HydroWrench: A Dataset and Benchmark for Dynamic Load Reconstruction in Bioinspired Aquatic Mechanisms

Anonymous authors
Paper under double-blind review

Abstract

Dynamic loads shape the propulsive performance and structural response of bioinspired aquatic mechanisms, making their reconstruction important for surrogate modeling and motion optimization. We introduce HydroWrench, a dataset and benchmark of 116,756 command–load trajectories from caudal-fin, webbed-foot, and flipper mechanisms representative of added-mass, drag-based, and lift-based propulsion. The dataset provides recorded joint-angle commands and derived kinematics as inputs, with measured six-axis interface loads as targets. With executed motion and flow unobserved, the benchmark compares nonlinear representation and additional command history under a shared evaluation protocol, with models fitted separately for each mechanism. A multilayer perceptron reduces normalized mean squared error by 25.5–81.0% relative to linear regression with matched features. Additional history improves time-varying reconstruction for the webbed foot and flipper, while caudal-fin model rankings depend on the temporal scoring grid. In the webbed-foot high-frequency holdout, a long-history predictor reduces error by 15.1% relative to its matched local-input counterpart, while its error remains 32.3% higher than under in-distribution conditions. These results connect reconstruction accuracy to temporal error structure and motion-condition coverage. HydroWrench provides a shared experimental basis for load-relevant representations, dynamic surrogate modeling, and input–output system identification in partially observed fluid–mechanism systems.

1 Introduction

Bioinspired aquatic mechanisms exchange momentum with the surrounding fluid through oscillating fins, paddling feet, and flapping appendages (Fish & Lauder, 2006; Smits, 2019). Their motion produces forces and moments whose mean levels and fluctuations depend on amplitude, frequency, joint coordination, and geometry (Triantafyllou et al., 2005; Izraelevitz & Triantafyllou, 2014). Learning this motion–load relation provides a computational interface for mechanism simulation, motion optimization, and system identification. It also supports load estimation and control-oriented models for robotic modules.

Physical models and experiments describe how unsteady motion modifies propulsive loads (Smits, 2019; Izraelevitz & Triantafyllou, 2014), and learned force surrogates have supported soft-fin policy training (Hamamatsu et al., 2025). Extending such models across motion conditions requires reconstructing both average loads and time-varying responses. The difficulty arises from a coupled, partially observed system: actuator dynamics, mechanical transmission, and evolving flow connect the recorded command to the measured interface load. Commands describe intended motion, while executed angles and the surrounding flow may remain unobserved. Local kinematic features and earlier commands consequently provide different information for reconstruction. State-space identification and memory-enhanced physical prediction offer ways to use temporal observations (Schoukens & Ljung, 2019; Beintema et al., 2021; Buitrago Ruiz et al., 2025). This motivates comparing expressive local predictors with models that receive additional motion history.

Scientific machine learning resources support comparisons on simulated partial differential equations (PDEs), flow fields, and diverse physical dynamics (Takamoto et al., 2022; Bonnet et al., 2022;

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.

Ohana et al., 2024). RealPDEBench also provides experimental measurements and corresponding simulations (Hu et al., 2026). For controllable bioinspired mechanisms, systematic comparison requires real load responses across motion conditions, together with motion histories and acquisition identities. These records connect an experimentally accessible input–output interface to questions of model expressivity, temporal information, and generalization.

We introduce **HydroWrench**, a measured dataset and reconstruction benchmark spanning a one-degree-of-freedom (1-DOF) caudal fin, a 2-DOF webbed foot, and a 3-DOF flipper. Its 116,756 trajectories pair recorded commands and derived kinematics with six-axis interface loads. Models are trained and evaluated separately for each mechanism. We compare linear and nonlinear predictors with matched local features, then examine how additional command history affects reconstruction. Error decomposition distinguishes improvements in mean load levels from those in time-varying waveforms. A grouped high-frequency holdout and nested training subsets assess generalization across motion conditions and the effect of training-data quantity.

Our contributions are:

1. We introduce a measured command-to-six-axis-interface-load dataset spanning three bioinspired mechanisms. Complete trajectories and acquisition identities provide a common experimental resource for training and evaluating models of dynamic response under incomplete motion observations.
2. We establish a unified benchmark for comparing nonlinear representation and motion-history modeling through acquisition-group splits, matched temporal support, and training-only normalization. Endpoint and waveform-level analyses characterize reconstruction quality, while nested training subsets and grouped frequency holdouts assess data quantity and motion-condition generalization.
3. We characterize the gains and limitations of the tested reconstruction models. Nonlinear local predictors improve on matched linear references across all three mechanisms, while history gains vary with the response features, temporal scoring support, and motion conditions. These findings provide empirical reference points for developing and evaluating dynamic surrogate models.

2 Related Work

Bioinspired motion and dynamic loads. Oscillatory propulsion involves added-mass and circulatory effects, while paddling feet can generate drag-based propulsion (Smits, 2019; Ribak et al., 2010). Flapping-foil experiments further show how coordinated motions modify mean lift, thrust, and oscillatory loading (Izraelevitz & Triantafyllou, 2014). These studies motivate our selection of caudal-fin oscillation emphasizing added-mass effects, drag-oriented webbed-foot paddling, and lift-oriented flipper motion. The mechanisms provide representative geometries and kinematic structures for studying measured dynamic loads; their responses contain interacting physical contributions. HydroWrench compares reconstruction of the aggregate interface response across these three experimental systems.

Nonlinear representation and identification. Nonlinear system identification learns dynamic input–output relations, with latent-state models organizing past observations into predictive states (Schoukens & Ljung, 2019; Beintema et al., 2021; Chen et al., 2025). Long short-term memory networks (LSTMs), temporal convolutional networks (TCNs), and selective state-space models aggregate history through different structures (Hochreiter & Schmidhuber, 1997; Bai et al., 2018; Dao & Gu, 2024). Attention and extended LSTM (xLSTM) models provide further temporal and cross-variable representations (Liu et al., 2025; 2024; Beck et al., 2024). Laplace neural operators (LNOs) and lifted linear dynamics introduce structured temporal models (Cao et al., 2024; Korda & Mezić, 2018). Their finite-window adaptations permit comparison at a common command-to-load interface. Geometry-based features, physics-informed learning, and sparse identification offer additional directions for exploiting structural knowledge (Karniadakis et al., 2021; Brunton et al., 2016).

Scientific machine learning benchmarks. PDEBench and The Well provide simulated field trajectories for learning physical evolution, while AirfRANS supplies geometry-dependent aerodynamic fields (Takamoto et al., 2022; Ohana et al., 2024; Bonnet et al., 2022). Their inputs and targets

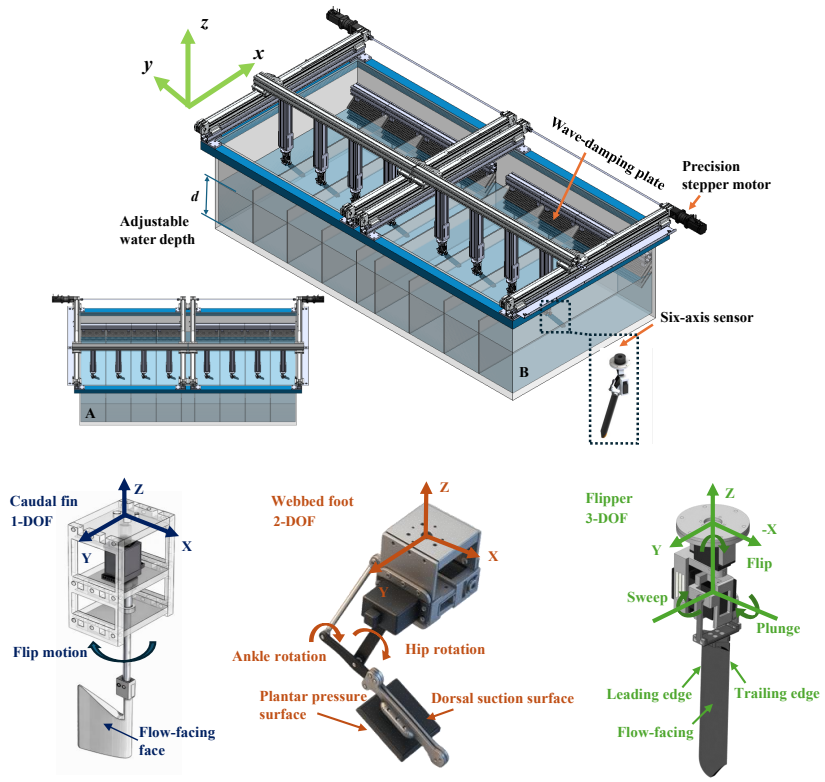

Figure 1: Experimental platform and mechanism motions, adapted from REEF (Han et al., 2026). Top: parallel towing lanes and interface-load sensing. Bottom, left to right: a caudal fin driven by one rotational joint, a webbed foot with coupled hip and ankle rotations, and a flipper with three serial rotational joints. Arrows indicate the motion directions and local coordinate axes. The benchmark links recorded joint commands to the six force/moment channels measured at the mechanism interface.

support learning spatial solutions or field transitions. RealPDEBench connects experimental observations with simulation-based physical modeling (Hu et al., 2026). HydroWrench adds a command-to-interface-load task: the accessible inputs are motion commands and derived kinematics, and the outputs are measured force and moment time series. Acquisition-group splits and matched temporal scoring support comparisons of representation, additional history, and motion-condition generalization at this observation interface.

