Identification-based prediction for event-triggered control in platooning

E. Gorski, I.-C. Morărescu , V. Satheeskumar Varma
Université de Lorraine, CNRS, CRAN,
Nancy, France
etienne.gorski@univ-lorraine.fr
constantin.morarescu@univ-lorraine.fr
vineethsvarma@gmail.com

L. Buşoniu
Technical University of Cluj-Napoca,
Cluj-Napoca, Romania
Corresponding Member of the Romanian Academy,
Bucharest, Romania
lucian@busoniu.net

***Abstract*—This paper presents a predictive event-triggered control strategy for vehicle platoons using cooperative adaptive cruise control. In contrast to the classical approach, which relies on continuous communication of control inputs between vehicles, we consider a framework where each vehicle intermittently transmits a predicted trajectory of its control signal according to a constant-threshold triggering rule. Data-based predictions are obtained using a recursive auto-regressive model with exogenous input, which identifies the local vehicle dynamics online and forecasts the future control actions. The framework guarantees bounded steady-state errors through Lyapunov analysis, excludes Zeno behavior, and aligns well with communication constraints. Numerical simulations demonstrate the reduction in communications of our identification-based approach compared to zero-order hold and model-based prediction approaches, while preserving safe spacing throughout emergency-braking and sustained perturbations.**

***Index Terms*—Platooning, Event-triggered control, System identification**

I. Introduction

Connected and automated vehicles improve traffic efficiency, safety, and energy use. Vehicle platooning, where vehicles travel in close formation with coordinated acceleration and braking, increases road throughput and reduces aerodynamic drag, resulting in fuel savings [1]. Safe platoon operation relies on accurate coordination, depending on both control design and inter-vehicle communication. Cooperative Adaptive Cruise Control (CACC) has emerged as the standard strategy for connected vehicle platooning, using vehicle-to-vehicle communication to share control information across the formation [2], [3]. Cooperation allows each vehicle to anticipate its predecessor's actions, improving tracking performance and enabling smaller inter-vehicle gaps. However, traditional CACC implementations assume continuous or high-frequency communication, which can overwhelm network bandwidth and increase latency, particularly in large platoons [4].

Event-Triggered Control (ETC) addresses these limitations by transmitting updates only when a monitoring error exceeds a prescribed threshold [5], reducing network load while preserving stability. Despite growing interest in event-triggered CACC, most existing approaches rely on simple reconstruction strategies where the received signal is held constant between transmissions [6], [7]. Although straightforward to implement, such methods lead to rapid prediction error growth whenever control signals vary significantly, resulting in frequent triggering events that limit communication savings. Predictive event-triggered control offers an alternative by transmitting predictions of future signal evolution [8].

In previous work [9], we designed a framework for predictive event-triggered CACC with constant-threshold triggering, where model-based prediction using nominal vehicle dynamics was introduced. In this paper, we develop a new data-driven prediction approach based on online system identification [10], using recursive Auto-Regressive with eXogenous input (ARX) modeling. Each follower vehicle continually learns the effective dynamics of its predecessor through recursive estimation, and uses this learned model to generate multi-step-ahead predictions of the control trajectory. Each vehicle's parameterized model relates its predecessor's control input to past values and to the control signal received from further upstream vehicles. At each sampling instant, the Recursive Least-Squares (RLS) algorithm updates the model parameters to minimize prediction error based on recent observations. Unlike methods that require accurate prior knowledge of vehicle models [11], [12], our approach adapts online to observed behavior, maintaining prediction accuracy even under model uncertainties, parameter variations, and external disturbances.

The remainder of this paper is organized as follows. Section II presents the vehicle platoon model and establishes the event-triggered communication framework. Section III details the system-identification-based predictor and the baselines. Section IV presents numerical results validating the framework. Finally, Section V summarizes the contributions and outlines future research directions.

II. Framework

A. Setup and Dynamics

We consider a platoon composed of N homogeneous vehicles indexed by $i \in \{1, \dots, N\}$, which follow a leading vehicle indexed by $i = 0$. The goal of each vehicle i is to follow its predecessor $i-1$ while maintaining a safe inter-vehicle distance defined by a constant time-gap policy. More precisely, the desired inter-distance is given by $d_{r,i}(t) = r + hv_i(t)$, where r is the minimum safety margin, $h > 0$ the desired time-gap, and $v_i(t)$ the velocity of vehicle i at time t.

This work was been financially supported by the project DECIDE, no. 57/14.11.2022, funded under the PNRR I8 scheme by the Romanian Ministry of Research, Innovation and Digitisation; and also partially funded by ANR under the grant COMMITS No. ANR-23-CE25-0005.

