A Compact Networks for Neural Policies

To obtain compact neural representations, there are three common approaches: 1) simply choose an RNN with small number of units densely wired to each other (e.g., a long short-term memory, LSTM, network [30], or a continuous-time network such as an ordinary differential equation, ODE, -based network [40, 31]). 2) sparsify a large network into a smaller system (e.g., lottery ticket winners [41], or sparse flows [42]), and 3) use neural circuit policies that are given by sparse architectures with added complexity to their neural and synaptic representations but have a light-weighted network architecture [3, 4, 32].

In the first approach the number of model parameters inversely affect interpretability, i.e., interpreting wider and/or deeper densely wired RNNs exponentially makes the interpretation of the system harder. Sparsity has been shown to help obtain a network with 95% less parameters compared to the initial model. However, recent studies show that such levels of sparsity affect the robustness of the model, thus make it more susceptible to perturbations [43]. Neural circuit policies (NCPs) [4] on the other hand have shown great promise in achieving attractive degrees of generalizability while maintaining robustness to environmental perturbations. This representation learning capability is rooted in their ability to capture the true cause and effect of a given task [5]. NCPs are sparse network architectures with their nodes and edges determined by a liquid time-constant (LTC) concept [3]. The state of a liquid network is described by the following set of ODEs [3]: $\frac{d\mathbf{x}(t)}{dt} = -\left[\frac{1}{\tau} + f(\mathbf{x}(t), \mathbf{I}(t), t, \theta)\right] \odot \mathbf{x}(t) + f(\mathbf{x}(t), \mathbf{I}(t), t, \theta) \odot A$. Here, $\mathbf{x}^{(D\times 1)}(t)$ is the hidden state with size D, $\mathbf{I}^{(m\times 1)}(t)$ is an input signal, $\tau^{(D\times 1)}$ is the fixed internal time-constant vector, $A^{(D\times 1)}$ is a bias parameter, and $\odot$ is the Hadamard product. In tasks involving spatiotemporal dynamics these networks showed significant benefit over their counterparts, both in their ODE form and in their closed-form representation termed Closed-form continuous-time (CfC) models [4, 5, 32].

Interpretation of Neuron Responses. Compact neural representations promise to enable the interpretability of decision-making by focusing post-hoc analysis on a limited number of neural responses. However, having merely a lower-dimensional space for visualization is not sufficient to identify consistent behaviors or strategies acquired by a learning agent. Emergent behaviors may distribute responses across numerous neurons with a high degree of entanglement. Even for models with a small number of neurons, it can be challenging to identify and interpret the behavior correlated with observed response patterns. In this paper, we hypothesize that abstraction with respect to a type of learned strategy within a single neuron is necessary for better interpretability of neural policies. We further desire semantic grounding of the neuron response, that is, associating neuron response to human-readable representation. The representation space should be abstract enough to be human-understandable and expressive enough to capture arbitrary types of emergent behaviors or strategies. We adopt the framework of logic programs due to their simple yet effective representations of decision-making processes.

B A Motivating Perspective Of Disentangled Representation

The underlying behaviors of neural policies involves descriptions with multiple levels of abstraction, from detailed states at every time instance to high-level strategies toward solving a task, spanning a continuum where the details can be summarized and reduced to gradually construct their concise counterparts. Among these descriptions of behaviors, a right amount of abstraction should be concise enough for human interpretability yet being sufficiently informative of how neural policies act locally toward solving the overall task. Relevant concepts about abstraction have been explored in the context of state abstraction in Markov Decision Process (MDP) [44], hierarchical reinforcement learning [45], and developmental psychology [46]. In the following, we aim to more formally define such abstraction for interpretability of neural policies and draw connection to disentangled representations. First, we define a MDP as a tuple $\{\mathcal{S}, \mathcal{A}, P_a, R\}$, where at time instance t, $s_t \in \mathcal{S}$ is the state, $a_t \in \mathcal{A}$ is the action, $P_a : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$ is the transition function, $R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function. The goal of policy learning in a MDP is to find a policy $\pi : \mathcal{S} \times \mathcal{A} \to [0, 1]$

that maximizes the expected future return (accumulated reward). The closed-loop dynamics (in deterministic setting) can then be written as

$$s_{t+1} = P_a(s_t, a_t), \quad \text{where } a_t = \pi(s_t)$$

Then, we construct an abstract MDP, with state presumably being the abstraction we are looking for interpretability, as a tuple $\{\hat{\mathcal{S}}, \hat{\mathcal{A}}, \hat{P}_a, \hat{R}\}$ that follows similar definition to the above-mentioned regular MDP. It follows the deterministic MDP homomorphism [47, 48] as follows,

$$\forall s_t, s_{t+1} \in \mathcal{S}, a_t \in \mathcal{A} \quad P_a(s_t, a_t) = s_{t+1} \Rightarrow \bar{P}_a(Q(s_t), \bar{A}(a_t)) = Q(s_{t+1})$$
$$\forall s_t \in \mathcal{S}, a_t \in \mathcal{A} \quad R(s_t, a_t) = R(Q(s_t), \bar{A}(a_t))$$

where $Q : \mathcal{S} \to \hat{\mathcal{S}}$ is the state embedding function and $\bar{A} : \mathcal{A} \to \hat{\mathcal{A}}$ is the action embedding function. The state embedding function can also be seen as an action-equivariant map that precisely satisfies the MDP homomorphism [48]. Next, we start to draw connection to disentangled representation from one of its formalism using symmetries and group theory [49]. Informally, disentanglement refers to the level of decomposition in representation that reflects the *factor of variation*. For example, one dimension of vector representations corresponds to color and the other corresponds to shape. In [49], these factor of variations are formally defined as symmetries of world state ($\mathcal{S}$ in our case). Given group G, binary operator $\circ : G \times G \to G$, group decomposition into a direct product of subgroups $G = G_1 \times G_2 \times \ldots$, and group action $\cdot_X : G \times X \to X$ with X as a set which the group action act upon, the idea is to "commute" symmetries from one set X to the other X'. Suppose there is a group G of geometries acting on the world state $\mathcal{S}$ via action $\cdot_{\mathcal{S}} : G \times \mathcal{S} \to \mathcal{S}$, we would like to find a corresponding action acting on representation $\cdot_Z : G \times Z \to Z$ that reflects the symmetric structure of $\mathcal{S}$ in $\mathcal{Z}$ (in our case neuron response $z_t \in Z$). This entails the equivariance condition,

$$g \cdot_Z E_{\mathcal{S} \to Z}(s_t) = E_{\mathcal{S} \to Z}(g \cdot_{\mathcal{S}} s_t)$$

where $E_{\mathcal{S} \to Z}$ commutes action across $\mathcal{S}$ and Z, and can be called a G-morphism or equivariant map.

$$\begin{array}{ccc} G \times \mathcal{S} & \xrightarrow{\cdot_{\mathcal{S}}} & \mathcal{S} \\ {\scriptstyle id_G \times E_{\mathcal{S} \to Z}}\downarrow & & \downarrow{\scriptstyle E_{\mathcal{S} \to Z}} \\ G \times Z & \xrightarrow{\cdot_Z} & Z \end{array}$$

A more concrete connection of group action to MDP can be seen in the analogy of agent-environment interaction [50],

$$g \cdot_{\mathcal{S}} s_t = s_{t+1} = P_a(s_t, a_t)$$

It is worth emphasizing the distinction of group action $\cdot_{\mathcal{S}}$ and regular action a_t: not all regular action a_t exhibit symmetry, as pointed out in [50]. And the group action upon neural state $\cdot_Z$ can be viewed as the transition dynamics of neural policies,

$$g \cdot_Z z_t = z_{t+1} = \pi_z(z_t) = \pi_s(P_a(s_t, \pi_a(z_t))$$

where $\pi = \pi_a \circ \pi_s, \pi_a : Z \to \mathcal{A}, \pi_s : \mathcal{S} \to Z$ is simply the decomposition of neural policies to explicitly extract neuron responses z_t and $\pi_z : Z \to Z$ is the transition function of neural states (note that this does not necessarily require recurrence structure of neural policies; instead this is more of a convenient notation here). Following the definition of [49], an agent's representation Z is disentangled with respect to the decomposition $G = G_1 \times G_2 \times \ldots$ if

1. There is a group action $\cdot_Z : G \times Z \to Z$.
2. The map $E_{\mathcal{S} \to Z} : \mathcal{S} \to Z$ is equivariant between the group actions on $\mathcal{S}$ and Z.
3. There is a decomposition $Z = Z_1 \times Z_2 \times \ldots$ such that each Z_i is fixed by the actions of all $G_j, j \neq i$ and affected only by G_i.

For the first condition, We already define $\cdot_Z$ in the above. For the second condition, we show that the equivariant map can follow the definition $E_{\mathcal{S} \to Z} = \pi_z \circ \pi_s$, i.e., $z_{t+1} = E_{\mathcal{S} \to Z}(s_t)$. This follows the proof as,

$$g \cdot_Z E_{\mathcal{S} \to Z}(s_t) = g \cdot_Z z_{t+1} = \pi_z(z_{t+1}) = \pi_z(\pi_s(s_{t+1})) = (\pi_z \circ \pi_s)(s_{t+1}) = E_{\mathcal{S} \to Z}(g \cdot_{\mathcal{S}} s_t)$$

Next, extending the formalism of disentangled representation in [49] with the above-mentioned MDP homomorphism [47], we define the equivariance condition between the regular MDP $\{\mathcal{S}, \mathcal{A}, P_a, R\}$ and the abstract MDP $\{\hat{\mathcal{S}}, \hat{\mathcal{A}}, \hat{P}_a, \hat{R}\}$,

$$g \cdot_{\mathcal{S}} E_{\hat{\mathcal{S}}\to\mathcal{S}}(\hat{s}_t) = E_{\hat{\mathcal{S}}\to\mathcal{S}}(g \cdot_{\hat{\mathcal{S}}} \hat{s}_t)$$

where $E_{\hat{\mathcal{S}}\to\mathcal{S}}$ commutes action across $\hat{\mathcal{S}}$ and $\mathcal{S}$, and can be defined with MDP homomorphism,

$$\begin{aligned}
\hat{s}_{t+1} &= Q(s_{t+1}) \\
\hat{P}_a(\hat{s}_t, \hat{a}_t) &= Q(P_a(s_t, a_t)) \\
g \cdot_{\hat{\mathcal{S}}} \hat{s}_t &= Q(g \cdot_{\mathcal{S}} s_t) \\
Q^{-1}(g \cdot_{\hat{\mathcal{S}}} \hat{s}_t) &= g \cdot_{\mathcal{S}} Q^{-1}(\hat{s}_t) \\
E_{\hat{\mathcal{S}}\to\mathcal{S}} &= Q^{-1}
\end{aligned}$$

Note that theoretically the state embedding function Q may not have an inverse mapping since going from $\mathcal{S}$ to $\hat{\mathcal{S}}$ is supposed to be more abstract (and thus concise with equal or less information). However, this does not matter since we don't necessarily require this recipe to tell us how exactly group actions in $\hat{\mathcal{S}}$ commute to $\mathcal{S}$. Overall, we establish the following group homomorphism across set $\hat{\mathcal{S}}$, $\mathcal{S}$, and Z,

$$\begin{array}{ccc}
G \times \hat{\mathcal{S}} & \xrightarrow{\cdot_{\hat{\mathcal{S}}}} & \hat{\mathcal{S}} \\
{\scriptstyle id_G \times E_{\hat{\mathcal{S}}\to\mathcal{S}}}\downarrow & & \downarrow{\scriptstyle E_{\hat{\mathcal{S}}\to\mathcal{S}}} \\
G \times \mathcal{S} & \xrightarrow{\cdot_{\mathcal{S}}} & \mathcal{S} \\
{\scriptstyle id_G \times E_{\mathcal{S}\to Z}}\downarrow & & \downarrow{\scriptstyle E_{\mathcal{S}\to Z}} \\
G \times Z & \xrightarrow{\cdot_{Z}} & Z
\end{array}$$

This connects *the right amount of abstraction for interpretability* discussed in the beginning, then associated with MDP homomorphism, to *factor of variation* in disentangled representation, which is formalized by symmetry and group theory. Disentanglement in Z can then be lifted to symmetries in abstract state space $\hat{\mathcal{S}}$. In [49], disentanglement of representation is lifted up to the symmetries in the world state space $\mathcal{S}$, e.g., a factor of group decomposition G_i can be color of an object. However, this is not sufficient to describe the behavior of policies since $\mathcal{S}$ lacks task structure. Hence, we further go from $\mathcal{S}$ to $\hat{\mathcal{S}}$ with MDP homomorphism to capture the essence of solving a task. The factor of group decomposition G_i can then be task-related, e.g., relative pose to a target object (which may be of high interest for tasks like object tracjing, and less so for tasks like locomotion). Overall, this provides a motivation to cast the problem of searching for proper description of the behavior of neural policies (for interpretability) to searching for disentanglement in neuron responses. In this paper, we therefore study how to measure interpretability of compact neural policies with disentangled representation.

C Calibration Of Mutual Information

Lemma C.1. *The calibration term $I[z^j; \mathcal{P}_k] - I[z^j; \mathcal{P}_k; \mathcal{P}_{k^j}]$ in both MIG (5) and Modularity (6) metrics, for $j \neq i^*$, without loss of generality has the following lower bound:*

$$I[z^j; \mathcal{P}_k] - I[z^j; \mathcal{P}_k; \mathcal{P}_{k^j}] \geq \max(0,\ I[z^j; \mathcal{P}_k] - I[\mathcal{P}_k; \mathcal{P}_{k^j}]) \tag{7}$$

Lemma C.1 is necessary because to compute the calibration term, we need access to the conditional distribution of the random variable $(\mathcal{P}_{k^j}|z^j)$, which is normally inaccessible. Hence, we derive a lower bound for the calibrated mutual information.

