Supplementary Material of Homeostatic Reinforcement Learning by Soft-Behavior-Switching based on Internal Body Condition

Anonymous Author

A Hyperparameters of PPO

The implementation of PPO is based on CleanRL with substantial modifications [1]. In the original PPO implementation, the return after the terminal states is forced to zero. However, this treatment of the terminal state has a negative effect in homeostatic RL. This is because the idea of resetting in homeostatic RL is intended to prevent the agent's diverging behavior from the setpoint and end an unsuccessful episode. Moreover, forcing the future returns to zero means that the agent is fixed at the setpoint! To avoid this problem, we removed the zero-forcing at the terminal state and instead used the value estimation as an approximated future return.

Table 1: PPO hyperparameters used for low-dimensional experiments

Hyperparameter	Value
SGD update epochs	30
Learning rate (initial)	3×10^{-4}
Adam epsilon	10^{-5}
Maximum gradient norm	0.5
Discount factor (γ)	0.99
GAE lambda (λ)	0.95
Clipping parameter (ϵ)	0.3
Value loss clipping parameter	10
Value loss coefficient	0.5
Entropy coefficient	10^{-3}
# of minibatches	6
# of sampler threads	10

Table 2: Sampling Parameters in *Line* Environment

Hyperparameter	Value
Total time steps	5×10^6
# of rollout samples per thread in a single iteration	5×10^3

Table 3: Sampling Parameters in *Field* Environment

Hyperparameter	Value
Total time steps	1.5×10^6
# of rollout samples per thread in a single iteration	5×10^3

Table 4: Sampling Parameters in *TRP* and *Thermal* Environment

Hyperparameter	Value
Total time steps	8×10^7
# of rollout samples per thread in a single iteration	3×10^4

B Details of Environments for Benchmarking Homeostatic RL

B.1 Line Environment

In this environment, the agent can move in a one-dimensional space by applying a force to the point. The dynamics of the position of the agent is defined as follows:

$$\delta y_t = \mathrm{clip}(k_1 \delta y_t + k_2 a_t, -k_3, k_3) \tag{1}$$
$$y_t = \mathrm{clip}(y_t + v_t, 0, n_{\mathrm{resource}} + 1) \tag{2}$$

where $k_1 = 0.3$, $k_2 = 0.5$, $k_3 = 2$, v_t is the velocity of the agent at time step t, y_t is the agent's position at the corresponding time step, $\mathrm{clip}(x, a, b)$ is the clipping function that restricts x into the range $[a, b]$ by $\mathrm{clip}(x, a, b) = a$ if $x < a$ and $\mathrm{clip}(x, a, b) = b$ if $b < x$. Moreover, the agent has an n_{resource} dimensional continuous nutrient state and the environment has the corresponding number of resource supply site is (orange, blue, and green areas in the panel). The space for the agent's position is $y \in [0, n_{\mathrm{resource}} + 1]$ and each resource supply sites are placed at $y^j_{\mathrm{res}} = j$, where j is the index of the corresponding resource dimension ($j \in \{0, 1, \ldots, n_{\mathrm{resource}} - 1\}$). The agent can recharge the predefined amount of the associated resource at those resource sites. Concretely, the agent's nutrient dynamics are described as follows.

$$\delta \boldsymbol{z}^{\mathrm{inlet}}_t = k_4 \delta \boldsymbol{z}^{\mathrm{inlet}}_t + k_5 \boldsymbol{i}_t, \tag{3}$$
$$\delta \boldsymbol{z}_t = \mathrm{clip}(\delta \boldsymbol{z}^{\mathrm{inlet}}_t, -k_6, k_6) - k_7, \tag{4}$$
$$\boldsymbol{z}_t = \boldsymbol{z}_t + \delta \boldsymbol{z}_t, \tag{5}$$

where $k_4 = 0.5$, $k_5 = 0.02$, $k_6 = 0.1$, $k_7 = 0.005$, and $\boldsymbol{i}_t$ is the indicator function, where in the j-th element of the vector becomes $\boldsymbol{i}^j_t = 1$ if $y_t \in [y^j_{\mathrm{res}} - 0.2, y^j_{\mathrm{res}} + 0.2]$, otherwise $\boldsymbol{i}^j_t = 0$. In summary, the action space is one continuous space $a \in [-1, +1]$ and the state space is an eight-dimensional continuous space when $n_{\mathrm{resource}} = 3$, which is composed of

$$s \triangleq (y/(n_{\mathrm{resource}} + 1), \delta y/2, \delta \boldsymbol{z}, s_i), \tag{6}$$
$$s_i \triangleq \boldsymbol{z}, \tag{7}$$

