mathematics

MDPI

Article

Exploring the Role of Sampling Time in String Stabilization for Platooning: An Experimental Case Study

Felipe I. Villenas [1], Francisco J. Vargas [1,*] and Andrés A. Peters [2]

[1] Departamento de Electrónica, Universidad Técnica Federico Santa María, Avda. España 1680, Valparaíso 2390123, Chile; felipe.villenas@sansano.usm.cl
[2] Faculty of Engineering and Sciences, Universidad Adolfo Ibáñez, Santiago 7941169, Chile; andres.peters@uai.cl
* Correspondence: francisco.vargasp@usm.cl

Abstract: In this article, we investigate the behavior of vehicle platoons operating in a predecessor-following configuration, implemented through sampled-data control systems. Our primary focus is to examine the potential influence of the sampling time on the string stability of the platoon. To address this, we begin by designing a string-stable platoon in continuous time. Subsequently, we consider the controller discretization process and proceed to simulate and implement the designed control strategy on an experimental platform at a scaled-down level. Through experimental testing and some theoretical results, we analyze the effects of different sampling times on the string stability performance of the platoon. We observe that an inappropriate selection of the sampling time can lead to a degradation in string stability within the platoon, making the choice of the sampling time crucial in maintaining the desired string stability properties. These findings highlight the importance of carefully considering the sampling time in the implementation of control systems for platooning applications.

Keywords: vehicle platoon; string stability; discrete time; sampled-data systems

MSC: 93-05

Citation: Villenas, F.I.; Vargas, F.J.; Peters, A.A. Exploring the Role of Sampling Time in String Stabilization for Platooning: An Experimental Case Study. *Mathematics* **2022**, *11*, 2923. https://doi.org/10.3390/math11132923

Academic Editors: Vlad Muresan and Mihail Ioan Abrudean

Received: 24 May 2023
Revised: 22 June 2023
Accepted: 27 June 2023
Published: 29 June 2023

1. Introduction

In recent years, the field of intelligent transportation systems (ITS) has grown significantly thanks to advancements in wireless communication, data processing, control systems, and artificial intelligence algorithms, among other areas [1–3]. One specific area of ITS research is cooperative adaptive cruise control systems, which enable vehicles to navigate at a cruise speed that adapts to the speed of preceding vehicles. Using information sent wirelessly by other vehicles can improve the goal of maintaining a desired inter-vehicle distance between them [4]. A well-designed ITS system with these features can allow a platoon of vehicles to navigate cooperatively and efficiently, resulting in reduced fuel usage, pollutant emissions, travel times, and improved transport safety [5]. By utilizing these technologies, we can create a smarter transportation infrastructure that meets the needs of modern society.

Naturally, in platooning applications, each vehicle's control system must achieve internal stability, the desired cruise speed, and maintain the desired distance between vehicles. However, there is an additional requirement when designing these control systems, which relates to how transient errors caused by external stimuli or disturbances propagate to the vehicles further back in the platoon. If these errors are amplified along the chain of vehicles, the platoon will exhibit erratic and dangerous behavior, which will worsen if more vehicles are added to the platoon. In this case, the platoon is said to be string-unstable. On the other hand, if the system is string-stable, the errors will be reduced or remain bounded as they propagate along a platoon, regardless of its length, reducing the danger in navigation and allowing more vehicles to be added without compromising performance [6]. The

importance of string stability in platooning has motivated researchers to obtain conditions on the design parameters of controllers that guarantee it. It has been found that these conditions depend on many factors, such as the topology of the communication between vehicles, the type of information that is sent, the spacing policy between vehicles, the quality of the communication channels, and the dynamic characteristics of the vehicles, among others [6–8].

Initially, most of the results related to string stability in platooning have been derived for continuous-time frameworks (see, for instance, [6,7,9,10]). However, in recent years, there has been a growing interest in studying platoons within a discrete-time setting. This is because a crucial aspect of many control systems is their practical implementation using computer-based control and, consequently, the use of a discrete-time framework. Additionally, the implementation of digital wireless communications among vehicles in the context of platooning further justifies the use of discrete-time analysis. As a result, a variety of digital technologies have gained attention in the field of platooning, with the ultimate aim of transforming it into a practical reality [11–13].

When implementing sampled-data control in platooning, it is crucial to carefully evaluate the impact of the sampling process on the system's performance. It is widely acknowledged that selecting an inappropriate sampling time in a feedback loop can result in the loss of internal stability [14]. However, the effect of the sampling time on string stabilization in platooning has received limited attention from researchers thus far. Instead, most studies in this area have primarily focused on addressing the challenges posed by unreliable communication links and utilizing discrete-time control techniques for platooning analysis. For instance, Ref. [15] employs Model Predictive Control (MPC) to handle switching communication topologies in a platooning setting, providing stability and string stability results. In [16], a sampled-data-based MPC-like control method is proposed to achieve string stability in platoons subject to disturbances and saturation constraints. Similarly, Ref. [17] considers platooning with non-ideal communications using LQR control, allowing for dealing with transmissions of data that are samples of the real acceleration of vehicles. In [18], platooning control using sampled positions of vehicles was investigated, focusing on the input-to-state stability of the multi-agent system instead of string stability. Various aspects of platooning under unreliable communication links, such as random data loss, have been studied in the literature, including a discrete-time approach. For instance, in this context, researchers have explored mean square stabilization [19] and string stability [20–22], the employment of data-loss compensation strategies [23], and also estimation techniques to deal with data loss [24,25]. Platooning with communication links subject to malicious attacks is studied in, for instance, [26,27], which naturally considers sampled systems. The authors of [28] investigate event-triggered platooning control for random communication topologies and different spacing policies but do not provide explicit string stability results. In [29], the authors primarily focus on sampled-data control of vehicular platoons, specifically considering undirected communication topology and analog fading networking, while omitting detailed analysis of string stability. Earlier works addressing sampling and platooning can be found in [30–32]. These works provide several insights into the relationships between sampling and important aspects of vehicular platooning. However, most of the previous works do not thoroughly consider the effect that varying the sampling time has on the scalability properties of a platoon, particularly its string stability.

