Chapter 8
Platoon Control Concepts

Alexander L. Gratzer, Alexander Schirrer, Sebastian Thormann, and Stefan Jakubek

Abstract Cooperative platoon control strategies utilise provided information from vehicle-to-everything (V2X) communication to reduce energy consumption and improve traffic flow and safety. In this chapter, a distributed control concept for cooperative platooning is developed that combines trajectory optimisation and local model-predictive control of each vehicle. The presented control architecture ensures collision safety by design, platoon efficiency and situational awareness with the option of exploiting V2X communication. The resulting platoon control performance is tested and validated in a realistic setting by utilising a co-simulation-based validation framework with detailed vehicle dynamics.

Keywords Distributed model predictive control (DMPC) · Platooning · Optimal control · String stability

8.1 Introduction

In order to enable high-performance, efficient and safe platooning control concepts, global properties such as surrounding traffic, infrastructure, platoon dynamics, road properties and route must be appropriately considered in the planning and optimisation of platoon trajectories. For the effective realisation of these movement patterns in vehicle control, the distributed or locally acting control on the individual vehicle level must be combined with the essential information from the broader, global context in a suitably prepared form. Cooperative platoon control strategies use provided information from V2X communication to reduce energy or fuel consumption, increase traffic flow and improve traffic safety. Thereby, local information and predictions can be shared with the entire platoon, thus improving the distributed control actions' effectiveness.

A. L. Gratzer (✉) · A. Schirrer · S. Thormann · S. Jakubek
Research Unit of Control and Process Automation, Institute of Mechanics & Mechatronics, TU Wien, Vienna, Austria
e-mail: alexander.gratzer@tuwien.ac.at

A. Schirrer et al. (eds.), *Energy-Efficient and Semi-automated Truck Platooning*,
Lecture Notes in Intelligent Transportation and Infrastructure,
https://doi.org/10.1007/978-3-030-88682-0_8

8.2 Methodology Overview

A holistic control concept for cooperative platooning, illustrated in Fig. 8.1, was developed within Connecting Austria. Here, the *platoon coordinator* issues recommendations for actions to the platooning vehicles based on manoeuvre-specific trajectory planning, which is described in Sect. 8.4. The ego vehicle is locally controlled by model-predictive control (MPC). This MPC is specifically formulated so that a safe stop is always possible. Available information from the platoon coordinator as well as communicated predictions and agreements from the preceding and the following platooning vehicle are exploited for improved efficiency. The concept remains highly scalable due to its distributed control structure.

Each platooning vehicle implements a safety-extended local model-predictive controller wherein two optimisation problems are formulated: the *tracking problem* and the *fail-safe problem*. While the tracking problem aims at the tracking control of a reference trajectory, the fail-safe problem guarantees the possibility of a safe stop at any time. Due to this formulation, the platooning vehicles are always located in separated and thus safe position areas as illustrated in Fig. 8.1 top middle. The local *safeMPC* also implements two innovative methods for cooperative platooning and is explained in more detail in Sect. 8.5. First, a strategy is used to reduce the safely realisable distance between platooning vehicles by means of guaranteed temporarily limited brake actions; see Sect. 8.5.3. Second, an event-triggered communication scheme is used wherein the predicted ego trajectories are transmitted when necessary; see Sect. 8.5.4. Based on these methods, the platooning operation on slippery roads is considered and a special implementation variant (*explicit MPC*) is realised, which minimises the computational effort for real-time operation [10], see Sect. 8.5.5.

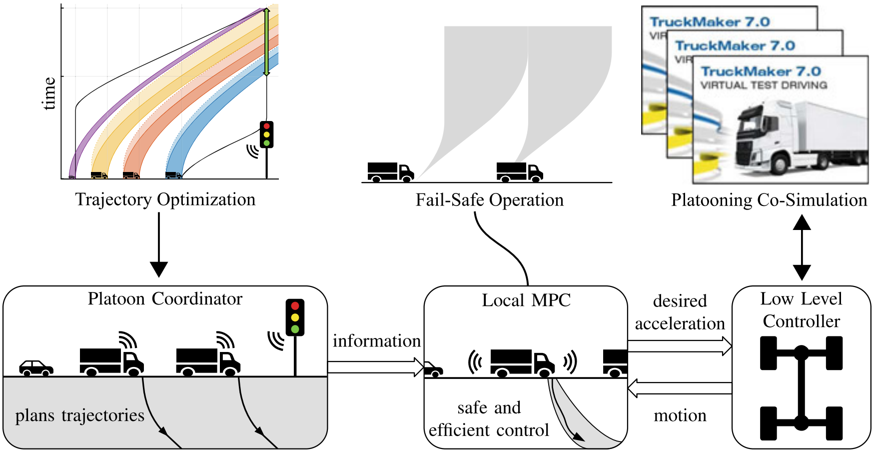

Fig. 8.1 Holistic control concept for cooperative platooning, ©2020 IEEE extended based on [13] with permission

Results for selected use cases are utilised for validation of the control concepts in Sect. 8.5. They are created by co-simulating detailed vehicle dynamics via the simulation software IPG Truckmaker® [6] whereby the control actions of the local MPCs are computed via MATLAB® [11]. A more detailed description of the co-simulation architecture is given in Sect. 8.3.

