

Congestion-mitigating MPC design for adaptive cruise control based on Newell's car following model: History outperforms prediction

Hao Zhou^a, Anye Zhou^a, Tienan Li^b, Danjue Chen^b, Srinivas Peeta^a, Jorge Laval^{a,*}

^a School of Civil and Environmental Engineering, Georgia Institute of Technology, Atlanta, United States

^b School of Civil and Environmental Engineering, University of Massachusetts Lowell, Lowell, United States

ARTICLE INFO

Keywords:

MPC
Adaptive cruise control
Newell's car following
String stability
Openpilot

ABSTRACT

Currently, model predictive control (MPC) for adaptive cruise control (ACC) systems relies on the prediction of the leader's motion to plan the follower's trajectory. However, such predictions must be accurate to guarantee string stability, which represents an ongoing challenge for machine learning approaches. This issue can be circumvented by simply incorporating the leader's history, which follows from Newell's car-following (CF) model where a trajectory under congestion corresponds to a temporal-spatial shift of the leader's past trajectories. By leveraging this insight, this paper develops a family of MPC models based on Newell's CF model, labeled Newell MPCs, which are safe and can reduce traffic congestion.

Specifically, We first present baseline Newell MPCs to replicate the original Newell's CF model, including the *Xbound*-Model, which uses the shifted leader trajectory as an upper envelope; and the *Xref*-Model, which adopts the shifted leader trajectory as a reference to avoid the issue of infeasible solution triggered by hard constraints. To further improve the control performance, we propose the *XV*-Model which uses the leader speed history as an additional reference to enhance the model robustness and regulate speed over/under-shootings. In addition, we extend the single-leader Newell's model through incorporating multiple leaders and propose the *Xmul*-Model, which can achieve driver anticipation, and correspondingly reduce reaction time and improve string stability. Finally, based on the *XV*-Model, we present two additional extensions: (i) the *XVrelax*-Model, which incorporates driver relaxation to achieve smooth response to merging traffic; and (ii) the *XVss*-Model, which achieves strict string stability to further dampen traffic oscillations. The proposed Newell MPCs are tested using both numerical simulations and field studies on a stock 2019 Honda Civic using Openpilot and Comma.ai; the source code is available at <https://github.com/HaoZhouGT/openpilot>.

1. Introduction

ACCs were first developed as linear controllers in the 1990s (Sheikholeslam and Desoer, 1993; Ioannou et al., 1993; Fancher et al., 1997; Jurgen, 2006; Gerrit Naus, 2008). Unfortunately, this technology did not gain much popularity with consumers, possibly due to the lack of driving comfort. Starting from the 2000s, MPC ACCs have become more and more popular (Bageshwar et al., 2004; Gerrit Naus, 2008; Luo et al., 2010; Li et al., 2010).

Over the past few years, ACC's market share has steadily increased. In the meantime, the string stability of ACC systems has attracted increasing interest in traffic flow theory due to its profound impact on traffic congestion. Recent empirical studies (Gunter

* Correspondence to: 790 Atlantic Dr NW, Atlanta, GA, 30313, United States.

E-mail address: jorge.laval@ce.gatech.edu (J. Laval).

et al., 2020; Makridis et al., 2021; Shi and Li, 2021; Li et al., 2021a,b) have demonstrated that most market ACCs are string unstable, which is unfortunate as it indicates ACCs will amplify congestion waves. Intriguingly, a few empirical studies also find that the string stability of market ACCs is significantly more complex than what linear controllers can explain, largely due to its non-linearity and the speed/scale dependence. Shi and Li (2021) finds that the estimated key parameters and string stability of the market ACCs substantially vary across different speed levels. According to Li et al. (2021a), market ACC products can be string stable under small oscillations but string unstable for large oscillations, even when the speed level and oscillation frequency are the same. These findings suggest that the string stability of market ACCs is both speed-dependent and scale-dependent, which is unfortunate as we expect ACCs to be string-stable under all speed and oscillation conditions.

Unlike linear controllers to regulate current spacings and speeds, MPC ACCs foresee a finite time-horizon, predict the leader movements first, and then solve the future trajectory from an optimization problem that can include multiple objectives such as safety, efficiency, and driving comfort. The predict-and-optimize MPC makes it easier for ACCs to add new objectives for trajectories, beyond the spacing and speed regulation that linear ACCs provide. Moreover, the receding horizon optimization mechanism of MPC also brings new opportunities to achieve more desirable control performance (e.g., fuel economy, comfort, smoothness, etc.) than linear ACCs (Musa et al., 2021). We hypothesize that current ACC manufacturers would utilize MPC ACC considering its inherent flexibility and outstanding performance. For instance, Tesla clearly disclosed its predict-and-optimize approach at its AI Day Event in 2021 (Zhou, 2021), as Waymo (Ettinger et al., 2021) and Baidu (Zhang et al., 2020) also announced similar predict-and-optimize methods. Although different manufacturers may embed the predict-and-optimize problem using frameworks other than MPC, in this paper we refer to these ACCs as MPC ACCs because they share the same spirit of the MPC, i.e., anticipating future events and optimizing the current timeslot while considering future timeslots.

While MPC ACCs are especially effective at incorporating and balancing multiple objectives, they have limitations including the string stability, which lacks a closed-form analysis due to its non-linearity and is hard to guarantee because of the prediction challenge. As mentioned earlier, current MPC ACCs need to predict leader positions before optimizing the follower's trajectory. We show later that the precision of the leader predict is important as a better trajectory prediction alone can dramatically improve the string stability. Unfortunately, to accurately predict the leader movement is challenging, as we are witnessing many ongoing efforts in this direction, such as the machine learning challenges specialized in forecasting leader movements held by Lyft (2021) and Waymo (2022). To train a decent model for leader prediction often requires deep neural networks and large datasets. Unfortunately no major success has been reported by the industry yet.

In light of all those limitations and challenges, a prediction-free, and string-stable ACC under all speeds and oscillations is highly anticipated. Fortunately, we already have such a model in the traffic flow theory, namely Newell's CF model, that satisfies all of these criteria. According to Newell's CF model, the follower's trajectory is merely a temporal-spatial shift of the leader's history position. Thereby, it ensures marginal string stability, meaning that the follower neither amplifies nor dampens the leader's oscillations. More interestingly, Newell's CF model does not require predictions of the leader at all, instead, it only requires the leader's history (position, speed or both), which can be easily obtained using current sensing technologies, such as a single radar. Such a simple and elegant model has not drawn much attention from the robotics/computer science community, possibly due to lack of disciplinary depth in traffic flow theory, and to the best of our knowledge, no self-driving company has reported the use of this idea to imitate the leader's history trajectory. Although we do not know whether Newell's CF model is not being used by market ACCs due to the opacity of self-driving companies, the aforementioned recent empirical tests have revealed the significant differences between Newell's CF model and current black-box ACCs, where ACCs exhibit poor string stability across different speeds and oscillations, which is substantially different from the constant marginal string stability of Newell's CF model. Those empirical observations suggest that Newell's CF model or a similar design is very unlikely to exist in market ACCs. To the best of our knowledge, no similar design of Newell's CF model under MPC has been documented by industry or academia, which motivates us to conduct this study.

The paper aims to bridge the aforementioned research gaps and develop a family of new MPCs based on Newell's CF model, i.e. Newell MPCs. As extensions to the original Newell's model, the strict string stability, driver relaxation (Laval and Leclercq, 2008) and driver anticipation (Treiber et al., 2007) will be incorporated to mitigate congestion and improve traffic capacity. To ensure transferability from the paper to the road, we use the same codebase for the simulations, including the processional MPC solver from a recent open-source ACC system, Openpilot (OP), Comma.ai. Later we will test the functioning of the same MPC on a production ACC vehicle (a 2019 Honda Civic) on the road.

The major contribution of this paper is to incorporate Newell's CF model into the MPC ACC, which provides a simple and elegant solution to string stable ACCs without prediction needs. Additionally, we present a framework to apply Newell's model for anticipative driving, which can reduce reaction time and improve string stability. We also provide two extensions of the baseline Newell MPC, one of which aims to address lane change disruptions and the other to achieve strict string stability in order to further dampen traffic oscillations. Finally, we implement a baseline Newell MPC (i.e., *XV*-Model) in a market ACC vehicle on the road, which demonstrates the feasibility of the design. The source code is available at <https://github.com/HaoZhouGT/openpilot>.

The paper is organized as follows: Section 2 surveys the MPC design in the literature and its recent development in the industry; Section 3 introduces our design of the family of Newell MPCs; Section 4 presents the simulation results; Section 5 describes the road test; and Section 6 provides some concluding comments.

2. Current MPCs and their limitations

2.1. State of the art

MPC was first introduced (Bageshwar et al., 2004) to ACC systems as a new distance controller to minimize the spacing error with a target value in an optimal control problem (OCP). For the target spacing, the safe inter-distance-policy was used (Chien and

Ioannou, 1992; Seiler et al., 1998; Brackstone and McDonald, 1999), which is a Newtonian motion equation; see (1).

$$s_{safe} = \lambda_1(v_{ego}^2 - v_{lead}^2) + \lambda_2 v_{ego} + \lambda_3 \quad (1)$$

where v_{ego} and v_{lead} refers to the speed of the follower and the leader respectively, and λ_1 , λ_2 and λ_3 are constants.

It was quickly noticed that the safe inter-distance-policy can produce undesirable large spacings, then the constant time headway (CTH) (2) took place its role which describes the desired spacing s_{ego} as follows.

$$s_{des} = \tau \cdot v_{ego} + \delta \quad (2)$$

where τ is the desired time headway, and δ is the jam spacing.

Initially, the MPC was simply a different approach to distance control, where safety, comfort and economy were not directly included in the objectives but instead served as hard constraints (Corona et al., 2006; Martinez and Canudas-de Wit, 2007). Gerrit Naus (2008), Luo et al. (2010), Li et al. (2010) extended the MPC from a distance controller to the solver of a multi-objective optimization problem involving other goals. In their design, except for the desired spacing, MPC also minimizes the relative speed with its leader, i.e. $v_{lead} - v_{ego}$. Meanwhile, some secondary objectives such as driving comfort and fuel assumptions are added by simply minimizing the jerks j_{ego} and accelerations a_{ego} respectively.

The hard constraints in those MPC controllers are usually constant, meaning that only the nominal maximum or minimum accelerations/jerks are applied, which is not very realistic considering the linear acceleration bounds found in traffic flow studies (Laval and Daganzo, 2006; Laval et al., 2014).

It is also worth noting that, the models used in MPCs for the ego vehicle are usually beyond the simplest motion dynamics of a point-mass. Typically the model incorporates an delay-actuator to capture the lag between commanded accelerations and the true output. For example, the first-order system delay model is also shown in Bageshwar et al. (2004), Zhou and Peng (2005) as follows:

$$t_{delay} \frac{d\ddot{s}(t)}{dt} + \ddot{s}(t) = u(t) \quad (3)$$

where t_{delay} refers to the time lag in the low-level controller, and u is the control input (desired acceleration), and s is the position state.

Another key component of the MPC ACC, the prediction of the leader movement, is usually over-simplified by assuming a zero acceleration in the literature (Bageshwar et al., 2004; Zhou and Peng, 2005; Naus et al., 2008; Luo et al., 2010). We will illustrate the significance of accurate prediction models shortly.

Those MPCs for ACCs are formulated mathematically as a discrete, constrained linear quadratic regulator (LQR) which can be solved by professional solvers. The follow-up studies on the MPC design employ the same general approach, with modest tweaks to enhance safety (Magdici and Althoff, 2017), energy consumption (Vajedi and Azad, 2015), or extend weight tuning to the real time (Zhao et al., 2017).

While sequel MPC studies for ACCs are sparse, a much larger body of research (Bu et al., 2010; Naus et al., 2010; Stanger and del Re, 2013; Gong and Du, 2018; Zhou et al., 2019, 2020) is working on developing MPCs for Cooperative Adaptive Cruise Control (CACC) systems, which feature accurate acceleration values transmitted by short-range communication technologies. More system-level optimization problems are studied in the CACC context. Notably, Gong and Du (2018) developed a MPC-CACC control algorithm to ensure the overall speed smoothness and stability. Zhou et al. (2019) presented a distributed MPC approach for CACC systems to achieve local and string stability. However, these insights cannot be directly transferred to MPCs in current non-cooperative ACC systems using only speed and spacing measurements from the leader.

2.2. State of the practice: the MPC design in open-source self-driving system

The market ACC technology is proprietary, but in recent years, a few self-driving companies have begun to release their source code to the general public. Comma.ai, which views itself as the Android to Tesla's Apple, is one of the first companies to offer an after-market device that can be connected to the ACC modules on recent car models and run customized ACC algorithms. The open-source control algorithms, including the MPC design, provide new opportunities to model the black-box market ACCs.

Now we formulate the MPC algorithms in OP's codebase. In OP, the total cost function $C(t)$ for the longitudinal MPC starting from time t is defined as a weighted sum of four sub-costs over all grid points in its rolling horizon $[t, t + T_{mpc}]$.

$$C(t) = 1/2 \sum_{t \leq \hat{i}(k) \leq t + T_{MPC}} f(k)[w_{ttc} C_{ttc}^2(\hat{i}(k)) + w_{dist} C_{dist}^2(\hat{i}(k)) + w_{accel} C_{accel}^2(\hat{i}(k)) + w_{jerk} C_{jerk}^2(\hat{i}(k))] \quad (4)$$

where $\hat{i}(k)$ denotes the time of each discrete planning step k in the planning horizon T_{MPC} of 10 s. $f(k)$ is a step multiplier to weight the costs from different time steps in the 10 s horizon, where $f(k) = 1$ for $k \leq 5$ with step size of 0.2s, and $f(k) = 3$ for $k > 5$ with step size of 0.6 s. We conjecture that the step multiplier 3 is used to balance different step lengths ($3 = 0.6/0.2$) between the first 5 steps and the remaining 15 steps. The weights for sub-costs are constant for all driving scenarios. Determining those values need careful tuning, where the default values in OP are $w_{ttc} = 5$, $w_{dist} = 0.1$, $w_{accel} = 10$, $w_{jerk} = 20$.