Proof. In the main paper, we adapting Mutual Information Gap (MIG) [20] to our framework as,

$$\frac{1}{K}\sum_{k=1}^{K}\frac{1}{H[\mathcal{P}_k]}\Big(I[z^{i^*}; \mathcal{P}_k] - \max_{j\neq i^*} I[z^j; \mathcal{P}_k] - I[z^j; \mathcal{P}_k; \mathcal{P}_{k^j}]\Big)$$

and Modularity score [28] as,

$$\frac{1}{\mathcal{I}}\sum_{i\in\mathcal{I}} 1 - \frac{\sum_{k\neq k^*}(I[z^i;\mathcal{P}_k] - I[z^i;\mathcal{P}_k;\mathcal{P}_{k^*}])^2}{(K-1)I[z^i;\mathcal{P}_{k^*}]^2}$$

Both involve the computation of $I[z^j;\mathcal{P}_k;\mathcal{P}_{k^j}]$. Without loss of generality for both cases (and with the notation of MIG), we simplify the calibration term for $j \neq i^*$ as follows,

$$\begin{aligned}
&I[z^j;\mathcal{P}_k] - I[z^j;\mathcal{P}_k;\mathcal{P}_{k^j}] \\
&= I[z^j;\mathcal{P}_k] - (I[z^j;\mathcal{P}_k] - I[z^j;\mathcal{P}_k|\mathcal{P}_{k^j}]) \\
&= I[z^j;\mathcal{P}_k|\mathcal{P}_{k^j}] \\
&= I[z^j;\mathcal{P}_k] + H[\mathcal{P}_{k^j}|z^j] + H[\mathcal{P}_{k^j}|\mathcal{P}_k] - H[\mathcal{P}_{k^j}|z^j,\mathcal{P}_k] - H[\mathcal{P}_{k^j}] \\
&= I[z^j;\mathcal{P}_k] - (H[\mathcal{P}_{k^j}] - H[\mathcal{P}_{k^j}|\mathcal{P}_k]) + (H[\mathcal{P}_{k^j}|z^j] - H[\mathcal{P}_{k^j}|z^j,\mathcal{P}_k]) \\
&= I[z^j;\mathcal{P}_k] - I[\mathcal{P}_k;\mathcal{P}_{k^j}] + I[\mathcal{P}_{k^j}|z^j;\mathcal{P}_k] \\
&\geq \max(0,\ I[z^j;\mathcal{P}_k] - I[\mathcal{P}_k;\mathcal{P}_{k^j}])
\end{aligned}$$

Most steps simply follow identities of mutual information and entropy. The last step requires access to the conditional distribution of random variable $(\mathcal{P}_{k^j}|z^j)$, which is normally inaccessible. Hence, we introduce an approximation that serves as a lower bound for the calibrated mutual information in our implementation. □

D Other Quantitative Measures

Decision Path Accuracy. During deployment, we use an inverse proxy q_{ϕ^i} for the decision tree T_{θ^i} and hence we compute the approximation error by measuring the accuracy of a state-grounded decision path inferred from the neuron response with q_{ϕ^i} compared to true states,

$$\frac{1}{|\mathcal{I}|}\sum_{i\in\mathcal{I}}\frac{1}{|\mathcal{D}_{\text{dt}}|}\sum_{(s_t,z_t^i)\in\mathcal{D}_{\text{dt}}}\frac{1}{|q_{\phi^i}(z_t^i)|}\sum_{\substack{n\in q_{\phi^i}(z_t^i)\\ j=g(n)}}\mathbb{1}[s_t^j \leq c_n] \tag{8}$$

where $\mathbb{1}$ is an indicator function, $q_{\phi^i}(z_t^i)$ is the inferred decision path with norm as number of decision rules. The condition $s_t^j \leq c_n$ validates if the current state s_t^j complies with the inferred rule defined by c_n (which is from T_{θ^i}). Since the discrepancy is computed at the decision rule level, it captures not only the error of the classifier model q_{ϕ^i} but also how accurately $f_{\mathcal{S}}^i$ parses z_t^i.

Cross-neuron Logic Conflict. When interpreting a neural policy as a whole instead of inspecting individual neuron response, it is straightforward to find the intersection across logic programs extracted from different neurons $l_t = \texttt{reduce}(\wedge_{i\in\mathcal{I}}\, l_t^i)$, where `reduce` summarizes and reduces logic programs to a more compact one. Intuitively, the neuron-wise logic program should summarize the operational domain of the strategy currently executed by the neuron, where intersection describes the domain of a joint strategy across neurons. However, the reduction of intersection can be invalid if there is conflict in the logical formulae across neurons, e.g., $a \leq 3$ from the first neuron and $a \geq 4$ from the second neuron. The conflict may imply, under the same configuration of $f_{\mathcal{S}}$, that (1) the policy fails to learn compatible strategies across neurons or (2) there is an error induced by the interpreter due to insufficient or ambiguous connection between the logic program and the neuron response, which implicitly indicates lack of interpretability.

Experimental Results. For classical control, we verify in Table 7 that all models achieve comparable performance when learning toward target -500 episode reward. For locomotion, in Table 8, most models achieve comparable task performance except for GRU and ODE-RNN being slightly worse. For end-to-end visual servoing, in Table 9, all models achieve good performance (> 0.9) except for ODE-RNN, which fails to learn a good policy within maximal training iterations.

Table 7: Other quantitative results of classical control.

Network Architecture	Decision Path Accuracy ↑	Logic Conflict ↓	Performance ↑
FCs	0.3015	0.2104	-488.55
GRU	0.2504	0.2832	-559.82
LSTM	0.2392	0.5072	**-467.95**
ODE-RNN	0.2980	0.2506	-533.93
CfC	0.2509	**0.1556**	-489.28
NCP	**0.4726**	0.2026	-556.64

Table 8: Other quantitative results of locomotion.

Network Architecture	Decision Path Accuracy ↑	Logic Conflict ↓	Performance ↑
FCs	0.5285	**0.1035**	5186.50
GRU	0.4924	0.1500	3857.21
LSTM	0.5283	0.2155	4122.74
ODE-RNN	0.4959	0.1474	3472.69
CfC	0.4841	0.1581	5195.46
NCP	**0.5859**	0.1105	**5822.73**

E Implementation Details

NCPs are designed by a four-layer structure consisting of sensory neurons (input layer), interneurons, command neurons (with recurrent connections), and motor neurons (output layer). To make a fair comparison, we augment all non-NCP models by a feed-forward layer, which is of equivalent size to the inter-neuron layer in NCPs.

E.1 Classical Control (Pendulum)

Network Architecture. With 3-dimensional observation space and 1-dimensional action space, we use the following network architecture for compact neural policies.

- *FCs*: a $3 \to 10 \to 4 \to 1$ fully-connected network with *tanh* activation.
- *GRU*: a $3 \to 10$ fully-connected network with *tanh* activation followed by GRU with cell size of 4, outputting a 1-dimensional action.
- *LSTM*: a $3 \to 10$ fully-connected network with *tanh* activation followed by LSTM with hidden size of 4, outputting a 1-dimensional action. Note that this effectively gives 8 cells by considering hidden and cell states.
- *ODE-RNN*: a $3 \to 10$ fully-connected network with *tanh* activation followed by a neural ODE with recurrent component both of size 4, outputting a 1-dimensional action.
- *CfC*: with backbone layer $= 1$, backbone unit $= 10$, backbone activation *silu*, hidden size $= 4$ without gate and mixed memory, outputting a 1-dimensional action.
- *NCP*: with 3 sensory neurons, 10 interneuron, 4 command neurons, 1 motor neuron, 4 output sensory synapses, 3 output inter-synapses, 2 recurrent command synapse, 3 motor synapses.

For all policies, we use a $3 \to 64 \to 64 \to 1$ fully-connected networks with *tanh* activation as value function. We interpret the layer of size 4 for each policy.

Training details. We use PPO with the following parameters for all models. Learning rate is 0.0003. Train batch size (of an epoch) is 512. Mini-batch size is 64. Number of iteration within a batch is 6. Value function clip parameter is 10.0. Discount factor of the MDP is 0.95. Generalized advantage estimation parameter is 0.95. Initial coefficient of KL divergence is 0.2. Clip parameter is 0.3. Training halts if reaching target average episode reward 150. Maximal training steps is 1M.

Table 9: Other quantitative results of visual servoing.

Network Architecture	Decision Path Accuracy ↑	Logic Conflict ↓	Performance ↑
FCs	0.5379	0.1354	**1.0000**
GRU	**0.6160**	0.1884	0.9210
LSTM	0.5174	0.4504	**1.0000**
ODE-RNN	0.5483	0.3786	0.4239
CfC	0.5549	0.2274	0.9922
NCP	0.5960	**0.1067**	**1.0000**

Interpreter details. For the decision tree T_{θ^i}, we set minimum number of samples required to be at a leaf node as 10% of the training data, criterion of a split as mean squared error with Friedman's improvement score, the maximum depth of the tree as 3, complexity parameter used for minimal cost-complexity pruning as 0.003; we use scikit-learn implementation of CART (Classification and Regression Trees). For simplicity, we use another decision tree as decision path classifier q_{ϕ^i} with maximal depth of tree as 3, minimum number of samples in a leaf node as 1% of data, complexity parameter for pruning as 0.01, criterion as Gini impurity. The state grounding $\mathcal{S}$ of the interpreter $f^i_{\mathcal{S}}$ is $\{\theta, \dot{\theta}\}$, where θ is joint angle and $\dot{\theta}$ is joint angular velocity.

E.2 Locomotion (HalfCheetah)

Network Architecture. With 17-dimensional observation space and 6-dimensional action space, we first use feature extractors of a shared architecture as a $17 \to 256$ fully-connected network, which then output features to compact neural policies with the following architectures,

- *FCs*: a $256 \to 20 \to 10 \to 6$ fully-connected network with *tanh* activation.
- *GRU*: a $256 \to 20$ fully-connected network with *tanh* activation followed by GRU with cell size of 10, outputting a 6-dimensional action.
- *LSTM*: a $256 \to 20$ fully-connected network with *tanh* activation followed by LSTM with hidden size of 10, outputting a 6-dimensional action. Note that this effectively gives 20 cells by considering hidden and cell states.
- *ODE-RNN*: a $256 \to 20$ fully-connected network with *tanh* activation followed by a neural ODE with recurrent component both of size 10, outputting a 6-dimensional action.
- *CfC*: with backbone layer $= 1$, backbone unit $= 20$, backbone activation *silu*, hidden size $= 10$ without gate and mixed memory.
- *NCP*: with 256 sensory neurons, 20 interneuron, 10 command neurons, 6 motor neuron, 4 output sensory synapses, 5 output inter-synapses, 6 recurrent command synapse, 4 input motor synapses.

For all policies, we use a $17 \to 256 \to 256 \to 1$ fully-connected networks with *tanh* activation as value function. We interpret the layer of size 10 for each policy.

Training details. We use PPO with the following parameters for all models. Learning rate is 0.0003. Train batch size (of an epoch) is 65536. Mini-batch size is 4096. Number of iteration within a batch is 32. Value function coefficient is 10.0. Discount factor of the MDP is 0.99. Generalized advantage estimation parameter is 0.95. Initial coefficient of KL divergence is 1.0. Clip parameter is 0.2. Gradient norm clip is 0.5. Training halts if reaching target average episode reward -500. Maximal training steps is 12M.

Interpreter details. For the decision tree T_{θ^i}, we set minimum number of samples required to be at a leaf node as 10% of the training data, criterion of a split as mean squared error with Friedman's improvement score, the maximum depth of the tree as 3, complexity parameter used for minimal cost-complexity pruning as 0.001; we use scikit-learn implementation of CART (Classification and Regression Trees). For simplicity, we use another decision tree as decision path classifier q_{ϕ^i} with

maximal depth of tree as 3, minimum number of samples in a leaf node as 1% of data, complexity parameter for pruning as 0.01, criterion as Gini impurity. The state grounding $\mathcal{S}$ of the interpreter $f^i_{\mathcal{S}}$ is $\{h_R, \theta_R, \theta_{T,B}, \theta_{S,B}, \theta_{F,B}, \theta_{T,F}, \theta_{S,F}, \theta_{F,F}, \dot{x}_R, \dot{h}_R, \dot{\theta}_R, \dot{\theta}_{T,B}, \dot{\theta}_{S,B}, \dot{\theta}_{F,B}, \dot{\theta}_{T,F}, \dot{\theta}_{S,F}, \dot{\theta}_{F,F}\}$, where $h_R, \dot{h}_R$ are position and velocity of z-coordinate of the front tip, $\theta_R, \dot{\theta}_R$ are angle and angular velocity of the front tip, $\theta_{T,B}, \dot{\theta}_{T,B}$ are angle and angular velocity of the thigh in the back, $\theta_{S,B}, \dot{\theta}_{S,B}$ are angle and angular velocity of the shin in the back, $\theta_{F,B}, \dot{\theta}_{F,B}$ are angle and angular velocity of the foot in the back, $\theta_{T,T}, \dot{\theta}_{T,T}$ are angle and angular velocity of the thigh in the front, $\theta_{S,T}, \dot{\theta}_{S,T}$ are angle and angular velocity of the shin in the front, $\theta_{F,T}, \dot{\theta}_{F,T}$ are angle and angular velocity of the foot in the front, $\dot{x}_R$ is the velocity of x-coordinate of the front tip.

E.3 End-to-end visual servoing (Image-based Driving)

Network Architecture. With image observation space of size $(200, 320, 3)$ and 2-dimensional action space, we first use feature extractors of a shared architecture as a convolutional neural network (CNN) in Table 10, which then output features to compact neural policies with the following architectures,

- *FCs*: a $1280 \to 20 \to 8 \to 2$ fully-connected network with *tanh* activation.
- *GRU*: a $1280 \to 20$ fully-connected network with *tanh* activation followed by GRU with cell size of 8, outputting a 2-dimensional action.
- *LSTM*: a $1280 \to 20$ fully-connected network with *tanh* activation followed by LSTM with hidden size of 8, outputting a 2-dimensional action. Note that this effectively gives 20 cells by considering hidden and cell states.
- *ODE-RNN*: a $1280 \to 20$ fully-connected network with *tanh* activation followed by a neural ODE with recurrent component both of size 8, outputting a 2-dimensional action.
- *CfC*: with backbone layer $= 1$, backbone unit $= 20$, backbone activation *silu*, hidden size $= 8$ without gate and mixed memory.
- *NCP*: with 1280 sensory neurons, 20 interneuron, 8 command neurons, 2 motor neuron, 4 output sensory synapses, 5 output inter-synapses, 6 recurrent command synapse, 4 input motor synapses.

Training details. Batch size is 64. Sequence size is 10. Learning rate is 0.001. Number of epochs is 10. We perform data augmentation on RGB images with randomized gamma of range $[0.5, 1.5]$, brightness of range $[0.5, 1.5]$, contrast of range $[0.7, 1.3]$, saturation of range $[0.5, 1.5]$.

Interpreter details. For the decision tree T_{θ^i}, we set minimum number of samples required to be at a leaf node as 10% of the training data, criterion of a split as mean squared error with Friedman's improvement score, the maximum depth of the tree as 3, complexity parameter used for minimal cost-complexity pruning as 0.003; we use scikit-learn implementation of CART (Classification and Regression Trees). For simplicity, we use another decision tree as decision path classifier q_{ϕ^i} with maximal depth of tree as 3, minimum number of samples in a leaf node as 1% of data, complexity parameter for pruning as 0.01, criterion as Gini impurity. The state grounding $\mathcal{S}$ of the interpreter $f^i_{\mathcal{S}}$ is $\{v, \delta, d, \Delta l, \mu, \kappa\}$, where v is vehicle speed, δ is heading, d is lateral deviation from the lane center, Δl is longtitudinal deviation from the lane center, μ is local heading error with respect to the lane center, κ is road curvature.

F Robustness Analysis

We propose to study the interpretability of neural policies through decision trees and present several quantitative measures of interpretability by analyzing various properties on top of neuron responses and corresponding decision trees, including *Neural-Response Variance*, *Mutual Information Gap*, *Modularity*, *Decision Path Accuracy*, and *Logic Conflict*. However, the extracted decision trees may differ across different configurations. Hence, to validate the robustness of the proposed metrics to hyperparameters, we compute all metrics with different decision tree parameters in classical control

Layer	Hyperparameters
Conv2d	(3, 24, 5, 2, 2)
GroupNorm2d	(16, 1e-5)
ELU	-
Dropout	0.3
Conv2d	(24, 36, 5, 2, 2)
GroupNorm2d	(16, 1e-5)
ELU	-
Dropout	0.3
Conv2d	(36, 48, 3, 2, 1)
GroupNorm2d	(16, 1e-5)
ELU	-
Dropout	0.3
Conv2d	(48, 64, 3, 1, 1)
GroupNorm2d	(16, 1e-5)
ELU	-
Dropout	0.3
Conv2d	(64, 64, 3, 1, 1)
AdaptiveAvgPool2d	reduce height dimension

Table 10: Network architecture of CNN feature extractor for end-to-end visual servoing. Hyperparameters for *Conv2d* are input channel, output channel, kernel size, stride, and padding; for *GroupNorm2d*, they are group size and epsilon; for *Dropout*, it is drop probability.

environment (Pendulum). We report the averaged results with 5 random seeds in Table 11 (*Neural-Response Variance*), Table 12 (*Mutal Information Gap*), Table 13 (*Modularity*), Table 14 (*Decision Path Accuracy*), Table 15 (*Logic Conflict*). Most metrics (variance, MI-gap, decision path accuracy, logic conflict) yield consistent top-1 results and agree with similar rankings among network architectures, except for modularity that is slightly less robust against hyperparameters yet still consistent in the top-3 set of models. This results demonstrate the reliability of the proposed interpretability analysis for neural policies.

