GEOMETRY BEYOND SYMMETRY: ON-MANIFOLD EQUIVARIANT GRAPH NETWORKS FOR ROBOT DYNAMICS

Anonymous authors
Paper under double-blind review

ABSTRACT

Predictive controllers rely on dynamics models to evaluate candidate action sequences, making accurate multi-step pose prediction critical to reliable planning and control. Robot dynamics present two intertwined structures: their predictions should remain consistent across equivalent reference frames, while their pose states evolve on a non-Euclidean configuration space. Existing $\mathrm{E}(n)$-equivariant graph neural networks encode symmetry for Euclidean point coordinates, but do not respect the manifold geometry of rotational states. We introduce an On-Manifold Equivariant Graph Neural Network (OM-EGNN) that represents robot poses on the product manifold of translation and rotation, computes relative poses in receiver-local tangent coordinates, and retracts learned updates onto the manifold. Moreover, the radial update of standard EGNNs is restricted to scalar-weighted combinations of pairwise displacements. OM-EGNN augments this update with a learned tangent-space residual that escapes this directional span while preserving task-relevant equivariance. On planar Ackermann-steered vehicle dynamics and spatial quadrotor dynamics, we compare OM-EGNN against standard canonicalized predictors and Euclidean equivariance-based graph baselines. OM-EGNN achieves the lowest position RMSE in both settings. Rotation and velocity are not uniformly best: planar velocity is lowest for a Euclidean lift, and 3D rotation is lowest for a canonicalized MLP. Unlike the Euclidean graph baselines, OM-EGNN keeps the frame-attack gap near zero under the task symmetry. The tangent residual substantially improves position prediction. In matched MPPI, OM-EGNN has the best mean progress, success, and tracking RMSE, but those margins lie within run variation of the strongest canonicalized predictors, and open-loop rankings do not determine the control ranking.

1 INTRODUCTION

Predictive controllers choose actions by repeatedly evaluating how candidate control sequences will affect future states (Kwon & Han, 2005; Williams et al., 2016; 2018). Their performance therefore depends on dynamics models that remain accurate over the planning horizon. Analytic models can be difficult to specify when tire–ground interaction, aerodynamics, actuator effects, or delay dominate the motion, motivating learned dynamics for ground and aerial robots (Torrente et al., 2021; Salzmann et al., 2023; Xiao et al., 2025). We study direct multi-step prediction, where one model call maps a measured state–control history and candidate future controls to a K-step trajectory (Mohajerin & Waslander, 2019; Asadi et al., 2019; Xiao et al., 2025). For mobile robots, this trajectory must predict both translation and attitude.

Robot dynamics contain two intertwined structures. First, equivalent choices of reference frame should produce correspondingly transformed predictions. This symmetry can be imposed by canonicalizing every input to a reference pose (Xiao et al., 2025; Puny et al., 2022; Kaba et al., 2023), or encoded in an equivariant architecture (Finzi et al., 2020; Satorras et al., 2021). Second, robot pose is not a single Euclidean vector: translation is Euclidean, whereas attitude lies on a rotation manifold (Solà et al., 2018; Barfoot, 2024). These requirements concern different aspects of the model. Frame consistency specifies how predictions must transform; pose geometry specifies where rotational states live and how they can be updated.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

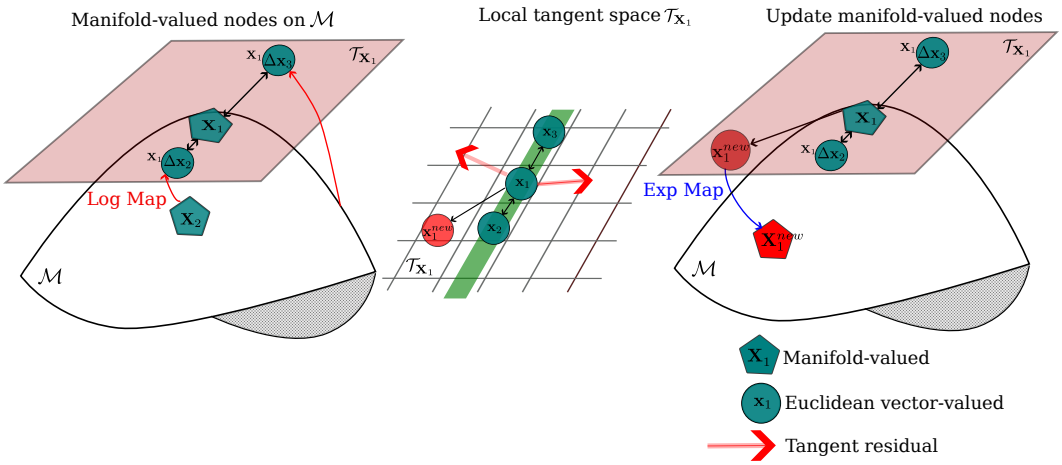


Figure 1: OM-EGNN pose update. Relative poses are expressed in receiver-local tangent coordinates; the displacement and residual branches produce an increment that is retracted onto the pose manifold.

Existing $\mathrm{E}(n)$-equivariant graph neural networks (EGNNs) encode symmetry for Euclidean point coordinates (Satorras et al., 2021). Applied to robot dynamics, straightforward variants either update position alone or concatenate planar (x, y, θ) or spatial position–quaternion (p, q) coordinates and treat them as Euclidean. Angle wrapping or quaternion normalization may return a valid attitude after the update, but does not make the update itself manifold-aware. Consequently, Euclidean equivariance does not simultaneously resolve the reference-frame symmetry and rotational geometry of pose dynamics.

We introduce an On-Manifold Equivariant Graph Neural Network (OM-EGNN) that addresses both structures directly. OM-EGNN represents a pose sequence as a causal graph on the product manifold of translation and rotation. Each layer compares neighboring poses in receiver-local tangent coordinates, aggregates invariant messages, and retracts the resulting update onto the pose manifold (Figure 1). The local coordinates are unchanged by a common reference-frame transformation, while retraction maps the invariant increment to an equivariant pose update. This construction guarantees exact equivariance under the relevant reference-frame transformations without replacing manifold-valued rotation by an ambient Euclidean coordinate.

Geometry and symmetry alone do not ensure an expressive update. The radial coordinate rule of a standard EGNN forms scalar-weighted combinations of pairwise displacements, so its update remains in their directional span. Control-conditioned motion, however, need not lie in that span; a robot starting from rest has no pairwise displacements at all, so such an update cannot move it for any control input. OM-EGNN therefore adds a learned tangent-space residual that can generate directions beyond the observed pairwise displacements. Because the residual is expressed in receiver-local coordinates, it improves update expressivity without sacrificing equivariance.

We evaluate OM-EGNN on planar vehicle dynamics (Srinivasa et al., 2019) and spatial quadrotor dynamics (Bauersfeld et al., 2021) against standard canonicalized predictors and Euclidean equivariance-based graph baselines. OM-EGNN achieves the lowest position RMSE in both settings. Planar velocity is lowest for a Euclidean lift, and 3D rotation is lowest for a canonicalized MLP. Reference-frame diagnostics leave the frame-attack gap of OM-EGNN near zero, while Euclidean graph baselines increase last-step error. The tangent residual substantially improves position prediction over the manifold-only ablation. Finally, using the same planar checkpoints in a matched MPPI evaluation, OM-EGNN achieves the best average progress, success rate, and tracking RMSE, although the control margin over strong canonicalized predictors is modest and open-loop prediction rankings do not reliably rank controllers.

