Towards a Fully Differentiable Framework for Autonomous Driving Based on Model-Structured Neural Networks

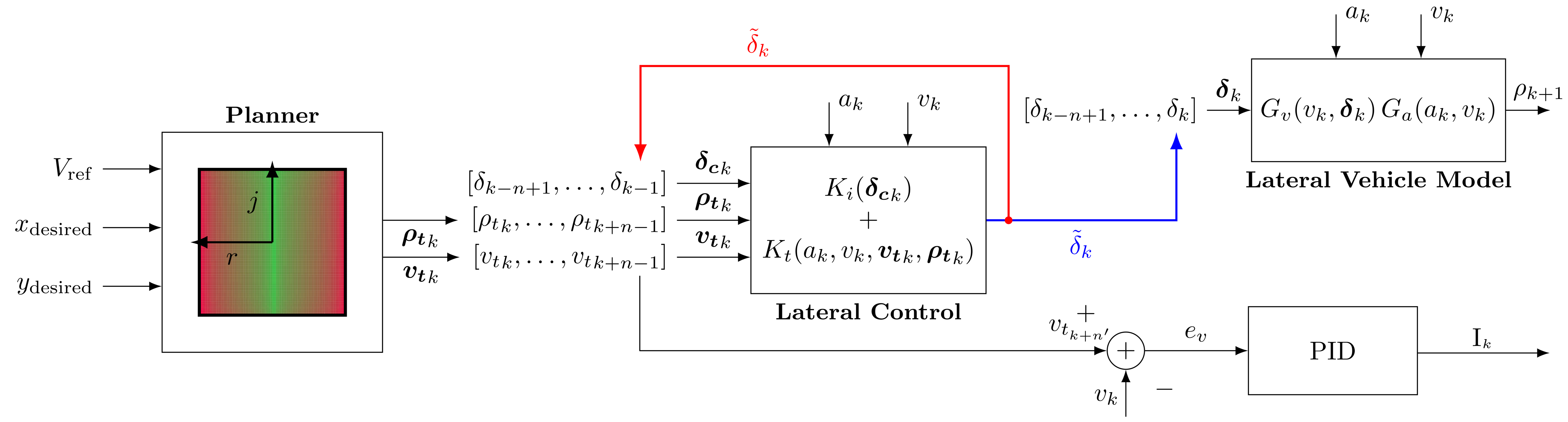
Sabrina Ciuffoletti, Gioele Defrancesco, Giovanni Scialla, Mattia Piazza, Sebastiano Taddei, Gastone Pietro Rosati Papini

GROUP

GitHub

ABSTRACT: Generalization in autonomous driving remains an open challenge, split between interpretable, safe model-based pipelines and flexible but unverified end-to-end learning methods. This work proposes a **hybrid framework** that bridges these paradigms through **MSNN**, laying the groundwork for a **fully differentiable architecture**. The system coordinates vehicle dynamics modeling, control, and motion planning within a unified pipeline: a structured neural forward model is identified from data, an inverse controller is synthesized via closed-loop mental simulation, and a modular motion planner interfaces directly with the control layer to enable future joint optimization. Supported by the **open-source nnodely** framework for MSNN design, the complete architecture is validated in real-world closed-loop experiments on an F1TENTH autonomous vehicle platform. Results demonstrate successful integration and highlight the potential of structured neural approaches for achieving robust, generalizable autonomous driving.

ARCHITECTURE OVERVIEW



LATERAL CONTROL

$$\rho(a_x, v_x, \delta) = \frac{\delta}{L + K_{us}(a_x) v_x^2}$$

Forward Model: $\rho_{k+1} = D_v(v_k, \boldsymbol{\delta}_k)\, F_a(a_k, v_x)$

$$D_v(v_k, \boldsymbol{\delta}_k) = \sum_{i=1}^{N_v} \phi_{v_i}(v_k)\,(\boldsymbol{w}_{\delta_i}^\top \boldsymbol{\delta}_k) \qquad F_a(a_k, v_k) = \sum_{i=1}^{N_a} \phi_{a_i}(a_k)\,\frac{1}{1 + A_i v_k^2}$$