Table 11: Robustness to hyperparameters for *Neural-Response Variance*. The results are averaged across 5 random seeds in classical control (Pendulum).

[Variance ↓] Network Architecture		FCs	GRU	LSTM	ODE-RNN	CfC	NCP
Cost Complexity Pruning	0.001	0.0232	0.0304	0.0209	0.0266	0.0254	**0.0207**
	0.003	0.0242	0.0329	**0.0216**	0.0287	0.0272	0.0240
	0.01	0.0261	0.0371	**0.0221**	0.0315	0.0267	0.0305
Minimal Leaf Sample Ratio	0.01	0.0154	0.0261	**0.0138**	0.0193	0.0189	0.0186
	0.1	0.0242	0.0329	**0.0216**	0.0287	0.0272	0.0240
	0.2	0.0334	0.0387	**0.0284**	0.0354	0.0295	0.0285

Table 12: Robustness to hyperparameters for *Mutual Information Gap*. The results are averaged across 5 random seeds in classical control (Pendulum).

[MI-Gap ↑] Network Architecture		FCs	GRU	LSTM	ODE-RNN	CfC	NCP
Cost Complexity Pruning	0.001	0.0284	0.2686	0.2026	0.2891	0.2544	**0.3403**
	0.003	0.3008	0.2764	0.2303	0.3062	0.2892	**0.3653**
	0.01	0.3482	0.3065	0.2547	0.3142	0.3567	**0.3664**
Minimal Leaf Sample Ratio	0.01	0.2824	0.2632	0.2040	0.2819	0.2433	**0.3456**
	0.1	0.3008	0.2764	0.2303	0.3062	0.2892	**0.3653**
	0.2	**0.3798**	0.3387	0.2528	0.3168	0.3342	0.3429

Table 13: Robustness to hyperparameters for *Modularity*. The results are averaged across 5 random seeds in classical control (Pendulum).

[Modularity ↑] Network Architecture		FCs	GRU	LSTM	ODE-RNN	CfC	NCP
Cost Complexity Pruning	0.001	**0.9519**	0.9558	0.9327	0.9485	0.9228	0.9438
	0.003	0.9471	0.9550	0.9402	0.9486	0.9116	**0.9551**
	0.01	0.9532	**0.9598**	0.9445	0.9487	0.8970	0.9593
Minimal Leaf Sample Ratio	0.01	0.9638	**0.9702**	0.9547	0.9630	0.9333	0.9651
	0.1	0.9471	0.9550	0.9402	0.9486	0.9116	**0.9551**
	0.2	**0.9475**	0.9372	0.9197	0.9404	0.8755	0.9301

Table 14: Robustness to hyperparameters for *Decision Path Accuracy*. The results are averaged across 5 random seeds in classical control (Pendulum).

[Decision Path Accuracy ↑] Network Architecture		FCs	GRU	LSTM	ODE-RNN	CfC	NCP
Cost Complexity Pruning	0.001	0.2815	0.2415	0.2195	0.2904	0.2250	**0.4294**
	0.003	0.3015	0.2504	0.2392	0.2980	0.2509	**0.4726**
	0.01	0.3074	0.3330	0.3161	0.3707	0.2864	**0.4390**
Minimal Leaf Sample Ratio	0.01	0.2950	0.2637	0.2270	0.2574	0.2452	**0.4287**
	0.1	0.3015	0.2504	0.2392	0.2980	0.2509	**0.4726**
	0.2	0.3572	0.3587	0.2794	0.3322	0.2784	**0.4684**

G Counterfactual Analysis via Removal of Neurons

There exist some neurons with logic programs that are sensible but may have little effect on task performance. For example, in NCPs (not confined to this specific architecture but just focus on it for discussion), we find a neuron that aligns its response purely with vehicle speed. Given the task objective is lane following without crashing, such neuron pays attention to useful (for temporal reasoning across frames) but relatively unnecessary (to the task) information. Furthermore, there are neurons that don't exhibit sufficient correlation with any of the environment state and fail to induce decision branching. In light of these observation, we try to remove neurons that we suspect to have little influence on the performance by inspecting their logic program. We show the results in Table 16. Removing neurons 3, 4, 7 has a marginal impact on task performance. Among them, neuron 3 and 4 mostly depends on vehicle speed v with a small tendency to the lateral deviation d. Neuron 7 fails to split a tree.

H Interpretation Of Driving Maneuver

In Figure 3, we describe interpretations similar to classical control (for a neuron in NCP). While the state space of driving is higher dimensional (5 with bicycle model for lane following), states of interest only include local heading error μ and lateral deviation from the lane center d in lane following task. We compute the statistics and plot neuron response and closed-loop dynamics in the d-μ phase portrait. This specific neuron develops more fine-grained control for situations when the vehicle is on the right of the lane center, as shown in Figure 3(a). We further show front-view images retrieved based on neuron response in Figure 3(b).

I Logic Program from Decision Trees

Here we show the corresponding logic program of the finite set of decision path $\{r(\mathcal{P}_k^i)\}_{k=1}^{K^i}$ for every interpreted neuron in all network architectures. The symbols used in the logic program follow the state grounding definition in Section E. We also briefly summarize the size of associated decision trees by computing the number of decision rules for each model (before logic program reduction and conflict checking).

Table 15: Robustness to hyperparameters for *Logic Conflict*. The results are averaged across 5 random seeds in classical control (Pendulum).

[Logic Conflict ↓] Network Architecture		FCs	GRU	LSTM	ODE-RNN	CfC	NCP
Cost Complexity Pruning	0.001	0.2451	0.3348	0.5240	0.2641	**0.2048**	0.3159
	0.003	0.2104	0.2832	0.5072	0.2506	**0.1556**	0.2026
	0.01	0.1766	0.1877	0.4325	0.1401	**0.1121**	0.2924
Minimal Leaf Sample Ratio	0.01	0.2672	0.4298	0.6791	0.3575	**0.2654**	0.2607
	0.1	0.2104	0.2832	0.5072	0.2506	**0.1556**	0.2026
	0.2	0.1796	0.1664	0.3842	0.2001	**0.1089**	0.1111

Table 16: Removing a single neuron based on explanation.

Remove Neuron	0	1	2	3	4	5	6	7
Performance ↑	0.24	0.07	0.09	1.00	0.969	0.29	0.03	1.00

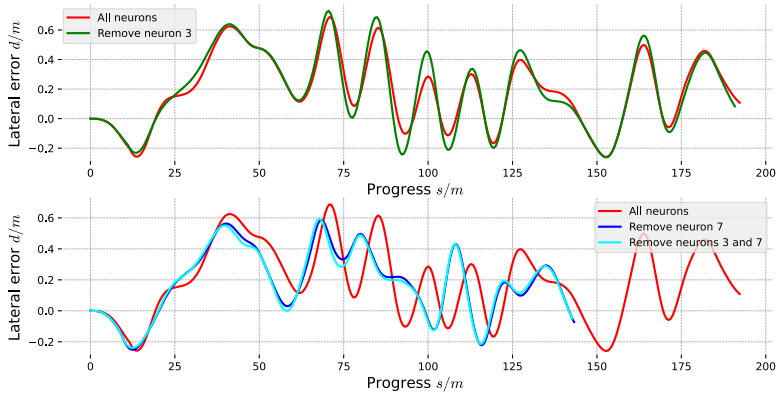

Figure 4: Driving profile when removing neurons according to decision tree interpretation.

In classical control (Pendulum), the extracted logic program are shown in Table 17 (FC; of size 39), Table 18 (GRU; of size 54), Table 19 (LSTM; of size 43), Table 20 (ODE-RNN; of size 40), Table 21 (CfC; of size 20), Table 22 (NCP; of size 26).

In locomotion (HalfCheetah), the extracted logic program are shown in Table 23 (FC; of size 171), Table 24 (GRU; of size 158), Table 25 (LSTM; of size 148), Table 26 (ODE-RNN; of size 156), Table 27 (CfC; of size 149), Table 28 (NCP; of size 81).

In end-to-end visual servoing (Image-based Driving), the extracted logic program are shown in Table 29 (FC; of size 92), Table 30 (GRU; of size 60), Table 31 (LSTM; of size 70), Table 32 (ODE-RNN; of size 94), Table 33 (CfC; of size 107), Table 34 (NCP; of size 66).

In a logic program, ”conflict” indicates there are conflict between predicates within the logic program as elaborated in Section 3.2.

Model	Neuron	Logic Program
FC	0	0: $(\dot{\theta} <= 0.69) \wedge (\theta <= -2.18)$ 1: $(\dot{\theta} > 0.69) \wedge (\theta <= -2.18)$ 2: (conflict) 3: $(\theta <= 2.41) \wedge (\theta > -2.18)$ 4: $(\theta > 2.41)$
	1	0: $(\dot{\theta} <= -1.16) \wedge (\theta <= -0.34)$ 1: $(\dot{\theta} <= 1.73) \wedge (\dot{\theta} > -1.16) \wedge (\theta <= -0.34)$ 2: $(\dot{\theta} > 1.73) \wedge (\theta <= -0.34)$ 3: $(\theta <= 2.03) \wedge (\theta > -0.34)$ 4: $(\theta <= 2.62) \wedge (\theta > 2.03)$ 5: $(\theta > 2.62)$
	2	0: $(\dot{\theta} <= -1.54) \wedge (\theta <= -1.68)$ 1: $(\dot{\theta} <= 1.47) \wedge (\dot{\theta} > -1.54) \wedge (\theta <= -1.68)$ 2: $(\dot{\theta} > 1.47) \wedge (\theta <= -1.68)$ 3: (conflict) 4: $(\theta <= 2.48) \wedge (\theta > -1.68)$ 5: $(\theta > 2.48)$
	3	0: $(\theta <= -2.76)$ 1: (conflict) 2: $(\theta <= 0.05) \wedge (\theta > -2.76)$ 3: $(\theta > 0.05)$

Table 17: Logic program of FC in classical control (Pendulum).

Model	Neuron	Logic Program
GRU	0	0: $(\theta <= -0.06)$ 1: (conflict) 2: $(\dot{\theta} <= -0.30) \wedge (\theta > -0.06)$ 3: $(\dot{\theta} <= 1.75) \wedge (\dot{\theta} > -0.30) \wedge (\theta > -0.06)$ 4: $(\dot{\theta} > 1.75) \wedge (\theta > -0.06)$
	1	0: $(\dot{\theta} <= -2.30) \wedge (\theta <= -1.27)$ 1: $(\dot{\theta} <= 1.83) \wedge (\dot{\theta} > -2.30) \wedge (\theta <= -1.27)$ 2: $(\dot{\theta} <= -0.37) \wedge (\theta > -1.27)$ 3: $(\dot{\theta} <= 1.83) \wedge (\dot{\theta} > -0.37) \wedge (\theta > -1.27)$ 4: $(\dot{\theta} <= 3.10) \wedge (\dot{\theta} > 1.83)$ 5: $(\dot{\theta} > 3.10)$
	2	0: $(\dot{\theta} <= -0.11) \wedge (\theta <= -0.05)$ 1: $(\dot{\theta} <= -0.11) \wedge (\theta > -0.05)$ 2: (conflict) $\wedge (\theta > -0.05)$ 3: $(\dot{\theta} > -0.11) \wedge (\theta <= -2.09)$ 4: $(\dot{\theta} > -0.11) \wedge (\theta <= 0.41) \wedge (\theta > -2.09)$ 5: $(\dot{\theta} <= 1.61) \wedge (\dot{\theta} > -0.11) \wedge (\theta > 0.41)$ 6: $(\dot{\theta} > 1.61) \wedge (\theta > 0.41)$
	3	0: $(\theta <= -2.61)$ 1: $(\dot{\theta} <= 2.44) \wedge (\theta <= 0.21) \wedge (\theta > -2.61)$ 2: $(\dot{\theta} > 2.44) \wedge (\theta <= 0.21) \wedge (\theta > -2.61)$ 3: $(\dot{\theta} <= -1.76) \wedge (\theta > 0.21)$ 4: $(\dot{\theta} <= 0.39) \wedge (\dot{\theta} > -1.76) \wedge (\theta > 0.21)$ 5: $(\dot{\theta} > 0.39) \wedge (\theta > 0.21)$

Table 18: Logic program of GRU in classical control (Pendulum).

Model	Neuron	Logic Program
LSTM	0	0: $(\theta <= -2.16)$ 1: (conflict) 2: $(\theta <= 0.01) \wedge (\theta > -2.16)$ 3: $(\dot{\theta} <= 0.48) \wedge (\theta > 0.01)$ 4: $(\dot{\theta} <= 3.00) \wedge (\dot{\theta} > 0.48) \wedge (\theta > 0.01)$ 5: $(\dot{\theta} > 3.00) \wedge (\theta > 0.01)$
	1	0: $(\dot{\theta} <= -2.57)$ 1: (conflict) 2: $(\dot{\theta} > -2.57) \wedge (\theta <= 0.35)$ 3: $(\dot{\theta} > -2.57) \wedge (\theta <= 2.03) \wedge (\theta > 0.35)$ 4: $(\dot{\theta} > -2.57) \wedge (\theta > 2.03)$
	2	0: $(\dot{\theta} <= 4.79) \wedge (\theta <= -2.31)$ 1: $(\dot{\theta} <= 4.79) \wedge (\theta <= 1.72) \wedge (\theta > -2.31)$ 2: $(\dot{\theta} <= -3.13) \wedge (\theta > 1.72)$ 3: $(\dot{\theta} <= 4.79) \wedge (\dot{\theta} > -3.13) \wedge (\theta > 1.72)$ 4: $(\dot{\theta} > 4.79)$
	3	0: $(\theta <= -2.32)$ 1: $(\theta <= 0.93) \wedge (\theta > -2.32)$ 2: $(\dot{\theta} <= -3.98) \wedge (\theta > 0.93)$ 3: $(\dot{\theta} > -3.98) \wedge (\theta <= 2.04) \wedge (\theta > 0.93)$ 4: $(\dot{\theta} > -3.98) \wedge (\theta > 2.04)$

Table 19: Logic program of LSTM in classical control (Pendulum).