Our contributions are:

1. An on-manifold graph dynamics model that combines product-manifold pose updates with exact equivariance under the relevant reference-frame transformations.

2. A learned tangent-space residual that escapes the directional span of standard EGNN coordinate updates while preserving equivariance.
3. Controlled 2D and 3D prediction studies, reference-frame diagnostics, and a matched predictive-control evaluation that separate the effects of geometry, symmetry, and update expressivity.

2 Problem Formulation

Let the robot state be $s_t = (X_t, v_t)$, where the pose $X_t = (p_t, R_t)$ lies on $\mathcal{M}_n = \mathbb{R}^n \times \mathrm{SO}(n)$ and v_t is the body-frame linear and angular velocity. Given a measured state–control history $\mathcal{H}_t = \{(s_\ell, u_\ell)\}_{\ell=t-H+1}^{t}$ and a candidate future control block $\mathcal{U}_t^+ = (u_{t+1}, \ldots, u_{t+K})$, we learn the direct multi-step predictor

$$\hat{s}_{t+1:t+K} = f_\theta\big(\mathcal{H}_t, \mathcal{U}_t^+\big). \tag{1}$$

All K states are produced in one forward call, following direct multi-step prediction (Mohajerin & Waslander, 2019; Asadi et al., 2019; Xiao et al., 2025).

For $g = (R_g, t_g) \in \mathrm{SE}(n)$, define the action on pose by

$$g \cdot (p, R) = \begin{bmatrix} R_g p + t_g \\ R_g R \end{bmatrix}. \tag{2}$$

This action is applied elementwise to pose histories and targets, while body-frame velocities and controls remain unchanged. For the task-preserving symmetry group G, a frame-consistent model must satisfy

$$f_\theta(g \cdot \mathcal{H}_t, \mathcal{U}_t^+) = g \cdot f_\theta(\mathcal{H}_t, \mathcal{U}_t^+), \qquad g \in G. \tag{3}$$

The 2D Ackermann-steering robot task uses $G = \mathrm{SE}(2)$. On the quadrotor task, gravity fixes a world direction, so roll and pitch are not symmetries. Ground effect depends on altitude, so vertical translation is excluded as well. The task symmetry is

$$G = \mathrm{SO}(2)_{\mathrm{yaw}} \ltimes \mathbb{R}^2_{xy}. \tag{4}$$

This distinguishes the symmetry of the task from the manifold structure of its state.

3 On-Manifold Equivariant Graph Dynamics

OM-EGNN represents each prediction window as a causal graph of measured history and future query nodes. Each layer aggregates receiver-local tangent updates and retracts them onto the pose manifold.

3.1 Manifold preliminaries

We manipulate rotations through local tangent vectors in $\mathbb{R}^{d_n}$, where $d_n = n(n-1)/2$. Following Solà et al. (2018) and Barfoot (2024), let $(\cdot)^\wedge$ and $(\cdot)^\vee$ denote the hat and vee maps and define

$$\mathrm{Exp}(\omega) = \exp(\omega^\wedge), \qquad \mathrm{Log}(R) = \log(R)^\vee, \qquad \omega \in \mathbb{R}^{d_n},\ R \in \mathrm{SO}(n). \tag{5}$$

The receiver-local relative rotation and right-composed update are

$$\omega_{R \to R'} = \mathrm{Log}(R^\top R'), \qquad R_{\mathrm{new}} = R\,\mathrm{Exp}(\delta\omega). \tag{6}$$

where Log uses the principal branch. The update remains in $\mathrm{SO}(n)$ by construction.

3.2 Receiver-local maps on the pose manifold

For $X = (p, R) \in \mathcal{M}_n$, identify $T_X\mathcal{M}_n \simeq \mathbb{R}^n \times \mathbb{R}^{d_n}$ and let $\tau = (\Delta p, \delta\omega)$. For $Y = (p', R')$, define the receiver-local difference and retraction

$$Y \ominus X = \begin{bmatrix} R^\top(p' - p) \\ \mathrm{Log}(R^\top R') \end{bmatrix}, \tag{7}$$

$$X \oplus \tau = \begin{bmatrix} p + R\,\Delta p \\ R\,\mathrm{Exp}(\delta\omega) \end{bmatrix}. \tag{8}$$

Both components of $Y \ominus X$ are expressed in the receiver frame, while $X \oplus \tau$ maps the increment back to the global frame. Translation and rotation are updated separately on $\mathbb{R}^n \times \mathrm{SO}(n)$.

For any common global left action g,

$$(g \cdot Y) \ominus (g \cdot X) = Y \ominus X, \qquad g \cdot (X \oplus \tau) = (g \cdot X) \oplus \tau. \tag{9}$$

Thus relative pose inputs are invariant and local increments produce equivariant pose updates; Appendix A gives the layer proof.

3.3 Causal history–query graph

Each prediction window is represented by H observed history nodes and K future query nodes. A history node at time $\ell \in \{t-H+1, \ldots, t\}$ contains the measured state (X_ℓ, v_ℓ) and control u_ℓ. The query at horizon k contains the candidate control u_{t+k} and is initialized from the last measured state, $X^0_{H+k} = X_t$ and $v^0_{H+k} = v_t$.

Information flows in temporal order: the prediction at $t+k$ may depend on the measured history and candidate controls $u_{t+1:t+k}$, but not on later controls. This enforces causality while producing all K states in one forward pass. Poses are maintained as geometric node states, whereas body-frame velocities and controls are encoded as invariant features h^0_i. These two channels are refined jointly by the on-manifold graph convolution described next; after L layers, the K query nodes represent the predicted trajectory.

3.4 On-manifold equivariant graph convolution

Starting from the causal graph above, layer ℓ propagates a message along each temporally admissible edge $j \to i$. The source pose is first expressed relative to the receiver, and the resulting local displacement is combined with the invariant node features:

$$\xi^\ell_{ij} = X^\ell_j \ominus X^\ell_i, \tag{10}$$

$$d^\ell_{ij} = \|\xi^\ell_{ij}\|_2^2, \tag{11}$$

$$m^\ell_{ij} = \phi^\ell_e\left(h^\ell_i, h^\ell_j, d^\ell_{ij}, \xi^\ell_{ij}\right). \tag{12}$$

Here, ϕ^ℓ_e is shared across edges, and d_{ij} is supplied as an additional invariant feature. With $C_i = 1/|\mathcal{N}(i)|$, the layer constructs a scalar-weighted displacement increment and a learned tangent residual,

$$c^\ell_i = C_i \sum_{j \in \mathcal{N}(i)} \xi^\ell_{ij}\, \alpha \tanh\big(\phi^\ell_x(m^\ell_{ij})\big), \tag{13}$$

$$r^\ell_i = \sum_{j \in \mathcal{N}(i)} \delta \tanh\big(\phi^\ell_\tau(m^\ell_{ij}, \xi^\ell_{ij})\big), \tag{14}$$

$$\nu^\ell_i = c^\ell_i + r^\ell_i, \tag{15}$$

Here, the MLP ϕ^ℓ_x returns one scalar per edge, whereas ϕ^ℓ_τ returns a full receiver-local tangent vector. In the reported models, ϕ^ℓ_τ is a single affine projection. The $\tanh$ acts elementwise, and δ bounds each residual component per edge ($\delta = 0.2$; $\alpha = 1$ in the planar model and $\alpha = \delta$ in the spatial model). The tangent increment is applied once through the manifold retraction, and invariant node features are updated by message aggregation:

$$X^{\ell+1}_i = X^\ell_i \oplus \nu^\ell_i, \tag{16}$$

$$h^{\ell+1}_i = \phi^\ell_h\left(h^\ell_i, \sum_{j \in \mathcal{N}(i)} m^\ell_{ij}\right). \tag{17}$$

By equation 9, $X^{\ell+1}_i$ is equivariant and $h^{\ell+1}_i$ remains invariant; Appendix A proves the composed layer map.