8.3 Co-simulation-Based Validation

For validation of the developed control concept, a co-simulation of the control algorithms with detailed vehicle simulations has been designed: with multiple instances of the simulation software IPG Truckmaker®, the vehicle dynamics are simulated in detail, and thus, a realistic test environment is achieved. Truckmaker® simulates multi-body dynamics enhanced with gear box, clutch, engine, and tyre models for each individual truck. The control law computation, communication, and synchronisation are realised via a central MATLAB® session in this co-simulation set-up. The complex nonlinear truck system dynamics considerably deviate from the idealised control design models, especially the drive train dynamics as well as the dynamic response to desired acceleration inputs. The implementation uses a central MATLAB® session which coordinates the individual Truckmaker® vehicle simulation instances and guarantees real-time capability for several trucks; see Fig. 8.2. Each vehicle instance is connected to a MATLAB®/Simulink®-worker, who also communicates with the platoon coordinator. The MPC problems are formulated and solved by parallel computing using the YALMIP toolbox [8] and Gurobi® [1] as optimisation problem solver. The simulation can also be connected to a force-feedback steering wheel, which allows to experience the platoon behaviour in traffic in a human-in-the-loop fashion. This way, new functionality of the platoon controller can be tested and evaluated effectively.

The functionalities of the platoon control concepts described hereafter are implemented and tested in the described co-simulation framework and driving simulator solution.

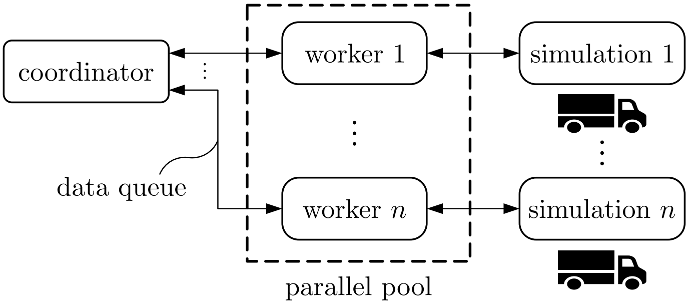

Fig. 8.2 Co-simulation architecture used for validating selected use cases and features

8.3.1 String Stability Considerations

String stability in a platoon is a property of platoon dynamics, expressing whether disturbances injected at the front of the platoon grow or decrease as they are propagated towards the platoon tail. It is especially important if platoons with short inter-vehicle distances and/or many vehicles should be built. Only string-stable platoon dynamics allows well-performing long platoons to be formed. String stability can be characterised by various approaches as surveyed in [4]. In particular, one group of string stability criteria is based on the $\mathcal{H}_\infty$ system norm for the transfer from the predecessor's velocity v_{pre} to the resulting ego vehicle's velocity v_{ego}, formally given by

$$\|G_v(z)\|_\infty \leq 1 \Leftrightarrow \text{predecessor-follower-string stable}\,, \tag{8.1}$$

in which $G_v(z)$ denotes the string stability transfer function from v_{pre} to v_{ego} in the case of linear discrete-time system dynamics.

For complex, nonlinear system dynamics, the $\mathcal{H}_\infty$-string stability criterion (where (8.1) represents the linear-dynamic case) is generally defined as

$$\sup_{\|v_{\mathrm{pre}}(t)\|_2 \neq 0} \frac{\|v_{\mathrm{ego}}(t)\|_2}{\|v_{\mathrm{pre}}(t)\|_2} \leq 1 \Leftrightarrow \text{predecessor-follower-string stable}\,. \tag{8.2}$$

Therein, $\|\cdot(t)\|_2$ denotes the signal energy norm of the corresponding time signal. Unfortunately, this criterion cannot directly be determined in practice because all possible trajectories would have to be evaluated.

However, focusing our attention on a particular manoeuvre with finite-energy error velocity signals (both v_{pre} and v_{ego} approach the same reference velocity v_{ref} as $t \to \infty$), the manoeuvre's signals can be evaluated and a lower bound for the $\mathcal{H}_\infty$ gain can be obtained:

$$\frac{\|v_{\mathrm{ego}}(t) - v_{\mathrm{ref}}\|_2}{\|v_{\mathrm{pre}}(t) - v_{\mathrm{ref}}\|_2} > 1 \Rightarrow \textbf{not}\ \text{predecessor-follower-string stable}\,. \tag{8.3}$$

Analogously, energy ratios in accelerations or distance errors can be formulated and utilised to interpret simulation results for string stability properties.

In [7], a set of tools for the assessment of string stability properties of platooning controllers is presented. Both linear and nonlinear (simulation-based) system dynamics can be used as an evaluation basis for the influence on congestion formation in real traffic. Both analytic and empirical settings are proposed therein.

