (0.056)	0.313 (0.012)	0.002045 (0.000307)	0.456 (0.026)	0.334 (0.024)	0.005103 (0.002161)
	DT2 [26]	0.377 (0.023)	0.358 (0.013)	0.001834 (0.000302)	0.403 (0.015)	0.332 (0.023)	0.003469 (0.002404)
	DET only	0.520 (0.012)	0.387 (0.051)	0.002582 (0.001558)	0.356 (0.024)	0.235 (0.041)	0.005036 (0.002472)
	DTT only	0.470 (0.014)	0.337 (0.017)	0.002551 (0.001649)	0.394 (0.081)	0.274 (0.020)	0.003706 (0.001422)
	BDETT	0.444 **(0.005)**	0.316 **(0.004)**	0.000993 **(0.000077)**	0.497 **(0.004)**	0.318 **(0.020)**	0.003855 **(0.002096)**

Assessment—Homeostatic

In the main manuscript, we show the quantified homeostasis changes induced during all successful trials with respect to the base condition (*i.e.*, the homeostasis obtained in the dynamic obstacle experiments) under the $T = 5$ setting. Table 5 provides the raw homeostasis measurements and the corresponding changes used for plotting the polar chart in the main manuscript.

We also offer the measured homeostasis and corresponding changes obtained under the $T = 25$ setting in Table 6. The corresponding polar plots are shown in Figures 7e and f. Similar to the observations obtained in the experiments with $T = 5$, the association between homeostasis and the obstacle avoidance SR still holds for $T = 25$; stronger homeostasis offers better performance. Notably, our approach induces the smallest changes in the three metrics across all experimental settings, except for the $\Delta\mathrm{FR}^m_{std}$ obtained in the SRM-based 8-bit Loihi weight experiment. Furthermore, the proposed BDETT delivers the best obstacle avoidance SRs in all designed experimental conditions with $T = 25$; see Tables 2, 3, and 4 and Figures 7e and f.

Table 6: The raw homeostasis measurements of successful trials and the corresponding changes with respect to the baseline condition in obstacle avoidance tasks with the $T = 25$ setting.

		LIF ($T = 25$)			**SRM** ($T = 25$)		
Type	**Name**	$\mathrm{FR}_m(\Delta)$	$\mathrm{FR}^m_{std}(\Delta)$	$\mathrm{FR}^s_{std}(\Delta)$	$\mathrm{FR}_m(\Delta)$	$\mathrm{FR}^m_{std}(\Delta)$	$\mathrm{FR}^s_{std}(\Delta)$
Dynamic obstacle (baseline condition)	SAN	0.524	0.325	0.002098	0.285	0.306	0.001504
	SAN-NR	0.515	0.331	0.001440	0.488	0.312	0.003486
	DT1 [24]	0.447	0.324	0.002155	0.485	0.319	0.002473
	DT2 [26]	0.401	0.347	0.002470	0.420	0.314	0.005771
	DET only	0.509	0.336	0.001206	0.384	0.279	0.002483
	DTT only	0.449	0.317	0.001736	0.475	0.301	0.002515
	BDETT	0.439	0.311	0.001035	0.501	0.301	0.002394
0.2	SAN	0.556 (0.032)	0.330 (0.005)	0.000798 (0.001300)	0.294 (0.009)	0.245 (0.061)	0.001004 (0.000500)
	SAN-NR	0.566 (0.051)	0.326 (0.005)	0.000768 (0.000670)	0.536 (0.048)	0.351 (0.039)	0.001224 (0.002260)
	DT1 [24]	0.481 (0.034)	0.387 (0.063)	0.001097 (0.001060)	0.517 (0.032)	0.344 (0.025)	0.003823 (0.001350)
	DT2 [26]	0.415 (0.014)	0.370 (0.023)	0.001662 (0.000810)	0.438 (0.018)	0.326 (0.012)	0.001781 (0.003990)
	DET only	0.496 (0.013)	0.353 (0.017)	0.001424 (0.000218)	0.380 (0.004)	0.246 (0.033)	0.003362 (0.000879)
	DTT only	0.485 (0.036)	0.336 (0.019)	0.000972 (0.000760)	0.420 (0.055)	0.335 (0.034)	0.003116 (0.000601)
	BDETT	0.444 **(0.005)**	0.314 **(0.003)**	0.001104 **(0.000069)**	0.494 **(0.007)**	0.310 **(0.010)**	0.002238 **(0.000160)**
6.0	SAN	0.558 (0.034)	0.341 (0.016)	0.001415 (0.000683)	0.275 (0.010)	0.303 (0.003)	0.000839 (0.000665)
	SAN-NR	0.550 (0.035)	0.340 (0.009)	0.001165 (0.000275)	0.484 (0.004)	0.306 (0.006)	0.003828 (0.000342)
	DT1 [24]	0.433 (0.014)	0.319 **(0.005)**	0.002449 (0.000294)	0.471 (0.014)	0.315 (0.004)	0.003621 (0.001148)
	DT2 [26]	0.406 (0.005)	0.353 (0.006)	0.003206 (0.000736)	0.406 (0.014)	0.320 (0.006)	0.005094 (0.000667)
	DET only	0.516 (0.007)	0.374 (0.038)	0.003172 (0.001966)	0.377 (0.007)	0.241 (0.038)	0.003248 (0.000765)
	DTT only	0.450 (0.001)	0.327 (0.010)	0.002744 (0.001008)	0.413 (0.062)	0.289 (0.012)	0.002742 (0.000227)
	BDETT	0.440 **(0.001)**	0.316 **(0.005)**	0.001247 **(0.000212)**	0.501 **(0.000)**	0.300 **(0.001)**	0.002092 **(0.000302)**
GN	SAN	0.538 (0.014)	0.311 (0.014)	0.002522 (0.000424)	0.288 **(0.003)**	0.209 (0.097)	0.000883 (0.000621)
	SAN-NR	0.530 (0.015)	0.318 (0.013)	0.003055 (0.001615)	0.503 (0.015)	0.299 (0.013)	0.005029 (0.001543)
	DT1 [24]	0.452 (0.005)	0.319 **(0.005)**	0.001408 (0.000747)	0.502 (0.017)	0.302 (0.017)	0.004470 (0.001997)
	DT2 [26]	0.405 **(0.004)**	0.336 (0.011)	0.002005 (0.000465)	0.423 **(0.003)**	0.283 (0.031)	0.005235 (0.000536)
	DET only	0.500 (0.009)	0.367 (0.031)	0.001424 (0.000218)	0.392 (0.008)	0.286 (0.007)	0.003130 (0.000647)
	DTT only	0.445 (0.004)	0.311 (0.006)	0.001260 (0.000476)	0.401 (0.074)	0.310 (0.009)	0.001795 (0.000720)
	BDETT	0.443 **(0.004)**	0.306 **(0.005)**	0.001085 **(0.000050)**	0.498 **(0.003)**	0.301 **(0.000)**	0.001962 **(0.000432)**
8-bit Loihi weight	SAN	0.519 (0.005)	0.319 (0.006)	0.001106 (0.000992)	0.290 (0.005)	0.315 (0.009)	0.000961 (0.000543)
	SAN-NR	0.510 (0.005)	0.336 (0.005)	0.001153 (0.000287)	0.478 (0.010)	0.317 (0.005)	0.005220 (0.001734)
	DT1 [24]	0.435 (0.012)	0.318 (0.006)	0.001467 (0.000688)	0.472 (0.013)	0.327 (0.008)	0.001854 (0.000619)
	DT2 [26]	0.411 (0.010)	0.339 (0.008)	0.001802 (0.000668)	0.428 (0.008)	0.317 **(0.003)**	0.005032 (0.000739)
	DET only	0.504 (0.005)	0.341 (0.005)	0.000924 (0.000282)	0.387 (0.003)	0.288 (0.009)	0.002816 (0.000333)
	DTT only	0.446 (0.003)	0.327 (0.010)	0.001302 (0.000434)	0.462 (0.013)	0.303 (0.002)	0.002038 (0.000477)
	BDETT	0.439 **(0.000)**	0.308 **(0.003)**	0.001107 **(0.000072)**	0.502 **(0.001)**	0.297 (0.004)	0.002164 **(0.000230)**
GN weight (5 rounds)	SAN	0.503 (0.021)	0.340 (0.015)	0.001241 (0.000857)	-	-	-
	SAN-NR	0.488 (0.027)	0.340 (0.009)	0.001796 (0.000356)	0.499 (0.011)	0.328 (0.016)	0.004204 (0.000718)
	DT1 [24]	0.413 (0.034)	0.346 (0.022)	0.002318 (0.000163)	0.491 (0.006)	0.320 **(0.001)**	0.003065 (0.000592)
	DT2 [26]	0.390 (0.011)	0.336 (0.011)	0.001382 (0.001088)	0.411 (0.007)	0.292 (0.017)	0.008895 (0.003124)
	DET only	0.516 (0.007)	0.372 (0.036)	0.002643 (0.001437)	0.397 (0.013)	0.218 (0.061)	0.003384 (0.000901)
	DTT only	0.467 (0.018)	0.328 (0.011)	0.001352 (0.000384)	0.393 (0.082)	0.303 (0.002)	0.003581 (0.001066)
	BDETT	0.444 **(0.005)**	0.318 **(0.007)**	0.001163 **(0.000128)**	0.498 **(0.003)**	0.300 **(0.001)**	0.001923 **(0.000471)**
30% Zero weight (5 rounds)	SAN	0.443 (0.081)	0.320 **(0.005)**	0.000862 (0.001236)	-	-	-
	SAN-NR	0.457 (0.058)	0.338 (0.007)	0.002461 (0.001021)	0.472 (0.016)	0.332 (0.020)	0.009380 (0.005894)
	DT1 [24]	0.388 (0.059)	0.312 (0.012)	0.001257 (0.000898)	0.456 (0.029)	0.334 **(0.015)**	0.005820 (0.003347)
	DT2 [26]	0.372 (0.029)	0.359 (0.012)	0.001930 (0.000540)	0.403 (0.017)	0.332 (0.018)	0.003060 (0.002711)
	DET only	0.524 (0.015)	0.368 (0.032)	0.002974 (0.001786)	0.360 (0.024)	0.245 (0.034)	0.004378 (0.001895)
	DTT only	0.472 (0.023)	0.334 (0.017)	0.002013 (0.000277)	0.402 (0.073)	0.280 (0.021)	0.004420 (0.001905)
	BDETT	0.446 **(0.007)**	0.316 **(0.005)**	0.001502 **(0.000467)**	0.488 **(0.013)**	0.316 **(0.015)**	0.004638 **(0.002244)**

Assessment—Ablation Studies

We conduct ablation studies to validate the effectiveness of the DET and DTT components of the proposed BDETT. The results obtained under different degraded conditions are reported in the rows named "DET only" and "DTT only" in Tables 1, 2, 3, and 4. The ablation study results are also illustrated in Figure 7. All listed evaluations validate that the BDETT scheme performs better than any single component. The dynamic threshold scheme with only one component cannot effectively regulate the firing rate statuses of the host SNNs, prohibiting meaningful homeostasis. One extreme example is illustrated in Figure 7f. The $\Delta\mathrm{FR}_m$ changes induced under the 'DTT only' setting are the largest among all competing approaches under all experimental conditions. Notably, when combined with the other component, the proposed BDETT provides the strongest homeostasis for the host SNNs. This validates the biologically observed positive correlation encoded by the DET and the negative correlation enforced by the DTT, which are equally essential for effectively maintaining the homeostasis of an SNN.