3.5 TANGENT RESIDUAL AND UPDATE EXPRESSIVITY

The displacement term c_i^ℓ is the direct on-manifold analogue of the EGNN coordinate update. Because ϕ_x returns one scalar per edge,

$$c_i^\ell \in \operatorname{span}\{\xi_{ij}^\ell : j \in \mathcal{N}(i)\}. \tag{18}$$

Changing Euclidean subtraction to $\ominus$ makes this update geometrically valid, but does not remove its directional restriction. The residual network ϕ_τ therefore returns a full receiver-local tangent vector rather than a scalar multiple of ξ_{ij}^ℓ. It can add directions outside equation 18 without breaking equivariance because its inputs and output coordinates are invariant under the common global left action.

The restriction is exact in a common case. With the copy-last pose prior, all query nodes start at the last measured pose. If the robot is at rest, all history poses coincide with it as well, so $\xi_{ij}^1 = 0$ on every edge and $c_i^1 = 0$. The poses are then unchanged, and by induction $c_i^\ell = 0$ in every layer: a displacement-only model returns the last pose for all K steps regardless of the candidate controls. The residual $\phi_\tau(m_{ij}, \xi_{ij})$ receives the controls through m_{ij} and can still produce a control-dependent increment, so it is required for the model to predict motion from rest.

The translational part of ν_i^ℓ is rotated by R_i^ℓ inside $\oplus$, while its rotational part is composed through Exp. The final map of ϕ_τ is zero-initialized, so every layer starts from the displacement update alone.

3.6 READOUT AND TRAINING OBJECTIVE

The final query poses are read directly from $X^L_{H+1:H+K}$, and a shared MLP decodes v from the corresponding invariant features. For batch element b and horizon step k, define $e^X_{bk} = \hat{X}_{bk} \ominus X_{bk}$ and $e^v_{bk} = \hat{v}_{bk} - v_{bk}$. The uniform direct-horizon objective is

$$\mathcal{L}(\theta) = \frac{1}{BK} \sum_{b=1}^{B} \sum_{k=1}^{K} \left(\lambda_X \|e^X_{bk}\|_2^2 + \lambda_v \|e^v_{bk}\|_2^2 \right), \tag{19}$$

with equal elementwise weighting in the reported experiments.

4 RELATED WORK

Learned dynamics for predictive control. Data-driven models have been used to complement analytic dynamics in agile ground and aerial control (Xiao et al., 2025; Torrente et al., 2021; Salzmann et al., 2023; Williams et al., 2018). Multi-step objectives can train a model for a planning-length prediction rather than only one-step fit (Mohajerin & Waslander, 2019; Asadi et al., 2019). AnyCar (Xiao et al., 2025) applies direct multi-step prediction to Ackermann-steering vehicles with a transformer over a canonicalized state–control history. We hold the direct-K protocol fixed across models and additionally evaluate the same planar checkpoints inside MPPI as a downstream test.

Equivariant geometric learning. Equivariant networks encode transformation structure in their computation (Finzi et al., 2020). EGNNs obtain $E(n)$ equivariance for Euclidean point coordinates through invariant messages and radial coordinate updates (Satorras et al., 2021); complete local frames improve directional expressivity (Du et al., 2022). Steerable models such as SE(3)-Transformers and SEGNN attach irreducible features to those points (Fuchs et al., 2020; Brandstetter et al., 2022), and EqMotion applies equivariant message passing to multi-agent trajectory forecasting (Xu et al., 2023). In these models the geometric state is a point in $\mathbb{R}^n$, whereas a robot pose also contains a rotation. LieConv and LieTransformer lift coordinates to functions on a Lie group and place equivariance in a regular-representation convolution or self-attention whose weights depend only on the relative group element, $\log(v^{-1}u)$ or $g^{-1}g'$ (Finzi et al., 2020; Hutchinson et al., 2021). The features are therefore functions on the lifted group. When the stabilizer is infinite, each location is replaced by finitely many lift samples. Even if a node is already a group element, so that the relative element carries the same information as $\ominus$, the layer still maps feature fields on G rather than moving a pose-valued state. We retain EGNN-style message passing, but each node stores its pose on $\mathbb{R}^n \times \mathrm{SO}(n)$. Neighbors are compared by $\ominus$, and $\oplus$ retracts a receiver-local tangent increment, including the residual of Section 3.5. The pose is updated in place, without lifting it into sampled copies on the group.

Table 1: Direct K prediction for the 2D Ackermann-steering task using an 80/10/10 random window split (five-run mean $\pm$ standard deviation). "Mean" averages over the twenty predicted steps; best values, including ties, are bold.

			Mean over horizon			Last step
Family	Model	Params	Pos. (m)	Rot. (rad)	Vel.	Pos. (m)
Canonical	MLP + canon.	304k	.039±.001	.052±.000	.310±.002	.060±.001
	RNN + canon.	228k	.100±.004	.079±.003	.286±.009	.174±.008
	LSTM + canon.	226k	.033±.001	.036±.001	.224±.006	.050±.002
	Transformer + canon.	212k	.042±.003	.086±.075	.309±.018	.072±.007
Graph	Generic GNN	224k	.051±.000	.042±.001	.219±.001	.072±.001
	$\mathbb{R}^2$-EGNN	233k	.053±.002	.034±.001	**.180±.009**	.098±.004
	$\mathbb{R}^3$-EGNN	233k	.047±.008	.036±.002	.184±.007	.082±.020
	ClofNet adapt.	232k	.162±.001	.035±.002	.188±.010	.362±.003
Ours	OM-EGNN ($-\phi_\tau$)	233k	.244±.002	.118±.001	.184±.007	.484±.003
	OM-EGNN	234k	**.026±.001**	**.033±.002**	.188±.012	**.047±.002**

Canonicalization. Canonicalization and frame averaging construct invariant or equivariant predictors (Puny et al., 2022; Kaba et al., 2023). For the present dynamics tasks, expressing all poses relative to the last measured pose removes the arbitrary global frame and is computationally cheap. Agent- and query-centric trajectory forecasters use the same idea by encoding scene elements in local reference frames (Zhou et al., 2022; 2023). We therefore treat canonicalization as a strong practical symmetry mechanism rather than a weak baseline.

5 Experiments

5.1 Questions and protocol

We evaluate three claims. First, direct-K prediction tests whether manifold-aware updates improve pose accuracy and whether the tangent residual is needed beyond geometric validity. The planar task predicts Ackermann-steering state from speed and steering in MuJoCo ($H{=}10$, $K{=}20$) (Srinivasa et al., 2019; Todorov et al., 2012; Xiao et al., 2025); the spatial task predicts position, attitude, and body velocity from motor commands in NeuroBEM the quadrotor dataset ($H{=}K{=}10$) (Bauersfeld et al., 2021). Canonicalized sequence models test symmetry by preprocessing, while Euclidean EGNN lifts retain the graph but not the manifold update; generic GNN and local-frame baselines (Du et al., 2022) broaden the comparison. The $-\phi_\tau$ ablation isolates the residual's contribution.