The sub-costs functions are with respect to the time to collision, spacing, acceleration and jerk values, which are defined as follows if the time index is omitted:

$$C_{ttc} = \exp\left\{\frac{0.3(s_{des} - s_{ego})}{\sqrt{v_{ego} + 0.5 + 0.1}}\right\} - 1 \quad (5)$$

$$C_{dist} = \frac{s_{ego} - s_{des}}{0.05v_{ego} + 0.5} \quad (6)$$

$$C_{accel} = a_{ego}(0.1v_{ego} + 1) \quad (7)$$

$$C_{jerk} = j_{ego}(0.1v_{ego} + 1) \quad (8)$$

where j_{ego} is the jerk of ego vehicle (i.e., derivative of acceleration). The desired spacing s_{des} for the MPC controller is defined as:

$$s_{des} = v_{ego} \cdot \tau - (v_{lead} - v_{ego}) \cdot \tau + \frac{v_{ego}^2 - v_{lead}^2}{2B} \quad (9)$$

where B is the vehicle's maximum deceleration rate.

Note that to calculate the desired and true spacing in the MPC prediction horizon, the lead vehicle trajectory and speed need to be estimated. OP uses a simple model to predict the lead vehicle movement as shown in by the dynamics model in (10) (a–d), which assumes an exponentially decaying acceleration of the lead vehicle with time parameter θ (1.5s). Recall that $\hat{t}$ denotes the time of discrete step in the MPC planning horizon T_{MPC} . Let $\Delta\hat{t}$ denote the step length between two consecutive steps, where $\Delta\hat{t}(k) = 0.2s$ when $k \leq 5$ and $\Delta\hat{t}(k) = 0.6s$ if $k > 5$. Starting from $\hat{t}(0) = 0$, the lead vehicle states are predicted and updated as follows:

$$a_{lead}(\hat{t}) = a_{lead}(t) \exp(-\theta \cdot \hat{t}^2/2) \quad (10a)$$

$$x_{lead}(\hat{t}) = x_{lead}(\hat{t}) + v_{lead}(\hat{t}) \cdot \Delta\hat{t} \quad (10b)$$

$$v_{lead}(\hat{t}) = v_{lead}(\hat{t}) + a_{lead}(\hat{t}) \cdot \Delta\hat{t} \quad (10c)$$

$$\hat{t} := \hat{t} + \Delta\hat{t} \quad (10d)$$

Recall that t is the current time, thus $a_{lead}(t)$ denotes the current leader acceleration measured by the sensor. The above model is just a simple prediction of the leader's future acceleration, speeds and positions.

The control variable in OP MPC is a vector of jerks in the future horizon, based on which the state variables of the ego vehicle, i.e., the acceleration, speed and position lists are calculated using the dynamics model:

$$\dot{x}_{ego} = v_{ego} \quad (11)$$

$$\dot{v}_{ego} = a_{ego} \quad (12)$$

$$\dot{a}_{ego} = j_{ego} \quad (13)$$

where $j_{ego}(k)$ is the jerk value at the MPC step $k = 0, 1, \dots, K$. Notice that the dynamics model in OP does not consider the actuator delay which is commonly used in literature.

With the prediction and dynamic models, we can formulate OP's MPC as follows:

$$\begin{aligned} \min_{\vec{j}_{ego}} \quad & C(t) \\ \text{s.t.} \quad & \text{Dynamics model of the ego vehicle in (11), (12), and (13)} \\ & \text{Predicted dynamics for the lead car in (10)} \\ & v_{target} \geq 0 \end{aligned} \quad (14)$$

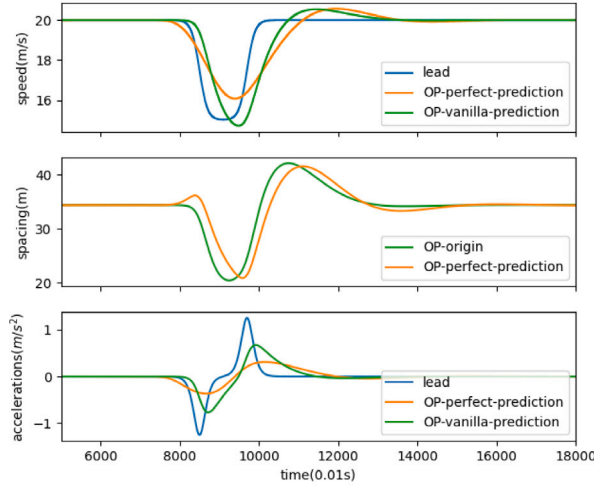
At each planning starting time t , OP solves the jerk vector $\vec{j}_{ego} = [j_{ego}(0), j_{ego}(1), \dots, j_{ego}(K)]$ for the MPC problem (14) and obtains the desired acceleration for the next MPC planning step, i.e. $a_{mpc}(t + \Delta\hat{t})$. The desired acceleration will be further passed to the low-level system for execution.

To sum up, the MPCs in both the literature and the industry use a few heuristics cost functions to drive the car, in a manner close to but not necessarily identical to that of humans. The distance term always moves the vehicle to the desired spacing, the safety term provides additional assurance that unsafe distances are avoided. The jerk term directly addresses comfort. Those three components address the three typical aspects of designing the optimal trajectory for a self-driving car, as Tesla does similarly (Zhou, 2021). The acceleration term accounts for the vehicle's limited acceleration/deceleration capabilities, which also addresses energy and comfort requirements.

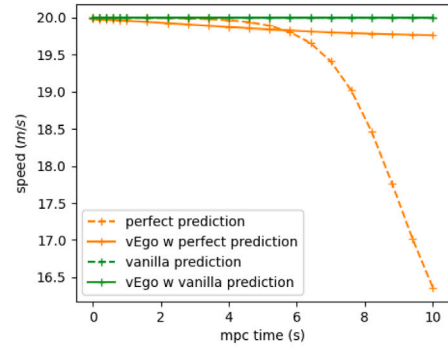
2.3. Comparisons between MPCs from the art and the practice

The objective functions in the factory-level OP MPC are defined differently than those in the literature. A significant distinction is the extra precaution taken with safety. In (9), the safe braking term is added to the CTH for enhanced safety. In addition to the regular spacing control that works both for larger or smaller spacing, OP adds an exponentially-increasing time-to-collision (TTC) term to prevent dangerously small spacing.

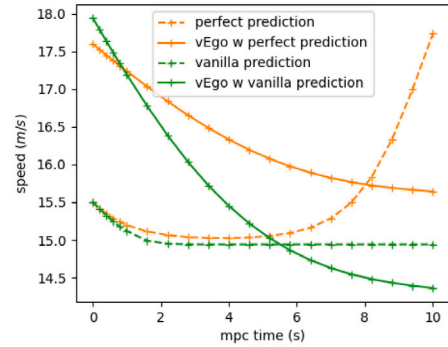
Another significant distinction is that the cost functions are all speed-dependent. The (7) and (8) suggest that acceleration and jerk costs grow with higher speeds, meaning the MPC will restrict them more at high speeds, which is reasonable given the well-known linear acceleration limit where the engine produces smaller accelerations at higher speeds. Instead, the safety and distance costs both decrease at higher speeds; see (5) and (6), indicating that MPC loosens spacing control at high speeds.



(a) Example of the Newell MPC planning step



(b) MPC plan when the leader starts to brake at step 7600



(c) MPC plan when the leader starts to accelerate again at step 8700

Fig. 1. Leader prediction plays a decisive role in the string stability of the MPC ACC : In the legends ‘lead’ means leader, w is short for ‘with’. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

In contrast to the prevalent delay model (3) in the literature, the dynamics in the factory-level OP MPC completely ignores the actuator delay, suggesting that the actuator delay might not be significant, or not a life-or-death issue for current ACC system. The impacts of such difference needs more investigation. Notice that the delay model used in the literature actually originates from 1990s, which might no longer be suitable for today’s market vehicles after 30 years development. On the other hand, OP has more than thousands of active users and accumulated millions of miles from over 140 vehicle models (Comma.ai, 2021). Although somewhat counter-intuitive, the OP MPC planner without specific delay treatment has not yet caused significant problems. This is likely due to their low-level control design, which compensates for actuator delay by slightly overshooting/undershooting the commanded gas/brake values (Zhou et al., 2022b).

While the literature often assumes a zero acceleration of the leader, OP uses an exponentially decaying acceleration model. The impact of the different predictions will be discussed shortly.

2.4. The leader prediction matters

As stated earlier, the current MPC optimizes the follower trajectory given the predicted leader movement. Now we show the significance of the prediction model by comparing the same MPC in OP (OP-MPC) with two different predictions: one is the default prediction model in (10), referred to as the Vanilla prediction, the other is the perfect prediction assuming the follower can perfectly foresee the future movement of the leader, i.e. the ground truth. The perfect prediction here is achieved in simulation by predefining the leader trajectory and equipping the follower with this prior knowledge. The comparison results are shown in Fig. 1. In Fig. 1(a), it is clear that the same MPC is string stable with the perfect prediction of the leader, but amplifies the oscillation provided the vanilla prediction. Notice that, due to the “foreseeing” capability of the perfect prediction model, the follower can respond much earlier; see the orange line (the follower’s speed profile) starts to change even earlier than the blue line (the leader’s speed profile), while the green line (the follower with the vanilla prediction) is lagging behind. It also suggests that, the string stability can be improved if the follower responds to the leader’s speed change earlier. At the beginning of the oscillation, the OP with vanilla prediction underestimates the speed change of the leader because the deceleration is not exponentially decaying. Such underestimation results in a smaller deceleration compared with the leader and the short spacing, which forces the follower to decelerate more to compensate. The above comparison indicates that prediction plays an important role in the string stability of MPC ACCs. To pursue better string stability, one can propose better prediction models. However, this is not an easy task since predicting

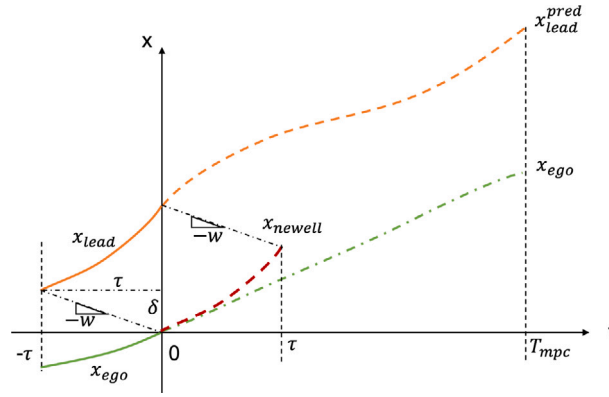


Fig. 2. Comparison of the existing MPC design and the proposed Newell MPC: w is the universal wave speed, τ is the desired time headway, and δ is the jam spacing. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 1
Summary of the family of Newell MPCs.

Name	Motivation	Feature	Pros and Cons	Application
X_{bound}	Directly translate Newell's model	The shifted leader position is used as the bound	Simplest model but prone to infeasible solutions due to sensor errors	Follow a fixed leader
X_{ref}	Avoid infeasibility caused by the bound in X_{bound} , try reference	The shifted leader position is used as the reference	Avoid hard constraints but can suffer from over/under-shootings	Follow a fixed leader
XV	Regulate speed over/under-shootings and add redundancy to X_{ref}	Use the leader history position and speed as references	Increase sensor redundancy and flexible for congestion-friendly extensions	Follow a fixed leader
X_{mul}	For anticipative driving using multiple look-ahead leaders in advanced ACCs	Use history positions of multiple leaders as the reference	Achieve strict string stability and reduce reaction time	Multiple leaders available
XV_{ss}	Towards the strict string stability	Adopt new speed references using discounted leader history speeds	Achieve strict string stability but face smaller spacings	Dampen oscillations
XV_{relax}	Smooth merging traffics	Incorporate driver relaxation into position references	Reduce oscillations caused by cut-ins but face smaller spacings	Follow the cut-in

the leader's movement may require a complex neural network trained on massive human-labeled data, as we have seen many ongoing machine learning challenges on forecasting leader movements (Lyft, 2021; Waymo, 2022). No major successes have been documented in the literature or in the industry. To run large machine learning models on automobiles, however, may necessitate the use of powerful chips, which is currently uncommon on market vehicles.

Instead of focusing on a precise prediction model, we argue that, the widely available sensor measurements of history (leader's speed and position) can be utilized more effectively in accordance with Newell's well-known CF model, which suggests that human drivers' following trajectory is just a temporal-spatial shift of the leaders' movement. Newell's model not only captures human drivers' CF behaviors, but also illustrates significant potential to improve traffic efficiency. It does not amplify the congestion wave when the leader decelerates, and catches up tightly to prevent extra spacings when the leader accelerates away, which is rarely observed from market ACC products (Li et al., 2021a). Inspired by these good qualities of the Newell's CF model, the next section will introduce the design of a group of Newell MPCs.

3. Design of the Newell MPCs

We have introduced the existing MPC models for ACC, which first predict the leader movement x_{lead}^{pred} and then solve the ego vehicle trajectory from an optimization problem formulated in (14). Motivated by the challenges of good predictions and their impact on traffic efficiency, we now present the design of the family of Newell MPCs. The Fig. 2 shows the backbone of the Newell MPC design, and compares the differences and similarities between the two approaches. As the figure indicates, while the existing MPC relies on prediction with a designed horizon of T_{mpc} , the Newell MPC has a fixed horizon of τ seconds. More importantly, the solution of ego trajectory in current MPCs results from heuristic optimization objectives, while the Newell MPC bases its solution on the shifted leader's position x_{lead} from $t - \tau$ to current time t , i.e. the Newell trajectory x_{Newell} .

In this section, several Newell MPCs are developed and organized as follows: first, we present baseline Newell MPCs to replicate the original Newell's CF model to achieve the marginal string stability. Towards this, we first propose the X_{bound} -Model, which uses

x_{Newell} as the upper bound, and then the *Xref*-Model, which uses x_{Newell} as a reference alternatively to avoid infeasibility issues caused by the bound. Further, we find that the leader speed history can be added as an additional reference to improve the model robustness and regulates speed over/under-shootings, thereby constructing the *XV*-Model. A natural extension of the single-leader Newell's model is to incorporate multiple look-ahead leaders, i.e., *Xmul*-Model, which achieves driver anticipation that helps to reduce reaction time and improve string stability. Finally, on the basis of the *XV*-Model, we present two additional extensions: the *XVrelax*-Model, which incorporates driver relaxation to smooth merging traffics, and the *XVss*-Model, which achieves strict string stability to further dampen traffic oscillations. The summary of the proposed six Newell MPC models is shown in Table 1. Note that we use X to denote position and V to denote speed for abbreviation.