Model	Neuron	Logic Program
ODE-RNN	0	0: $(\dot{\theta} <= -1.48) \wedge (\theta <= -0.02)$ 1: $(\dot{\theta} <= -1.48) \wedge (\theta > -0.02)$ 2: $(\dot{\theta} > -1.48)$
	1	0: $(\dot{\theta} <= -0.08) \wedge (\theta <= -1.45)$ 1: $(\dot{\theta} > -0.08) \wedge (\theta <= -1.45)$ 2: $(\theta <= 2.05) \wedge (\theta > -1.45)$ 3: $(\theta <= 2.50) \wedge (\theta > 2.05)$ 4: $(\dot{\theta} <= -0.40) \wedge (\theta > 2.50)$ 5: $(\dot{\theta} <= 0.03) \wedge (\dot{\theta} > -0.40) \wedge (\theta > 2.50)$ 6: $(\dot{\theta} > 0.03) \wedge (\theta > 2.50)$
	2	0: $(\dot{\theta} <= -0.56)$ 1: $(\dot{\theta} > -0.56) \wedge (\theta <= -2.16)$ 2: $(\dot{\theta} > -0.56) \wedge (\theta <= 2.44) \wedge (\theta > -2.16)$ 3: (conflict) $\wedge (\theta > 2.44)$ 4: $(\dot{\theta} > -0.56) \wedge (\theta > 2.44)$
	3	0: $(\theta <= -2.18)$ 1: $(\theta <= 0.04) \wedge (\theta > -2.18)$ 2: $(\theta <= 2.65) \wedge (\theta > 0.04)$ 3: $(\dot{\theta} <= -0.21) \wedge (\theta > 2.65)$ 4: $(\dot{\theta} > -0.21) \wedge (\theta > 2.65)$

Table 20: Logic program of ODE-RNN in classical control (Pendulum).

Model	Neuron	Logic Program
CfC	0	0: $(\theta <= -0.03)$ 1: (conflict) 2: (conflict) 3: $(\theta > -0.03)$
	1	0: $(\theta <= -0.05)$ 1: (conflict) 2: (conflict) 3: $(\theta > -0.05)$
	2	0: $(\theta <= -2.02)$ 1: $(\theta > -2.02)$
	3	0: $(\theta <= -0.12)$ 1: $(\theta <= 0.24) \wedge (\theta > -0.12)$ 2: $(\theta <= 2.14) \wedge (\theta > 0.24)$ 3: $(\theta > 2.14)$

Table 21: Logic program of CfC in classical control (Pendulum).

Model	Neuron	Logic Program
NCP	0	0: $(\dot{\theta} <= 0.33)$ 1: $(\dot{\theta} > 0.33)$
	1	0: $(\theta <= -0.07)$ 1: (conflict) 2: $(\theta <= 0.27) \wedge (\theta > -0.07)$ 3: $(\theta > 0.27)$
	2	0: $(\dot{\theta} <= 4.80) \wedge (\theta <= -1.27)$ 1: $(\dot{\theta} <= 4.80) \wedge (\theta <= 1.66) \wedge (\theta > -1.27)$ 2: $(\dot{\theta} <= 4.80) \wedge (\theta > 1.66)$ 3: $(\dot{\theta} > 4.80)$
	3	0: $(\dot{\theta} <= -0.33)$ 1: $(\dot{\theta} <= 0.44) \wedge (\dot{\theta} > -0.33)$ 2: $(\dot{\theta} > 0.44) \wedge (\theta <= -1.31)$ 3: $(\dot{\theta} > 0.44) \wedge (\theta <= 1.44) \wedge (\theta > -1.31)$ 4: $(\dot{\theta} > 0.44) \wedge (\theta > 1.44)$

Table 22: Logic program of NCP in classical control (Pendulum).

Model	Neuron	Logic Program
FC	0	0: $(\dot{\theta}_R <= -0.22) \wedge (\dot{\theta}_{T,F} <= -3.26)$ 1: $(\dot{\theta}_R > -0.22) \wedge (\dot{\theta}_{T,F} <= -3.26)$ 2: $(\dot{\theta}_{T,F} <= 6.46) \wedge (\dot{\theta}_{T,F} > -3.26) \wedge (\theta_{F,F} <= -0.50)$ 3: $(\dot{\theta}_{T,F} <= 6.46) \wedge (\dot{\theta}_{T,F} > -3.26) \wedge (\theta_{F,F} > -0.50)$ 4: $(\dot{\theta}_{T,F} > 6.46) \wedge (h_R <= 0.05)$ 5: $(\dot{\theta}_{T,F} > 6.46) \wedge (h_R > 0.05)$
	1	0: $(\theta_{F,B} <= -0.05) \wedge (\theta_{S,F} <= 0.61) \wedge (\theta_{T,B} <= 0.05)$ 1: $(\theta_{F,B} <= -0.05) \wedge (\theta_{S,F} > 0.61) \wedge (\theta_{T,B} <= 0.05)$ 2: $(\dot{\theta}_{T,F} <= -11.24) \wedge (\theta_{F,B} > -0.05) \wedge (\theta_{T,B} <= 0.05)$ 3: $(\dot{\theta}_{T,F} > -11.24) \wedge (\theta_{F,B} > -0.05) \wedge (\theta_{T,B} <= 0.05)$ 4: $(\theta_{T,B} > 0.05) \wedge (\theta_{T,F} <= 0.40)$ 5: $(\theta_{T,B} > 0.05) \wedge (\theta_{T,F} <= 0.62) \wedge (\theta_{T,F} > 0.40)$ 6: $(\dot{\theta}_{T,B} <= 2.08) \wedge (\theta_{T,B} > 0.05) \wedge (\theta_{T,F} > 0.62)$ 7: $(\dot{\theta}_{T,B} > 2.08) \wedge (\theta_{T,B} > 0.05) \wedge (\theta_{T,F} > 0.62)$
	2	0: $(\dot{\theta}_{F,F} <= -12.45) \wedge (\theta_{F,B} <= 0.06)$ 1: $(\dot{\theta}_{F,F} > -12.45) \wedge (\theta_{F,B} <= 0.06) \wedge (\theta_{S,F} <= 0.65)$ 2: $(\dot{\theta}_{F,F} > -12.45) \wedge (\theta_{F,B} <= 0.06) \wedge (\theta_{S,F} > 0.65)$ 3: $(\dot{\theta}_{T,B} <= 6.33) \wedge (\theta_{F,B} > 0.06) \wedge (\theta_{T,F} <= 0.33)$ 4: $(\dot{\theta}_{T,B} <= 6.33) \wedge (\theta_{F,B} > 0.06) \wedge (\theta_{T,F} > 0.33)$ 5: $(\dot{\theta}_{T,B} > 6.33) \wedge (\theta_{F,B} > 0.06)$
	3	0: $(\theta_{F,B} <= 0.38) \wedge (\theta_{S,F} <= 0.17) \wedge (\theta_{T,F} <= 0.58)$ 1: $(\theta_{F,B} <= 0.38) \wedge (\theta_{S,F} <= 0.17) \wedge (\theta_{T,F} > 0.58)$ 2: $(\theta_{F,B} <= 0.38) \wedge (\theta_{S,F} > 0.17) \wedge (\theta_{T,B} <= 0.07)$ 3: $(\theta_{F,B} <= 0.38) \wedge (\theta_{S,F} > 0.17) \wedge (\theta_{T,B} > 0.07)$ 4: $(\theta_{F,B} > 0.38)$
	4	0: $(\dot{\theta}_{S,B} <= 1.79) \wedge (\theta_R <= 0.19) \wedge (\theta_{T,B} <= 0.06)$ 1: $(\dot{\theta}_{S,B} > 1.79) \wedge (\theta_R <= 0.19) \wedge (\theta_{T,B} <= 0.06)$ 2: $(\dot{\theta}_{F,F} <= 9.07) \wedge (\theta_R > 0.19) \wedge (\theta_{T,B} <= 0.06)$ 3: $(\dot{\theta}_{F,F} > 9.07) \wedge (\theta_R > 0.19) \wedge (\theta_{T,B} <= 0.06)$ 4: $(\theta_R <= -0.03) \wedge (\theta_{T,B} > 0.06)$ 5: $(\theta_R > -0.03) \wedge (\theta_{F,F} <= -0.49) \wedge (\theta_{T,B} > 0.06)$ 6: $(\theta_R > -0.03) \wedge (\theta_{F,F} > -0.49) \wedge (\theta_{T,B} > 0.06)$
	5	0: $(\dot{\theta}_{T,B} <= 1.10) \wedge (\dot{\theta}_{T,F} <= -6.72) \wedge (\theta_{T,F} <= 0.67)$ 1: $(\dot{\theta}_{T,B} <= 1.10) \wedge (\dot{\theta}_{T,F} > -6.72) \wedge (\theta_{T,F} <= 0.67)$ 2: $(\dot{\theta}_{T,B} <= 1.10) \wedge (\theta_{T,F} > 0.67)$ 3: $(\dot{\theta}_{T,B} > 1.10) \wedge (\dot{h}_R <= -0.33) \wedge (\theta_R <= 0.29)$ 4: $(\dot{\theta}_{T,B} > 1.10) \wedge (\dot{h}_R > -0.33) \wedge (\theta_R <= 0.29)$ 5: $(\dot{\theta}_{T,B} > 1.10) \wedge (\dot{h}_R <= -0.48) \wedge (\theta_R > 0.29)$ 6: $(\dot{\theta}_{T,B} > 1.10) \wedge (\dot{h}_R > -0.48) \wedge (\theta_R > 0.29)$
	6	0: $(\dot{\theta}_R <= -0.83) \wedge (\dot{\theta}_{F,B} <= 4.65) \wedge (\dot{\theta}_{F,F} <= -0.07)$ 1: $(\dot{\theta}_R <= -0.83) \wedge (\dot{\theta}_{F,B} > 4.65) \wedge (\dot{\theta}_{F,F} <= -0.07)$ 2: $(\dot{\theta}_R > -0.83) \wedge (\dot{\theta}_{F,F} <= -0.07) \wedge (\theta_{S,B} <= -0.33)$ 3: $(\dot{\theta}_R > -0.83) \wedge (\dot{\theta}_{F,F} <= -0.07) \wedge (\theta_{S,B} > -0.33)$ 4: $(\dot{\theta}_{F,F} > -0.07) \wedge (\theta_R <= 0.52)$ 5: $(\dot{\theta}_{F,F} > -0.07) \wedge (\theta_R > 0.52)$
	7	0: $(\dot{\theta}_{T,F} <= -0.36) \wedge (\theta_R <= 0.33) \wedge (\theta_{S,F} <= 0.56)$ 1: $(\dot{\theta}_{T,F} <= -0.36) \wedge (\theta_R > 0.33) \wedge (\theta_{S,F} <= 0.56)$ 2: $(\dot{\theta}_{T,F} <= -0.36) \wedge (\theta_{S,F} > 0.56)$ 3: $(\dot{\theta}_{T,F} <= 6.91) \wedge (\dot{\theta}_{T,F} > -0.36) \wedge (\theta_{S,B} <= 0.09)$ 4: $(\dot{\theta}_{T,F} > 6.91) \wedge (\theta_{S,B} <= 0.09)$ 5: $(\dot{\theta}_{T,F} > -0.36) \wedge (\theta_{S,B} > 0.09)$
	8	0: $(\dot{\theta}_R <= -0.10) \wedge (\dot{\theta}_{T,B} <= -3.31)$ 1: $(\dot{\theta}_R > -0.10) \wedge (\dot{\theta}_{T,B} <= -3.31)$ 2: $(\dot{\theta}_{T,B} <= 0.97) \wedge (\dot{\theta}_{T,B} > -3.31)$ 3: $(\dot{\theta}_{T,B} > 0.97) \wedge (\theta_{S,B} <= -0.35)$ 4: $(\dot{\theta}_{T,B} > 0.97) \wedge (\theta_R <= 0.20) \wedge (\theta_{S,B} > -0.35)$ 5: $(\dot{\theta}_{T,B} > 0.97) \wedge (\theta_R > 0.20) \wedge (\theta_{S,B} > -0.35)$
	9	0: $(\dot{\theta}_{F,B} <= 0.63) \wedge (\dot{\theta}_{S,F} <= 3.07)$ 1: $(\dot{\theta}_{F,B} <= 0.63) \wedge (\dot{\theta}_{S,F} > 3.07) \wedge (\theta_R <= 0.46)$ 2: $(\dot{\theta}_{F,B} <= 0.63) \wedge (\dot{\theta}_{S,F} > 3.07) \wedge (\theta_R > 0.46)$ 3: $(\dot{\theta}_{F,B} > 0.63) \wedge (\dot{\theta}_{T,F} <= 5.96) \wedge (h_R <= 0.03)$ 4: $(\dot{\theta}_{F,B} > 0.63) \wedge (\dot{\theta}_{T,F} <= 5.96) \wedge (h_R > 0.03)$ 5: $(\dot{\theta}_{F,B} > 0.63) \wedge (\dot{\theta}_{T,F} > 5.96) \wedge (\theta_{T,F} <= 0.38)$ 6: $(\dot{\theta}_{F,B} > 0.63) \wedge (\dot{\theta}_{T,F} > 5.96) \wedge (\theta_{T,F} > 0.38)$

Table 23: Logic program of FC in locomotion (HalfCheetah).

