Whole-Brain Connectomic Graph Neural Networks Enable Whole-Body Locomotion Control in Drosophila

Zehao Jin
Tsinghua University
lunamos.thu@gmail.com

Yanan Sui
Tsinghua University
ysui@tsinghua.edu.cn

Abstract

Whole-brain connectomics provides a structural blueprint for understanding how neural circuits generate behavior, yet its integration with embodied locomotion models remains largely unexplored. Using FlyWire, the complete connectome of an adult female Drosophila, we present fly-connectomic Graph Neural Networks (flyGNN), a graph-based algorithm that implements the fruit fly connectome as a recurrent message passing network, enabling locomotion control of a physics-based fruit fly model. Empirically, flyGNN reproduces whole-body locomotion behaviors, including gait initiation, straight walking, and turning, directly from connectome-structured dynamics. Analysis of neuron states further reveals the emergence of functional specialization during locomotion, which can be captured through low-dimensional embeddings. These results demonstrate that whole-brain structural maps can directly support embodied control, establishing a framework for investigating how connectome-derived architectures give rise to sensorimotor coordination in animals.

1 Introduction

Understanding how neural circuits generate behavior is a central goal of neuroscience. Recent advances in whole-brain connectomics have provided unprecedented structural maps of the fruit fly (Drosophila melanogaster) at synaptic resolution [2, 5, 7, 8, 11, 16, 18, 19, 27, 28]. As a representative example, the FlyWire project provides detailed, high-resolution data about the entire brain's neural connections. These resources enable the possibility of linking complete neural wiring diagrams to the sensorimotor behaviors they underlie. Yet a fundamental challenge remains: how can static connectomic data be transformed into dynamic, functional models that reproduce the rich and adaptive motor behavior of animals in real-world environments?

Progress toward this goal has been made in domain-specific settings. Azevedo et al. (2024) reconstructed the Drosophila ventral nerve cord connectome, revealing neural circuits that coordinate leg and wing movements [1, 15]. Shiu et al. (2024) modeled motor neuron activity for feeding and grooming behaviors [20]. Lappalainen et al. (2024) showed that connectome-constrained networks can predict neural activity in the visual system [14]. These studies illustrate the explanatory power of connectomic structure, but they are typically restricted to simplified network or task-specific training. A comprehensive, whole-brain approach that directly integrates the connectome into embodied simulations of locomotion remains missing.

In parallel, embodied intelligence research has produced increasingly sophisticated simulations of animal and robotic movement. Reinforcement learning has enabled robust locomotion in humanoids [3, 13], quadrupeds [6] and musculoskeletal systems [10, 26]. In the context of Drosophila, Wang-Chen et al. (2024) developed NeuroMechFly v2, which integrates multiple sensory modalities but relies on hand-crafted Central Pattern Generator modules [17, 25]. Vaxenburg et al. (2025) extended

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: Data on the Brain & Mind.

this direction with a whole-body physics model capable of realistic walking and flight, yet driven by generic multilayer perceptron (MLP) policy networks with no explicit biological grounding [22]. These efforts underscore the promise of embodied modeling, but they stop short of exploiting the structural prior embedded in whole-brain connectivity.

Here, we bridge these domains by directly applying the Drosophila connectome to embodied motor control. Our work benefits from FlyWire[7, 19], which provides a whole-brain connectome of the adult fruit fly, and from flybody[22], a physics-based model of Drosophila locomotion in MuJoCo[21] with validated tasks. We introduce fly-connectomic Graph Neural Networks (flyGNN) - a recurrent message-passing model whose architecture is derived entirely from the connectome. In contrast to conventional controllers based on MLPs or task-specific designs, flyGNN uses the neural wiring diagram itself to structure information flow, hypothesizing that this is sufficient to generate naturalistic locomotor behaviors such as gait initiation, walking, and turning.

Our objective is to explore the feasibility of connectome-driven embodied control and to evaluate whether structural wiring alone can guide motor outputs in realistic environments. By demonstrating locomotion directly from the FlyWire connectome, this work establishes a framework for studying how whole-brain architectures support behavior, advancing both neuroscience and embodied artificial intelligence.