The vehicle spacing error of vehicle i is defined as $e_i(t) = q_{i-1}(t) - q_i(t) - hv_i(t) - r$, where $q_i(t)$ denotes the position of vehicle i. Each vehicle is modeled as a third-order dynamical system with input filtering, according to [6]. Specifically, the longitudinal dynamics of vehicle $i \geq 1$ are described by

$$\dot{e}_i(t) = v_{i-1}(t) - v_i(t) - ha_i(t), \tag{1a}$$

$$\nu_i : \begin{cases} \dot{v}_i(t) = a_i(t), \\ \dot{a}_i(t) = -\frac{1}{\tau}a_i(t) + \frac{1}{\tau}u_i(t), \end{cases} \tag{1b}$$

$$\dot{u}_i(t) = -\frac{1}{h}u_i(t) + \frac{1}{h}\chi_i(t), \tag{1c}$$

where $a_i(t)$ is the acceleration, $u_i(t)$ the desired acceleration, $\chi_i(t)$ the control input, and $\tau > 0$ the inertia constant, common for all vehicles. The filter's dynamics (1c) implement a pre-compensation of the time-gap policy with transfer function $H(s) = hs + 1$, whose inverse cancels the delay induced by h in the control loop. For the leading vehicle $i = 0$, the desired acceleration $u_0(t)$ is an exogenous input representing the leader's driving commands.

B. Control Architecture

In the CACC scheme, the control input of vehicle i

$$\chi_i(t) = k_p e_i(t) + k_d \dot{e}_i(t) + \hat{u}_{i-1}(t), \tag{2}$$

combines a feedback term based on local sensors that provide measurements of the spacing error $e_i(t)$ where $k_p, k_d > 0$ are controller gains, and a feedforward term based on the estimated predecessor's desired acceleration $\hat{u}_{i-1}(t)$. The latter is obtained through a communication network that connects consecutive vehicles. This term plays an important role: it allows vehicle i to anticipate the actions of its predecessor, thereby improving tracking performance and enabling tighter formations. However, conventional CACC implementations requires high-frequency communication, which can lead to network congestion and increased energy consumption.

C. Predictive Event-Triggered Framework

To reduce communication load while preserving stability, we employ an ETC strategy where transmissions occur only when necessary based on the reconstruction error $e_{u,i-1}(t) = \hat{u}_{i-1}(t) - u_{i-1}(t)$, which measures the mismatch between the reconstructed signal $\hat{u}_{i-1}(t)$ used by vehicle i and the actual signal $u_{i-1}(t)$ generated by vehicle $i-1$. We adopt a constant-threshold triggering rule: a new transmission occurs whenever

$$\|e_{u,i-1}(t)\| \geq \varepsilon, \tag{3}$$

where $\varepsilon > 0$ is a prescribed design parameter. At each triggering instant t_k, the transmission resets the error to zero: $e_{u,i-1}(t_k) = 0$. Between consecutive triggering instants $t \in [t_k, t_{k+1})$, our method departs from classical approaches where the reconstructed signal is held constant. Instead, we transmit additional information at each triggering instant that enables vehicle i to reconstruct the anticipated evolution of its predecessor's control input. Specifically, the signal $\hat{u}_{i-1}(t)$ evolves as follows

$$\dot{\hat{u}}_{i-1}(t) = f_{i-1}(t - t_k), \quad t \in [t_k, t_{k+1}), \tag{4}$$

where f_{i-1} is a prediction function that depends on the reconstruction strategy. Different strategies will be detailed in Section III.

This time-varying reconstruction offers several advantages:

- It reduces the mismatch between $u_{i-1}(t)$ and $\hat{u}_{i-1}(t)$ and so extends inter-event times, thereby improving control performance even with sparse transmissions.
- It allows communication network faults to more easily be taken into account: delays, packet loss, quantization, etc.
- It provides a framework that can accommodate different levels of model knowledge, from simple non-predictive baselines to adaptive data-driven methods.

For analytical purposes, we express the closed-loop dynamics in state-space form. For each follower $i \geq 1$, we define the state vector $x_i := (v_{i-1} - v_i, a_{i-1}, u_{i-1}, e_i, a_i, u_i)^\top \in \mathbb{R}^6$ comprising the relative velocity $v_{i-1} - v_i$, the predecessor's acceleration a_{i-1} and desired acceleration u_{i-1}, the spacing error e_i, and vehicle i's acceleration a_i and desired acceleration u_i. The closed-loop dynamics is expressed as

$$\begin{cases} \dot{x}_i(t) = Ax_i(t) + B\chi_{i-1}(t) + Ee_{u,i-1}(t), \\ \dot{e}_{u,i-1}(t) = Dx_i(t) - \frac{1}{h}\chi_{i-1}(t) + f_{i-1}(t - t_k), \\ \chi_i(t) = Cx_i(t) + e_{u,i-1}(t). \end{cases} \tag{5}$$