3 Dataset and Observation Interface

3.1 Experimental source and measured quantities

Measurements originate from the REEF experimental platform (Han et al., 2026), which combines parallel mechanism lanes and six-axis sensing (Figure 1). The current acquisitions use a towing speed of 0.167 m/s. A trial identifies one mechanism, acquisition group, and lane. The three domains differ in geometry and motion distribution as well as degrees of freedom; each defines a separate learning problem with independently fitted models.

The inputs are recorded joint-angle commands q_{cmd} and their numerically derived velocities and accelerations. The targets are the stored filtered interface loads $(F_x, F_y, F_z, T_x, T_y, T_z)$, in N and N m. The sensor is zeroed after motion stops and the flow stabilizes for each trial; movement-induced mechanical inertia remains in the recorded response. Independent executed-angle and flow-state measurements are unavailable. Models therefore reconstruct the measured response of the

coupled actuation–mechanism–fluid system. All six channels enter fitting and model selection; per-channel dimensional errors retain their physical units.

The 1-DOF caudal fin uses a NACA 0012 foil undergoing single-axis oscillation. The 2-DOF webbed foot coordinates hip and ankle rotations through a four-bar linkage, and the 3-DOF flipper uses a NACA 0016 foil driven by three serial rotational joints (Figure 1). Dimensions and labeled kinematic diagrams are provided in Appendix A.

3.2 Data preparation and inventory

Data preparation checks record validity, restricts evaluation to intervals supported by both commands and sensor records, and applies a channel-wise extreme-amplitude screen. The amplitude thresholds are three times the reference 99.9th percentiles, with the reference population and audit chronology specified in Appendix A. The retained benchmark contains **116,756 trials** (Table 1). All methods within a mechanism use the same retained records and evaluation times. Inputs and targets are standardized using training-set statistics; reported normalized errors use fixed channel-wise training variances. The stored filtered loads are used without additional target filtering.

Table 1: Current benchmark inventory. All lanes belonging to an acquisition group remain in one split. Counts refer to individual trials.

Domain	Train	Validation	Test	Total
1-DOF caudal fin	19,190	2,879	1,920	23,989
2-DOF webbed foot	26,201	3,932	2,634	32,767
3-DOF flipper	48,000	7,200	4,800	60,000

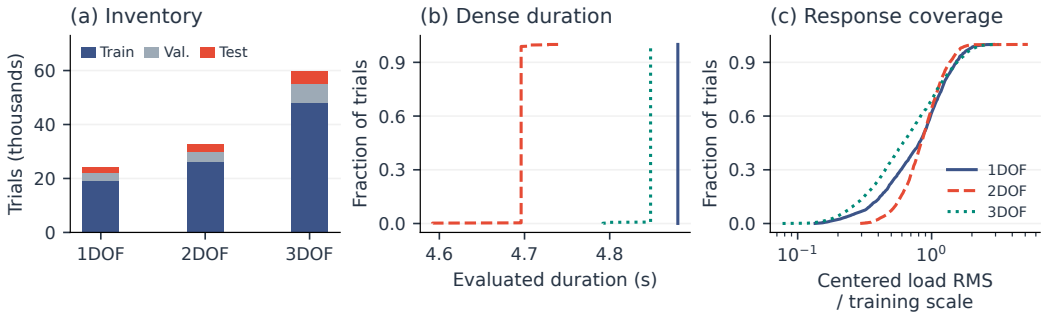

Figure 2: Dataset inventory and response coverage. Duration refers to the evaluated dense interval. Centered root mean square (RMS) uses training scales and equal trial weights; curves describe response distributions.

Figure 2 summarizes the inventory and measured test-load fluctuations, linking the number of trajectories to their evaluated duration and response range. Acquisition identities and complete records support subsequent studies of condition coverage, sequence models, and history lengths. **Data access.** Example trajectories are available in the anonymous repository. The complete dataset, split identities, and preprocessing, training, and evaluation code will be publicly released upon acceptance.

4 Benchmark Protocol

4.1 Temporal support and information settings

Commands use nominal 20-Hz timing because individual command timestamps are unavailable. First and second derivatives are estimated directly on this native grid by a five-point, third-degree Savitzky–Golay polynomial (SG5). Commands and both derivatives are then linearly interpolated to a 125-Hz analysis grid. Sensor loads are interpolated from their recorded timestamps to the same

Table 2: Input-window definitions: sample count W and first-to-last span $(W-1)/125$ seconds. W1 uses local derivative features; its zero span denotes no additional feature history.

Mechanism	W1	Short	W63	Long
1-DOF caudal fin	1 / 0	7 / 0.048	63 / 0.496	140 / 1.112
2-DOF webbed foot	1 / 0	7 / 0.048	63 / 0.496	157 / 1.248
3-DOF flipper	1 / 0	19 / 0.144	63 / 0.496	138 / 1.096

grid, preserving the sensor time origin. Thus, 125 Hz specifies the aligned model inputs and targets; recorded commands retain their nominal 20-Hz information rate. The controller's execution timing is separate from this offline analysis grid.

With $x_t = [q_{\mathrm{cmd}}, \dot{q}_{\mathrm{cmd}}, \ddot{q}_{\mathrm{cmd}}]_t$, the supervised task is $\hat{y}_t = f_\theta(x_{t-W+1:t})$, where $y_t \in \mathbb{R}^6$. Here t is an analysis-grid index, $x_t \in \mathbb{R}^{3d}$ contains features for d commanded joints, and θ denotes fitted model parameters. The window length W counts analysis-grid samples. W1 uses a single kinematic feature vector, whose derivatives already contain local temporal information; longer windows add feature history. W1 and W63 contain 1 and 63 samples, respectively, for every mechanism, whereas the short and long windows use mechanism-specific lengths (Table 2). The centered derivative estimator uses neighboring command samples on both sides of each interior time point, so the benchmark evaluates offline reconstruction. All windows are evaluated on the same valid interval supported by the longest window. Persistence, autoregressive (AR), and autoregressive with exogenous input (ARX) models receive measured load history and are reported separately in Appendix B.

4.2 Models, fitting, and scoring

Ridge regression compares successive additions of derivative features; a finite impulse response (FIR) model supplies a linear history reference. A multilayer perceptron (MLP) maps one kinematic feature vector to six endpoint loads. LSTM compares four windows (W1, short, W63, long), and TCN three (W1, short, long). The remaining sequence and operator-inspired models use the long windows. Appendix B specifies their endpoint regression adaptations.

Input normalization and target scales use only the retained training subset. Validation selects regularization and neural checkpoints. Neural results report three seeds under a common recipe: Adam, learning rate 0.001, batch size 128, dropout 0.21, gradient clipping at 1, and a 60-epoch cap with validation-based early stopping. Compute use and optimization progress vary across architectures. Some TCN runs reach the cap; a completed three-seed extension is reported separately from the main comparison in Appendix D.

For N evaluated endpoints, normalized mean squared error (nMSE) is

$$\mathrm{nMSE} = \frac{1}{6N} \sum_{j=1}^{6} \sum_{i=1}^{N} \frac{(\hat{y}_{ij} - y_{ij})^2}{\sigma^2_{j,\mathrm{train}}}.$$

Here i indexes evaluated endpoints, j indexes the six load channels, and y_{ij} and $\hat{y}_{ij}$ are the measured and predicted loads. The quantity $\sigma^2_{j,\mathrm{train}}$ is the training-endpoint variance of load channel j, fixed across models within each mechanism. Variance normalization makes force and moment channels comparable and gives them equal weight in the six-axis mean. The primary score uses prescribed endpoints; dense evaluation scores every 8 ms on the same valid interval to describe waveforms and sampling sensitivity. Counts are tabulated in Appendix C. Both scores use the same error formula and scales. Neural results report sample standard deviation (SD) across three random seeds; selected diagnostics also resample acquisition groups to assess group-composition variability. Displayed errors are rounded to four decimal places; relative changes are computed before rounding.