2 Method

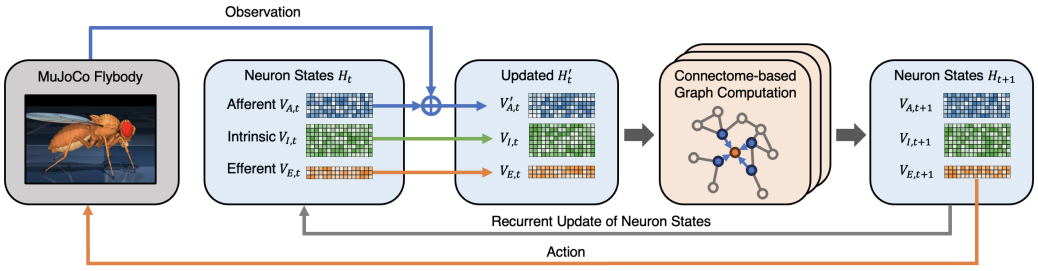

Figure 1: **Model architecture.** Overview of the fly-connectomic Graph Neural Networks (flyGNN). Observations are mapped into afferent neuron states through dimensional transformation. Neural states are then propagated through the connectome-based graph computation module, where update rules are directly constrained by the FlyWire connectome. The updated efferent neuron states are converted into motor outputs that drive the MuJoCo-based simulated Drosophila.

We consider the problem of embodied sensorimotor control for a virtual fruit fly agent interacting with a physics-based environment. The agent perceives the environment through a vector of observations and generates motor commands to actuate its body. Formally, let the environment state at time step t be denoted by $s_t \in \mathcal{S}$. The agent receives a observation $x_t \in \mathbb{R}^{d_{\text{in}}}$, which corresponds to a set of processed features such as proprioceptive and exteroceptive signals. Based on this input, the agent produces an action $a_t \in \mathbb{R}^{d_{\text{out}}}$, representing the motor outputs that drive the flybody physics model of locomotion.

The control policy is parameterized by a graph neural network whose architecture mirrors the anatomical connectivity of the Drosophila brain. Specifically, we represent the connectome as a directed graph $G = (V, E)$, where each node $v \in V$ corresponds to a neuron and each directed edge $(u, v) \in E$ indicates the existence of a synaptic connection from neuron u to neuron v. For simplicity, we discard additional biological details such as neurotransmitter types, synapse counts, or cell morphology, and model the connectome as an **unweighted directed graph** that captures only the existence and direction of connections.

Following FlyWire's classification of flow type, we partition the nodes into three disjoint sets:

- **Afferent neurons**: $V_a \subset V$, which receive external sensory inputs,
- **Intrinsic neurons**: $V_i \subset V$, which mediate recurrent dynamics within the network, and
- **Efferent neurons**: $V_e \subset V$, which produce motor outputs to the body model.

Thus, $V = V_a \cup V_i \cup V_e$ with $V_a \cap V_i \cap V_e = \emptyset$. Each neuron $v \in V$ is associated with a latent state vector $h_v \in \mathbb{R}^C$, and we collect all states at time t as a matrix

$$H_t \in \mathbb{R}^{|V| \times C} \tag{1}$$

At each step, the sensory input x_t is first embedded into afferent states by an encoder layer Enc_θ:

$$z_t = \text{Enc}_\theta(x_t) \in \mathbb{R}^{|V_a| \times C} \tag{2}$$

The afferent states are then updated by a gated mechanism that combines the encoded sensory features and the previous afferent states:

$$h'_a = \tanh\Big(W_g\,[\,z_t \,\|\, H_t[V_a]\,] + b_g\Big), \quad a \in V_a, \tag{3}$$

where $[\,\cdot\|\cdot\,]$ denotes concatenation and W_g, b_g are learnable parameters. This update ensures that sensory information is integrated with the existing neural states before message passing.

After afferent updates, the complete node states H_t are propagated through the connectome graph by a message passing operator $\text{MP}(\cdot)$. In the simplest case of a Graph Convolutional Network (GCN)[12], the update rule is

$$H^{(l+1)} = \sigma\Big(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}\Big) \tag{4}$$

where $\tilde{A} = A + I$ is the adjacency matrix with self-loops, $\tilde{D}$ is the diagonal degree matrix, $W^{(l)}$ is a learnable weight matrix, and $\sigma(\cdot)$ is a nonlinear activation. This operator propagates neural activity along the edges of the connectome, producing updated states for both intrinsic and efferent neurons.

In practice, the message passing module can be instantiated not only as a GCN but also with more expressive architectures such as GraphSAGE[9], GAT[23], or PNA[4], which offer improved performance and scalability (See Section 3.4 for comparative experiments).

