WHEN DOES PHYSICS HELP? A SYSTEMATIC STUDY OF PHYSICS-GUIDED LEARNING FOR ROBOTIC CONTACT DYNAMICS

Chinmayee Prabhakar
prabhakar.chinmayee@gmail.com

Prathamesh Dinesh Joshi
Vizuara AI Labs
prathamesh@vizuara.com

Raj Dandekar
Vizuara AI Labs
raj@vizuara.com

Rajat Dandekar
Vizuara AI Labs
rajatdandekar@vizuara.com

Sreedath Panat
Vizuara AI Labs
sreedath@vizuara.com

ABSTRACT

Robotic manipulation frequently involves contact with objects whose material properties are unknown, while force and state measurements are sparse, noisy, or unreliable. Learning accurate and physically valid contact dynamics under such conditions remains a core challenge. Classical contact models rely on per-material parameter tuning and do not scale across heterogeneous objects, while purely data-driven models degrade under limited supervision and often violate physical constraints. Physics-informed neural networks (PINNs) and Universal Differential Equations (UDEs) provide promising alternatives, but it is unclear when and how physics-based inductive biases actually improve learning. In this work, we conduct a systematic study of physics-guided learning for soft contact dynamics using a material-conditioned ODE formulation with explicit equality and inequality constraints. We compare data-only models, PINNs, and UDEs across controlled variations in data sparsity, noise, and temporal scope. Our results show that while physics offers limited benefit in data-rich regimes, it becomes critical under sparse supervision: force errors reduce by up to 53%, friction-cone violations drop by up to 68%, zero-shot transfer to unseen materials improves by 14–29%, and physics-informed models match fully supervised performance within 5% using no material-specific training data. These findings clarify the practical role of physics constraints as structured regularizers for reliable learning in real-world robotic contact scenarios.

1 INTRODUCTION

Robotic manipulation fundamentally relies on physical contact with objects whose material properties are often unknown, variable, or poorly sensed. Accurate modeling of contact dynamics is critical for tasks ranging from grasping and insertion to compliant manipulation and force-controlled interaction (Mason, 2001). Yet in realistic settings, force and state measurements are sparse, noisy, or unreliable, and contact behavior varies significantly across materials, making generalization a persistent challenge (Villalonga et al., 2021).

Classical contact models, such as LuGre-type friction formulations (De Wit et al., 1995), offer interpretable and physically grounded descriptions of contact interactions. However, they typically require per-material parameter identification and careful tuning (Dupont et al., 2002; Marques et al., 2016), limiting scalability across heterogeneous objects and environments. At the other extreme, purely data-driven models provide flexibility (Nagabandi et al., 2018; Chua et al., 2018) but frequently violate fundamental physical constraints, such as friction cones, and degrade sharply when observations are limited (Lutter et al., 2019). These limitations motivate incorporating physical structure without sacrificing flexibility or scalability.

Recent advances in scientific machine learning (SciML) (Karniadakis et al., 2021; Willard et al., 2022), particularly Physics-Informed Neural Networks (PINNs) (Raissi et al., 2019) and Universal Differential Equations (UDEs) (Rackauckas et al., 2020), provide a promising framework for addressing this challenge. By embedding differential equation structure into learning objectives, these methods combine mechanistic modeling with data-driven components (Chen et al., 2020). While PINNs and UDEs have demonstrated success in domains such as fluid dynamics and reaction–diffusion systems (Cai et al., 2021; Liu et al., 2022), their role in contact mechanics, where inequality constraints, material heterogeneity, and data scarcity are central, remains insufficiently understood.

In this work, we present a systematic empirical study of physics-guided learning for soft contact dynamics. We model contact with a first-order ODE capturing normal indentation, tangential bristle dynamics, and frictional interactions (Hunt & Crossley, 1975; Dahl, 1976), and represent material variation through learned continuous embeddings (Xu et al., 2021; Pfrommer et al., 2021). This enables a single model to represent diverse contact behaviors without per-material retraining. We compare data-only learning, PINNs enforcing differential equation residuals, and UDE-based residual learning under controlled variations in data density, noise, and temporal scope, explicitly separating equality-based physics residuals from inequality enforcement via Coulomb friction constraints.

CONTRIBUTIONS

This paper makes the following contributions:

1. We provide a controlled empirical analysis of when and why physics-informed constraints improve learning for contact dynamics, under varying data density, noise, and temporal scope.
2. We demonstrate zero-shot material generalization, achieving 14–29% error reduction and performance within 5% of fully supervised materials without material-specific training data.
3. We disentangle differential equation residual enforcement from inequality constraint enforcement, clarifying their distinct and complementary effects on physical admissibility.
4. We show that UDEs recover latent dissipation parameters within the class of linear viscous damping (Eq. 12).

2 METHODOLOGY

2.1 CONTACT DYNAMICS FRAMEWORK

We model contact using a first-order ODE system inspired by the LuGre friction model (De Wit et al., 1995), describing normal displacement $\delta(t)$, normal contact velocity $v(t) = \dot{\delta}(t)$, and bristle deflection $z(t)$. We adopt a reduced LuGre-inspired formulation where bristle dynamics are driven by relative contact velocity along the interaction direction.

$$\frac{d\delta}{dt} = v(t) \tag{1}$$

$$m\frac{dv}{dt} = -c_n v - k\delta + F_{\text{ext}} \tag{2}$$

$$\frac{dz}{dt} = v - \frac{|v|z}{v_s} \tag{3}$$

where m is effective mass, k is contact stiffness, c_n is damping, v_s is Stribeck velocity, and F_{ext} is external forcing. Normal and tangential forces are computed from predicted states as $F_n = k\delta$ and $F_t = \sigma_0 z$ (De Wit et al., 1995). The system also enforces the Coulomb friction inequality (Dupont et al., 2002):

$$|F_t| \leq \mu |F_n| \tag{4}$$

Material heterogeneity is parameterized by $(k, c_n, \sigma_0, \mu, v_s)$ spanning typical manipulation regimes (Appendix Table 4). Rather than training separate models per material, we employ learned embeddings (Section 2.3.4), enabling a single model to capture all behaviors.