Supplementary Note 5: Additional Details on Continuous Control Experiments—HalfCheetah-v3

Training

The adopted population-coded SAN (PopSAN) and its modified variants are trained by using the twin-delayed deep deterministic policy gradient (TD3) off-policy algorithm [42] and the following hyperparameter settings: $D = 0.75$ for the LIF; $\eta = 0.01$ and $\psi = 6.0$ for the DET; $C = 3.0$ for the DTT; and $\tau_s = \tau_r = 1.0$ for the SRM. Compared to the settings of the obstacle avoidance tasks, the only different setting is the value of ψ for the DET. Following the training protocol of the PopSAN, we set the batch size to 100 and the learning rates to 0.0001 for both the actor and critic networks. The reward discount factor is set to 0.99, and the maximum length of the replay buffer is set to 1 million. We use PyTorch [53] to train all competing SNNs with an i7-7700 CPU and an NVIDIA GTX 1080Ti GPU.

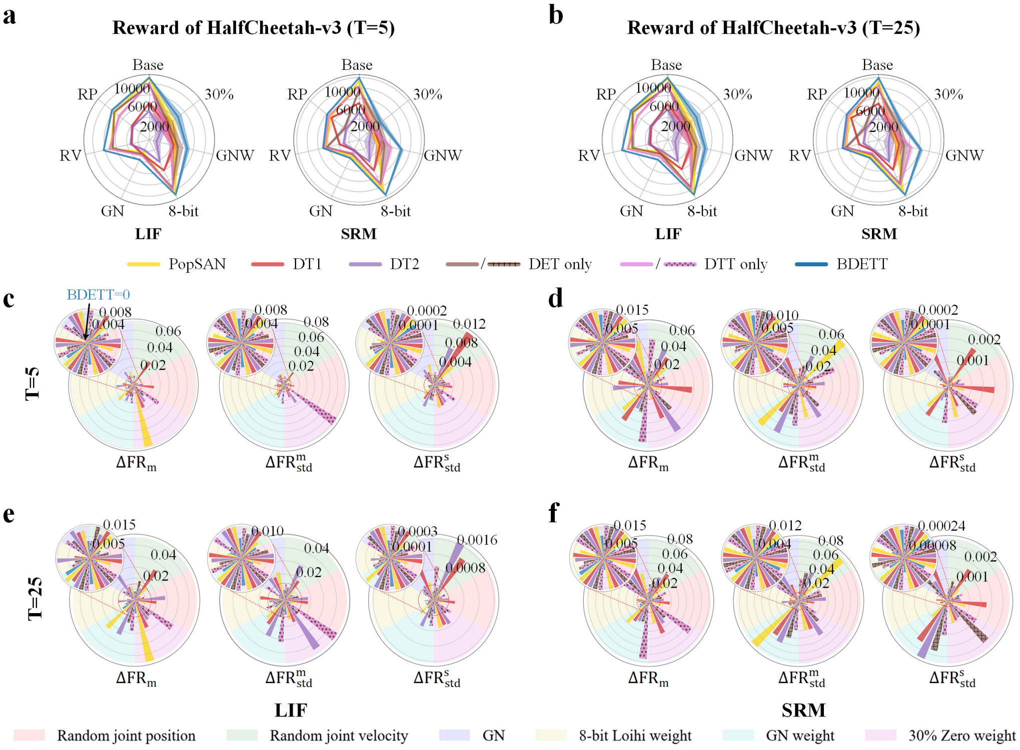

Figure 8: The experimental results obtained in the HalfCheetah-v3 tasks. a & b. The rewards obtained under normal and different degraded conditions with $T = 5$ and $T = 25$ settings, respectively. ‘Base’ indicates the base condition; ‘RP’ indicates random joint position; ‘RV’ denotes random joint velocity; ‘8-bit’, ‘GNW’, and ‘30%’ denote the 8-bit Loihi weights, GN weights, and 30% zero weights, respectively. c & d. Homeostasis measurements obtained with the $T = 5$ setting for the LIF- and SRM-based host SNNs, respectively. e & f. Homeostasis results obtained with the $T = 25$ setup for the LIF- and SRM-based host SNNs, respectively.

Assessment—Reward

After determining the evaluation settings of the PopSAN [35], we train ten models corresponding to ten random seeds, and the best-performing model is used for our assessments under different degraded conditions. In particular, the best-performing model is evaluated ten times under each experimental condition, and the mean reward of the ten evaluations represents the model’s performance. Each evaluation consists of ten episodes, and each episode lasts for a maximum of 1000 execution steps. Table 7 shows the ten evaluations’ average rewards and the corresponding standard deviations of all competing SNNs under a normal testing condition. Here, we also present the quantitative performance achieved under the $T = 5$ and $T = 25$ settings. With both the LIF and SRM models, the proposed BDETT offers the host SNNs the best rewards in all experimental settings under normal testing conditions (*i.e.*, the base conditions).

Table 7: Quantitative Performance of Mujoco HalfCheetah-v3 Tasks under standard testing condition.

	LIF ($T = 5$)	**SRM** ($T = 5$)	**LIF** ($T = 25$)	**SRM** ($T = 25$)
Name	Reward↑	Reward↑	Reward↑	Reward↑
PopSAN	10989 (σ-49)	11268 (σ-149)	11137 (σ-70)	11247 (σ-132)
DT1 [24]	6572 (σ-85)	7085 (σ-69)	6438 (σ-102)	7001 (σ-82)
DT2 [26]	5110 (σ-30)	5262 (σ-77)	5523 (σ-67)	5311 (σ-113)
DET only	9794 (σ-107)	9694 (σ-112)	9704 (σ-125)	9619 (σ-105)
DTT only	10104 (σ-44)	10332 (σ-106)	10221 (σ-57)	10563 (σ-83)
BDETT	**11064** (σ-28)	**11960** (σ-86)	**11209** (σ-56)	**11956** (σ-95)

Unlike in the obstacle avoidance tasks, even with grid searches, the rewards offered by DT1 and DT2 are significantly lower than those provided by the baseline PopSAN model. This indicates that dynamic threshold schemes may perform worse than a simple static threshold, especially for heuristic-based schemes. More importantly, we observe similar patterns in the obstacle avoidance tasks; the SRs offered by DT1 and DT2 are lower than those of the baseline SAN-NR model in most experimental conditions.

Tables 8 and 9 show the quantitative performance of all competing approaches under degraded input and weight uncertainty conditions, respectively. The results are also illustrated in Figures 8a and b for the $T = 5$ and $T = 25$ settings, respectively. We provide a more detailed analysis for each degraded condition in the following.

Degraded Inputs In the HalfCheetah-v3 continuous control task, an observation (state) s represents 17-dimensional data consisting of 8-dimensional joint position information and 9-dimensional joint velocity information. Similar to the degraded input experiments conducted in the robot obstacle avoidance tasks, we disturb a HalfCheetah-v3's observations in three different ways. a) "Random joint position": for each episode, one of the eight joint positions is randomly selected, and its original position is replaced by a random number sampled from a Gaussian distribution $\mathcal{N}(0, 0.1)$. b) "Random joint velocity": we randomly select one of the nine joint velocities in each episode and change its observed velocity to a random number sampled from a Gaussian distribution $\mathcal{N}(0, 10.0)$. c) "GN": For each episode, we add Gaussian noise (*i.e.*, $s_{input} + \mathcal{N}(0, 1.0)$, as suggested in a study regarding LSTM-LMC [43]) to each of the 17 joint states. The average rewards obtained from the ten evaluations conducted under these three different conditions are shown in Table 8.

Under all experimental settings, the proposed BDETT offers the host SNNs the highest rewards, significantly improving upon the rewards of the baseline PopSAN model by at least 438 with $T = 5$, and 358 with $T = 25$. Compared to the other two degraded input conditions, the "GN" condition disturbs all dimensions of the HalfCheetah-v3 state. Therefore, we observe that the lowest rewards obtained by all host SNNs occur with the "GN" setting. Even though the DT1 method hosted by an LIF-based SNN reduces the rewards of the baseline PopSAN model by almost half, it outperforms the baseline model with both T settings under the "GN" condition. Furthermore, the proposed BDETT provides the most stable performance, highlighted by it obtaining the smallest standard deviations under all the degraded input settings.

Weight Uncertainty We leverage the same weight uncertainty conditions as those introduced in robot obstacle avoidance experiments to demonstrate the effectiveness of all competing dynamic threshold schemes, and the corresponding results are shown in Table 9. The proposed BDETT remains the best performer under the weight uncertainty conditions. Note that the SRM-based BDETT outperforms other methods by significant margins under the "GN weight" settings, highlighting that the proposed dynamic threshold scheme can effectively deal with weight uncertainty errors. We also notice that the DT2 scheme produces the lowest rewards under all experimental weight uncertainty settings, indicating that predefining a target firing rate does not work well with weight uncertainty conditions. Surprisingly, even with low-precision 8-bit weights, the proposed BDETT helps the SRM-based host SNN achieve higher rewards than those obtained with high-precision weights under the $T = 5$ and $T = 25$ settings (11767 vs. 11268 with $T = 5$ and 11760 vs. 11247 with $T = 25$).

Assessment—Homeostatic

In the main manuscript, the changes in the quantified homeostasis values with respect to the base condition (*i.e.*, the normal Mujoco testing condition) under $T = 5$ are illustrated. The raw homeostasis measurements and the corresponding changes used for plotting the polar chart in the main manuscript are reported in Table 10. In addition, we also provide the experimental homeostasis results obtained under $T = 25$ in Table 11. The corresponding polar plots obtained under the $T = 5$ and $T = 25$ setups are shown in Figures 8c-f.

The results are consistent with those obtained in the obstacle avoidance tasks. The proposed BDETT scheme offers the strongest homeostasis, indicating the effectiveness of the proposed dynamic threshold scheme. The essential goal of homeostasis is to enhance the host SNN's performance. Therefore, we expect the SNNs with stronger homeostasis (*i.e.*, smaller ΔFR_m, ΔFR^m_{std}, and ΔFR^s_{std} values) to outperform those with weaker homeostasis. Our experimental results confirm this expectation.

Table 8: Quantitative performance of Mujoco HalfCheetah-v3 tasks under degraded input conditions.

		LIF ($T = 5$)	**SRM** ($T = 5$)	**LIF** ($T = 25$)	**SRM** ($T = 25$)
Type	**Name**	Reward↑	Reward↑	Reward↑	Reward↑
Random joint position	PopSAN	7832 (σ-222)	7120 (σ-214)	7947 (σ-253)	7167 (σ-197)
	DT1 [24]	3923 (σ-204)	6830 (σ-140)	3835 (σ-248)	6792 (σ-157)
	DT2 [26]	3750 (σ-171)	3230 (σ-239)	3950 (σ-192)	3213 (σ-230)
	DET only	7954 (σ-103)	3582 (σ-284)	8051 (σ-148)	3502 (σ-323)
	DTT only	6817 (σ-221)	7428 (σ-234)	6922 (σ-202)	7493 (σ-182)
	BDETT	**8465** (σ-121)	**7883** (σ-78)	**8463** (σ-130)	**7846** (σ-70)
Random joint velocity	PopSAN	7020 (σ-146)	6576 (σ-147)	7223 (σ-165)	6583 (σ-168)
	DT1 [24]	3187 (σ-142)	3836 (σ-181)	3203 (σ-148)	3855 (σ-202)
	DT2 [26]	3395 (σ-209)	3031 (σ-239)	3506 (σ-208)	2965 (σ-241)
	DET only	6664 (σ-179)	6392 (σ-206)	6498 (σ-219)	6435 (σ-213)
	DTT only	7249 (σ-137)	6772 (σ-299)	7363 (σ-150)	6762 (σ-247)
	BDETT	**8302** (σ-84)	**7116** (σ-146)	**8422** (σ-94)	**7127** (σ-131)
GN	PopSAN	2440 (σ-199)	3457 (σ-187)	2393 (σ-214)	3494 (σ-187)
	DT1 [24]	2790 (σ-187)	2210 (σ-124)	2773 (σ-198)	2355 (σ-120)
	DT2 [26]	1994 (σ-175)	2307 (σ-272)	2281 (σ-223)	2210 (σ-251)
	DET only	2831 (σ-157)	3013 (σ-130)	2807 (σ-163)	3061 (σ-155)
	DTT only	2974 (σ-194)	2851 (σ-81)	3281 (σ-173)	2855 (σ-115)
	BDETT	**3909** (σ-101)	**3895** (σ-81)	**3965** (σ-83)	**3852** (σ-69)

Table 9: Quantitative performance of Mujoco HalfCheetah-v3 tasks under weight uncertainty conditions.