Second, a planar frame-transformation diagnostic tests whether receiver-local computation is unchanged by a common frame transformation, separating symmetry failure from predictive error. Third, matched MPPI with the same planar checkpoints tests downstream control; rest initialization also exposes the displacement-only model's zero-span failure. Models share data, objectives, and optimization, and uncertainty is across five seeds. Appendix C gives baseline realizations, data, splits, optimization, metrics, and controller details.

5.2 Direct multi-step prediction

Table 1 and Figure 2 show the planar result. Mean-over-horizon and last-step RMSE are reported separately, because a similar average can hide different terminal behavior. Mean-horizon position RMSE is clearly separated: 0.026 m versus 0.033 m for the closest canonical sequence model and 0.047 m for the best Euclidean EGNN lift. At the last step, OM-EGNN retains the best position result (0.047 m). Mean-horizon velocity is lowest for $\mathbb{R}^2$-EGNN. Thus the evidence supports a pose-accuracy claim, not uniform dominance of every state component.

Figure 2 shows that the position gap grows with the prediction horizon. Against canonicalized sequence models, OM-EGNN stays below LSTM+canon. at every step. Against graph baselines, Euclidean EGNN lifts accumulate position error faster.

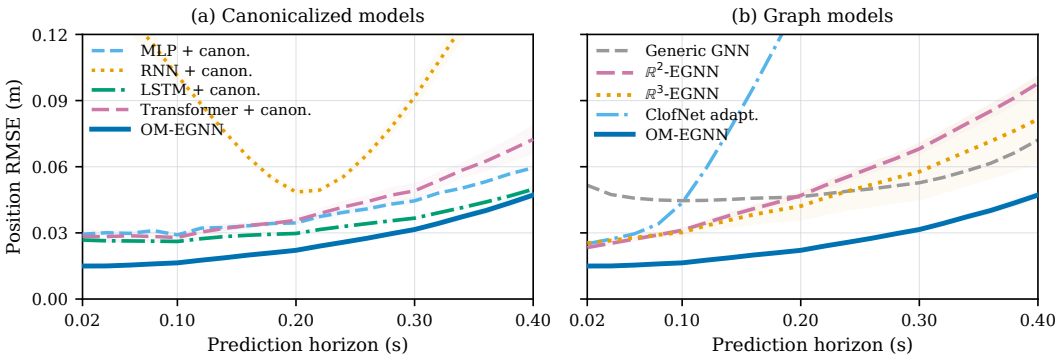


Figure 2: Horizon-wise position RMSE on the 2D Ackermann-steering task. Curves are five-run means; light bands show one standard deviation. Panel (a) compares canonicalized sequence models with OM-EGNN; panel (b) compares graph baselines with OM-EGNN. The vertical axis is clipped to the competitive range; ClofNet and late RNN steps leave this window.

Table 2: Direct K prediction on NeuroBEM (five-run mean $\pm$ standard deviation). Best values are bold.

			Mean over horizon			Last step
Family	Model	Params	Pos. (m)	Rot. (rad)	Vel.	Pos. (m)
Canonical	MLP + canon.	236k	.084±.042	**.006±.000**	.228±.059	.082±.039
	RNN + canon.	235k	.050±.003	.015±.000	.154±.013	.077±.003
	LSTM + canon.	236k	.090±.017	.015±.001	.280±.037	.079±.010
	Transformer + canon.	234k	.035±.006	.024±.020	.316±.027	.061±.009
Graph	Generic GNN	236k	.184±.012	.700±.083	.330±.014	.222±.014
	$\mathbb{R}^3$-EGNN	234k	.057±.006	.689±.266	.349±.055	.094±.018
	$\mathbb{R}^7$-EGNN	236k	.046±.021	.288±.368	.200±.093	.073±.031
	ClofNet adapt.	234k	.023±.002	.747±.115	.366±.011	.037±.005
Ours	OM-EGNN ($-\phi_\tau$)	234k	.066±.000	.022±.000	**.122±.009**	.136±.000
	OM-EGNN	236k	**.018±.002**	.024±.004	.125±.007	**.028±.002**

The $-\phi_\tau$ ablation remains geometrically valid but its last-step position error rises from 0.047 to 0.484 m. Moving the EGNN displacement update onto the manifold is therefore not sufficient. During motion the displacement span is not empty, so this open-loop error reflects both the span restriction and the bounded magnitude of the degree-normalized update, $\|c_i\| \leq \alpha \max_j \|\xi_{ij}\|$; the case in which the span alone is decisive appears in closed loop (Section 5.4).

The spatial result in Table 2 is directionally similar but less settled. OM-EGNN gives mean-horizon position RMSE 0.018 m. The $\mathbb{R}^3$-EGNN and ClofNet baselines include the raw quaternion in h and still decode attitude absolutely; their mean rotation errors stay at 0.689 and 0.747 radians. On the planar task, treating heading as a Euclidean coordinate remains competitive (Table 1, $\mathbb{R}^3$-EGNN rotation 0.036 rad). In 3D, the curvature of $\mathrm{SO}(3)$ dominates that approximation, and the rotation errors above are where the prediction breaks down. The Euclidean $\mathbb{R}^7$ lift remains unstable. Removing ϕ_τ again mainly harms position, while the canonical models remain competitive on rotation and velocity. We therefore describe the 3D result as evidence that pose geometry matters, not as a claim of universal superiority.

5.3 Reference-frame diagnostics

For the 2D Ackermann-steering task, we sample a global rotation uniformly in $[-\pi, \pi]$ and translations in $[-100, 100]^2$, transform history and targets together, and leave body-frame quantities unchanged.

Table 3: Frame-attack gap ϵ_{atk}, mean $\pm$ standard deviation. The quadrotor column samples $G = \mathrm{SO}(2)_{\text{yaw}} \ltimes \mathbb{R}^2_{xy}$. A negative entry is attacked error slightly below the clean error and is numerical noise at this scale. Canonicalized models are frame-consistent by construction.

Model	2D Ackermann steering	3D quadrotor dataset
Generic GNN	24.98 ± 5.78	51.89 ± 6.81
$\mathbb{R}^2$-EGNN	0.546 ± 0.112	–
$\mathbb{R}^3$-EGNN	0.224 ± 0.275	0.934 ± 0.407
$\mathbb{R}^7$-EGNN	–	0.308 ± 0.406
ClofNet adapt.	0.080 ± 0.032	0.805 ± 0.135
OM-EGNN ($-\phi_\tau$)	$(7.46 \pm 2.34)\,10^{-8}$	$(-1.05 \pm 4.10)\,10^{-6}$
OM-EGNN	$(2.05 \pm 3.05)\,10^{-7}$	$(-5.60 \pm 2.53)\,10^{-5}$

The frame-attack gap is the relative change in last-step total RMSE,

$$\epsilon_{\text{atk}} = \frac{\mathrm{RMSE}(f(g \cdot x),\, g \cdot y)}{\mathrm{RMSE}(f(x),\, y)} - 1. \tag{20}$$

The denominator is the clean prediction error, so the score does not shrink when the global translation is large. On the 3D quadrotor dataset the attack samples Equation 4: yaw uniform in $[-\pi, \pi]$ and horizontal translation uniform in $[-100, 100]^2$, with vertical translation held at zero. The planar column of Table 3 uses the checkpoints in Table 1; the quadrotor column uses those in Table 2. Only OM-EGNN and the $-\phi_\tau$ ablation keep this gap near zero on both tasks. Every other graph model is not equivariant, and the gap is substantially larger in 3D than in 2D. Removing ϕ_τ does not change this, so the ablation's large position error is an expressivity failure rather than a symmetry failure. Last-pose canonicalization is frame-consistent by construction and is therefore omitted from this architecture diagnostic.