When the environment is reset, the agent's position y and the nutrient state z are initialized randomly, whereas the velocity of each state δy and

δz are initialized by zero values. The initial y_0 and z_0 are sampled by an uniform distribution $y_0 \sim \mathcal{U}(0, n_{\text{resource}} + 1)$ and $z_0^j \sim \mathcal{U}(-1/6, 1/6)$ for all $j \in \{1, 2, \ldots, n_{\text{resource}}\}$. A single episode terminates if any dimension of the nutrient internal state exceeds the area $\|z_t^j\|_2 < 1$. Then the environment is reset accordingly. Finally, the reward is defined by $r = r_h$ in this environment.

B.2 Field Environment

This is the homeostatic RL environment with two-dimensional space. The agent can freely move in the squared area by applying the force to the point. The dynamics of the environment are the same as in the *Line* environment, with the differences that the position $\boldsymbol{y}$ is a two-dimensional space, and the resource supply sites are also two-dimensional. In this environment, the agent's position is restricted in the range $\boldsymbol{y}_t \in [-1, 1]^2$. The behavior of the indicator function $\boldsymbol{i}$ differs from the *Line* environment. The j-th element of the vector becomes $\boldsymbol{i}_t^j = 1$ if the distance between the agent's position and the j-th resource site are sufficiently close ($\|\boldsymbol{y}_t - \boldsymbol{y}_{\text{res}}^j\|_2 \leq 0.2$), otherwise $\boldsymbol{i}_t^j = 0$. Each resource site is placed at the position $\boldsymbol{y}_{\text{res}}^j = (0.4\cos(\omega j), 0.4\sin(\omega j))^\top$ for the j-th resource type, where $\omega = 2\pi/n_{\text{resource}}$. Finally, the parameter of the internal nutrient dynamics uses $k_7 = 0.001$. This environment has a two-dimensional continuous action space $a \in [-1, +1]^2$ and the state space is the ten-dimensional continuous space when $n_{\text{resource}} = 3$, which is composed of

$$s \triangleq (\boldsymbol{y}, \delta\boldsymbol{y}/2, \delta\boldsymbol{z}, s_i), \tag{8}$$

$$s_i \triangleq \boldsymbol{z}. \tag{9}$$

B.3 Two-Resource Problem Environment

This environment contains an agent and randomly distributed food resources in the field. The agent was developed using the *Mujoco* dynamics simulator [2]. The environment contains a quadrupedal robot *Ant* [3] in a food-gathering environment [4, 5]. We used a *low-gear* Ant model asset as suggested by previous studies [5,6]. The agent had a two-dimensional continuous nutritional state $\boldsymbol{z}$ that corresponded to its nutritional level. Two types of food resources (represented with four red balls and six blue balls) corresponded to dimensions of the agent's nutritional state. If the agent's torso

(body sphere) position gets within the range of a food resource (< 1 m in the simulator), the food is consumed, and the nutritional state is recharged with a predefined quantity. Subsequently, new food resources are generated at random positions. We used a simple metabolic model of the nutritional state described in a previous study [6,7]. The updates of the nutritional state $\boldsymbol{z}$ is described as

$$\boldsymbol{z}_{t+1} = \boldsymbol{z}_t + k_1^{\mathrm{TRP}} \boldsymbol{i}_t - k_2^{\mathrm{TRP}}, \tag{10}$$