3.1. Baseline newell MPC

We first present the baseline Newell MPC that replicates the original Newell's CF model. Recall that the original Newell's CF model adopts the shifted leader trajectory, i.e. the Newell trajectory, as the upper envelope for the follower. Motivated by this, we will first present the *Xbound*-Model that directly translates the same spirit.

3.1.1. Using the leader history as the bound: *Xbound*-Model

The *Xbound*-Model utilizes the shifted leader history trajectory, i.e. the Newell trajectory/bound x_{Newell} as the upper limit for the follower's position x_{ego} between t and $t + \tau$. Recall that k denotes the index of discrete time steps in the planning horizon starting from time t , the Newell trajectory x_{Newell} is defined as:

$$x_{Newell}(t, k) = x_{lead}(t - \tau + k\Delta t) - \delta \quad (15)$$

The optimization goal for the *Xbound*-Model is as simple as the entropy condition (EC) found in human drivers, which is to aim for higher speeds subject to the constraint of the Newell bound $x_{Newell}(k)$ and the acceleration limits $a_{max}(v_{ego}(k))$, which are a function of the ego vehicle speed $v_{ego}(k)$ according to the linear acceleration bound model in Laval and Daganzo (2006), Laval et al. (2014). As depicted in Fig. 2, the design concept is to advance the ego position (green dashed line) below the shifted Newell trajectory (red dashed line) to the extent feasible. Mathematically, the optimization problem is formulated as follows:

$$\min_{j_{ego}} C_{Newell}(t) = 1/2 \sum_{0 \leq k \leq \lceil \tau/\Delta t \rceil} C_{speed}^2(t, k) \quad (16a)$$

$$\text{s.t. } x_{ego}(t, k) \leq x_{Newell}(t, k) \quad \forall k \in \{0, 1, \dots, \lceil \tau/\Delta t \rceil\} \quad (16b)$$

$$a_{ego}(t, k) \leq a_{max}(v_{ego}(t, k)) \quad \forall k \in \{0, 1, \dots, \lceil \tau/\Delta t \rceil\} \quad (16c)$$

The C_{speed} is the cost function that always rewards higher speeds:

$$C_{speed}(t, k) = v_{max} - v_{ego}(t, k) \quad (17)$$

where v_{max} is the desired free flow speed, or the target speed set by the driver in the ACC system. The (16a) suggests that the MPC wants to go as fast as possible, which is in spirit of the EC in human drivers who want to drive as fast as we can but stay within the set limit.

The preceding *Xbound*-Model is similar to the original Newell's CF model, with the addition of a realistic linear acceleration restriction, a realistic linear acceleration constraint. It can probably be the simplest MPC design that can drive a car provided an uninterrupted and smooth Newell bound. The model is parsimonious, without requiring any heuristics objective functions to reason the driving rules of human drivers.

However, such a design quickly falls short when the Newell bound is discontinuous, such as when a vehicle suddenly cuts in, or when the Newell bound is noisy due to inaccurate sensor measurements, e.g. $x_{Newell}(t, k)$ is an outlier at some k . Those scenarios or conditions can be devastating because the hard constraint in (16b) cannot be satisfied, resulting in an infeasibility problem for the MPC where no solutions can be found.

To avoid those issues, we propose substituting the soft constraints for the hard constraints in (16b) and designing a exponential-shape cost function to penalize any positive errors without compromising feasibility:

$$C_{bound}(t, k) = \exp\left\{c \cdot \frac{x_{ego}(t, k) - x_{Newell}(t, k)}{v_{ego} \cdot \tau}\right\} - 1 \quad (18)$$

where c is a positive coefficient needs tuning.

To make the MPC more robust to noises in sensor measurements, we also incorporate similar sub-cost designs in OP to prevent exploding accelerations or jerks.

$$C_a(t, k) = a_{ego}(t, k) \cdot v_{ego}(t, k) \quad (19a)$$

$$C_j(t, k) = j_{ego}(t, k) \cdot v_{ego}(t, k) \quad (19b)$$

Combining all, the total cost for the *Xbound*-Model is defined as:

$$C_{Newell}^{bound}(t) = 1/2 \sum_{0 \leq k \leq \lceil \tau/\Delta t \rceil} w_{speed} C_{speed}^2(t, k) + w_{bound} C_{bound}^2(t, k) + w_a C_a^2(t, k) + w_j C_j^2(t, k) \quad (20)$$

where w_{speed} , w_{bound} , w_a , and w_j are weights that need further tuning.

Finally, the *Xbound*-Model is formulated as follows:

$$\min_{j_{ego}} C_{Newell}^{bound}(t) \quad (21a)$$

$$\text{s.t. } a_{ego}(t, k) \leq a_{max}(v_{ego}(t, k)) \quad \forall k \in \{0, 1, \dots, \lfloor \tau/\Delta t \rfloor\} \quad (21b)$$

$$\text{Dynamics model of the ego vehicle in (11), (12), and (13)} \quad (21c)$$

where the explicit acceleration bound is designed to be the linear, i.e., $a_{max}(v) = a_{max} - \beta v$, which originates from [Laval and Daganzo \(2006\)](#), [Laval et al. \(2014\)](#) and captures the comfortable accelerating behaviors of human drivers. The parameters a_{max} and β can differ across car models based on engine performance. Note that we use the explicit linear acceleration limits instead of the soft acceleration constraints in (19a) to get the most out of the vehicle acceleration capabilities. This is designed to preclude the extra large spacings, i.e. capacity loss, that are caused by timid accelerations found on market ACC vehicles ([Li et al., 2021a](#)).

3.1.2. Using the leader histories as references: *Xref*-Model and *XV*-Model

As stated earlier, using the Newell trajectory as the bound might not be flexible enough when the follower is cut off or the sensor has noisy measurements. A simple variation is to use the Newell trajectory x_{Newell} as the reference, i.e. minimizing the squared 2-norm of the errors as one of the sub-costs, which we refer to as the *Xref*-Model. Recall the popular CTH-RV (Constant time headway and relative speed) model ([Willis, 1999](#); [Gunter et al., 2019](#)) in ACC literatures consists of two terms which respectively regulate the spacing and speed. Inspired by this, if we consider position reference as a spacing regulator, we may add speed reference as the speed regulation into the Newell MPC, which we refer to as the *XV*-Model.

Now we present the detailed design of *Xref*-Model and *XV*-Model. First we introduce the definition of the Newell speed reference v_{Newell} at time t :

$$v_{Newell}(t, k) = v_{lead}(t + k\Delta t - \tau) \quad \forall k \in \{0, 1, \dots, \lfloor \tau/\Delta t \rfloor\} \quad (22)$$

Then total objective for the *XV*-Model is formulated as follows:

$$\min_{j_{ego}} C_{Newell}^{xv}(t) = 1/2 \sum_{0 \leq k \leq \lfloor \tau/\Delta t \rfloor} w_x C_x^2(t, k) + w_v C_v^2(t, k) + w_a C_a^2(t, k) + w_j C_j^2(t, k) \quad (23a)$$

$$\text{s.t. } a_{ego}(t, k) \leq a_{max}(v_{ego}(t, k)) \quad (23b)$$

$$x_{ego}(t, k) \leq x_{lead}^{pred}(t, k) - \delta \quad (23c)$$

where the two new cost terms to regulate the positions and speeds are defined as follows:

$$C_x(t, k) = x_{ego}(t, k) - x_{Newell}(t, k) \quad (24a)$$

$$C_v(t, k) = v_{ego}(t, k) - v_{Newell}(t, k) \quad (24b)$$

For *Xref*-Model, since it only adopts the position reference, one can just omit the speed term $w_v C_v^2$ in the objective function (23a) while the rest of the design remains the same as *XV*-Model.

The safety constraint in (23c) requires a prediction of the leader position x_{lead}^{pred} . Although we still need predictions for safety in this model, it should be noted that such prediction does not have to be as accurate as those required for the string stability. Even a zero-acceleration leader prediction model (25) can serve its purpose here because the MPC has a rolling horizon that moves forward at the radar rate (e.g. 0.05 s). At every new planning step, the MPC receives new sensor inputs including spacing, the leader speed and acceleration. The receding sensing and planning process of the MPC will help update and correct the leader prediction immediately with the updated measurements and estimations, which helps to prevent a potential crash due to the poor prediction from the last single time step. On the other hand, the predicted safety bound will not affect the solution of the MPC in most of the time since the bound is most likely inactive. The safety bound can be calculated either with the OP default prediction model (10) or the simplest prediction of the leader trajectory that assumes the same speed for τ seconds.

$$x_{lead}^{pred}(t, k) = x_{lead}(t) + \frac{k}{\lfloor \tau/\Delta t \rfloor} \cdot v_{lead}(t) \quad (25)$$

This design makes use of both the leader history positions and speeds, with the speed reference being optional but providing redundancy to strengthen the model's robustness and regulating speed over/under-shootings relative to the leader, which we will show later in simulation. Moreover, the *XV*-Model makes it possible to make more designs that reduce traffic jams, which will be described in more detail below.

3.2. Newell MPC for anticipative driving using multiple look-ahead leaders: *Xmul*-Model

Thanks to recent advancements in sensing technologies, many AVs are now capable of detecting multiple downstream leaders in the same lane, particularly on curves and grades. Such innovation is unprecedented, which enabling AVs to see more downstream leaders than human drivers; see the example of Tesla's Autopilot in [Fig. 16\(a\)](#). Unfortunately, the additional information of multiple

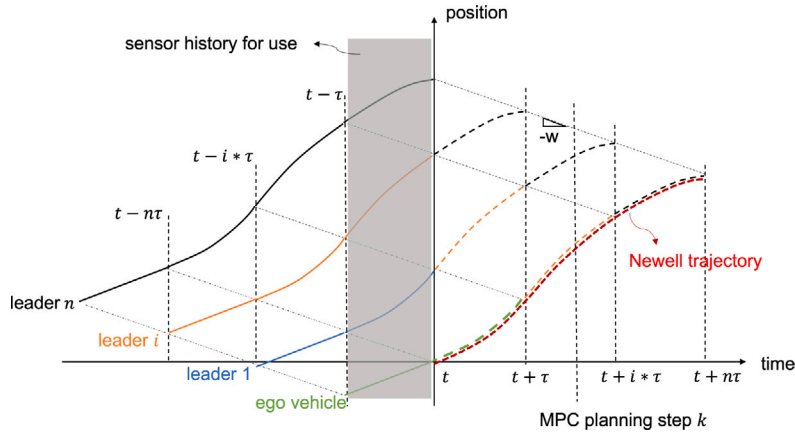


Fig. 3. $Xmul$ -Model design that incorporates multiple leader histories in the past τ seconds.

downstream leaders other than the immediate predecessor has not yet been incorporated into ACCs yet as no relevant design has been documented by either industry or academics. In contrast, human drivers with more experience are better equipped to use this information, known as driver anticipation (Treiber et al., 2007). For example, human drivers are willing to accept a small time headway, or reaction time, if they see the far downstream leaders are running at a stable speed and show no intention to decelerate. We conjecture this is because human drivers are able to use downstream traffic to predict whether a congestion wave will occur and propagate backwards. In other words, if the driver observes the leader's leader applying the brakes, he/she will instantly know that the immediate leader will soon slow down. Depending on the driver's ability to foresee the future, such behaviors would allow human drivers to respond even earlier than the immediate leader. These driving behaviors provide ACCs with a valuable lesson: seeing multiple leads can reduce response time, which not only improves safety but also contributes to string stability given the detrimental effects of response delays (Xiao and Gao, 2011; Jin and Orosz, 2016; Makridis et al., 2020). Considering the sensor's ability to detect several leaders, we advocate designing new ACCs that can maximize this technology's potential, which will also be studied here.

Now we propose an extension of the Newell MPC to fully leverage multiple downstream leaders that may be available in more advanced ACC systems. Suppose the ACC can detect $n \geq 1$ downstream leaders, then the prediction/planning horizon for the MPC is $n \cdot \tau$ assuming the leaders have the same τ values. According to the original Newell's CF model, the trajectory of the ego vehicle is the lower envelope between the shifted trajectory of any downstream leader i ($1 \leq i \leq n$) and its trajectory under free-flow conditions. Similar to the XV -Model, we use the shifted leaders trajectories as the reference; see Fig. 3; rather than the bound, in order to avoid the infeasibility issue. Note that using the Newell trajectory (the red dashed line) as the reference also makes the design robust to any discontinuity between the shifted trajectories from multiple leaders due to different τ_i and δ_i values. The total objective function is defined as follows:

$$\min_{j_{ego}} C_{Newell}^{mul}(t) = 1/2 \sum_{k=0}^{\lfloor n \cdot \tau / \Delta t \rfloor} w_x C_x^2(t, k) + w_a C_a^2(t, k) + w_j C_j^2(t, k) \quad (26a)$$

$$\text{s.t. } a_{ego}(t, k) \leq a_{max}(v_{ego}(t, k)) \quad (26b)$$

$$x_{ego}(t, k) \leq x_{lead}^{pred}(t, k) - \delta \quad \forall k \in \{0, 1, \dots, \lfloor \tau / \Delta t \rfloor\} \quad (26c)$$

where the definition of the sub-cost functions remains the same as above.

As shown in Fig. 3, the Newell trajectory in $Xmul$ -Model is now the combination of all shifted history trajectories from multiple leaders. The $Xmul$ -Model still only has a memory of past τ seconds, but incorporates multiple leader histories such that its planning horizon can be extended to $n \cdot \tau$. Mathematically, the Newell trajectory/reference for the $Xmul$ -Model at the planning time t is defined as:

$$x_{Newell}^{mul}(t, k) = x_{leader}^i(t + k \Delta t - i \tau) - i \delta \quad \forall k \in \{0, 1, \dots, \lfloor n \tau / \Delta t \rfloor\} \quad (27)$$

where i denotes the index of leader from 1, 2, ..., i , ..., n , and $i = 1 + \lfloor k \Delta t / \tau \rfloor$. Notice that when $n = 1$, it reduces to the $Xref$ -Model with a single leader.