5.4 Control performance evaluation

We evaluate the same planar $K{=}20$ checkpoints under one MPPI controller (Williams et al., 2016; 2018) ($B{=}256$, $\lambda{=}0.40$, steering cost 0.01, velocity cost 0.02), tuned jointly on MLP + canon. and OM-EGNN and then frozen for every model (Appendix C.2). Model training seed and MPPI random seed are paired as $\{0, 1, 2, 3, 4\}$. Each model is tested on three frictions and four feasible trajectories: circle, S-curve, and slalom at radius 2 m, and lemniscate at radius 4 m. Reported uncertainty is the sample standard deviation across five seed means after averaging the twelve environment conditions within each seed.

Table 4 shows that OM-EGNN has the highest mean progress (0.887 ± 0.021), success ($65.0 \pm 7.0\%$), and the lowest mean tracking RMSE (0.167 ± 0.027 m). The progress margin over MLP + canon. and the tracking margin over RNN + canon. both lie inside one standard deviation. Canonicalized sequence models otherwise outperform the non-equivariant graph baselines on both completion and tracking: MLP, Transformer, LSTM, and RNN reach progress 0.794–0.883 and success 18.3–61.7%, whereas ClofNet, $\mathbb{R}^2$-EGNN, $\mathbb{R}^3$-EGNN, and the generic GNN remain at progress 0.512–0.707 and success 1.7–16.7%. These graph models do not enforce the frame symmetry of Table 3, so a closed-loop state can leave the relative-pose distribution seen in training. Figure 3 shows the resulting departures: the Euclidean lifts leave the circle and the lemniscate, which cuts both progress and tracking accuracy. OM-EGNN avoids this failure because its pose update is equivariant on $\mathbb{R}^n \times \mathrm{SO}(n)$. Separately, $-\phi_\tau$ almost never moves. This stall is the degenerate case of Section 3.5: every trial starts from rest, so $-\phi_\tau$ predicts the same stationary trajectory for every sampled control sequence, the MPPI costs carry no information about the controls, and the vehicle stays at rest.

6 Discussion and Limitations

The evidence points to two architectural effects and one design equivalence. First, manifold structure matters most clearly for pose accuracy and frame consistency: Euclidean lifts increase last-step error under the frame attack, whereas OM-EGNN and the $-\phi_\tau$ ablation stay near zero. Component-wise

Table 4: Matched-seed MPPI on feasible 2D trajectories (five-paired-seed mean ± sample standard deviation). Success denotes final nearest-path progress ≥ 0.9; stalled $-\phi_\tau$ RMSE is marked †.

Model	Progress ↑	Success ↑	Track. RMSE ↓	Speed
OM-EGNN	**.887±.021**	**65.0±7.0%**	**.167±.027**	1.636±.049
MLP + canon.	.883±.021	61.7±11.2%	.175±.023	1.607±.031
Transformer + canon.	.851±.039	46.7±17.3%	.318±.117	**1.695±.077**
LSTM + canon.	.841±.056	38.3±19.2%	.276±.160	1.585±.121
RNN + canon.	.794±.054	18.3±13.7%	.169±.030	1.449±.107
ClofNet adapt.	.707±.063	11.7±13.9%	.311±.209	1.357±.217
$\mathbb{R}^2$-EGNN	.703±.045	10.0±9.1%	.342±.122	1.399±.108
$\mathbb{R}^3$-EGNN	.685±.083	16.7±8.3%	1.182±.919	1.561±.078
Generic GNN	.512±.030	1.7±3.7%	.802±.320	1.233±.152
OM-EGNN ($-\phi_\tau$)	.004±.007	0.0±0.0%	.062±.039†	.026±.009

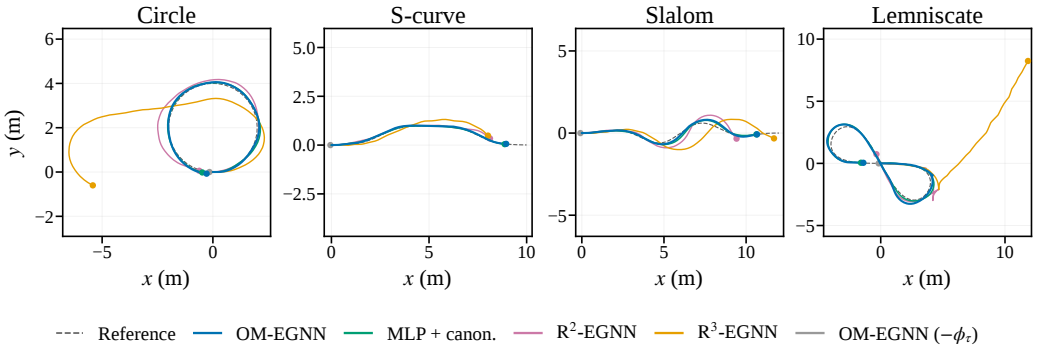


Figure 3: Representative matched-seed MPPI bird's-eye traces at μ=0.5 (paired seed 1, the seed among $\{0, 1, 2\}$ closest to the five-seed mean progress). Each panel overlays five models on one feasible trajectory. Dashed curves are the references. Euclidean lifts leave the circle or lemniscate, while $-\phi_\tau$ remains at the start.

results also show that position, rotation, and velocity can favor different models. Second, the tangent residual addresses an expressivity limitation that is separate from symmetry. Third, architectural equivariance and canonicalization are both valid symmetry mechanisms for these tasks, so symmetry alone does not explain the differences among models.

The MPPI study supports a carefully qualified control claim: under a shared controller, OM-EGNN is the best average performer on feasible trajectories, but the margin over strong canonicalized predictors is mixed across conditions and does not follow the open-loop component-wise error rankings. We therefore do not claim that lower generic K-step prediction error implies better control.

Two limitations remain. Ground effect excludes vertical translation from the quadrotor symmetry, so Table 3 reports Equation 4 rather than a larger translation group. A full $\mathrm{SE}(3)$ attack is outside this submission. We do not evaluate on a physical robot. We also do not run MPPI on the 3D quadrotor, because that task uses the public quadrotor dataset(NeuroBEM) rather than a simulator.

7 Conclusion

We presented OM-EGNN, a direct multi-step robot dynamics model that updates pose-valued graph nodes on $\mathbb{R}^n \times \mathrm{SO}(n)$. Controlled 2D and 3D comparisons show that product-manifold updates and an expressive tangent residual improve pose prediction and reference-frame consistency over Euclidean EGNN lifts, while canonicalization remains a competitive practical alternative. The same planar checkpoints remain competitive under matched MPPI control on feasible trajectories, although

open-loop prediction rankings do not directly determine closed-loop control rankings. The results support treating manifold structure, problem symmetry, and update expressivity as distinct design choices in learned robot dynamics.