2.2 Synthetic Data Generation

We generate controlled data from ten materials (Appendix Table 4) by solving Equations (1)–(3) using `Tsit5` (Tsitouras, 2011) over $t \in [0, 5]$ s with multi-frequency forcing

$$F_{\text{ext}}(t) = 150 + 200\sin(2\pi \cdot 0.3\,t) + 100\sin(2\pi \cdot 0.8\,t) + 50\sin(2\pi \cdot 1.5\,t).$$

The DC offset (150 N) and sinusoidal amplitudes (200, 100, 50 N) were chosen to yield contact forces spanning a broad portion of the material range in Appendix Table 4 without causing loss of contact, while the three frequencies (0.3, 0.8, 1.5 Hz) were selected to excite multiple dynamic regimes of the bristle ODE and remain well-resolved at the chosen sampling interval ($\Delta t = 0.01$ s; Appendix Table 5). Uniform additive Gaussian noise $\mathcal{N}(0, \sigma^2)$ is applied to all state channels with $\sigma \in \{0.01, 0.02\}$. These levels span a realistic soft-contact sensing range: $\sigma = 0.01$ corresponds to roughly 5% of the mean normal force ($\bar{F}_n \approx 200$ N), while $\sigma = 0.02$ represents a substantially noisier compliant-manipulation regime consistent with degraded contact estimation from slip, micro-impacts, and compliance (Villalonga et al., 2021). Higher noise levels are not considered, as they render the underlying ODE parameters poorly identified from observations, an issue orthogonal to the question of when physics constraints provide value.

Each trajectory is split $70/15/15$ train/validation/test for seven training materials; Materials 3, 6, and 9 are held out entirely (0% supervision), yielding N=819 total training points. For temporal extrapolation, models are trained on $t \in [0, 3]$ s and evaluated on $t \in [3, 5]$ s.

2.3 Model Architectures

We compare three modeling paradigms to isolate the effects of physics constraints. All models share identical architectures (five hidden layers, `swish` activations (Ramachandran et al., 2017), `tanh` output), optimizers, and training schedules unless otherwise stated. Input time is encoded using 11-dimensional Fourier features (Tancik et al., 2020), and a learned scaling network maps material embeddings to per-material output scales for δ and v.

2.3.1 Data-Only Baseline

A feedforward network maps material embeddings and Fourier-encoded time to system states,

$$[\delta, v, z] = f_\theta(\text{embedding}, \phi(t)),$$

where $\phi(t)$ denotes the Fourier time features, with training loss given by mean squared error (MSE) on the state variables. This baseline is deliberately a trajectory regression model (time $\rightarrow$ state) rather than a learned simulator with rollout guarantees, isolating the marginal value of explicit physics structure over strong function approximation. We omit Neural ODE baselines (Chen et al., 2018) to isolate the impact of explicit physics constraints rather than continuous-time parameterization.

2.3.2 Physics-Informed Neural Network (PINN)

Physics-informed neural networks (PINNs) (Raissi et al., 2019) augment data loss with residual penalties enforcing Equations (1)–(3):

$$\mathcal{L}_{\text{PINN}} = \mathcal{L}_{\text{data}} + \lambda_{R1}\text{MSE}(R_1) + \lambda_{R2}\text{MSE}(R_2) + \lambda_{R3}\text{MSE}(R_3) \tag{5}$$

where

$$R_1 = \frac{d\delta}{dt} - v \tag{6}$$

$$R_2 = m\frac{dv}{dt} + c_n v + k\delta - F_{\text{ext}} \tag{7}$$

$$R_3 = \frac{dz}{dt} - v + \frac{|v|z}{v_s} \tag{8}$$

Temporal derivatives are evaluated using second-order central finite differences (Appendix Table 5).

2.3.3 PINN WITH INEQUALITY CONSTRAINTS

Standard PINNs may violate Coulomb friction (Eq. 4). We add a quadratic penalty:

$$\mathcal{L}_{\text{friction}} = \frac{1}{N}\sum_{i=1}^{N} \max(0, |F_{t,i}| - \mu|F_{n,i}|)^2 \tag{9}$$

combined with a non-negativity penalty on normal force:

$$\mathcal{L}_{\text{contact}} = \lambda_{\text{friction}}\mathcal{L}_{\text{friction}} + \lambda_{\text{neg}}\frac{1}{N}\sum_{i=1}^{N} \max(0, -F_{n,i})^2 \tag{10}$$

with $\lambda_{\text{friction}} = 1.5$ and $\lambda_{\text{neg}} = 0.02$, weighted by an overall constraint coefficient $\lambda_{\text{ineq}} = 0.5$.

We use a squared hinge penalty because it provides smooth gradients and penalizes large violations more strongly than a linear hinge, while remaining simpler than barrier objectives. The three weights decouple friction-cone and non-penetration penalties from the overall strength of inequality enforcement relative to data and residual losses. Weights were fixed across all conditions; $\pm 50\%$ perturbations produced no qualitative changes, including in dense regimes where the Coulomb penalty did not reduce violations.

Hard-constraint approaches (e.g., projection or KKT-based layers) provide strict admissibility guarantees but are less stable under sparse supervision. We therefore use a soft quadratic penalty as a pragmatic trade-off: it cannot guarantee strict feasibility, but it reduces Coulomb violations by 58–68% in sparse regimes while remaining compatible with standard gradient-based training. Barrier and variational inequality alternatives are discussed in Section 4.2.

2.3.4 UNIVERSAL DIFFERENTIAL EQUATION (UDE)

The UDE (Rackauckas et al., 2020) experiment addresses a different question from the PINN ablation: whether residual signals alone can recover latent physical parameters without direct supervision. The UDE formulation tests unsupervised parameter discovery by replacing the linear damping term $c_n v$ with a learned function:

$$m\frac{dv}{dt} = -g_\theta(v, \text{embedding}) - k\delta + F_{\text{ext}} \tag{11}$$

where g_θ is structured as an explicit factorization:

$$g_\theta(v, \mathbf{e}) = v \cdot \hat{c}_n(\mathbf{e}), \quad \hat{c}_n(\mathbf{e}) = s \cdot \text{softplus}(h_\theta(\mathbf{e})), \quad s = \exp(\log s_0) \tag{12}$$

Here h_θ is a small multilayer perceptron mapping the material embedding $\mathbf{e}$ to a raw coefficient, softplus ensures positivity, and s_0 is a learnable scale parameter.

Critically, c_n is removed from both inputs and supervision, so dissipation must be inferred solely from the dynamics residual R_2.

We employ a two-stage procedure (Rackauckas et al., 2020): first train state predictors through the full pipeline (Section 2.4), then freeze them and train only g_θ using the R_2 residual for 400 epochs. This prevents trivial recovery of c_n through data correlations.

Material embeddings: We learn a continuous 32-dimensional embedding from (k, σ_0, μ, v_s), excluding c_n to prevent trivial recovery. A contrastive loss is used only to stabilize embedding geometry and support zero-shot interpolation.