Inverse Model: $\tilde{\delta}_k = K_t(a_k, v_k, \boldsymbol{v}_{t_k}, \boldsymbol{\rho}_{t_k}) + \boldsymbol{w}_{\delta_c}^\top \boldsymbol{\delta}_{c_k}$

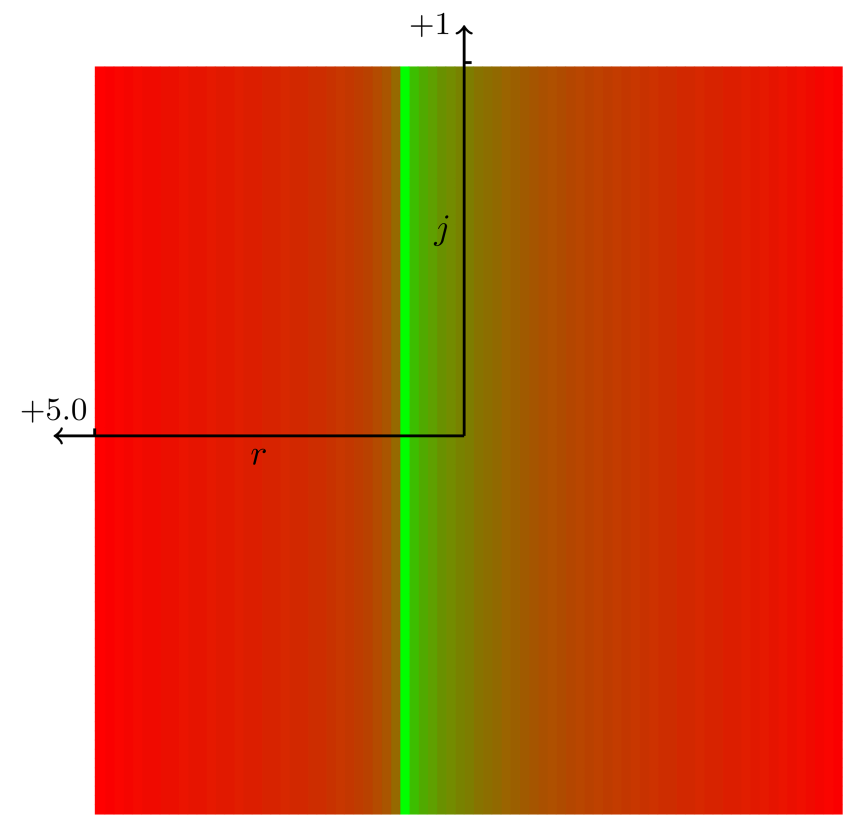
$$K_t(a_k, \boldsymbol{v}_{t_k}, \boldsymbol{\rho}_{t_k}) = \sum_{i=1}^{N_v} \phi_{v_i}(v_k)\,(\boldsymbol{w}_{\rho_i}^\top \nabla(a_k, \boldsymbol{v}_{t_k}, \boldsymbol{\rho}_{t_k}))$$

$$\nabla(a_k, \boldsymbol{v}_{t_k}, \boldsymbol{\rho}_{t_k}) = \boldsymbol{\rho}_{t_k} \odot \left(1 + \sum_{i=1}^{N_a} \phi_{a_i}(a_k)\, A_i\, \boldsymbol{v}_{t_k}^2\right)$$