		LIF ($T = 5$)	**SRM** ($T = 5$)	**LIF** ($T = 25$)	**SRM** ($T = 25$)
Type	**Name**	Reward↑	Reward↑	Reward↑	Reward↑
8-bit Loihi weight	PopSAN	10728 (σ-47)	10802 (σ-32)	10926 (σ-59)	10850 (σ-44)
	DT1 [24]	6026 (σ-63)	6569 (σ-46)	5883 (σ-102)	6420 (σ-97)
	DT2 [26]	4372 (σ-54)	4629 (σ-50)	4301 (σ-87)	4636 (σ-74)
	DET only	9455 (σ-125)	9398 (σ-60)	9474 (σ-137)	9376 (σ-82)
	DTT only	9803 (σ-44)	9636 (σ-84)	9968 (σ-69)	9645 (σ-113)
	BDETT	**10823** (σ-37)	**11767** (σ-45)	**10990** (σ-61)	**11760** (σ-68)
GN weight	PopSAN	4640 (σ-510)	3583 (σ-347)	4816 (σ-583)	3597 (σ-426)
	DT1 [24]	4483 (σ-491)	4128 (σ-754)	4365 (σ-466)	4051 (σ-760)
	DT2 [26]	1334 (σ-616)	2028 (σ-1026)	1402 (σ-721)	1982 (σ-993)
	DET only	5251 (σ-859)	5032 (σ-705)	5313 (σ-801)	5035 (σ-652)
	DTT only	4013 (σ-423)	6250 (σ-368)	4238 (σ-468)	6327 (σ-403)
	BDETT	**6928** (σ-373)	**8381** (σ-320)	**6957** (σ-429)	**8321** (σ-352)
30% Zero weight	PopSAN	5020 (σ-923)	3233 (σ-879)	5078 (σ-1031)	3304 (σ-950)
	DT1 [24]	3995 (σ-1319)	3503 (σ-571)	3927 (σ-1406)	3484 (σ-772)
	DT2 [26]	2721 (σ-1281)	3056 (σ-555)	2713 (σ-1352)	3002 (σ-582)
	DET only	4436 (σ-801)	4682 (σ-540)	4406 (σ-822)	4692 (σ-515)
	DTT only	3583 (σ-692)	3268 (σ-641)	3604 (σ-662)	3359 (σ-705)
	BDETT	**6551** (σ-679)	**5386** (σ-443)	**6619** (σ-712)	**5474** (σ-388)

Assessment—Ablation Studies

The ablation study results are reported in the rows named “DET only” and “DTT only” in Tables 7, 8, and 9. In addition, the results are illustrated in Figure 8. The results reflect the same facts that we observed in the obstacle avoidance tasks. The dynamic threshold schemes with only the DET or DTT components cannot effectively regulate the firing rate statuses of the host SNNs, prohibiting meaningful homeostasis. For the LIF-based host SNNs, one extreme example is illustrated in the “30% Zero weight" sections of Figures 8c and e, where ‘DTT only’ reports the largest change among all competing approaches under all experimental conditions in terms of $\Delta\text{FR}^{m}_{std}$. With the $T = 25$ setup, as shown in Figures 8d and f, the ΔFR_{m} values of ‘DTT only’ in the “GN weight" sections are the largest across all experimental settings.

Table 10: The raw homeostasis measurements and the corresponding changes with respect to the baseline condition in Mujoco HalfCheetah-v3 tasks with $T = 5$.

		LIF ($T = 5$)			**SRM** ($T = 5$)		
Type	**Name**	$\text{FR}_m(\Delta)$	$\text{FR}^m_{std}(\Delta)$	$\text{FR}^s_{std}(\Delta)$	$\text{FR}_m(\Delta)$	$\text{FR}^m_{std}(\Delta)$	$\text{FR}^s_{std}(\Delta)$
baseline condition	PopSAN	0.433	0.228	0.000978	0.427	0.241	0.002146
	DT1 [24]	0.412	0.239	0.001342	0.474	0.248	0.002166
	DT2 [26]	0.697	0.298	0.000911	0.530	0.302	0.001472
	DET only	0.284	0.220	0.001084	0.335	0.243	0.001558
	DTT only	0.646	0.257	0.002291	0.501	0.334	0.002541
	BDETT	0.249	0.190	0.001152	0.212	0.160	0.000952
Random joint position	PopSAN	0.426 (0.007)	0.237 (0.009)	0.001230 (0.000252)	0.440 (0.013)	0.258 (0.017)	0.002852 (0.000706)
	DT1 [24]	0.435 (0.023)	0.252 (0.013)	0.001713 (0.000371)	0.426 (0.048)	0.269 (0.021)	0.004145 (0.001979)
	DT2 [26]	0.686 (0.011)	0.289 (0.009)	0.001239 (0.000328)	0.508 (0.022)	0.333 (0.031)	0.002164 (0.000692)
	DET only	0.289 (0.005)	0.227 (0.007)	0.001323 (0.000239)	0.347 (0.012)	0.261 (0.018)	0.002006 (0.000448)
	DTT only	0.640 (0.006)	0.263 (0.006)	0.002077 (0.000214)	0.521 (0.020)	0.380 (0.046)	0.003124 (0.000583)
	BDETT	0.246 **(0.003)**	0.186 **(0.004)**	0.001268 **(0.000116)**	0.209 **(0.003)**	0.152 **(0.008)**	0.001071 **(0.000119)**
Random joint velocity	PopSAN	0.439 (0.006)	0.240 (0.012)	0.001502 (0.000524)	0.453 (0.026)	0.314 (0.073)	0.001513 (0.000633)
	DT1 [24]	0.381 (0.031)	0.259 (0.020)	0.002602 (0.001260)	0.510 (0.036)	0.261 (0.013)	0.004051 (0.001885)
	DT2 [26]	0.690 (0.007)	0.289 (0.009)	0.001725 (0.000814)	0.487 (0.043)	0.340 (0.038)	0.001896 (0.000424)
	DET only	0.280 **(0.004)**	0.214 (0.006)	0.000905 **(0.000179)**	0.351 (0.016)	0.219 (0.024)	0.001846 (0.000288)
	DTT only	0.656 (0.010)	0.246 (0.011)	0.002027 (0.000264)	0.550 (0.049)	0.355 (0.021)	0.002891 (0.000350)
	BDETT	0.245 **(0.004)**	0.186 **(0.004)**	0.001336 (0.000184)	0.204 **(0.008)**	0.168 **(0.008)**	0.001104 **(0.000152)**
GN	PopSAN	0.418 (0.015)	0.243 (0.015)	0.001400 (0.000422)	0.476 (0.049)	0.221 (0.020)	0.002481 (0.000335)
	DT1 [24]	0.423 (0.011)	0.225 (0.014)	0.001623 (0.000281)	0.535 (0.061)	0.283 (0.035)	0.004810 (0.002644)
	DT2 [26]	0.680 (0.017)	0.312 (0.014)	0.001273 (0.000362)	0.481 (0.049)	0.362 (0.060)	0.002215 (0.000743)
	DET only	0.278 (0.006)	0.212 (0.008)	0.001353 (0.000269)	0.302 (0.033)	0.277 (0.034)	0.002411 (0.000853)
	DTT only	0.638 (0.008)	0.269 (0.012)	0.002448 (0.000157)	0.434 (0.067)	0.301 (0.033)	0.002062 (0.000479)
	BDETT	0.245 **(0.004)**	0.184 **(0.006)**	0.001300 **(0.000148)**	0.225 **(0.013)**	0.171 **(0.011)**	0.001093 **(0.000141)**
8-bit Loihi weight	PopSAN	0.430 (0.003)	0.221 (0.007)	0.001061 (0.000083)	0.420 (0.007)	0.252 (0.011)	0.001847 (0.000299)
	DT1 [24]	0.424 (0.012)	0.248 (0.009)	0.001285 (0.000057)	0.442 (0.032)	0.230 (0.018)	0.002520 (0.000354)
	DT2 [26]	0.678 (0.019)	0.285 (0.013)	0.001022 (0.000111)	0.492 (0.038)	0.320 (0.018)	0.001701 (0.000229)
	DET only	0.290 (0.006)	0.229 (0.009)	0.001211 (0.000127)	0.327 (0.008)	0.235 (0.008)	0.001303 (0.000255)
	DTT only	0.655 (0.009)	0.264 (0.007)	0.002402 (0.000111)	0.488 (0.013)	0.343 (0.009)	0.002707 (0.000166)
	BDETT	0.249 **(0.000)**	0.186 **(0.004)**	0.001114 **(0.000038)**	0.215 **(0.003)**	0.163 **(0.003)**	0.000907 **(0.000045)**
GN weight	PopSAN	0.456 (0.023)	0.210 (0.018)	0.001249 (0.000271)	0.464 (0.037)	0.305 (0.064)	0.003358 (0.001212)
	DT1 [24]	0.426 (0.014)	0.223 (0.016)	0.001004 (0.000338)	0.429 (0.045)	0.285 (0.037)	0.003702 (0.001536)
	DT2 [26]	0.678 (0.019)	0.316 (0.018)	0.001381 (0.000470)	0.503 (0.027)	0.362 (0.060)	0.002172 (0.000700)
	DET only	0.271 (0.013)	0.227 (0.007)	0.001385 (0.000301)	0.364 (0.029)	0.208 (0.035)	0.001042 (0.000516)
	DTT only	0.626 (0.020)	0.281 (0.024)	0.001972 (0.000319)	0.562 (0.061)	0.287 (0.047)	0.004133 (0.001592)
	BDETT	0.256 **(0.007)**	0.185 **(0.005)**	0.001264 **(0.000112)**	0.219 **(0.007)**	0.165 **(0.005)**	0.001049 **(0.000097)**
30% Zero weight	PopSAN	0.502 (0.069)	0.220 (0.008)	0.001441 (0.000463)	0.412 (0.015)	0.279 (0.038)	0.003522 (0.001376)
	DT1 [24]	0.439 (0.027)	0.256 (0.017)	0.001784 (0.000442)	0.436 (0.038)	0.263 (0.015)	0.001216 (0.000950)
	DT2 [26]	0.675 (0.022)	0.325 (0.027)	0.001327 (0.000416)	0.472 (0.058)	0.258 (0.044)	0.000994 (0.000478)
	DET only	0.268 (0.016)	0.210 (0.010)	0.001329 (0.000245)	0.370 (0.035)	0.210 (0.033)	0.003205 (0.001647)
	DTT only	0.680 (0.034)	0.338 (0.081)	0.002804 (0.000513)	0.485 (0.016)	0.317 (0.017)	0.001024 (0.001517)
	BDETT	0.243 **(0.006)**	0.184 **(0.006)**	0.001306 **(0.000154)**	0.217 **(0.005)**	0.154 **(0.006)**	0.001058 **(0.000106)**

Table 11: The raw homeostasis measurements and the corresponding changes with respect to the baseline condition in Mujoco HalfCheetah-v3 tasks with the $T = 25$ setup.