Model	Neuron	Logic Program
GRU	0	0: $(\dot{\theta}_R <= 1.54) \wedge (\dot{\theta}_{T,B} <= -2.33) \wedge (\theta_R <= 0.52)$ 1: $(\dot{\theta}_R > 1.54) \wedge (\dot{\theta}_{T,B} <= -2.33) \wedge (\theta_R <= 0.52)$ 2: $(\dot{\theta}_{T,B} > -2.33) \wedge (\theta_R <= 0.11)$ 3: $(\dot{\theta}_{T,B} > -2.33) \wedge (\theta_R <= 0.52) \wedge (\theta_R > 0.11)$ 4: $(\dot{\theta}_{T,F} <= -7.48) \wedge (\theta_R > 0.52)$ 5: $(\dot{\theta}_{T,F} > -7.48) \wedge (\theta_R <= 0.97) \wedge (\theta_R > 0.52)$ 6: $(\dot{\theta}_{T,F} > -7.48) \wedge (\theta_R > 0.97)$
	1	0: $(\dot{\theta}_{S,B} <= 10.33) \wedge (\theta_R <= 0.50) \wedge (\theta_{F,B} <= -0.41)$ 1: $(\dot{\theta}_{S,B} <= 10.33) \wedge (\theta_R <= 0.50) \wedge (\theta_{F,B} > -0.41)$ 2: $(\dot{\theta}_{S,B} <= 10.33) \wedge (\dot{h}_R <= -0.07) \wedge (\theta_R > 0.50)$ 3: $(\dot{\theta}_{S,B} <= 10.33) \wedge (\dot{h}_R > -0.07) \wedge (\theta_R > 0.50)$ 4: $(\dot{\theta}_{S,B} > 10.33)$
	2	0: $(\dot{\theta}_{T,B} <= 3.33) \wedge (\theta_R <= 0.12)$ 1: $(\dot{\theta}_{T,B} > 3.33) \wedge (\theta_R <= 0.12)$ 2: $(\dot{\theta}_{T,B} <= 6.59) \wedge (\theta_R > 0.12) \wedge (\theta_{T,F} <= 0.70)$ 3: $(\dot{\theta}_{T,B} <= 6.59) \wedge (\theta_R > 0.12) \wedge (\theta_{T,F} > 0.70)$ 4: $(\dot{\theta}_{T,B} > 6.59) \wedge (\theta_R <= 0.54) \wedge (\theta_R > 0.12)$ 5: $(\dot{\theta}_{T,B} > 6.59) \wedge (\theta_R > 0.54)$
	3	0: $(\dot{\theta}_R <= -0.78) \wedge (\theta_{T,F} <= 0.17)$ 1: $(\dot{\theta}_R > -0.78) \wedge (\theta_R <= 0.68) \wedge (\theta_{T,F} <= 0.17)$ 2: $(\dot{\theta}_R > -0.78) \wedge (\theta_R > 0.68) \wedge (\theta_{T,F} <= 0.17)$ 3: $(\dot{h}_R <= 0.64) \wedge (\theta_{T,B} <= -0.14) \wedge (\theta_{T,F} > 0.17)$ 4: $(\dot{h}_R <= 0.64) \wedge (\theta_{T,B} > -0.14) \wedge (\theta_{T,F} > 0.17)$ 5: $(\dot{h}_R > 0.64) \wedge (\theta_{T,F} > 0.17)$
	4	0: $(\dot{\theta}_{S,B} <= 1.92) \wedge (\theta_{S,F} <= 0.02)$ 1: $(\dot{\theta}_{S,B} > 1.92) \wedge (\theta_{S,F} <= 0.02)$ 2: $(\dot{\theta}_{S,B} <= 6.10) \wedge (\dot{\theta}_{S,F} <= 7.21) \wedge (\theta_{S,F} > 0.02)$ 3: $(\dot{\theta}_{S,B} <= 6.10) \wedge (\dot{\theta}_{S,F} > 7.21) \wedge (\theta_{S,F} > 0.02)$ 4: $(\dot{\theta}_{S,B} > 6.10) \wedge (\theta_{S,F} > 0.02)$
	5	0: $(\dot{\theta}_{T,B} <= 2.59) \wedge (\theta_{F,B} <= 0.10) \wedge (\theta_{T,B} <= -0.16)$ 1: $(\dot{\theta}_{T,B} <= 2.59) \wedge (\theta_{F,B} <= 0.10) \wedge (\theta_{T,B} > -0.16)$ 2: $(\dot{\theta}_{T,B} > 2.59) \wedge (\theta_{F,B} <= 0.10)$ 3: $(\dot{\theta}_{T,B} <= 1.45) \wedge (\dot{\theta}_{T,F} <= 6.43) \wedge (\theta_{F,B} > 0.10)$ 4: $(\dot{\theta}_{T,B} > 1.45) \wedge (\dot{\theta}_{T,F} <= 6.43) \wedge (\theta_{F,B} > 0.10)$ 5: (conflict) $\wedge (\theta_{F,B} > 0.10)$ 6: $(\dot{\theta}_{T,F} > 6.43) \wedge (\theta_{F,B} > 0.10)$
	6	0: $(\theta_R <= 0.17) \wedge (\theta_{F,B} <= -0.12)$ 1: $(\theta_R <= 0.62) \wedge (\theta_R > 0.17) \wedge (\theta_{F,B} <= -0.12)$ 2: $(\dot{\theta}_{F,B} <= 6.66) \wedge (\theta_R <= 0.62) \wedge (\theta_{F,B} > -0.12)$ 3: $(\dot{\theta}_{F,B} > 6.66) \wedge (\theta_R <= 0.62) \wedge (\theta_{F,B} > -0.12)$ 4: $(\dot{\theta}_{T,B} <= -5.47) \wedge (\theta_R > 0.62)$ 5: $(\dot{\theta}_{T,B} > -5.47) \wedge (\theta_R <= 0.89) \wedge (\theta_R > 0.62)$ 6: $(\dot{\theta}_{T,B} > -5.47) \wedge (\theta_R > 0.89)$
	7	0: $(\dot{h}_R <= -0.27) \wedge (\theta_{T,F} <= 0.20)$ 1: $(\dot{\theta}_{T,B} <= -0.25) \wedge (\dot{h}_R > -0.27) \wedge (\theta_{T,F} <= 0.20)$ 2: $(\dot{\theta}_{T,B} > -0.25) \wedge (\dot{h}_R > -0.27) \wedge (\theta_{T,F} <= 0.20)$ 3: $(\dot{h}_R <= 0.12) \wedge (\theta_R <= 0.77) \wedge (\theta_{T,F} > 0.20)$ 4: $(\dot{h}_R <= 0.12) \wedge (\theta_R > 0.77) \wedge (\theta_{T,F} > 0.20)$ 5: $(\dot{h}_R > 0.12) \wedge (\theta_{T,F} > 0.20)$
	8	0: $(\dot{h}_R <= 0.13) \wedge (\theta_{F,B} <= -0.16) \wedge (h_R <= 0.07)$ 1: $(\dot{h}_R <= 0.13) \wedge (\theta_{F,B} > -0.16) \wedge (h_R <= 0.07)$ 2: $(\dot{h}_R <= 0.13) \wedge (h_R > 0.07)$ 3: $(\dot{h}_R <= 1.11) \wedge (\dot{h}_R > 0.13) \wedge (h_R <= 0.08)$ 4: $(\dot{h}_R > 1.11) \wedge (h_R <= 0.08)$ 5: $(\dot{h}_R > 0.13) \wedge (h_R > 0.08)$
	9	0: $(\dot{\theta}_{T,F} <= 0.51) \wedge (\theta_{S,F} <= -0.07)$ 1: $(\dot{\theta}_{F,F} <= 3.81) \wedge (\dot{\theta}_{T,F} <= 0.51) \wedge (\theta_{S,F} > -0.07)$ 2: $(\dot{\theta}_{F,F} > 3.81) \wedge (\dot{\theta}_{T,F} <= 0.51) \wedge (\theta_{S,F} > -0.07)$ 3: $(\dot{\theta}_{T,F} > 0.51) \wedge (\theta_{F,B} <= 0.34)$ 4: $(\dot{\theta}_R <= -1.78) \wedge (\dot{\theta}_{T,F} > 0.51) \wedge (\theta_{F,B} > 0.34)$ 5: $(\dot{\theta}_R > -1.78) \wedge (\dot{\theta}_{T,F} > 0.51) \wedge (\theta_{F,B} > 0.34)$

Table 24: Logic program of GRU in locomotion (HalfCheetah).

Model	Neuron	Logic Program
LSTM	0	0: $(\dot{\theta}_R <= 2.01) \wedge (\theta_{S,B} <= 0.53) \wedge (\theta_{T,F} <= -0.28)$ 1: $(\dot{\theta}_R <= 2.01) \wedge (\theta_{S,B} <= 0.53) \wedge (\theta_{T,F} > -0.28)$ 2: $(\dot{\theta}_R > 2.01) \wedge (\theta_{S,B} <= 0.53)$ 3: $(\theta_{S,B} > 0.53)$
	1	0: $(\theta_{F,F} <= 0.02) \wedge (\theta_{T,F} <= -0.35)$ 1: $(\dot{\theta}_{F,F} <= 1.23) \wedge (\theta_{F,F} <= 0.02) \wedge (\theta_{T,F} > -0.35)$ 2: $(\dot{\theta}_{F,F} > 1.23) \wedge (\theta_{F,F} <= 0.02) \wedge (\theta_{T,F} > -0.35)$ 3: $(\theta_{F,F} > 0.02) \wedge (\theta_{T,F} <= -0.07)$ 4: $(\dot{\theta}_{F,F} <= 0.95) \wedge (\theta_{F,F} > 0.02) \wedge (\theta_{T,F} > -0.07)$ 5: $(\dot{\theta}_{F,F} > 0.95) \wedge (\theta_{F,F} > 0.02) \wedge (\theta_{T,F} > -0.07)$
	2	0: $(\theta_{F,F} <= -0.46) \wedge (h_R <= -0.07)$ 1: $(\dot{\theta}_{F,B} <= -4.76) \wedge (\theta_{F,F} > -0.46) \wedge (h_R <= -0.07)$ 2: $(\dot{\theta}_{F,B} > -4.76) \wedge (\theta_{F,F} > -0.46) \wedge (h_R <= -0.07)$ 3: $(\theta_R <= 0.11) \wedge (h_R > -0.07)$ 4: $(\theta_R > 0.11) \wedge (h_R > -0.07)$
	3	0: $(\dot{\theta}_{S,B} <= 0.14) \wedge (\theta_{T,F} <= -0.01) \wedge (h_R <= -0.11)$ 1: $(\dot{\theta}_{S,B} <= 0.14) \wedge (\theta_{T,F} > -0.01) \wedge (h_R <= -0.11)$ 2: $(\dot{\theta}_{S,B} > 0.14) \wedge (\theta_{S,B} <= 0.42) \wedge (h_R <= -0.11)$ 3: $(\dot{\theta}_{S,B} > 0.14) \wedge (\theta_{S,B} > 0.42) \wedge (h_R <= -0.11)$ 4: $(\theta_{T,B} <= 0.27) \wedge (h_R > -0.11)$ 5: $(\theta_{T,B} <= 0.53) \wedge (\theta_{T,B} > 0.27) \wedge (h_R > -0.11)$ 6: $(\theta_{T,B} > 0.53) \wedge (h_R > -0.11)$
	4	0: $(\theta_{F,F} <= -0.00)$ 1: (conflict) 2: $(\theta_R <= 0.12) \wedge (\theta_{F,F} <= 0.39) \wedge (\theta_{F,F} > -0.00)$ 3: $(\theta_R > 0.12) \wedge (\theta_{F,F} <= 0.39) \wedge (\theta_{F,F} > -0.00)$ 4: $(\dot{\theta}_R <= -0.51) \wedge (\theta_{F,F} > 0.39)$ 5: $(\dot{\theta}_R > -0.51) \wedge (\theta_{F,F} > 0.39)$
	5	0: $(\dot{\theta}_R <= 0.07) \wedge (\theta_{F,F} <= -0.05)$ 1: $(\dot{\theta}_R <= 0.07) \wedge (\theta_{F,F} <= 0.35) \wedge (\theta_{F,F} > -0.05)$ 2: $(\dot{\theta}_R > 0.07) \wedge (\theta_{F,F} <= 0.35) \wedge (h_R <= -0.18)$ 3: $(\dot{\theta}_R > 0.07) \wedge (\theta_{F,F} <= 0.35) \wedge (h_R > -0.18)$ 4: $(\dot{\theta}_{S,B} <= -2.15) \wedge (\theta_{F,F} > 0.35)$ 5: $(\dot{\theta}_{S,B} > -2.15) \wedge (\theta_{F,F} > 0.35)$
	6	0: $(\dot{\theta}_{T,F} <= 1.81) \wedge (\dot{h}_R <= -1.08) \wedge (\theta_{T,F} <= -0.14)$ 1: $(\dot{\theta}_{T,F} <= 1.81) \wedge (\dot{h}_R > -1.08) \wedge (\theta_{T,F} <= -0.14)$ 2: $(\dot{\theta}_{T,F} > 1.81) \wedge (\theta_{T,F} <= -0.14)$ 3: $(\dot{h}_R <= -0.47) \wedge (\theta_{T,F} > -0.14)$ 4: $(\dot{\theta}_{T,B} <= -1.89) \wedge (\dot{h}_R > -0.47) \wedge (\theta_{T,F} > -0.14)$ 5: $(\dot{\theta}_{T,B} > -1.89) \wedge (\dot{h}_R > -0.47) \wedge (\theta_{T,F} > -0.14)$
	7	0: $(\theta_{F,F} <= -0.07) \wedge (\theta_{T,F} <= 0.36)$ 1: $(\theta_{F,F} <= -0.07) \wedge (\theta_{T,F} > 0.36)$ 2: $(\dot{x}_R <= 4.14) \wedge (\theta_{F,F} > -0.07) \wedge (\theta_{T,B} <= 0.34)$ 3: $(\dot{x}_R > 4.14) \wedge (\theta_{F,F} > -0.07) \wedge (\theta_{T,B} <= 0.34)$ 4: $(\dot{\theta}_{T,F} <= -2.37) \wedge (\theta_{F,F} > -0.07) \wedge (\theta_{T,B} > 0.34)$ 5: $(\dot{\theta}_{T,F} > -2.37) \wedge (\theta_{F,F} > -0.07) \wedge (\theta_{T,B} > 0.34)$
	8	0: $(\dot{\theta}_{F,B} <= 11.09) \wedge (\dot{\theta}_{F,F} <= 1.05) \wedge (\theta_{T,F} <= -0.46)$ 1: $(\dot{\theta}_{F,B} <= 11.09) \wedge (\dot{\theta}_{F,F} <= 1.05) \wedge (\theta_{T,F} > -0.46)$ 2: $(\dot{\theta}_{F,B} > 11.09) \wedge (\dot{\theta}_{F,F} <= 1.05)$ 3: $(\dot{\theta}_{F,F} > 1.05) \wedge (\dot{\theta}_{T,B} <= -11.38)$ 4: $(\dot{\theta}_{F,F} > 1.05) \wedge (\dot{\theta}_{T,B} > -11.38) \wedge (\theta_R <= 0.16)$ 5: $(\dot{\theta}_{F,F} > 1.05) \wedge (\dot{\theta}_{T,B} > -11.38) \wedge (\theta_R > 0.16)$
	9	0: $(\theta_{F,F} <= -0.40) \wedge (\theta_{S,B} <= -0.33)$ 1: $(\theta_{F,F} <= -0.40) \wedge (\theta_{S,B} > -0.33)$ 2: $(\dot{\theta}_{T,F} <= -10.21) \wedge (\dot{h}_R <= -0.80) \wedge (\theta_{F,F} > -0.40)$ 3: $(\dot{\theta}_{T,F} > -10.21) \wedge (\dot{h}_R <= -0.80) \wedge (\theta_{F,F} > -0.40)$ 4: $(\dot{h}_R > -0.80) \wedge (\theta_{F,B} <= -0.14) \wedge (\theta_{F,F} > -0.40)$ 5: $(\dot{h}_R > -0.80) \wedge (\theta_{F,B} > -0.14) \wedge (\theta_{F,F} > -0.40)$

Table 25: Logic program of LSTM in locomotion (HalfCheetah).