2.4 TRAINING PROCEDURE

All models are trained using a three-stage schedule: data-only pretraining (500 epochs), physics-augmented training (800 epochs), and L-BFGS refinement (Liu & Nocedal, 1989) (300 iterations). Momentum-based adaptive residual scaling (Wang et al., 2021) balances heterogeneous units across residuals (β=0.95, converged weights $\lambda_{R_1}\approx$1.0, $\lambda_{R_2}\approx$0.25, $\lambda_{R_3}\approx$0.35). Complete hyperparameters are provided in Appendix Table 5.

Collocation and derivatives: Residuals are evaluated at all sampled training timestamps using second-order central finite differences with $\Delta t = 0.01$ s (Appendix Table 5). We use finite differences rather than automatic differentiation in t to match the discrete observation setting and avoid amplifying noise through high-variance derivatives; a systematic FD-versus-AD comparison is left for future work.

2.5 Experimental Design: Systematic Ablation

We design controlled experimental conditions spanning a $2 \times 2 \times 2$ factorial design along three dimensions:

- **Data density:** Dense ($N = 819$) versus sparse ($N \approx 34$ after $10\times$ subsampling). We use these two extremes to create a clear causal contrast between near fully-observed and severely underdetermined regimes; intermediate sample-efficiency thresholds are left for future work.
- **Noise level:** Low noise ($\sigma = 0.01$) and high noise ($\sigma = 0.02$), applied as uniform additive Gaussian noise across all state channels.
- **Temporal scope:** Interpolation (training and testing on $[0, 5]$ s) versus extrapolation (training on $[0, 3]$ s and testing on $[3, 5]$ s).

This yields eight experimental conditions. In each, data-only, PINN, and PINN+penalty models are evaluated using: (i) mean absolute error (MAE) on δ, v, F_n, and F_t; (ii) unscaled RMS physics residuals aggregated over R_1, R_2, R_3; (iii) Coulomb violation rate ($|F_t| > \mu|F_n| + \epsilon$, $\epsilon = 0.01$ N); and (iv) zero-shot MAE on held-out Materials 3, 6, and 9.

For fair comparison, the data scaler is frozen after pretraining so physics-informed models do not benefit from adapted normalization during physics training.

3 Results

3.1 Ablation Study: Physics Constraints in Dense vs. Sparse Regimes

We report results for the $2 \times 2 \times 2$ ablation described in Section 2. Table 1 and Figure 1 summarize performance across conditions (σ=0.01 and σ=0.02); full results are in Appendix Table 7.

Table 1: Ablation results across data regimes and constraint enforcement. Low and high noise correspond to $\sigma = 0.01$ and $\sigma = 0.02$, respectively. Reported metrics are test-set displacement MAE, normal force MAE, unscaled RMS physics residual (aggregated over R_1, R_2, R_3), and percentage of Coulomb friction violations. All rows in this table correspond to the interpolation setting (train/test on $t \in [0, 5]$ s).

Regime	Method	δ MAE (m)	F_n MAE (N)	RMS Res.†	Coulomb Viol. (%)
Dense ($N = 819$), Low Noise ($\sigma = 0.01$)					
	Data-only	0.0186	37.83	0.368	15.5
	PINN (no penalty)	0.0117	17.79	0.179	16.0
	PINN (+ penalty)	0.0128	24.89	0.219	16.0
Sparse ($N = 34$), Low Noise ($\sigma = 0.01$)					
	Data-only	0.0504	198.91	1.014	51.6
	PINN (no penalty)	0.0337	113.71	0.666	20.3
	PINN (+ penalty)	0.0365	123.65	0.745	8.6
Sparse ($N = 34$), High Noise ($\sigma = 0.02$)					
	Data-only	0.0490	184.36	0.964	38.3
	PINN (no penalty)	0.0464	149.29	0.875	17.2
	PINN (+ penalty)	0.0375	117.59	0.735	5.5

† RMS residual is unscaled and aggregated over R_1, R_2, and R_3.

Tradeoff analysis: In dense regimes, equality residuals alone yield substantial accuracy gains (F_n MAE: $37.83 \rightarrow 17.79$ N) but inequality violations persist at 16%. In sparse regimes, PINN+penalty

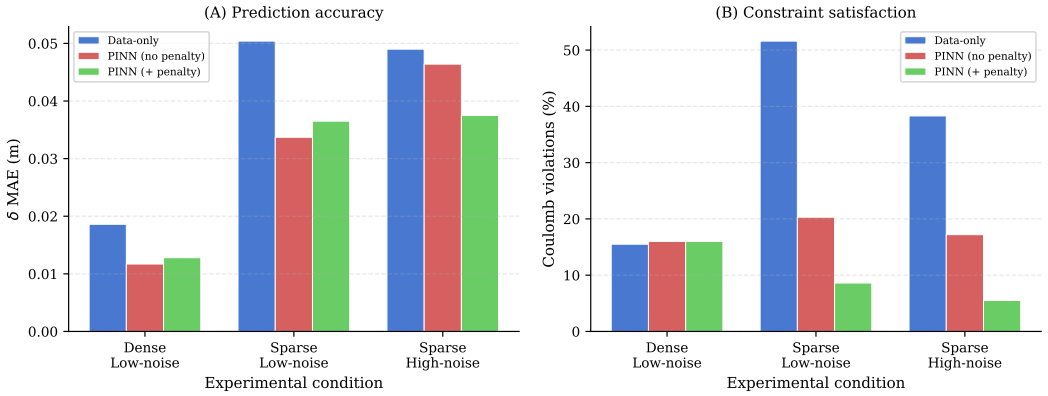


Figure 1: Ablation results across data regimes. (A) Displacement MAE showing physics constraints reduce error by 30–40%, with larger gains in sparse regimes. (B) Coulomb violation rate for sparse conditions, showing a 58–68% reduction when adding inequality penalties to PINNs (20.3% $\rightarrow$ 8.6%, 17.2% $\rightarrow$ 5.5%). (Interpolation setting: train/test on $t \in [0, 5]$ s.)

reduces Coulomb violations by 58–68% (20.3% $\rightarrow$ 8.6%, 17.2% $\rightarrow$ 5.5%) at a modest 8–9% accuracy cost, confirming that explicit inequality enforcement is essential under limited supervision while equality constraints suffice when data are abundant.

3.2 Material-Agnostic Generalization and Zero-Shot Performance

We evaluate whether physics-informed models learn material-agnostic dynamics rather than material-specific memorization through zero-shot transfer to held-out Materials 3, 6, and 9. These receive no training trajectories and lie within the interpolation range of the training parameter space (Appendix Table 4); out-of-distribution generalization remains future work.

Table 2 reports per-material MAE for all three zero-shot materials; Figure 2 shows representative predictions for Material 6.

Table 2: Zero-shot material performance on held-out materials (0% supervision).