Surprisingly, we observed that the $Xmul$ -Model can respond to a downstream wave as soon as the furthest leader n begins to change its speed, either due to an emergency brake or a small perturbation. In Fig. 3, when the furthest downstream leader n starts to change speed at $t - \tau$, such a change will be promptly reflected on the Newell reference at t , triggering a potential immediate response. By contrast, ACCs following a single leader must wait until the wave propagates back from the downstream to its immediate leader before starting to respond. As the wave needs $(n - 1)\tau$ seconds to travel, the possible response time of the regular MPC ACC

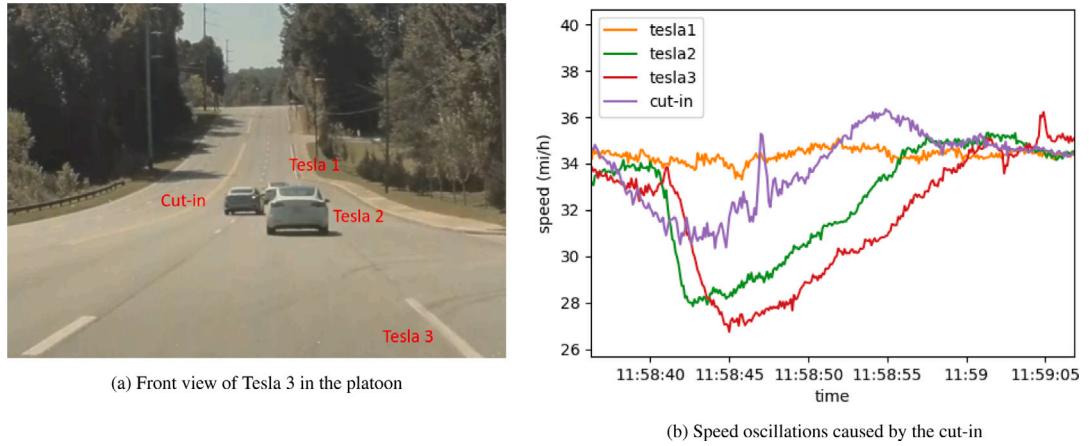


Fig. 4. Road tests of the oscillation caused by a cut-in and its propagation along a Tesla platoon: a platoon of three 2021 T Model-3s is cut by a human-driven vehicle.

is delayed by $(n - 1)\tau$ seconds, although the true observable response time still depends on the magnitude of the perturbation and the sensitivity of the X_{mul} -Model in reacting to the far downstream speed change.

The multiple leaders in Fig. 3 share the same τ and δ , which makes the Newell position reference a continuous curve. It is worth noting that different leaders' shifted trajectories may not overlap perfectly in the real world, but the X_{mul} -Model still works thanks to the design of using their positions only as a reference. We will justify this finding using numerical simulations in the following section. Alternative designs will also be shown in **Discussion** to address the different shifted trajectories from multiple leaders.

3.3. Congestion-mitigation extensions based on XV -Model

Developing a ready-for-practice ACC planner requires more than a standard CF model, mainly because ACC is also responsible for disruptive maneuvers such as being cut off or changing lanes. Human drivers exhibit a high degree of adaptability during disruptive maneuvers, as they are willing to accept spacings much smaller than the equilibrium values at the onset of these maneuvers and then gradually increase them until reaching equilibrium again. This process, which typically lasts around 20 or 30 s, is known as driver relaxation, which improves comfort and capacity (Kim and Coifman, 2013; Milanés and Shladover, 2016). Based on our knowledge, no ACC systems have displayed similar features of driver relaxation.

To provide more evidence, the paper conducts a simple experiment to evaluate the responses to cut-ins using Tesla's Autopilot, arguably the most advanced ACC system on the market. Three 2021 T Model 3s were running at 35mph in equilibrium, and the Tesla 2 in Fig. 4(a) was cut off by the Civic in the middle of Tesla 1 and Tesla 2. The speed profiles are shown in Fig. 4(b), which suggest that the Autopilot not only generates a larger speed perturbation than the cut-in, but also amplifies the upstream oscillation. As we know, the string stability theory can well explain the amplification of those upstream oscillations. In comparison, the oscillation amplification from the cut-in to its immediate follower has been less of a concern in the literature, until a recent study (Zhou et al., 2022a) found that it is caused by the absence of driver relaxation in ACC design. In short, the experiment indicates that Autopilot is not only string unstable, but also lacks driver relaxation. We believe that similar characteristics are likely shared by other ACC manufacturers. Therefore, improving string stability and incorporating driver relaxation are both worth more investigation for ACC design. In this subsection we will present two congestion-mitigation designs as extensions to the baseline XV -Model, which incorporate driver relaxation and strict string stability respectively.

3.3.1. Incorporating driver relaxation to smooth merging traffic: XV_{relax} -Model

When a cut-in vehicle enters the lane and becomes the new leader for the ACC follower, the Newell position reference x_{Newell} is supposed to immediately change to the shifted trajectory of the cut-in vehicle if there is no driver relaxation. This abrupt change of the reference trajectory often generates large position errors which in turn causes large speed reductions in the following ACC. After entering the new lane, the cut-in vehicle needs to re-adjust the spacing with its new leader, which normally undergoes a decelerate-and-accelerate process and forces the immediate ACC follower to have a speed oscillation as well. As depicted in Fig. 4, such an oscillation caused by the cut-in can further propagate upstream, resulting in potential stop-and-go waves.

To reduce the congestion caused by disruptive maneuvers, we must first contain the speed oscillation from the source, i.e. the perturbation of the immediate ACC follower after the cut-in. Similar to the scenario when following a fixed leader, we would expect the immediate ACC follower to experience a dampened oscillation compared to the cut-in leader. However, this can be dangerous due to the typically tight spacing after the follower has been cut off. Hence we propose to pursue marginal string stability in the cut-in scenario, meaning the ACC follower will seek to maintain speed identical to that of the cut-in leader. Correspondingly, the key design principles are (i) having the MPC follow the same speeds as the cut-in leader during the deceleration process, and (ii)

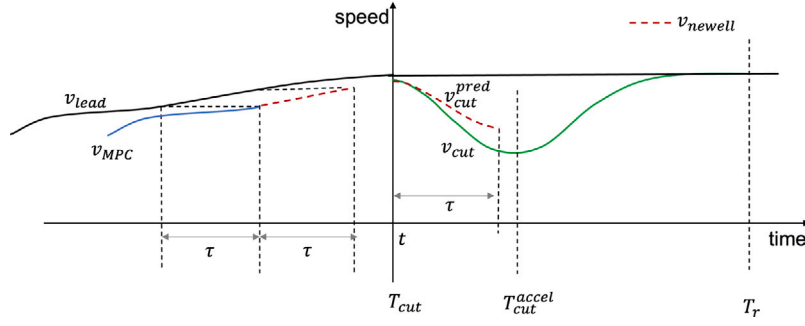


Fig. 5. The speed reference design in *XVrelax*-Model: the cut-in starts at time T_{cut} , T_{cut}^{accel} denote the time when the cut-in vehicle is about to accelerate, and T_r is the relaxation period.

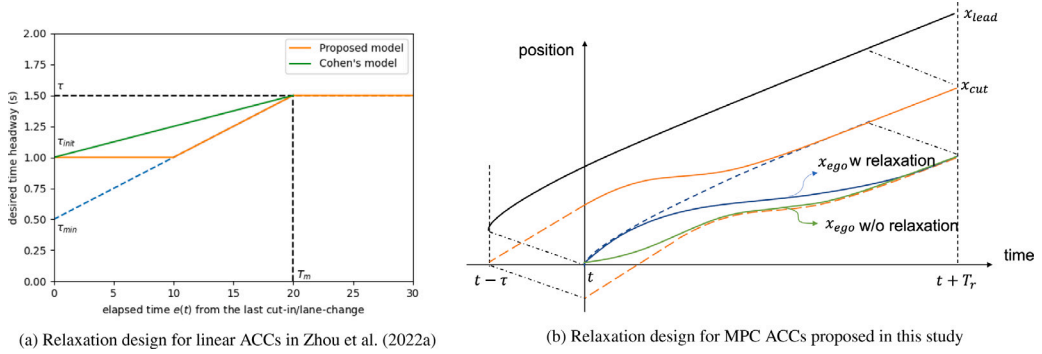


Fig. 6. Incorporating driver relaxation into ACCs. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

gradually recovering the spacing to the equilibrium value once cut-in leader begins to accelerate. We choose to recover the spacing only when the cut-in begins to accelerate because after that point, the follower MPC can easily regain the equilibrium spacing with smaller accelerations than the cut-in vehicle. This will enable the follower's recovery process to be similar to cruising, which is relaxing and does not require the use of brakes.

An intuitive option to follow the cut-in leader speed under the Newell MPC framework is to add an equality hard constraint for the first planning discrete step, $v_{ego}(t + \Delta\hat{t}) = v_{cut}(t)$, which, as previously stated, may cause infeasibility due to hard constraints. To circumvent this, we propose that the MPC can try to use the predicted cut-in speeds $v_{cut}^{pred}(t)$ as the speed reference; as shown in Fig. 5. Mathematically, the Newell speed reference in the *XVrelax*-Model can be defined as follows:

$$v_{Newell}(t, k) = \begin{cases} v_{cut}^{pred}(t + k\Delta\hat{t}) & T_{cut} \leq t \leq T_r \\ v_{cut}(t + k\Delta\hat{t} - \tau) & \text{otherwise} \end{cases} \quad (28)$$

for $\forall k \in \{1, 2, \dots, \lceil \tau/\Delta\hat{t} \rceil\}$ in the MPC planning horizon.

Since the sole purpose of the prediction is to track the leader speed, any plausible prediction model, such as (10) in OP, can be used. Formally, the predicted speed of the cut-in vehicle v_{cut} can be calculated as follows:

$$a_{cut}^{pred}(\hat{t}) := a_{cut}(t) \exp\{-\theta \cdot \hat{t}^2/2\} \quad (29a)$$

$$v_{cut}^{pred}(\hat{t}) := v_{cut}^{pred}(\hat{t}) + a_{cut}^{pred}(\hat{t}) \cdot \Delta\hat{t} \quad (29b)$$

$$x_{cut}^{pred}(\hat{t}) := x_{cut}^{pred}(\hat{t}) + v_{cut}^{pred}(\hat{t}) \cdot \Delta\hat{t} \quad (29c)$$

$$\hat{t} := \hat{t} + \Delta\hat{t} \quad (29d)$$

Recall that t is the starting time of each planning horizon and $\hat{t}$ is the time of the discrete steps that starts from 0. Notice that such prediction also gives the estimated future positions of the cut-in leader, x_{cut}^{pred} , which can be adapted to construct a safety constraint as shown in (23c) to prevent following ACC from collisions in the relaxation phase.

At the next stage when the cut-in leader accelerates from time T_{cut}^{accel} to re-adjust the spacing with its own leader, a good opportunity arises for the following ACC to recover to the equilibrium spacing. Now we introduce the recipe to incorporate driver relaxation in this process.

Fig. 6(a) shows how driver relaxation is incorporated into a linear ACC system in Zhou et al. (2022a), where the idea is to linearly recover the desired time headway $\tau(t)$ from the initial headway τ_{init} after the cut-in to the equilibrium value τ . In the same

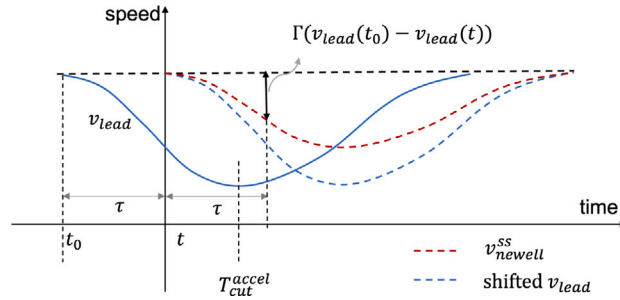


Fig. 7. The speed reference design in XV_{ss} -Model.

spirit, we extend the relaxation design to Newell MPC as depicted in Fig. 6(b). In contrast to the gradual change in the desired time headway for a linear ACC, the relaxation in the Newell MPC is now the transition of the Newell reference, x_{Newell} , which moves from the shifted trajectory of the original leader (blue dashed line) to the shifted trajectory of the new cut-in leader (orange dashed line) during a period of T_r (seconds).

To summarize, the incorporation of driver relaxation into the Newell MPC has two steps: (i) when the cut-in leader still decelerates from T_{cut} to T_{cut}^{accel} , the MPC is designed to track the speed of the cut-in, in the mean time the Newell position reference is the shifted history cut-in trajectory; see the blue dashed line in Fig. 6(b), and (ii) after the cut-in leader begins to accelerate at T_{cut}^{accel} , the relaxation process starts where the Newell position reference takes T_r seconds to gradually recover from its current position to the shifted history position of the cut-in leader. Mathematically, the two steps relaxation process is defined as follows:

$$x_{Newell}^{relax}(t, k) = \begin{cases} x_{cut}(t + k\Delta t - \tau) - x_{cut}(t - \tau) & 0 \leq t \leq T_{cut}^{accel} \\ x_{cut}(t + k\Delta t - \tau) - x_{cut}(t - \tau) + \alpha(t)[x_{cut}(t - \tau) - \delta] & T_{cut}^{accel} < t \leq T_r \end{cases} \quad (30)$$

where α ($0 \leq \alpha(t) \leq 1$) is a gradually varying factor to capture the transition of the reference trajectory during the driver relaxation process. Formally, the relaxed trajectory reference line is defined as:

$$\alpha(t) = \frac{\max(0, t - T_{cut}^{accel})}{T_r} \quad (T_{cut} < t < T_{cut} + T_r) \quad (31)$$

Incorporating driver relaxation helps to reduce oscillations, albeit at the expense of smaller spacings that can be challenging. Nevertheless, we believe that the proposed XV_{relax} -Model is still safe because, during deceleration the ACC, by design, tries to maintain the same speed as the cut-in leader, which will roughly keep the initial spacing unchanged after the cut-in. In other words, as long as the initial spacing after the cut-in is safe, the relaxation process will not affect the ACC with significantly smaller spacings. Thus whether it is safe to incorporate driver relaxation largely depends on the initial spacing after the ACC is being cut off. To ensure safety in cut-in scenarios, we also add the safety constraint similar to cases when following a fixed leader. When a cut-in leader is detected, the predicted leader position in the safety constraint (23c) will be replaced with the predicted cut-in position (29c), which prevent potential collisions.

The reaction model for the ACC to make a lane change is almost the same with the cut-in case. One just need to replace the cut-in leader with the new leader on the target lane in Eqns from (29) to (31).

3.3.2. Achieving the strict string stability to further dampen oscillations: XV_{ss} -Model

Another desired function of the ACC is to dampen congestion waves, which necessitates string stability in the model. In control theory, the transfer function Γ in the frequency domain is used to measure the string stability, or how much the congestion waves can be dampened. As marginal string stability requires $\Gamma = 1$, strict string stability means $\Gamma < 1$, i.e. the speed/spacing oscillations of the leader will be dampened by the follower. Note that when the leader is accelerating, the marginal string stability is already optimal for traffic efficiency because the strict string stability will produce lower speeds and extra spacings that waste capacity. Hence the strict string stability is only designed for the decelerating case. Rigorously, XV_{ss} -Model is only strictly string stable for deceleration but marginally string stable for acceleration to avoid capacity loss.