D. Practical Individual Stability Requirement

The introduction of the constant-threshold triggering rule (3) changes the stability properties of the closed-loop system. To analyze this, consider first the nominal case without predecessor input ($\chi_{i-1} = 0$) and without communication error ($e_{u,i-1} = 0$), in which system (5) reduces to $\dot{x}_i = Ax_i$. Individual stability then requires A to be Hurwitz, which by Routh's criterion is equivalent to the gain condition $k_d > \tau k_p$. When communication errors are reintroduced through the triggering rule (3), asymptotic convergence to the origin can no longer be guaranteed. Instead, the platoon must satisfy a practical individual stability property: in the absence of predecessor input, each vehicle's spacing error $e_i(t)$ must converge to a bounded neighborhood of zero, ensuring that small perturbations do not lead to unbounded error growth.

Theorem 1 (Practical Individual Stability): Consider system (5) with A Hurtwitz, $\chi_{i-1} = 0$ and the constant-threshold triggering rule (3). Suppose that there exist $P = P^\top \succ 0$ and $\eta > 0$ such that

$$\begin{bmatrix} A^\top P + PA & PE \\ E^\top P & -\frac{1}{\eta}I \end{bmatrix} \prec 0. \tag{6}$$

And define the threshold $\rho := \sqrt{\frac{1}{\eta|\lambda_{\max}|}}$ with $\lambda_{\max}$ the maximum eigenvalue of $A^\top P + PA + \eta PEE^\top P$. Then all trajectories converge asymptotically to the invariant set $\mathcal{S}_\rho = \{\|x_i\| \leq \rho\varepsilon\}$.

Proof: Consider the candidate Lyapunov function $V(x_i) = x_i^\top P x_i$, where $P = P^\top \succ 0$ is a positive definite matrix. Its time derivative along the trajectories of system (5) is

$$\dot{V}(x_i) = x_i^\top (A^\top P + PA)x_i + 2x_i^\top PB\chi_{i-1} + 2x_i^\top PEe_{u,i-1}.$$

To establish practical stability in the absence of predecessor input ($\chi_{i-1} = 0$), we seek conditions under which $\dot{V}$ is negative definite outside a bounded region. Using the inequality

$$2x_i^\top PEe_{u,i-1} \leq \eta \|PE\|^2 \|x_i\|^2 + \frac{1}{\eta}\|e_{u,i-1}\|^2,$$

for any $\eta > 0$, we obtain

$$\dot{V}(x_i) \leq x_i^\top \left(A^\top P + PA + \eta PEE^\top P\right) x_i + \frac{1}{\eta}\|e_{u,i-1}\|^2.$$

Suppose that LMI (6) is guaranteed. Then, defining the eigenvalue $\lambda_{\max} := \lambda_{\max}(A^\top P + PA + \eta PEE^\top P) < 0$:

$$\dot{V}(x_i) \leq \lambda_{\max}\|x_i\|^2 + \frac{1}{\eta}\varepsilon^2,$$

where we used the triggering condition $\|e_{u,i-1}(t)\| \leq \varepsilon$. From there, it follows that $\dot{V}(x_i) < 0$ whenever $\|x_i\| > \sqrt{\frac{\varepsilon^2}{\eta|\lambda_{\max}|}} = \rho\varepsilon$, which implies that trajectories starting outside the set $\mathcal{S}_\rho = \{\|x_i\| \leq \rho\varepsilon\}$, enter it in finite time and remain inside thereafter. Since V is radially unbounded, all trajectories remain bounded and converge to $\mathcal{S}_\rho$ as $t \to \infty$.

Remark 1: Scalar ρ quantifies the sensitivity of the closed-loop system to communication errors. The value of ρ depends on the system dynamics (through A, P) and the tuning parameter η. Minimizing ρ over feasible values of η yields the tightest invariant set for a given threshold ε.

Remark 2: The constant threshold ε induces a trade-off between performance and communication. A smaller ε reduces the practical stability bound $\rho\varepsilon$, improving spacing accuracy at the cost of more frequent transmissions. Conversely, a larger ε decreases the communication rate but enlarges the steady-state deviation from the desired inter-vehicle distance. In practice, ε should be chosen such that $\rho\varepsilon$ remains well below the minimum safety margin r to ensure collision-free operation.