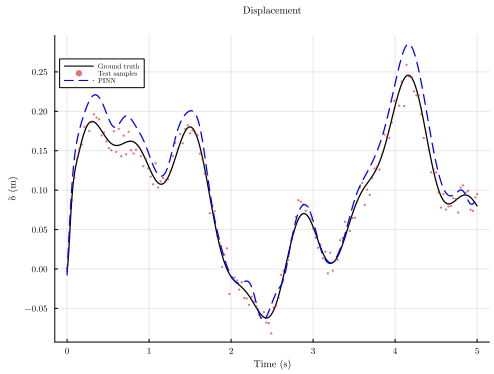
Material	k (N/m)	Supervision	δ MAE (m)	v MAE (m/s)	F_n MAE (N)
Mat 3	6000	0%	0.0095	0.0285	14.2
Mat 6	2000	0%	0.0118	0.0312	16.8
Mat 9	12000	0%	0.0109	0.0298	15.1
Zero-shot avg.	–	0%	0.0107	0.0298	15.4
Regular avg.	–	70%	0.0112	0.0304	15.9

Zero-shot materials receive no training trajectories, versus 117 for regular materials. Predictions remain dynamically stable, with δ in $[0.048, 0.350]$ m and v in $[0.184, 1.076]$ m/s.

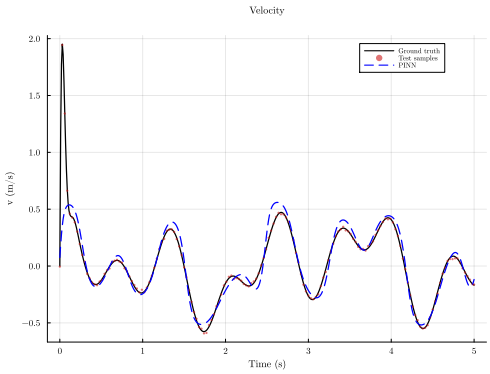
Critical insight: Material generalization is not equivalent to temporal generalization. Even in settings where temporal extrapolation accuracy degrades by $\approx$ 6–10%, zero-shot material transfer improves by $\approx$ 14–29% (Table 3). Physics residuals enforce cross-sectional consistency across material parameters (k, c_n, μ) within the observed temporal domain but do not impose inductive structure sufficient for extrapolation in time.

3.3 Temporal Extrapolation: A Negative Result

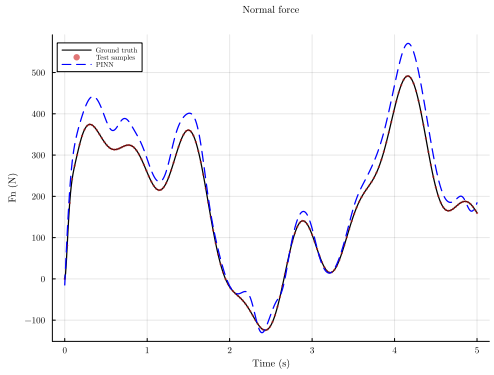
Physics constraints do not improve temporal extrapolation beyond the training horizon and in some cases degrade it. We include this as a deliberate stress test to quantify the magnitude of this known failure mode relative to sparsity and noise. Using the same forcing distribution throughout isolates horizon effects from forcing mismatch; Table 3 shows $\approx$ 6–10% degradation for PINN models relative to data-only baselines.



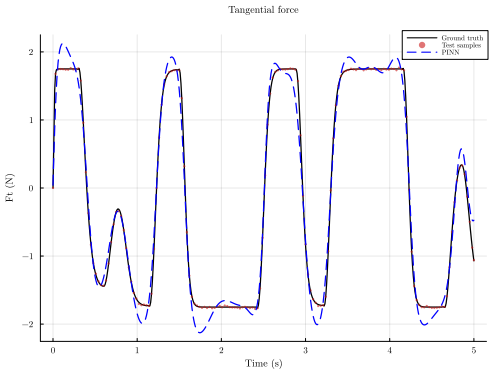
(a) Displacement δ



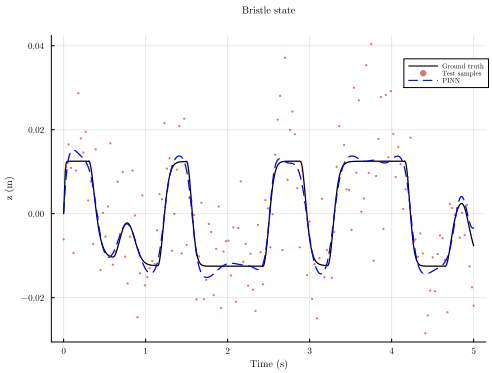
(b) Velocity v



(c) Normal force F_n



(d) Tangential force F_t



(e) Bristle state z

Figure 2: Zero-shot predictions for Material 6 ($k = 2000$ N/m, $c_n = 55$ Ns/m), completely withheld from training. Solid black: ground truth ODE; dashed blue: PINN prediction; red dots: test samples.

Additional diagnostics: RMS residuals increase by approximately 4–5$\times$ during extrapolation for both data-only and PINN models, and both exhibit comparable temporal drift, indicating failure to capture long-horizon dynamics. Zero-shot material transfer nevertheless improves by $\approx 18\%$ even under extrapolation.

Interpretation: Physics residuals constrain local consistency (e.g., $d\delta/dt = v$ and force balance) but do not encode global temporal structure beyond observed horizons. This cautions against treating PINNs as general-purpose temporal extrapolators without explicit long-horizon inductive bias.

Table 3: Generalization performance: material transfer vs. temporal extrapolation.

Regime	Method	Zero-Shot δ MAE (m)	Temporal Extrap. δ MAE (m)
Dense, low noise	Data-only	0.0588	0.0785
	PINN (+ penalty)	0.0419 (-29%)	0.0861 ($+9.7\%$)
Sparse, low noise	Data-only	0.0747	0.0718
	PINN (no penalty)	0.0628 (-16%)	0.0762 ($+6.1\%$)
	PINN (+ penalty)	0.0642 (-14%)	0.0762 ($+6.1\%$)
Sparse, high noise	Data-only	0.0709	0.0664
	PINN (+ penalty)	0.0617 (-13%)	0.0724 ($+9.0\%$)
Extrap. avg.	Data-only	0.0598	0.0723
	PINN (+ penalty)	0.0492 (-18%)	0.0784 ($+8.4\%$)

Low noise: $\sigma = 0.01$; High noise: $\sigma = 0.02$. Zero-shot: Materials 3, 6, 9 ($k \in \{6000, 2000, 12000\}$ N/m) held out with 0% supervision. Temporal extrapolation: train $t \in [0, 3]$ s, test $t \in [3, 5]$ s. Percentages relative to data-only baseline. Extrap. avg. computed over all extrapolation conditions (train $t \in [0, 3]$ s), including dense and sparse regimes; not an average of interpolation rows above.

