# Physics-Informed Neural Networks for Vehicle Lateral Dynamics modelling

Yuchuan Dong Rishikesh Vishnu Sivakumar

Zeekr Tech EU Chalmers University of Technology

ZEEKR

CHALMERS UNIVERSITY OF TECHNOLOGY

## Overview

Accurate real-time knowledge of a vehicle's lateral dynamics is pivotal for safe automated driving, precise motion control, energy-efficient control and robust trajectory planning.

In this study, we combined **Deep Neural Network** with **Physical Laws** to model Lateral Dynamics.

Our method accurately infers unobservable variables—including **lateral velocity** and **tyre coefficient**—and cuts state-estimation errors relative to purely data-driven method.

## Background:Model families

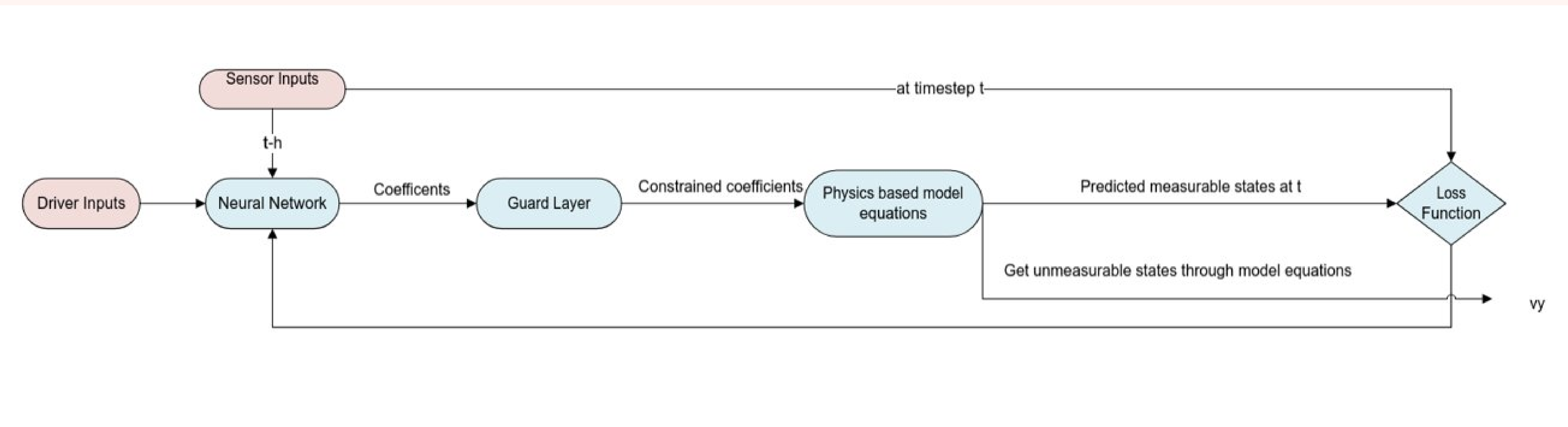
Existing approaches include:

- **Physics-based methods** (e.g., Bicycle model, Pacejka tyre formula)
  - \+ Transparent and interpretable
  - − Require accurate tyre and vehicle parameters (often unknown or hard to measure)
- **Purely data-driven methods** (stacked Long Short-Term Memory/Gated Recurrent Unit)
  - \+ Learn directly from data, capturing complex dynamics
  - − Black-box models without physical insights
  - − Infering unobservable states needs expensive measurement equipments and lot of vehicle tests(e.g., lateral velocity, tyre coefficient or friction.)
- **Physics-Informed Neural Network** (e.g. Deep Dynamics)
  - \+ Combines physics and learning effectively
  - − Relies on expensive and complicated to measure inputs (e.g., lateral velocity), reducing practical utility

**Our method** (*PhysAttenderNet*) integrates physics information and attention-based neural networks, enabling inference of latent states and achieving accurate, physically-consistent estimations.