PLANNING FRAMEWORK

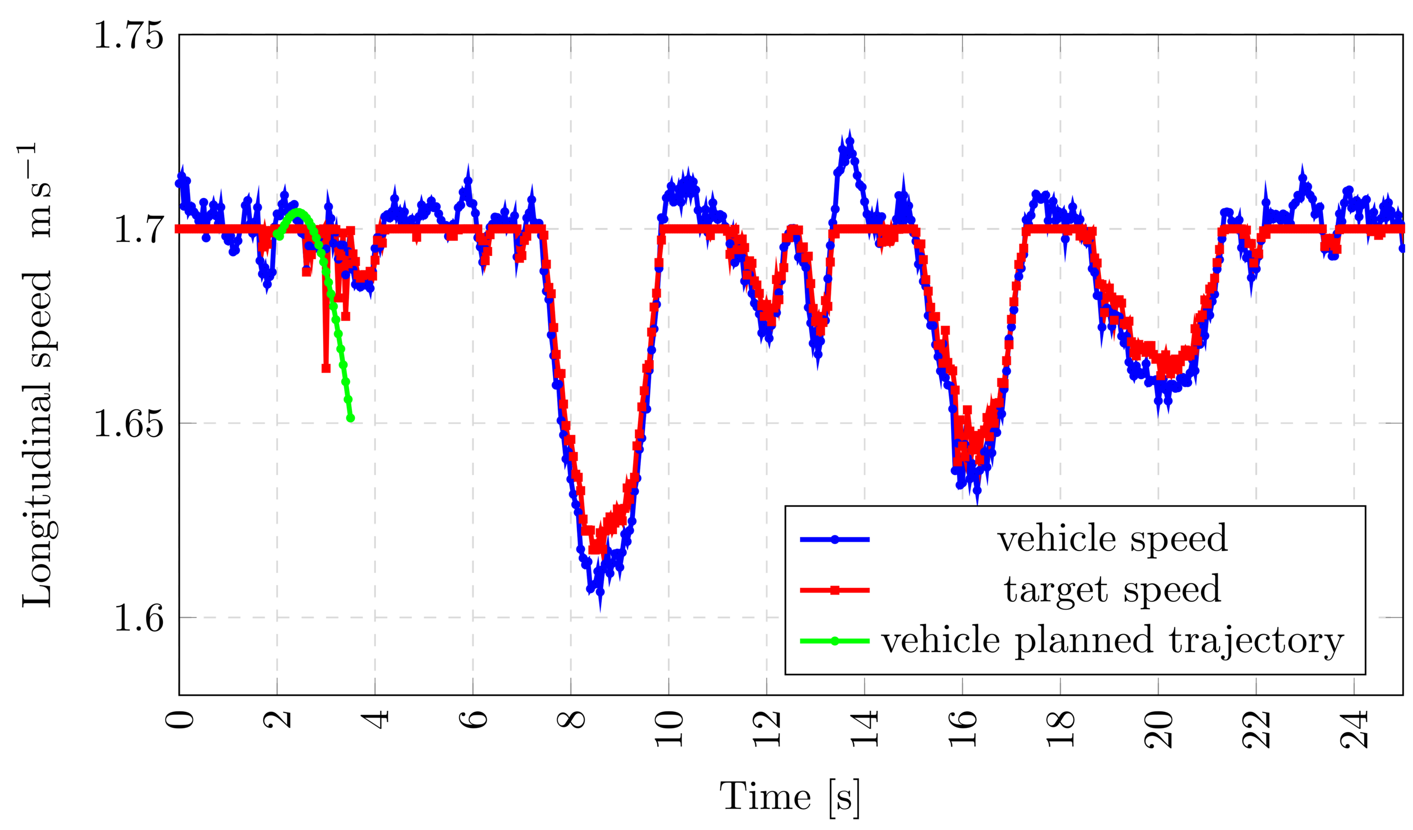
$$\underset{\mathbf{x}(t),\mathbf{u}(t)}{\text{minimize}} \int_0^{t_f} \mathbf{u}(t)^2\, dt \qquad \mathbf{u}(t) = j(t), r(t)$$