where k_1^{TRP} is the scaling factor of the nutritional inlet from foods, k_2^{TRP} is the default consumption of resources. $\boldsymbol{i}_t^j$ is the indicator function of the capture of the food for j-th resource at time step t. We employed the: $k_1^{\mathrm{TRP}} = 0.1$ and $k_2^{\mathrm{TRP}} = 0.00015$ as is suggested in [6, 7]. During the training stage, a single episode starts from nutritional states uniformly sampled from $\mathcal{U}[-\frac{1}{6}, \frac{1}{6}]$ for each nutrient. As in previous environments, an episode terminates if any dimension of the nutrient internal state exceeds the area $\|z_t^j\|_2 < 1$. The action of the agent was the motor torque of each joint. The dimension of control was eight, and the control space was normalized to the eight-dimensional cube $a \in [-1, 1]^8$. The agent's observations are composed of a 40-dimensional exteroception vector $\boldsymbol{s}_e$, a 27-dimensional proprioception vector $\boldsymbol{s}_p$ and a two-dimensional interoception vector $\boldsymbol{s}_i$. Hence,

$$s \triangleq (\boldsymbol{s}_e, \boldsymbol{s}_p, \boldsymbol{s}_i), \tag{11}$$

$$s_i \triangleq \boldsymbol{z}. \tag{12}$$

Here, exteroception refers to the agent's perceptual signals outside of the body, composed of range sensor stimuli (20 different directions around the agent) for two types of food resources. Proprioception is the 27-dimensional observation that reports self-movement and positions of the agent's body; in this study, it was assumed to include the agent's joint angles, rotational and positional speed, height and posture information of the torso [3]. Interoception is the agent's two-dimensional internal nutritional state $s_i \triangleq \boldsymbol{z}$. The reward in this environment is defined by a homeostatic term in addition to the cost terms

$$r = r_h - 0.005\|\tilde{p} - \tilde{p}^*\|_2^2 - 0.001\|a\|_2^2, \tag{13}$$

where $\tilde{p}$ and $\tilde{p}^*$ are agent's rotational angles of the torso and its target value (the agent's torso standing vertically).

B.4 Thermal Regulation Environment

In this environment, the agent needs to regulate its core body temperature through environmental interactions while maintaining energy homeostasis. The arena of the environment is similar to that of *TRP* . Six food resources were randomly distributed over a square area with randomly generated terrain. We used temperature dynamics in addition to one-dimensional nutritional dynamics. The dynamics of the core temperature is obtained from a previous study [6], which is based on the thermal dynamics of a reptile [8, 9] and the heat dynamics on an electric motor [10]. The dynamics of the body temperature τ is described by differential equation $C\frac{d\tau}{dt} = \delta Q(\tau, u, u_{\mathrm{ev}})$, where τ is the agent's core body temperature, C is the representative heat capacity of the agent, and δQ is the amount of heat added to its body. A detailed calculation of δQ are provided in [6]. The agent performed a nine-dimensional action $a \in [-1, 1]^9$, comprising an eight-dimensional motor output, and a one-dimensional evaporative action. We used the normalized body temperature by mapping $\tau \in [307, 315]$ in Kelvin degrees to $\tilde{\tau} = [-1, 1]$, and the setpoint of the normalized body core temperature was set to zero. The agent receives the same exteroception s_e and proprioception s_p as those in the *TRP* environment, where the range finder stimulus represents the distance to food resources (red balls, 20 dimensions). The interoceptive signal s_i is defined as a two-dimensional signal composed of a one-dimensional nutrient state and the normalized core temperature of the agent $s_i \triangleq (\boldsymbol{z}, \tilde{\tau})$. The full-observation in this environment is

$$\boldsymbol{s} \triangleq (\boldsymbol{s}_e, \boldsymbol{s}_p, \boldsymbol{s}_i), \tag{14}$$

$$\boldsymbol{s}_i \triangleq (z, \tilde{\tau}). \tag{15}$$

In the thermal environment the reward as is defined in the *TRP* .