The updated efferent states $H_t[V_e]$ are flattened and passed through a decoder layer Dec_ϕ:

$$a_t = \text{Dec}_\phi\Big(H_t[V_e]\Big). \tag{5}$$

This step maps the neural features of efferent neurons directly into continuous motor commands. The environment dynamics are governed by flybody, a biomechanical model of Drosophila implemented in MuJoCo that enables realistic locomotion tasks such as walking [22]. Given the agent's motor outputs a_t, the simulator updates the physical state of the fly and produces the next sensory observation x_{t+1}. Over a trajectory $\tau = (x_0, a_0, x_1, \ldots, x_T)$, the agent's objective is to generate actions that realize stable and efficient locomotion.

To summarize, Algorithm 1 outlines the complete forward computation of our graph neural policy reflecting the anatomical connectivity of the Drosophila brain, from sensory input through afferent gating and message passing on the connectome to efferent decoding into motor actions.

3 Experiments

We applied the flyGNN algorithm to learn the locomotion trajectories provided by flybody [22] and evaluated its ability to reproduce key locomotor behaviors. Our experiments cover gait initiation, straight-line walking and turning. In addition, we visualized neuron state dynamics to examine how internal representations evolve during gait initiation and we compared different message passing operators to analyze their impact on performance.

3.1 Environment

For all experiments we used the flybody, the fruit fly body model for MuJoCo physics, which provides a physics-based model of the adult Drosophila and expert locomotion trajectories for imitation learning.

The observation space includes proprioceptive and exteroceptive signals (e.g., joint positions and velocities, forces, reference trajectories), totaling 741 input dimensions. The action space covers 59 control channels for adhesion, head and abdomen movement, and leg joints.

Algorithm 1: fly-connectomic Graph Neural Network (flyGNN)

Input: Sensory input x_t, connectome graph $G = (V, E)$ with node partitions V_a (afferent), V_i (intrinsic), V_e (efferent); encoder Enc; gate (W_g, b_g); message passing MP; decoder Dec

Output: Motor output a_t

for *each time step* t **do**

```
// 1. Encode sensory input
```
$z_t \leftarrow \mathrm{Enc}(x_t)$;

```
// 2. Gated update of afferent neurons
```
$H_t[V_a] \leftarrow \tanh\big(W_g[\, z_t, H_t[V_a]\,] + b_g\big)$;

```
// 3. Message passing on connectome
```
$H_{t+1} \leftarrow \mathrm{MP}(H_t, E)$;

```
// 4. Decode efferent states into motor outputs
```
$a_t \leftarrow \mathrm{Dec}(H_{t+1}[V_e])$;

```
// 5. Environment transition
```
Apply a_t to MuJoCo to obtain x_{t+1} ;

end

3.2 Training

Our training strategy aims to transfer the walking behavior from the pretrained policy released with flybody into flyGNN. The teacher policy π_t is a multilayer perceptron with four hidden layers of size 512, using LayerNorm and a tanh activation in the first layer followed by ELU activations in the subsequent layers, which outputs the mean and variance of a Gaussian distribution over 59 action dimensions. This policy was originally trained with reinforcement learning and serves as the reference for student supervision.

The student policy π_s adopts the flyGNN architecture described in Section 2, but with a minimal encoder and decoder design to keep the network lightweight. The encoder is a linear layer with ReLU activation mapping sensory observations into afferent neuron features. After message passing through the connectome graph, the efferent states are decoded by a linear layer with ReLU activation, followed by two parallel linear heads that generate the action mean and log-standard deviation. These shallow encoder and decoder components are included only to adapt tensor dimensions, with the majority of computation occurring within the message passing module.

To construct the supervision signal, we roll out the teacher model on the provided imitation dataset and record expert trajectories, truncated to the first 150 steps. The student policy π_s is trained to match the teacher policy π_t in action distribution. At each time step t, the teacher produces a Gaussian distribution over actions, parameterized by mean μ_t and log-standard deviation $\log \sigma_t$. The student predicts $\mu_s, \log \sigma_s$ given the same observation. The train loss combines a Kullback–Leibler divergence term with mean squared error (MSE) regularization:

$$\mathcal{L}_t = D_{\mathrm{KL}}\big(\mathcal{N}(\mu_t, \sigma_t^2) \parallel \mathcal{N}(\mu_s, \sigma_s^2)\big) + \lambda(t)\Big(\|\mu_s - \mu_t\|_2^2 + \alpha \,\|\log \sigma_s - \log \sigma_t\|_2^2\Big) \tag{6}$$

where $\alpha = 21.6$ balances the scale of variance matching, and $\lambda(t)$ is an annealing coefficient that gradually decreases with training steps. As training progresses, the MSE term is annealed to 0, leaving only the KL divergence term to enforce alignment in action distributions. This procedure encourages the student to capture both the mean behavior and the uncertainty of the teacher, while progressively shifting emphasis to distributional matching.