Model	Neuron	Logic Program
ODE-RNN	0	0: $(\dot{h}_R <= -0.27) \wedge (\theta_{S,B} <= -0.49)$ 1: $(\dot{h}_R > -0.27) \wedge (\theta_{S,B} <= -0.49)$ 2: $(\dot{h}_R <= -0.00) \wedge (\theta_{F,F} <= -0.22) \wedge (\theta_{S,B} > -0.49)$ 3: $(\dot{h}_R <= -0.00) \wedge (\theta_{F,F} > -0.22) \wedge (\theta_{S,B} > -0.49)$ 4: $(\dot{h}_R > -0.00) \wedge (\theta_{S,B} > -0.49) \wedge (\theta_{T,B} <= 0.22)$ 5: $(\dot{h}_R > -0.00) \wedge (\theta_{S,B} > -0.49) \wedge (\theta_{T,B} > 0.22)$
	1	0: $(\dot{\theta}_{T,B} <= -2.91) \wedge (\theta_{F,B} <= -0.39)$ 1: $(\dot{\theta}_{T,B} > -2.91) \wedge (\dot{h}_R <= -0.21) \wedge (\theta_{F,B} <= -0.39)$ 2: $(\dot{\theta}_{T,B} > -2.91) \wedge (\dot{h}_R > -0.21) \wedge (\theta_{F,B} <= -0.39)$ 3: $(\dot{\theta}_R <= -0.00) \wedge (\theta_{F,B} > -0.39)$ 4: $(\dot{\theta}_R > -0.00) \wedge (\theta_R <= -0.04) \wedge (\theta_{F,B} > -0.39)$ 5: $(\dot{\theta}_R > -0.00) \wedge (\theta_R > -0.04) \wedge (\theta_{F,B} > -0.39)$
	2	0: $(\theta_R <= 0.02) \wedge (\theta_{T,B} <= -0.16)$ 1: $(\theta_R <= 0.02) \wedge (\theta_{T,B} > -0.16) \wedge (h_R <= 0.00)$ 2: $(\theta_R <= 0.02) \wedge (\theta_{T,B} > -0.16) \wedge (h_R > 0.00)$ 3: $(\theta_R > 0.02) \wedge (h_R <= -0.02)$ 4: $(\theta_R > 0.02) \wedge$ (conflict) 5: $(\theta_R > 0.02) \wedge (h_R > -0.02)$
	3	0: $(\dot{h}_R <= 0.55) \wedge (\theta_R <= 0.08) \wedge (\theta_{T,B} <= 0.42)$ 1: $(\dot{h}_R > 0.55) \wedge (\theta_R <= 0.08) \wedge (\theta_{T,B} <= 0.42)$ 2: $(\theta_R <= 0.08) \wedge (\theta_{S,B} <= 0.16) \wedge (\theta_{T,B} > 0.42)$ 3: $(\theta_R <= 0.08) \wedge (\theta_{S,B} > 0.16) \wedge (\theta_{T,B} > 0.42)$ 4: $(\theta_R > 0.08) \wedge (\theta_{T,B} <= 0.42) \wedge (h_R <= -0.05)$ 5: $(\theta_R > 0.08) \wedge (\theta_{T,B} <= 0.42) \wedge (h_R > -0.05)$ 6: $(\theta_R > 0.08) \wedge (\theta_{T,B} > 0.42)$
	4	0: $(\dot{\theta}_{T,B} <= -3.83) \wedge (\dot{\theta}_{T,F} <= -0.87)$ 1: $(\dot{\theta}_{T,B} <= -3.83) \wedge (\dot{\theta}_{T,F} > -0.87) \wedge (h_R <= -0.08)$ 2: $(\dot{\theta}_{T,B} <= -3.83) \wedge (\dot{\theta}_{T,F} > -0.87) \wedge (h_R > -0.08)$ 3: $(\dot{\theta}_{T,B} > -3.83) \wedge (\theta_{S,B} <= -0.11) \wedge (h_R <= -0.09)$ 4: $(\dot{\theta}_{T,B} > -3.83) \wedge (\theta_{S,B} > -0.11) \wedge (h_R <= -0.09)$ 5: $(\dot{\theta}_{T,B} > -3.83) \wedge (\theta_{T,B} <= -0.33) \wedge (h_R > -0.09)$ 6: $(\dot{\theta}_{T,B} > -3.83) \wedge (\theta_{T,B} > -0.33) \wedge (h_R > -0.09)$
	5	0: $(\dot{x}_R <= 3.31)$ 1: $(\dot{x}_R > 3.31) \wedge (\theta_{S,B} <= 0.01) \wedge (\theta_{T,F} <= 0.57)$ 2: $(\dot{x}_R > 3.31) \wedge (\theta_{S,B} > 0.01) \wedge (\theta_{T,F} <= 0.57)$ 3: $(\dot{x}_R > 3.31) \wedge (\theta_{T,F} > 0.57)$
	6	0: $(\dot{\theta}_{T,B} <= 1.22) \wedge (\theta_{S,B} <= -0.54)$ 1: $(\dot{\theta}_{T,B} > 1.22) \wedge (\theta_{S,B} <= -0.54)$ 2: $(\dot{\theta}_{T,B} <= -3.70) \wedge (\theta_{S,B} > -0.54) \wedge (\theta_{T,B} <= 0.16)$ 3: $(\dot{\theta}_{T,B} <= -3.70) \wedge (\theta_{S,B} > -0.54) \wedge (\theta_{T,B} > 0.16)$ 4: $(\dot{\theta}_{T,B} > -3.70) \wedge$ (conflict) 5: $(\dot{\theta}_{T,B} > -3.70) \wedge (\theta_{S,B} > -0.54)$
	7	0: $(\dot{h}_R <= 0.02) \wedge (\theta_{S,F} <= -0.19) \wedge (\theta_{T,B} <= 0.53)$ 1: $(\dot{h}_R <= 0.02) \wedge (\theta_{S,F} <= -0.19) \wedge (\theta_{T,B} > 0.53)$ 2: $(\dot{h}_R > 0.02) \wedge (\theta_{S,F} <= -0.19)$ 3: $(\dot{\theta}_R <= 0.53) \wedge (\theta_{S,F} <= 0.16) \wedge (\theta_{S,F} > -0.19)$ 4: $(\dot{\theta}_R <= 0.53) \wedge (\theta_{S,F} > 0.16)$ 5: $(\dot{\theta}_R > 0.53) \wedge (\theta_{S,F} > -0.19)$
	8	0: $(\dot{\theta}_{T,B} <= -0.39) \wedge (\theta_{T,B} <= -0.37)$ 1: $(\dot{\theta}_{T,B} <= -0.39) \wedge (\theta_{S,B} <= 0.04) \wedge (\theta_{T,B} > -0.37)$ 2: $(\dot{\theta}_{T,B} <= -0.39) \wedge (\theta_{S,B} > 0.04) \wedge (\theta_{T,B} > -0.37)$ 3: $(\dot{\theta}_{T,B} > -0.39) \wedge (\theta_{T,B} <= 0.55) \wedge (\theta_{T,F} <= 0.09)$ 4: $(\dot{\theta}_{T,B} > -0.39) \wedge (\theta_{T,B} <= 0.55) \wedge (\theta_{T,F} > 0.09)$ 5: $(\dot{\theta}_{T,B} > -0.39) \wedge (\theta_{T,B} > 0.55)$
	9	0: $(\theta_{T,F} <= -0.24)$ 1: $(\dot{\theta}_{T,F} <= -6.60) \wedge$ (conflict) 2: $(\dot{\theta}_{T,F} > -6.60) \wedge$ (conflict) 3: $(\theta_{T,F} > -0.24) \wedge (h_R <= -0.11)$ 4: $(\theta_{T,F} <= 0.32) \wedge (\theta_{T,F} > -0.24) \wedge (h_R > -0.11)$ 5: $(\theta_{T,F} > 0.32) \wedge (h_R > -0.11)$

Table 26: Logic program of ODE-RNN in locomotion (HalfCheetah).

Model	Neuron	Logic Program
CfC	0	0: $(\dot{\theta}_{F,F} <= -9.40)$ 1: $(\dot{\theta}_{F,F} > -9.40) \wedge (\dot{\theta}_{T,B} <= -1.68) \wedge (\theta_{F,F} <= 0.23)$ 2: $(\dot{\theta}_{F,F} > -9.40) \wedge (\dot{\theta}_{T,B} > -1.68) \wedge (\theta_{F,F} <= 0.23)$ 3: $(\dot{\theta}_{F,F} > -9.40) \wedge (\dot{\theta}_{T,F} <= -5.69) \wedge (\theta_{F,F} > 0.23)$ 4: $(\dot{\theta}_{F,F} > -9.40) \wedge (\dot{\theta}_{T,F} > -5.69) \wedge (\theta_{F,F} > 0.23)$
	1	0: $(\theta_R <= 0.06) \wedge (\theta_{T,F} <= -0.46)$ 1: $(\theta_R > 0.06) \wedge (\theta_{T,F} <= -0.46)$ 2: $(\theta_R <= 0.02) \wedge (\theta_{T,F} > -0.46)$ 3: $(\theta_R > 0.02) \wedge$ (conflict) 4: $(\theta_R > 0.02) \wedge (\theta_{T,F} > -0.46)$
	2	0: $(\dot{\theta}_{T,B} <= 1.92) \wedge (\dot{\theta}_{T,F} <= -0.59) \wedge (\dot{h}_R <= 0.43)$ 1: $(\dot{\theta}_{T,B} <= 1.92) \wedge (\dot{\theta}_{T,F} > -0.59) \wedge (\dot{h}_R <= 0.43)$ 2: $(\dot{\theta}_{T,B} > 1.92) \wedge (\dot{\theta}_{T,F} <= 5.02) \wedge (\dot{h}_R <= 0.43)$ 3: $(\dot{\theta}_{T,B} > 1.92) \wedge (\dot{\theta}_{T,F} > 5.02) \wedge (\dot{h}_R <= 0.43)$ 4: $(\dot{\theta}_{S,F} <= 13.53) \wedge (\dot{h}_R > 0.43) \wedge (\theta_{F,B} <= -0.01)$ 5: $(\dot{\theta}_{S,F} <= 13.53) \wedge (\dot{h}_R > 0.43) \wedge (\theta_{F,B} > -0.01)$ 6: $(\dot{\theta}_{S,F} > 13.53) \wedge (\dot{h}_R > 0.43)$
	3	0: $(\dot{\theta}_{T,F} <= 8.70) \wedge (\theta_{S,F} <= 0.53) \wedge (h_R <= -0.09)$ 1: $(\dot{\theta}_{T,F} <= 8.70) \wedge (\theta_{S,F} <= 0.53) \wedge (h_R > -0.09)$ 2: $(\dot{\theta}_{T,F} <= 8.70) \wedge (\theta_{S,F} > 0.53)$ 3: $(\dot{\theta}_{T,F} > 8.70) \wedge (\theta_R <= 0.20)$ 4: $(\dot{\theta}_{T,F} > 8.70) \wedge (\theta_R > 0.20)$
	4	0: $(\theta_{T,F} <= -0.51) \wedge (h_R <= -0.04)$ 1: $(\theta_{T,F} <= -0.51) \wedge (h_R > -0.04)$ 2: $(\dot{\theta}_{T,B} <= 0.58) \wedge (\theta_{T,F} <= 0.50) \wedge (\theta_{T,F} > -0.51)$ 3: $(\dot{\theta}_{T,B} <= 0.58) \wedge (\theta_{T,F} > 0.50)$ 4: $(\dot{\theta}_{T,B} > 0.58) \wedge (\dot{h}_R <= -0.44) \wedge (\theta_{T,F} > -0.51)$ 5: $(\dot{\theta}_{T,B} > 0.58) \wedge (\dot{h}_R > -0.44) \wedge (\theta_{T,F} > -0.51)$
	5	0: $(\dot{\theta}_{T,B} <= 2.61) \wedge (\theta_{S,F} <= 0.08) \wedge (h_R <= 0.03)$ 1: $(\dot{\theta}_{T,B} <= 2.61) \wedge (\theta_{S,F} <= 0.08) \wedge (h_R > 0.03)$ 2: $(\dot{\theta}_R <= 0.63) \wedge (\dot{\theta}_{T,B} <= 2.61) \wedge (\theta_{S,F} > 0.08)$ 3: $(\dot{\theta}_R > 0.63) \wedge (\dot{\theta}_{T,B} <= 2.61) \wedge (\theta_{S,F} > 0.08)$ 4: $(\dot{\theta}_{T,B} > 2.61) \wedge (\theta_{T,F} <= -0.13)$ 5: $(\dot{\theta}_{T,B} > 2.61) \wedge (\theta_{T,F} > -0.13)$
	6	0: $(\dot{\theta}_{T,B} <= -0.60) \wedge (\theta_{F,F} <= -0.11)$ 1: $(\dot{\theta}_{T,B} > -0.60) \wedge (\theta_{F,F} <= -0.11) \wedge (h_R <= -0.09)$ 2: $(\dot{\theta}_{T,B} > -0.60) \wedge (\theta_{F,F} <= -0.11) \wedge (h_R > -0.09)$ 3: $(\dot{\theta}_{S,F} <= 11.48) \wedge (\theta_{F,F} > -0.11) \wedge (\theta_{T,B} <= -0.10)$ 4: $(\dot{\theta}_{S,F} <= 11.48) \wedge (\theta_{F,F} > -0.11) \wedge (\theta_{T,B} > -0.10)$ 5: $(\dot{\theta}_{S,F} > 11.48) \wedge (\theta_{F,F} > -0.11)$
	7	0: $(\dot{\theta}_{T,F} <= -8.38) \wedge (\dot{h}_R <= 0.45) \wedge (\theta_{T,F} <= 0.47)$ 1: $(\dot{\theta}_{T,F} > -8.38) \wedge (\dot{h}_R <= 0.45) \wedge (\theta_{T,F} <= 0.47)$ 2: $(\dot{h}_R > 0.45) \wedge (\theta_{F,F} <= 0.14) \wedge (\theta_{T,F} <= 0.47)$ 3: $(\dot{h}_R > 0.45) \wedge (\theta_{F,F} > 0.14) \wedge (\theta_{T,F} <= 0.47)$ 4: $(\theta_{T,F} <= 0.69) \wedge (\theta_{T,F} > 0.47)$ 5: $(\theta_{T,F} > 0.69)$
	8	0: $(\dot{\theta}_{T,B} <= -6.34) \wedge (h_R <= -0.02)$ 1: $(\dot{\theta}_{T,B} > -6.34) \wedge (h_R <= -0.02)$ 2: $(\dot{\theta}_{T,B} > -6.34) \wedge$ (conflict) 3: $(\dot{\theta}_{T,F} <= 10.03) \wedge (\theta_{S,F} <= 0.50) \wedge (h_R > -0.02)$ 4: $(\dot{\theta}_{T,F} > 10.03) \wedge (\theta_{S,F} <= 0.50) \wedge (h_R > -0.02)$ 5: $(\theta_{S,F} > 0.50) \wedge (h_R > -0.02)$
	9	0: $(\dot{\theta}_{T,B} <= 1.14) \wedge (\theta_{S,B} <= -0.06) \wedge (\theta_{S,F} <= 0.46)$ 1: $(\dot{\theta}_{T,B} <= 1.14) \wedge (\theta_{S,B} > -0.06) \wedge (\theta_{S,F} <= 0.46)$ 2: $(\dot{\theta}_{T,B} <= 1.14) \wedge (\theta_{S,F} > 0.46)$ 3: $(\dot{\theta}_{T,B} > 1.14) \wedge (\theta_{T,F} <= 0.56) \wedge (h_R <= -0.10)$ 4: $(\dot{\theta}_{T,B} > 1.14) \wedge (\theta_{T,F} <= 0.56) \wedge (h_R > -0.10)$ 5: $(\dot{\theta}_{T,B} > 1.14) \wedge (\theta_{T,F} > 0.56)$

Table 27: Logic program of CfC in locomotion (HalfCheetah).