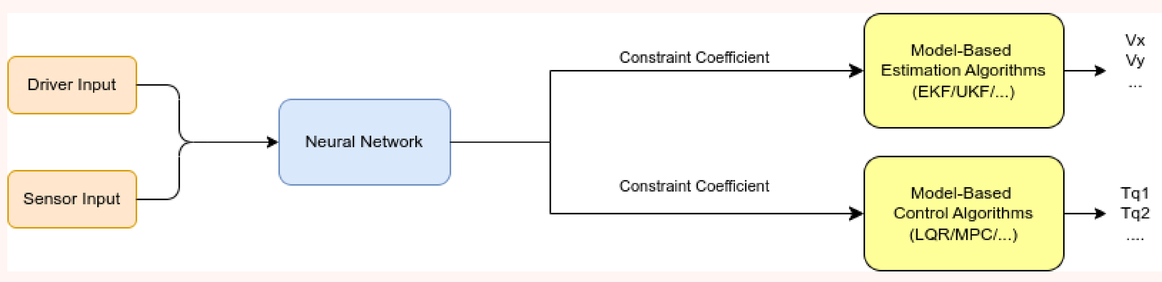
## Workflow & Method

Training Pipeline



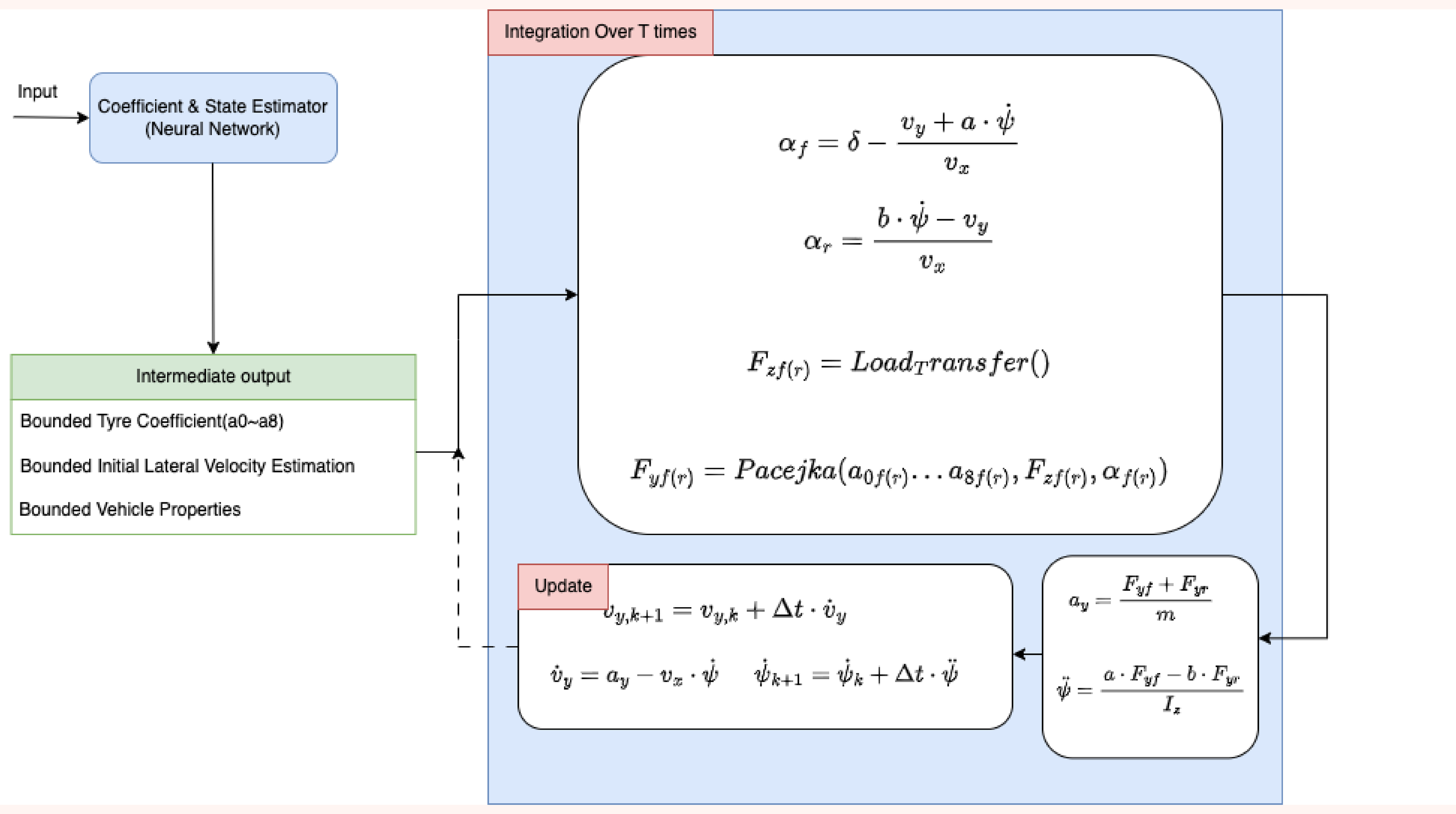

- **Sensor Measurements and Driver Commands** collected (IMU, wheel speeds, suspension travel, etc.)
- **Neural Network** predicts ODE parameters guided by domain knowledge constraints.
- **Single Track ODE Model** simulates vehicle dynamics.