3.3 Locomotion Results

Gait Initiation

Building on the training procedure described above, we evaluated flyGNN on a walking task at a target velocity of 3 cm/s. The student network used in this evaluation was a GraphSAGE variant with 16 node channels and 4 message passing layers.

We first examined the process of gait initiation, focusing on the transition from rest to the onset of stable locomotion. Figure 2 illustrates snapshots of the simulated fly prior to the first complete gait

cycle. The initiation phase lasted for roughly the first 80 ms, during which irregular steps gradually gave way to rhythmic and coordinated leg movements.

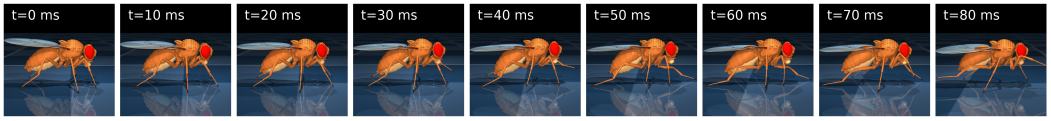

Figure 2: **Gait initiation trajectory visualization.** Snapshots of the simulated fly during the onset of locomotion, prior to the first complete gait cycle. The sequence illustrates how the agent transitions from rest into stepping, with irregular and asymmetric leg movements gradually giving rise to a coordinated pattern. This stage corresponds to the initial gait initiation phase before stable tripod coordination emerges.

To further probe the neural dynamics underlying this transition, we visualized internal neuron states sampled from the same episode before the first stable gait cycle. Specifically, we collected 16-dimensional features from all 139,246 neurons at selected times $[0, 20, 40, 60, 80]$ ms and reduced them to two dimensions using UMAP.

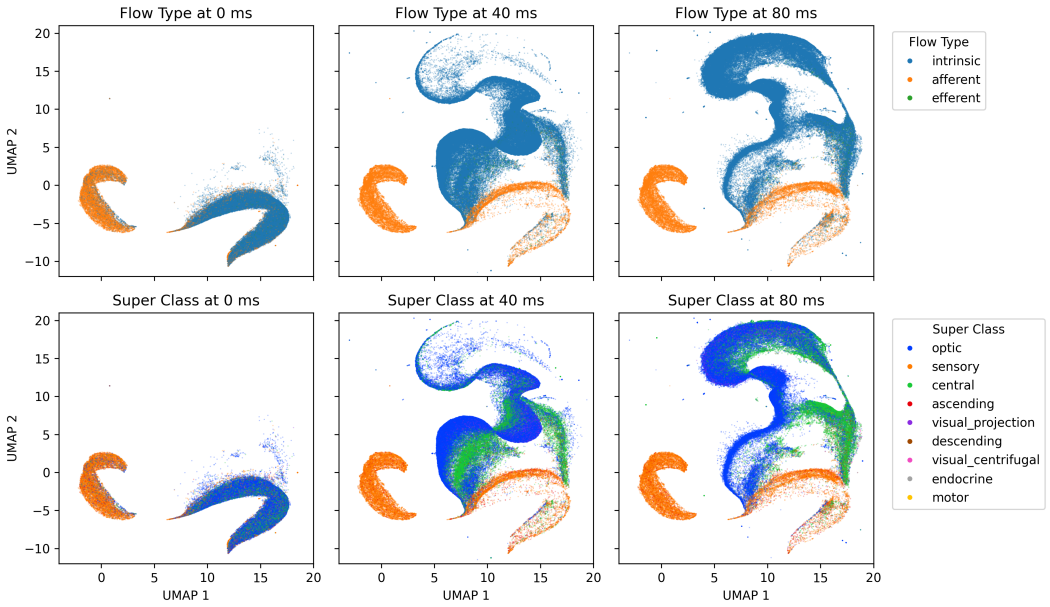

Figure 3: **Neuron state embeddings during gait initiation.** UMAP projections of neuronal features at different time points (0 ms, 40 ms, 80 ms). Top: neurons colored by FlyWire flow type (afferent, intrinsic, efferent). Bottom: neurons colored by super class annotations. The visualization shows that initially mixed features gradually separate into distinct clusters, with optic and central populations becoming clearly differentiated by 80 ms. This structured reorganization indicates that the flyGNN policy induces meaningful neural dynamics, supporting functional specialization as gait coordination emerges.