Figure 8.3 illustrates string-stable (*left*) versus not string-stable (*right*) platoon dynamics in a transient manoeuvre. The platoon initially drives at $v = 80\,\mathrm{km/h}^{-1}$ with an inter-vehicle distance of $d_\infty = 18\,\mathrm{m}$ respective $d_\infty = 15\,\mathrm{m}$, and an external non-platoon vehicle induces a brake pulse. In the string-stable case, the resulting disturbances decay as they are propagated towards the platoon tail vehicle, whereas in the other configuration, the disturbance amplitudes significantly increase along the

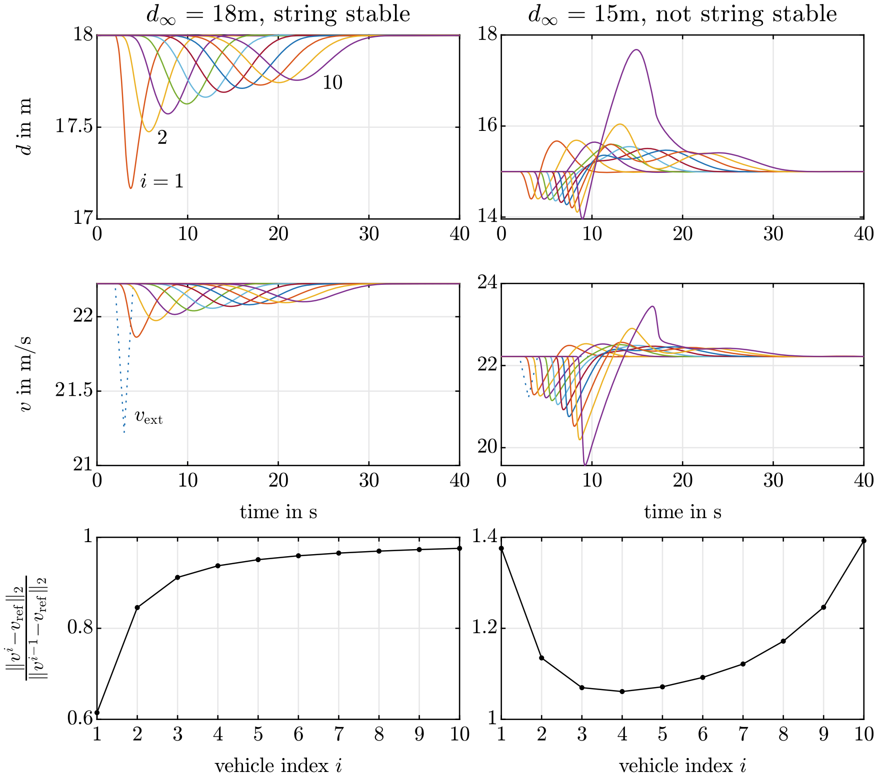

Fig. 8.3 Inter-vehicle distance d, velocity v and evaluated Eq. (8.3) for initial inter-vehicle distance $d_\infty = 18\text{m}$ (*left*, string stable) and $d_\infty = 15\text{m}$ (*right*, not string stable)

platoon. Note that each vehicle nevertheless shows a stable response. Figure 8.3 also shows the empirical string stability measure (8.3) for the cases when string stability is present (*left*) or not (*right*), respectively.

Further analysis regarding string stability of cooperative vehicle platoons with consideration of collision safety constraints, as implemented in the proposed *safeMPC* concept, can be found in [5].

8.4 Trajectory Optimisation Methodology

Trajectory planning enhances the performance of the implemented distributed control scheme from a global perspective by incorporating information obtained from the infrastructure, for example speed limits, road conditions or green light timing information.

In order to enable group manoeuvres, such as a simultaneous start-up in front of a traffic light or an efficient transition to manual driving when approaching a hazardous location, a trajectory optimisation task is carried out by a *platoon coordinator* instance. This platoon coordinator is understood to be an entity that issues recommendations for actions to the platooning vehicles based on additional information obtained through V2X communication including positions and velocities of the controlled platooning members. These recommendations include position trajectories as well as desired velocities and desired minimal inter-vehicle distances. The optimised trajectories, which are planned in a way that safety and optimality objectives are achieved, are used as reference values for the distributed local MPCs, which aim to track these planned trajectories and also take into account the current traffic situation in order to calculate collision-safe control inputs.

In practice, the implementation of the platoon coordinator can be done, for example, at the platoon leader (first vehicle of the platoon), but also an implementation in a road-side unit (RSU) or a completely distributed implementation is possible.

8.4.1 Optimisation Problem Formulation

The trajectories are planned in such a way that application-specific safety and optimality goals are achieved. The underlying optimisation problem is generally of the form

$$\boldsymbol{U}^* = \arg\min_{\boldsymbol{U}} J(\boldsymbol{U}; \boldsymbol{\theta})$$

such that the restrictions

$$\boldsymbol{g}(\boldsymbol{U}; \boldsymbol{\theta}) \leq \boldsymbol{0}$$

are satisfied. Thus, the scalar cost function $J(\boldsymbol{U}; \boldsymbol{\theta})$ is to be minimised by an optimal choice of the decision variables $\boldsymbol{U}$ at fixed values of the parameters $\boldsymbol{\theta}$ while at the same time the constraints modelled as a vector-valued constraint function $\boldsymbol{g}(\boldsymbol{U}; \boldsymbol{\theta})$ must be obeyed. Thus, the optimal values of the decision variables $\boldsymbol{U}^*$ are exactly those which minimise the costs and simultaneously satisfy the given constraints. The trajectory optimisation sequentially solves quadratic programming problems with linear constraints. For the concrete applications in the project, the general form of the optimisation problem will be interpreted illustratively in the following. The decision variables $\boldsymbol{U}$ are the control inputs of the platooning vehicles, which are understood as acceleration values at a range of time steps covered by the optimisation horizon. The cost function $J(\boldsymbol{U}; \boldsymbol{\theta})$ is composed of application-specific terms, such as costs due to the control input values, the time losses during passing an intersection or the deviations from desired inter-vehicle distances. The prioritisation of these terms is achieved by weighting factors in the objective formulation of the cost function. The restrictions $\boldsymbol{g}(\boldsymbol{U}; \boldsymbol{\theta})$ take into account, among other things, simplified vehicle dynamics with limited acceleration (characteristic curve dependent on