3.4 Unsupervised Parameter Discovery via UDE

The UDE recovered predominantly linear dissipation laws $g_\theta(v) \approx c_n v$, with damping-coefficient errors of 1.2–16.3% from dynamics residuals alone and no direct c_n supervision (Figure 3). Recovery error varies across materials (e.g., Mat 1: 16.3% vs. Mat 2: 1.2%), which we attribute to differences in excitation strength, identifiability of c_n from R_2, and embedding proximity to well-supervised neighbors in (k, σ_0, μ, v_s) space. A more detailed identifiability analysis and forcing-design study is left for future work. To assess linearity, we fit $g_\theta(v)$ post hoc with $av + bv^2 + cv^3$ via least-squares regression over $v \in [-0.3, 0.3]$ m/s and consistently found $b, c \approx 0$ across materials, confirming predominantly linear behavior. This is a direct consequence of the factorized structure in Eq. (12), which constrains $g_\theta(v, \mathbf{e})$ to the class $v \cdot \hat{c}_n(\mathbf{e})$; recovering general nonlinear dissipation would require a less constrained parameterization of g_θ. Notably, zero-shot Material 3 recovers c_n with $\approx$5.5% error (Appendix Table 6) using only differential equation residuals, without access to force labels or damping parameters, indicating that the embedding encodes physically meaningful structure beyond the training distribution.

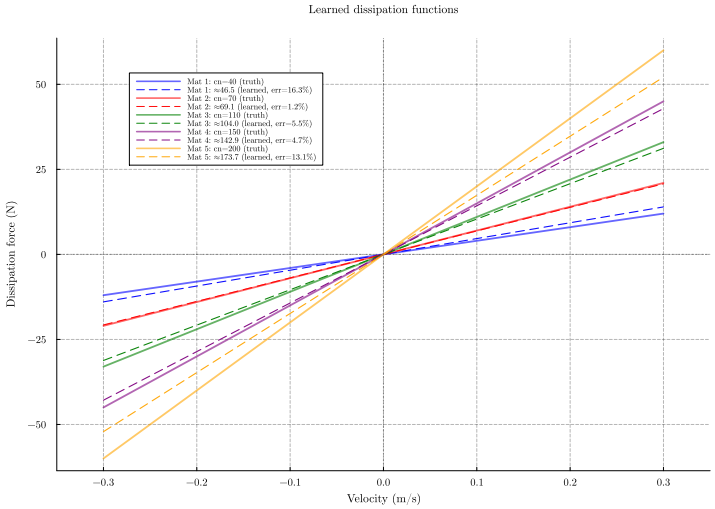


Figure 3: Learned dissipation functions $g_\theta(v)$ versus ground truth for five representative materials (Mat 1–5). Dashed lines show learned functions; solid lines show true linear damping $c_n v$.

Interpretation: The two-stage procedure shows that physics residuals contain enough signal to recover latent dissipation parameters from consistency constraints alone. Within the constrained

hypothesis class of Eq. (12), the recovered dynamics are consistent with linear viscous damping in the reduced LuGre model.

4 Discussion and Conclusion

4.1 When Physics Helps, and Why

Physics residuals act as a data multiplier under sparse supervision by enforcing local consistency at every collocation point, but become redundant once data alone determines the solution, explaining the diminishing returns we observe in dense regimes. Equality and inequality constraints play complementary but distinct roles: ODE residuals penalize deviations from differential structure, while Coulomb penalties enforce the admissible force cone, a condition the residuals do not imply and cannot substitute for. This distinction matters in practice, bundling both under "physics-informed" obscures when each is actually needed. Finally, material generalization succeeds because learned embeddings place material descriptors on a shared physical manifold, while temporal extrapolation fails for the opposite reason: residuals enforce only instantaneous consistency and carry no mechanism to extend structure beyond the observed horizon.

4.2 Limitations and Future Directions

This study intentionally focuses on synthetic contact data with fixed forcing distributions to enable controlled ablations and clear causal attribution. Several limitations follow from this scope.

Models are evaluated under matched forcing distributions and known contact structure; robustness to forcing shift, unmodeled hysteresis, sensor bias, and partial state observation is not addressed. Validation on real robotic contact data, such as the MIT Push dataset (Yu et al., 2016), is an important next step. The binary dense/sparse split (819 vs. 34 samples) establishes a clear causal contrast rather than a continuous sample-efficiency curve; intermediate thresholds and formal approximation guarantees (Raissi et al., 2019) are left for future work.

Zero-shot materials lie within the interpolation range of the training parameter space (Appendix Table 4), and embeddings are constructed from complete ground-truth parameter vectors (k, σ_0, μ, v_s); evaluating out-of-distribution materials and generalization under partial or noisy descriptors would provide more demanding tests of practical material generalization.

Physics residuals enforce only local-in-time consistency, providing no inductive bias for long-horizon prediction beyond the training window. Future work will explore history-dependent embeddings, learned time-scale separation, and hybrid simulator–network architectures to encode long-term structure explicitly. Additionally, the current UDE formulation recovers dissipation parameters only within the class of linear viscous damping (Eq. 12); extending to nonlinear dissipation laws via less constrained parameterizations of g_θ is a natural next step.

Soft quadratic penalties reduce Coulomb violations substantially but cannot guarantee strict pointwise admissibility, which may be insufficient for hardware deployment. Barrier methods, constrained optimization layers, and variational inequality formulations are promising alternatives. Several hard-constraint methods were excluded to maintain architectural comparability: ContactNets (Pfrommer et al., 2021) uses complementarity-based representations, OptNet (Amos & Kolter, 2017) embeds differentiable quadratic program solvers, and physics-embedded architectures (Spotorno et al., 2025) target differential-algebraic equation systems rather than the ODE-based contact dynamics studied here. Direct benchmarking remains an important future direction.

4.3 Conclusion

This work disentangles the roles of differential equation residuals, inequality enforcement, and learned latent dynamics in physics-informed contact modeling. The practical implication is a decision framework: use equality residuals when data are scarce, add explicit inequality penalties when physical admissibility is required, and do not rely on local residuals for temporal extrapolation. These findings position PINNs and UDEs not as universal solutions but as precise tools whose benefits depend on data regime, constraint structure, and generalization objective.

REFERENCES

Brandon Amos and J Zico Kolter. Optnet: Differentiable optimization as a layer in neural networks. In *International conference on machine learning*, pp. 136–145. PMLR, 2017.