The visualization in Figure 3 reveals clear temporal changes in the organization of neural states. At initialization (0 ms), neuron features were randomly distributed, with optic and central neurons largely overlapping. As the episode progressed, neurons formed increasingly distinct clusters, with optic and central populations separating into identifiable regions by 80 ms. This indicates that the flyGNN architecture simulates connectome-based neural dynamics that promote functional specialization of different neural groups during locomotor tasks.

These results suggest that the flyGNN framework can capture meaningful reorganization of neural states during behavior. The observed emergence of functional segregation highlights the potential of this approach for investigating neural plasticity, especially the interplay between developmental trajectories, learning mechanisms, and the structural organization of the connectome.

Straight-Line Walking

Building on the analysis of gait initiation, we next evaluated flyGNN on a straight-walking task at a target velocity of 3 cm/s.

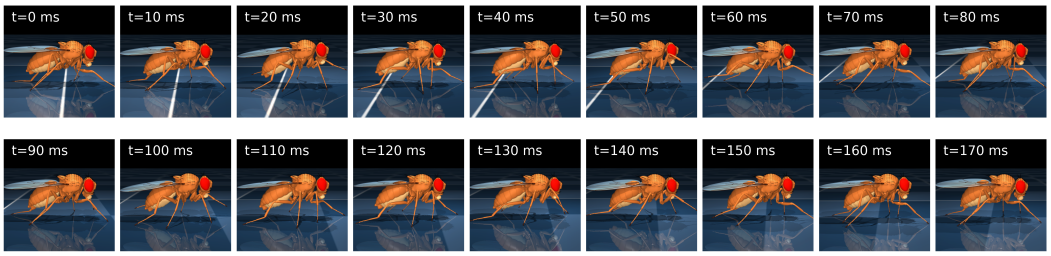

Figure 4: **Walking trajectory visualization.** Snapshots of a simulated fly walking in a straight line at 3 cm/s. The model maintains stable stepping sequences with tripod coordination emerging naturally from flyGNN policy model.

As illustrated in Figure 4, the model produced stable forward locomotion with clear tripod coordination. The simulated fly maintained a consistent body trajectory over hundreds of milliseconds, without exhibiting drift or collapse, indicating that the learned controller generalizes well to sustained walking.

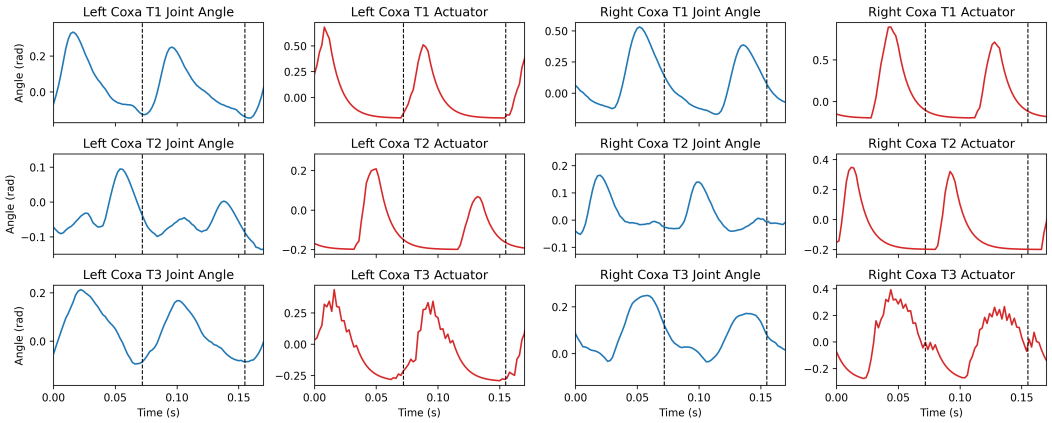

Figure 5: **Joint kinematics and actuator activations during walking.** We visualize the angles (blue) and actuator activations (red) of coxa joints from left (T1–T3) and right (T1–T3) legs over multiple gait cycles. Dashed vertical lines mark gait phases based on the troughs of the left T1 coxa. The model reproduces alternating tripod-like coordination, with left T1/T3 synchronized with right T2, and left T2 synchronized with right T1/T3, consistent with expected fly walking patterns.