		LIF ($T = 25$)			**SRM** ($T = 25$)		
Type	**Name**	$\mathrm{FR}_m(\Delta)$	$\mathrm{FR}^m_{std}(\Delta)$	$\mathrm{FR}^s_{std}(\Delta)$	$\mathrm{FR}_m(\Delta)$	$\mathrm{FR}^m_{std}(\Delta)$	$\mathrm{FR}^s_{std}(\Delta)$
baseline condition	PopSAN	0.436	0.230	0.001156	0.440	0.252	0.002217
	DT1 [24]	0.420	0.240	0.001158	0.470	0.247	0.001869
	DT2 [26]	0.676	0.290	0.000939	0.521	0.305	0.001785
	DET only	0.292	0.225	0.001237	0.341	0.253	0.001631
	DTT only	0.635	0.257	0.002433	0.493	0.340	0.002722
	BDETT	0.251	0.192	0.001292	0.215	0.173	0.001074
Random joint position	PopSAN	0.426 (0.010)	0.238 (0.008)	0.001381 (0.000225)	0.458 (0.018)	0.264 (0.012)	0.003048 (0.000831)
	DT1 [24]	0.440 (0.020)	0.258 (0.018)	0.001721 (0.000563)	0.442 (0.028)	0.268 (0.021)	0.003632 (0.001763)
	DT2 [26]	0.648 (0.028)	0.272 (0.018)	0.001348 (0.000409)	0.501 (0.020)	0.342 (0.037)	0.002384 (0.000599)
	DET only	0.301 (0.009)	0.236 (0.011)	0.001464 (0.000227)	0.359 (0.018)	0.285 (0.032)	0.002179 (0.000548)
	DTT only	0.630 (0.005)	0.265 (0.008)	0.002104 (0.000329)	0.518 (0.025)	0.372 (0.032)	0.003438 (0.000716)
	BDETT	0.247 **(0.004)**	0.187 **(0.005)**	0.001431 **(0.000139)**	0.210 **(0.005)**	0.163 **(0.010)**	0.001242 **(0.000168)**
Random joint velocity	PopSAN	0.450 (0.014)	0.239 (0.009)	0.001425 (0.000269)	0.474 (0.034)	0.332 (0.080)	0.002922 (0.000705)
	DT1 [24]	0.387 (0.033)	0.258 (0.018)	0.002474 (0.001316)	0.427 (0.043)	0.268 (0.021)	0.003572 (0.001703)
	DT2 [26]	0.664 (0.012)	0.261 (0.029)	0.002582 (0.001643)	0.487 (0.034)	0.347 (0.042)	0.002529 (0.000744)
	DET only	0.310 (0.018)	0.235 (0.010)	0.001582 (0.000345)	0.368 (0.027)	0.292 (0.039)	0.002544 (0.000913)
	DTT only	0.627 (0.008)	0.269 (0.012)	0.001548 (0.000885)	0.515 (0.022)	0.370 (0.030)	0.003282 (0.000560)
	BDETT	0.245 **(0.006)**	0.185 **(0.007)**	0.001478 **(0.000186)**	0.208 **(0.007)**	0.183 **(0.010)**	0.001305 **(0.000231)**
GN	PopSAN	0.423 (0.013)	0.252 (0.022)	0.001633 (0.000477)	0.403 (0.037)	0.236 (0.016)	0.002574 (0.000357)
	DT1 [24]	0.433 (0.013)	0.221 (0.019)	0.002061 (0.000903)	0.552 (0.082)	0.271 (0.024)	0.003784 (0.001915)
	DT2 [26]	0.642 (0.034)	0.310 (0.020)	0.001385 (0.000446)	0.480 (0.041)	0.373 (0.068)	0.002833 (0.001048)
	DET only	0.281 (0.011)	0.219 **(0.006)**	0.001610 (0.000373)	0.304 (0.037)	0.290 (0.037)	0.003082 (0.001451)
	DTT only	0.622 (0.013)	0.274 (0.017)	0.002762 (0.000329)	0.453 (0.040)	0.305 (0.035)	0.002048 (0.000674)
	BDETT	0.243 **(0.008)**	0.184 (0.008)	0.001512 **(0.000220)**	0.226 **(0.011)**	0.180 **(0.007)**	0.001283 **(0.000209)**
8-bit Loihi weight	PopSAN	0.432 (0.004)	0.224 (0.006)	0.001310 (0.000154)	0.452 (0.012)	0.238 (0.014)	0.001849 (0.000368)
	DT1 [24]	0.427 (0.007)	0.221 (0.019)	0.001035 (0.000123)	0.485 (0.015)	0.259 (0.012)	0.002363 (0.000494)
	DT2 [26]	0.661 (0.015)	0.278 (0.012)	0.001174 (0.000235)	0.535 (0.014)	0.328 (0.023)	0.001976 (0.000191)
	DET only	0.299 (0.007)	0.234 (0.009)	0.001379 (0.000142)	0.351 (0.010)	0.267 (0.014)	0.001275 (0.000356)
	DTT only	0.644 (0.009)	0.242 (0.015)	0.002210 (0.000223)	0.506 (0.013)	0.354 (0.014)	0.002894 (0.000172)
	BDETT	0.250 **(0.001)**	0.197 **(0.005)**	0.001393 **(0.000101)**	0.211 **(0.004)**	0.178 **(0.005)**	0.001169 **(0.000094)**
GN weight	PopSAN	0.471 (0.035)	0.211 (0.019)	0.001637 (0.000481)	0.479 (0.039)	0.336 (0.084)	0.003181 (0.000964)
	DT1 [24]	0.433 **(0.013)**	0.218 (0.022)	0.000904 (0.000254)	0.438 (0.032)	0.302 (0.055)	0.004041 (0.002172)
	DT2 [26]	0.642 (0.034)	0.320 (0.030)	0.001683 (0.000744)	0.492 (0.029)	0.366 (0.061)	0.004585 (0.002800)
	DET only	0.271 (0.021)	0.238 (0.013)	0.001720 (0.000483)	0.369 (0.028)	0.202 (0.051)	0.003927 (0.002296)
	DTT only	0.607 (0.028)	0.289 (0.032)	0.003104 (0.000671)	0.569 (0.076)	0.301 (0.039)	0.004273 (0.001551)
	BDETT	0.264 **(0.013)**	0.183 **(0.009)**	0.001512 **(0.000220)**	0.229 **(0.014)**	0.182 **(0.009)**	0.001199 **(0.000125)**
30% Zero weight	PopSAN	0.493 (0.057)	0.215 (0.015)	0.001892 (0.000736)	0.410 (0.030)	0.292 (0.040)	0.003237 (0.001020)
	DT1 [24]	0.447 (0.027)	0.258 (0.018)	0.001859 (0.000701)	0.424 (0.046)	0.288 (0.041)	0.000926 (0.000943)
	DT2 [26]	0.655 (0.021)	0.334 (0.044)	0.001674 (0.000735)	0.486 (0.035)	0.251 (0.054)	0.000896 (0.000889)
	DET only	0.273 (0.019)	0.239 (0.014)	0.001692 (0.000455)	0.371 (0.030)	0.216 (0.037)	0.004106 (0.002475)
	DTT only	0.677 (0.042)	0.305 (0.048)	0.002976 (0.000543)	0.426 (0.067)	0.312 (0.028)	0.001176 (0.001546)
	BDETT	0.239 **(0.012)**	0.181 **(0.011)**	0.001642 **(0.000350)**	0.221 **(0.006)**	0.180 **(0.007)**	0.001183 **(0.000109)**

Supplementary Note 6: Additional Details on Continuous Control Experiments—Ant-v3

The training and experimental setups are the same as those used for the HalfCheetah-v3 tasks.

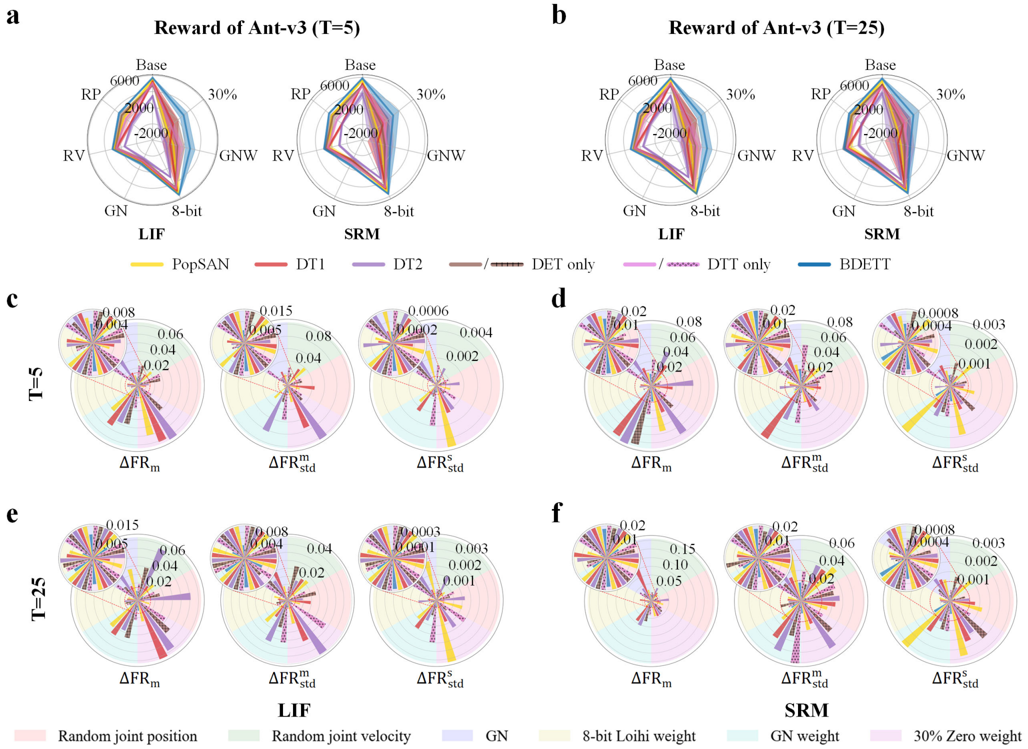

Figure 9: The experimental results obtained in the Ant-v3 tasks. a & b. The rewards obtained under normal and different degraded conditions under the $T = 5$ and $T = 25$ settings, respectively. 'Base' denotes the base condition; 'RP' and 'RV' represent random joint position and random joint velocity, respectively; '8-bit', 'GNW', and '30%' denote the 8-bit Loihi weights, GN weights, and 30% zero weights, respectively. c & d. Homeostasis measurements obtained under the $T = 5$ setting for the LIF- and SRM-based host SNNs, respectively. e & f. Homeostasis results obtained with the $T = 25$ setup for the LIF- and SRM-based host SNNs, respectively.

Assessment—Reward

As in the HalfCheetah-v3 tasks, we present the rewards of all competing host SNNs under the original normal conditions with the $T = 5$ and $T = 25$ settings; see Table 12. Compared to HalfCheetah-v3's 17-dimensional state, the state of an Ant-v3 task has 111 dimensions. Thus, the rewards obtained from the Ant-v3 experiments are much lower than those obtained in the HalfCheetah-v3 tasks. Nevertheless, the proposed BDETT offers the highest rewards in the Ant-v3 tasks, and it improves upon the rewards of the LIF- and SRM-based baseline models by at least 173 and 236, respectively.

Relative to the HalfCheetah-v3 tasks, DT1 offers much better rewards in the Ant-v3 tasks under normal conditions. However, the rewards provided by DT1 and DT2 are still lower than those of the baseline PopSAN model. This observation is consistent with those obtained in the obstacle avoidance and HalfCheetah-v3 experiments.

We show the quantitative performance of all competing methods under degraded input and weight uncertainty conditions in Tables 13 and 14, respectively. In Figures 9a and b, we also intuitively present the results. The proposed BDETT is still the best performer under all experimental conditions based on the obtained results. A more detailed analysis for each degraded condition is provided in the following.