Unlike previous ACC models that attempt to achieve string stability by manipulating CF parameters or optimization heuristics, the XV_{ss} -Model offers a more direct method by tracking a string stable future speed reference based on adjustments to the leader's history speeds. As shown by Fig. 7, the adjustment is to multiply the shifted leader speeds with Γ to obtain the Γ -string-stable speed reference v_{Newell}^{ss} . Mathematically, it can be defined as follows:

$$v_{Newell}^{ss}(t, k, \Gamma) = (1 - \Gamma)v_{lead}(t_0) + \Gamma \cdot v_{Newell}(t, k) \quad (32)$$

where $v_{lead}(t_0)$ is the equilibrium speed of the leader and $a_{lead}(t_0) = 0$.

To make it easier for the MPC to follow the strict string stable speed reference, we propose to relax the position reference in a similar way of what has been presented in (30) for the XV_{relax} -Model. We relax the position reference when the leader decelerates,

and gradually recover to the equilibrium spacing after the leader starts to accelerate at time T_{lead}^{accel} . Mathematically, the position reference is designed as follows:

$$x_{Newell}^{ss}(t, k) = \begin{cases} x_{Newell}(t, k) - x_{Newell}(t, 0) & a_{lead}(t) \leq 0 \\ x_{Newell}(t, k) - \gamma(t)x_{Newell}(t, 0) & a_{lead}(t) > 0 \end{cases} \quad (33)$$

where $\gamma(t)$ ($0 \leq \gamma(t) \leq 1$) is a gradually varying factor to capture the recovering process of the driver relaxation when the leader starts to accelerates again. In this process, the follower is able to advance with smaller accelerations/speeds than the leader to regain the desired spacing. The relaxation coefficient $\gamma(t)$ is designed as:

$$\gamma(t) = 1 - \frac{t - T_{lead}^{accel}}{T_r} \quad (T_{lead}^{accel} \leq t \leq T_r) \quad (34)$$

where $t - T_{lead}^{accel}$ is the elapsed time after the leader accelerates.

To summarize, this section introduced a family of Newell MPCs. We found the Newell trajectory can be used either as a upper bound or a reference to achieve similar performance to the original Newell's CF model. The leader speed history can be leveraged as an extra reference to improve the model robustness and prevent speed over/under-shootings. The same design philosophy was extended to ACCs with multiple look-ahead leaders to achieve driver anticipation that reduces reaction time and improves string stability. Based on the baseline *XV*-Model, two congestion-mitigation designs were developed, including driver relaxation to smooth merging traffics and strict string stability to effectively dampen congestion waves.

4. Simulation results

This section discusses the performance of the Newell MPCs designed above using numerical simulations. We use the same MPC solver, Acado (Diehl, 0000), that runs on real cars in OP's codebase, implying that all following MPC simulations can be transferred to the real world. Nevertheless, there may still be discrepancies between simulations and reality due to different inputs and low-level control, as all leader speeds and spacings are perfectly emulated in simulations but are susceptible to sensor errors on real cars. We will run those MPCs first in simulations and test their functioning on real cars in the next section.

4.1. Numerical results of the *Xbound*-Model

Now we show the simulation results of the *Xbound*-Model in (16) where the leader history trajectory is used as follower's upper bound. The MPC is tested in the scenario where the leader performs a decelerate-and-accelerate oscillation.

The Fig. 8(a) showcases one example of the planning step, where the history leader trajectory, the shifted leader trajectory, i.e. the Newell bound, and the solution of the ego trajectory using the *Xbound*-Model are all presented in the diagram. Fig. 8(b) provides a zoom-in view of how the ego vehicle's trajectory is effectively bounded by the shifted leader trajectory as a result of the soft constraint in (18). The two lines are not exactly overlapping thanks to the other cost terms introduced in (20).

In Fig. 8(c), we compare the *Xbound*-Model with the default MPC in OP (OP-MPC), where the *Xbound*-Model shows the marginal string stability while the OP is string unstable which undershoots and overshoots the leader speed in the deceleration and acceleration case respectively.

The OP-MPC also shows larger oscillations in its spacing. The trajectory is compared in Fig. 8(d), where we see the capacity (spacing) loss from OP-MPC compared to the *Xbound*-Model in the acceleration oscillation. Combined with the acceleration profile in Fig. 8(c), such extra spacing/capacity loss is induced by small accelerations, which is a result of the constraints on acceleration/jerks for the driving comfort purpose.

4.2. Testing and comparing *Xref*-Model and *XV*-Models

We test and compare the *Xref*-Model using the only the position history and *XV*-Model using both the speed and position histories. The objective is to analyze if adding the speed reference has an impact in CF scenarios where the position reference accumulates substantial errors. To address this, a simulation is designed in which the leader vehicle is a high-performance car with better acceleration capabilities, implying that the follower ACC may experience significant spacing errors with the Newell position reference due to its engine limitations. Then we compare the results of *Xref*-Model MPC against the *XV*-Model.

The Fig. 9(a) shows the issue of the *Xref*-Model that uses only the leader position as the reference. The *Xref*-Model experiences significant speed/acceleration oscillations around step 9200 when closing the gap with the faster leader. More specifically, the *Xref*-Model displays too aggressive speeds when trying to minimize the position errors. It overshoots the leader speed a lot, and later re-adjusts the spacing with large decelerations. In comparison, *XV*-Model incorporates an additional speed reference to prevent large speed overshoots while reducing the position errors. When the leader already stabilized, the speed reference v_{Newell} also helps stabilize the follower to prevent large speed oscillations.

To see more details, starting from approximately step 9200, the differences of the two MPC followers start to grow. The planning diagram at the step 9200 is shown in Fig. 9(b), where the shifted leader trajectory is above both of the two following MPCs due to larger accelerating performances of the lead vehicle. In Fig. 9(c), the leader speed, i.e. the Newell speed already stabilized, the *XV*-Model decelerates thanks to the speed reference, but the *Xref*-Model continues to overshoot the leader speed to close the

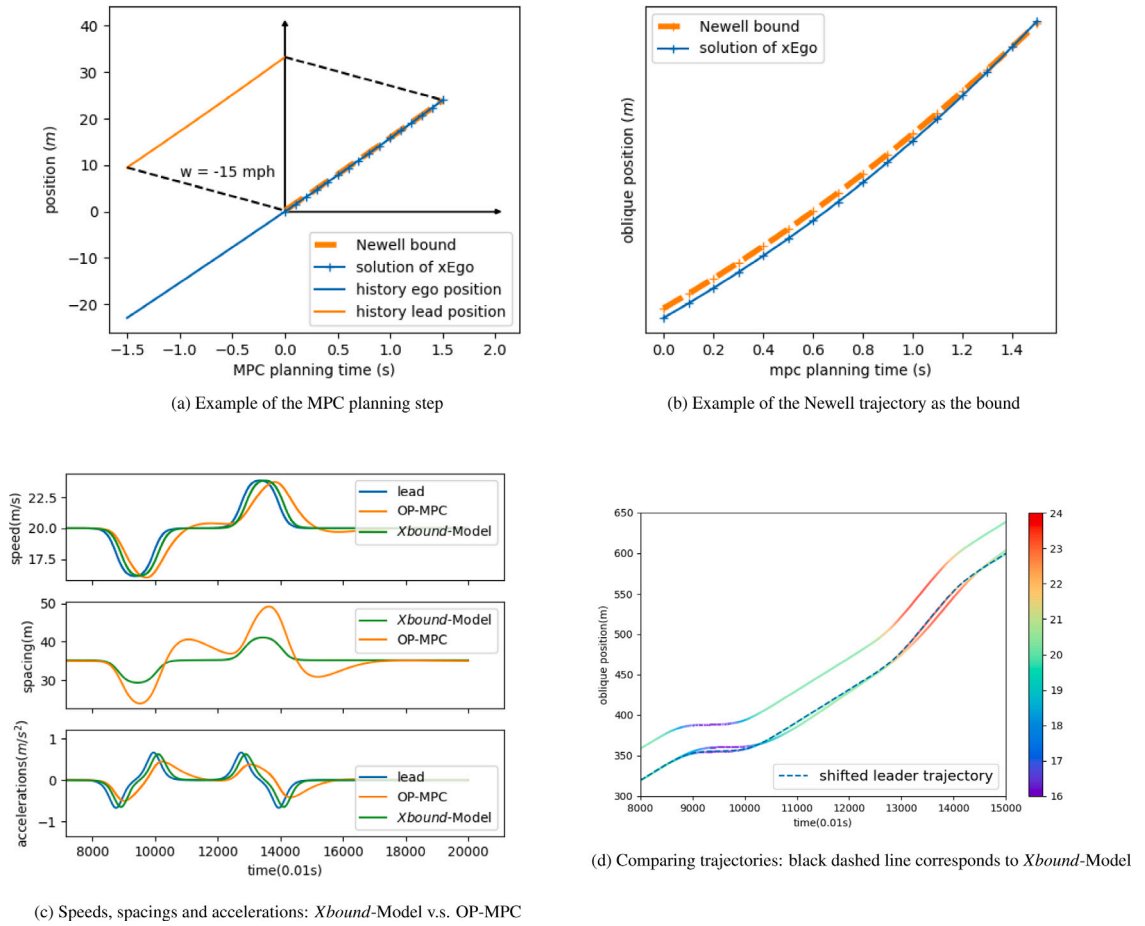


Fig. 8. Simulation of the *Xbound-Model* and comparison with OP-MPC.

position errors because it only has the position reference. It shows that the speed regulation is necessary for better stability of the controller.

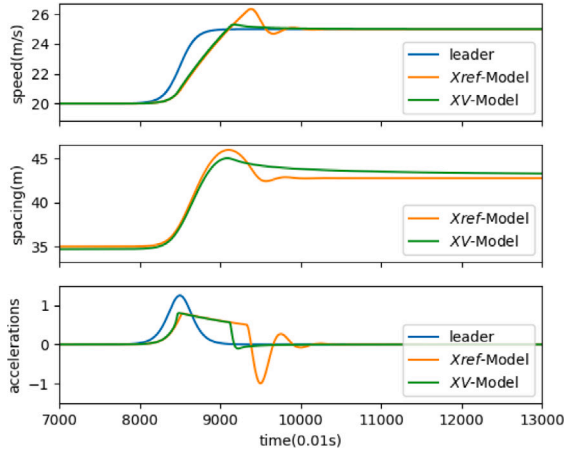
Through the comparison, we can conclude that *XV-Model* is not only more robust against sensor faults due to input redundancy, but also reduces speed over/undershoots with the additional speed regulation. Such an advantage of the *XV-Model* can be more significant in the presence of large and stable position errors that may result in oscillatory speeds.

4.3. Numerical testing of the *Xmul-Model*

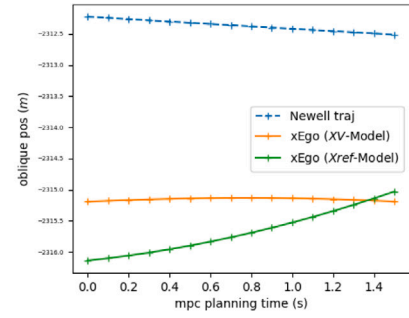
Now we present the simulation results of the *Xmul-Model* designed to incorporate multiple downstream leaders.

In this simulation, we have a four vehicle platoon where the ego vehicle (*Xmul-Model*) has 3 downstream leaders, lead 1 and lead 2 are two OP-MPCs and lead 3 is a human-driven vehicle that performs a perturbation. Notice that their true sequence in the platoon should be lead 3, lead 2, lead 1 and ego *Xmul-Model* from the upstream to the downstream. The Fig. 10(b) depicts one planning step of the *Xmul-Model*, where the history trajectories of three downstream leaders are shifted and projected to the MPC planning horizon. The discontinuity between the adjacent shifted trajectories is caused by different τ and δ values used by the unknown leaders. However, those errors can be ignored when using those shifted trajectories only as the references, considering the jerk and acceleration constraints will always produce a smooth planning trajectory. In this case, we assume the same $\tau = 1.5$ (s) and $\delta = 8.5$ (m) for all leaders.

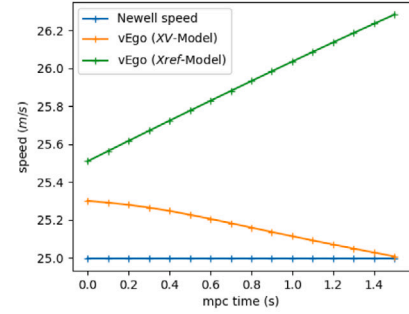
Fig. 10(a) shows the speed, spacing and accelerations of all four vehicles in the platoon. When introducing the design for the *Xmul-Model*, we also conjectured that because of the driver anticipation, the reacting time of the *Xmul-Model* can be moved forward by $(n-1)\tau$ seconds in theory. In Fig. 10(a), the *Xmul-Model* reacts almost the same time with its predecessor, the second vehicle in the platoon, which means its responding time is brought forward for about τ seconds. Although the reduction of the reaction time is not as large as $(n-1)\tau = 2\tau$ seconds, it still verifies that the *Xmul-Model* can benefit from the downstream leader information and possibly help prevent collisions if a downstream collision happens unexpectedly. The similar impact of reduced reaction time is also found in Treiber et al. (2007) which investigates driver anticipation with a different CF model.