Reproducibility Statement

Appendices A–C specify the equivariance argument, architectures, data, optimization, metrics, seeds, reference-frame diagnostic, and matched MPPI protocol. The supplementary material contains the training and evaluation code for the models in Tables 1–3, the three MuSHR traces used for the planar task, and the checkpoints with per-seed evaluation files for those tables. The spatial task uses the public NeuroBEM logs (Bauersfeld et al., 2021). Closed-loop results reuse the same planar checkpoints inside MPPI (Williams et al., 2016; 2018) under Appendix C.2. That controller follows the cited method, so its source is not included.

LLM Usage Disclosure

In this work, we used generative AI tools (Cursor and Codex) to implement methods, including the experimental code, and to help organize and phrase reported experimental results. We have not used generative AI tools to generate synthetic datasets, develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proofs, assist in the writing of proofs, propose or refine hypotheses, design the research methodology or experimental protocol, assist with translation, or clean and reformat datasets. Qualitative and thematic data analysis is not applicable to this work. Additionally, we used these tools to edit the English text of the paper for readability and correctness. We have reviewed all AI-assisted work. The theory, proofs, and experimental protocol were written by the authors. Generated code was checked and tested by the authors, and AI-edited prose and the organization of reported results were verified against the experimental records. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

References

Kavosh Asadi, Dipendra Misra, Seungchan Kim, and Michael L. Littman. Combating the compounding-error problem with a multi-step model. *arXiv preprint arXiv:1905.13320*, 2019.

Timothy D. Barfoot. *State Estimation for Robotics*. Cambridge University Press, 2024.

Leonard Bauersfeld, Elia Kaufmann, Philipp Foehn, Sihao Sun, and Davide Scaramuzza. NeuroBEM: Hybrid aerodynamic quadrotor model. *Robotics: Science and Systems*, 2021.

Johannes Brandstetter, Rob Hesselink, Elise van der Pol, Erik J. Bekkers, and Max Welling. Geometric and physical quantities improve E(3) equivariant message passing. In *International Conference on Learning Representations*, 2022.

Weitao Du, He Zhang, Yuanqi Du, Qimin Meng, Wei Chen, Nanning Zheng, Bin Shao, and Tie-Yan Liu. SE(3) equivariant graph neural networks with complete local frames. In *International Conference on Machine Learning*, pp. 5583–5608, 2022.

Marc Finzi, Samuel Stanton, Pavel Izmailov, and Andrew Gordon Wilson. Generalizing convolutional neural networks for equivariance to Lie groups on arbitrary continuous data. In *International Conference on Machine Learning*, pp. 3165–3176, 2020.

Fabian B. Fuchs, Daniel E. Worrall, Volker Fischer, and Max Welling. SE(3)-transformers: 3D roto-translation equivariant attention networks. In *Advances in Neural Information Processing Systems*, 2020.

Michael J. Hutchinson, Charline Le Lan, Sheheryar Zaidi, Emilien Dupont, Yee Whye Teh, and Hyunjik Kim. LieTransformer: Equivariant self-attention for Lie groups. In *International Conference on Machine Learning*, pp. 4533–4543, 2021.

Sékou-Oumar Kaba, Arnab Kumar Mondal, Yan Zhang, Yoshua Bengio, and Siamak Ravanbakhsh. Equivariance with learned canonicalization functions. In *International Conference on Machine Learning*, pp. 15546–15566, 2023.

Wook Hyun Kwon and Soohee Han. *Receding horizon control: model predictive control for state models*. Springer, 2005.

Nima Mohajerin and Steven L. Waslander. Multistep prediction of dynamic systems with recurrent neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 30(11):3370–3383, 2019.

Omri Puny, Matan Atzmon, Edward J. Smith, Ishan Misra, Aditya Grover, Heli Ben-Hamu, and Yaron Lipman. Frame averaging for invariant and equivariant network design. In *International Conference on Learning Representations*, 2022.

Tim Salzmann, Elia Kaufmann, Jon Arrizabalaga, Marco Pavone, Davide Scaramuzza, and Markus Ryll. Real-time neural MPC: Deep learning model predictive control for quadrotors and agile robotic platforms. *IEEE Robotics and Automation Letters*, 8(4):2397–2404, 2023.

Víctor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E(n) equivariant graph neural networks. In *International Conference on Machine Learning*, pp. 9323–9332, 2021.

Joan Solà, Jérémie Deray, and Dinesh Atchuthan. A micro Lie theory for state estimation in robotics. *arXiv preprint arXiv:1812.01537*, 2018.

Siddhartha S. Srinivasa, Patrick Lancaster, Johan Michalove, Matt Schmittle, Colin Summers, Matthew Rockett, Rosario Scalise, Joshua R. Smith, Sanjiban Choudhury, Christoforos Mavrogiannis, and Fereshteh Sadeghi. MuSHR: A low-cost, open-source robotic racecar for education and research. *arXiv preprint arXiv:1908.08031*, 2019.

Emanuel Todorov, Tom Erez, and Yuval Tassa. MuJoCo: A physics engine for model-based control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5026–5033, 2012.

Guillem Torrente, Elia Kaufmann, Philipp Föhn, and Davide Scaramuzza. Data-driven MPC for quadrotors. *IEEE Robotics and Automation Letters*, 6(2):3769–3776, 2021.

Grady Williams, Paul Drews, Brian Goldfain, James M. Rehg, and Evangelos A. Theodorou. Aggressive driving with model predictive path integral control. In *IEEE International Conference on Robotics and Automation*, pp. 1433–1440, 2016.

Grady Williams, Paul Drews, Brian Goldfain, James M. Rehg, and Evangelos A. Theodorou. Information-theoretic model predictive control: Theory and applications to autonomous driving. *IEEE Transactions on Robotics*, 34(6):1603–1622, 2018.

Wenli Xiao, Haoru Xue, Tony Tao, Dvij Kalaria, John M. Dolan, and Guanya Shi. Anycar to anywhere: Learning universal dynamics model for agile and adaptive mobility. In *IEEE International Conference on Robotics and Automation*, pp. 8819–8825, 2025.

Chenxin Xu, Robby T. Tan, Yuhong Tan, Siheng Chen, Yu Guang Wang, Xinchao Wang, and Yanfeng Wang. EqMotion: Equivariant multi-agent motion prediction with invariant interaction reasoning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1410–1420, 2023.

Zikang Zhou, Luyao Ye, Jianping Wang, Kui Wu, and Kejie Lu. HiVT: Hierarchical vector transformer for multi-agent motion prediction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8823–8833, 2022.

Zikang Zhou, Jianping Wang, Yung-Hui Li, and Yu-Kai Huang. Query-centric trajectory prediction. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17863–17873, 2023.