4.3 Explanatory experiments

For condition generalization, a fixed upper-5% command-frequency rule excludes groups containing high-frequency trials from training and validation. Original test groups supply separate in-

Table 3: Motion-only test nMSE (lower is better). Neural entries: mean ± sample SD over three seeds; bold: lowest mean per mechanism across all rows. All neural models and FIR use $(q, \dot{q}, \ddot{q})$. Window W counts analysis samples; Short/Long lengths appear in Table 2. FIR samples history every four grid steps. † denotes a nonzero SD smaller than 0.0001.

Model	Window W	1-DOF caudal fin	2-DOF webbed foot	3-DOF flipper
(a) Linear references				
Training mean	—	1.0419	1.0260	1.0320
Ridge q	1	0.9604	0.9868	0.8587
Ridge $q, \dot{q}$	1	0.5814	0.7706	0.6552
Ridge $q, \dot{q}, \ddot{q}$	1	0.5789	0.7435	0.6414
FIR	Long	0.5655	0.7096	0.6273
(b) Local nonlinear models				
MLP	1	0.4315 ± 0.0004	0.4022 ± 0.0004	0.1216 ± 0.0003
LSTM	1	0.4292 ± 0.0003	0.3910 ± 0.0005	0.1118 ± 0.0002
TCN	1	0.4318 ± 0.0002	0.3964 ± 0.0003	0.1134 ± 0.0002
(c) History-input models				
LSTM	Short	0.4281 ± 0.0001	0.3661 ± 0.0008	0.1080 ± 0.0008
LSTM	63	**0.4220** ± 0.0006	0.3442 ± 0.0006	0.1070 ± 0.0006
LSTM	Long	0.4221 ± †	**0.3390** ± 0.0005	**0.1060** ± 0.0005
TCN	Short	0.4314 ± 0.0005	0.3765 ± 0.0008	0.1093 ± 0.0002
TCN	Long	0.4281 ± 0.0003	0.3576 ± 0.0026	0.1083 ± 0.0003
Mamba-2	Long	0.4327 ± 0.0015	0.3468 ± 0.0026	0.1142 ± 0.0018
TimeBridge	Long	0.4371 ± 0.0016	0.4290 ± 0.0232	0.1285 ± 0.0035
iTransformer	Long	0.4339 ± 0.0019	0.4235 ± 0.0066	0.1282 ± 0.0045
xLSTM	Long	0.4352 ± 0.0011	0.3565 ± 0.0009	0.1178 ± 0.0029
LNO	Long	0.4515 ± 0.0007	0.4254 ± 0.0005	0.1660 ± 0.0002
KoopmanIO	Long	0.4280 ± †	0.3612 ± 0.0002	0.1094 ± 0.0011

distribution (ID) and out-of-distribution (OOD) subsets, with the latter containing held-out high-frequency trials. Groups with missing parameters are excluded. Each fitted model is evaluated on both subsets. Appendix E specifies the reference population, grouping, and normalization conventions.

For data quantity, webbed-foot LSTMs at W1/W157 use nested 25%/50% acquisition-group subsets and the existing 100% runs. Sampling seeds 101/202/303 pair with initialization seeds 0/1/2, with identical selected groups across windows. Selected groups retain all valid trials; validation/test remain fixed. The three replicates quantify joint sampling and training variation. Finally, W63 fills the context-length gap, and a three-seed TCN cap extension from 60 to 120 epochs provides a test of optimization-budget sensitivity. The primary benchmark remains fixed across these diagnostics.

5 Dynamic Load Reconstruction

5.1 Nonlinear representation and context length

Table 3 groups linear references, local nonlinear models, and history-input models. With matched command-angle, velocity, and acceleration features, MLP reduces endpoint nMSE relative to Ridge by 25.5% for the 1-DOF caudal fin, 45.9% for the 2-DOF webbed foot, and 81.0% for the 3-DOF flipper (Figure 3). FIR improves on local Ridge, while retaining higher error than MLP.

Extending LSTM from W1 to the long window reduces endpoint nMSE by 1.7% for the caudal fin, 13.3% for the webbed foot, and 5.2% for the flipper. TCN also improves in each mechanism. The intermediate W63 window captures much of the observed gain; extending it further reduces webbed-foot and flipper errors by 1.5% and 0.9%, while the caudal-fin mean changes by less than 0.03%. The incremental benefit therefore varies across the tested contexts and mechanism-specific data distributions.

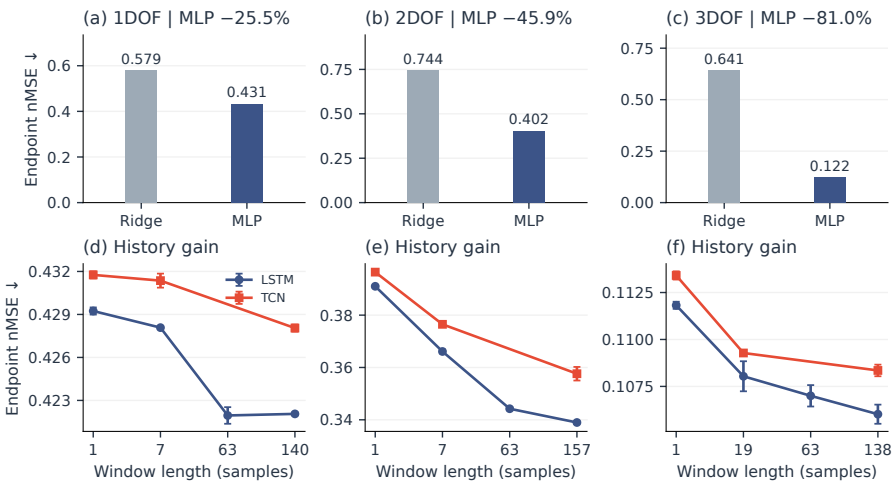

Figure 3: Nonlinear representation and history. Top: Ridge and MLP share the same input features; bars show their nMSE. Bottom: LSTM at four windows and TCN at three, with matched endpoints. Error bars are sample SD over three seeds; vertical scales differ across domains.

5.2 Reconstruction across temporal model classes

At matched long-window inputs, LSTM has the lowest endpoint mean among the tested temporal models in each mechanism. For the 2-DOF webbed foot, Mamba-2 scores 0.3468, between TCN (0.3576) and LSTM (0.3390); its paired three-seed improvement interval over TCN includes zero (Appendix C). Attention, matrix-memory, LNO, and controlled latent-linear adaptations broaden the comparison but retain higher endpoint means under this recipe. A three-seed TCN budget extension reduces long-window webbed-foot error by 1.1% on average (Appendix D). These results describe the implemented adaptations and optimization settings, and motivate joint study of temporal structure and optimization.

5.3 Waveform fidelity and temporal evaluation support

Figure 4 relates history gains to reconstructed load features. For each trial and axis, let $\bar{e}$ be the mean prediction error. Mean squared error decomposes exactly into $\bar{e}^2$ plus the mean squared difference between separately centered predicted and true waveforms. We aggregate these terms using point counts and the same training scales. The second term measures centered-waveform error, combining amplitude, shape, and temporal-alignment differences.

On the dense grid, LSTM history reduces webbed-foot and flipper errors by 13.3% and 5.2%. Reductions in centered-waveform error account for 99.0% and 93.7% of these gains. TCN shows the same pattern: total reductions of 9.8% and 4.5%, with 96.4% and 90.0% contributed by the centered term. For these models and mechanisms, additional motion history primarily improves fluctuations around the trial mean. The decomposition quantifies this contribution; the illustrative traces show the corresponding time-domain response.

For the caudal fin, long-window LSTM improves the prescribed endpoint score, whereas dense nMSE rises from W1's 0.4335 to 0.4382. Long-window TCN obtains 0.4332 on the dense grid, reversing its endpoint ordering with LSTM. Selecting the registered endpoints from the same predictions recovers the original ranking (Appendix C). The error formula is unchanged: the difference concerns the time points receiving weight in the score. Reporting both supports makes the temporal extent of an improvement explicit.

5.4 Generalization to held-out motion conditions

Figure 5 separates the ID and OOD results of the same fitted models. For the 2-DOF webbed foot, long-window endpoint nMSE rises from 0.3304 on ID to 0.4373 on OOD, a 32.3% increase.