Remark 3: The constant-threshold rule (3) excludes Zeno behavior, where infinitely many triggering events occur in finite time. At each triggering instant t_k, the error resets to zero, and then grows continuously according to $\dot{e}_{u,i-1}(t) = f_{i-1}(t - t_k) - Dx_i(t) + \frac{1}{h}\chi_{i-1}(t)$. Since x_i, χ_{i-1}, and f_{i-1} are continuous and bounded, the error derivative remains bounded by some Lipschitz constant $L > 0$, implying that there exists a strictly positive minimum inter-event time such as $t_{k+1} - t_k \geq \varepsilon/L > 0$. More accurate predictions that closely track u_{i-1} yield smaller L, thereby increasing the minimum inter-event time.

III. Prediction Strategies for Event-Triggered CACC

In this section, we take the perspective of vehicle i, which constructs and sends the prediction to its follower $i+1$. At each triggering instant t_k, the error is reset to zero and vehicle i sends updated information $\hat{u}_i(t)$ to vehicle $i+1$. The challenge is to predict how $u_i(t)$ will evolve over the interval $[t_k, t_{k+1})$ so that the error $e_{u,i}(t) = \hat{u}_i(t) - u_i(t)$ grows slowly, ensuring a large period between consecutive triggering instants. This section presents three strategies to design $\hat{u}_i(t)$: Zero-Order Hold (ZOH) as a non-predictive baseline [6], a model-based predictive baseline exploiting known nominal dynamics [9], and our new system identification-based prediction through online learning. Figure 1, showing actual data from the 2nd follower simulation of Section IV, compares the behavior of the three strategies over a representative inter-event interval.

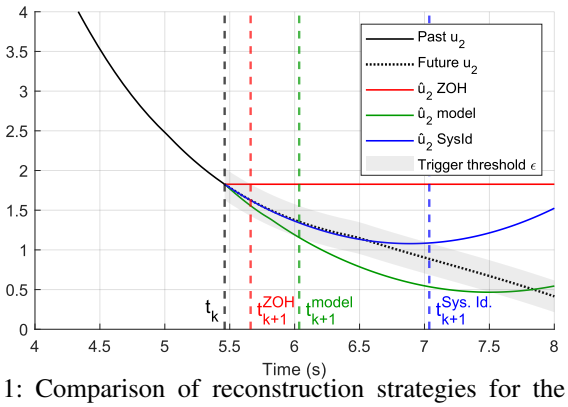

Fig. 1: Comparison of reconstruction strategies for the 2nd follower between triggering instants t_k and t_{k+1}.

A. Non-Predictive Baseline Zero-Order Hold

The simplest strategy maintains the last transmitted value constant: $\hat{u}_i(t) = u_i(t_k)$, $t \in [t_k, t_{k+1})$. This approach requires no model knowledge and minimal computational effort. Upon reception of $u_i(t_k)$, vehicle $i+1$ stores this value and uses it directly in the control law (2). However, since $\hat{u}_i(t)$ remains constant while the true signal $u_i(t)$ continues to evolve, the error grows as $e_{u,i}(t) = -\int_{t_k}^{t} \dot{u}_i(s)\, ds$. Whenever $u_i(t)$ varies significantly during acceleration or braking, the error grows rapidly. As shown by the red curve in Figure 1, the Zero-Order Hold (ZOH) reconstruction quickly deviates from the true signal, causing the threshold ε to be reached at the earliest time t_{k+1}^{ZOH} among all three methods. Despite its limitations, ZOH serves as a natural baseline for evaluating the performance gains of predictive strategies.

B. Model-Based Predictive Baseline

The model-based approach exploits knowledge of nominal vehicle dynamics to simulate the expected evolution of $u_i(t)$ forward in time. This method, explored in our previous work [9], provides a foundation for comparison with adaptive approaches. Under ideal conditions where communication errors are absent, the dynamics of vehicle i follow the same structure as (1) but use the received prediction $\hat{u}_{i-1}(t)$ instead of the actual signal. We define the nominal sub-state $\hat{x}_i = (v_{i-1} - v_i, a_{i-1}, e_i, a_i, u_i)^\top$, which evolves according to the nominal model

$$\dot{\hat{x}}_i(t) = \hat{A}\hat{x}_i(t) + \hat{B}\hat{u}_{i-1}(t), \quad \hat{u}_i(t) = \hat{C}\hat{x}_i(t). \tag{7}$$

At transmission instant t_k, vehicle i integrates this system forward over a prediction horizon T_p. When the actual vehicle behavior matches this nominal model, the prediction closely tracks the true signal. As shown by the green curve in Figure 1, the predicted trajectory follows the actual evolution more accurately than ZOH, resulting in slower error growth and a

later triggering time $t_{k+1}^{\text{model}} > t_{k+1}^{\text{ZOH}}$. However, performance depends on model accuracy. Discrepancies arise from parametric uncertainties such as variations in τ, unmodeled disturbances like road grade or wind resistance, and actuator nonlinearities including saturation and rate limits.