On the other hand, experimental validation in vehicular platooning is a challenging but important task. In both continuous and discrete-time literature results, some works obtained analytical expressions for the string stability conditions and sometimes complemented them with simulations to verify their consistency [9,10]. In other cases, the analysis has been reduced to numerical studies only [20,23,33]. In very few cases, there is experimental validation of the results, mainly due to cost limitations associated with implementing the techniques on real vehicles, which limits the size of the platoon [34]. However, some studies use scale experimental platforms to test the proposed algorithms [24,35–38]. In the realm of platooning research, most analytical findings have been derived within the

continuous-time framework despite the practical implementation necessitating the use of sampled-data control systems. Consequently, the selection of an appropriate sampling time becomes a fundamental yet critical task. The lack of results concerning the interplay between the sampling time and string stability further complicates this task.

The previous literature review reveals a noticeable gap in the literature relating sampling time and string stability. Motivated by this, in this work, we aim to investigate whether the choice of sample time can impact the string stability property in a real implementation. To achieve this, we first design a string-stable platoon using a continuous-time setting. Next, we adapt the continuous-time design to a sample-data feedback control framework and simulate the behavior of the platoon for different sampling times. We chose sampling times that guarantee internal stability for the feedback system, enabling us to assess the string stability property separately from any internal instability concerns. We also provide theoretical results that motivate the design of platooning experiments. Finally, we implement this sample-data control scheme on the scaled-down experimental platform, PL-TOON [39]. Our simulation results and experimental findings provide empirical evidence that the selection of sample time can indeed influence the string stability property of a platoon. These results highlight the need for a deeper analysis of the relationship between sampling time and string stabilization, incorporating both theoretical and practical analyses. Our results are an initial step towards a derivation of necessary and sufficient conditions on the sample time for string stability. This is necessary to ensure the proper functioning of platoons in practical applications.

The rest of the article is structured as follows. Section 2 introduces the setup and the problem of interest. Section 3 describes the materials and methods used. Section 4 presents and discusses the simulation and experimental results. Finally, Section 5 provides the conclusions.

2. Problem Formulation

2.1. Preliminaries and Notation

Throughout the manuscript, we consider most of the standard systems and control literature notation. For a signal $x(t)$ of the continuous time variable $t \in \mathbb{R}$, we denote $X(s)$, with $s \in \mathbb{C}$ as its Laplace transform. If the transfer function of a single-input single-output LTI continuous time system is given by $G(s)$, we denote its $\mathcal{H}_\infty$-norm as

$$||G(s)||_\infty = \sup_\omega |G(j\omega)|, \tag{1}$$

with $j^2 = -1$. Analogously, for discrete time, we use horizontal bars to denote the discrete-time version of signals $\bar{x}(k)$, $k \in \mathbb{Z}$ of a continuous-time signal $x(t)$. The respective $\mathcal{Z}$-transform of $\bar{x}(k)$ is given by $\bar{X}(z)$, $z \in \mathbb{C}$, and the sampled equivalent or discrete-time implementation of a continuous-time transfer function $G(s)$ is given by $\bar{G}(z)$, with its respective $\mathcal{H}_\infty$-norm given by

$$||\bar{G}(z)||_\infty = \sup_\theta |G(e^{j\theta})|. \tag{2}$$

We base the following analysis on theoretical results for platooning that were provided for continuous-time models (see, e.g., [40]). However, when implementing control algorithms on real hardware, mostly through the use of digital micro-controllers, sensors, and actuators, it is often necessary to convert most, if not all, signals to their sampled-based versions. In the following, we will consider that digital controllers will be implemented using discrete-time approximations of continuous-time controllers and that continuous-time plants will be shifted to their zero-order hold (ZOH) equivalents, which is one of the most common approaches for dealing with discrete-time control implementations [14].

2.2. Platooning Setup

In this work, we consider a one-dimensional platoon consisting of $N \in \mathbb{N}$ vehicles. Each vehicle is labelled with an index $i = 1, ..., N$ and is equipped with a controller that aims to maintain a desired inter-vehicle distance while following the vehicle in front, with the exception of the leader vehicle ($i = 1$), which operates independently from the rest of the platoon. To aid in accomplishing this task, each vehicle utilizes wireless communication to transmit pertinent information to the subsequent vehicles. We assume that the platoon adopts a predecessor-following configuration; that is, each vehicle receives information only from its predecessor. This configuration is illustrated in Figure 1, where $\ell_i(k)$ denotes the inter-vehicle distance between vehicle i and its predecessor. Here, we consider that each vehicle in the platoon is capable of measuring its own position, $y_i(t)$, and its own speed, $v_i(t)$. Each vehicle also obtains its predecessor position, $y_{i-1}(t)$. Hence, each vehicle has access to the inter-vehicle distance

$$\ell_i(t) = y_{i-1}(t) - y_i(t) - L_{i-1}, \tag{3}$$

where L_{i-1} is the length of the predecessor vehicle.

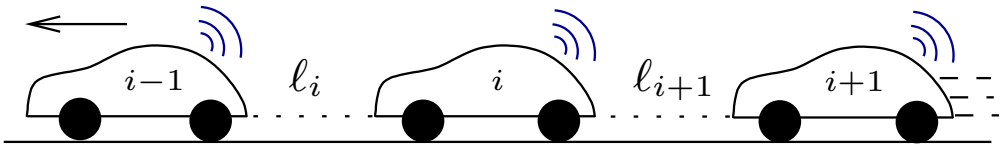

Figure 1. Vehicle platoon with a predecessor-following topology.

We assume that the platoon under study is composed of a set of homogeneous agents, and, thus, each agent has the same dynamical behavior. Specifically, we consider a continuous-time linear-time-invariant (LTI) system to model the dynamics of each agent, characterized by the transfer function $G(s)$, which is assumed to be strictly proper. Additionally, all vehicles are equipped with an identical local stabilizing controller described by the LTI system $C(s)$.