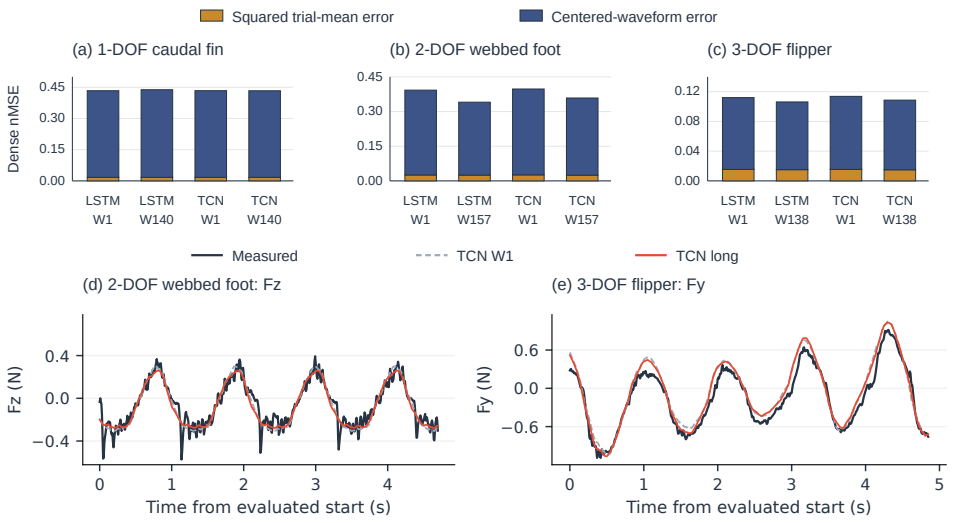

Figure 4: Temporal components of reconstruction error. Top: dense nMSE decomposed into squared trial-mean error and centered-waveform error, averaged over three seeds. Bottom: 2-DOF webbed foot F_z and 3-DOF flipper F_y, with seed-0 TCN predictions. Cases are selected by the median target-only normalized fluctuation magnitude. W1 and long windows are defined in Table 2; aggregate scores include all eligible trials.

OOD W1 scores 0.5150, so history still reduces error by 15.1%. Paired-seed and acquisition-group intervals support a positive history gain alongside the ID–OOD gap (Appendix E). History remains useful at held-out frequencies while reconstruction remains less accurate than within the training-supported range.

For the caudal fin and flipper, mean OOD history reductions are 1.4% and 2.0%, with paired three-seed intervals spanning zero. The flipper's mean reduction is smaller than its 4.9% ID reduction. These runs provide weaker evidence for a repeatable OOD history benefit than the webbed-foot runs; the small seed sample leaves the size of that benefit uncertain.

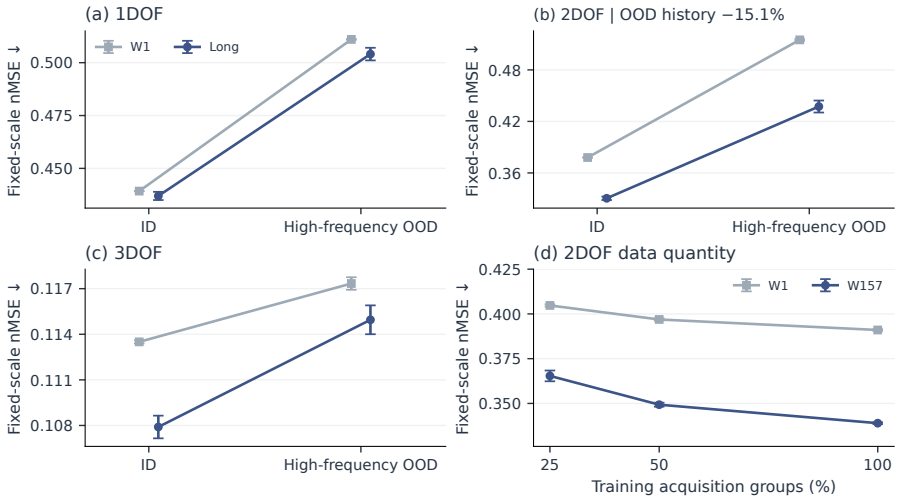

Figure 5: Condition support and data quantity. (a–c) Separate ID/OOD scores from each frequency-holdout LSTM. (d) Nested acquisition-group subsets, reusing 100% runs. All use fixed main-training scales; error bars show SD over three paired replicates. OOD and ID comprise different groups.

Dense webbed-foot errors locate the OOD deterioration in time-varying response. For long LSTM, squared trial-mean error falls from 0.0198 (ID) to 0.0120 (OOD), while centered-waveform error rises from 0.3107 to 0.4231. Centered-RMS amplitude error also falls slightly, from 0.0864 to 0.0837; the remaining correlation-related discrepancy increases (Appendix E). Thus an average-load or amplitude-only evaluation would miss the harder aspect of this extrapolation. This decomposition locates the dominant predictive difficulty in waveform fidelity.

5.5 Training-group coverage and predictive accuracy

For webbed-foot W157, increasing the retained training groups from 25% to 50% to 100% reduces test nMSE from 0.3654 to 0.3493 to 0.3390: gains of 4.4% and 3.0%. W1 gains 1.9% and 1.5%. Data quantity remains relevant, especially for learning temporal representations, although doubling the larger subset yields a modest gain. Under the shared epoch cap, larger subsets also entail more parameter updates. These curves describe this combined training setting; the separate frequency holdout probes generalization beyond the retained motion range.

6 Discussion and Limitations

The measured command–load interface supports studying partially observed dynamics at a common observation budget. Local nonlinear predictors provide a reference for assessing history, while waveform analysis locates the response features that benefit. LSTM's competitive results motivate examining architecture and optimization together. Aggregate error, temporal decomposition, and condition-specific evaluation characterize complementary aspects of reconstruction within a common squared-error definition; sequence-level objectives offer a direction for improving temporal fidelity.

Data quantity and condition coverage pose distinct generalization challenges. Webbed-foot history gains persist at held-out frequencies alongside a substantial ID–OOD gap, showing that useful temporal context does not by itself ensure robustness to distribution shift. Larger training subsets improve accuracy, but these learning curves do not establish improved extrapolation. Since the webbed-foot OOD deterioration is concentrated in waveform fidelity, generalization should also be assessed through response shape and timing rather than only mean load or amplitude.

Mechanism geometry and motion parameters provide structural information for gray-box surrogates, physics-informed learning, and sparse identification (Karniadakis et al., 2021; Brunton et al., 2016). Latent-state learning can investigate compact representations of command history (Beintema et al., 2021; Chen et al., 2025). The present scope is offline reconstruction at a fixed towing speed, using nominal command timing and locally two-sided derivatives. Hidden dynamics include execution, structure, and flow; predictive accuracy alone does not identify their separate physical contributions (Schoukens & Ljung, 2019). Post hoc curation, scoring-grid sensitivity, and optimization diagnostics are documented in the appendices. Causal preprocessing and target-device validation would support subsequent online applications.

7 Conclusions

HydroWrench provides measured motion–six-axis-interface-load trajectories and a unified reconstruction benchmark for three bioinspired aquatic mechanisms. Its command-derived inputs, acquisition identities, and common evaluation protocol support comparisons of nonlinear representation, additional motion history, and condition generalization. Under the tested settings, nonlinear local prediction improves on linear regression, and history gains for the webbed foot and flipper primarily concern the time-varying response.

The additional analyses locate remaining difficulties: caudal-fin comparisons depend on scoring times, and webbed-foot high-frequency reconstruction retains a gap despite useful history. Larger training subsets improve webbed-foot accuracy under the common epoch-based recipe. Together, these results provide reference observations for studying load-relevant latent representations, geometry-aware features, and physical priors. The dataset and protocol offer a shared experimental basis for dynamic surrogate modeling and input–output system identification in partially observed fluid–mechanism systems.

AI USE STATEMENT

Generative AI tools were used for limited tasks, including data organization, preliminary data checks, helping debug parts of the analysis code, and improving manuscript readability. The authors reviewed all AI-assisted outputs and take responsibility for this work.

REFERENCES

Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. arXiv preprint arXiv:1803.01271, 2018. URL https://arxiv.org/abs/1803.01271.

Maximilian Beck, Korbinian Pöppel, Markus Spanring, Andreas Auer, Oleksandra Prudnikova, Michael Kopp, Günter Klambauer, Johannes Brandstetter, and Sepp Hochreiter. xlstm: Extended long short-term memory. *Advances in Neural Information Processing Systems*, 37:107547–107603, 2024.

