AI Group

Interpretable Hybrid Modeling of Orbital Dynamics using Kolmogorov–Arnold Networks

Ahmed Belgacem, Vincent Boudart, Horacio Tellez

March 2026

1 Profile of the Team Leader(s) & Expected Team Composition

The project is co-led by researchers with complementary expertise in machine learning, scientific computing, and applied physics. The leadership combines experience in deep learning (PyTorch), simulation-based modeling, and data-driven approaches to physical systems.

We aim to recruit a multidisciplinary team of 4–6 participants with backgrounds spanning AI, physics, and software engineering. Target profiles include:

- Machine Learning / AI: neural networks, PyTorch, evaluation protocols
- Physics / Applied Mathematics / Aerospace: dynamical systems, orbital mechanics
- Software / Data / Simulation Engineering: Python, numerical simulation, data pipelines

Participants are expected to collaborate across domains rather than in isolated roles, enabling tight integration between physics-based modeling and learning-based components. The team structure emphasizes overlapping responsibilities, rapid iteration, and shared ownership of deliverables.

The project is designed to balance methodological rigor and practical implementation. This composition supports efficient prototyping within the 2-week timeframe while ensuring scientific quality and reproducibility, and encourages collaboration across institutions and disciplines.

2 Abstract

Can we learn missing physical laws while keeping models interpretable and auditable?

This project investigates Kolmogorov–Arnold Networks (KANs) for modeling orbital dynamics under perturbations, with a dual objective: achieving high predictive accuracy while enabling interpretable and auditable models.

While Newtonian gravity models accurately describes ideal two-body motion, real-world satellite trajectories are influenced by perturbations such as atmospheric drag, Earth's oblateness (J2), solar radiation pressure, and third-body effects [1–3]. These phenomena are typically approximated through simplified models that may fail to capture nonlinear interactions across regimes.

We develop a hybrid framework combining analytical physics with a data-driven KAN component that learns residual perturbation forces from simulated trajectories. Unlike standard neural networks, KANs provide a functional decomposition enabling direct inspection of relationships between physical variables (e.g., altitude, velocity) and perturbations [4–8].

The project delivers: (i) a hybrid orbital propagator, (ii) controlled comparisons with baseline models (MLPs and operator-learning approaches such as DeepONet [9]), and (iii) an analysis of interpretability and auditability of learned components.

A central contribution is the evaluation of **interpretability and auditability of learned physical models**, including alignment with known laws, identification of failure modes, and definition of trust criteria. This contributes to scientific machine learning [10, 11] and addresses broader questions at the interface of society and intelligent models regarding transparency and validation of AI-driven scientific methods.

3 Background Information & Problem Statement

Orbital motion is classically described by Newtonian dynamics, where gravitational acceleration is given by:

$$a = -\mu \frac{\dot{\mathbf{r}}}{|\mathbf{r}|^3} \tag{1}$$

with $\mu = GM$. While this formulation accurately models ideal two-body motion, real trajectories are affected by perturbations such as atmospheric drag, Earth's oblateness (J2), solar radiation pressure, and third-body interactions [1–3]. These effects are often approximated analytically, limiting their ability to capture nonlinear and regime-dependent behaviors.

Scientific machine learning seeks to complement physical modeling with data-driven approaches. Physics-informed neural networks (PINNs) [11] integrate differential equations into training, but face limitations such as training instability and spectral bias [10]. Operator-learning methods (e.g., DeepONet [9], Fourier Neural Operators [12]) improve scalability but typically sacrifice interpretability.

Kolmogorov–Arnold Networks (KANs) [10] provide an alternative representation by replacing fixed activations with learnable univariate functions, enabling direct inspection of learned relationships. Recent works (e.g., KINNs, KAN-ODEs) show promising results in scientific modeling, while highlighting challenges in stability and scalability [6, 7, 13–15].

This project addresses three key gaps: (i) accurate modeling of nonlinear perturbations, (ii) interpretable representations of learned physical relationships, and (iii) transparent validation of data-driven components.

Research question : Can Kolmogorov–Arnold Networks learn nonlinear perturbation forces in orbital dynamics while remaining interpretable and physically consistent within a hybrid framework?