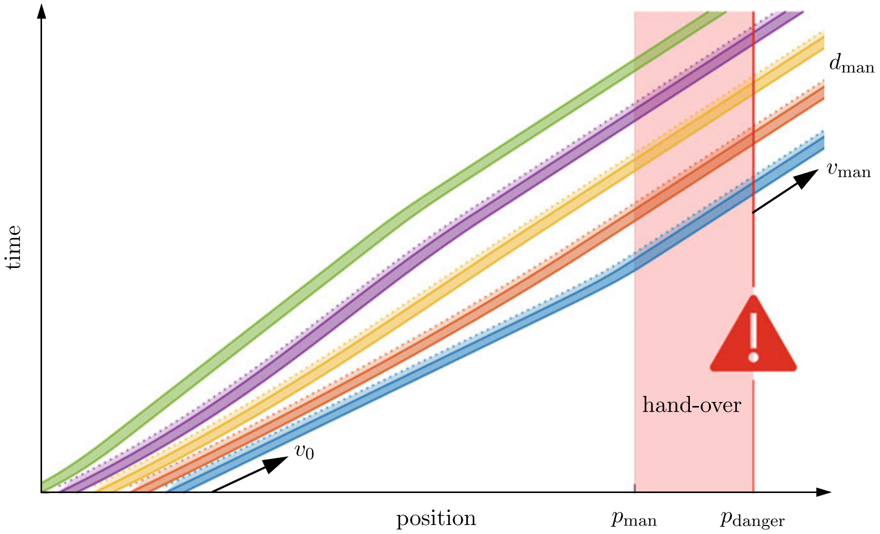

Fig. 8.4 Optimisation of a handover to the human driver in front of a hazardous location (construction site) at p_{danger} with demanded velocities v_{man} and distances d_{man} during the handover phase

speed) and minimum required distances in terms of space and time gaps. In order to effectively use the trajectory optimisation as a tool set, a formal interface definition was formulated and implemented, which simplifies the definition of the parameters $\boldsymbol{\theta}$. Also, semantic tests with respect to the given problem specification and the calculated results are carried out to allow safely automated parameterisation and Monte Carlo-type simulation studies.

Since the results of the trajectory planning task have only informative character, the optimisation problem presents a less time critical operation [12].

The two scenarios *use case 2: truck platoon approaching a hazardous location*, as illustrated in Fig. 1.2, and *use case 4: truck platoon crossing an intersection*, as depicted in Fig. 1.4, are considered in particular in the following Sects. 8.4.2 and 8.4.3, respectively.

8.4.2 Trajectory Optimisation for Approaching a Hazardous Location

Figure 8.4 shows an example for the case of a platoon disbanding and the subsequent handover to the human driver when approaching a *hazardous location*, such as a construction site with a reduced speed limit. In front of the construction site, at a distance of 600 m from the initial platoon location, the platoon should expand to an inter-vehicle distance of $d_{\mathrm{man}} = 50\,\mathrm{m}$, and a speed of $v_{\mathrm{man}} = 60\,\mathrm{km/h}^{-1}$ should be maintained during the handover phase. The vehicle lengths are 15 m. In order to allow a sufficient time span for the handover to the human driver, this required platoon state has to be realised already 5 s before reaching the construction site. Since the velocity during the handover phase is assumed to be kept constant at v_{man}, the handover for each vehicle begins at the position

$$p_{\mathrm{man}} = p_{\mathrm{danger}} - v_{\mathrm{man}}\, T_{\mathrm{man}}\,.$$

In this optimisation problem, an energy-efficient trajectory is sought which meets the required platoon state with high accuracy during the handover phase.

8.4.3 Trajectory Optimisation for Crossing an Intersection

Given the use case of a platoon crossing a traffic light-controlled intersection, it is assumed that the platoon coordinator receives accurate real-time information of the next green light phase via infrastructure-to-vehicle (I2V) communication. By carrying out a centralised but simplified trajectory optimisation where time and energy consumption are minimised, the platoon coordinator is able to plan and recommend efficient position trajectories as well as minimum inter-vehicle distances and velocities for crossing the intersection. This functionality represents an extension to Green Light Optimal Speed Advisory (GLOSA) features already available in autonomous driving use cases [3].

The trajectory optimisation problem set-up is illustrated in Fig. 8.5. Here, a so-called *optimal drive* is sought, which allows the platoon to pass the intersection with efficient control inputs as early as possible during the green light phase of the traffic light. In detail, two auxiliary problems are solved: the *maximum drive* of the platoon leader is the fastest possible manoeuvre at any time until the beginning of the green light phase, so that the traffic light is not run over when still red. The *minimum drive* of the platoon tail (last vehicle of the platoon) is the slowest possible manoeuvre at any time until the end of the green light phase, so that the traffic light is reached in the last moment of the green phase. The optimal drive is then solved using the results of the minimum and maximum drives as bounding constraints [12].