Gerben Beintema, Roland Toth, and Maarten Schoukens. Nonlinear state-space identification using deep encoder networks. In *Proceedings of the 3rd Conference on Learning for Dynamics and Control*, volume 144 of *Proceedings of Machine Learning Research*, pp. 241–250, 2021. URL https://proceedings.mlr.press/v144/beintema21a.html.

Florent Bonnet, Jocelyn Ahmed Mazari, Paola Cinnella, and Patrick Gallinari. AirfRANS: High fidelity computational fluid dynamics dataset for approximating reynolds-averaged navier–stokes solutions. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/94ab7b23a345f93333eac8748a66c763-Paper-Datasets_and_Benchmarks.pdf.

Steven L. Brunton, Joshua L. Proctor, and J. Nathan Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences*, 113(15):3932–3937, 2016. doi: 10.1073/pnas.1517384113.

Ricardo Buitrago Ruiz, Tanya Marwah, Albert Gu, and Andrej Risteski. On the benefits of memory for modeling time-dependent PDEs. In *International Conference on Learning Representations*, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/hash/89379d5fc6eb34ff98488202fb52b9d0-Abstract-Conference.html.

Qianying Cao, Somdatta Goswami, and George Em Karniadakis. Laplace neural operator for solving differential equations. *Nature Machine Intelligence*, 6:631–640, 2024. doi: 10.1038/s42256-024-00844-4.

Chuanqi Chen, Nan Chen, Yinling Zhang, and Jin-Long Wu. CGKN: A deep learning framework for modeling complex dynamical systems and efficient data assimilation. *Journal of Computational Physics*, 532:113950, 2025. doi: 10.1016/j.jcp.2025.113950.

Tri Dao and Albert Gu. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 10041–10071, 2024. URL https://proceedings.mlr.press/v235/dao24a.html.

Frank E. Fish and George V. Lauder. Passive and active flow control by swimming fishes and mammals. *Annual Review of Fluid Mechanics*, 38:193–224, 2006. doi: 10.1146/annurev.fluid.38.050304.092201.

Yuya Hamamatsu, Pavlo Kupyn, Roza Gkliva, Asko Ristolainen, and Maarja Kruusmaa. Underwater soft fin flapping motion with deep neural network based surrogate model. In *2025 IEEE 8th International Conference on Soft Robotics (RoboSoft)*, pp. 1–6. IEEE, 2025.

Fei Han, Xinyu Cui, Zhipeng Wang, Ning Yang, Hang Xu, Haifeng Zhang, Zhongming Hu, Jun Wang, Junfeng Du, and Dixia Fan. Embracing flow unsteadiness: A high-throughput learning platform enables vortex-exploiting bioinspired propulsion. arXiv preprint arXiv:2609.14254, 2026. URL https://arxiv.org/abs/2609.14254. Preprint, version 1.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8): 1735–1780, 1997. doi: 10.1162/neco.1997.9.8.1735.

Peiyan Hu, Haodong Feng, Hongyuan Liu, Tongtong Yan, Wenhao Deng, Tianrun Gao, Rong Zheng, Haoren Zheng, Chenglei Yu, Chuanrui Wang, Kaiwen Li, Zhi-Ming Ma, Dezhi Zhou, Xingcai Lu, Dixia Fan, and Tailin Wu. RealPDEBench: A benchmark for complex physical systems with real-world data. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=y3oHMcoItR.

Jacob S. Izraelevitz and Michael S. Triantafyllou. Adding in-line motion and model-based optimization offers exceptional force control authority in flapping foils. *Journal of Fluid Mechanics*, 742: 5–34, 2014. doi: 10.1017/jfm.2014.7.

George Em Karniadakis, Ioannis G. Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3:422–440, 2021. doi: 10.1038/s42254-021-00314-5.

Milan Korda and Igor Mezić. Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control. *Automatica*, 93:149–160, 2018. doi: 10.1016/j.automatica.2018.03.046.

Peiyuan Liu, Beiliang Wu, Yifan Hu, Naiqi Li, Tao Dai, Jigang Bao, and Shu-Tao Xia. TimeBridge: Non-stationarity matters for long-term time series forecasting. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 39815–39840, 2025. URL https://proceedings.mlr.press/v267/liu25cb.html.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. iTransformer: Inverted transformers are effective for time series forecasting. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2310.06625.

Ruben Ohana, Michael McCabe, Lucas Meyer, Rudy Morel, Fruzsina J. Agocs, Miguel Beneitez, Marsha Berger, Blakesley Burkhart, Stuart B. Dalziel, Drummond B. Fielding, Daniel Fortunato, Jared A. Goldberg, Keiya Hirashima, Yan-Fei Jiang, Rich R. Kerswell, Suryanarayana Maddu, Jonah Miller, Payel Mukhopadhyay, Stefan S. Nixon, Jeff Shen, Romain Watteaux, Bruno Régaldo-Saint Blancard, François Rozet, Liam H. Parker, Miles Cranmer, and Shirley Ho. The well: a large-scale collection of diverse physics simulations for machine learning. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://proceedings.nips.cc/paper_files/paper/2024/hash/4f9a5acd91ac76569f2fe291b1f4772b-Abstract-Datasets_and_Benchmarks_Track.html.

Gal Ribak, John G. Swallow, and David R. Jones. Drag-based 'hovering' in ducks: The hydrodynamics and energetic cost of bottom feeding. *PLOS ONE*, 5(9):e12565, 2010. doi: 10.1371/journal.pone.0012565.

Johan Schoukens and Lennart Ljung. Nonlinear system identification: A user-oriented road map. *IEEE Control Systems Magazine*, 39(6):28–99, 2019.

Alexander J. Smits. Undulatory and oscillatory swimming. *Journal of Fluid Mechanics*, 874:P1, 2019. doi: 10.1017/jfm.2019.284.

Makoto Takamoto, Timothy Praditia, Raphael Leiteritz, Daniel MacKinlay, Francesco Alesiani, Dirk Pflüger, and Mathias Niepert. PDEBench: An extensive benchmark for scientific machine learning. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/0a9747136d411fb83f0cf81820d44afb-Abstract-Datasets_and_Benchmarks.html.

Michael S. Triantafyllou, Franz S. Hover, Alexandra H. Techet, and Dick K. P. Yue. Review of hydrodynamic scaling laws in aquatic locomotion and fishlike swimming. *Applied Mechanics Reviews*, 58(4):226–237, 2005. doi: 10.1115/1.1943433.

A DATA PREPARATION AND MECHANISM GEOMETRY

Valid records and amplitude screening. Evaluation uses the shared valid interval of motion and sensor records, with support for the longest input window. The audit checks finite values, timestamp order, time origin, and endpoint identities. For each channel and lane, the amplitude threshold is three times the 99.9th percentile of trial-wise absolute peaks among previously valid records from other lanes of the same mechanism. A trial is excluded if any channel exceeds its threshold. Reference populations, fixed thresholds, and retained identities accompany the data. The rule was developed in a post hoc audit involving training, validation, and test records. All reported models use the common retained data and training-derived normalization.

Sampling scope. Linear interpolation aligns recorded loads with the analysis grid. The audited webbed-foot records include timestamp gaps up to 0.174 s; the amplitude screen does not exclude records solely on gap duration, so interpolation can bridge such intervals. Original timestamps, retained identities, and normalization scales support alternative preprocessing studies. Mean squared error and mean absolute error are also retained per channel in dimensional units.

Kinematic geometry. Table 4 summarizes the principal dimensions. The caudal-fin rotation axis is at one-quarter chord from the leading edge. For the webbed-foot linkage, let O be the common drive origin and A, B, C the linkage points. The collinear extension from C ends at reference point Q, with $CQ = 116$ mm. The 60 mm plate is centered at M, where $CM = 62$ mm. With unit vector $\mathbf{u} = (C - B)/\|C - B\|$ directed from C toward Q, the reference points satisfy $Q = C + 116\mathbf{u}$ and $M = C + 62\mathbf{u}$ in millimeters; the plate endpoints are $M \pm 30\mathbf{u}$. Figure 6 defines these points and dimensions.

Table 4: Principal mechanism dimensions. Lengths are in millimeters.