In this context, the local control loop is designed to track the predecessor vehicle and keep the inter-vehicle distance as close as possible to a predefined reference signal denoted by $r_i(t)$. In order to reduce the risk of collisions, it is prudent to set a larger distance reference when the vehicle is operating at high speeds. Conversely, for low-speed navigation, it is advisable to maintain a smaller reference distance. This approach aims for both safety and efficient operation. Hence, it is common to set $r_i(t)$ as

$$r_i(t) = \epsilon_i + hv_i(t) \tag{4}$$

where $v_i(t)$ is the velocity of the vehicle i, $\epsilon_i \geq 0$ is a constant that defines the minimum desired distance, and $h > 0$ is the time headway constant, which is a design parameter. This spacing policy has been widely studied, and it is known as the constant time headway spacing policy (see, e.g., [6,40]).

2.3. Controller Design for String Stability

Since the goal is to follow a given reference, the tracking error is defined as

$$e_i(t) = \ell_i(t) - r_i(t), \tag{5}$$

and given the previous setup, we can use Laplace domain analysis and the fact that $Y_i(s) = G(s)U_i(s)$, $U_i(s) = C(s)E_i(s)$, and $V_i(s) = sY_i(s)$ to conclude that

$$\begin{aligned} Y_i(s) &= G(s)C(s)(Y_{i-1}(s) - Y_i(s) - L_i(s) - \epsilon_i(s) - hV_i(s)) \\ &= G(s)C(s)(Y_{i-1}(s) - L_i(s) - \epsilon_i(s) - (1 + hs)Y_i(s)), \end{aligned}$$

which implies then that

$$Y_i(s) = T(s)(Y_{i-1}(s) - L_i(s) - \epsilon_i(s)), \tag{6}$$

where $T(s)$ is the transfer function

$$T(s) = \frac{G(s)C(s)}{1 + G(s)H(s)C(s)}, \tag{7}$$

with $H(s) = 1 + hs$. This shows that the local control in each vehicle can be written equivalently as the feedback loop depicted in Figure 2, where $T(s)$ corresponds to the closed-loop transfer function.

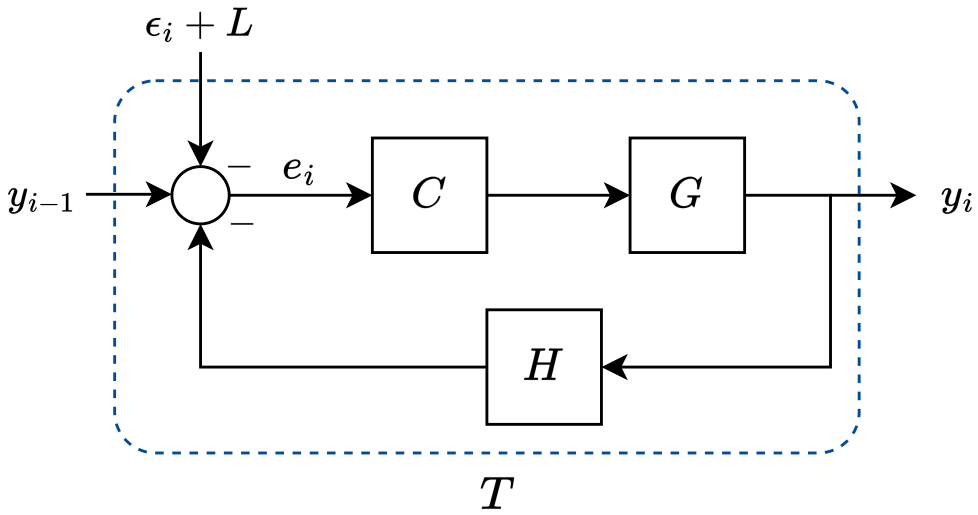

Figure 2. Feedback control loop for the position y_i in each vehicle. The time headway spacing policy is included in H.

As a primary goal, each local controller must be designed to stabilize the feedback loop and achieve zero tracking error in a steady state. In the platooning problem context, the position $y_{i-1}(t)$ becomes a ramp signal in steady state; hence, to perform the tracking task with zero stationary error, it is required that $C(s)G(s)$ has at least double integration [41].

Aside from the primary objective of tracking the predecessor, it is crucial to design the controller in such a way that it achieves the property of string stability. This property is related to the amplification of the effect of a disturbance on an individual vehicle as it propagates through the chain of vehicles. We use the following definition for string stability [42]:

Definition 1. *Let $F_n(s)$ be a sequence of stable transfer functions. The sequence is said to be string-stable if there exists $c \in \mathbb{R}$, independent of $n \in \mathbb{N}$, such that $||F_n(s)||_\infty \leq c$ for all n. The sequence will be called string-unstable otherwise.*

Numerous research efforts have been devoted to comprehending and enhancing control designs aimed at ensuring string stability. These efforts encompass various aspects, including communication topologies, vehicle models, information flow, and other relevant assumptions.

In order to contextualize platooning and the scalability of a multi-vehicle configuration in both continuous and discrete-time frameworks, with respect to the previous definition, we can recall the following lemmas [9,43].