(a) Comparing the speed, spacing and accelerations

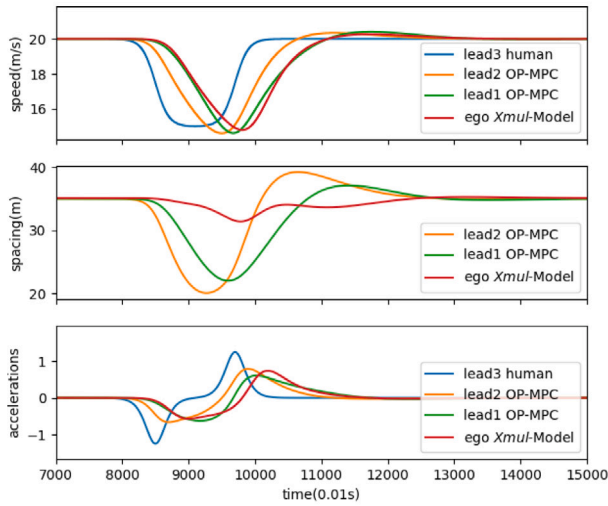


(b) Comparing trajectory plans at step 9200

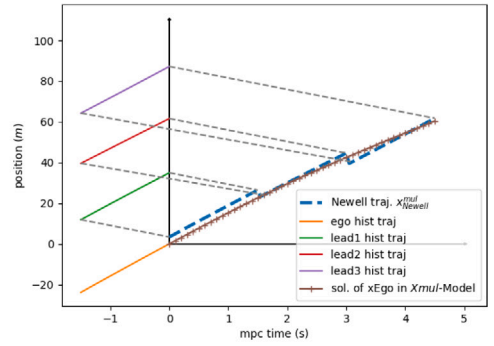


(c) Comparing speed plans at step 9200

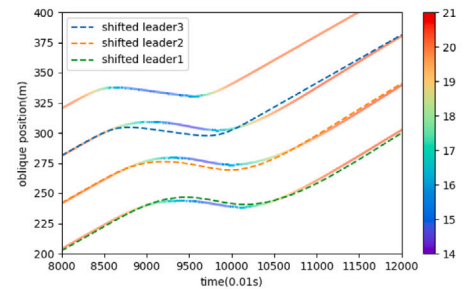
Fig. 9. Comparison of the Xref-Model and XV-Model.



(a) speed, spacing and accelerations



(b) Example of trajectory planning



(c) trajectory details

Fig. 10. Simulation results of the Xmul-Model.

Surprisingly, we found that the $Xmul$ -Model also improves string stability without any specific design as required for the single leader Newell MPC $XVss$ -Model. In Fig. 10(a), the speed oscillation on the $Xmul$ -Model is smaller than its immediate leader mpc-follower2. The reasons for the improved string stability of the $Xmul$ -Model could be complex and needs more theoretical investigation. A similar finding is reported by Treiber et al. (2007), which shows by simulation that a variation of the IDM model with multiple look-ahead leaders can also improve string stability. We conjecture that driver anticipation contributes to the improved string stability in a similar fashion either for the IDM or the $Xmul$ -Model, although the underlying mechanism needs further investigation.

The Fig. 10(c) showcases the trajectories of all vehicles in the platoon, where the dashed lines denote the shifted leader trajectory with τ and δ . Interestingly, the two string unstable OP MPCs both experience overshootings followed by under-shootings in the space-time diagram. It also indicates that the overshooting ($\eta < 1$ in Chen et al. (2012)) in the deceleration phase does not necessarily guarantee string stability.

4.4. Testing driver relaxation in $XVrelax$ -Model against a cut-in

We simulate the $XVrelax$ -Model against a cut-in in order to test the function of driver relaxation. As shown in Fig. 11, a cut-in vehicle decelerates after the lane-change to adjust its spacing with the target lane leader, whose speed and trajectory are shown in Fig. 11(a) and Fig. 11(b) respectively. In the top graph in Fig. 11(a), the $XVrelax$ -Model follows the speed of the cut-in vehicle according to the Newell speed reference design in (28). In this process, the predicted cut-in speeds result from the model (10), where the exponentially decaying acceleration and the corresponding speed predictions are shown in Fig. 11(c) and Fig. 11(d). Note that the ACC's spacing does not significantly decrease because it maintains nearly the same speed as the cut-in leader in the deceleration phase; see Fig. 11(a)'s spacing and speed plots. This supports our earlier claim that relaxation is safe so long as the initial spacing right after the cut-in is not too small.

Recall the relaxation design in (30) has two steps. For the first stage, or the deceleration phase, the Newell position reference is the history cut-in position shifted to the current ego position of the follower; see Fig. 11(e); causing a zero cost which does not take effects. For the second phase, the recovery of spacing happens after the leader starts to accelerate. The follower Newell MPC now gradually relax its trajectory reference from its current position to the shifted cut-in trajectory; see the transition of the x_{Newell} (orange dashed line) towards the shifted cut-in history position (green dashed line) in Figs. 11(e)–11(g). In this process, the cut-in leader already starts to accelerate, the driver relaxation in the following $XVrelax$ -Model produces smaller accelerations compared with the cut-in leader which helps recover the spacing. Such “cruising” behaviors effectively prevent the speed overshootings as shown in the speed profile in Fig. 11(a).

Overall, the $XVrelax$ -Model produces a trajectory as shown in Fig. 11(b), where the dashed line is the Newell shift of the cut-in trajectory. The color in Fig. 11(b) suggests the speed of the cut-in and follower vehicle, which is a justification of our design presented in Fig. 6(b). The whole reaction process after the cut-in is more congestion-friendly compared with the non-relaxation case where the follower is likely to abruptly decelerate at the beginning and to catch up with speed overshootings later.

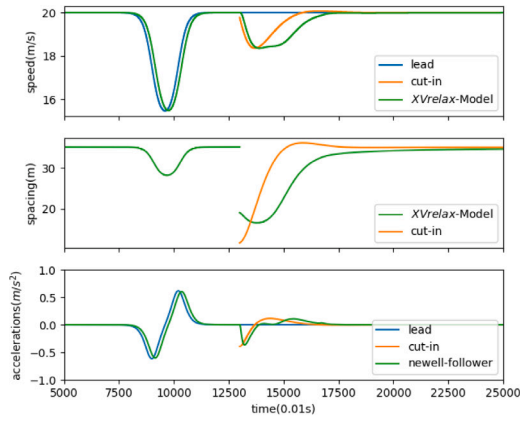
4.5. Testing the strict string stability of the $XVss$ -Model

Here we present the simulation results of $XVss$ -Model that aims to achieve the strict string stability parameterized with Γ . The Fig. 12(a) presents the speed, spacing and acceleration results. In the speed profiles, we see the ego vehicle is able to follow the strict string stable speed (the purple dashed line) when the leader decelerates, as we expect from Fig. 7. When the leader starts to accelerate again, the $XVss$ -Model gradually recovers its desired spacing, i.e. slowly moves towards the leader trajectory shifted by τ and δ ; see example in Fig. 12(c), where the position reference (red dashed line) is interpolated between the current position and the regular Newell trajectory (orange dashed line) according to the rule in (33). As the ACC gradually recovers to the original non-relaxation Newell reference, the ego vehicle starts to deviate from the v_{Newell}^{ss} designed for the strict stability (orange dashed line) and chooses a lower speed to regain more spacing, as depicted in Fig. 12(d).

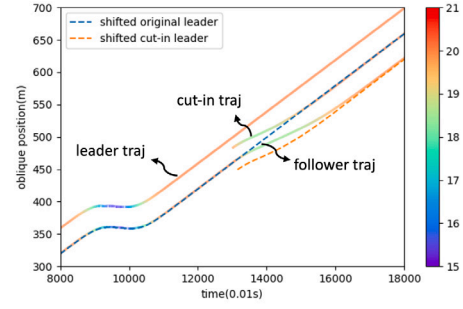
The Fig. 12(b) compares the trajectory of the marginal SS and the strict string stability achieved by the $XVss$ -Model. Apparently the strict string stable trajectory is above the marginal one, coinciding with the convex pattern in the η theory in Chen et al. (2012). It also shows that strict string stability produces smaller spacings, which more or less sacrifices safety.

5. Field experiments

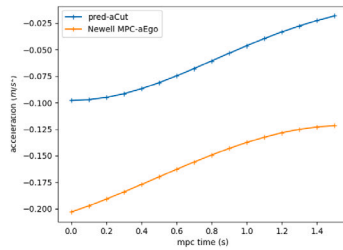
As stated previously, the simulations utilize the same professional MPC solver that OP employs for real cars. Hence, these models likely can be transferred to the real world. However, this needs to be verified and their true performance need to be tested against factors not accounted for in simulations, such as sensor measurement errors and imperfect low-level control. Consequently, this section will use a market 2019 Honda Civic to: (i) verify the feasibility of the proposed Newell MPC planner on market ACCs, (ii) evaluate the model performance, and (iii) identify the potential limitations of implementing the model on real cars.



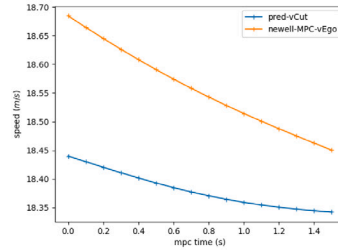
(a) Speed, spacing and accelerations



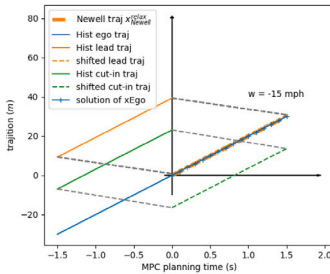
(b) Trajectories of the original leader, the cut-in leader, and the XVrelax-Model follower



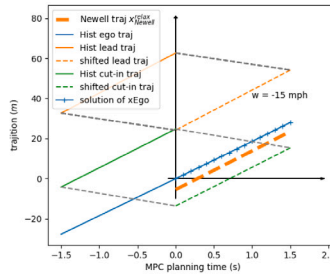
(c) Simple predictions of cut-in accelerations



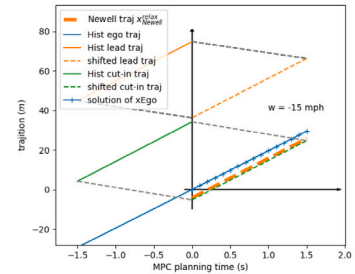
(d) Following the same speed of the cut-in leader



(e) Planning in the beginning of driver relaxation



(f) Planning in the middle of driver relaxation



(g) Planning in the end of driver relaxation

Fig. 11. Testing driver relaxation in XVrelax-Model that prevents speed amplifications caused by cut-ins: In the legends “hist” means history and “traj” is short for “trajectory”. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

5.1. Experimental method

The experimental method of testing custom ACC algorithms on market car models was first introduced in our previous work (Zhou et al., 2022b) and Zhou et al. (2022a). To make this paper self-contained, we added some details here as well. The hardware preparation is rather simple. We only need to connect the Comma.ai’s after-market device Comma Two, to the interface of the ACC unit of the car, which will be overwriting the stock ACC algorithms and running our custom OP instead. A more detailed installation tutorial can be found here (Comma.ai, 2020). The experiment set-up is shown in Fig. 13.

It is worth noting that an ACC system consists of the upper level planner that outputs the desired acceleration, and the low-level control system which is responsible for execution. The upper-level planner can be a CF model, a linear controller or an MPC. The low-level control system includes multiple components such as the proportional-integral-feedforward (PIF) controller and the actuator model translating command accelerations to gas/brake values; see details in Zhou et al. (2022b). All those low-level control parameters are specific to car models and need careful tuning. In this paper, to test the feasibility of the Newell MPC as a new planner, we only replaced the current MPC planner in OP, and maintain the default low-level controller and the actuator model.

Unlike linear ACCs and CF models, implementing an MPC for real cars needs both coding and compiling to ensure the running speed. To construct an MPC, one needs to translate the mathematical formulation as required by a professional solver. The authors use the same MPC solver as OP does, i.e., Acado (Diehl, 0000). The detailed coding and compiling process includes a lot of technical

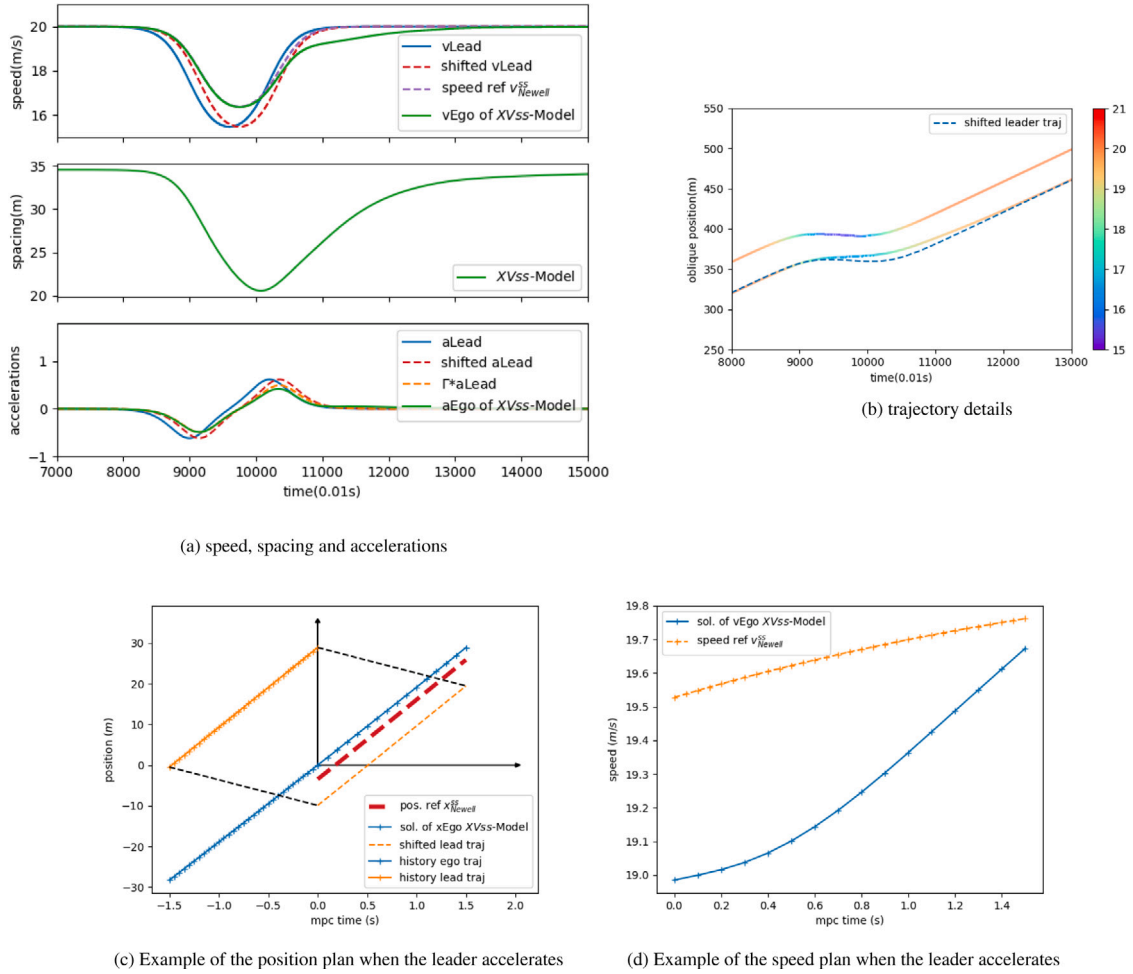


Fig. 12. Simulation results of the XV_{ss} -Model that achieves the strict string stability: In the legends “pos” means “position”, “sol” means “solution”, and “ref” is short for “reference”. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

details, unfortunately Comma.ai does not provide any tutorials on this. The authors successfully explored the method and plan to publish it later since it is not within the scope of this paper.

