BALL PINN PROJECT

Research Summary Report

Original Vision

We were developing a Physics-Informed Neural Network (PINN) system that:

1. Learned governing equations of motion
2. Generalized to unseen initial conditions
3. Reconstructed full trajectories from arbitrary partial slices
4. Served as an MVP for missile trajectory PINNs

Our initial system modeled ideal 2D projectile motion under gravity only.

This approach directly followed the foundational PINN framework introduced in **Raissi et al., 2019 – Journal of Computational Physics [1]**, where neural networks are trained while embedding differential equation residuals into the loss function.

Baseline System — Ideal Projectile Under Gravity

We began with an ideal projectile governed by:

$$\begin{aligned} \dot{x} &= v_x, \quad \dot{y} = v_y \\ \dot{v}_x &= 0, \quad \dot{v}_y = -g \end{aligned}$$

We generated synthetic trajectories using analytical physics and trained PINNs using:

• Data loss
• Physics residual loss
• Boundary conditions

This setup aligned with the scientific machine learning paradigm described in **Karniadakis et al., 2021 – Nature Reviews Physics [2]**.

Chronological Development Attempts (A–D)

The following sequence strictly follows the progression documented in our development log.

Attempt 1 — Vanilla Position-Level PINN

We directly trained a network to predict (x(t), y(t)) using:

Loss = data + physics residual.

What Happened

• We did not even obtain a stable parabolic trajectory.
• The network behaved erratically.
• Physics loss and data loss conflicted heavily.
• No clean curve emerged.

This failure led us to question whether the model could even learn the basic shape.

Literature Context

Naive PINNs are known to suffer from gradient pathologies and training instability as discussed in **Wang et al., 2022 – SIAM Journal on Scientific Computing [3]** and further characterized in **Krishnapriyan et al., 2021 – NeurIPS [5]**.

Our observations were consistent with these known failure modes.

Attempt 2 — Boundary & Velocity Anchoring

We then added:

• Initial position boundary conditions
• Initial velocity anchoring
• Velocity loss terms

What Happened

• Losses exploded.
• Training became unstable.
• We still did not obtain a reliable parabolic trajectory.

At this stage, we were not even confident that the system could learn the basic projectile shape.

This directly led us to Attempt 3.

Attempt 3 — Data-Dominant Training (Stage-A)

To confirm whether the architecture could at least learn the shape, we removed physics constraints and trained purely on data.

What Happened

• A parabolic-like curve finally emerged.
• However, it did not match the true physics.
• Generalization completely failed.

Interpretation

This confirmed that:

• The network could learn curve shapes.
• But without physics structure, it was memorizing geometry rather than dynamics.

This aligned with standard ML limitations discussed in **Karniadakis et al., 2021 – Nature Reviews Physics [2]**.

Residual PINN (Analytical + Residual Correction)

This was one of our first promising breakthroughs.

We modeled:
$x(t) = x_{analytical}(t) + r(t)$

Where a neural network learned a residual correction.

What Happened

• For the first time, we obtained very clean parabolic shapes.
• Early results looked highly convincing.
• However, over time the residual drifted.
• Long-horizon divergence appeared.
• Residual magnitude exploded.

Why It Failed

• The residual was time-dependent.
• Residual bias integrated into position error.
• No physical scale constrained it.
• Error accumulated structurally.

This matched known instability behavior in residual ODE learning described in **Krishnapriyan et al., 2021 – NeurIPS [5]**.

Velocity-Level and Acceleration-Level PINNs

We then progressively moved deeper into the dynamics:

• Velocity-integrated PINN
• Acceleration PINN
• Gravity-anchored acceleration PINN
• Normalized time variants

What Happened

• Early segments matched extremely well.
• Short-horizon predictions looked excellent.
• Long-horizon trajectories diverged.
• Losses eventually exploded.

Structural Explanation

If acceleration bias = ε:

Velocity error grew ~ εt
Position error grew ~ εt^2

Double integration amplified even tiny systematic bias.

Theoretical instability in higher-order PINNs has been studied in **Krishnapriyan et al., 2021 – NeurIPS [5]**.

Our experiments confirmed this empirically.

Inverse PINN (D1–D5 Stages)

This phase was particularly important.

We shifted from forward modeling to inverse modeling.

D1–D3: Inverse Physics PINN

We trained models that inferred latent parameters across trajectories.

What Happened

• D1: Perfect fit to a single trajectory (expected)
• D2: Good performance across seen trajectories
• D3: Reasonable generalization to held-out trajectories

These results were strong and convincing.