We combine analytical dynamics with a learned residual model, and assess both predictive performance and the **interpretability and auditability of learned physical components**, contributing to trustworthy scientific AI. Importantly, the proposed approach is not limited to orbital mechanics and can be transferred to other physical systems, enabling broader adoption of auditable and trustworthy AI in scientific modeling.

4 Project Objectives & Concrete Implementation

We develop and evaluate a hybrid orbital dynamics model combining analytical physics with data-driven learning using KANs. The approach prioritizes a minimal but fully functional pipeline, with advanced components treated as optional extensions.

Objectives

- Develop a hybrid Newtonian–KAN modeling framework
- Learn key perturbations (e.g., drag, J2) from simulated data
- Perform controlled comparisons with MLP and operator-learning baselines
- Analyze interpretability and auditability of learned components
- Evaluate robustness and generalization across regimes

Deliverables

- Hybrid orbital propagator
- Reproducible simulation and dataset pipeline
- Experimental framework (code, configs, scripts, checkpoints)
- Evaluation and interpretability report
- Lightweight audit framework for validating learned physical models
- Final report, presentation, and demonstrator

All outputs are designed as a **reusable research and prototyping brick**. If the quality of the results warrants it, the outcomes will be consolidated into a scientific publication.

4.1 Implementation

Simulation & Dataset A synthetic pipeline generates trajectories across orbital regimes (LEO–GEO) with perturbations. Data includes state variables and derived features, with targets defined as total or residual acceleration. Splits enable generalization testing.

Models We implement:

- KAN (primary)
- MLP (baseline)
- simplified operator-learning model (optional)

Hybrid Model We explore two complementary strategies for modeling perturbations with Kolmogorov–Arnold Networks (KANs), both based on a progressive increase in dynamical complexity. Starting from ideal Newtonian trajectories (a_N), we incrementally introduce additional perturbations (J2, drag, solar radiation pressure), generating increasingly realistic trajectories: a_N, $a_{N,J2}$, $a_{N,J2,D}$, $a_{N,J2,D,S}$.

Approach 1 — Additive Residual Decomposition

In this approach, each physical effect is modeled explicitly through a dedicated KAN component trained on residual dynamics. The method follows a sequential residual learning strategy:

1. Learn Newtonian dynamics: $a_N \approx m_N$
2. Learn J2 residual: $a_{N,J2} - m_N \approx m_{J2}$
3. Learn drag residual: $a_{N,J2,D} - m_N - m_{J2} \approx m_D$
4. Learn solar radiation residual: $a_{N,J2,D,S} - m_N - m_{J2} - m_D \approx m_S$

This yields an additive hybrid model:

$$a_{\text{total}} = m_N + m_{J2} + m_D + m_S \tag{2}$$

This formulation provides a direct correspondence between model components and physical effects, enabling **fine-grained interpretability and auditability** of each contribution.

Approach 2 — Incremental Model Enrichment

Leveraging the flexible structure of KANs, this approach incrementally enriches a single model by adding new functional components (neurons) as complexity increases. Instead of decomposing effects explicitly, the model is progressively extended to capture additional dynamics:

1. Learn Newtonian dynamics: $a_N \approx m_N$
2. Extend model with J2: $a_{N,J2} \approx m_{N,J2}$
3. Extend model with drag: $a_{N,J2,D} \approx m_{N,J2,D}$
4. Extend model with solar radiation: $a_{N,J2,D,S} \approx m_{N,J2,D,S}$

This yields a unified model:

$$a_{\text{total}} = m_{N,J2,D,S} \tag{3}$$

While less explicitly decomposed, this approach enables a more compact representation and allows analysis of how learned functional components evolve as new physical effects are introduced.

Comparison and Expected Outcomes

The two approaches reflect different trade-offs between modularity and compactness. The additive formulation is expected to favor interpretability and explicit attribution of physical effects, while the incremental approach may provide improved efficiency and smoother integration of dynamics.