Lemma 1. *Let $T(s)$ be a real rational scalar function of the complex variable s. Suppose that $T(0) = 1$ and also that T is stable (analytic in the closed right half complex plane). Then,*

$$\int_0^\infty \ln |T(j\omega)| \frac{d\omega}{\omega^2} \geq \frac{\pi}{2} T'(0). \tag{8}$$

Lemma 2. *Let $\bar{T}(z)$ be a real rational scalar function of $z \in \mathbb{C}$. Suppose that $\bar{T}(1) = 1$ and also that $\bar{T}(z)$ is stable (analytic in the closed unit disk). Then,*

$$\int_0^{\pi} \ln\left|\bar{T}(e^{j\theta})\right| \frac{d\theta}{1-\cos(\theta)} \geq \pi \bar{T}'(1). \tag{9}$$

The previous lemmas establish that any stable complementary sensitivity function relating to standard control loops must attain a magnitude greater than 1 for some frequency (either $x = j\omega$ or $x = e^{j\theta}$) whenever there is double integration in the open loop. Indeed, in continuous time, double integration in the product $G(s)C(s)$ implies that $T(0) = 1$, and, in discrete time, double integration in the product $\bar{G}(z)\bar{C}(z)$ implies that $\bar{T}(1) = 1$. As mentioned above, double integration is needed for tracking constant speeds and inter-vehicle distances in the formation control of platooning systems. Now, it is possible to show in both cases that $T'(0) = \bar{T}'(1) = 0$, which then implies that $|T(j\omega)|$ and $|\bar{T}(e^{j\theta})|$ must be greater than 1 for some frequency. Given this observation, it is clear that the absence of measures to prevent the effect of the double integration in the local closed loops will imply string instability of the multi-agent system according to Definition 1 since the cascading effect of disturbances at a particular vehicle will affect the vehicle k positions behind it through a multiple of T^k [9]. Hence, given the problem setup in this paper, string stability is achieved in continuous time if and only if $||T(s)||_\infty \leq 1$ and $||\bar{T}(z)||_\infty \leq 1$ for a discrete-time setting (see, e.g., [6,20,23]).

2.4. Problem of Interest

While the topic of platooning string stabilization has been extensively investigated in continuous-time scenarios, the exploration and understanding of its implementation in discrete time remain relatively limited. The effect of implementing platooning controllers in discrete time on the property of string stability is still not well-understood.

Indeed, one notable challenge in discrete-time implementation arises from the practical limitation imposed on the choice of the sample time, a critical parameter in discrete-time implementations. However, the influence of this parameter on string stability has not been thoroughly examined. The effect of the sample time choice may be further compounded by the practical inability to instantaneously measure velocity and acceleration, which are commonly assumed to be available to controllers in many continuous-time designs [6]. These assumptions, however, are not always feasible in practice.

With the aim of reducing this gap, in this paper, we examine the impact that the choice of the sample time, inherent in the discrete-time implementation, can have on the crucial property of string stability. In order to do so, we first design a string-stable controller within a continuous-time framework. Then, we obtain a discrete-time representation of such a design to implement in practice. Finally, we implement the case study on the scale-down experimental platform PL-TOON [39] and verify whether the string stability property is preserved or not for different sampling times.

3. Materials and Methods

3.1. PL-TOON Experimental Platform

We use the PL-TOON platform [39] for the experimental implementation, with 5 homogeneous agents. The agents on the platform are mounted on rails, which simplifies the control problem to one dimension, eliminating the need to deal with lateral control issues. By reducing the system to a one-dimensional setting, the focus can be placed primarily on assessing the string stability of the platoon without the influence introduced by lateral control challenges.

Regarding the capabilities of the agents, each agent on the platform is equipped with a DC motor and two sensors. The first sensor is a frontal time-of-flight sensor, specifically the VL53L0X, which measures the distance to the preceding vehicle. The second sensor is an optical sensor called ANDS3080, which is used for independent velocity estimation. The primary microcontroller unit used for each vehicle is the ESP32, which offers wireless

capabilities for broadcasting data to other agents. The experimental platform is illustrated in Figure 3. In our previous work [39], a complete description of the PL-TOON platform can be found.

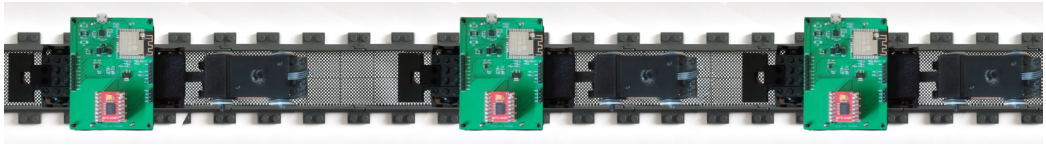

Figure 3. Three agents of the PL-TOON experimental platform.

In platooning studies, it is common to model the vehicle dynamics $G(s)$ as simple integrator [9,20], or as a second-order system with integration [24,34]. For the experimental platform, each agent's movement is governed by the DC motor dynamics. A common model for these motors is given by

$$\dot{v}_i(t) + \alpha v_i(t) = \beta u_i(t), \tag{10}$$

where $v_i(t)$ represents the velocity of the agent i, $\dot{v}_i(t)$ denotes its acceleration, and $u_i(t)$ is the motor input, which is given by the local controller. α and β are constant parameters that depend on the agent mass and friction coefficient. From (10), we obtain the transfer function

$$\frac{V_i(s)}{U_i(s)} = \frac{\beta}{s+\alpha}, \tag{11}$$

where $V_i(s)$ and $U_i(s)$ correspond to the Laplace transform of the velocity and input, respectively. Furthermore, due to Laplace properties, we have $V_i(s) = sY_i(s)$, where $Y_i(s)$ is the position of the agent in the Laplace domain. Hence, it is reasonable to describe the following model for the position

$$G(s) = \frac{Y_i(s)}{U_i(s)} = \frac{\beta}{s(s+\alpha)}. \tag{12}$$

We perform standard system identification techniques for the first-order model in (11). Specifically, the convergence rate of the system is easily deduced from the step response of the system, as well as the DC gain [44]. From this, we experimentally obtain the values of $\alpha = 4.9$ and $\beta = 1.1$. It is worth noting that, while all agents on the platform have the same design and share the same components, there may be slight variations in these parameters for each individual agent.