Joint-level analysis in Figure 5 shows that actuator outputs are tightly coupled with kinematic trajectories: contralateral legs alternate in phase, producing the classical tripod gait pattern seen in Drosophila. These results demonstrate that flyGNN is sufficient to generate stable straight walking once locomotion is initiated.

Turning

We next assessed whether the same policy could generalize to directional maneuvers. In the turning task, the model was instructed to walk at a forward velocity of 3 cm/s while executing a leftward turn at 10 rad/s. As shown in Figure 6, the simulated fly successfully produced a smooth curved trajectory by modulating stride lengths asymmetrically across the body: legs on the inner side of the turn reduced their stance amplitude, while contralateral legs extended their strides. This modulation of gait symmetry arises naturally from the network dynamics, without requiring task-specific tuning or additional control rules.

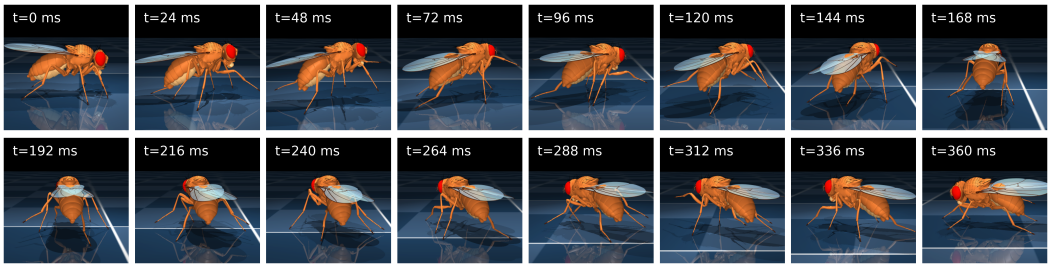

Figure 6: **Turning trajectory visualization.** Snapshots of the virtual fly executing a high-speed left turn at 3 cm/s and 10 rad/s. Over the course of the trajectory, stride lengths on the turning side decrease while those on the contralateral side increase, producing a smooth curved trajectory. This asymmetry demonstrates that flyGNN can generalize beyond straight walking to produce directed maneuvers.

The ability to perform both straight walking and turning indicates that the learned flyGNN policy does not simply memorize a single stereotyped gait, but rather encodes a flexible control strategy that can adapt to new locomotor demands. These findings highlight the robustness of the architecture and suggest its potential for modeling a wider repertoire of multi-task behaviors.

3.4 Performance Across Message Passing Operators

We evaluated flyGNN across several message passing operators, including GCN[12], EdgeCNN[24], GAT[23], GraphSAGE[9] and PNA[4]. Table 1 shows the evaluation KL loss and the average reward within 500 episodes of different message passing operators. We used the imitation learning dataset provided by flybody, and because different imitation trajectories exhibit variability, the rewards naturally show high fluctuation. The results show that simpler operators (e.g., GCN, EdgeCNN) generally converged with higher KL divergence to the teacher and lower average reward, while more expressive operators (e.g., PNA, GraphSAGE) achieved closer distributional matching and longer average rollout stability.

Table 1: Performance comparison of flyGNN with different message passing operators.

Model	Node dim	Depth	Eval KL ↓	Avg. reward ↑
GCN	16	4	7.93	43.13
GraphSAGE($c = 2$)	2	6	6.85	62.85
EdgeCNN	4	2	5.54	90.48
GAT	8	2	3.43	132.65
GraphSAGE	16	4	3.31	125.55
PNA	4	2	2.89	**145.33**

Across all variants, the models exhibited stable locomotion; notably, even the weaker ones retained sufficient ability for walking and turning, and an extreme GraphSAGE variant with only two node channels still performed acceptably.

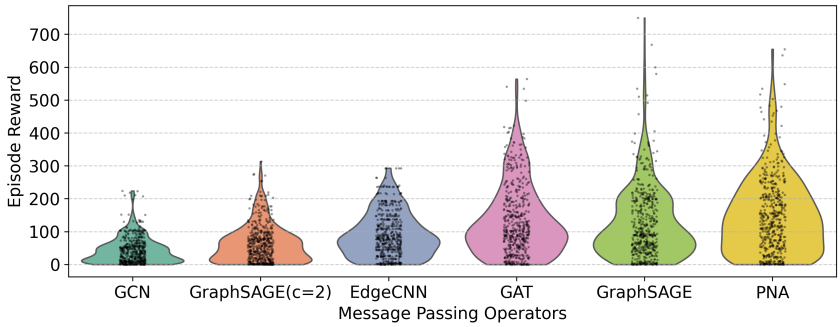

Figure 7: **Performance across message passing operators.** Distribution of episode rewards obtained by GCN, GraphSAGE, EdgeCNN, GAT, and PNA. More expressive operators (GraphSAGE, PNA) achieved higher average rewards and more stable rollouts compared to simpler operators (GCN, EdgeCNN).