Mechanism	Geometry
1-DOF caudal fin	NACA 0012; chord 100; span 110; pivot 25 from leading edge
2-DOF webbed foot	Plate length 60; $OA = 35$, $AB = 115$, $BC = 35$, $OC = 125$; $CQ = 116$, $CM = 62$
3-DOF flipper	NACA 0016; chord 45; span 179; three serial rotation axes

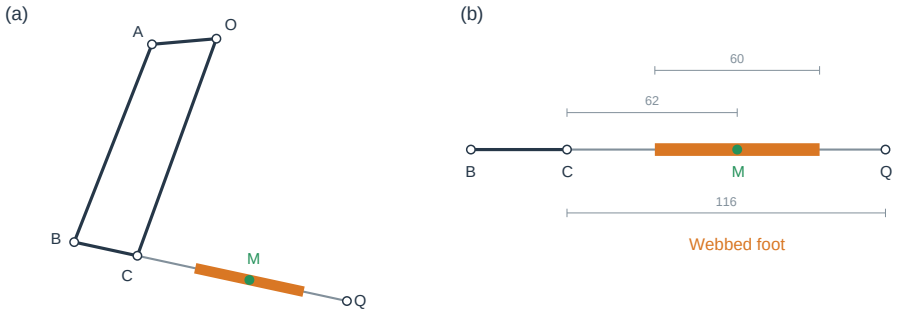

Figure 6: Two-degree-of-freedom webbed-foot geometry. (a) Example four-bar pose: O is the common drive origin; A, B, and C are linkage points, with $OA = 35$, $AB = 115$, $BC = 35$, and $OC = 125$ mm. (b) Collinear extension and plate dimensions. Dark segments denote linkage bars, gray denotes the extension to reference point Q, orange denotes the 60 mm plate, and green marks its center M. The reference distances are $CQ = 116$ and $CM = 62$ mm. All dimensions are in millimeters; the pose is shown in an arbitrary drawing orientation.

The flipper uses three serial rotation axes. Figure 7 and Table 5 specify their reference geometry in the computer-aided design (CAD) frame. The points J_1, J_2, J_3 lie on the successive joint axes; R and T denote root and tip references. Initial axis directions are $+Y$, $+X$, and $+Z$, respectively, with downstream geometry moving under the preceding rotations. The root-to-tip distance is reported as 179 mm. These coordinates define the kinematic reference geometry; measured loads retain the sensor-coordinate convention.



Figure 7: Three-degree-of-freedom flipper reference geometry, projected from CAD coordinates. J_1, J_2, and J_3 mark points on the serial rotation axes; colored arrows indicate their initial directions. R and T are root and tip references. The orange segment indicates the span direction, with the NACA 0016 foil having a 45 mm chord and 179 mm span. Coordinates appear in Table 5.

Table 5: Flipper reference points in the CAD frame (mm). Directions refer to the initial joint axes.

Point	X	Y	Z	Axis direction
J_1	-17.58	16.74	9.03	$+Y$
J_2	5.19	72.60	9.05	$+X$
J_3	-11.96	72.10	-20.35	$+Z$
R	-12.46	113.31	4.32	—
T	-18.09	292.76	3.13	—

B Model and Execution Details

Table 6 groups the reference methods by the representations they use and summarizes their implementations.

All motion-only networks accept a batch of standardized command-derived windows and output six standardized loads at the endpoint. For d commanded axes, the input has $3d$ channels. Neural input statistics use valid training-grid samples, and target statistics use training endpoints. Models are fitted separately per mechanism. Ridge regularization is selected using validation data; training means, variances, and all fitted weights exclude validation and test samples. Detailed configurations and identities are preserved in the run manifests.

Linear references. The training-mean predictor outputs the six channel means of the training endpoints. Ridge uses q, $[q, \dot{q}]$, or $[q, \dot{q}, \ddot{q}]$ at the endpoint. FIR concatenates the full kinematic vector at lags $0, 4, 8, \ldots < W_{\text{long}}$, where lags count 8-ms grid steps and W_{long} is 140, 157, or 138 for the caudal fin, webbed foot, or flipper. Thus FIR uses a sparse set of historical features within the permitted window. Centered inputs and targets are standardized using training endpoints. The solver uses

$$(X^\top X/n + \alpha I)B = X^\top Y/n, \qquad \alpha \in \{10^{-10}, 10^{-9}, \ldots, 10^{4}\},$$

where X and Y are the standardized training feature and six-load matrices, n is the number of training endpoints, B is the coefficient matrix, and I is the identity matrix. Validation six-axis nMSE selects α separately for each mechanism and feature setting. Centering supplies the intercept.

Local and recurrent neural references. MLP flattens its W1 input and applies three width-256 fully connected layers, each followed by a rectified linear unit (ReLU) and dropout 0.21, then a linear six-output head. LSTM has three unidirectional layers with hidden width 256 and dropout 0.21 between recurrent layers. Hidden and cell states start at zero for each window; a linear head maps the top-layer final hidden state to six loads. Both optimize MSE over the six standardized endpoint targets. The LSTM architecture and readout remain unchanged across its four window lengths.

TCN aggregates the supplied history through temporal convolutions. When its receptive field exceeds the observed window, the remaining positions are padded. Its deterministic implementation

computes convolutions through equivalent shifted linear operations, preserving the same observed input window.

Table 6: Model classes, their roles in the reconstruction benchmark, and key implementation details.

Class	Role in the comparison	Implementation scope
Ridge / FIR	Linear instantaneous and temporal references	Derivative ablations and linear history; validation-selected regularization
MLP	Local nonlinear mapping	Three width-256 hidden layers; rectified linear units (ReLU) and dropout 0.21; six-output linear head
LSTM	Recurrent finite history	Three layers of hidden width 256; inter-layer dropout 0.21; final hidden state and six-output linear head
TCN	Convolutional finite-history comparison	Width 200; six residual blocks, two kernel-3 convolutions per block; receptive field 253
Mamba-2	Selective state-space sequence representation	Width 224; four layers; state size 64; expansion 2; head width 64; local kernel 4
TimeBridge	Temporal and channel mixing with attention	Model width 128; feed-forward width 512; four heads; patch length 16; task-specific load head
iTransformer	Attention across motion-variable tokens	Width 192; three layers; four heads; feed-forward width 768; flattened six-load head
xLSTM (matrix-memory LSTM, mLSTM)	Matrix-memory recurrence	Width 256; three blocks; uniformly stabilized normalization; matrix-memory blocks only
LNO adaptation	Pole–residue temporal representation	Width 128; 16 stable complex modes per input/output feature pair; endpoint head
KoopmanIO	Controlled latent-linear prediction	Three-layer width-512 nonlinear lift; 256 latent dimensions; stable 2-by-2 transition blocks

Mamba-2 uses a task adaptation of structured state-space duality (Dao & Gu, 2024), with sequence states initialized for each window and a six-output endpoint head. The implementation uses deterministic tensor operations instead of an atomic-add fused kernel. TimeBridge adapts the original architecture (Liu et al., 2025) using 9/10/9 input patches, four output patches, an internal stable length of eight, and a flattened load head. Its layer counts are three within-patch, one patch-downsampling, and two cross-patch blocks. Reversible instance normalization and output denormalization are disabled; calendar inputs are zero. These changes reflect command-to-load regression rather than reproducing long-term forecasting benchmarks. iTransformer embeds each variable's entire allowed history as a token, applies attention across variables, and flattens their representations into six endpoint loads (Liu et al., 2024). Instance normalization, output denormalization, and calendar inputs are disabled; pointwise feed-forward convolutions are implemented as equivalent linear maps. Parameter counts are 1,365,510/1,372,230/1,372,038 across the three domains. Attention is restricted to the supplied feature window; the offline derivative preprocessing retains its local two-sided support.

Operator-inspired adaptations. LNO follows the pole–residue formulation (Cao et al., 2024), with stable negative-real-part poles and bounded imaginary frequencies. A per-channel-pair transient term and algebraic endpoint contraction adapt it to window-to-six-load regression. The lifted local branch and sine head preserve amplitude information; no sequence targets or extra measurements are used. Parameter counts are 1,082,886/1,083,270/1,083,654. All six fin and six webbed-foot operator-inspired runs reach the 60-epoch cap; several best checkpoints are near the cap, so optimization completeness is not established. The reported adaptation evaluates finite-window endpoint regression.

Within an input window, $k = 0, \ldots, W-1$ indexes samples. KoopmanIO learns command features ϕ_k and latent states z_k through the stable controlled recurrence $z_k = Az_{k-1} + \Delta t B\phi_k$, with $z_{-1} = 0$ and endpoint output $Cz_{W-1} + D\phi_{W-1}$. Here A, B, C, D are learned state-transition, input, output, and direct-feedthrough matrices, respectively. The transition uses 128 real rotation-decay blocks; $\Delta t = 0.008$ s. Parameter counts are 663,302/664,838/666,374. Training uses only endpoint mean squared error (MSE): there is no invariant-subspace, state-reconstruction, or rollout-consistency loss. It supplies a Koopman-inspired input–output baseline under the endpoint supervision used here (Korda & Mezić, 2018).