It is important to note that, in our setup, the transfer function between two consecutive inter-vehicle distances is equivalent to the transfer function between two consecutive vehicle positions. This means that we can describe the platoon using either the absolute positions y_i or the inter-vehicle distances ℓ_i, and the closed-loop transfer function T, as well as the criterion for string stability, remains the same. This is particularly relevant for the real implementation as the PL-TOON experimental platform does not have the capability to directly measure the absolute position y_i of each agent, but it can measure ℓ_i. Nevertheless, it is worth mentioning that the absolute position y_i of each agent can be inferred by considering the initial positions, the sampled distance measurement between vehicles, and the length of each agent L, which is the same for all agents and given by $L = 23.9$.

3.2. Continuous-Time Controller Design

We use a proportional-integrative (PI) controller for each agent, which is one of the most common types of controllers in practice, and is able to achieve string stability for our platooning setup. This type of controller incorporates integration, which, combined with the integrator of the plant $G(s)$, enables tracking of ramp references. The continuous-time

PI controller is designed to ensure internal stability for T, zero stationary error, and also string stability. This last criterion requires $||T(s)||_\infty \leq 1$.

Given the system model, we set $h = 0.62$ as a desired constant time headway policy and design the PI controller $C(s)$ as

$$C(s) = K_p + \frac{K_i}{s}, \tag{13}$$

with $K_p = 20$ and $K_i = 20$. This yields

$$T(s) = \frac{22s + 22}{s^3 + 18.54s^2 + 35.64s + 22},$$

and is such that $||T(s)||_\infty = 1$. This continuous-time design is verified in Matlab 2023a considering $N = 15$ vehicles, obtaining the tracking errors reported in Figure 4. It is clear that the platoon is string-stable since the error magnitude is reduced as i increases, as expected.

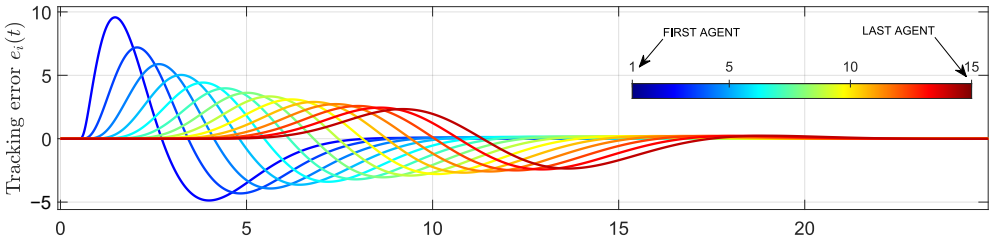

Figure 4. Tracking errors for the string-stable continuous-time platoon of size 15. The first agent is represented by darker blue, while the last agent corresponds to the dark red curve.

3.3. Discrete-Time Implementation

Although the vehicle dynamics are described in continuous time, and the design of both the spacing policy and the controller is typically completed in continuous time as well, the actual implementation requires both the controller and the spacing policy to be implemented in discrete time. This means that, in practice, the control loop is a sampled-data control system.

The sampled-data control system is obtained using a standard zero-order hold approach, with sampling time Δ. The zero-order hold discrete time equivalent of $G(s)$ can be computed as [14]

$$\bar{G}(z) = (1 - z^{-1})\mathcal{Z}\left\{\mathcal{L}^{-1}\left(\frac{G(s)}{s}\right)\Big|_{t=k\Delta}\right\}. \tag{14}$$

The control algorithm implementation in each agent was carried out using a PID control library developed for the Arduino IDE, whose coding environment is compatible with ESP32. This library approximates the continuous-time designed controller using a digital controller, namely $\bar{C}(z)$, based on the forward Euler discretization. That is

$$\bar{C}(z) = K_p + \frac{K_i\Delta}{(z-1)} = \bar{K}_p + \frac{\bar{K}_i}{(z-1)}, \tag{15}$$

with $\bar{K}_i = Ki\Delta$, and where z is the variable in the $\mathcal{Z}$-transform domain.

On the other hand, the velocity is measured considering the current and past positions as

$$\bar{v}_i(k) = \frac{\bar{y}_i(k) - \bar{y}_i(k-1)}{\Delta} \tag{16}$$

where $k \in \mathbb{Z}$ refers to the discrete-time index. Hence, the reference in discrete time is given by

$$\bar{r}_i(k) = \epsilon_i + h\bar{v}_i(k) \tag{17}$$

Note that, given $\bar{v}_i(k)$, the equivalent discrete-time version of $H(s)$ is given by

$$\bar{H}(z) = 1 + h\frac{(1-z^{-1})}{\Delta} = \frac{(1+\bar{h})z - \bar{h}}{z}, \tag{18}$$

which is a backward Euler approximation of $H(s)$.

In Figure 5, the equivalent sampled control scheme is depicted. To convert the controller output into a continuous-time signal for the plant, the zero-order hold method is employed. This method, part of the digital/analog block, is widely utilized in practice. The measurements are consistently sampled at a constant rate denoted by Δ in the analog/digital device. Then, the equivalent sampled-data closed-loop transfer function is given by

$$\bar{T}(z) = \frac{\bar{G}(z)\bar{C}(z)}{1 + \bar{G}(z)\bar{H}(z)\bar{C}(z)}.$$

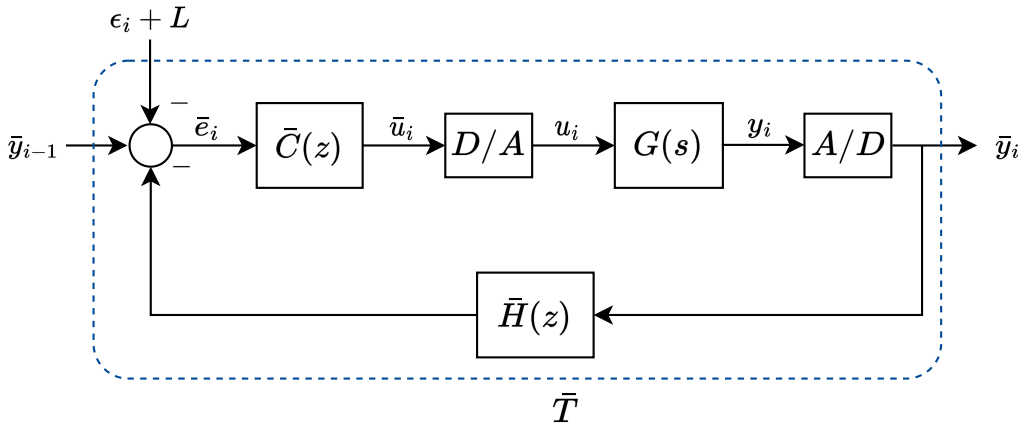

Figure 5. Feedback control loop implementation in discrete time. Both the controller $\bar{C}$ and the time headway block $\bar{H}$ are implemented digitally, whereas the input and output of $G(s)$ are converted to the appropriate time domain using a digital-to-analog converter (D/A) and an analog-to-digital converter (A/D), respectively.