- **Loss Function**: Minimizes difference between predicted and actual sensor states(Only take lateral acceleration and yaw rate).
- Outputs **unobservable states** (e.g., lateral velocity $v_y$), and enables estimation of model parameters.

Inference Pipeline



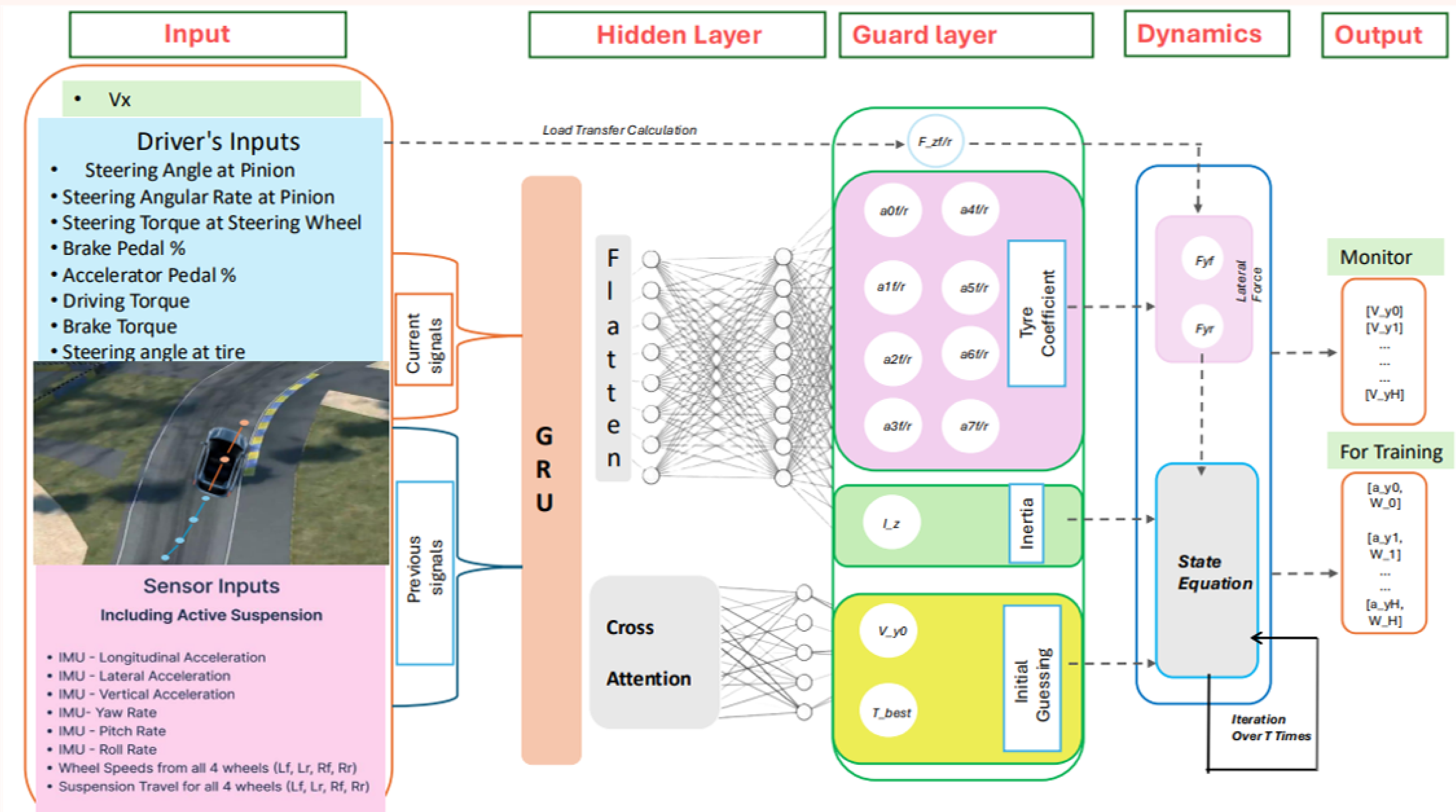

- **Estimate Constraint coefficient only**
- **Integration** into downstream estimators and controllers (e.g., UKF, MPC).