4 Conclusion and Discussion

This study demonstrates that fly-connectomic Graph Neural Networks can generate stable locomotion behaviors in a simulated fruit fly, including gait initiation, straight walking and turning. By structuring information flow according to the FlyWire connectome, flyGNN replaces large fully connected policy networks with graph-based networks that directly reflect the wiring diagram of the brain.

From a neuroscience perspective, these results suggest that the connectivity structure of the fruit fly brain carries sufficient information to guide embodied sensorimotor control, even when reduced to an unweighted directed graph without additional annotations such as neurotransmitter types, synapse counts, or cell morphology. This observation is consistent with recent work in sensory domains, where models reflecting connectome architecture alone were able to predict neural activity. Our results extend this line of evidence into the domain of locomotion control, showing that whole-brain connectivity can provide a strong inductive bias for generating embodied behavior.

Beyond reproducing locomotion, flyGNN also yields interpretable internal dynamics. Low-dimensional projections of neural states revealed the emergence of functional segregation during gait initiation, with optic and central populations gradually differentiating. These results suggest that connectivity-based architectures not only support motor control but also induce structured neural activity patterns, providing a potential tool for studying neural plasticity and the interplay between structure, learning, and function.

From the viewpoint of embodied intelligence, this approach highlights the value of biological structure as a prior for controller design. Rather than relying on generic multilayer perceptrons, flyGNN organizes computation around the topology of the connectome, providing an interpretable framework where the architecture is directly grounded in neural connectivity. This perspective shifts the emphasis from arbitrary architectural choices toward neuroscience-inspired structure, offering a principled way to study how connectivity can shape embodied control strategies in complex bodies.

At the same time, several limitations highlight opportunities for future progress. First, our implementation discards biophysical details and only preserves the existence and direction of connections, leading to the loss of important biological information. Incorporating richer annotations — such as synapse weights, neurotransmitter types, or cell morphologies — could improve biological fidelity. Second, training efficiency remains a bottleneck: compared to MLP-based controllers, our model requires longer per-step computation and higher VRAM usage, which prolongs training and complicates debugging. Finally, our experiments are limited to locomotion; extending the framework to multimodal behaviors such as visually guided navigation, escape responses, or flight control will provide a more comprehensive test of the generality of connectome-based control.

In summary, flyGNN demonstrates that whole-brain connectomes can be used for embodied motor control. By grounding controller design directly in neural architecture, this work connects connectomics with embodied artificial intelligence and opens a pathway toward mechanistic models that explain how structural wiring supports the adaptive behaviors of animals.

References

[1] Azevedo, A. et al. Connectomic reconstruction of a female drosophila ventral nerve cord. *Nature*, 631(8020):360–368, July 2024. ISSN 0028-0836, 1476-4687. doi: 10.1038/s41586-024-07389-x.

[2] Buhmann, J. et al. Automatic detection of synaptic partners in a whole-brain drosophila electron microscopy data set. *Nature Methods*, 18(7):771–774, July 2021. ISSN 1548-7091, 1548-7105. doi: 10.1038/s41592-021-01183-7.

[3] Cheng, X. et al. Expressive whole-body control for humanoid robots, 2024. URL `https://arxiv.org/abs/2402.16796`.

[4] Corso, G. et al. Principal neighbourhood aggregation for graph nets, December 2020.

[5] Deutsch, D. et al. Sexually-dimorphic neurons in the drosophila whole-brain connectome, June 2025.

[6] Ding, Y. et al. Representation-free model predictive control for dynamic motions in quadrupeds. *IEEE Transactions on Robotics*, 37(4):1154–1171, August 2021. ISSN 1941-0468. doi: 10.1109/tro.2020.3046415. URL `http://dx.doi.org/10.1109/TRO.2020.3046415`.

[7] Dorkenwald, S. et al. Neuronal wiring diagram of an adult brain. *Nature*, 634(8032):124–138, October 2024. ISSN 0028-0836, 1476-4687. doi: 10.1038/s41586-024-07558-y.