C. System Identification-Based Prediction

The method that we propose in this paper addresses the limitations of model-based prediction by designing $\hat{u}_i(t)$ through online system identification. Rather than relying on a fixed nominal model, vehicle i continuously learns the effective dynamics of its predecessor from observed data and adapts its predictor accordingly. This adaptive mechanism allows the predictor to capture not only nominal dynamics but also unmodeled disturbances like road-grade variations or wind perturbations.

1) Simplified Dynamics and ARX Model Structure: We treat the design problem as a forecasting task, relating past values of $u_i(t)$ to the received prediction $\hat{u}_{i-1}(t)$. Assuming that the received prediction closely approximates the actual input and that spacing errors e_i remain small under effective CACC operation, from dynamics (1) we obtain

$$u_i(t) + h\dot{u}_i(t) \approx \hat{u}_{i-1}(t). \tag{8}$$

This relationship, which reflects the pre-compensation filter in the CACC architecture, suggests an ARX structure [10]. For online identification, we discretize with sampling period T_s and adopt a finite-difference approximation $\dot{u}_i(kT_s) \approx [u_i(k) - u_i(k-1)]/T_s$, yielding

$$u_i(k) \approx \frac{h}{h+T_s} u_i(k-1) + \frac{T_s}{h+T_s} \hat{u}_{i-1}(k). \tag{9}$$

To capture higher-order dynamics and improve prediction accuracy, we use to an ARX structure with orders (n_a, n_b, n_k):

$$u_i(k) + \sum_{j=1}^{n_a} a_j\, u_i(k-j) = \sum_{j=1}^{n_b} b_j\, \hat{u}_{i-1}(k-n_k-j+1) + e(k), \tag{10}$$

where $e(k)$ represents residual effects due to modeling inaccuracies, spacing errors, and prediction errors. There, n_a is the order of past outputs, n_b is the order of past inputs, and n_k is the input delay.

2) Recursive Parameter Estimation: Rewrite (10) in regression form as $u_i(k) = \varphi^\top(k)\theta_i + e(k)$, where

$$\begin{aligned}\varphi(k) = [&-u_i(k-1) \cdots - u_i(k-n_a) \\ &\hat{u}_{i-1}(k-n_k) \ldots \hat{u}_{i-1}(k-n_k-n_b+1)]^\top\end{aligned} \tag{11}$$

is the regressor vector and $\theta_i = [a_1 \cdots a_{n_a}\, b_1 \cdots b_{n_b}]^\top$ is the parameter vector. Vehicle i continuously updates the parameter estimate using the RLS algorithm:

$$\theta_i(k) = \theta_i(k-1) + K(k)\left[u_i(k) - \varphi^\top(k)\theta_i(k-1)\right], \tag{12}$$

$$K(k) = \frac{P(k-1)\varphi(k)}{\lambda + \varphi^\top(k)P(k-1)\varphi(k)}, \tag{13}$$

$$P(k) = \frac{1}{\lambda}\left[P(k-1) - K(k)\varphi^\top(k)P(k-1)\right], \tag{14}$$

where $P(k)$ is the covariance matrix, $K(k)$ is the gain vector, and $\lambda \in (0, 1]$ is a forgetting factor that enables tracking of time-varying dynamics. This identification operates continuously in the background, while communication is governed by the event-triggering rule. When the prediction error remains small, the estimated model provides accurate forecasts and no transmission is required. When a disturbance occurs, the mismatch increases, triggering communication and simultaneously enriching the data used by the RLS estimator. For the latter to estimate a good model, the regressor must be persistently exciting, a condition naturally satisfied during transient acceleration or braking maneuvers and speed adjustments. During steady-state cruising, parameter convergence is less critical because the triggering mechanism ensures that errors remain bounded regardless of parameter accuracy.

3) Multi-Step-Ahead Prediction: Once the parameters are updated, vehicle i generates an N_p-step-ahead prediction:

$$\begin{aligned}\hat{u}_i(k+j|k) = &- \sum_{m=1}^{\min(n_a, j-1)} \hat{a}_m(k)\, \hat{u}_i(k+j-m|k) \\ &- \sum_{m=j}^{n_a} \hat{a}_m(k)\, u_i(k+j-m) \\ &+ \sum_{m=1}^{n_b} \hat{b}_m(k)\, \hat{u}_{i-1}(k+j-n_k-m+1),\end{aligned} \tag{15}$$