Method




## PhysAttenderNet

*Physics and Attention Attend into Network.*




- **Input**. 50-step window of driver commands + sensor signals. *What do we feed the net?*
- **GRU Encoder**. Learns temporal patterns. *How are past moments related?*
- **Cross-Attention**. Picks the most informative instants to make estimation. *When should we focus?*
- **Guard Layer**. Clamps coefficients to physics-valid ranges. *Do the numbers make physical sense?*
- **Dynamics Layer**. 3-DOF ODE + compensation inside the NN. *Can we merge physics and learning?*
- **Training Loss**. Only $a_y$ and $\dot{\psi}$ — no extra hardware needed. *Can we scale to fleet data?*
- **Outcome**. Real-time inference of $v_y$, tyre–road coefficient, etc. *What new states do we unlock?*

## Result & Evaluation

**Datasets**: Public-road "normal" driving and track "extreme" manoeuvres. All data were generated from CarMaker.

**Baselines**: Purely data-driven LSTM (green) vs *PhysAttenderNet* (yellow); ground truth in black.

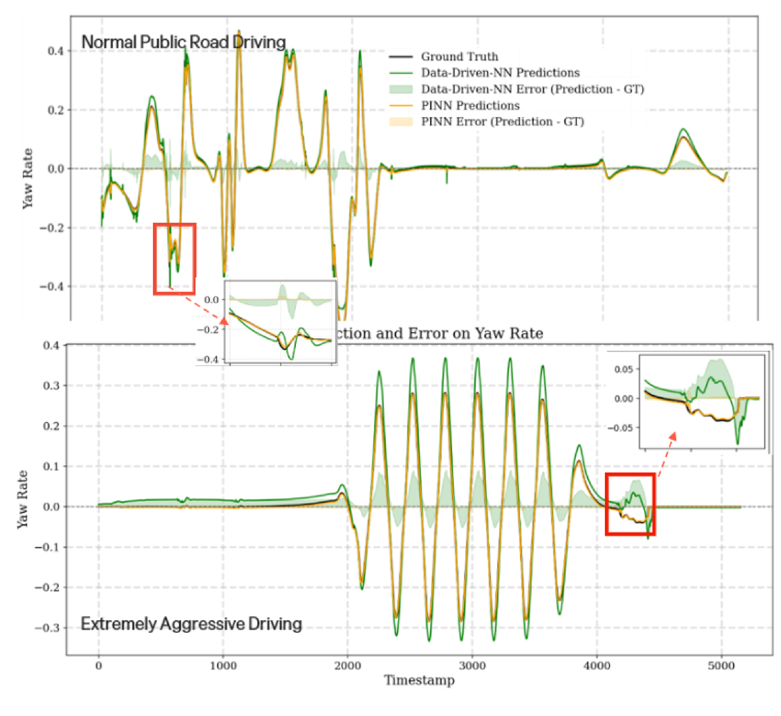



Figure 1. Result on YawRate

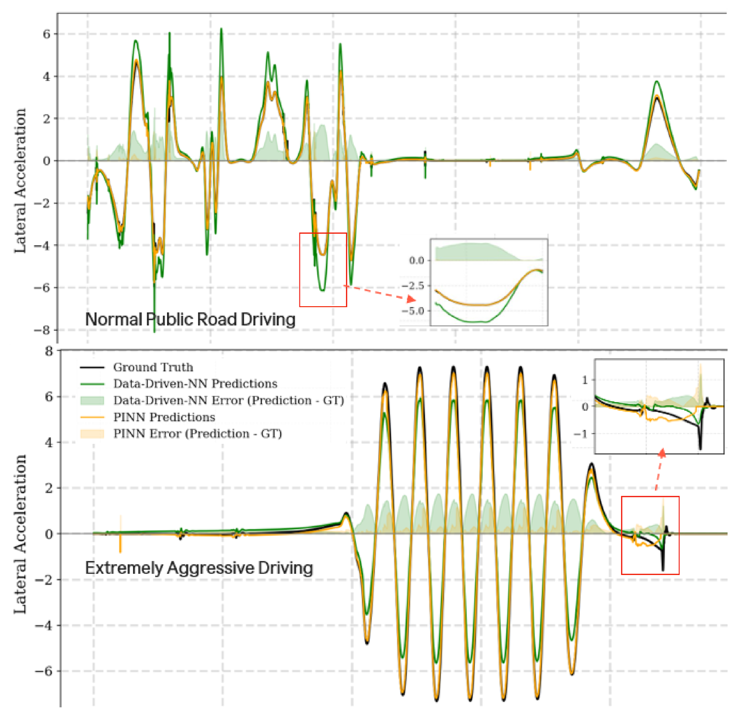



Figure 2. Result on Lateral Acceleration

- **Lateral acceleration** $a_y$: Lower RMSE and visibly narrower shaded error band across both datasets.