Model	Neuron	Logic Program
NCP	0	0: $(\theta_{F,B} <= -0.05)$ 1: $(\theta_{F,B} > -0.05)$
	1	0: $(\theta_{F,B} <= -0.04)$ 1: $(\theta_{F,B} > -0.04)$
	2	0: $(\theta_{T,B} <= -0.29)$ 1: $(\theta_{T,B} > -0.29)$
	3	0: $(\dot{\theta}_{T,F} <= -6.91) \wedge (h_R <= -0.08)$ 1: $(\dot{\theta}_{T,F} <= -6.91) \wedge (h_R > -0.08)$ 2: $(\dot{\theta}_{T,F} > -6.91) \wedge (\theta_{F,F} <= -0.13)$ 3: $(\dot{\theta}_{T,F} > -6.91) \wedge (\theta_{F,F} > -0.13)$
	4	0: $(\dot{\theta}_{F,B} <= -6.37) \wedge (\theta_{T,B} <= -0.36)$ 1: $(\dot{\theta}_{F,B} > -6.37) \wedge (\theta_{T,B} <= -0.36)$ 2: $(\dot{h}_R <= -0.59) \wedge (\theta_{T,B} <= 0.59) \wedge (\theta_{T,B} > -0.36)$ 3: $(\dot{h}_R > -0.59) \wedge (\theta_{T,B} <= 0.59) \wedge (\theta_{T,B} > -0.36)$ 4: $(\dot{\theta}_{F,B} <= 0.64) \wedge (\theta_{T,B} > 0.59)$ 5: $(\dot{\theta}_{F,B} > 0.64) \wedge (\theta_{T,B} > 0.59)$
	5	0: $(\theta_{F,B} <= -0.02) \wedge (\theta_{T,B} <= -0.40)$ 1: $(\theta_{F,B} <= -0.02) \wedge (\theta_{T,B} > -0.40)$ 2: $(\theta_{F,B} > -0.02)$
	6	0: $(\dot{\theta}_{F,F} <= -3.61) \wedge (\theta_{F,B} <= -0.06)$ 1: $(\dot{\theta}_{F,F} > -3.61) \wedge (\theta_{F,B} <= -0.06)$ 2: $(\dot{h}_R <= 0.51) \wedge (\theta_{F,B} > -0.06) \wedge (\theta_{T,F} <= -0.70)$ 3: $(\dot{h}_R <= 0.51) \wedge (\theta_{F,B} > -0.06) \wedge (\theta_{T,F} > -0.70)$ 4: $(\dot{h}_R > 0.51) \wedge (\theta_{F,B} > -0.06)$
	7	0: $(\theta_{T,B} <= -0.27)$ 1: (conflict) 2: $(\dot{\theta}_{S,F} <= -8.45) \wedge (\dot{h}_R <= 0.18) \wedge (\theta_{T,B} > -0.27)$ 3: $(\dot{\theta}_{S,F} <= -8.45) \wedge (\dot{h}_R > 0.18) \wedge (\theta_{T,B} > -0.27)$ 4: $(\dot{\theta}_{S,F} > -8.45) \wedge (\theta_{F,B} <= 0.22) \wedge (\theta_{T,B} > -0.27)$ 5: $(\dot{\theta}_{S,F} > -8.45) \wedge (\theta_{F,B} > 0.22) \wedge (\theta_{T,B} > -0.27)$
	8	0: $(\dot{\theta}_{F,B} <= 6.33) \wedge (\dot{\theta}_{T,F} <= 12.79) \wedge (\theta_{S,B} <= -0.04)$ 1: $(\dot{\theta}_{F,B} > 6.33) \wedge (\dot{\theta}_{T,F} <= 12.79) \wedge (\theta_{S,B} <= -0.04)$ 2: $(\dot{\theta}_{T,F} <= 12.79) \wedge (\theta_{S,B} > -0.04)$ 3: $(\dot{\theta}_{T,F} > 12.79)$
	9	0: $(\dot{\theta}_{F,B} <= 6.97) \wedge (h_R <= -0.12)$ 1: $(\dot{\theta}_{F,B} <= 6.97) \wedge (\theta_{F,F} <= 0.15) \wedge (h_R > -0.12)$ 2: $(\dot{\theta}_{F,B} <= 6.97) \wedge (\theta_{F,F} > 0.15) \wedge (h_R > -0.12)$ 3: $(\dot{\theta}_{F,B} > 6.97) \wedge (\dot{\theta}_{S,B} <= -8.53)$ 4: $(\dot{\theta}_{F,B} > 6.97) \wedge (\dot{\theta}_{S,B} > -8.53)$

Table 28: Logic program of NCP in locomotion (HalfCheetah).

Model	Neuron	Logic Program
FC	0	0: $(\kappa <= 0.00) \wedge (v <= 7.40)$ 1: $(\kappa <= 0.00) \wedge (v <= 7.71) \wedge (v > 7.40)$ 2: $(\kappa > 0.00) \wedge (v <= 7.71)$ 3: $(\kappa <= 0.00) \wedge (d <= 0.12) \wedge (v > 7.71)$ 4: $(\kappa <= 0.00) \wedge (d > 0.12) \wedge (v > 7.71)$ 5: $(\kappa > 0.00) \wedge (v > 7.71)$
	1	0: $(v <= 7.30)$ 1: $(\delta <= 0.00) \wedge (d <= 0.19) \wedge (v > 7.30)$ 2: $(\delta <= 0.00) \wedge (d > 0.19) \wedge (v > 7.30)$ 3: $(\delta > 0.00) \wedge (d <= 0.29) \wedge (v > 7.30)$ 4: $(\delta > 0.00) \wedge (d > 0.29) \wedge (v > 7.30)$
	2	0: $(\delta <= -0.02) \wedge (\kappa <= 0.02) \wedge (\mu <= 0.01)$ 1: $(\delta > -0.02) \wedge (\kappa <= 0.02) \wedge (\mu <= 0.01)$ 2: $(\kappa > 0.02) \wedge (\mu <= 0.01)$ 3: $(\mu > 0.01)$
	3	0: $(\kappa <= -0.00)$ 1: $(\kappa > -0.00) \wedge (\mu <= 0.01) \wedge (d <= 0.07)$ 2: $(\kappa > -0.00) \wedge (\mu <= 0.01) \wedge (d > 0.07)$ 3: $(\kappa > -0.00) \wedge (\mu <= 0.02) \wedge (\mu > 0.01)$ 4: $(\kappa > -0.00) \wedge (\mu > 0.02)$
	4	0: $(\kappa <= 0.00) \wedge (\mu <= 0.01) \wedge (v <= 7.66)$ 1: $(\kappa <= 0.00) \wedge (\mu <= 0.01) \wedge (v > 7.66)$ 2: $(\kappa <= 0.00) \wedge (\mu > 0.01)$ 3: $(\kappa > 0.00) \wedge (\mu <= -0.02)$ 4: $(\kappa > 0.00) \wedge (\mu <= 0.01) \wedge (\mu > -0.02)$ 5: $(\kappa > 0.00) \wedge (\mu > 0.01)$
	5	0: $(\mu <= 0.01) \wedge (v <= 7.49)$ 1: $(\delta <= -0.02) \wedge (\mu <= 0.01) \wedge (v > 7.49)$ 2: $(\delta > -0.02) \wedge (\mu <= 0.01) \wedge (v > 7.49)$ 3: $(\mu <= 0.02) \wedge (\mu > 0.01)$ 4: $(\mu > 0.02)$
	6	0: $(\kappa <= -0.01)$ 1: $(\delta <= -0.01) \wedge (\kappa > -0.01) \wedge (\mu <= 0.01)$ 2: $(\delta > -0.01) \wedge (\kappa > -0.01) \wedge (\mu <= 0.01)$ 3: $(\kappa > -0.01) \wedge (\mu > 0.01)$
	7	0: $(\mu <= -0.02)$ 1: $(\delta <= 0.02) \wedge (\mu <= 0.01) \wedge (\mu > -0.02)$ 2: $(\delta > 0.02) \wedge (\mu <= 0.01) \wedge (\mu > -0.02)$ 3: $(\mu > 0.01)$

Table 29: Logic program of FC in end-to-end visual servoing (Image-based Driving).

Model	Neuron	Logic Program
GRU	0	0: $(d <= -0.06)$ 1: $(d <= 0.11) \wedge (d > -0.06)$ 2: $(d > 0.11)$
	1	0: $(\mu <= 0.04)$ 1: $(\mu <= 0.09) \wedge (\mu > 0.04)$ 2: $(\mu > 0.09)$
	2	0: $(\mu <= 0.04)$ 1: $(\mu <= 0.10) \wedge (\mu > 0.04)$ 2: $(\mu > 0.10)$
	3	0: $(v <= 5.26)$ 1: $(\mu <= 0.01) \wedge (v <= 6.88) \wedge (v > 5.26)$ 2: $(\mu > 0.01) \wedge (v <= 6.88) \wedge (v > 5.26)$ 3: $(v <= 7.34) \wedge (v > 6.88)$ 4: $(v > 7.34)$
	4	0: None
	5	0: $(v <= 5.26)$ 1: $(\mu <= -0.01) \wedge (d <= 0.20) \wedge (v > 5.26)$ 2: $(\mu <= -0.01) \wedge (d > 0.20) \wedge (v > 5.26)$ 3: $(\mu > -0.01) \wedge (v <= 6.81) \wedge (v > 5.26)$ 4: $(\mu > -0.01) \wedge (v > 6.81)$
	6	0: $(\delta <= -0.04)$ 1: $(\delta <= 0.05) \wedge (\delta > -0.04) \wedge (v <= 7.81)$ 2: $(\delta <= 0.05) \wedge (\delta > -0.04) \wedge (v > 7.81)$ 3: $(\delta <= 0.09) \wedge (\delta > 0.05)$ 4: $(\delta > 0.09)$
	7	0: $(\kappa <= 0.02) \wedge (\mu <= -0.05) \wedge (d <= 0.61)$ 1: $(\kappa <= 0.02) \wedge (\mu > -0.05) \wedge (d <= 0.61)$ 2: $(\kappa > 0.02) \wedge (d <= 0.61)$ 3: $(\delta <= 0.06) \wedge (\mu <= -0.04) \wedge (d > 0.61)$ 4: $(\delta <= 0.06) \wedge (\mu > -0.04) \wedge (d > 0.61)$ 5: $(\delta > 0.06) \wedge (d > 0.61)$

Table 30: Logic program of GRU in end-to-end visual servoing (Image-based Driving).

Model	Neuron	Logic Program
LSTM	0	0: $(\kappa <= 0.00) \wedge (d <= 0.07)$ 1: $(\kappa <= 0.00) \wedge (d > 0.07)$ 2: $(\kappa <= 0.00) \wedge (\kappa > 0.00)$ 3: $(\kappa > 0.00) \wedge (v <= 7.58)$ 4: $(\kappa > 0.00) \wedge (v > 7.58)$
	1	0: $(d <= -0.08) \wedge (v <= 7.54)$ 1: $(d <= 0.04) \wedge (d > -0.08) \wedge (v <= 7.54)$ 2: $(d <= 0.04) \wedge (v <= 7.68) \wedge (v > 7.54)$ 3: $(d <= 0.04) \wedge (v > 7.68)$ 4: $(d <= 0.30) \wedge (d > 0.04)$ 5: $(d > 0.30)$
	2	0: $(\kappa <= 0.00) \wedge (d <= -0.11)$ 1: $(\kappa > 0.00) \wedge (d <= -0.11)$ 2: $(d <= 0.03) \wedge (d > -0.11) \wedge (v <= 7.50)$ 3: $(d <= 0.03) \wedge (d > -0.11) \wedge (v > 7.50)$ 4: $(d <= 0.29) \wedge (d > 0.03)$ 5: $(d > 0.29)$
	3	0: $(v <= 7.25)$ 1: $(d <= 0.03) \wedge (v <= 7.66) \wedge (v > 7.25)$ 2: $(d > 0.03) \wedge (v <= 7.66) \wedge (v > 7.25)$ 3: $(\delta <= 0.00) \wedge (v > 7.66)$ 4: $(\delta > 0.00) \wedge (v > 7.66)$
	4	0: $(\delta <= 0.05) \wedge (\kappa <= 0.00) \wedge (d <= 0.23)$ 1: $(\delta <= 0.05) \wedge (\kappa > 0.00) \wedge (d <= 0.23)$ 2: $(\delta <= 0.05) \wedge (d > 0.23)$ 3: $(\delta > 0.05)$
	5	0: $(\delta <= 0.04)$ 1: $(\delta > 0.04)$
	6	0: None
	7	0: $(\delta <= -0.01) \wedge (v <= 7.39)$ 1: $(\delta > -0.01) \wedge (v <= 7.39)$ 2: $(\kappa <= 0.02) \wedge (d <= 0.01) \wedge (v > 7.39)$ 3: $(\kappa > 0.02) \wedge (d <= 0.01) \wedge (v > 7.39)$ 4: $(d > 0.01) \wedge (v > 7.39)$

Table 31: Logic program of LSTM in end-to-end visual servoing (Image-based Driving).

Model	Neuron	Logic Program
CfC	0	0: $(\delta <= -0.03) \wedge (\mu <= 0.02)$ 1: $(\delta > -0.03) \wedge (\mu <= -0.00)$ 2: $(\delta > -0.03) \wedge (\mu <= 0.02) \wedge (\mu > -0.00)$ 3: $(\mu > 0.02) \wedge (v <= 6.79)$ 4: $(\mu > 0.02) \wedge (v > 6.79)$
	1	0: $(\mu <= -0.05)$ 1: $(\delta <= -0.03) \wedge (\mu > -0.05)$ 2: $(\delta > -0.03) \wedge (\kappa <= 0.00) \wedge (\mu > -0.05)$ 3: $(\delta > -0.03) \wedge (\kappa > 0.00) \wedge (\mu > -0.05)$
	2	0: $(\mu <= 0.02) \wedge (d <= 0.12) \wedge (v <= 7.23)$ 1: $(\mu <= 0.02) \wedge (d <= 0.12) \wedge (v > 7.23)$ 2: $(\mu <= 0.00) \wedge (d > 0.12)$ 3: $(\mu <= 0.02) \wedge (\mu > 0.00) \wedge (d > 0.12)$ 4: $(\mu > 0.02) \wedge (v <= 6.79)$ 5: $(\mu > 0.02) \wedge (v > 6.79)$
	3	0: $(\delta <= -0.04)$ 1: (conflict) 2: $(\delta > -0.04) \wedge (d <= 0.16)$ 3: $(\delta > -0.04) \wedge (d > 0.16)$
	4	0: $(\kappa <= 0.00) \wedge (\mu <= -0.00)$ 1: $(\kappa > 0.00) \wedge (\mu <= -0.00)$ 2: $(\mu <= 0.02) \wedge (\mu > -0.00) \wedge (d <= 0.40)$ 3: $(\mu <= 0.02) \wedge (\mu > -0.00) \wedge (d > 0.40)$ 4: $(\mu > 0.02) \wedge (v <= 6.87)$ 5: $(\mu > 0.02) \wedge (v > 6.87)$
	5	0: $(v <= 6.41)$ 1: $(\mu <= 0.00) \wedge (v <= 7.15) \wedge (v > 6.41)$ 2: $(\mu <= 0.00) \wedge (v > 7.15)$ 3: $(\mu > 0.00) \wedge (d <= 0.51) \wedge (v > 6.41)$ 4: $(\mu > 0.00) \wedge (d > 0.51) \wedge (v > 6.41)$
	6	0: $(\delta <= -0.00)$ 1: (conflict) 2: $(\delta <= 0.04) \wedge (\delta > -0.00) \wedge (v <= 7.18)$ 3: $(\delta <= 0.04) \wedge (\delta > -0.00) \wedge (v > 7.18)$ 4: $(\delta <= 0.07) \wedge (\delta > 0.04)$ 5: $(\delta > 0.07)$
	7	0: $(\delta <= -0.02) \wedge (v <= 7.26)$ 1: $(\delta > -0.02) \wedge (v <= 6.56)$ 2: $(\delta > -0.02) \wedge (v <= 7.26) \wedge (v > 6.56)$ 3: $(\delta <= 0.00) \wedge (v <= 7.55) \wedge (v > 7.26)$ 4: $(\delta > 0.00) \wedge (v <= 7.55) \wedge (v > 7.26)$ 5: $(v > 7.55)$

Table 32: Logic program of ODE-RNN in end-to-end visual servoing (Image-based Driving).