The neural recipe uses a validation plateau scheduler (factor 0.5, patience 3), early-stopping patience 10, and minimum improvement 10^{-5}. Validation selects the best checkpoint; final-epoch metrics are retained for diagnosis. Deterministic execution is enabled, while both matrix-multiplication and cuDNN TensorFloat-32 (TF32) flags are true in the checked TCN and modern-model runs. Concurrency varies with memory requirements and is not an additional scientific factor. Common caps and recipes do not guarantee equal wall time or tuning effort.

Measured-load-assisted setting. Persistence predicts y_t by the preceding observed load y_{t-1}. AR uses load lags $1, 5, 9, \ldots < W_{\text{long}}$, and ARX concatenates these with the FIR motion features. AR and ARX use the same standardization, regularization grid, and validation selection as Ridge. These models use measured load history and are kept outside Table 3. On the current 2-DOF webbed foot dataset, their test nMSE values are 0.3333, 0.2263, and 0.2226, respectively. They demonstrate what these particular models achieve under additional observations; their scores characterize the measured-load-assisted observation setting. They are evaluated with observed history, not as unconstrained autonomous rollouts.

C Dense errors, uncertainty, and sampling sensitivity

Table 7: Test scoring support within the shared valid interval. Endpoint spacing is 17 grid steps for 1-DOF/3-DOF and 19 for 2-DOF; dense spacing is 8 ms for all three. Points within a trajectory are temporally correlated.

Mechanism	Registered endpoints	Dense scoring points
1-DOF caudal fin	69,120	1,173,120
2-DOF webbed foot	81,649	1,548,773
3-DOF flipper	172,768	2,913,400

For trial r, channel j, and evaluated time index t, write $e_{rjt} = \hat{y}_{rjt} - y_{rjt}$. Let n_r be the number of evaluated points in trial r; bars denote time averages over those points. The decomposition used in Figure 4 is

$$\frac{1}{n_r}\sum_t e_{rjt}^2 = \bar{e}_{rj}^2 + \frac{1}{n_r}\sum_t \left[(\hat{y}_{rjt} - \bar{\hat{y}}_{rj}) - (y_{rjt} - \bar{y}_{rj})\right]^2 .$$

Each term is divided by the same training variance, weighted by n_r, and averaged across axes and seeds. Centered waveform error includes mismatches in amplitude, shape, and timing; it does not isolate them. Separate centered-RMS amplitude errors are retained in the analysis tables. Physical phase shifts and spectral peaks require a valid periodic-interval definition and are not inferred from grid-residue analysis.

Figure 8 examines scoring-position sensitivity in the caudal-fin data. Let $g = 0, 1, \ldots$ index a trial's original 125-Hz aligned grid, before valid points are selected. Registered endpoints satisfy $(g - 139) \bmod 17 = 0$ within the valid interval: 139 is the first candidate endpoint for a 140-sample window, 17 is the stride in grid samples, and mod denotes the remainder. Validity filtering retains the original indices. On the same dense prediction arrays, registered-point nMSE is 0.4221 for long LSTM and 0.4281 for long TCN, whereas the other 1,104,000 points give 0.4392 and 0.4335. The very small LSTM difference from the official endpoint score reflects the evaluation path and is far smaller than the reversal. Residue classes describe sampling positions on the reconstructed time grid, not identified physical phases.

Table 8: Dense test nMSE averaged over three seeds. Methods share the same temporal support and training scales within each domain. This table complements Table 3.

Model	1-DOF caudal fin	2-DOF webbed foot	3-DOF flipper
LSTM W1	0.4335	0.3926	0.1121
LSTM long	0.4382	0.3405	0.1063
TCN W1	0.4337	0.3976	0.1138
TCN long	0.4332	0.3586	0.1087
Mamba-2 long	0.4398	0.3484	0.1145
TimeBridge long	0.4380	0.4304	0.1287
iTransformer long	0.4361	0.4272	0.1292
LSTM W63	0.4386	0.3456	0.1073
LNO long	0.4554	0.4258	0.1662
KoopmanIO long	0.4334	0.3633	0.1098
xLSTM long	0.4460	0.3579	0.1180

The dense TCN advantage occurs in 223 of 240 test acquisition groups and seven of eight lanes. The two largest axis contributions are F_x and T_y. A 5,000-resample acquisition-group bootstrap gives a 95% interval of [0.0037, 0.0061] for the dense LSTM-minus-TCN nMSE difference, conditional on the three fitted seeds. In validation, the corresponding advantage occurs in 325 of 360 groups. Validation has already been used for model selection, and this analysis followed the test discrepancy; it is supporting diagnosis, not an independent prospective replication.

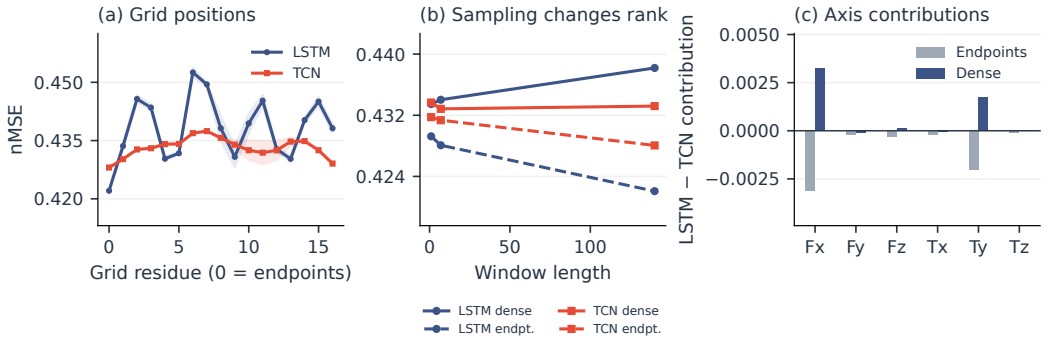

Figure 8: Fin-domain sampling sensitivity. The same predictions yield different rankings at registered endpoints and other grid positions. Grid residues denote sampling indices; shaded bands show seed SD.

For Mamba-2 versus TCN on webbed-foot endpoints, the three paired seed differences give a 95% t interval of [-0.0016, 0.0232] for TCN-minus-Mamba-2 nMSE. Three seeds provide limited information about training variability. Group bootstrap intervals answer a different question, conditional on the fitted models; neither interval treats correlated time points as independent acquisitions.

D Optimization and Numerical Diagnostics

All 18 Mamba-2/TimeBridge runs completed with finite reported best-checkpoint results. They stopped between epochs 12 and 45 under the frozen stopping rule. Early stopping means validation failed to improve under that rule; it does not prove convergence to each architecture's optimum.

In one webbed-foot TimeBridge seed, validation error worsened sharply after the second epoch and the final model became nearly constant. The best checkpoint, with test nMSE 0.4558, remains in the reported three-seed mean; the other two seeds obtain 0.4155 and 0.4156. The final failed-to-improve state has test nMSE approximately 1.0260. Large finite pre-clipping gradient norms were observed, but clipping was active and no non-finite failure was established for this run. The record supports optimization sensitivity under this recipe, not a uniquely diagnosed physical or architectural cause.

D.1 Conditional training-budget diagnostic

Figure 9 compares six webbed-foot TCN runs at a 120-epoch cap against their original 60-epoch parents, at W1/W157 and seeds 0/1/2. Validation sequences match their parents exactly through the original run lengths; early stopping and all other parameters are unchanged. Mean W1 nMSE changes from 0.3964 to 0.3957 (0.2%); W157 changes from 0.3576 to 0.3538 (1.1%). The paired seed intervals for these improvements are [-0.0001, 0.0017] and [-0.0053, 0.0130], respectively. The extended W157 mean remains 4.4% above LSTM (0.3390). This bounded extension does not erase the gap, but three seeds do not establish budget-insensitive superiority. W157 seed 1 reaches its best at epoch 120, so convergence is not claimed. Extended-budget results remain separate from Table 3.