Some insight into the role of the sample time on string stability can be obtained from $\bar{T}(z)$. Indeed, from (12), we have that $G(s)/s$ expanded through partial fraction decomposition is

$$\frac{G(s)}{s} = -\frac{\beta}{\alpha^2}\frac{1}{s} + \frac{\beta}{\alpha}\frac{1}{s^2} + \frac{\beta}{\alpha^2}\frac{1}{(s+\alpha)}, \tag{19}$$

which yields

$$\mathcal{L}^{-1}\left(\frac{G(s)}{s}\right)\Big|_{t=k\Delta} = -\frac{\beta}{\alpha^2}\mu(k\Delta) + \frac{\beta}{\alpha}(k\Delta)\mu(k\Delta) + \frac{\beta}{\alpha^2}e^{-\alpha k\Delta}\mu(k\Delta). \tag{20}$$

Consequently, from (14), we obtain that

$$\bar{G}(z) = (1-z^{-1})\left(\frac{-\beta}{\alpha^2}\frac{z}{z-1} + \frac{\beta}{\alpha}\frac{\Delta z}{(z-1)^2} + \frac{\beta}{\alpha^2}\frac{z}{z-e^{-\alpha\Delta}}\right) \tag{21}$$

$$= \beta\frac{(e^{-\alpha\Delta} + \alpha\Delta - 1)z + 1 - \alpha e^{-\alpha\Delta}\Delta - e^{-\alpha\Delta}}{\alpha^2(z-1)(z-e^{-\alpha\Delta})}. \tag{22}$$

Although the combined effect that the three transfer functions $\bar{C}(z)$, $\bar{G}(z)$, and $\bar{H}(z)$ have on the closed-loop dynamics may be hard to assess by inspection, we can explore the effect of the sample time and the time headway constant on the locations of the poles of $\bar{T}(z)$. Figure 6 shows the locations of the closed-loop poles of the sampled-time closed loop for 4 values of Δ and h varying from 0 to 1 with $G(s) = 1.5/(s(s+5.3))$ and $C(s) = 20 + 20/s$.

We can see that, as Δ and h increase, the closed loop may become internally unstable. Interestingly, it may be argued that the most impactful of the three parameters is $\bar{H}(z)$ as its frequency response, in (18), varies considerably as h and Δ change, as opposed to those of $\bar{C}(z)$ and $\bar{G}(z)$, in (15) and (21), respectively, which remain less perturbed. We will explore this further with the following derivations.