Degraded Inputs Compared to that of a HalfCheetah-v3 agent, the observation (state) of an Ant-v3 agent s represents a 111-dimensional data consisting of 13-dimensional joint position information, 14-dimensional joint velocity information, and 84-dimensional contact force data. We disturb Ant-v3's observation in the same three ways introduced in the HalfCheetah-v3 tasks: "Random joint position", "Random joint velocity", and "GN".

Table 12: Quantitative performance of Mujoco Ant-v3 tasks under standard testing condition.

	LIF ($T=5$)	**SRM** ($T=5$)	**LIF** ($T=25$)	**SRM** ($T=25$)
Name	Reward↑	Reward↑	Reward↑	Reward↑
PopSAN	5526 (σ-81)	5643 (σ-84)	5711 (σ-105)	5612 (σ-105)
DT1 [24]	5272 (σ-142)	5179 (σ-157)	5218 (σ-164)	5121 (σ-117)
DT2 [26]	3454 (σ-183)	3925 (σ-483)	3628 (σ-180)	4016 (σ-445)
DET only	4836 (σ-82)	4971 (σ-128)	4957 (σ-113)	5125 (σ-144)
DTT only	5041 (σ-294)	4883 (σ-154)	5192 (σ-267)	4864 (σ-187)
BDETT	**5726** (σ-61)	**5879** (σ-117)	**5884** (σ-97)	**5942** (σ-136)

Table 13: Quantitative performance of Mujoco Ant-v3 tasks under degraded input conditions.

		LIF ($T=5$)	**SRM** ($T=5$)	**LIF** ($T=25$)	**SRM** ($T=25$)
Type	**Name**	Reward↑	Reward↑	Reward↑	Reward↑
Random joint position	PopSAN	2503 (σ-503)	3004 (σ-131)	2544 (σ-337)	3036 (σ-152)
	DT1 [24]	1435 (σ-130)	1333 (σ-122)	1380 (σ-158)	1258 (σ-150)
	DT2 [26]	1280 (σ-234)	1330 (σ-99)	1335 (σ-206)	1364 (σ-152)
	DET only	2907 (σ-320)	2836 (σ-392)	2862 (σ-342)	2994 (σ-332)
	DTT only	2213 (σ-389)	2190 (σ-119)	2305 (σ-373)	2273 (σ-162)
	BDETT	**3339** (σ-111)	**3450** (σ-75)	**3320** (σ-126)	**3427** (σ-115)
Random joint velocity	PopSAN	2890 (σ-115)	2372 (σ-390)	2858 (σ-149)	2287 (σ-427)
	DT1 [24]	2628 (σ-232)	2508 (σ-166)	2643 (σ-259)	2574 (σ-232)
	DT2 [26]	1579 (σ-89)	1025 (σ-139)	1595 (σ-131)	1009 (σ-208)
	DET only	2720 (σ-365)	2809 (σ-296)	2802 (σ-197)	2896 (σ-372)
	DTT only	2635 (σ-234)	2515 (σ-201)	2699 (σ-255)	2618 (σ-260)
	BDETT	**3103** (σ-95)	**2984** (σ-176)	**3217** (σ-119)	**2996** (σ-195)
GN	PopSAN	977 (σ-320)	1031 (σ-212)	1022 (σ-358)	1059 (σ-217)
	DT1 [24]	922 (σ-234)	958 (σ-156)	875 (σ-270)	1012 (σ-178)
	DT2 [26]	560 (σ-179)	583 (σ-158)	664 (σ-163)	623 (σ-235)
	DET only	782 (σ-246)	1048 (σ-345)	844 (σ-304)	1105 (σ-364)
	DTT only	849 (σ-177)	1172 (σ-209)	906 (σ-170)	1255 (σ-218)
	BDETT	**1269** (σ-166)	**1559** (σ-138)	**1339** (σ-156)	**1576** (σ-161)

The average rewards obtained in the ten evaluations conducted under these three different conditions are shown in Table 13. Under all experimental settings, the proposed BDETT offers the host SNNs the highest rewards, significantly improving upon the reward of the baseline PopSAN model by at least 213.

Weight Uncertainty We leverage the same weight uncertainty conditions as those used in the robot obstacle avoidance and HalfCheetah-v3 experiments. The experimental results are shown in Table 14. The proposed BDETT remains the best performer under all weight uncertainty conditions. As in the HalfCheetah-v3 experiments, even with low-precision 8-bit weights, the proposed BDETT helps both the LIF- and SRM-based host SNNs achieve higher rewards than those offered by the baseline counterparts with high-precision floating-point weights under $T = 5$ (*i.e.*, 5570 vs. 5526 and 5648 vs. 5643, respectively). With the $T = 25$ setup, the SRM-based host SNN exhibits the same pattern.

Assessment—Homeostatic

The raw homeostasis measurements obtained with both T=5 and T=25 are provided in Tables 15 and 16, respectively. The corresponding homeostasis plots are shown in Figures 9c-f.

The homeostasis results obtained in the Ant-v3 tasks demonstrate the effectiveness of the proposed BDETT in terms of regulating the neuronal firing rates of the host SNNs, inducing minimal changes in all three metrics when transferring from the base conditions to all other experimental settings. We witness that the strongest homeostasis again provides the highest rewards.

Assessment—Ablation Studies

The experimental results obtained in the Ant-v3 task ablation studies under different conditions are reported in the rows named 'DET only' and 'DTT only' in Tables 12, 13, and 14. All listed evaluations validate that the BDETT scheme performs better than any single component. As shown in Table 14, under the "GN weight" condition, both 'DET only' and 'DTT only' offer both the LIF- and SRM-based host SNNs higher rewards than the other competing dynamic threshold schemes. As in the other tasks, the dynamic threshold scheme with only one component cannot effectively regulate the firing rate statuses of the host SNNs, prohibiting meaningful

Table 14: Quantitative performance of Mujoco Ant-v3 tasks with weight uncertainty conditions.

		LIF ($T = 5$)	**SRM** ($T = 5$)	**LIF** ($T = 25$)	**SRM** ($T = 25$)
Type	**Name**	Reward↑	Reward↑	Reward↑	Reward↑
8-bit Loihi weight	PopSAN	5347 (σ-175)	5285 (σ-158)	5504 (σ-210)	5228 (σ-139)
	DT1 [24]	5004 (σ-88)	4889 (σ-163)	4826 (σ-102)	4902 (σ-182)
	DT2 [26]	3122 (σ-77)	3463 (σ-99)	3266 (σ-93)	3676 (σ-86)
	DET only	4561 (σ-135)	4634 (σ-111)	4663 (σ-156)	4727 (σ-153)
	DTT only	4703 (σ-56)	4722 (σ-87)	4903 (σ-63)	4779 (σ-126)
	BDETT	**5570** (σ-59)	**5648** (σ-73)	**5606** (σ-52)	**5620** (σ-143)
GN weight	PopSAN	637 (σ-860)	467 (σ-951)	667 (σ-1002)	444 (σ-1105)
	DT1 [24]	221 (σ-949)	-57 (σ-1245)	155 (σ-839)	6 (σ-1722)
	DT2 [26]	-265 (σ-488)	-173 (σ-640)	-226 (σ-628)	-198 (σ-883)
	DET only	1208 (σ-855)	940 (σ-750)	1258 (σ-638)	923 (σ-883)
	DTT only	1392 (σ-467)	1204 (σ-746)	1448 (σ-644)	1310 (σ-867)
	BDETT	**2782** (σ-599)	**1658** (σ-640)	**2780** (σ-621)	**1669** (σ-612)
30% Zero weight	PopSAN	287 (σ-524)	372 (σ-994)	273 (σ-633)	407 (σ-959)
	DT1 [24]	1247 (σ-801)	1450 (σ-863)	1200 (σ-1020)	1552 (σ-996)
	DT2 [26]	-548 (σ-354)	-203 (σ-901)	-563 (σ-743)	-183 (σ-1125)
	DET only	1007 (σ-960)	1136 (σ-1179)	1084 (σ-1092)	1186 (σ-1084)
	DTT only	908 (σ-428)	1559 (σ-1167)	1038 (σ-487)	1563 (σ-1049)
	BDETT	**2931** (σ-544)	**3046** (σ-886)	**2978** (σ-605)	**3152** (σ-924)

Table 15: The raw homeostasis measurements and the corresponding changes with respect to the baseline condition in Mujoco Ant-v3 tasks with the $T = 5$ setting.