However, the boundary conditions of such traffic optimisations are only partly known. For example, if non-platooning vehicles are present and their future behaviour is uncertain, then continuous adaptation of these planned trajectories is necessary. Utilising event-triggered vehicle-to-vehicle (V2V) communication, the platoon is able to efficiently adapt its controlled motion to the sudden presence of a non-platooning vehicle and thus to realise situational awareness as will be described in Sect. 8.5.4.

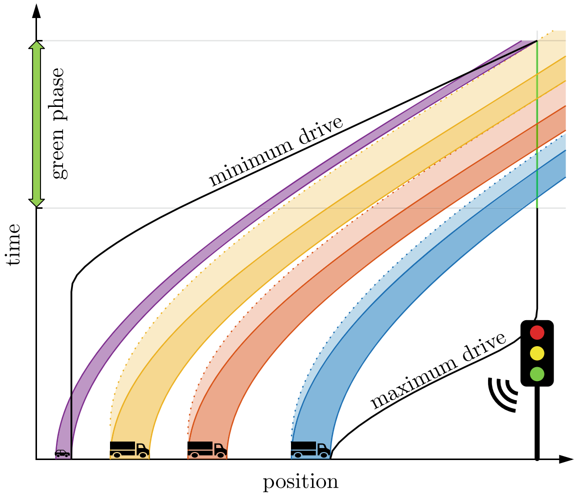

Fig. 8.5 Trajectory optimisation for the use case of a truck platoon crossing an intersection. The individual coloured areas represent the occupied space of each vehicle respective the demanded velocity-dependent time gaps

8.5 Distributed Model-Predictive Platoon Control

The basic concept of MPC is to use a dynamic model to predict the behaviour of the controlled system and recurrently optimise the control input in order to obtain an optimal predicted system response, as introduced in the textbook [9] and illustrated here in Fig. 8.6. The dynamic model used to calculate predictions, the *prediction model*, can be based on known physical laws, empirical data and/or expert knowledge. Usually, the predictions are considered only over a limited future time interval (up to the *prediction horizon*) to render the involved calculations feasible in real-time. The meaning of an optimal system behaviour has to be specified. For this purpose, the predicted system behaviour (predicted state and predicted input) is assessed from the current point in time to the prediction horizon. This assessment is implemented by means of a cost function which quantifies the attainment of specified control goals, such as accurate reference tracking, energy-efficient actuation or time optimality. Constraints of the controlled system, such as permitted inter-vehicle distances, or speed limits, can be directly incorporated. Eventually, an optimisation problem of the form (8.4.1)–(8.4.1) is solved recurrently to calculate the currently applied input. Thereby the value of the cost function is minimised while the given constraints are obeyed.

In *distributed MPC*, control actions are only based on a subset of the overall system information. Reasons for this may include unavailable information of distant subsystems, reduced communication effort and/or reduced computational complexity. Platoon control is a typical application domain for distributed MPC. For individual vehicles, local measurements are available. Additionally, in cooperative platoon control, neighbourhood information may be communicated. The distributed MPC problem is now characterised by control goals and/or constraints which involve non-

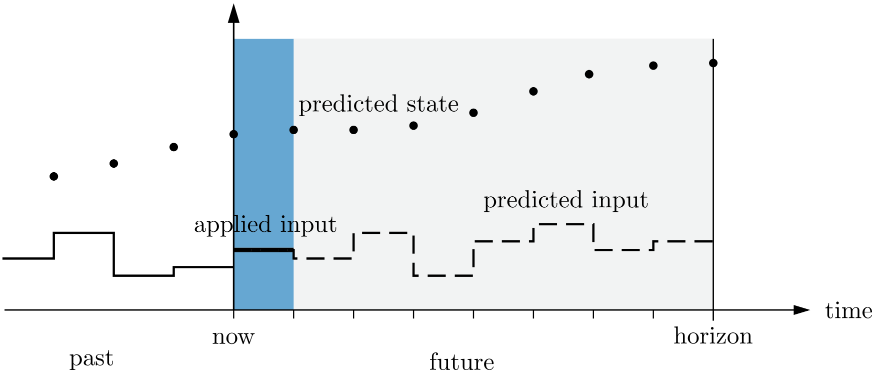

Fig. 8.6 Basic concept of model-predictive control

local system states, while the control actions are based on a local information scope and communicated information.

8.5.1 Safe-by-Design Local MPC Formulation

Each platoon vehicle needs a local controller to track its preceding vehicle safely and efficiently. The local platooning vehicle implements a model-predictive controller (MPC), which is formulated considering collision safety explicitly. These concepts have been developed in [13] and [10] and are summarised in the following.

The local MPC structure follows a safe-by-design paradigm and at the same time takes communicated information from the platoon coordinator as well as from the front vehicle into account to improve efficiency. Figure 8.7 illustrates the safety aspects of the MPC formulation. Two optimisation problems are formulated (Fig. 8.7, *left*), the *tracking problem* and the *fail-safe problem*, which are coupled over a time span ("tolerance time") T_{tol}. While the tracking problem aims to track a particular reference trajectory (determined by the desired velocity, predecessor predictions and corresponding gap policies), the fail-safe problem ensures the feasibility of a safe (i.e. collision-free) stop behind its predecessor at all times. Consequently, the platooning vehicles are always kept in separated and thus safe areas (Fig. 8.7, *right*).