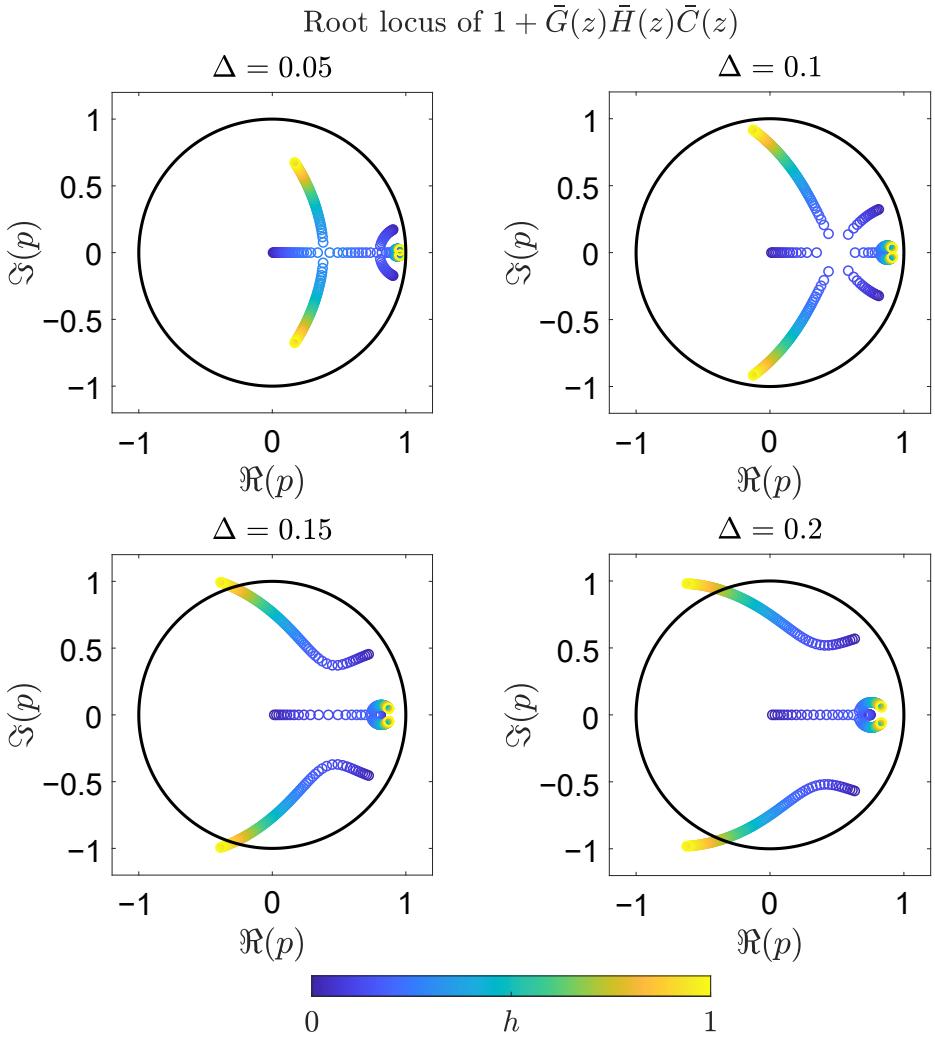

Figure 6. Pole locations of the closed-loop sampled-time transfer function $\bar{T}(z)$ for varying h with 4 values of the sample time Δ. (**Top left**) $\Delta = 0.05$ (s); (**top right**) $\Delta = 0.1$ (s); (**bottom left**) $\Delta = 0.15$ (s); (**bottom right**) $\Delta = 0.2$ (s).

To shed light on the possibility of the configuration being string-stable for some sampling time Δ, we will focus our attention on computing $\bar{T}'(1)$ when we have $\bar{H}(z)$ in the feedback loop, as in Figure 5. Its value is due to the implications that Δ may have on T given Lemma 2, which also depends on $\bar{T}'(1)$. Indeed Lemma 2 provides a Bode-type integral restriction to a closed loop that satisfies $\bar{T}(1) = 1$, which is a well-known fundamental limit in control theory (see, e.g., [43]). In our setup, the local closed loops at each vehicle must satisfy $\bar{T}(1) = 1$ to achieve predecessor tracking (zero tracking error to constant references in steady state), which is the first goal of platooning. Hence, this integral provides valuable information regarding the potential for $\left|\bar{T}(e^{j\theta})\right|$ to exceed 1 for certain values of θ, which implies string instability. The following proposition establishes a relationship between $\bar{T}'(1)$ and the sampling time, offering insights into the influence of the sampling time on the string stability given Lemma 2.

Proposition 1. *Let $\bar{T}(z)$ be the transfer function of the discrete-time loop in Figure 5, and then $\bar{T}'(1) = -h/\Delta$.*

Proof. Noting that

$$\frac{d}{dz}(\bar{T}) = \left(\frac{\bar{G}(z)\bar{C}(z)}{1+\bar{G}(z)\bar{H}(z)\bar{C}(z)}\right)' = \left(\frac{1}{\bar{H}(z)}\frac{\bar{G}(z)\bar{H}(z)\bar{C}(z)}{1+\bar{G}(z)\bar{H}(z)\bar{C}(z)}\right)' \tag{23}$$

we can now compute

$$\frac{d}{dz}(\bar{T})\Big|_{z=1} = \lim_{z\to 1} -\frac{\bar{H}'(z)}{\bar{H}(z)^2}\bar{F}(z) + \frac{1}{\bar{H}(z)}\bar{F}'(z), \tag{24}$$

where

$$\bar{F}(z) = \frac{\bar{G}(z)\bar{H}(z)\bar{C}(z)}{1+\bar{G}(z)\bar{H}(z)\bar{C}(z)}. \tag{25}$$

As mentioned after Lemmas 1 and 2, $\bar{G}(z)\bar{H}(z)\bar{C}(z)$ has, at most, two poles at $z = 1$, in particular $\bar{G}(z)\bar{H}(z)\bar{C}(z) = \bar{\Theta}(z)/(z-1)^2$ with $\bar{\Theta}(1) \neq 0$ and finite. We can now compute

$$\lim_{z\to 1}\bar{F}(z) = \lim_{z\to 1}\frac{\bar{\Theta}(z)}{(z-1)^2+\bar{\Theta}(z)} = 1, \tag{26}$$

and it is possible to write

$$\bar{F}(z) = 1 - \bar{S}(z) = 1 - \frac{1}{1+\bar{G}(z)\bar{H}(z)\bar{C}(z)}, \tag{27}$$

which implies that $\bar{S}(z)$ has two zeros at $z = 1$, and, hence, $\bar{F}'(1) = 0$. Therefore, as $\bar{H}(1) = 1$, we obtain

$$\bar{T}'(1) = -\bar{H}'(1) = -\frac{h}{\Delta}. \tag{28}$$

□

Given Lemma 2, the previous result establishes that the closed-loop transfer function that must be analyzed to verify the string stability of a cascade interconnection of vehicles satisfies the Bode-type integral condition

$$\int_0^{\pi} \ln\left|\bar{T}(e^{j\theta})\right| \frac{d\theta}{1-\cos(\theta)} \geq -\frac{\pi h}{\Delta}. \tag{29}$$

We can conclude that the effect of increasing the sampling time Δ has a potentially detrimental effect on the string stability property of the interconnected sampled system. As Δ grows, the right side of (29) approaches 0, which in turn makes it more likely that $|\bar{T}(e^{j\theta})| > 1$ for some value of θ. In particular, note that $1-\cos(\theta)$ is positive in $(0,\pi)$, which then implies that, although the integral is greater than a negative value that is small in magnitude, $\ln|\bar{T}(e^{j\theta})|$ may be positive in a sub-interval of $(0,\pi)$. On the other hand, increasing h should have a positive effect on achieving string stability, which is well-known in the continuous-time case [40] but seems to remain true for the sampled data case. As discussed before, increasing both the sampling time and the time headway constant has a negative effect on the internal stability of the closed loop, so care must be taken in order to design these parameters. Note that we cannot conclude string stability from the condition in (29) as the bound for the Bode integral being negative does not prevent $\bar{T}(z)$ from having an infinity norm greater than 1. However, the result does suggest that having $h > 0$ and $\Delta << 1$ such that h/Δ is large should contribute positively towards achieving string stability.