		LIF ($T = 5$)			**SRM** ($T = 5$)		
Type	**Name**	$\mathrm{FR}_m(\Delta)$	$\mathrm{FR}^m_{std}(\Delta)$	$\mathrm{FR}^s_{std}(\Delta)$	$\mathrm{FR}_m(\Delta)$	$\mathrm{FR}^m_{std}(\Delta)$	$\mathrm{FR}^s_{std}(\Delta)$
baseline condition	PopSAN	0.548	0.262	0.002169	0.197	0.150	0.001823
	DT1 [24]	0.531	0.253	0.002483	0.446	0.251	0.002172
	DT2 [26]	0.770	0.223	0.004427	0.580	0.167	0.003238
	DET only	0.289	0.246	0.001820	0.328	0.187	0.002027
	DTT only	0.547	0.280	0.001554	0.443	0.276	0.002942
	BDETT	0.271	0.204	0.001325	0.213	0.199	0.001535
Random joint position	PopSAN	0.556 (0.008)	0.242 (0.020)	0.003210 (0.001041)	0.208 (0.011)	0.134 (0.016)	0.003022 (0.001199)
	DT1 [24]	0.521 (0.010)	0.209 (0.044)	0.002859 (0.000376)	0.420 (0.026)	0.275 (0.024)	0.002819 (0.000647)
	DT2 [26]	0.757 (0.013)	0.211 (0.012)	0.002735 (0.001692)	0.529 (0.051)	0.188 (0.021)	0.002521 (0.000717)
	DET only	0.262 (0.027)	0.270 (0.024)	0.002575 (0.000755)	0.339 (0.011)	0.210 (0.023)	0.002831 (0.000804)
	DTT only	0.519 (0.028)	0.256 (0.024)	0.002481 (0.000927)	0.430 (0.013)	0.244 (0.032)	0.002454 (0.000488)
	BDETT	0.275 **(0.004)**	0.209 **(0.005)**	0.001240 **(0.000085)**	0.207 **(0.006)**	0.192 **(0.007)**	0.001308 **(0.000227)**
Random joint velocity	PopSAN	0.526 (0.022)	0.223 (0.039)	0.001743 (0.000426)	0.180 (0.017)	0.117 (0.033)	0.003239 (0.001416)
	DT1 [24]	0.516 (0.015)	0.221 (0.032)	0.003031 (0.000548)	0.415 (0.031)	0.278 (0.027)	0.002766 (0.000594)
	DT2 [26]	0.760 (0.010)	0.220 (0.003)	0.003049 (0.001378)	0.534 (0.046)	0.182 (0.015)	0.002749 (0.000489)
	DET only	0.265 (0.024)	0.266 (0.020)	0.001443 (0.000377)	0.345 (0.017)	0.206 (0.019)	0.003182 (0.001155)
	DTT only	0.524 (0.023)	0.251 (0.029)	0.003020 (0.001466)	0.410 (0.033)	0.225 (0.051)	0.002385 (0.000557)
	BDETT	0.265 **(0.006)**	0.206 **(0.002)**	0.001452 **(0.000127)**	0.207 **(0.006)**	0.193 **(0.006)**	0.001882 **(0.000347)**
GN	PopSAN	0.560 (0.012)	0.244 (0.018)	0.004782 (0.002613)	0.180 (0.017)	0.123 (0.027)	0.002589 (0.000766)
	DT1 [24]	0.515 (0.016)	0.229 (0.024)	0.002723 (0.000240)	0.470 (0.024)	0.289 (0.038)	0.003259 (0.001087)
	DT2 [26]	0.727 (0.043)	0.238 (0.015)	0.002130 (0.002297)	0.527 (0.053)	0.204 (0.037)	0.004192 (0.000954)
	DET only	0.302 (0.013)	0.260 (0.014)	0.003762 (0.001942)	0.312 (0.016)	0.161 (0.026)	0.002495 **(0.000468)**
	DTT only	0.506 (0.041)	0.241 (0.039)	0.003306 (0.001752)	0.390 (0.053)	0.320 (0.044)	0.004334 (0.001392)
	BDETT	0.262 **(0.009)**	0.198 **(0.006)**	0.001539 **(0.000214)**	0.198 **(0.015)**	0.178 **(0.021)**	0.002068 (0.000533)
8-bit Loihi weight	PopSAN	0.540 (0.008)	0.269 (0.007)	0.001838 (0.000331)	0.206 (0.009)	0.167 (0.017)	0.001602 (0.000221)
	DT1 [24]	0.519 (0.012)	0.261 (0.008)	0.002217 (0.000266)	0.459 (0.013)	0.237 (0.014)	0.002406 (0.000234)
	DT2 [26]	0.758 (0.012)	0.217 (0.006)	0.003884 (0.000543)	0.566 (0.014)	0.193 (0.026)	0.002513 (0.000725)
	DET only	0.281 (0.008)	0.250 (0.004)	0.001933 (0.000113)	0.323 (0.005)	0.194 (0.007)	0.001869 (0.000158)
	DTT only	0.539 (0.008)	0.286 (0.006)	0.001463 (0.000091)	0.433 (0.010)	0.285 (0.009)	0.003206 (0.000264)
	BDETT	0.274 **(0.003)**	0.206 **(0.002)**	0.001280 **(0.000045)**	0.215 **(0.002)**	0.203 **(0.004)**	0.001602 **(0.000067)**
GN weight	PopSAN	0.507 (0.041)	0.291 (0.029)	0.003844 (0.001675)	0.190 (0.007)	0.138 (0.012)	0.004859 (0.003036)
	DT1 [24]	0.475 (0.056)	0.261 (0.008)	0.004749 (0.002266)	0.519 (0.073)	0.328 (0.077)	0.003441 (0.001269)
	DT2 [26]	0.722 (0.048)	0.305 (0.082)	0.003632 (0.000795)	0.503 (0.077)	0.212 (0.045)	0.004833 (0.001595)
	DET only	0.242 (0.047)	0.218 (0.028)	0.000966 (0.000854)	0.402 (0.074)	0.209 (0.022)	0.003346 (0.001319)
	DTT only	0.520 (0.027)	0.223 (0.057)	0.004540 (0.002986)	0.408 (0.035)	0.322 (0.046)	0.004416 (0.001474)
	BDETT	0.268 **(0.003)**	0.208 **(0.004)**	0.001548 **(0.000223)**	0.208 **(0.005)**	0.190 **(0.009)**	0.002351 **(0.000816)**
30% Zero weight	PopSAN	0.488 (0.060)	0.290 (0.028)	0.006801 (0.004632)	0.140 (0.057)	0.130 (0.020)	0.004246 (0.002423)
	DT1 [24]	0.460 (0.071)	0.328 (0.075)	0.004668 (0.002185)	0.411 (0.035)	0.223 (0.028)	0.003402 (0.001230)
	DT2 [26]	0.696 (0.074)	0.326 (0.103)	0.002308 (0.002119)	0.511 (0.069)	0.203 (0.036)	0.004283 (0.001045)
	DET only	0.250 (0.039)	0.279 (0.033)	0.001023 (0.000797)	0.364 (0.036)	0.213 (0.026)	0.003631 (0.001604)
	DTT only	0.566 (0.019)	0.308 (0.028)	0.003243 (0.001689)	0.465 (0.022)	0.268 **(0.008)**	0.003563 **(0.000621)**
	BDETT	0.258 **(0.013)**	0.217 **(0.013)**	0.001840 **(0.000515)**	0.194 **(0.019)**	0.191 **(0.008)**	0.002276 (0.000741)

Table 16: The raw homeostasis measurements and the corresponding changes with respect to the baseline condition in Mujoco Ant-v3 tasks with the $T = 25$ setup.

		LIF ($T = 25$)			**SRM** ($T = 25$)		
Type	**Name**	$\mathrm{FR}_m(\Delta)$	$\mathrm{FR}^m_{std}(\Delta)$	$\mathrm{FR}^s_{std}(\Delta)$	$\mathrm{FR}_m(\Delta)$	$\mathrm{FR}^m_{std}(\Delta)$	$\mathrm{FR}^s_{std}(\Delta)$
baseline condition	PopSAN	0.535	0.258	0.002247	0.213	0.166	0.002027
	DT1 [24]	0.530	0.252	0.002688	0.453	0.244	0.001694
	DT2 [26]	0.753	0.230	0.004728	0.563	0.182	0.003493
	DET only	0.302	0.250	0.001947	0.334	0.192	0.001872
	DTT only	0.541	0.276	0.001526	0.451	0.281	0.002485
	BDETT	0.275	0.204	0.001503	0.222	0.195	0.001829
Random joint position	PopSAN	0.549 (0.014)	0.242 (0.016)	0.003794 (0.001547)	0.248 (0.035)	0.140 (0.026)	0.003682 (0.001655)
	DT1 [24]	0.522 (0.008)	0.223 (0.029)	0.003074 (0.000386)	0.426 (0.027)	0.277 (0.033)	0.002951 (0.001257)
	DT2 [26]	0.692 (0.061)	0.211 (0.019)	0.003076 (0.001652)	0.513 (0.050)	0.219 (0.037)	0.002523 (0.000970)
	DET only	0.276 (0.026)	0.286 (0.036)	0.002744 (0.000797)	0.352 (0.018)	0.224 (0.032)	0.002642 (0.000770)
	DTT only	0.520 (0.021)	0.252 (0.024)	0.002636 (0.001110)	0.430 (0.021)	0.246 (0.035)	0.002206 (0.000279)
	BDETT	0.280 **(0.005)**	0.210 **(0.006)**	0.001386 **(0.000117)**	0.216 **(0.006)**	0.188 **(0.007)**	0.001682 **(0.000147)**
Random joint velocity	PopSAN	0.510 (0.025)	0.224 (0.034)	0.001589 (0.000658)	0.189 (0.024)	0.121 (0.045)	0.003472 (0.001445)
	DT1 [24]	0.510 (0.020)	0.220 (0.032)	0.003236 (0.000548)	0.424 (0.029)	0.281 (0.037)	0.002692 (0.000998)
	DT2 [26]	0.688 (0.065)	0.203 (0.027)	0.003163 (0.001565)	0.522 (0.041)	0.220 (0.038)	0.002732 (0.000761)
	DET only	0.263 (0.039)	0.294 (0.044)	0.002583 (0.000636)	0.353 (0.019)	0.209 (0.017)	0.003135 (0.001263)
	DTT only	0.522 (0.019)	0.259 (0.017)	0.002184 (0.000658)	0.426 (0.025)	0.252 (0.029)	0.001921 (0.000564)
	BDETT	0.267 **(0.008)**	0.206 **(0.002)**	0.001632 **(0.000129)**	0.213 **(0.009)**	0.190 **(0.005)**	0.002005 **(0.000176)**
GN	PopSAN	0.572 (0.037)	0.240 (0.018)	0.004581 (0.002334)	0.180 (0.033)	0.120 (0.046)	0.003148 (0.001121)
	DT1 [24]	0.512 (0.018)	0.214 (0.038)	0.002012 (0.000676)	0.633 **(0.018)**	0.269 (0.025)	0.002885 (0.001191)
	DT2 [26]	0.718 (0.035)	0.259 (0.029)	0.002833 (0.001895)	0.520 (0.043)	0.232 (0.050)	0.002148 (0.001345)
	DET only	0.323 (0.021)	0.299 (0.049)	0.003665 (0.001718)	0.305 (0.029)	0.169 (0.023)	0.002684 (0.000812)
	DTT only	0.512 (0.029)	0.244 (0.032)	0.003522 (0.001996)	0.412 (0.039)	0.328 (0.047)	0.003144 (0.000659)
	BDETT	0.264 **(0.011)**	0.200 **(0.004)**	0.001400 **(0.000103)**	0.204 **(0.018)**	0.177 **(0.018)**	0.002129 **(0.000300)**
8-bit Loihi weight	PopSAN	0.546 (0.011)	0.275 (0.017)	0.002569 (0.000322)	0.225 (0.012)	0.179 (0.013)	0.002581 (0.000554)
	DT1 [24]	0.542 (0.012)	0.264 (0.012)	0.003189 (0.000501)	0.421 (0.032)	0.257 (0.013)	0.001184 (0.000510)
	DT2 [26]	0.738 (0.015)	0.220 (0.010)	0.003680 (0.001048)	0.545 (0.018)	0.194 (0.012)	0.003026 (0.000467)
	DET only	0.316 (0.014)	0.236 (0.014)	0.002292 (0.000345)	0.346 (0.012)	0.212 (0.020)	0.002214 (0.000342)
	DTT only	0.554 (0.013)	0.290 (0.014)	0.001174 (0.000352)	0.438 (0.013)	0.294 (0.013)	0.002763 (0.000278)
	BDETT	0.270 **(0.005)**	0.200 **(0.004)**	0.001576 **(0.000073)**	0.218 **(0.004)**	0.199 **(0.004)**	0.001722 **(0.000107)**
GN weight	PopSAN	0.493 (0.042)	0.295 (0.037)	0.003726 (0.001479)	0.190 (0.023)	0.141 (0.025)	0.005216 (0.003189)
	DT1 [24]	0.482 (0.048)	0.277 (0.025)	0.004663 (0.001975)	0.522 (0.069)	0.287 (0.043)	0.003540 (0.001846)
	DT2 [26]	0.715 (0.038)	0.286 (0.056)	0.003431 (0.001297)	0.486 (0.077)	0.235 (0.053)	0.004632 (0.001139)
	DET only	0.257 (0.045)	0.215 (0.035)	0.001036 (0.000911)	0.394 (0.060)	0.227 (0.035)	0.003373 (0.001501)
	DTT only	0.518 (0.023)	0.223 (0.053)	0.003998 (0.002472)	0.410 (0.041)	0.342 (0.061)	0.004025 (0.001540)
	BDETT	0.265 **(0.010)**	0.211 **(0.007)**	0.001729 **(0.000226)**	0.212 **(0.010)**	0.206 **(0.011)**	0.002773 **(0.000944)**
30% Zero weight	PopSAN	0.510 (0.025)	0.288 (0.030)	0.005942 (0.003695)	0.168 (0.045)	0.137 (0.029)	0.004893 (0.002866)
	DT1 [24]	0.457 (0.073)	0.307 (0.055)	0.004043 (0.001355)	0.406 (0.047)	0.220 (0.024)	0.003909 (0.002215)
	DT2 [26]	0.685 (0.068)	0.309 (0.079)	0.002863 (0.001865)	0.517 (0.046)	0.236 (0.054)	0.005426 (0.001933)
	DET only	0.252 (0.050)	0.243 **(0.007)**	0.001132 (0.000815)	0.376 (0.042)	0.232 (0.040)	0.004387 (0.002515)
	DTT only	0.579 (0.038)	0.332 (0.056)	0.003538 (0.002012)	0.487 (0.036)	0.256 (0.025)	0.003692 (0.001207)
	BDETT	0.263 **(0.012)**	0.213 (0.009)	0.001883 **(0.000380)**	0.204 **(0.018)**	0.183 **(0.012)**	0.002485 **(0.000656)**

homeostasis. When combining the DTT and DET components, we witness much more stable homeostasis for all host SNNs.