A Equivariance of the Product-Manifold Layer

Let $g = (R_g, t_g)$ act as in Equation 2. For $X = (p, R)$ and $Y = (p', R')$,

$$(g \cdot Y) \ominus (g \cdot X) = \begin{bmatrix} (R_g R)^\top \big[(R_g p' + t_g) - (R_g p + t_g)\big] \\ \mathrm{Log}\big((R_g R)^\top R_g R'\big) \end{bmatrix} = \begin{bmatrix} R^\top (p' - p) \\ \mathrm{Log}(R^\top R') \end{bmatrix} = Y \ominus X. \tag{21}$$

Similarly, for $\tau = (\Delta p, \delta\omega)$,

$$g \cdot (X \oplus \tau) = \begin{bmatrix} R_g(p + R\,\Delta p) + t_g \\ R_g R\,\mathrm{Exp}(\delta\omega) \end{bmatrix} = \begin{bmatrix} (R_g p + t_g) + (R_g R)\,\Delta p \\ (R_g R)\,\mathrm{Exp}(\delta\omega) \end{bmatrix} = (g \cdot X) \oplus \tau. \tag{22}$$

Thus ξ_{ij} and every message input in Equation 12 are invariant when h_i is invariant. The learned scalar coefficient and tangent residual are therefore unchanged in receiver-local coordinates. Their sum ν_i is unchanged, and Equation 22 implies $g \cdot (X_i \oplus \nu_i) = (g \cdot X_i) \oplus \nu_i$. The feature update is invariant because it depends only on invariant quantities. By induction over layers, OM-EGCL is equivariant to the common global left action.

This statement applies to the planar model under $\mathrm{SE}(2)$. For the spatial model, $g^{\mathrm{body}} = R^\top g^{\mathrm{world}}$ is invariant under translation and under rotations that preserve g^{world}. With vertical gravity those rotations are yaw. Ground effect depends on altitude, so vertical translation is not a symmetry of the quadrotor task, and the claim is restricted to Equation 4. The layer identity holds on this group. It does not give full $\mathrm{SE}(3)$ equivariance under fixed gravity.

B Architecture Details

Planar model. Each graph has $N = 30$ nodes. A shared $5 \to 64 \to 128$ ReLU encoder maps body velocity and control to invariant features. The reported model has two single-channel OM-EGCL layers of width 128. The edge MLP is $260 \to 128 \to 128$ with ReLU activations; its input concatenates the two 128-D node features, the 3-D local displacement ξ_{ij}, and d_{ij}. The scalar displacement branch ϕ_x is $128 \to 64 \to 1$ followed by $\tanh$ ($\alpha = 1$), with Xavier initialization; the feature update is $256 \to 128 \to 128$. The tangent residual concatenates the 128-D message and 3-D local displacement before a $131 \to 3$ map, bounded by $\delta \tanh$ with $\delta = 0.2$ and zero-initialized. A $128 \to 64 \to 3$ MLP decodes body velocity. The resulting model has 233,501 parameters.

Spatial model. State is (p, q, v, ω) with q in scalar-first convention. The node encoder receives six body-velocity components, four motor commands, and three body-frame gravity components. Two width-128 OM-EGCL layers use a six-dimensional local tangent. The edge MLP input is $2 \cdot 128 + 1 + 6 = 263$. Both ϕ_x ($\alpha = \delta = 0.2$) and ϕ_τ ($\delta = 0.2$) have zero-initialized final layers. Future query poses and velocities copy the last measured pose and body velocity before message passing. A shared decoder predicts the six body-velocity components. The model has 235,548 parameters. The $\mathrm{SO}(3)$ Log uses a near-π numerical guard; quaternions are normalized after every update.

Causal mask. With nodes ordered as history followed by future queries, the receiver–source mask is $A_{ij} = 1[j \le i]$. Self edges contribute feature messages but zero relative pose. The normalizer in the displacement update is receiver in-degree. The reported residual branch sums bounded edge residuals; degree-mean aggregation was tested but did not improve the selected planar model.

C Experimental Details

Comparison protocol and baselines. Unless noted otherwise, models within each task use the same data, prediction objective, and optimization protocol. Each of five run seeds controls model initialization and the random window split. Planar graph models have 224k–234k parameters; the

direct-K MLP has 304k because its output head grows with the horizon. Spatial models are matched near 235.5k parameters.

The Euclidean EGNN variants retain the $H + K$ graph, causal mask, query construction, feature widths, and unconstrained residual branch of OM-EGNN. In 2D, $\mathbb{R}^2$-EGNN updates (x, y) and keeps heading in invariant features, whereas $\mathbb{R}^3$-EGNN treats (x, y, θ) as Euclidean coordinates. In 3D, $\mathbb{R}^3$-EGNN updates position and decodes an absolute quaternion from features. The reported $\mathbb{R}^3$-EGNN and ClofNet rows also concatenate the raw quaternion into those features. $\mathbb{R}^7$-EGNN treats position and quaternion as ambient coordinates and renormalizes the decoded quaternion. We also evaluate a generic GNN and a task-adapted complete local-frame model (Du et al., 2022).

Canonicalized sequence models. Let $A = X_t$ be the last measured pose. Each history pose is replaced by $A^{-1} \cdot X_i$, the predictor outputs the future block in that local frame, and the block is mapped back by A; body-frame controls remain unchanged. This wrapper removes the global pose gauge for the present tasks. Transformer + canon. follows the AnyCar dynamics model (Xiao et al., 2025); MLP, RNN, and LSTM use the same wrapper.

Task-specific realizations and ablation. The planar model represents R_i by heading θ_i, wraps pose differences, and re-wraps updates to $(-\pi, \pi]$. The spatial model stores unit quaternions, maps relative rotation through the $\mathrm{SO}(3)$ Log, and applies each tangent update with the $\mathrm{SO}(3)$ Exp followed by quaternion renormalization. Its invariant features include body-frame gravity $g_i^{\mathrm{body}} = R_i^\top g^{\mathrm{world}}$. Gravity removes roll and pitch from the symmetry, and ground effect removes vertical translation, leaving Equation 4 rather than full $\mathrm{SE}(3)$. The $OM - EGNN(-\phi_\tau)$ ablation removes the learned tangent residual while retaining the product-manifold displacement update.

2D Ackermann-steering robot data. Following AnyCar (Xiao et al., 2025), we simulate a MuSHR car (Srinivasa et al., 2019) in MuJoCo (Todorov et al., 2012). State is $(x, y, \theta, v_x, v_y, \omega)$, control is (speed, steering), and Gaussian observation noise has $\sigma = 0.015$. The dataset contains 30 minutes from three MPPI-driven traces at friction coefficients $\{1.0, 0.5, 0.2\}$ with floor contact priority enabled. At 50 Hz, stride-one windowing with H=10 and K=20 gives 89,866 examples, randomly split 80/10/10 into 71,892/8,986/8,988 samples for each run seed. Because the partition is applied after window construction, adjacent windows may occur in different subsets. Inputs concatenate ten history state–control tuples and twenty future controls; targets contain twenty six-dimensional future states.

2D Ackermann-steering robot optimization. We use Adam with initial learning rate 10^{-3}, ReduceLROnPlateau factor 0.5, patience 3, minimum learning rate 10^{-6}, gradient-norm clipping at 1.0, at most 400 epochs, and early-stopping patience 10. Batch size is 1024 for graph models and 2048 for canonical sequence models. Main planar statistics, including Table 3, use seeds $\{0, 1, 2, 3, 4\}$.

NeuroBEM data and optimization. We use all 247 recorded CSV trajectories from NeuroBEM (Bauersfeld et al., 2021). State comprises 3D position, a unit quaternion, and six body-frame velocity components; action is the four-motor command. The comparison uses $H = K = 10$, 120 epochs with patience 20, batch size 1024, Adam learning rate 10^{-4}, and five seeds. The models are size-matched to approximately 235.5k parameters. All trajectories are used at their native rate of about 400 Hz without resampling, so H=K=10 covers about 25 ms each. Valid windows from all trajectories are concatenated and randomly split 80/10/10 for each run seed.