Model	Neuron	Logic Program
ODE-RNN	0	0: $(d <= -0.04) \wedge (v <= 5.48)$ 1: $(d <= -0.04) \wedge (v <= 7.46) \wedge (v > 5.48)$ 2: $(\mu <= 0.02) \wedge (d > -0.04) \wedge (v <= 7.46)$ 3: $(\mu > 0.02) \wedge (d > -0.04) \wedge (v <= 7.46)$ 4: $(\delta <= 0.05) \wedge (v <= 7.84) \wedge (v > 7.46)$ 5: $(\delta <= 0.05) \wedge (v > 7.84)$ 6: $(\delta > 0.05) \wedge (v > 7.46)$
	1	0: $(d <= -0.72)$ 1: $(d > -0.72) \wedge (v <= 5.80)$ 2: $(\delta <= -0.00) \wedge (d > -0.72) \wedge (v > 5.80)$ 3: $(\delta > -0.00) \wedge (d > -0.72) \wedge (v > 5.80)$
	2	0: $(v <= 5.00)$ 1: $(\delta <= 0.09) \wedge (v <= 6.99) \wedge (v > 5.00)$ 2: $(\delta <= 0.09) \wedge (v > 6.99)$ 3: $(\delta > 0.09) \wedge (v <= 6.98) \wedge (v > 5.00)$ 4: $(\delta > 0.09) \wedge (v > 6.98)$
	3	0: $(d <= -0.04)$ 1: $(d <= 0.04) \wedge (d > -0.04)$ 2: $(\mu <= -0.11) \wedge (d > 0.04)$ 3: $(\mu > -0.11) \wedge (d <= 0.92) \wedge (d > 0.04)$ 4: $(\mu > -0.11) \wedge (d > 0.92)$
	4	0: $(\mu <= 0.02) \wedge (d <= 0.60) \wedge (v <= 7.04)$ 1: $(\mu <= 0.02) \wedge (d > 0.60) \wedge (v <= 7.04)$ 2: $(\mu > 0.02) \wedge (d <= -0.43) \wedge (v <= 7.04)$ 3: $(\mu > 0.02) \wedge (d > -0.43) \wedge (v <= 7.04)$ 4: $(v <= 7.53) \wedge (v > 7.04)$ 5: $(d <= 0.38) \wedge (v > 7.53)$ 6: $(d > 0.38) \wedge (v > 7.53)$
	5	0: $(v <= 5.38)$ 1: $(\delta <= -0.03) \wedge (d <= 0.10) \wedge (v > 5.38)$ 2: $(\delta > -0.03) \wedge (d <= 0.10) \wedge (v > 5.38)$ 3: $(\delta <= -0.02) \wedge (d > 0.10) \wedge (v > 5.38)$ 4: $(\delta > -0.02) \wedge (d > 0.10) \wedge (v > 5.38)$
	6	0: $(\mu <= 0.01) \wedge (d <= 0.18)$ 1: $(\mu <= 0.01) \wedge (d <= 0.47) \wedge (d > 0.18)$ 2: $(\delta <= 0.10) \wedge (\mu > 0.01) \wedge (d <= 0.47)$ 3: $(\delta > 0.10) \wedge (\mu > 0.01) \wedge (d <= 0.47)$ 4: $(d > 0.47) \wedge (v <= 6.60)$ 5: $(\mu <= -0.00) \wedge (d > 0.47) \wedge (v > 6.60)$ 6: $(\mu > -0.00) \wedge (d > 0.47) \wedge (v > 6.60)$
	7	0: $(v <= 5.15)$ 1: $(\mu <= 0.12) \wedge (d <= -0.08) \wedge (v > 5.15)$ 2: $(\mu <= 0.12) \wedge (d > -0.08) \wedge (v > 5.15)$ 3: $(\mu > 0.12) \wedge (v > 5.15)$

Table 33: Logic program of CfC in end-to-end visual servoing (Image-based Driving).

Model	Neuron	Logic Program
NCP	0	0: $(\delta <= -0.05)$ 1: $(\delta <= 0.02) \wedge (\delta > -0.05) \wedge (\mu <= -0.01)$ 2: $(\delta <= 0.02) \wedge (\delta > -0.05) \wedge (\mu > -0.01)$ 3: $(\delta <= 0.09) \wedge (\delta > 0.02) \wedge (\mu <= -0.01)$ 4: $(\delta <= 0.09) \wedge (\delta > 0.02) \wedge (\mu > -0.01)$ 5: $(\delta > 0.09)$
	1	0: $(\mu <= 0.05)$ 1: $(\mu > 0.05)$
	2	0: $(\delta <= 0.02) \wedge (\mu <= -0.03)$ 1: $(\delta > 0.02) \wedge (\mu <= -0.03)$ 2: $(\delta <= -0.02) \wedge (\mu > -0.03)$ 3: $(\delta > -0.02) \wedge (\mu > -0.03)$
	3	0: $(v <= 7.41)$ 1: $(v <= 7.72) \wedge (v > 7.41)$ 2: $(d <= -0.02) \wedge (v > 7.72)$ 3: $(d <= 0.10) \wedge (d > -0.02) \wedge (v > 7.72)$ 4: $(d > 0.10) \wedge (v <= 8.05) \wedge (v > 7.72)$ 5: $(d > 0.10) \wedge (v > 8.05)$
	4	0: $(v <= 7.45)$ 1: $(v <= 7.78) \wedge (v > 7.45)$ 2: $(v <= 8.08) \wedge (v > 7.78)$ 3: $(v > 8.08)$
	5	0: $(d <= -0.15) \wedge (v <= 7.65)$ 1: $(d <= -0.15) \wedge (v > 7.65)$ 2: $(\mu <= 0.06) \wedge (d <= 0.04) \wedge (d > -0.15)$ 3: $(\mu <= 0.06) \wedge (d > 0.04)$ 4: $(\mu > 0.06) \wedge (d > -0.15)$
	6	0: $(\delta <= -0.03) \wedge (v <= 7.73)$ 1: $(\delta <= -0.03) \wedge (v > 7.73)$ 2: $(\delta > -0.03) \wedge (\mu <= 0.07) \wedge (d <= 0.02)$ 3: $(\delta > -0.03) \wedge (\mu <= 0.07) \wedge (d > 0.02)$ 4: $(\delta > -0.03) \wedge (\mu > 0.07)$
	7	0: None

Table 34: Logic program of NCP in end-to-end visual servoing (Image-based Driving).

References

[1] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[2] M. Lechner, R. Hasani, M. Zimmer, T. A. Henzinger, and R. Grosu. Designing worm-inspired neural networks for interpretable robotic control. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 87–94. IEEE, 2019.

[3] R. Hasani, M. Lechner, A. Amini, D. Rus, and R. Grosu. Liquid time-constant networks. *arXiv preprint arXiv:2006.04439*, 2020.

[4] M. Lechner, R. Hasani, A. Amini, T. A. Henzinger, D. Rus, and R. Grosu. Neural circuit policies enabling auditable autonomy. *Nature Machine Intelligence*, 2(10):642–652, 2020.

[5] C. Vorbach, R. Hasani, A. Amini, M. Lechner, and D. Rus. Causal navigation by continuous-time neural networks. *Advances in Neural Information Processing Systems*, 34, 2021.

[6] J. Schmidhuber. Learning factorial codes by predictability minimization. *Neural computation*, 4(6):863–879, 1992.

[7] J. Peters, D. Janzing, and B. Schölkopf. *Elements of causal inference: foundations and learning algorithms*. The MIT Press, 2017.

[8] F. Locatello, S. Bauer, M. Lucic, G. Raetsch, S. Gelly, B. Schölkopf, and O. Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *international conference on machine learning*, pages 4114–4124. PMLR, 2019.

[9] S. Emmons, B. Eysenbach, I. Kostrikov, and S. Levine. Rvs: What is essential for offline RL via supervised learning? In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=S874XAIpkR-.

[10] C. R. Banbury, V. J. Reddi, M. Lam, W. Fu, A. Fazel, J. Holleman, X. Huang, R. Hurtado, D. Kanter, A. Lokhmotov, et al. Benchmarking tinyml systems: Challenges and direction. *arXiv preprint arXiv:2003.04821*, 2020.

[11] C. Baykal, L. Liebenwein, I. Gilitschenski, D. Feldman, and D. Rus. Sensitivity-informed provable pruning of neural networks. *SIAM Journal on Mathematics of Data Science*, 4(1): 26–45, 2022. doi:10.1137/20M1383239. URL https://doi.org/10.1137/20M1383239.

[12] C. Hawkins, X. Liu, and Z. Zhang. Towards compact neural networks via end-to-end training: A bayesian tensor approach with automatic rank determination. *SIAM Journal on Mathematics of Data Science*, 4(1):46–71, 2022. doi:10.1137/21M1391444. URL https://doi.org/10.1137/21M1391444.

[13] S. Oymak. Learning compact neural networks with regularization. In J. Dy and A. Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 3966–3975. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/oymak18a.html.

[14] J. Bu, A. Daw, M. Maruf, and A. Karpatne. Learning compact representations of neural networks using discriminative masking (dam). *Advances in Neural Information Processing Systems*, 34, 2021.

[15] M. Torkamani, P. Wallis, S. Shankar, and A. Rooshenas. Learning compact neural networks using ordinary differential equations as activation functions. *arXiv preprint arXiv:1905.07685*, 2019.

[16] B. A. Toms, E. A. Barnes, and I. Ebert-Uphoff. Physically interpretable neural networks for the geosciences: Applications to earth system variability. *Journal of Advances in Modeling Earth Systems*, 12(9):e2019MS002002, 2020.

[17] R. Hasani, A. Amini, M. Lechner, F. Naser, R. Grosu, and D. Rus. Response characterization for auditing cell dynamics in long short-term memory networks. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2019.

[18] D. Alvarez Melis and T. Jaakkola. Towards robust interpretability with self-explaining neural networks. *Advances in neural information processing systems*, 31, 2018.

[19] Q. Zhang, Y. N. Wu, and S.-C. Zhu. Interpretable convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8827–8836, 2018.

[20] R. T. Chen, X. Li, R. B. Grosse, and D. K. Duvenaud. Isolating sources of disentanglement in variational autoencoders. *Advances in neural information processing systems*, 31, 2018.

[21] A. Heuillet, F. Couthouis, and N. Díaz-Rodríguez. Explainability in deep reinforcement learning. *Knowledge-Based Systems*, 214:106685, 2021.

[22] O. Bastani, Y. Pu, and A. Solar-Lezama. Verifiable reinforcement learning via policy extraction. *Advances in neural information processing systems*, 31, 2018.

[23] A. Silva, M. Gombolay, T. Killian, I. Jimenez, and S.-H. Son. Optimization methods for interpretable differentiable decision trees applied to reinforcement learning. In *International conference on artificial intelligence and statistics*, pages 1855–1865. PMLR, 2020.

[24] Z. Ding, P. Hernandez-Leal, G. W. Ding, C. Li, and R. Huang. Cdt: Cascading decision trees for explainable reinforcement learning. *arXiv preprint arXiv:2011.07553*, 2020.

[25] A. Pace, A. J. Chan, and M. van der Schaar. Poetree: Interpretable policy learning with adaptive decision trees. *arXiv preprint arXiv:2203.08057*, 2022.

[26] D. Amir and O. Amir. Highlights: Summarizing agent behavior to people. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, pages 1168–1176, 2018.

[27] N. Topin and M. Veloso. Generation of policy-level explanations for reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 2514–2521, 2019.

[28] K. Ridgeway and M. C. Mozer. Learning deep disentangled embeddings with the f-statistic loss. *Advances in neural information processing systems*, 31, 2018.

[29] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, 2014.

[30] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[31] Y. Rubanova, R. T. Chen, and D. K. Duvenaud. Latent ordinary differential equations for irregularly-sampled time series. *Advances in neural information processing systems*, 32, 2019.

[32] R. Hasani, M. Lechner, A. Amini, L. Liebenwein, M. Tschaikowski, G. Teschl, and D. Rus. Closed-form continuous-depth models. *arXiv preprint arXiv:2106.13898*, 2021.

[33] I. Lage, E. Chen, J. He, M. Narayanan, B. Kim, S. Gershman, and F. Doshi-Velez. An evaluation of the human-interpretability of explanation. *arXiv preprint arXiv:1902.00006*, 2019.

[34] H. Lakkaraju, S. H. Bach, and J. Leskovec. Interpretable decision sets: A joint framework for description and prediction. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1675–1684, 2016.

[35] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

[36] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[37] A. Amini, T.-H. Wang, I. Gilitschenski, W. Schwarting, Z. Liu, S. Han, S. Karaman, and D. Rus. Vista 2.0: An open, data-driven simulator for multimodal sensing and policy learning for autonomous vehicles. In *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022.

[38] A. Camacho and S. A. McIlraith. Learning interpretable models expressed in linear temporal logic. In *Proceedings of the International Conference on Automated Planning and Scheduling*, volume 29, pages 621–630, 2019.

[39] L. Console, C. Picardi, and D. T. Duprè. Temporal decision trees: Model-based diagnosis of dynamic systems on-board. *Journal of artificial intelligence research*, 19:469–512, 2003.

[40] R. T. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018.

[41] J. Frankle and M. Carbin. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *International Conference on Learning Representations*, 2018.

[42] L. Liebenwein, R. Hasani, A. Amini, and D. Rus. Sparse flows: Pruning continuous-depth models. *Advances in Neural Information Processing Systems*, 34, 2021.

[43] L. Liebenwein, C. Baykal, B. Carter, D. Gifford, and D. Rus. Lost in pruning: The effects of pruning neural networks beyond test accuracy. *Proceedings of Machine Learning and Systems*, 3:93–138, 2021.

[44] L. Li, T. J. Walsh, and M. L. Littman. Towards a unified theory of state abstraction for mdps. In *AI&M*, 2006.

[45] T. D. Kulkarni, K. Narasimhan, A. Saeedi, and J. Tenenbaum. Hierarchical deep reinforcement learning: Integrating temporal abstraction and intrinsic motivation. *Advances in neural information processing systems*, 29, 2016.

[46] E. S. Spelke and K. D. Kinzler. Core knowledge. *Developmental science*, 10(1):89–96, 2007.

[47] T. Dean and R. Givan. Model minimization in markov decision processes. In *AAAI/IAAI*, pages 106–111, 1997.

[48] E. van der Pol, T. Kipf, F. A. Oliehoek, and M. Welling. Plannable approximations to mdp homomorphisms: Equivariance under actions. *arXiv preprint arXiv:2002.11963*, 2020.

[49] I. Higgins, D. Amos, D. Pfau, S. Racaniere, L. Matthey, D. Rezende, and A. Lerchner. Towards a definition of disentangled representations. *arXiv preprint arXiv:1812.02230*, 2018.

[50] H. Caselles-Dupré, M. Garcia Ortiz, and D. Filliat. Symmetry-based disentangled representation learning requires interaction with environments. *Advances in Neural Information Processing Systems*, 32, 2019.