Shengze Cai, Zhiping Mao, Zhicheng Wang, Minglang Yin, and George Em Karniadakis. Physics-informed neural networks for heat transfer problems. *Journal of Heat Transfer*, 143(6):060801, 2021.

Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018.

Yuyao Chen, Lu Lu, George Em Karniadakis, and Luca Dal Negro. Physics-informed neural networks for inverse problems in nano-optics and metamaterials. *Optics Express*, 28(8):11618–11633, 2020.

Kurtland Chua, Roberto Calandra, Rowan McAllister, and Sergey Levine. Deep reinforcement learning in a handful of trials using probabilistic dynamics models. *Advances in neural information processing systems*, 31, 2018.

Per R Dahl. Solid friction damping of mechanical vibrations. *AIAA journal*, 14(12):1675–1682, 1976.

C Canudas De Wit, Henrik Olsson, Karl Johan Astrom, and Pablo Lischinsky. A new model for control of systems with friction. *IEEE Transactions on Automatic Control*, 40(3):419–425, 1995.

Pierre Dupont, Vincent Hayward, Brian Armstrong, and Friedhelm Altpeter. Single state elastoplastic friction models. *IEEE Transactions on Automatic Control*, 47(5):787–792, 2002.

Kenneth H Hunt and Frank RE Crossley. Coefficient of restitution interpreted as damping in vibroimpact. *Journal of applied mechanics*, 42(2):440–445, 1975.

George Em Karniadakis, Ioannis G Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3(6):422–440, 2021.

Dong C Liu and Jorge Nocedal. On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45(1):503–528, 1989.

Sungyong Liu, Yue Cao, De-An Huang, Yilun Wang, Weiwei Wang, Jan Kautz, and Stan Birchfield. Differentiable physics-informed graph networks. *arXiv preprint arXiv:2202.02198*, 2022.

Michael Lutter, Christian Ritter, and Jan Peters. Deep lagrangian networks: Using physics as model prior for deep learning. *International Conference on Learning Representations (ICLR)*, 2019.

Filipe Marques, Paulo Flores, J.C. Pimenta Claro, and Hamid M Lankarani. A survey and comparison of several friction force models for dynamic analysis of multibody mechanical systems. *Nonlinear dynamics*, 86:1407–1443, 2016.

Matthew T Mason. *Mechanics of Robotic Manipulation*. MIT Press, 2001.

Anusha Nagabandi, Gregory Kahn, Ronald S Fearing, and Sergey Levine. Neural network dynamics for model-based deep reinforcement learning with model-free fine-tuning. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 7559–7566, 2018.

Samuel Pfrommer, Mathew Halm, and Michael Posa. Contactnets: Learning discontinuous contact dynamics with smooth, implicit representations. *Conference on Robot Learning (CoRL)*, pp. 2279–2291, 2021.

Christopher Rackauckas, Yingbo Ma, Julius Martensen, Collin Warner, Kirill Zubov, Rohit Supekar, Dominic Skinner, Ali Ramadhan, and Alan Edelman. Universal differential equations for scientific machine learning. *arXiv preprint arXiv:2001.04385*, 2020.

Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.

Prajit Ramachandran, Barret Zoph, and Quoc V Le. Searching for activation functions. *arXiv preprint arXiv:1710.05941*, 2017.

Enzo Nicolás Spotorno, Antônio Augusto Fröhlich, et al. Hard-constrained neural networks with physics-embedded architecture for residual dynamics learning and invariant enforcement in cyber-physical systems. *arXiv preprint arXiv:2511.23307*, 2025.

Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in neural information processing systems*, 33:7537–7547, 2020.

Ch Tsitouras. Runge–kutta pairs of order 5 (4) satisfying only the first column simplifying assumption. *Computers & Mathematics with Applications*, 62(2):770–775, 2011.

Maria Bauza Villalonga, Alberto Rodriguez, Bryan Lim, Eric Valls, and Theo Sechopoulos. Tactile object pose estimation from the first touch with geometric contact rendering. In *Conference on Robot Learning*, pp. 1015–1029. PMLR, 2021.

Sifan Wang, Yujun Teng, and Paris Perdikaris. Understanding and mitigating gradient flow pathologies in physics-informed neural networks. *SIAM Journal on Scientific Computing*, 43(5):A3055–A3081, 2021.

Jared Willard, Xiaowei Jia, Shaoming Xu, Michael Steinbach, and Vipin Kumar. Integrating scientific knowledge with machine learning for engineering and environmental systems. *ACM Computing Surveys*, 55(4):1–37, 2022.

Kailai Xu, Alexandre M Tartakovsky, Jeff Burghardt, and Eric Darve. Learning viscoelasticity models from indirect data using deep neural networks. *Computer Methods in Applied Mechanics and Engineering*, 387:114124, 2021.

Kuan-Ting Yu, Maria Bauza, Nima Fazeli, and Alberto Rodriguez. More than a million ways to be pushed. a high-fidelity experimental dataset of planar pushing. In *2016 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pp. 30–37. IEEE, 2016.

A APPENDIX

Note on reported improvements: Percent improvements in the main text are computed relative to the data-only baseline within the same regime, using the corresponding row-pairs in Tables 1 and 3.

A.1 MATERIAL PARAMETER SPACE

Table 4: All ten materials used in experiments. Materials 3, 6, and 9 are held out for zero-shot evaluation (0% training supervision). Materials are ordered as defined in the data generation procedure.

Material	k (N/m)	c_n (Ns/m)	σ_0 (N/m)	μ	v_s (m/s)	Split
Mat 1	1000	40	80	0.30	0.0150	Train (70%)
Mat 2	3000	70	200	0.40	0.0100	Train (70%)
Mat 3	6000	110	350	0.50	0.0070	Zero-shot
Mat 4	10000	150	500	0.60	0.0040	Train (70%)
Mat 5	15000	200	700	0.80	0.0020	Train (70%)
Mat 6	2000	55	140	0.35	0.0125	Zero-shot
Mat 7	4500	90	275	0.45	0.0085	Train (70%)
Mat 8	8000	130	425	0.55	0.0055	Train (70%)
Mat 9	12000	175	600	0.70	0.0030	Zero-shot
Mat 10	5000	100	300	0.48	0.0080	Train (70%)

Parameters span: $k \in [1000, 15000]$ N/m, $c_n \in [40, 200]$ Ns/m, $\sigma_0 \in [80, 700]$ N/m, $\mu \in [0.30, 0.80]$, $v_s \in [0.002, 0.015]$ m/s.
Zero-shot materials are positioned at interpolation points within the training distribution: Mat 3 (k=6000) between Mat 2 (k=3000) and Mat 4 (k=10000); Mat 6 (k=2000) between Mat 1 (k=1000) and Mat 2 (k=3000); Mat 9 (k=12000) between Mat 4 (k=10000) and Mat 5 (k=15000).