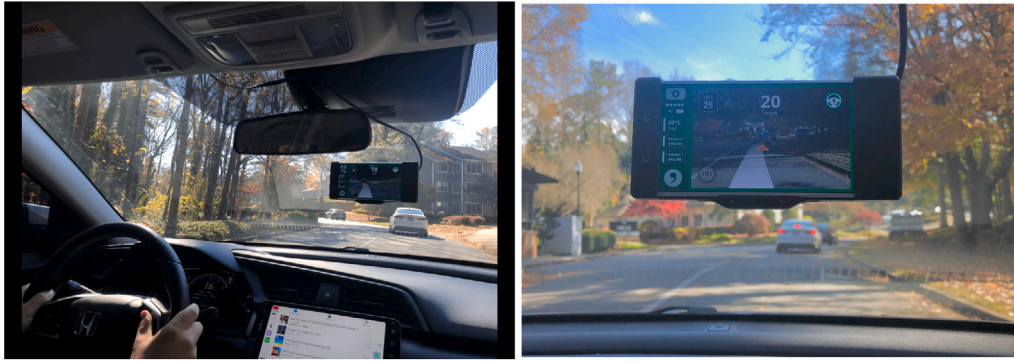
The driving logs from Comma Two include all the ACC-related variables as well as the CAN (Car Area Network) bus messages of the car, such as v_{ego} and a_{ego} . The method does not require any modifications of a regular commercial car and the full access of the ACC system allows us to dive into the details of the control algorithm and analyze its impact. In theory, the experiment method can be applied to testing any other CF models and even lane-changing MPCs.

5.2. Field test of the newell MPC

This section discusses the empirical results of the baseline XV -Model on a 2019 Honda Civic. Due to liability concerns, the authors can only test the model on local roads at low speeds.

Fig. 14 reports the planning details and trajectories from two field drives. The leader speed is directly measured by the sensor, and the position is calculated based on the spacing outputs. Overall, the trajectory plots in Figs. 14(a) and 14(b) demonstrate that the XV -Model is feasible, although the results are not good because we still see much discrepancies between the Newell trajectory and the true ego trajectory.

Next we identify the limitations of implementing this model on current market ACC vehicles. Unlike the simulations, empirical results of the model rely on sensor measurements for the history input, and the execution of desired accelerations largely depends on low-level control. Note that our test vehicle, a 2019 Honda Civic does not have an available radar for OP to use. Thus the leader spacings and speeds are both estimated using vision techniques with a mono-camera, which partially explains the choppy spacing/speed measurements in Figs. 14(c) to 14(f). Some points in the figures are outliers, which helps to justify our earlier decision to incorporate soft constraints rather than the default hard bound in the origin Newell’s model. However, the authors postulate that,



(a) Road test of the Newell MPC: the black phone, Comma Two, is connected to the camera interface of the Civic's ACC module (b) Device UI: green box indicates the Newell MPC is active, the orange triangle labels the detected lead vehicle ahead

Fig. 13. Test the Newell MPC on the road: experiment set-up.

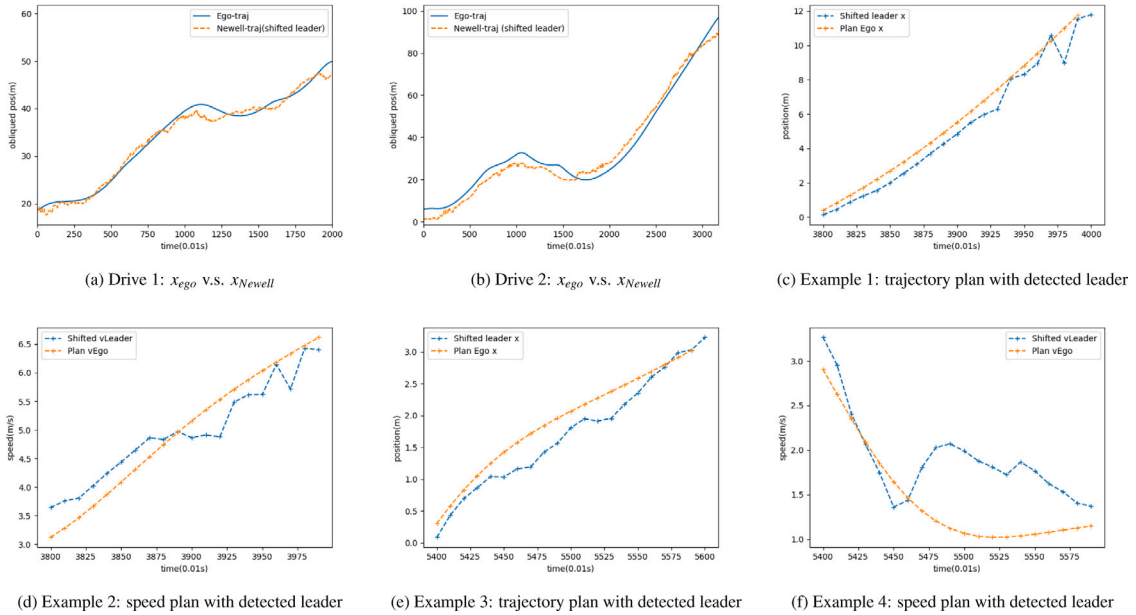


Fig. 14. Road tests of the XV-Model on a 2019 Honda Civic: sensor measurements and control details.

with a different car model equipped with a radar or more advanced sensors such as multiple cameras/lidar, the leader measurements can be more accurate and smooth, which will improve the performance of the Newell MPC.

The experiment results presented here are the holistic outcome of the full ACC control loop, which consists of not only the upper-level planner (XV-Model), but also the sensor (a mono-camera), and the low-level controller. Consequently, the discrepancy between the planned and true trajectory is a result of all those components. The sensor delay, measurement errors, the actuator delay, and actuator errors all contribute to the discrepancy here. Our previous work (Zhou et al., 2022b) has specifically studied the significance of the low-level controller in the performance of ACC systems. In other words, to achieve the desired performance of the Newell MPC planner proposed in this paper, the low-level control has to be carefully designed as well, including the well-known actuator delay that has been extensively studied in the literature.

Those experiment results indicate that the Newell MPC appears to be sensitive to sensor errors, as outliers in spacing or speed measurements can both contribute to large discrepancies. Note that those results are not satisfactory because we have not applied any smoothing techniques to the raw sensor measurement. To address noises or errors in sensor measurements, we hypothesize that a filter could be used to pre-process the history data that will be fed to the planner, thereby helping to smooth out the Newell speed and position references. The similar concept can also be applied to situations in which the leader causes undesirable perturbations, such as a “phantom” brake, which the filter can choose to ignore. In this regard, pattern recognition techniques may be required to filter out these small perturbations and determine whether it is safe to neglect them. The XV_{SS}-Model can also be used to dampen these small oscillations, preventing them from spreading across the platoon.

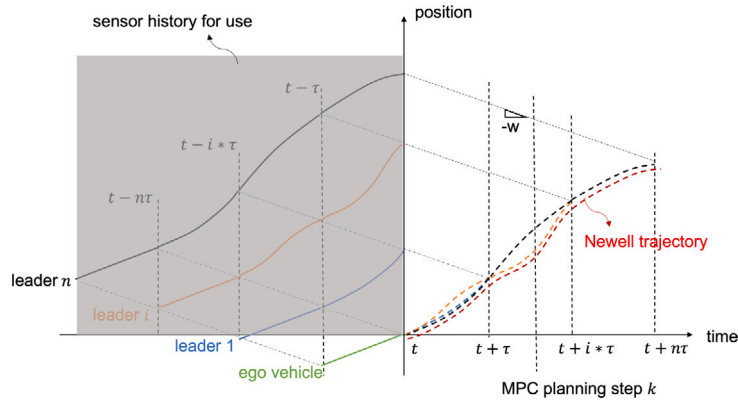


Fig. 15. Extensions for *Xmul*-Model with longer memories: the Newell trajectory becomes the lower bound of shifted leader positions from different leaders.

6. Discussion

This paper presented a family of novel MPC ACCs inspired by the well-known the Newell's CF model in traffic flow theory. To circumvent the issues and challenges of the predict-and-optimize method prevalent in current MPCs, the proposed Newell MPCs utilize the history of the leader, which is easily obtainable using existing sensors on market ACCs.

The family of Newell MPCs developed in this paper enables ACCs to circumvent prediction efforts/challenges while illustrating substantial potential to improve traffic efficiency, i.e. alleviating “phantom” congestion waves. We find that: (i) the historical leader trajectory can either be used as a bound or a reference to achieve marginal string stability; (ii) the leader history speed can be used as an additional reference along with the history position, which helps to improve the model robustness and prevent speed over/under-shootings; iii the single-leader Newell MPC can be extended to incorporate multiple downstream leaders, achieving driver anticipation that reduces reaction time and improves string stability, and (iv) the Newell MPCs are extendable to incorporate congestion-mitigation designs, including driver relaxation to smooth merging traffics, and the strict string stability to dampen congestion waves.

The field test on real roads validates the desired performance of the baseline Newell MPC *XV*-Model using a market ACC vehicle, 2019 Honda Civic. The experiment results suggest that the performance of the Newell MPCs not only depend on the model itself, but also on sensor precision and low-level execution. Interested readers are encouraged to test/improve the baseline *XV*-Model shared at <https://github.com/HaoZhouGT/openpilot>. As one of the limitations of this work, we need more experiments to identify the detailed impact of factors other than the planner model, such as how does actuator/sensor delay would change the performance of the Newell MPCs, and whether it is better to address the delay in upper-level planner or low-level controller. Since ACC's string stability is an overall outcome of the whole system, we cannot guarantee the proposed models can achieve the desired performance without carefully addressing other influencing factors.

Regarding the proposed *XVrelax*-Model and *XVss*-Model, although these two extensions already include the safety bound in their design, more research is needed to further examine their feasibility in practice, especially the *XVrelax*-Model which accepts smaller spacings after cut-ins. Human drivers tend to accept small spacings only when they can confidently anticipate future incidents, i.e., a potential emergency brake of the leader is less likely to happen. The current *XVrelax*-Model is still incapable of this and additional safety design needs to be added to safely execute driver relaxation in order to smooth the merging traffic.

Remarkably, the essence of the proposed family of Newell MPCs is to “copy” the leader trajectory in congestion (i.e., achieving marginal string stability). Although this design is reasonable in most cases because it does not amplify oscillations, it is important to investigate whether consistently “copy” the leader trajectory is prudent, particularly in the case of abnormal lead behaviors with high variances. To address this, the proposed *XVss*-Model featuring strict string stability can be applied, which could help dampen those oscillations in the platoon. However, additional research effort is still required to decide when to switch to the *XVss*-Model and how strict string stable the model should be in order to balance safety and efficiency. Correspondingly, the key parameter Γ in the *XVss*-Model must be optimized to consider the leader oscillation pattern, platoon size, and safety hazards. Moreover, to recognize these abnormal leader behaviors, the acceleration or jerks in the leader's past can be leveraged for driving pattern recognition, which may also require relevant machine learning algorithms. In addition, the *Xmul*-Model, which employs multiple downstream leaders rather than just one, can also be used to mitigate the negative effects of an abnormal leader. The performance of this method also needs further investigation under abnormal traffic conditions.

Next, we share some future research directions in the following subsections.

6.1. Implications on future self-driving systems

The *Xmul*-Model proposed in this study investigates the possibility of utilizing advanced sensing and computer-vision technologies. Furthermore, we recognize that the *Xmul*-Model could benefit from additional extensions along these lines. For instance, if the

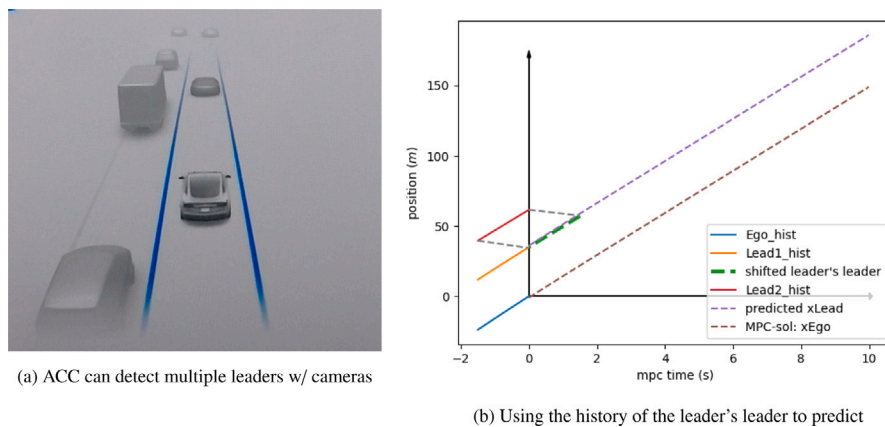


Fig. 16. Multiple leaders' information can be helpful for existing MPC planners based on predictions. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

MPC has longer memory of the past, more history information can be utilized to improve input redundancy. However, not every human follows the exact Newell's CF model, aggressive or timid behavior is common. As Fig. 15 shows, the leader i might not be a perfect shift of its leader $i + 1$. Correspondingly, multiple shifted leader trajectories can be projected to the same planning horizon, e.g. there are 3 different shifted references from t to $t + \tau$. To solve this conflict, we can extend the Newell's model by always picking the lowest bound of all shifted trajectories, which gives a safer reference to follow; see the red dashed line. Doing so can effectively get rid of the effect of aggressive downstream leaders, whose impact on string stability merits investigation in future research.