In [13], two additional, innovative methods for cooperative platooning have been developed: first, a strategy was developed to reduce the safely realisable distance between platooning vehicles by committing to temporarily limited brake authority. Second, an event-triggered communication scheme for the efficient transmission of local MPC predictions was developed, which allows for an early response when a control intervention is necessary. Based on this fundamental control structure, in [10] the platooning operation on slippery roads was considered and a special implemen-

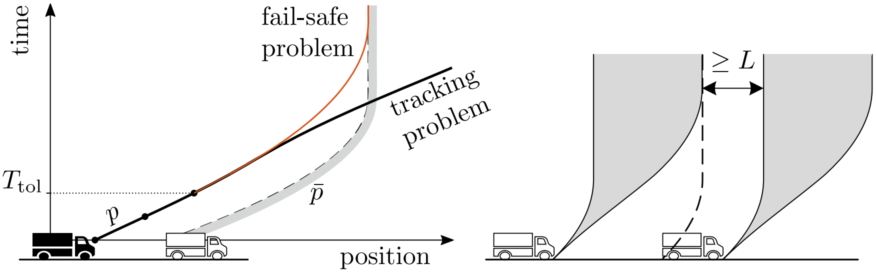

Fig. 8.7 Safety aspects of the MPC formulation, ©2020 IEEE adapted from [13] with permission: Keeping a fail-safe trajectory feasible at all times, the coupled tracking problem can be designed separately, and collision safety is guaranteed (*left*). Each vehicle can realise a safe set of trajectories which is disjoint from the other vehicles' trajectories (*right*).

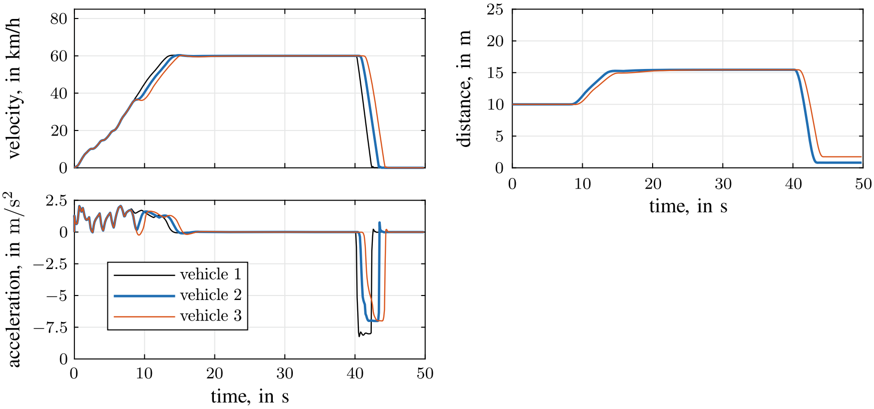

Fig. 8.8 Simulation scenario: emergency braking

tation variant (explicit MPC [2]) was realised, which minimises the computing effort for real-time operation.

8.5.2 Validation of Collision Safety via Co-simulation

The presented control concept meets realistic requirements regarding the robustness against model errors, which was validated by co-simulations using the commercial software IPG Truckmaker® for the simulation of detailed vehicle dynamics. Two simulation scenarios will now be explained by way of example. Figure 8.8 shows the simulation results of the controlled platoon during emergency braking of the platoon leader at time $t = 20\,\text{s}$. All platoon vehicles come to a safe (i.e. collision-free) standstill.

In addition to these co-simulation studies, a human-in-the-loop driving simulator based on the same co-simulation framework has been developed. It allows to gather first-hand experience of the platoon control features from the driver's perspective.

8.5.3 Safe Reduction of Inter-vehicle Distances

The safety constraint illustrated above is significantly affected by the bounded or estimated braking authority of the predecessor and the tolerance time. To safely reduce inter-vehicle distances, the preceding vehicle could commit to a reduced deceleration magnitude. This *hold-back* strategy for temporarily reduced braking authority is detailed in [13]. Using absolute time stamps of the expiration time of this commitment, the approach becomes robust against a loss of V2V communication.

Use case 2 has already been discussed in trajectory planning. In terms of use case 2, moreover, the hold-back feature is relevant: on a free highway, the predecessor (ultimately the leader vehicle of the platoon) can commit to significantly reduce deceleration magnitudes for relevant time spans safely, which, in turn, allows the platoon to realise tight inter-vehicle gaps safely. When the surrounding traffic situation changes, as for example in the approach to a hazardous location (Use Case 2), the hold-back phase safely expires, and the platoon automatically and safely opens up to larger, safe inter-vehicle gaps.

Figure 8.9 shows the platoon behaviour in case of this limited brake authority (hold-back strategy). With active hold-back (time ranges marked in grey), each platooning vehicle guarantees a defined, reduced deceleration magnitude bound until the expiration of a communicated time, which is regularly updated and thus postponed into the future. This allows the safe minimum distance between the platooning vehicles to be significantly reduced. If the hold-back strategy is now deactivated, for example by interrupting the communication (comm. lost in Fig. 8.9), the platoon will automatically trigger an always safe expansion of the platoon.