Prediction metrics. For each component family we compute RMSE over examples, reporting both the average across K forecast steps and the terminal step. Planar rotation uses a wrapped heading error. Spatial rotation uses the norm of the $\mathrm{SO}(3)$ logarithm.

Transformation diagnostics. The frame-attack gap is Equation 20, the relative change in last-step total RMSE. For every planar batch, rotation is sampled uniformly from $[-\pi, \pi]$ and translation uniformly from $[-100, 100]^2$. Pose inputs and targets receive the same transform; body-frame state and controls are unchanged. Those rows use the five checkpoints of Table 1. For every quadrotor batch, yaw is sampled uniformly from $[-\pi, \pi]$ and horizontal translation uniformly from $[-100, 100]^2$, with vertical translation fixed at zero (Equation 4). Those rows use the five checkpoints of Table 2.

C.1 TRAJECTORY-DISJOINT FRICTION HOLDOUTS

The main planar split randomly partitions windows from all three friction traces. To test unseen friction, we retrain on trajectory-disjoint friction files: train $\mu \in \{1.0, 0.5\}$ and test μ=0.2, or train $\mu \in \{0.2, 0.5\}$ and test μ=1.0. Validation windows are drawn from the training files. All ten models are trained with five seeds on both holdouts (Table 5). Canonicalized Transformer and MLP remain competitive on both holdouts. OM-EGNN has the lowest mean last-step position error on both held-out frictions, but the overlap in run variation and the limited number of source trajectories do not support a broad out-of-distribution superiority claim. We do not report holdout frame-attack gaps as evidence. Closed-loop MPPI on held-out friction is not included.

Table 5: Trajectory-disjoint friction holdouts (H=10, K=20; five-run mean $\pm$ standard deviation). Entries are last-step position RMSE.

Model	n	$1.0{+}0.5 \to 0.2$ Pos. last (m)	$0.2{+}0.5 \to 1.0$ Pos. last (m)
MLP + canon.	5	.124±.005	.074±.001
Transformer + canon.	5	.113±.032	.082±.008
LSTM + canon.	5	.329±.217	.113±.060
RNN + canon.	5	.245±.022	.191±.024
$\mathbb{R}^2$-EGNN	5	.155±.003	.129±.002
$\mathbb{R}^3$-EGNN	5	.135±.024	.119±.015
ClofNet adapt.	5	.359±.000	.355±.015
Generic GNN	5	.285±.008	.241±.015
OM-EGNN ($-\phi_\tau$)	5	.483±.002	.484±.001
OM-EGNN	5	**.099±.046**	**.062±.015**

C.2 MPPI PROTOCOL

Every model uses the MPPI controller (Williams et al., 2016; 2018) in Table 6. History and horizon match the planar direct-K models. The path cursor follows the nearest waypoint, and the speed command is capped at 3 m/s. The setting in Table 6 was tuned while evaluating MLP + canon. and OM-EGNN together, then frozen for every model.

Table 6: Shared MPPI controller. The same setting is used for every model.

Setting	Value
Samples B	256
Temperature λ	0.40
Position cost (x, y)	$[10,\ 40]$
Heading cost	0
Steering cost	0.01
Velocity cost	0.02
History / horizon	H=10, K=20
Control period	20 ms (50 Hz)
Speed cap	3 m/s

Each paired seed is evaluated on four references and three floor frictions, twelve conditions in all. Circle is a closed loop of radius 2 m. The S-curve and the slalom use the same radius: the S-curve reverses curvature once, and the slalom repeats lateral reversals. The lemniscate is a figure-eight of radius 4 m, length 22.065 m, and maximum curvature $0.934\,\mathrm{m}^{-1}$, below the approximate steering limit $1.33\,\mathrm{m}^{-1}$. A radius-2 lemniscate reaches $1.798\,\mathrm{m}^{-1}$ and is omitted from this aggregate. The friction coefficients are $\mu \in \{0.2, 0.5, 1.0\}$, the same three floor values as the planar training traces. Each reference is run at all three frictions. Table 4 averages the twelve conditions inside each seed, then reports the mean and sample standard deviation across the five paired seeds.

At 50 Hz the control period is 20 ms. Control computation time stayed under 20 ms for every model.

The component-wise open-loop rankings in Table 1 do not directly match the closed-loop progress ranking, so prediction accuracy and control utility are reported separately. We do not report frame-transformed closed-loop MPPI; Table 3 remains an open-loop network diagnostic.

D Additional Planar Ablations

Table 7 shows the selected OM-EGNN model at seed 0. Position error saturates for histories longer than five steps, while terminal pose error generally grows with prediction horizon.

Table 7: Planar OM-EGNN last-step RMSE, seed 0. Left: history sweep at $K = 10$. Right: horizon sweep at $H = 10$.

H	Pos.	Rot.	Vel.	K	Pos.	Rot.	Vel.
1	.035	.049	.260	1	.012	.009	.170
3	.029	.044	.248	5	.015	.019	.211
5	.022	.034	.240	10	.024	.038	.224
10	.024	.038	.224	15	.035	.054	.230
15	.025	.038	.218	20	.041	.055	.213
20	.023	.031	.206	–	–	–	–

Depth and width. Table 8 varies the number of single-channel OM-EGCL layers L and the hidden width h at seed 0. History and horizon were fixed arbitrarily at $H{=}10$ and $K{=}10$. One layer is worse on every component. Position changes little once $L{\geq}2$ and $h{\geq}128$, while velocity continues to fall as width and depth increase.

Table 8: Planar OM-EGNN last-step RMSE, seed 0. Depth L and width h with history and horizon fixed at $H{=}10$ and $K{=}10$.

L	h	Pos.	Rot.	Vel.
1	64	.036	.048	.280
1	128	.032	.046	.270
1	256	.030	.045	.250
2	64	.027	.042	.243
2	128	.024	.038	.224
2	256	.023	.034	.189
3	64	.026	.039	.240
3	128	.024	.036	.218
3	256	.023	.030	.180

Per-trajectory MPPI progress. Table 9 reports the five-seed per-trajectory progress means. OM-EGNN, MLP, and Transformer remain close on the feasible lemniscate, while $\mathbb{R}^3$-EGNN leads only on slalom and fails to transfer that advantage to circle or lemniscate.

Table 9: Per-trajectory MPPI progress on the feasible suite (five-paired-seed mean ± standard deviation).

Model	Circle $r{=}2$	S-curve $r{=}2$	Slalom $r{=}2$	Lemniscate $r{=}4$
OM-EGNN	$.975{\pm}.017$	$.904{\pm}.003$	$.788{\pm}.022$	$.879{\pm}.070$
MLP + canon.	$.956{\pm}.011$	$.904{\pm}.003$	$.802{\pm}.050$	$.869{\pm}.034$
Transformer + canon.	$.929{\pm}.068$	$.891{\pm}.012$	$.752{\pm}.068$	$.830{\pm}.085$
LSTM + canon.	$.902{\pm}.117$	$.866{\pm}.034$	$.800{\pm}.064$	$.796{\pm}.059$
$\mathbb{R}^2$-EGNN	$.793{\pm}.056$	$.802{\pm}.099$	$.699{\pm}.082$	$.517{\pm}.187$
$\mathbb{R}^3$-EGNN	$.578{\pm}.228$	$.852{\pm}.045$	$.855{\pm}.149$	$.458{\pm}.189$