for $j = 1, \ldots, N_p$, where predicted values $\hat{u}_i(k+j-m|k)$ are used for $j - m \geq 1$ and measured values $u_i(k+j-m)$ for $j - m \leq 0$. These discrete samples $\{\hat{u}_i(k+j|k)\}_{j=1}^{N_p}$ are then converted into a continuous-time profile $\hat{u}_i(t)$ over $[t_k, t_k + N_p T_s]$ using piecewise-linear interpolation. Beyond the prediction horizon, the signal is extrapolated using a spline, in order to provide coverage over $[t_k, +\infty)$. Having a prediction over a large enough time interval is necessary for the next vehicle $i+1$ to perform its own prediction and identification procedure without requiring immediate communication updates. As illustrated by the blue curve in Figure 1, identification-based prediction tracks the actual signal evolution, delaying the triggering instant to $t_{k+1}^{\text{SysId}} > t_{k+1}^{\text{model}} > t_{k+1}^{\text{ZOH}}$.

The leader vehicle constitutes a special case since it does not receive any predicted input from a predecessor. For the leader, prediction reduces to a pure autoregressive model $u_0(k) + \sum_{j=1}^{n_a} a_j\, u_0(k-j) = e(k)$, which performs time-series forecasting based solely on the leader's own past control inputs. The recursive estimation follows the same RLS structure with a simplified regressor $\varphi_0(k) = [-u_0(k-1) \cdots - u_0(k-n_a)]^\top$.

Remark 4: From a practical point of view, the identification and prediction scheme can be realized using standard system identification tools. For instance, in MATLAB, the recursive ARX model can be updated inside flow dynamic using new input–output data via the `recursiveARX` routine. After updating θ_i, multi-step ahead predictions of the control input can be generated using the `forecast` function.

D. Integration with the Event-Triggered Framework

The three strategies produce continuous-time predictions $\hat{u}_i(t)$ over the inter-event interval. To integrate these into the event-triggered control framework of Section II, we need the derivative $\dot{\hat{u}}_i(t) = f_i(t - t_k)$ as specified in (4). For the ZOH strategy, $\hat{u}_i(t)$ is constant, so $f_i(t - t_k) = 0$. For the model-based approach, system (7) yields $f_i(t - t_k) = \hat{C}\left(\hat{A}\hat{x}_i(t) + \hat{B}\hat{u}_{i-1}(t)\right)$. Finally, for the identification-based method, $f_i(t - t_k)$ is obtained by numerical differentiation of the interpolated continuous-time profile.

E. Transmission of the Predicted Input

Transmitting f_i as a continuous-time signal is not feasible in practice due to the payload and rate limitations of vehicle-to-vehicle communication standards such as IEEE 802.11p. To comply with these constraints, only a compact representation of the predicted input is transmitted at each triggering instant t_k. Specifically, vehicle i sends K samples that best describe the predicted profile for a weight of a few bytes for the ZOH case (1 sample) and a few hundred for the more advanced cases. In particular, for our system-identification predictor, if samples are transmitted using piecewise-linear interpolation, exactly two numbers per sample are required, plus one number to encode the sampling period for reconstructing the time axis. Assuming single-precision representation (4 bytes per number), the total payload is $4\times(2K+1)$ bytes. For reasonable values of K, this amount is well within the hundreds of bytes allowed by standard communication packets and can be efficiently broadcast. Upon reception, vehicle $i+1$ reconstructs an approximation of $\hat{u}_i(t)$ by interpolation and extrapolation.

IV. Numerical Simulations

To validate the proposed predictive event-triggered control framework, we conduct numerical simulations on a seven-vehicle platoon consisting of one leader ($i = 0$) and six followers ($i = 1, \ldots, 6$). Each vehicle follows the dynamics (1) with actuator time constant $\tau = 0.1$ s, time-gap $h = 0.5$ s, and minimum safety margin $r = 10$ m. The controller gains are set to $k_p = 2$ and $k_d = 1$, which satisfy the stability condition under continuous communication established before Theorem 1. The leader follows a reference trajectory that accelerates the platoon from rest to a cruising speed of 120 km/h but which will be subject to two types of disturbance:

- An **instantaneous perturbation**, representing an unexpected obstacle detected by the leader's sensors. This results in a controlled braking maneuver modeled as a smooth velocity bump over the interval $[18, 22]$s.
- A **sustained, unknown perturbation**, modeled as $\Delta v_0(t) = 5\sin(t)$ km/h, a low-amplitude sinusoidal disturbance on the leader's velocity, representing driver oscillations or small road-grade variations.