8.5.4 Situation-Aware Platoon Behaviour via V2V-Communication

This section outlines a corridor-based platoon communication strategy to realise situation awareness with parsimonious V2V communication [13]. This feature has direct application in use case 4 and provides a flexible, generic and efficient solution to the case when the platoon has to react appropriately to individual, non-platoon vehicles on the road.

We illustrate the proposed method at the case when an *individual, non-platooning vehicle is present in front of a traffic light* as shown in Fig. 8.10. Initially, the platooning vehicles track the planned trajectories (dashed lines), compare Sect. 8.4.3.

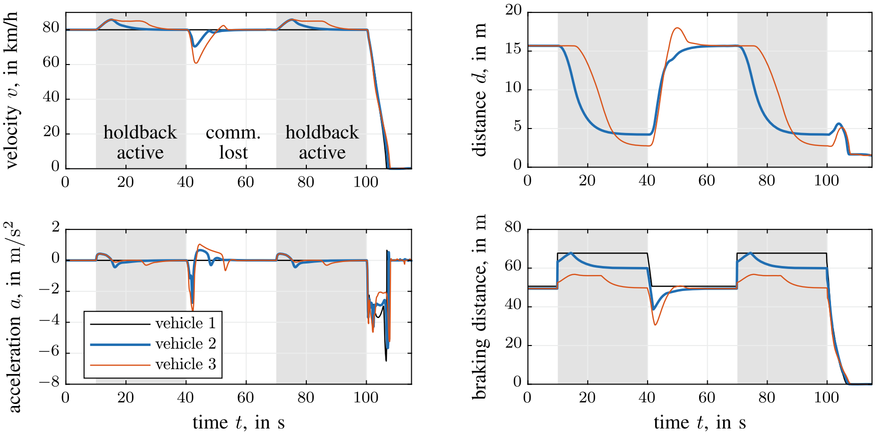

Fig. 8.9 Use case 2: truck platoon approaching a hazardous location (hold-back, communication loss at $t = 40$s, emergency brake at $t = 100$ s)

Since a non-platooning vehicle (dash-dotted line), waiting at standstill before the intersection, was not considered in the trajectory optimisation, the platoon adapts to this unpredicted traffic situation: the individual vehicle's motion is predicted based on current sensor data measured by the leader and considered in solving its predictive control problem. The leader communicates its own predicted trajectories to its followers if this data is sufficiently different to the last transmitted trajectories. As a result of these automated event-triggered prediction updates within the platoon, an efficient braking manoeuvre as well as a simultaneous start-up is realised automatically in this distributed control setting without centralised guidance. This case demonstrates robust situation awareness capabilities of the outlined control concept. The necessary communication bandwidth is kept low. Also, the communication of accurate predictive information within the platoon establishes string-stable behaviour with respect to the updated manoeuvre, as seen in the essentially congruent trajectories around $t = 15$ s.

8.5.5 *Consideration of Varying Road Conditions*

The ideas of the safety-extended distributed model-predictive platoon control concept shown in [13] have been utilised to design an explicit distributed platoon MPC for varying road slip conditions in [10]. The shown controller provides collision safety and realises platooning functions under the assumption that the current friction coefficient of the road is available or estimated. The explicit control law is a precomputed form of the MPC which can be constructed if the number of problem parameters (initial state, references, constraint parameters) is sufficiently low. Then, a lookup surface in this parameter space is constructed which represents the solution

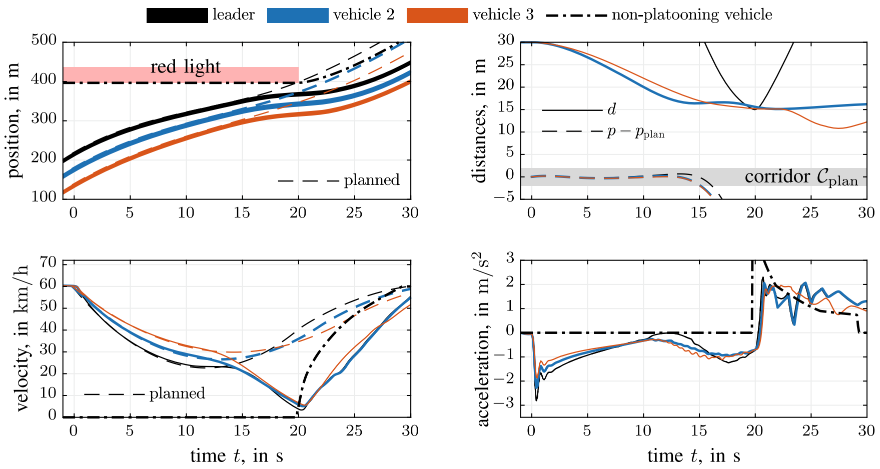

Fig. 8.10 Simulation results of the cooperative platooning strategy for the case of an present non-platooning vehicle in front of a traffic light. It can be seen that the platoon adapts to the unpredicted new situation in an optimal way by diverging from the planned trajectories for about 20 s

of the underlying optimisation problem directly. It was shown that the main features of the safe-by-design local MPC outlined in [13] can be simplified in a suitable way to obtain a tractable explicit MPC formulation (see [10]). To achieve this simplification, a non-uniform sampling of the optimisation horizon of the discrete-time MPC formulation has been considered, and the tracking of vehicle reference velocities and inter-vehicle distances has been parameterised in an efficient way through a corresponding preprocessing step. Also, the collision safety constraints have been formulated in a simplified and approximated fashion. The resulting control performance and collision safety have been tested in platoon co-simulation studies with detailed vehicle dynamics. Finally, the platoon behaviour on a dry road as well as on a slippery road has been validated in co-simulations.