However, later we realized:

• The model was partially learning dataset discretization bias.
• True unseen initial condition generalization (D4) failed.

D4 — True Unseen Initial Conditions

When tested on genuinely unseen parameter regimes:

• The model collapsed.
• Trajectories diverged significantly.

This matched generalization limitations discussed in **Mishra & Molinaro, 2022 – IMA Journal of Numerical Analysis [4]**.

D5 — Partial Trajectory Conditioning

This was the most critical turning point.

We attempted:

Given arbitrary middle slice → reconstruct full past + future.

What Happened

• The model failed.
• Even on seen trajectories, it could not reconstruct from middle slices.
• Different valid parabolas passed through the same local segment.

We realized we were facing a non-identifiability problem.

Fundamental Identifiability Discovery

A middle slice of an ideal projectile:

• Does NOT uniquely determine initial velocity.
• Does NOT uniquely determine launch angle.
• Allows infinitely many valid solutions.

This is a classical ill-posed inverse problem described in **Hadamard, 1902 – Ill-posed Problems [6]**.

We did not find explicit literature stating:

> **Ideal projectile motion is fundamentally ill-posed for arbitrary partial trajectory inversion using PINNs.**

This specific formulation and systematic empirical validation emerged from our work.

We concluded:

> **A vanilla PINN without hidden dynamics cannot reconstruct full ideal projectile trajectories from arbitrary partial slices due to fundamental non-identifiability.**
>
> **Or we can also say**
>
> **A PINN without hidden dynamics cannot demonstrate true generalization or partial trajectory inference**

This was not a training failure.
It was a physics limitation.

Stage-E — Latent-Conditioned Hidden Force Model

We then redesigned the system.

We introduced hidden wind gust forces:
$\ddot{x} = a\cos\phi \cdot \mathbf{1}_{[t_s, t_s+\Delta t]}$
$\ddot{y} = -g + a\sin\phi \cdot \mathbf{1}_{[t_s, t_s+\Delta t]}]$

We separated:

• Universal physics (gravity)
• Trajectory-specific latent dynamics (wind parameters)

This aligned with latent-conditioned operator learning frameworks such as **Lu et al., 2021 – Nature Machine Intelligence (DeepONet) [7]** and **Kovachki et al., 2023 – Acta Numerica (Neural Operators) [8]**.

Now:

• Middle slices encoded curvature information.
• Latent parameters became identifiable.
• Inverse reconstruction became structurally feasible.

Final Technical Position (Retrospective)

From this development journey, we established:

① Ideal forward PINNs worked under gravity — **Raissi et al., 2019 [1]**, **Karniadakis et al., 2021 [2]**.
② Vanilla PINNs suffered known generalization limits — **Wang et al., 2022 [3]**, **Mishra & Molinaro, 2022 [4]**.

③ Acceleration-level PINNs were structurally unstable — **Krishnapriyan et al., 2021 [5]**.
④ Ideal projectile inverse completion was mathematically ill-posed — **Hadamard, 1902 [6]**.
⑤ Latent-conditioned hidden-force PINNs aligned with modern scientific ML frameworks — **Lu et al., 2021 [7]**, **Kovachki et al., 2023 [8]**.

Research Contributions from Our Work

Our work contributed:

- A systematic empirical comparison of 8–9 PINN formulations for ideal projectile motion.
- A clear demonstration of non-identifiability in partial-trajectory inversion for trivial deterministic systems.
- Architectural justification for latent-conditioned dynamics as necessary for true inverse generalization.

REFERENCES

[1] Raissi et al., 2019 – Journal of Computational Physics
https://doi.org/10.1016/j.jcp.2018.10.045

[2] Karniadakis et al., 2021 – Nature Reviews Physics
https://doi.org/10.1038/s42254-021-00314-5

[3] Wang et al., 2022 – Gradient Pathologies in PINNs
https://doi.org/10.1137/20M1318043

[4] Mishra & Molinaro, 2022 – Generalization Error in PINNs
https://doi.org/10.1093/imanum/drab093

[5] Krishnapriyan et al., 2021 – Failure Modes in PINNs
https://arxiv.org/abs/2109.01050

[6] Hadamard, 1902 – Ill-posed Problems

[7] Lu et al., 2021 – DeepONet
https://doi.org/10.1038/s42256-021-00302-5

[8] Kovachki et al., 2023 – Neural Operators
https://doi.org/10.1017/S0962492922000088