C Results of the environments with different number of resources

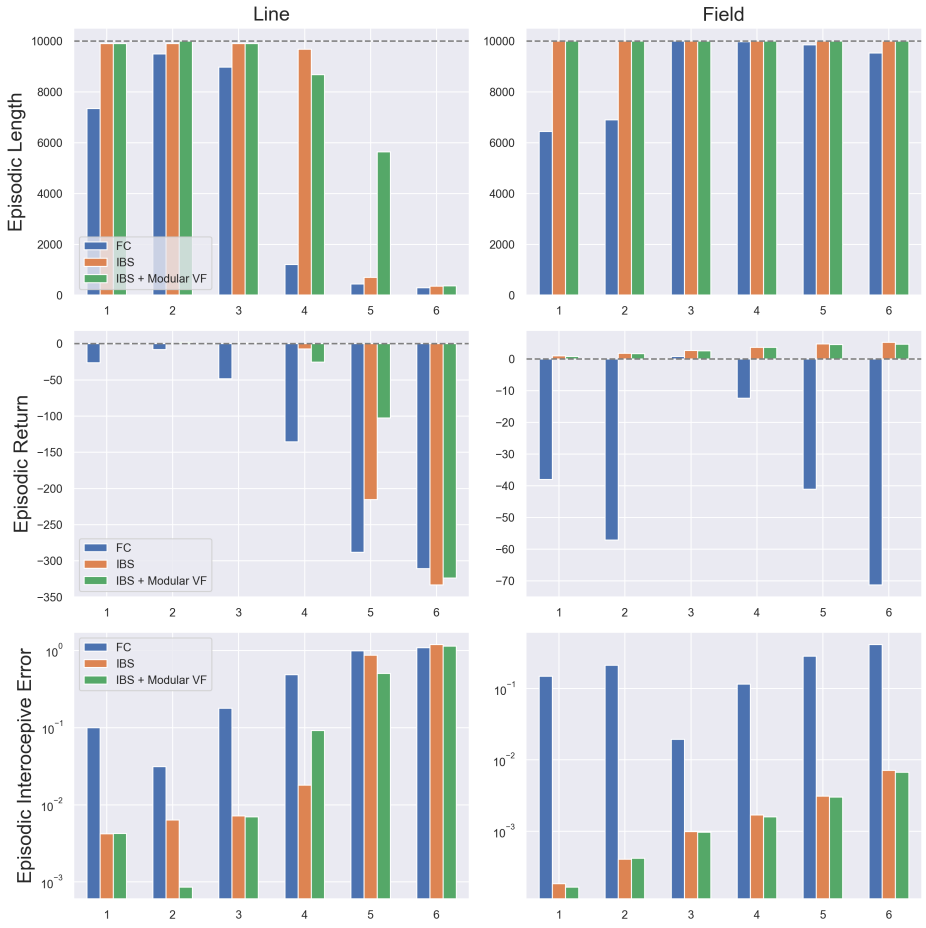

Figure 1: Evaluation of the modular value function architecture (MVF) in homeostatic RL environments with various numbers of resource types.