- **Yaw rate** $\dot{\psi}$: Comparable accuracy in normal driving; Data-driven model failed under aggressive inputs, while ours remains stable.
- **Take-away**: Physics constraints curb overfitting and deliver robust generalisation.

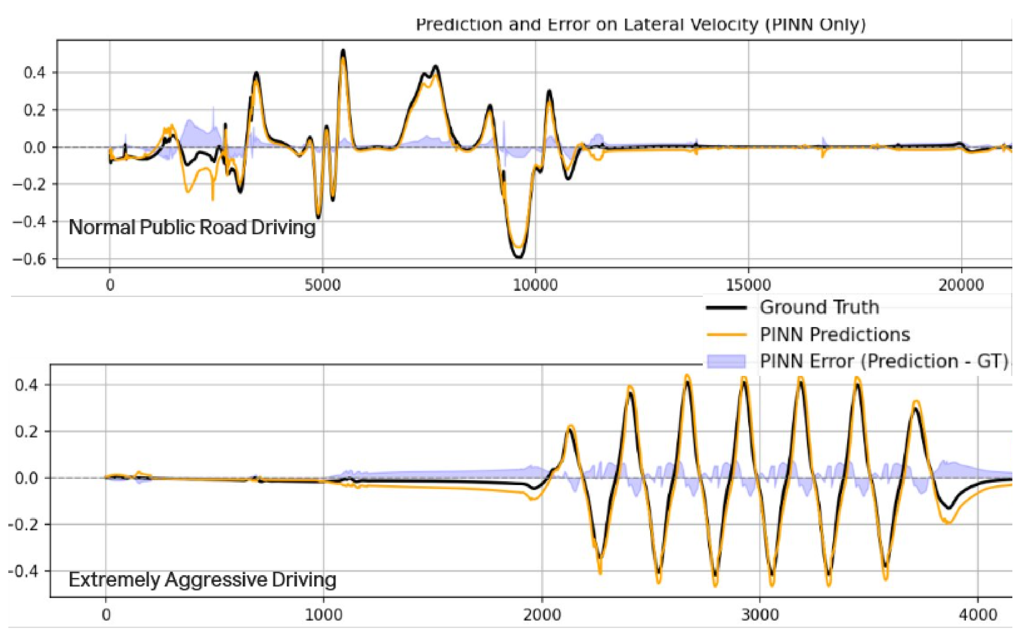



Figure 3. Result on Vy.

- **Challenge**: $v_y$ is *unobservable* with production sensors; no ground truth is available during training.
- **Purely data-driven models**: Cannot learn or output $v_y$ because the target signal is missing.
- **PhysAttenderNet**: Leverages physics constraints to – infer $v_y$ online – without ever seeing its label.

RMSE Comparison (Normal + Aggressive Driving)

| Model | Lateral Acc. $a_y$ (m/s$^2$) | Yaw Rate $\dot{\psi}$ (rad/s) | Lat. Vel. $v_y$ (m/s) |
|---|---|---|---|
| Purely Data-Driven (LSTM) | 0.493+0.648 | 0.0167+0.0303 | – |
| *PhysAttenderNet* | 0.0664+0.243 | 0.00257+0.0023 | 0.0389+0.0723 |

Table 1. Root Mean Square Error (RMSE) comparison. Lower is better. $v_y$ not available from data-driven model.

## References

1. H. B. Pacejka and E. Bakker, "The Magic Formula Tyre Model", Veh. Syst. Dyn., 1992.
2. J. Chrosniak et al., "Deep Dynamics", IEEE RA-L, 2024.
3. M. Raissi et al., "Physics Informed Deep Learning", arXiv:1711.10561, 2017.