Table 17: Quantitative performance of obstacle avoidance tasks with different constant coefficient settings under static obstacle condition.

	LIF ($T = 5$)	**SRM** ($T = 5$)
Constant	SR↑	SR↑
0.1	97.5%	**96.5%**
0.2(original)	**98.5%**	**96.5%**
0.3	**98.5%**	**96.5%**
0.4	97.5%	95.5%
0.5	98%	95%
1.0	93.5%	91.5%

Table 18: Quantitative performance of obstacle avoidance tasks with different constant coefficient settings under dynamic obstacle condition.

	LIF ($T = 5$)	**SRM** ($T = 5$)
Constant	SR↑	SR↑
0.1	91.5%	**90.5%**
0.2(original)	**92.5%**	**90.5%**
0.3	92%	90%
0.4	92%	89%
0.5	91%	89%
1.0	86%	80.5%

Supplementary Note 7: Impact of Constant Coefficient ‘0.2’ in the DET

In the proposed DET component, a constant ‘0.2’ is used for balancing the contributions of the specifically designed bias items, *i.e.*, $0.2(\max(v_i^l(t)) - \min(v_i^l(t)))$ and $0.2(\max(\Theta_i^l(t)) - \min(\Theta_i^l(t)))$.

Based on our experimental results, the effectiveness of the proposed BDETT is not sensitive to this value. In the obstacle avoidance tasks (see Tables 18, 19, and 20), when the constant value is within the range of $[0.1, 0.5]$, the standard deviations of the SRs for the LIF- and SRM-based host SNNs are 0.005 and 0.007, respectively. More importantly, even with an extreme value of 1.0, the corresponding SRs are higher than those offered by all other competing dynamic threshold approaches.

In the HalfCheetah-v3 tasks (see Tables 21, 22, and 23) and the Ant-v3 tasks (see Table 24, 25, and 26), the proposed BDETT method is more sensitive to the coefficient value than in the obstacle avoidance tasks. When the constant value is within the range of $[0.1, 0.5]$, the standard deviations of the rewards are 94 and 107 for the LIF- and SRM-based host SNNs in the HalfCheetah-v3 tasks, respectively. In the Ant-v3 experiments, the LIF- and SRM-based SNNs provide rewards of 114 and 111, respectively. With the extreme value of 1.0, under some experimental conditions, the rewards offered by our approach are still higher than those provided by other methods (e.g., the “30% zero weight" condition of HalfCheetah-v3 and the “GN weight" condition of Ant-v3). However, with the extreme value of 1.0, the effectiveness of the proposed BDETT scheme is reduced. This means that the sensitivity to the constant value increases as the complexity of the given task increases.

Note that the value of ‘0.2’ offers the most effective and robust performance across all three tasks under all experimental conditions. Therefore, we set the coefficient to ‘0.2’ in our proposed dynamic threshold scheme.

Table 19: Quantitative performance of obstacle avoidance tasks with different constant coefficient settings under degraded input conditions.

		LIF ($T = 5$)	**SRM** ($T = 5$)			**LIF** ($T = 5$)	**SRM** ($T = 5$)			**LIF** ($T = 5$)	**SRM** ($T = 5$)
Type	**Constant**	SR↑	SR↑	**Type**	**Constant**	SR↑	SR↑	**Type**	**Constant**	SR↑	SR↑
0.2	0.1	89%	78%	0.6	0.1	83.5%	82%	GN	0.1	83.5%	82%
	0.2(original)	**90%**	**79.5%**		0.2(original)	**84.5%**	**83%**		0.2(original)	**84.5%**	**82.5%**
	0.3	89%	77.5%		0.3	83%	81.5%		0.3	84%	82%
	0.4	87%	75%		0.4	82%	79%		0.4	84%	81.5%
	0.5	87.5%	75%		0.5	81.5%	80.5%		0.5	83%	81%
	1.0	83.5%	70.5%		1.0	76%	76.5%		1.0	77%	70.5%

Table 20: Quantitative performance of obstacle avoidance tasks with different constant coefficient settings under weight uncertainty conditions.

		LIF ($T = 5$)	**SRM** ($T = 5$)			**LIF** ($T = 5$)	**SRM** ($T = 5$)			**LIF** ($T = 5$)	**SRM** ($T = 5$)
Type	**Constant**	SR↑	SR↑	**Type**	**Constant**	SR↑	SR↑	**Type**	**Constant**	SR↑	SR↑
8-bit Loihi weight	0.1	89.5%	87.5%	GN weight (5 rounds)	0.1	87.2%	60.5%	30% Zero weight (5 rounds)	0.1	**77.2%**	64.0%
	0.2(original)	**90%**	**88.5%**		0.2(original)	**87.7%**	**61.8%**		0.2(original)	**77.2%**	**65.2%**
	0.3	**90%**	88%		0.3	86.3%	60.0%		0.3	75.8%	64.2%
	0.4	88.5%	88%		0.4	85.7%	58.6%		0.4	74.3%	63.5%
	0.5	87.5%	87%		0.5	84.1%	57.4%		0.5	72.8%	83.6%
	1.0	83%	79.5%		1.0	80.3%	52.3%		1.0	67.1%	52.9%

Table 21: Quantitative performance of HalfCheetah-v3 tasks with different constant coefficient settings under standard testing conditions.

	LIF ($T = 5$)	**SRM** ($T = 5$)
Constant	Reward↑	Reward↑
0.1	11029	11903
0.2(original)	**11064**	**11960**
0.3	10987	11875
0.4	10976	11682
0.5	10793	11724
1.0	10028	11123

Table 22: Quantitative performance of HalfCheetah-v3 tasks with different constant coefficient settings under degraded input conditions.

		LIF ($T = 5$)	**SRM** ($T = 5$)			**LIF** ($T = 5$)	**SRM** ($T = 5$)			**LIF** ($T = 5$)	**SRM** ($T = 5$)
Type	**Constant**	Reward↑	Reward↑	**Type**	**Constant**	Reward↑	Reward↑	**Type**	**Constant**	Reward↑	Reward↑
Random joint position	0.1	8379	7767	Random joint velocity	0.1	8241	7023	GN	0.1	3832	3825
	0.2(original)	**8465**	**7883**		0.2(original)	**8302**	**7116**		0.2(original)	**3909**	**3895**
	0.3	8302	7748		0.3	8159	6968		0.3	3790	3810
	0.4	8351	7703		0.4	8113	6743		0.4	3673	3724
	0.5	8188	7615		0.5	8044	6702		0.5	3711	3641
	1.0	7580	7180		1.0	7702	6231		1.0	3420	3172

Table 23: Quantitative performance of HalfCheetah-v3 tasks with different constant coefficient settings under weight uncertainty conditions.

		LIF ($T = 5$)	**SRM** ($T = 5$)			**LIF** ($T = 5$)	**SRM** ($T = 5$)			**LIF** ($T = 5$)	**SRM** ($T = 5$)
Type	**Constant**	Reward↑	Reward↑	**Type**	**Constant**	Reward↑	Reward↑	**Type**	**Constant**	Reward↑	Reward↑
8-bit Loihi weight	0.1	10780	11624	GN weight	0.1	6798	8142	30% Zero weight	0.1	6428	5250
	0.2(original)	**10823**	**11767**		0.2(original)	**6928**	**8381**		0.2(original)	**6551**	**5386**
	0.3	10672	11584		0.3	6920	8077		0.3	6531	5188
	0.4	10658	11467		0.4	6818	7936		0.4	6286	5102
	0.5	10583	11385		0.5	6674	7769		0.5	6290	4975
	1.0	9757	10648		1.0	6113	7019		1.0	5680	4562

Table 24: Quantitative performance of Ant-v3 tasks with different constant coefficient settings under standard testing conditions.

	LIF ($T = 5$)	**SRM** ($T = 5$)
Constant	Reward↑	Reward↑
0.1	5662	5803
0.2(original)	**5726**	**5879**
0.3	5648	5747
0.4	5570	5589
0.5	5394	5610
1.0	5104	5226

Table 25: Quantitative performance of Ant-v3 tasks with different constant coefficient settings under degraded inputs conditions.

		LIF ($T = 5$)	**SRM** ($T = 5$)			**LIF** ($T = 5$)	**SRM** ($T = 5$)			**LIF** ($T = 5$)	**SRM** ($T = 5$)
Type	**Constant**	Reward↑	Reward↑	**Type**	**Constant**	Reward↑	Reward↑	**Type**	**Constant**	Reward↑	Reward↑
Random joint position	0.1	3241	3368	Random joint velocity	0.1	2983	2772	GN	0.1	1214	1478
	0.2(original)	**3339**	**3450**		0.2(original)	**3103**	**2984**		0.2(original)	**1269**	**1559**
	0.3	3188	3380		0.3	3032	2704		0.3	1148	1409
	0.4	3213	3217		0.4	2844	2655		0.4	1003	1255
	0.5	3062	3048		0.5	2697	2517		0.5	980	1261
	1.0	2676	2572		1.0	2230	2280		1.0	772	1083

Table 26: Quantitative performance of Ant-v3 tasks with different constant coefficient settings under weight uncertainty conditions.

		LIF ($T = 5$)	**SRM** ($T = 5$)			**LIF** ($T = 5$)	**SRM** ($T = 5$)			**LIF** ($T = 5$)	**SRM** ($T = 5$)
Type	**Constant**	Reward↑	Reward↑	**Type**	**Constant**	Reward↑	Reward↑	**Type**	**Constant**	Reward↑	Reward↑
8-bit Loihi weight	0.1	5413	5600	GN weight	0.1	2703	1596	30% Zero weight	0.1	2883	2925
	0.2(original)	**5570**	**5648**		0.2(original)	**2782**	**1658**		0.2(original)	**2931**	**3046**
	0.3	5373	5583		0.3	2636	1554		0.3	2945	2990
	0.4	5230	5349		0.4	2488	1433		0.4	2802	2731
	0.5	5022	5224		0.5	2523	1382		0.5	2652	2583
	1.0	4448	4783		1.0	2205	1071		1.0	2217	1992

Table 27: Quantitative performance of obstacle avoidance tasks under the standard static obstacle conditions with respect to random seeds.

	LIF ($T = 5$)	**SRM** ($T = 5$)
Random Seed	SR↑	SR↑
1	**98.5%**	96%
2	**98.5%**	**96.5%%**
3	97%	95%
4	96.5%	95.5%
5	98%	96%
Mean	97.7%	95.8%
Standard Deviation	0.008	0.005
Coefficient of Variation	0.008	0.005

Table 28: Quantitative performance of obstacle avoidance tasks under dynamic obstacle conditions with respect to random seeds.

	LIF ($T = 5$)	**SRM** ($T = 5$)
Random Seed	SR↑	SR↑
1	92%	89.5%
2	**92.5%**	**90.5%**
3	91%	90.5%
4	92%	90%
5	92.5%	89.5%
Mean	92%	90%
Standard Deviation	0.005	0.004
Coefficient of Variation	0.006	0.005

Supplementary Note 8: Impact of Random Seeds

In this section, we study the impact of the random seeds during the training process on the proposed BDETT. For the obstacle avoidance tasks, we train five models for each LIF- and SRM-based host SNN, corresponding to five different random seeds. The SRs and the corresponding error bars of the trained host SNNs are reported in Figure 10a and Tables 27, 28, 29, and 30.

For the continuous control tasks, the rewards obtained in the HalfCheetah-v3 tasks under all experimental conditions are shown in Figure 10b. The corresponding experimental results are reported in Tables 31, 32, and 33. In Tables 34, 35, and 36, we report the experimental results of the Ant-v3 tasks under all experimental settings. The results are also illustrated in Figure 10c.