Another takeaway from this paper is that the look-ahead sensing can be the answer to better predictions. The study shows that current MPCs are sensitive to accurate leader predictions for improved performance, especially string stability. Despite our preference for history over prediction, we do not advocate abandoning all MPC models that rely on predictions. Instead, we would like to emphasize that the spirit of Newell MPCs can be easily incorporated into current MPCs by recognizing that look-ahead sensing can enhance prediction. Next, we show how to incorporate the Newell MPC design into current predict-and-optimize MPCs using advanced sensing techniques. As Fig. 16(b) indicates, if the ACC's sensor is able to detect the leader's leader, we can certainly utilize that information, i.e. the history trajectory of the leader's to leader, to improve the prediction of the future trajectory of the immediate leader. In Fig. 16(b), the green dashed line, i.e. the shifted history of the leader's leader, is almost the perfect prediction of the leader's movement. While a machine learning model could potentially achieve similar performance, it certainly requires big data and deep models, which can be costly. Instead of laboring over machine learning models, we can certainly better leverage the existing sensor technologies, i.e. the leader's leader to obtain a better prediction as the input for the ACC system, which can easily benefit all the control objectives in current MPC designs. Consequently, we strongly recommend that future ACC systems should use cameras to detect multiple leaders, which is a simple way to improve their current MPC performance.

We recommend that future ACC systems investigate the possibility of utilizing adjacent lane detections more effectively. The paper shows that the Newell MPC can be extended to leverage the history information of multiple downstream leaders if they are detectable by new sensors. Such information is already available in some advanced-level systems such as Tesla and Kia. The authors have done a simple experiment and show the detection results on Tesla's Autopilot display in Fig. 16(a). Note that cameras are even able to detect multiple vehicles from the adjacent lanes, which can be another source of reference points for the ego vehicle. For example, when a congestion wave propagates back, the downstream leaders from adjacent lanes may brake and signal earlier than those on the current lane. Human drivers are able to react to those signals from adjacent lanes, possibly decelerating in advance even if their leaders on their own lane are not decelerating yet. Unfortunately similar behaviors have not been found on ACC systems. Since the sensing technologies are already able to detect multiple downstream leaders including those from adjacent lanes, we encourage more investigation in the future work to leverage them.

6.2. Additional future works

ACCs have been shown inefficient in previous studies (Makridis et al., 2021) and entail environmental costs. While our study does not specifically address environmental costs of the presented models, recent study (Liu et al., 2021) have found string stable platoons using CACC technologies can significantly reduce vehicle emissions especially for highway traffic. The proposed Newell MPCs in this study achieve string stability without requiring vehicle-to-vehicle communication and are ready for implementation on current market ACC vehicles with no additional devices. Hence, the proposed Newell MPCs will likely reduce environmental costs in a similar way and an ongoing study aims to test their environmental impacts empirically.

The paper presents the X_{mul} -Model designed for future ACC systems with the key capability of detecting multiple leaders. The feasibility of such design is affected by the road geometry, for example, it would be easier for both human drivers and vehicle sensors to detect more leaders on curves and grades. The detection also depends on the number and location of the cameras installed on the

vehicle. We conjecture that detection capability can be improved if cameras are installed at a higher position, such as the top of the vehicle where the lidar is usually placed. Hence, further empirical tests are needed to validate whether *Xmul*-Model is practically feasible and worth investing in market ACCs.

CRedit authorship contribution statement

Hao Zhou: Methodology, Experiment, Conceptualization, Visualization, Writing – original draft, Revision. **Anye Zhou:** Methodology, Writing – original draft, Revision. **Tienan Li:** Methodology, Experiment, Writing – original draft, Revision. **Danjue Chen:** Funding acquisition, Conceptualization, Revision. **Srinivas Peeta:** Funding acquisition, Revision. **Jorge Laval:** conceptualization, Methodology, Writing – original draft, Revision.

Acknowledgment

This study was supported by National Science Foundation (NSF) Cyber Physical System (CPS) grant #1932451 and #1826162.

References

- Bageshwar, V.L., Garrard, W.L., Rajamani, R., 2004. Model predictive control of transitional maneuvers for adaptive cruise control vehicles. *IEEE Trans. Veh. Technol.* 53 (5), 1573–1585.
- Brackstone, M., McDonald, M., 1999. Car-following: a historical review. *Transp. Res. F Traffic Psychol. Behav.* 2 (4), 181–196.
- Bu, F., Tan, H.-S., Huang, J., 2010. Design and field testing of a cooperative adaptive cruise control system. In: *Proceedings of the 2010 American Control Conference*. IEEE, pp. 4616–4621.
- Chen, D., Laval, J., Zheng, Z., Ahn, S., 2012. A behavioral car-following model that captures traffic oscillations. *Transp. Res. B* 46 (6), 744–761. <http://dx.doi.org/10.1016/j.trb.2012.01.009>.
- Chien, C.-C., Ioannou, P., 1992. Automatic vehicle-following. In: *1992 American Control Conference*. IEEE, pp. 1748–1752.
- Comma.ai, 2020. Comma.ai two setup. URL: <https://comma.ai/setup>.
- Comma.ai, 2021. Comma.ai – introducing openpilot. URL: <https://comma.ai/>.
- Corona, D., Lazar, M., De Schutter, B., Heemels, M., 2006. A hybrid MPC approach to the design of a smart adaptive cruise controller. In: *2006 IEEE Conference on Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control*. IEEE, pp. 231–236.
- Diehl, M., 0000. Toolkit for automatic control and dynamic optimization. URL: <https://acado.github.io/>.
- Ettinger, S., Cheng, S., Caine, B., Liu, C., Zhao, H., Pradhan, S., Chai, Y., Sapp, B., Qi, C.R., Zhou, Y., et al., 2021. Large scale interactive motion forecasting for autonomous driving: The waymo open motion dataset. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 9710–9719.
- Fancher, P., Bareket, Z., Bogard, S., MacAdam, C., Ervin, R., 1997. Tests Characterizing Performance of an Adaptive Cruise Control System. Technical Report, SAE Technical Paper.
- Gerrit Naus, 2008. Explicit MPC Design and Performance Evaluation of an ACC Stop and Go. pp. 224–229.
- Gong, S., Du, L., 2018. Cooperative platoon control for a mixed traffic flow including human drive vehicles and connected and autonomous vehicles. *Transp. Res. B* 116, 25–61. <http://dx.doi.org/10.1016/j.trb.2018.07.005>.
- Gunter, G., Gloudemans, D., Stern, R.E., McQuade, S., Bhadani, R., Bunting, M., Delle Monache, M.L., Lysecky, R., Seibold, B., Sprinkle, J., et al., 2020. Are commercially implemented adaptive cruise control systems string stable? *IEEE Trans. Intell. Transp. Syst.*
- Gunter, G., Janssen, C., Barbour, W., Stern, R.E., Work, D.B., 2019. Model-based string stability of adaptive cruise control systems using field data. *IEEE Trans. Intell. Veh.* 5 (1), 90–99.
- Ioannou, P., Xu, Z., Eckert, S., Clemons, D., Sieja, T., 1993. Intelligent cruise control: theory and experiment. In: *Proceedings of 32nd IEEE Conference on Decision and Control*. IEEE, pp. 1885–1890.
- Jin, I.G., Orosz, G., 2016. Optimal control of connected vehicle systems with communication delay and driver reaction time. *IEEE Trans. Intell. Transp. Syst.* 18 (8), 2056–2070.
- Jurgen, R.K., 2006. Adaptive Cruise Control. Technical Report, SAE Technical Paper.
- Kim, S., Coifman, B., 2013. Driver relaxation impacts on bottleneck activation, capacity, and the fundamental relationship. *Transp. Res. C* 36, 564–580.
- Laval, J.A., Daganzo, C.F., 2006. Lane-changing in traffic streams. *Transp. Res. B* 40 (3), 251–264.
- Laval, J.A., Leclercq, L., 2008. Microscopic modeling of the relaxation phenomenon using a macroscopic lane-changing model. *Transp. Res. B* 42 (6), 511–522.
- Laval, J.A., Toth, C.S., Zhou, Y., 2014. A parsimonious model for the formation of oscillations in car-following models. *Transp. Res. B* 70, 228–238.
- Li, T., Chen, D., Zhou, H., Laval, J., Xie, Y., 2021a. Car-following behavior characteristics of adaptive cruise control vehicles based on empirical experiments. *Transp. Res. B* 147, 67–91. <http://dx.doi.org/10.1016/j.trb.2021.03.003>.
- Li, T., Chen, D., Zhou, H., Laval, J., Xie, Y., 2021b. On the fundamental diagrams of commercial adaptive cruise control: Worldwide experimental evidence. *arXiv preprint arXiv:2105.05380*.
- Li, S., Li, K., Rajamani, R., Wang, J., 2010. Model predictive multi-objective vehicular adaptive cruise control. *IEEE Trans. Control Syst. Technol.* 19 (3), 556–566.
- Liu, H., Shladover, S.E., Lu, X.-Y., Kan, X., 2021. Freeway vehicle fuel efficiency improvement via cooperative adaptive cruise control. *J. Intell. Transp. Syst.* 25 (6), 574–586.
- Luo, L.H., Liu, H., Li, P., Wang, H., 2010. Model predictive control for adaptive cruise control with multi-objectives: Comfort, fuel-economy, safety and car-following. *J. Zhejiang Univ. Sci. A* 11 (3), 191–201. <http://dx.doi.org/10.1631/jzus.A0900374>.
- Lyft, 2021. Lyft level 5 prediction challenge 2021. <https://level-5.global/data/prediction/>.
- Magdici, S., Althoff, M., 2017. Adaptive cruise control with safety guarantees for autonomous vehicles. *IFAC-PapersOnLine* 50 (1), 5774–5781.
- Makridis, M., Mattas, K., Anesiadou, A., Ciuffo, B., 2021. OpenACC. An open database of car-following experiments to study the properties of commercial ACC systems. *Transp. Res. C* 125, 103047.
- Makridis, M., Mattas, K., Ciuffo, B., Re, F., Kriston, A., Minarini, F., Rognelund, G., 2020. Empirical study on the properties of adaptive cruise control systems and their impact on traffic flow and string stability. *Transp. Res. Rec.* 2674 (4), 471–484.
- Martinez, J.-J., Canudas-de Wit, C., 2007. A safe longitudinal control for adaptive cruise control and stop-and-go scenarios. *IEEE Trans. Control Syst. Technol.* 15 (2), 246–258.
- Milanés, V., Shladover, S.E., 2016. Handling cut-in vehicles in strings of cooperative adaptive cruise control vehicles. *J. Intell. Transp. Syst.* 20 (2), 178–191.
- Musa, A., Picipelli, M., Spano, M., Tufano, F., De Nola, F., Di Blasio, G., Gimelli, A., Misul, D.A., Toscano, G., 2021. A review of model predictive controls applied to advanced driver-assistance systems. *Energies* 14 (23), 7974.

- Naus, G., Ploeg, J., Van de Molengraft, M., Heemels, W., Steinbuch, M., 2010. Design and implementation of parameterized adaptive cruise control: An explicit model predictive control approach. *Control Eng. Pract.* 18 (8), 882–892.
- Naus, G., Ploeg, J., Van De Molengraft, R., Steinbuch, M., 2008. Explicit MPC design and performance-based tuning of an adaptive cruise control stop-&-go. In: 2008 IEEE Intelligent Vehicles Symposium. IEEE, pp. 434–439.
- Seiler, P., Song, B., Hedrick, J.K., 1998. Development of a collision avoidance system. *SAE Trans.* 1334–1340.
- Sheikholeslam, S., Desoer, C.A., 1993. Longitudinal control of a platoon of vehicles with no communication of lead vehicle information: A system level study. *IEEE Trans. Veh. Technol.* 42 (4), 546–554.
- Shi, X., Li, X., 2021. Empirical study on car-following characteristics of commercial automated vehicles with different headway settings. *Transp. Res. C* 128, 103134.
- Stanger, T., del Re, L., 2013. A model predictive cooperative adaptive cruise control approach. In: 2013 American Control Conference. IEEE, pp. 1374–1379.
- Treiber, M., Kesting, A., Helbing, D., 2007. Influence of reaction times and anticipation on stability of vehicular traffic flow. *Transp. Res. Rec.* 1999 (1), 23–29.
- Vajedi, M., Azad, N.L., 2015. Ecological adaptive cruise controller for plug-in hybrid electric vehicles using nonlinear model predictive control. *IEEE Trans. Intell. Transp. Syst.* 17 (1), 113–122.
- Waymo, 2022. Waymo open dataset challenge 2022. https://waymo.com/intl/en_us/dataset-download-terms/.
- Willis, M., 1999. Proportional-Integral-Derivative Control. Dept. of Chemical and Process Engineering University of Newcastle.
- Xiao, L., Gao, F., 2011. Practical string stability of platoon of adaptive cruise control vehicles. *IEEE Trans. Intell. Transp. Syst.* 12 (4), 1184–1194.
- Zhang, Y., Sun, H., Zhou, J., Pan, J., Hu, J., Miao, J., 2020. Optimal vehicle path planning using quadratic optimization for baidu apollo open platform. In: 2020 IEEE Intelligent Vehicles Symposium. IV, IEEE, pp. 978–984.
- Zhao, R., Wong, P., Xie, Z., Zhao, J., 2017. Real-time weighted multi-objective model predictive controller for adaptive cruise control systems. *Int. J. Automat. Technol.* 18 (2), 279–292.
- Zhou, H., 2021. Takeaways from tesla AI DAY event for autopilot's motion planning methods. <https://howardchow92.medium.com/quick-takeaways-from-tesla-ai-day-event-for-av-smotion-planning-13baa2a395cd>.
- Zhou, A., Gong, S., Wang, C., Peeta, S., 2020. Smooth-switching control-based cooperative adaptive cruise control by considering dynamic information flow topology. *Transp. Res. Rec.* 2674, 444–458.
- Zhou, J., Peng, H., 2005. Range policy of adaptive cruise control vehicles for improved flow stability and string stability. *IEEE Trans. Intell. Transp. Syst.* 6 (2), 229–237.
- Zhou, Y., Wang, M., Ahn, S., 2019. Distributed model predictive control approach for cooperative car-following with guaranteed local and string stability. *Transp. Res. B* 128, 69–86. <http://dx.doi.org/10.1016/j.trb.2019.07.001>.
- Zhou, H., Zhou, A., Laval, J., Liu, Y., Peeta, S., 2022a. Incorporating driver relaxation into factory adaptive cruise control to reduce lane-change disruptions. *Transp. Res. Rec.* 03611981221085517.
- Zhou, H., Zhou, A., Li, T., Chen, D., Peeta, S., Laval, J.A., 2022b. Significance of low-level control to string stability under adaptive cruise control: Algorithms, theory and experiments. *Transp. Res. C* 140.