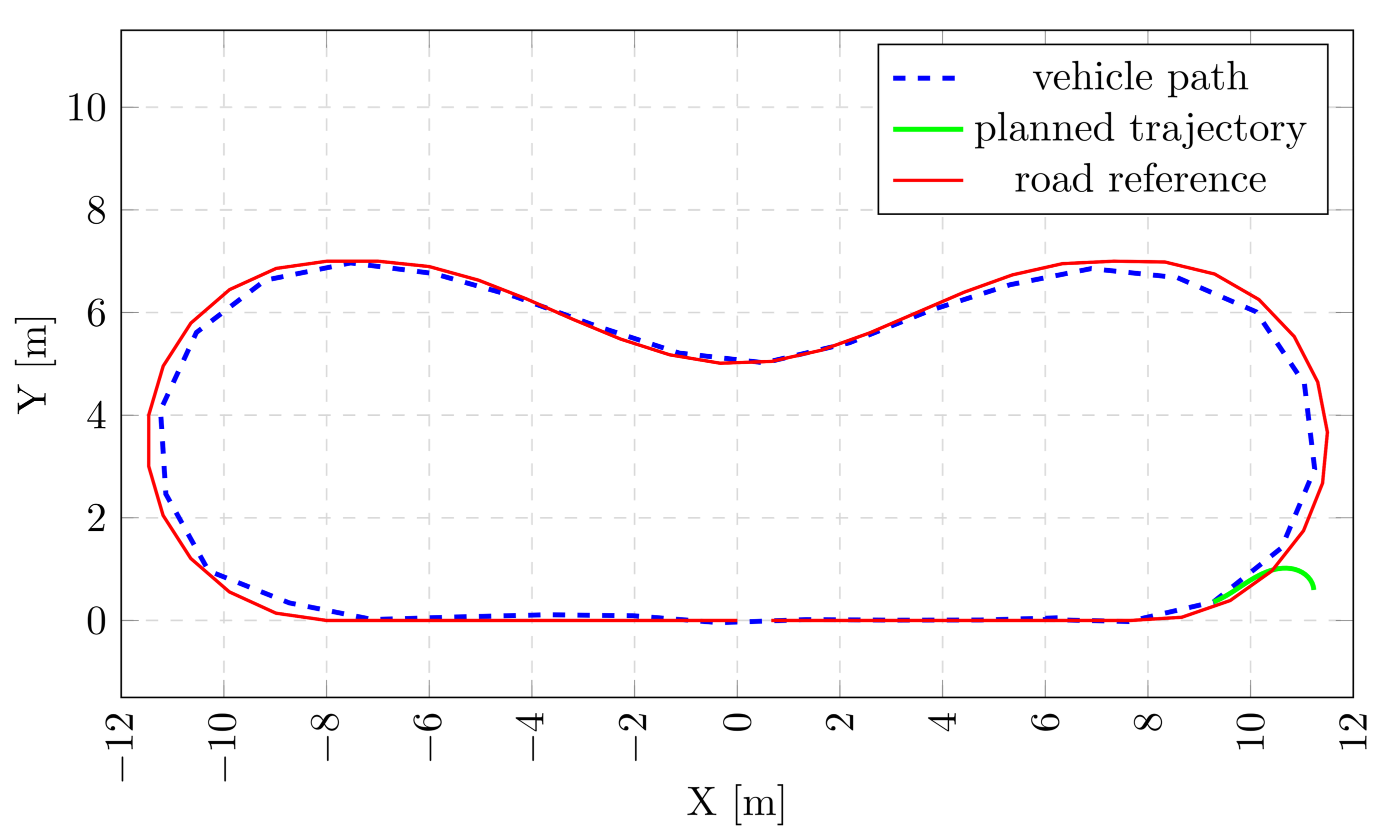
$$\text{subject to} \quad \dot{\mathbf{x}} = f(\mathbf{x}(t), \mathbf{u}(t)) \quad \mathbf{b}[\mathbf{x}(t_0), \mathbf{x}(t_f)] = 0$$



RESULTS

- **Hybrid planning and control framework**
- **Real-world Roboracer validation**
- **Differentiable low-level control**
- **Future end-to-end Neural ODE training**





Funded with the contribution of the Ministry of University and Research pursuant to Decree No. 1236 of 01/08/2023 – FIS 2 Call; project FIS-2023-03684 CUP: E53C24003800001 Structured neural network framework for modeling and control of autonomous systems - Neu4mes'