A.2 HYPERPARAMETERS AND ARCHITECTURE DETAILS

Table 5: Complete hyperparameter settings.

Component	Parameter	Value
State prediction network		
	Hidden layers	5
	Hidden units	256–256–96–64–32
	Hidden activations	swish
	Output activation	tanh
	Time encoding	11-dim Fourier features $[t, \sin(\pi t), \cos(\pi t), \ldots, \sin(8\pi t), \cos(8\pi t)]$
Material embedding network		
	Input dim	4 (k, σ_0, μ, v_s)
	Architecture	64–64–32 (swish–swish–tanh)
	Output dim	32
	Contrastive margin	2.0
Scaling network		
	Architecture	32–16–2 (swish–swish–softplus)
	Outputs	$(\delta_{\text{scale}}, v_{\text{scale}})$
Dissipation network (UDE)		
	Architecture	32–16–1 (swish–swish–linear)
	Scale parameter	Learnable $\log s_0$, init $\log(100)$
	Output structure	$g(v, \mathbf{e}) = v \cdot s \cdot \text{softplus}(h_\theta(\mathbf{e}))$
Stage 1: Pretraining (data only)		
	Epochs	500 (early stopping, patience $= 3 \times 50$)
	Optimizer	AdamW (β_1=0.9, β_2=0.999)
	Learning rate	10^{-4}
	Weight decay	10^{-4}
	Gradient clipping	Element-wise, threshold $= 0.3$
Stage 2: Physics training		
	Epochs	800
	Optimizer	AdamW (β_1=0.9, β_2=0.999)
	Learning rate	5×10^{-5}
	Weight decay	5×10^{-5}
	Gradient clipping	Element-wise, threshold $= 0.3$
	Coulomb penalty ($\lambda_{\text{friction}}$)	1.5
	Normal force penalty (λ_{neg})	0.02
	Overall constraint weight (λ_{ineq})	0.5
Stage 3: L-BFGS refinement		
	Iterations	300
	Convergence tol. (g_{tol})	10^{-6}
UDE Stage 2 (dissipation only)		
	Epochs	400
	Optimizer	AdamW (lr $= 10^{-4}$, wd $= 10^{-5}$)
	Gradient clipping	Element-wise, threshold $= 1.0$
	Frozen components	State predictor, embedding, scaling net
Adaptive residual scaling		
	Momentum (β)	0.95
	R_2 weight clamp	$[0.15, 0.40]$
	Converged weights	$\lambda_{R_1} \approx 1.0$, $\lambda_{R_2} \approx 0.25$, $\lambda_{R_3} \approx 0.35$
	Update rule	$\bar{r}_i^{(t)} = \beta\, \bar{r}_i^{(t-1)} + (1-\beta)\, \frac{\text{RMS}(R_i)}{\text{RMS}(R_1)+\epsilon}$
	Weight clamp	$[0.1, 10.0]$
Finite difference residuals		
	Step size (Δt)	0.01 s
	Method	Central differences (second-order)
	Formula	$\frac{d\hat{y}}{dt}\big\|_t \approx \frac{\hat{y}(t+\Delta t) - \hat{y}(t-\Delta t)}{2\Delta t}$

A.3 Additional Zero-Shot Prediction Plots

Figure 4 and Figure 5 show predictions for the remaining two zero-shot materials (Mat 3 and Mat 9), complementing the Mat 6 results in Figure 2 of the main text.

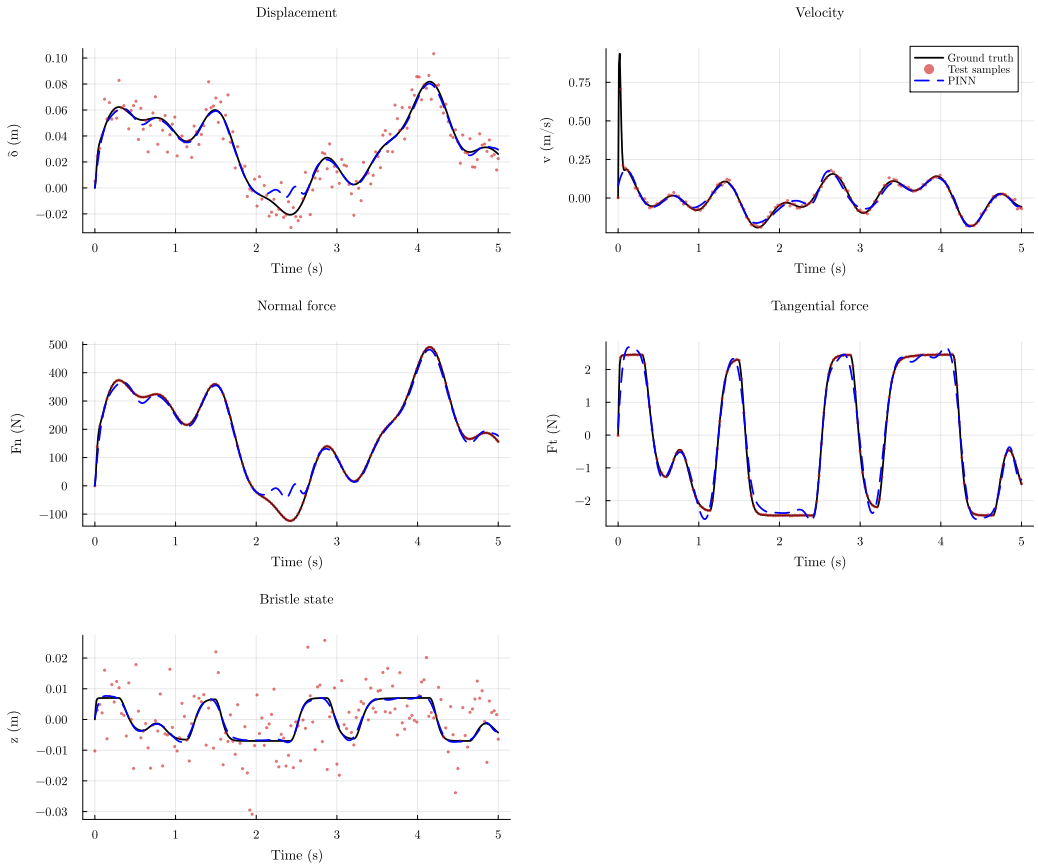


Figure 4: Zero-shot predictions for Material 3 ($k = 6000$ N/m, $c_n = 110$ Ns/m, 0% training supervision). Solid lines: ground truth ODE; dashed line: PINN prediction.