Since the mean success rates and rewards obtained in the three tasks differ significantly, we calculate coefficients of variation to produce fair comparisons. We observe that the random seeds have the lowest impact on the obstacle avoidance tasks and the most substantial influence on the Ant-v3 tasks.

Table 29: Quantitative performance of obstacle avoidance tasks under degraded input conditions with respect to random seeds.

		LIF ($T = 5$)	**SRM** ($T = 5$)			**LIF** ($T = 5$)	**SRM** ($T = 5$)			**LIF** ($T = 5$)	**SRM** ($T = 5$)
Type	**Random Seed**	SR↑	SR↑	**Type**	**Random Seed**	SR↑	SR↑	**Type**	**Random Seed**	SR↑	SR↑
0.2	1	89%	79%	0.6	1	83.5%	82%	GN	1	82.5%	81.5%
	2	**90%**	**79.5%**		2	**84.5%**	**83%**		2	**84.5%**	**82.5%**
	3	89.5%	79%		3	85%	82.5%		3	83%	81.5%
	4	88%	78.5%		4	84%	81.5%		4	83.5%	81%
	5	89%	78%		5	83%	82%		5	82%	82%
	Mean	89.1%	78.8%		Mean	84%	82.2%		Mean	83.1%	81.7%
	Standard Deviation	0.007	0.005		Standard Deviation	0.007	0.005		Standard Deviation	0.009	0.005
	Coefficient of Variation	0.007	0.006		Coefficient of Variation	0.008	0.006		Coefficient of Variation	0.010	0.006

Table 30: Quantitative performance of obstacle avoidance tasks under weight uncertainty conditions with respect to random seeds.

		LIF $(T = 5)$	**SRM** $(T = 5)$			**LIF** $(T = 5)$	**SRM** $(T = 5)$			**LIF** $(T = 5)$	**SRM** $(T = 5)$
Type	**Random Seed**	SR↑	SR↑	**Type**	**Random Seed**	SR↑	SR↑	**Type**	**Random Seed**	SR↑	SR↑
8-bit Loihi weight	1	**90%**	87%	GN weight (5 rounds)	1	85.8%	61%	30% Zero weight (5 rounds)	1	76.8%	64.5%
	2	**90%**	**88.5%**		2	**87.7%**	**61.8%**		2	**77.2%**	**65.2%**
	3	88.5%	86%		3	87.1%	61.3%		3	75.9%	63.9%
	4	88.5%	88.5%		4	86.6%	60.2%		4	75.4%	64.5%
	5	89%	86.5%		5	87.5%	60.8%		5	76.9%	65.0%
	Mean	89.2%	87.3%		Mean	86.9%	61%		Mean	76.4%	64.6%
	Standard Deviation	0.007	0.010		Standard Deviation	0.007	0.005		Standard Deviation	0.007	0.005
	Coefficient of Variation	0.008	0.012		Coefficient of Variation	0.008	0.009		Coefficient of Variation	0.009	0.007

Table 31: Quantitative performance of HalfCheetah-v3 tasks under standard testing condition with respect to random seeds.

	LIF $(T = 5)$	**SRM** $(T = 5)$
Random Seed	Reward↑	Reward↑
1	11064	11960
2	10979	11873
3	9848	10474
4	10881	11061
5	8992	11644
6	10977	10939
7	10975	11337
8	10869	11777
9	10932	10673
10	10993	11841
Mean	10651	11358
Standard Deviation	647	513
Coefficient of Variation	0.061	0.045

Table 32: Quantitative performance of the HalfCheetah-v3 tasks under degraded input conditions with respect to random seeds.

		LIF $(T = 5)$	**SRM** $(T = 5)$			**LIF** $(T = 5)$	**SRM** $(T = 5)$			**LIF** $(T = 5)$	**SRM** $(T = 5)$
Type	**Random Seed**	Reward↑	Reward↑	**Type**	**Random Seed**	Reward↑	Reward↑	**Type**	**Random Seed**	Reward↑	Reward↑
Random joint position	1	8465	7883	Random joint position	1	8302	7116	GN	1	3909	3895
	2	8452	7788		2	8239	7101		2	3820	3854
	3	7617	6618		3	7575	5684		3	3292	2648
	4	8399	7001		4	8274	6208		4	3797	2979
	5	7003	7622		5	7216	6869		5	2816	3561
	6	8320	7280		6	8226	6540		6	3675	3197
	7	8339	7557		7	8288	6806		7	3790	3636
	8	8348	7832		8	8129	6805		8	3743	3582
	9	8329	6777		9	8118	6044		9	3858	2770
	10	8423	7781		10	8208	6964		10	3806	3667
	Mean	8169	7414		Mean	8058	6614		Mean	3651	3379
	Standard Deviation	454	443		Standard Deviation	345	459		Standard Deviation	322	426
	Coefficient of Variation	0.056	0.06		Coefficient of Variation	0.043	0.069		Coefficient of Variation	0.088	0.126

Table 33: Quantitative performance of the HalfCheetah-v3 tasks under weight uncertainty conditions with respect to random seeds.

		LIF $(T = 5)$	**SRM** $(T = 5)$			**LIF** $(T = 5)$	**SRM** $(T = 5)$			**LIF** $(T = 5)$	**SRM** $(T = 5)$
Type	**Random Seed**	Reward↑	Reward↑	**Type**	**Random Seed**	Reward↑	Reward↑	**Type**	**Random Seed**	Reward↑	Reward↑
8-bit Loihi weight	1	10823	11767	GN weight	1	6928	8381	30% Zero weight	1	6551	5386
	2	10767	11749		2	6704	8307		2	6486	5308
	3	10062	10406		3	6155	7208		3	5671	3970
	4	10648	10780		4	6858	7527		4	6296	4254
	5	9368	11452		5	5857	8216		5	5217	5249
	6	10532	11097		6	6656	7537		6	6282	5007
	7	10284	11358		7	6838	8046		7	6341	5040
	8	10617	11768		8	6729	8306		8	6231	5337
	9	10788	10733		9	6740	7598		9	6537	4227
	10	10787	11496		10	6765	8282		10	6481	5081
	Mean	10468	11261		Mean	6623	7941		Mean	6209	4886
	Standard Deviation	435	460		Standard Deviation	324	407		Standard Deviation	410	501
	Coefficient of Variation	0.042	0.041		Coefficient of Variation	0.049	0.051		Coefficient of Variation	0.066	0.103

Table 34: Quantitative performance of the Ant-v3 tasks under standard testing condition with respect to random seeds.

	LIF $(T = 5)$	**SRM** $(T = 5)$
Random Seed	Reward↑	Reward↑
1	5726	5879
2	5678	5758
3	5306	5798
4	5553	5480
5	3980	5508
6	4657	5590
7	5692	4063
8	5595	5829
9	5688	5429
10	5696	5616
Mean	5357	4933
Standard Deviation	553	500
Coefficient of Variation	0.103	0.101

Table 35: Quantitative performance of Ant-v3 tasks under degraded input conditions with respect to random seeds.

		LIF $(T = 5)$	**SRM** $(T = 5)$			**LIF** $(T = 5)$	**SRM** $(T = 5)$			**LIF** $(T = 5)$	**SRM** $(T = 5)$
Type	**Random Seed**	Reward↑	Reward↑	**Type**	**Random Seed**	Reward↑	Reward↑	**Type**	**Random Seed**	Reward↑	Reward↑
Random joint position	1	3339	3450	Random joint position	1	3103	2984	GN	1	1269	1559
	2	3323	3423		2	2879	2869		2	1166	1524
	3	2973	3059		3	2651	2930		3	1030	1432
	4	3161	3049		4	2763	2731		4	1093	1357
	5	2053	3234		5	2105	2870		5	796	1440
	6	2424	3339		6	2547	2863		6	881	1396
	7	2993	2252		7	2822	2032		7	1132	830
	8	3216	3365		8	2892	2829		8	1105	1501
	9	3217	3300		9	3047	2815		9	1268	1464
	10	3218	3237		10	2860	2692		10	1020	1385
	Mean	2992	2847		Mean	2767	2492		Mean	1076	1250
	Standard Deviation	402	332		Standard Deviation	270	256		Standard Deviation	144	196
	Coefficient of Variation	0.134	0.117		Coefficient of Variation	0.098	0.103		Coefficient of Variation	0.134	0.157

Table 36: Quantitative performance of Ant-v3 tasks under weight uncertainty conditions with respect to random seeds.

		LIF ($T = 5$)	**SRM** ($T = 5$)			**LIF** ($T = 5$)	**SRM** ($T = 5$)			**LIF** ($T = 5$)	**SRM** ($T = 5$)
Type	**Random Seed**	Reward↑	Reward↑	**Type**	**Random Seed**	Reward↑	Reward↑	**Type**	**Random Seed**	Reward↑	Reward↑
8-bit Loihi weight	1	5570	5648	GN weight	1	2782	1658	30% Zero weight	1	2931	3046
	2	5398	5641		2	2668	1503		2	2849	2859
	3	5219	5611		3	2518	1467		3	2729	2674
	4	5309	5291		4	2580	1341		4	2741	2929
	5	4395	5201		5	1816	1522		5	1840	2794
	6	4853	5394		6	2303	1466		6	2125	2730
	7	5166	3968		7	2426	959		7	2654	1895
	8	5365	5610		8	2646	1595		8	2826	2760
	9	5544	5416		9	2601	1427		9	2874	2635
	10	5254	5343		10	2422	1497		10	2802	2644
	Mean	5207	4778		Mean	2476	1294		Mean	2637	2432
	Standard Deviation	333	473		Standard Deviation	257	181		Standard Deviation	342	294
	Coefficient of Variation	0.064	0.099		Coefficient of Variation	0.104	0.140		Coefficient of Variation	0.130	0.121

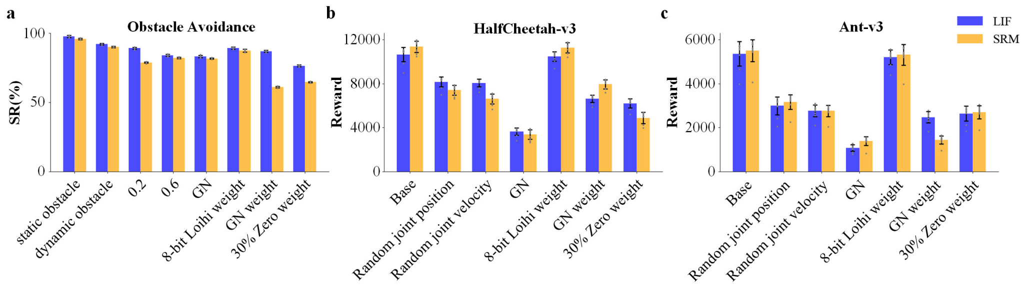

Figure 10: Quantitative performance of the LIF- and SRM-based BDETThost SNNs with respect to random seeds.

Supplementary Note 9: BDETT without Statistical Parameter Settings

We study the impact of the proposed layerwise statistical parameter settings, an extension of section 4.3 of our main paper. The experimental settings and the corresponding results are reported in Table 37. Without replacing the constants of the original biological model, the proposed method is is only slightly better than the ones without any training. Both LIF- and SRM-based experiments validate that the proposed statistical cues are essential to the proposed method.

Table 37: Quantitative performance of BDETT without statistical parameter settings (SPS) under the stardard testing conditions. OA means obstacle avoidance; HC-v3 indicates HalfCheetah-v3.

			LIF			**SRM**		
Approach	SPS	Trained	OA (SR↑)	HC-v3 (Reward↑)	Ant-v3 (Reward↑)	OA (SR↑)	HC-v3 (Reward↑)	Ant-v3 (Reward↑)
BDETT	Yes	Yes	92.5%	11064	5726	90.5%	11960	5879
BDETT	No	Yes	0%	-35	-9	0%	-28	-18
BDETT	Yes	No	0%	-124	-73	0%	-59	3