8.6 Conclusion

A holistic predictive control concept has been developed that involves manoeuvre planning and optimisation, safe-by-design distributed model-predictive platoon control of the vehicles, and a detailed co-simulation validation framework.

Trajectory planning methods are utilised to generate efficient real-time trajectories of each platooning vehicle (position, velocity and acceleration over a defined time interval from the current time into the future). A general formulated based on optimal control has been formulated which allows to solve many relevant platoon manoeuvres and scenarios, such as the approach to a hazardous location, or the task of crossing an intersection efficiently and safely. Real-time information from the road infrastructure is incorporated, and relevant constraints such as safety distances and traffic rules are considered. The resulting trajectories serve as references for the distributed model-predictive platoon vehicle controllers which control the individual vehicles accordingly.

The proposed vehicle control architecture safely and efficiently controls the platoon vehicles, incorporates infrastructure information in real time into the control task and provides collision safety guarantees. Moreover, suitable measures are outlined which allow to exploit V2V communication between platooning vehicles to safely reduce inter-vehicle distances and improve platoon efficiency.

The validation framework is based on the co-simulation of all platoon vehicles, each simulated with detailed vehicle dynamics by a physically realistic, industry-grade vehicle simulator and equipped with the novel distributed MPC and communication functions. The range of platoon control functionalities is further extended by parameterising the controller with a real-time road friction estimate, so that safety is still ensured if road conditions deteriorate.

The outlined collection of methods, tools and control functionality provides a versatile and comprehensive basis for intelligent, situation-aware and safe platooning, directly applicable in realistic platooning scenarios.

References

1. Manual Gurobi Optimizer Reference (2018) Version 8:1
2. Bemporad A, Morari M, Dua V, Pistikopoulos EN (2002) The explicit linear quadratic regulator for constrained systems. Automatica 38(1):3–20
3. CAR 2 CAR Communication Consortium (2019) Guidance for day 2 and beyond roadmap
4. Feng S, Zhang Y, Eben Li S, Cao Z, Liu HX, Li L (2019) Definitions and analysis methods. String stability for vehicular platoon control. Annu Rev Control 47:81–97
5. Gratzer A, Thormann S, Schirrer A, Jakubek S (2021) String stability of cooperative vehicle platoons with consideration of collision safety constraints. submitted, under review
6. IPG Automotive GmbH. CarMaker release notes 7.0.3
7. Kalteis C, Thormann S, Schirrer A, Jakubek S (2020) Efficient methods to assess linear and non-linear automotive platoon control stability and performance. In: XI international conference on structural dynamics:748–757
8. Löfberg J (2004) YALMIP: a toolbox for modeling and optimization in MATLAB. In: Proceedings of the IEEE international symposium on computer-aided control system design, 284–289
9. Rawlings JB, Mayne DQ, Diehl M (2017) Model predictive control: theory, computation, and design. Nob Hill Publishing Madison, WI, 2 edition
10. Schirrer A, Hanis T, Klauco M, Thormann S, Hromcik M, Jakubek S (2020) Safety-extended explicit MPC for autonomous truck platooning on varying road conditions. IFAC world congress
11. The MathWorks Inc. MATLAB R2018b, 2018
12. Thormann S (2020) Cooperative platooning—development and co-simulation-based validation of distributed model predictive control methods for safe and efficient cooperative platooning. Master's thesis, TU Wien
13. Thormann S, Schirrer A, Jakubek S (2020) Safe and efficient cooperative platooning. IEEE Trans Intell Transp Sys

Alexander L. Gratzer has been a Project Assistant with the Institute of Mechanics and Mechatronics, TU Wien, since 2019 and currently works toward the Ph.D. degree. Alexander studied mechanical engineering and is involved in international research projects and teaching of graduate-level lectures. His research interests include simulation, optimisation, and control of complex industrial systems.

Alexander Schirrer has been a Postdoctoral Researcher and Teacher of graduatelevel lectures with the Institute for Mechanics and Mechatronics, TU Wien, since 2011. His research interests include modelling, simulation, optimisation, and control of complex and distributed-parameter systems.

Sebastian Thormann has been a Project Assistant with the Automation and Control Institute, TU Wien, since 2021 and currently works toward the Ph.D. degree. He studied mechanical engineering and was a student member of the project team with the Institute for Mechanics and Mechatronics, TU Wien, from 2017 to 2020. His research interests include modelling, simulation, optimisation, and control of complex and distributed-parameter systems.

Stefan Jakubek is Professor and Head of the Institute of Mechanics and Mechatronics, TU Wien. From 2007 to 2009, he was the Head of Development for Hybrid Powertrain Calibration and Battery Testing Technology with AVL List GmbH, Graz, Austria. His research interests include fault diagnosis and system identification.