Comparing these strategies provides a controlled setting to evaluate how KAN architectures support **interpretable and auditable modeling of physical systems**, a central objective of the project. This comparison also serves as an ablation-style study on the role of model structure in interpretable scientific machine learning.

Evaluation

- metrics: MSE, MAE
- trajectory rollouts
- controlled comparisons and ablation-style evaluation
- generalization across regimes

Interpretability & Auditability

- visualization of learned functions
- analysis of physical relationships
- **consistency checks with known laws**
- identification of failure modes
- definition of trust criteria

Optional Extension If time permits, the proposed approach will be validated on publicly available real-world orbital datasets. This includes data sources such as satellite ephemerides accessible through public repositories (e.g., TLE-based datasets or NASA-provided data services). This additional validation step aims to assess the transferability of the model from controlled synthetic environments to real-world conditions.

5 Do you plan to deliver, as an outcome of your project, a reusable "brick" for the TRAIL Factory

Briefly describe what the brick would be and its intended users. The simulation framework developed for orbital dynamics is inherently modular and can be readily adapted to other physical systems governed by differential equations. The hybrid modeling approach—combining analytical laws with data-driven residual learning—applies broadly to domains such as fluid mechanics, heat transfer, and energy systems, where simplified physical models are often complemented by empirical corrections. In fluid dynamics, for instance, KAN-based residual models could learn turbulence closures or unmodeled flow interactions on top of Navier–Stokes approximations. Similarly, in energy systems (e.g., thermal networks, heat exchangers), the framework can capture complex nonlinear transfer phenomena that are difficult to model analytically. In both cases, the interpretability of KANs enables inspection of learned relationships, supporting model validation and engineering insight. More generally, this approach provides a pathway toward **auditable and trustworthy AI for industrial physics-based modeling**, where transparency, reliability, and integration with existing simulation pipelines are critical for adoption.

6 Project Dataset

The project relies on a fully synthetic dataset, ensuring immediate availability, full control, and no legal constraints.

Generation Using astrodynamics libraries (e.g., poliastro), trajectories are simulated across:

- orbit types: circular, elliptical
- regimes: LEO, MEO, GEO
- perturbations: drag, J2 (with optional extensions)

Structure

- inputs: position, velocity, derived features
- targets: total or residual acceleration

Data splits support generalization evaluation.

Experimental Readiness

- fast generation cycles
- progressive complexity
- compatibility with PyTorch pipelines

Data Governance & Reproducibility

- fully documented generation process
- reproducible datasets from code
- dataset versioning
- explicit analysis of modeling assumptions and limitations

This contributes to a **data auditability framework for scientific ML**.

Optional Validation Evaluation on public datasets (e.g., TLE-based) if time permits.

7 Detailed Work Plan

The project is structured into milestones with parallel work streams.

Milestone 1 — Setup, Simulation & Baselines (Days 1–3)

- setup environment and repository
- implement and validate simulation pipeline
- generate dataset
- train MLP baseline and initial KAN

Deliverables: dataset, pipeline, baseline results

Milestone 2 — Residual Learning & Hybrid Model (Days 3–7)

- compute perturbation residuals
- build both KANs architecture approaches
- integrate hybrid models
- initial trajectory validation

Deliverables: trained KAN, hybrid propagator

Milestone 3 — Evaluation & Interpretability (Days 7–9)

- quantitative evaluation
- generalization analysis
- interpretability analysis
- auditability assessment

Deliverables: evaluation results, interpretability outputs, audit protocol

Milestone 4 — Consolidation & Dissemination (Days 8–9)

- finalize codebase and documentation
- package reproducible pipeline
- prepare report and presentation
- build demonstrator
- optional real-data validation

Deliverables: reproducible repository, final report, demo

Execution Strategy & Risk Mitigation

- synthetic data ensures feasibility
- progressive modeling reduces complexity risk
- parallel tasks accelerate development
- early baselines guarantee fallback results

Optional components are explicitly decoupled from the core pipeline to prevent scope overload. The workflow is designed to support parallel work streams, enabling efficient pacing, rapid iteration, and asynchronous collaboration across the team.