The event-triggering mechanism employs the constant threshold rule $\|e_{u,i-1}(t)\| \leq \varepsilon$ with $\varepsilon = 0.2$. Through numerical optimization of the LMI (6), we obtain $\eta^* \approx 4.94$, yielding the sensitivity parameter $\rho^* \approx 2.18$. This guarantees that the trajectories converge asymptotically to the set $\mathcal{S}_\rho = \{\|x_i\| \leq \rho\varepsilon \approx 0.44\}$, which remains well within the minimum safety margin r, ensuring collision-free operation. For the system identification-based prediction, we employ an ARX model with orders $(n_a, n_b, n_k) = (2, 2, 1)$, which best align with the dominant first-order dynamics (8). By trial and error, the prediction horizon is set to $N_p T_s = 50 \times 0.05 = 2.5$ s. Beyond this horizon, the predicted signal is extrapolated using a cubic spline. The RLS forgetting factor is $\lambda = 0.98$, allowing gradual adaptation to parameter variations while maintaining numerical stability.

A. Platoon-Level Performance

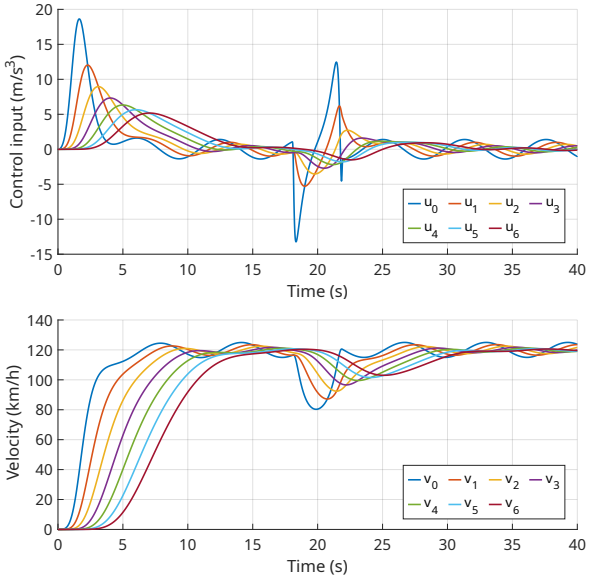

Fig. 2: Control inputs (top) and velocity trajectories (bottom) of leader 0 and its followers i under predictive event-triggered control with system identification-based reconstruction.

Figure 2 displays the control inputs and velocity profiles of leader 0 and its followers $1 \leq i \leq 6$ produced with our identification-based predictions. The platoon accelerates smoothly to the cruising speed, demonstrating effective coordination. When the braking occurs, the disturbance propagates downstream with clear attenuation, illustrating that the predictive event-triggered framework preserves practical individual stability. The small oscillations induced by the sinusoidal perturbation remain bounded and do not amplify along the platoon.

B. Comparison of Reconstruction Strategies

To quantify the relative performance of the three prediction methods, we compare the communication efficiency for the second follower, $i = 2$. Recall that we take the perspective of vehicle i, which sends predictions to $i+1$. Figure 3 shows the actual control input u_2 of follower 2 alongside its computed predicted signal $\hat{u}_2$ to be transmitted, for the model-based and system identification-based approaches. In the model-based case, the prediction closely follows the true signal during the

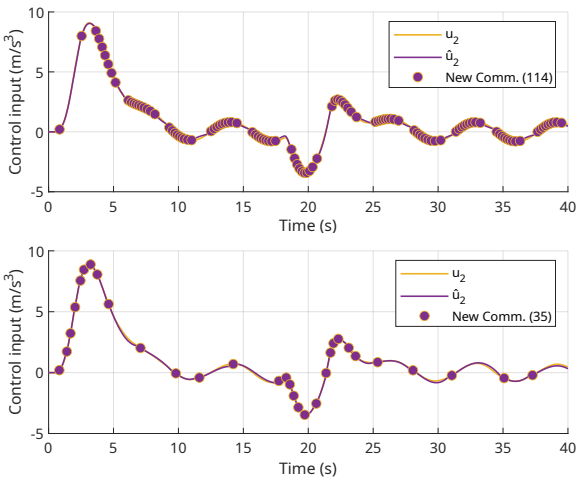

Fig. 3: Control input u_2 (yellow) and predicted signal $\hat{u}_2$ (purple) of the 2nd follower via model-based (top) and identification-based (bottom). Disks are communication events.

initial acceleration phase when the system operates near nominal conditions. However, once the sustained sinusoidal disturbance becomes significant, the prediction error grows rapidly. Since the nominal model cannot reproduce this unmodeled perturbation, the threshold ε is reached frequently, leading to clusters of communication events. The purple dots, which mark triggering instants, appear densely spaced during periods of high disturbance activity. In contrast, the identification-based approach maintains close tracking throughout the simulation. The recursive ARX estimator continuously adapts its parameters to capture both the nominal dynamics and the behavior induced by the sinusoidal disturbance. As a result, the prediction remains accurate even under sustained perturbations, and communication events are more evenly spaced with longer inter-event times.