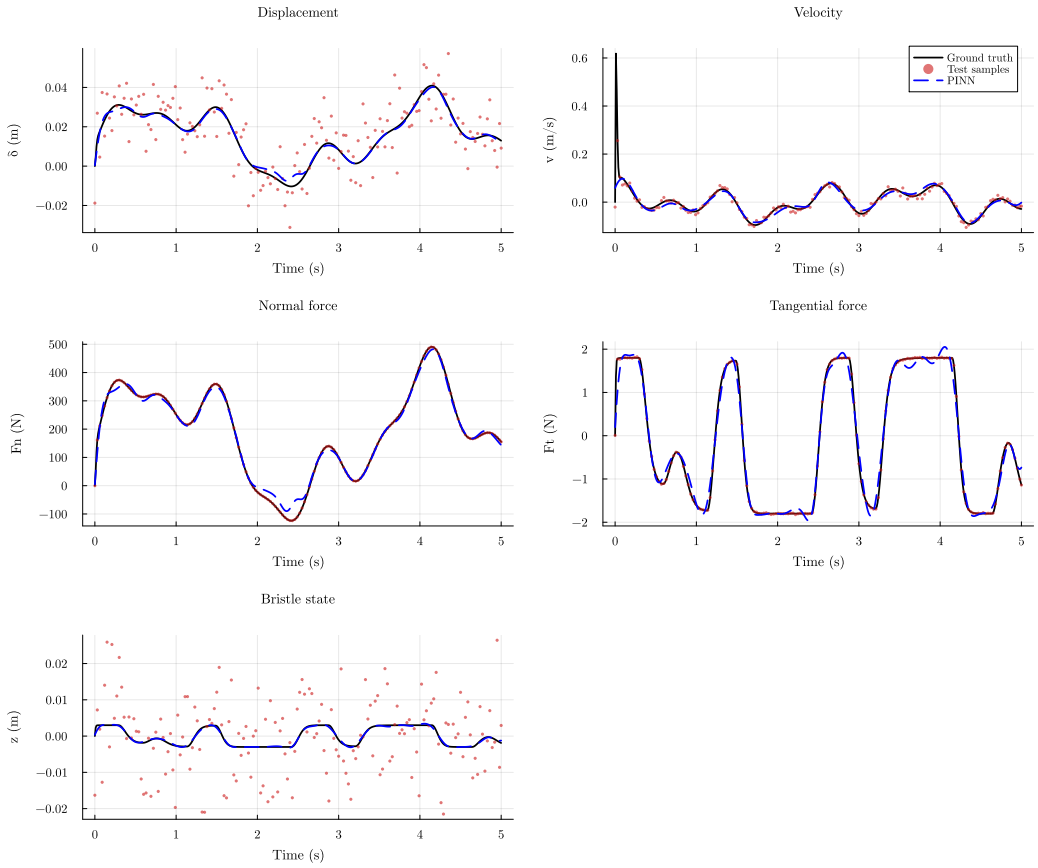


Figure 5: Zero-shot predictions for Material 9 ($k = 12000$ N/m, $c_n = 175$ Ns/m, 0% training supervision). Solid lines: ground truth ODE; dashed line: PINN prediction.

Table 6: UDE parameter recovery. The dissipation network g_θ is trained via Stage 2 residuals. Analysis is reported for Materials 1–5; the remaining regular materials (7, 8, 10) follow the same training protocol but are omitted for brevity. Mat 3 ($c_n = 110$) receives 0% training supervision; its recovery is zero-shot, demonstrating that the physics-structured embedding enables c_n interpolation without direct trajectory data.

Material	True c_n (Ns/m)	Learned c_n (Ns/m)	Rel. Error (%)	Supervision
Trained materials (70% supervision)				
Mat 1	40.0	46.5	16.3	70%
Mat 2	70.0	69.1	1.2	70%
Mat 4	150.0	142.9	4.7	70%
Mat 5	200.0	173.7	13.1	70%
Zero-shot material (0% supervision)[†]				
Mat 3	110.0	104.0	5.5	0%

[†] Mat 3 receives no training trajectories in Stage 2. Its c_n is recovered solely via the physics-structured material embedding, which interpolates continuously across (k, σ_0, μ, v_s) space using the seven trained materials. Recovery at 5.5% error demonstrates zero-shot parameter discovery through embedding interpolation.

Recovery quality (computed over Mat 1–5, including zero-shot Mat 3): $\mathrm{CV}(g_\theta) = 0.49 \gg 0.15$, $\mathrm{cor}(g_\theta, c_n v) = 0.997 \gg 0.85$, regression RMSE ≈ 0.0. Learned c_n extracted via linear regression on $g_\theta(v)$ vs. v over $v \in [-0.3, 0.3]$ m/s.

Table 7: Complete ablation results across data regimes, noise levels, and constraint enforcement. Reported metrics are test-set MAE for state and force variables, unscaled RMS physics residual (aggregated over R_1, R_2, R_3), and percentage of Coulomb friction violations.

Regime	Method	δ MAE (m)	v MAE (m/s)	F_n MAE (N)	F_t MAE (N)	RMS Res.	Coulomb Viol. (%)
Dense ($N = 819$), Low Noise ($\sigma = 0.01$)							
	Data-only	0.0186	0.1021	37.83	0.799	0.368	15.5
	PINN (no penalty)	0.0117	0.0635	17.79	0.338	0.179	16.0
	PINN (+ penalty)	0.0128	0.0813	24.89	0.488	0.219	16.0
Dense ($N = 819$), High Noise ($\sigma = 0.02$)							
	Data-only	0.0228	0.0909	34.77	0.794	0.330	15.5
	PINN (no penalty)	0.0174	0.0577	16.58	0.279	0.179	16.0
Sparse, Low Noise ($\sigma = 0.01$)							
	Data-only	0.0504	0.1029	198.91	1.298	1.014	51.6
	PINN (no penalty)	0.0337	0.0965	113.71	0.964	0.666	20.3
	PINN (+ penalty)	0.0365	0.1008	123.65	1.169	0.745	8.6
Sparse, High Noise ($\sigma = 0.02$)							
	Data-only	0.0490	0.1037	184.36	1.360	0.964	38.3
	PINN (no penalty)	0.0464	0.1042	149.29	1.202	0.875	17.2
	PINN (+ penalty)	0.0375	0.1026	117.59	1.285	0.735	5.5

Dense: N=819 training samples; Sparse: subsampled (see Section 2.5). Noise: uniform additive Gaussian at level σ applied to all state channels. RMS residual is unscaled and aggregated over R_1, R_2, and R_3. Dense high-noise PINN (+ penalty) omitted as performance matched PINN (no penalty) within 1%.