8 Does the project include multidisciplinarity between STEM & SSH?

The project bridges Science/Technology and Social Sciences/Humanities by combining advanced scientific machine learning methods with a structured focus on interpretability, auditability, and trust in AI-:w driven modeling. On the technical side, it develops hybrid models that integrate physical laws with data-driven learning (KANs), enabling accurate simulation of complex dynamical systems. On the SSH side, it explicitly addresses how such models can be made transparent, interpretable, and verifiable, which are key concerns in domains where decisions rely on AI-assisted scientific outputs. This integration is operationalized through the development of an auditability framework that evaluates whether learned components align with known physical principles, remain stable across conditions, and avoid misleading patterns. By emphasizing validation protocols, reproducibility, and interpretability, the project contributes to broader SSH questions around trust, accountability, and the responsible use of AI in scientific and industrial contexts, ensuring that advanced modeling techniques remain understandable and usable by human experts.

References

[1] MIT OpenCourseWare. *Effect of Atmospheric Drag on Satellite Orbits*. 16.346 Astrodynamics lecture notes. Accessed 2026-04-14. 2008. URL: https://ocw.mit.edu/courses/16-346-astrodynamics-fall-2008/resources/lec_28/ (pages 1, 2).

[2] MIT OpenCourseWare. *Effect of J2 on a Satellite Orbit of the Earth*. 16.346 Astrodynamics lecture notes. Accessed 2026-04-14. 2008. URL: https://ocw.mit.edu/courses/16-346-astrodynamics-fall-2008/resources/lec_30/ (pages 1, 2).

[3] O. F. Graf Jr., ed. *Effects of solar radiation pressure on the orbit of the Solar Power Satellite*. Sept. 1977 (pages 1, 2).

[4] Ziming Liu et al. "KAN: Kolmogorov-Arnold Networks". In: *arXiv preprint arXiv:2404.19756* (2024) (page 1).

[5] Y. Wang et al. "Kolmogorov Arnold Informed Neural Network". In: *arXiv preprint arXiv:2406.11045* (2024) (page 1).

[6] Kürşat Tekbıyık, Güneş Karabulut Kurt, and Antoine Lesage-Landry. *PIKAN: Physics-Inspired Kolmogorov-Arnold Networks for Explainable UAV Channel Modelling*. 2026. arXiv: 2510.06355 [cs.LG]. URL: https://arxiv.org/abs/2510.06355 (pages 1, 2).

[7] B. C. Koenig et al. "KAN-ODEs: Kolmogorov-Arnold Network Ordinary Differential Equations for Learning Dynamical Systems and Hidden Physics". In: *arXiv preprint arXiv:2407.04192* (2024) (pages 1, 2).

[8] K. Shukla et al. "A comprehensive and FAIR comparison between MLP and KAN representations for differential equations and operator networks". In: *arXiv preprint arXiv:2406.02917* (2024) (page 1).

[9] Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. "Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators". In: *Nature Machine Intelligence* 3.3 (Mar. 2021), pp. 218–229. ISSN: 2522-5839. DOI: 10.1038/s42256-021-00302-5. URL: http://dx.doi.org/10.1038/s42256-021-00302-5 (pages 1, 2).

[10] Salvatore Cuomo et al. "Scientific machine learning through physics-informed neural networks: Where we are and what's next". In: *Journal of Scientific Computing* (2022) (pages 1, 2).

[11] George E. Karniadakis et al. "Physics-informed machine learning". In: *Nature Reviews Physics* 3 (2021), pp. 422–440 (pages 1, 2).

[12] Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. "Fourier Neural Operator for Parametric Partial Differential Equations". In: *International Conference on Learning Representations (ICLR)* (2021) (page 2).

[13] S. Somvanshi et al. "A Survey on Kolmogorov-Arnold Networks". In: *arXiv preprint arXiv:2411.06078* (2024) (page 2).

[14] Sifan Wang, Yujun Teng, and Paris Perdikaris. "Understanding and mitigating gradient flow pathologies in physics-informed neural networks". In: *SIAM Journal on Scientific Computing* (2021) (page 2).

[15] 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". In: *Journal of Computational Physics* 378 (2019), pp. 686–707 (page 2).