D Results of the evaluation of Modular Value Function in Benchmark Environments

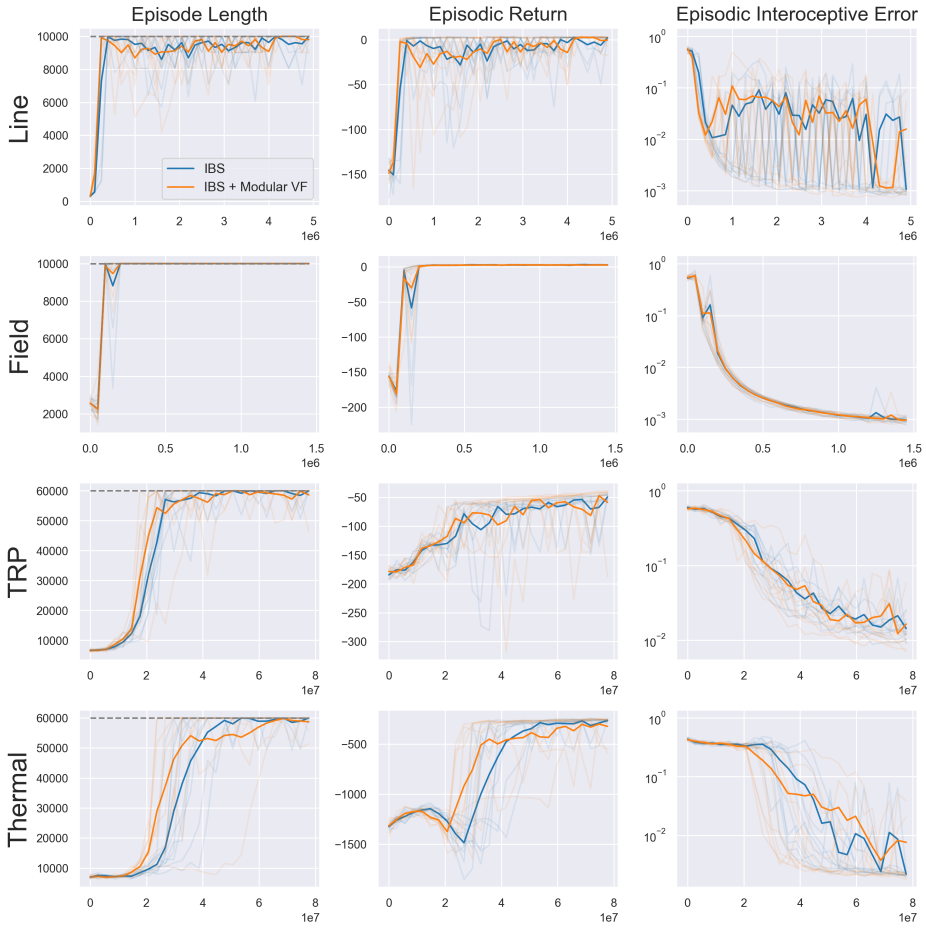

Figure 2: Evaluation of interoceptive behavior-switching (IBS) and the modular value function architecture (IBS+MVF) in four homeostatic RL environments.

References

[1] S. Huang, R. F. J. Dossa, C. Ye, J. Braga, D. Chakraborty, K. Mehta, and J. G. Araújo, "Cleanrl: High-quality single-file implementations of deep reinforcement learning algorithms," *Journal of Machine Learning Research*, vol. 23, no. 274, pp. 1–18, 2022. [Online]. Available: http://jmlr.org/papers/v23/21-1342.html

[2] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2012, pp. 5026–5033.

[3] J. Schulman, P. Moritz, S. Levine, M. Jordan, and P. Abbeel, "High-dimensional continuous control using generalized advantage estimation," in *International Conference on Learning Representations (ICLR)*, 2016.

[4] Y. Duan, X. Chen, R. Houthooft, J. Schulman, and P. Abbeel, "Benchmarking deep reinforcement learning for continuous control," in *International conference on machine learning*. PMLR, 2016, pp. 1329–1338.

[5] A. Li, C. Florensa, I. Clavera, and P. Abbeel, "Sub-policy adaptation for hierarchical reinforcement learning," in *International Conference on Learning Representations (ICLR)*, 2020.

[6] N. Yoshida, T. Daikoku, Y. Nagai, and Y. Kuniyoshi, "Embodiment perspective of reward definition for behavioural homeostasis," in *Deep RL Workshop NeurIPS 2021*, 2021.

[7] G. Konidaris and A. Barto, "An adaptive robot motivational system," in *From Animals to Animats 9*. Springer, 2006, pp. 346–356.

[8] W. P. Porter, J. W. Mitchell, W. A. Beckman, and C. B. DeWitt, "Behavioral implications of mechanistic ecology," *Oecologia*, vol. 13, no. 1, pp. 1–54, 1973.

[9] T. Fei, A. K. Skidmore, V. Venus, T. Wang, M. Schlerf, B. Toxopeus, S. Van Overjijk, M. Bian, and Y. Liu, "A body temperature model for lizards as estimated from the thermal environment," *Journal of Thermal Biology*, vol. 37, no. 1, pp. 56–64, 2012.

[10] B. Venkataraman, B. F. Godsey, W. Premerlani, E. Shulman, M. Thakur, and R. Midence, "Fundamentals of a motor thermal model and its applications in motor protection," in *58th Annual Conference for Protective Relay Engineers, 2005.* IEEE, 2005, pp. 127–144.