[8] Eckstein, N. et al. Neurotransmitter classification from electron microscopy images at synaptic sites in drosophila melanogaster. *Cell*, 187(10):2574–2594.e23, May 2024. ISSN 00928674. doi: 10.1016/j.cell.2024.03.016.

[9] Hamilton, W.L., Ying, R. and Leskovec, J. Inductive representation learning on large graphs, September 2018.

[10] He, K. et al. Dynsyn: Dynamical synergistic representation for efficient learning and control in overactuated embodied systems, 2024. URL `https://arxiv.org/abs/2407.11472`.

[11] Heinrich, L. et al. Synaptic cleft segmentation in non-isotropic volume electron microscopy of the complete drosophila brain. In Frangi, A.F. et al, editors, *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*, volume 11071, pages 317–325. Springer International Publishing, Cham, 2018. ISBN 978-3-030-00933-5 978-3-030-00934-2. doi: 10.1007/978-3-030-00934-2_36.

[12] Kipf, T.N. and Welling, M. Semi-supervised classification with graph convolutional networks, February 2017.

[13] Kumar, A. et al. Rma: Rapid motor adaptation for legged robots, 2021. URL `https://arxiv.org/abs/2107.04034`.

[14] Lappalainen, J.K. et al. Connectome-constrained networks predict neural activity across the fly visual system. *Nature*, 634(8036):1132–1140, 2024. ISSN 0028-0836. doi: 10.1038/s41586-024-07939-3.

[15] Lesser, E. et al. Synaptic architecture of leg and wing premotor control networks in drosophila. *Nature*, 631(8020):369–377, July 2024. ISSN 1476-4687. doi: 10.1038/s41586-024-07600-z.

[16] Lin, A. et al. Network statistics of the whole-brain connectome of drosophila. *Nature*, 634(8032):153–165, October 2024. ISSN 0028-0836, 1476-4687. doi: 10.1038/s41586-024-07968-y.

[17] Lobato-Rios, V. et al. NeuroMechFly, a neuromechanical model of adult Drosophila melanogaster. *Nature Methods*, 19(5):620–627, May 2022. doi: 10.1038/s41592-022-01466-7. URL `https://doi.org/10.1038/s41592-022-01466-7`.

[18] Matsliah, A. et al. Neuronal parts list and wiring diagram for a visual system. *Nature*, 634(8032):166–180, October 2024. ISSN 0028-0836, 1476-4687. doi: 10.1038/s41586-024-07981-1.

[19] Schlegel, P. et al. Whole-brain annotation and multi-connectome cell typing of drosophila. *Nature*, 634(8032):139–152, October 2024. ISSN 0028-0836, 1476-4687. doi: 10.1038/s41586-024-07686-5.

[20] Shiu, P.K. et al. A drosophila computational brain model reveals sensorimotor processing. *Nature*, 634(8032):210–219, October 2024. ISSN 1476-4687. doi: 10.1038/s41586-024-07763-9.

[21] Todorov, E., Erez, T. and Tassa, Y. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5026–5033. IEEE, 2012. doi: 10.1109/IROS.2012.6386109.

[22] Vaxenburg, R. et al. Whole-body physics simulation of fruit fly locomotion. *Nature*, 643(8074): 1312–1320, July 2025. ISSN 0028-0836, 1476-4687. doi: 10.1038/s41586-025-09029-4.

[23] Veličković, P. et al. Graph attention networks, February 2018.

[24] Wang, Y. et al. Dynamic graph cnn for learning on point clouds, June 2019.

[25] Wang-Chen, S. et al. Neuromechfly v2: simulating embodied sensorimotor control in adult drosophila. *Nature Methods*, 21(12):2353–2362, November 2024. ISSN 1548-7105. doi: 10.1038/s41592-024-02497-y. URL `http://dx.doi.org/10.1038/s41592-024-02497-y`.

[26] Wei, Y. et al. Motion control of high-dimensional musculoskeletal systems with hierarchical model-based planning, 2025. URL `https://arxiv.org/abs/2505.08238`.

[27] Yu, S.C. et al. New synapse detection in the whole-brain connectome of drosophila, July 2025. ISSN 2692-8205.

[28] Zheng, Z. et al. A complete electron microscopy volume of the brain of adult drosophila melanogaster. *Cell*, 174(3):730–743.e22, July 2018. ISSN 00928674. doi: 10.1016/j.cell.2018.06.019.