TABLE I: Number of communication sent per vehicle.

Vehicle	ZOH	Model-based	Sys.Id.-based
0	587	215	57
1	306	151	46
2	201	114	35
3	147	84	38
4	118	59	37
5	94	42	31
Total	**1453**	**665**	**244**

Table I quantifies the communication load by reporting the total number of transmissions sent by each vehicle over the 40-second simulation under the three reconstruction strategies. The ZOH strategy, lacking any predictive capability, triggers the most frequent communications. The reconstruction error grows rapidly whenever the control signal varies, resulting in 587 events for vehicle 0 alone and 1453 events across the entire platoon. The model-based approach achieves improvement, reducing the total communication count to 665 events—a 54% reduction compared to ZOH. However, the system identification-based method delivers the most gains, requiring only 244 total events—an 83% reduction relative to ZOH and a 63% reduction relative to the model-based strategy. Table II summarizes the trade-offs between the three prediction methods.

TABLE II: Comparison of reconstruction strategies.

Feature	ZOH	Model-based	Sys.Id.-based
Model knowledge	Not required	Nominal required	Not required
Dist. robustness	Poor	Moderate	High
Computational cost	Minimal	Low	Moderate
Packet size (bytes)	∼4	∼100-200	∼100–200
Total events (sim.)	1453	665	244
Reduction vs. ZOH	—	54%	83%

V. Conclusion

This paper presented a predictive event-triggered control framework for vehicle platooning that reduces the communication load while preserving practical individual stability guarantees. We compared three reconstruction strategies: ZOH and model-based prediction as baselines, and our proposed system identification-based approach using recursive ARX estimation. Numerical simulations on a seven-vehicle platoon demonstrated a reduction in communication events of system identification compared to ZOH and model-based prediction, while maintaining safe spacing throughout transient maneuvers and sustained perturbations. In the future, communication delays and packet losses should be handled.

References

[1] S. E. Shladover, "Connected and automated vehicle systems: Introduction and overview," *Journal of Intelligent Transportation Systems*, vol. 22, no. 3, pp. 190–200, 2018.

[2] H. Swaroop, "String stability of interconnected systems: An application to platooning in automated highway systems," *Transportation Research Part A: Policy and Practice*, vol. 31, no. 1, p. 65, 1997.

[3] J. Ploeg, N. Van De Wouw, and H. Nijmeijer, "Lp string stability of cascaded systems: Application to vehicle platooning," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 2, pp. 786–793, 2013.

[4] G. Nardini, A. Virdis, C. Campolo, A. Molinaro, and G. Stea, "Cellular-V2X Communications for Platooning: Design and Evaluation," *Sensors*, vol. 18, no. 5, p. 1527, 2018.

[5] P. Tabuada, "Event-triggered real-time scheduling of stabilizing control tasks," *IEEE Transactions on Automatic Control*, vol. 52, no. 9, pp. 1680–1685, 2007.

[6] V. S. Dolk, J. Ploeg, and W. M. H. Heemels, "Event-triggered control for string-stable vehicle platooning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 12, pp. 3486–3500, 2017.

[7] N. He, D. Shi, and T. Chen, "Self-triggered model predictive control for networked control systems based on first-order hold," *International Journal of Robust and Nonlinear Control*, vol. 28, no. 4, pp. 1303–1318, 2018.

[8] P. Varutti, B. Kern, T. Faulwasser, and R. Findeisen, "Event-based model predictive control for networked control systems," in *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, 2009, pp. 567–572.

[9] E. Gorski, I.-C. Morărescu, V. Satheeskumar Varma, and L. Buşoniu, "Predictive event-triggered control for string-stable platooning," 2026, preprint. [Online]. Available: https://hal.science/hal-05515573

[10] L. Ljung, "System Identification," in *Signal Analysis and Prediction*, A. Procházka, J. Uhlíř, P. W. J. Rayner, and N. G. Kingsbury, Eds. Boston, MA: Birkhäuser, 1998, pp. 163–173.

[11] A. Ibrahim, D. Goswami, H. Li, I. M. Soroa, and T. Basten, "Multi-layer multi-rate model predictive control for vehicle platooning under IEEE 802.11p," *Transportation Research Part C: Emerging Technologies*, vol. 124, p. 102905, 2021.

[12] Y. H. Jang and H. S. Kim, "Sampled-Data Cooperative Adaptive Cruise Control for String-Stable Vehicle Platooning with Communication Delays: A Linear Matrix Inequality Approach," *Machines*, vol. 12, no. 3, p. 165, 2024.