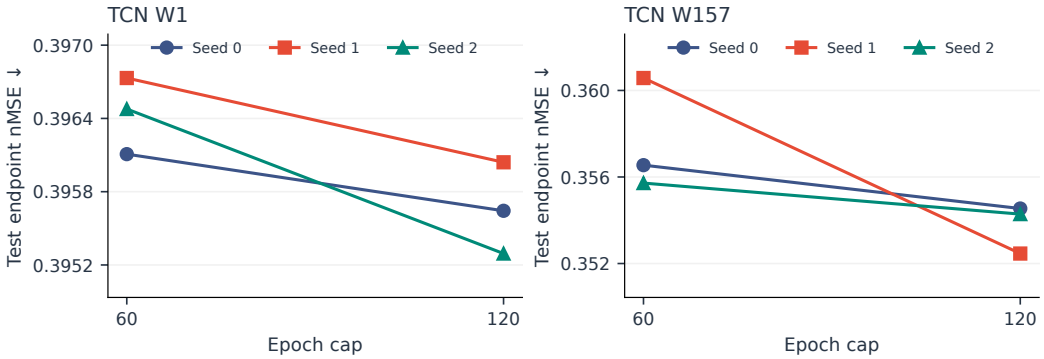

Figure 9: Three-seed conditional budget diagnostic. Each line pairs the same initialization and recipe at 60/120-epoch caps; validation-based early stopping is unchanged. These scores remain separate from the main table.

D.2 Sequence-model optimization and numerical stability

The three flipper iTransformer runs reach the 60-epoch cap, with best epochs 52/57/52. All three seeds enter the main comparison for every domain.

The mLSTM-only xLSTM adaptation (Beck et al., 2024) uses a common-scale reformulation of exponential normalization to avoid a reproduced float32 overflow mechanism. The reformulation preserves the formula and gradients in real arithmetic; gate values, learning rate, clipping, and batch handling remain unchanged. All reported xLSTM results use this same stabilized implementation. Its nine runs complete with finite recorded training metrics; the original failures are retained in the execution archive.

E Frequency support and data quantity

E.1 Frozen grouped frequency protocol

Figure 10 shows the frequency support of the grouped holdout. The splitting variable is command angular frequency: the caudal-fin joint frequency, the webbed-foot shared frequency, and the maximum of the three flipper joint frequencies. The respective cutoffs are 6.86005, 6.35099, and 7.14917 rad/s (rounded here; full precision is retained in the protocol files). They are upper-5% cutoffs computed by linear quantiles from available all-split parameter metadata. The protocol was developed after the original benchmark analysis and fixed before fitting these holdout models.

Table 9 summarizes the retained samples. Original acquisition-group split identities are preserved. Training and validation exclude groups containing high-frequency trials or missing parameters. Parameter-complete original test groups provide ID trials when the whole group is low-frequency. High-frequency test groups contribute their high-frequency trials to OOD; their low-frequency companions are excluded. Groups with incomplete parameters are excluded throughout this experiment. The 2-DOF data contain 540 trials with missing parameter entries across 536 groups.

Table 9: Grouped frequency-holdout inventory. Counts refer to individual trials, distinct acquisition groups, and registered scoring endpoints. ID and OOD denote in-distribution and out-of-distribution test subsets.

Mechanism	Train trials	Val. trials	ID trials / groups / endpoints	OOD trials / groups / endpoints
1-DOF	12,731	1,903	1,216 / 152 / 43,776	104 / 88 / 3,744
2-DOF	16,445	2,376	1,676 / 258 / 51,957	117 / 103 / 3,626
3-DOF	31,592	4,816	3,264 / 408 / 117,472	228 / 192 / 8,208

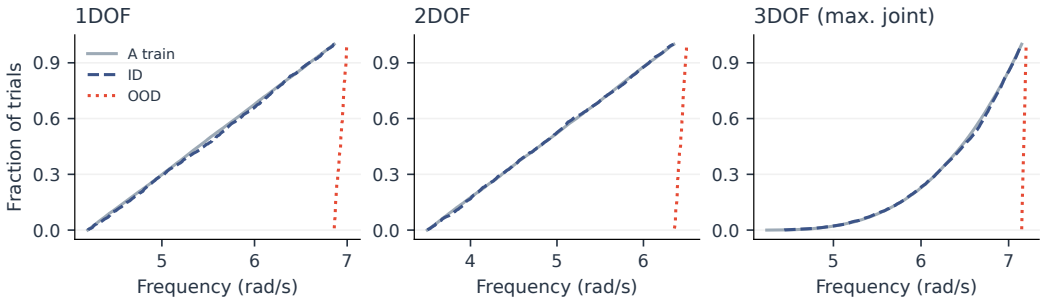

Figure 10: Held-out command-frequency support. Trial-weighted empirical cumulative distribution functions (CDFs); the flipper scalar is the maximum joint frequency. Acquisition groups remain split-disjoint. Thresholds use available all-split metadata.

Scalers are fitted on the retained training subset, and its disjoint validation subset selects checkpoints. Reporting uses three conventions: per-axis dimensional root mean squared error (RMSE) and mean absolute error (MAE), nMSE with the original main training variances, and nMSE with the retained training-subset variances. Both nMSE conventions average six channels and use exactly the same subgroup samples. Force and moment errors retain their respective units; ID and OOD scores are reported separately. Per-axis dimensional errors and both scale vectors accompany the analysis tables.

Table 10: LSTM frequency-holdout test nMSE. Fixed-main columns use the main training variances and report mean $\pm$ sample SD over three seeds; subset-scale columns use the retained holdout-training variances and report means over the same seeds. Each row uses the same predictions under both normalizations. † denotes a nonzero SD smaller than 0.0001.

Mechanism / window	Fixed main ID	Fixed main OOD	Subset scale ID	Subset scale OOD
1-DOF / W1	$0.4393 \pm$ †	0.5110 ± 0.0002	0.4515	0.5245
1-DOF / W140	0.4370 ± 0.0019	0.5041 ± 0.0030	0.4491	0.5174
2-DOF / W1	0.3779 ± 0.0003	0.5150 ± 0.0011	0.3928	0.5358
2-DOF / W157	0.3304 ± 0.0020	0.4373 ± 0.0070	0.3433	0.4549
3-DOF / W1	0.1135 ± 0.0001	0.1173 ± 0.0004	0.1151	0.1189
3-DOF / W138	0.1079 ± 0.0007	0.1150 ± 0.0009	0.1094	0.1165

Table 10 reports both normalization conventions. Fixed-scale endpoint history improvements in ID/OOD are 0.5%/1.4% (fin), 12.6%/15.1% (webbed-foot), and 4.9%/2.0% (flipper). Three-seed paired t intervals quantify initialization variation; the six subgroup-specific 5,000-resample acquisition-group bootstrap intervals condition on the fitted seeds. Fin OOD has a positive group-bootstrap interval but a seed interval spanning zero, while flipper OOD spans zero under both. Independent resampling of ID and OOD groups supports a positive webbed-foot generalization gap, whereas the fin and flipper gap intervals include zero. These different uncertainty sources are reported separately. Similar marginal coverage does not control joint frequency combinations, derivatives, or group selection, so the gap is not attributed solely to frequency or an unobserved physical process. Earlier trial-level holdouts are exploratory and not pooled with the grouped holdout.

E.2 LOCATING THE EXTRAPOLATION ERROR

At dense webbed-foot OOD points, W1-to-W157 history lowers nMSE from 0.5136 to 0.4351. Of the 0.0785 reduction, 0.0767 (97.7%) comes from centered-waveform error and 0.0018 from squared trial-mean error. Long-window ID-to-OOD centered error increases by 0.1124 while squared trial-mean error decreases by 0.0078. The amplitude contribution decreases: its normalized contribution falls from 0.0864 to 0.0837.

For each trial and axis, centered MSE further equals $(s_p - s_y)^2 + 2(s_p s_y - \mathrm{cov}(p, y))$, where p and y are the predicted and measured time series for that trial and channel, s_p and s_y their population standard deviations, and $\mathrm{cov}(p, y)$ their population covariance over the same evaluated points. The first term is centered-RMS amplitude mismatch; the second measures correlation-related discrepancy, without dividing by a potentially zero standard deviation. In webbed-foot long-LSTM predictions, the aggregate second term rises from 0.2243 (ID) to 0.3395 (OOD). It mixes waveform shape and temporal alignment and does not identify a physical phase lag. Neither decomposition isolates a unique causal source; additional measured execution or controlled interventions would be needed for that claim.

E.3 GROUP-SAMPLED DATA CURVE

Sampling uses a seeded permutation of the 3,996 webbed-foot training acquisition groups. Prefixes of 999 and 1,998 groups supply 25% and 50%; all valid trials within each selected group are retained. Their respective training endpoint counts are 201,864/202,922/202,990 and 404,764/406,590/406,519, compared with 812,129 at 100%. Independent sampling seeds 101/202/303 are paired with training seeds 0/1/2. Group and trial hashes were reconstructed, and selected sets are nested and identical across W1/W157. Training normalization is subset-specific, while the evaluation denominator remains the fixed full training scale. Original 100% runs are reused. This design estimates joint sampling-and-initialization variability, not its two components independently.