Figure 7 shows different Bode plots as the value of h varies from 0 to 1 for fixed values of $C(s)$, $G(s)$, which have been transformed accordingly for $\Delta \in \{0.05, 0.1, 0.15, 0.2\}$. It is possible to see that, for $\Delta = 0.05$ (s), there exists a minimum value of h_m for which any

$h > h_m$ achieves $|\bar{T}(e^{j\theta})| \leq 1$ for $\theta \in (0, 2\pi)$. For $\Delta = 0.1$ (s), we see that there is a range of values of h that achieve string stability, but, as h grows, $||\bar{T}(z)||_\infty > 1$. For $h = 1$, we again have $||\bar{T}(z)||_\infty = 1$. Similar behaviors occur for larger Δ, with certain resonance phenomena arising around the higher values of the frequency range. This behavior will become evident in some of the experimental results (Figure 8).

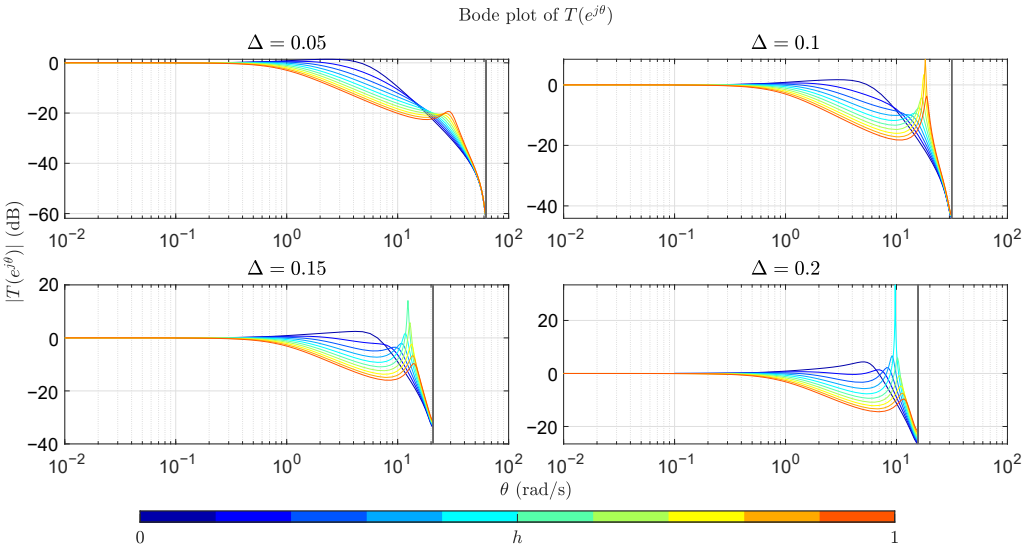

Figure 7. Magnitude plots for the transfer function $\bar{T}(z)$ for varying h with 4 values of the sample time Δ. (**Top left**) $\Delta = 0.05$ (s); (**top right**) $\Delta = 0.1$ (s); (**bottom left**) $\Delta = 0.15$ (s); (**bottom right**) $\Delta = 0.2$ (s).

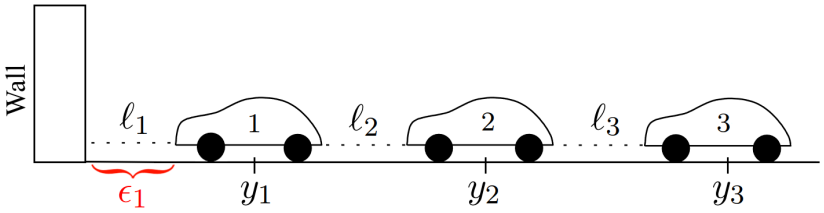

Figure 8. Setup of the experiment to perform showing 3 PL-TOON agents. The platoon starts in formation with the desired distance ϵ_i between vehicles and then the setpoint ϵ_1 changes its value periodically, which provokes a tracking error in the first vehicle that then propagates to the rest of the platoon.

4. Simulation and Experimental Results

In this section, we report the results obtained via simulations, which will then be corroborated by using the PL-TOON platform.

4.1. Experiment Description

The experiment to evaluate the string stability of the platoon consists of five agents in formation, with the first agent of the platoon facing a wall. At the start, all agents are set to be at rest and in the desired formation. To introduce a disturbance, the distance setpoint of the first car, denoted as ϵ_1, is varied every 30 s. The remaining values for ϵ_i are held constant at 20 cm. As a result, the first agent accelerates to achieve the new desired position, yielding a transient tracking error. This transient error subsequently propagates to the following vehicles in the platoon. Figure 3 provides a visual representation of the platoon configuration for this experimental setup.

In order to observe the effect of the sample time Δ in the string-stable behavior of the vehicles, we perform the experiment and collect data multiple times for different sample time values. The minimum sampling rate of the distance sensor is 20 ms, and, hence, we cannot go below this sample time value since a new measurement will not be available. On the other hand, we set 0.20 s as the upper limit for the sample time since it is sufficient

to observe the effect on string stability without compromising internal stability. Thus, we repeat the experiment for sample time values within 0.02 and 0.20 s. Except for the sampling time, all other parameters involved in the control loop and its implementation remain invariant.

It is worth mentioning that conducting experiments with a larger number of agents, such as 10, 20, or more, could provide even clearer evidence of string stability/instability. However, practical challenges arise when attempting to perform experiments with such a significant number of agents due to technical and financial constraints. This situation further emphasizes the limitations faced when trying to analyze string stability in real-world scenarios.

4.2. Simulation Results

For the simulation results, we calculate the discrete-time closed-loop transfer function considering three different sampling times, $\Delta = 0.02$, $\Delta = 0.125$, and $\Delta = 0.17$.

In the case of $\Delta = 0.02$, we have

$$\bar{T}(z) = 0.001 \, \frac{4.26z^3 - 0.0517z^2 - 4.04z}{z^4 - 2.77z^3 + 2.68z^2 - 1.034z + 0.1253},$$

yielding $||\bar{T}(z)||_\infty = 1$. Hence, the platoon remains string-stable for this sample time. This is observed in Figure 9, where the behavior of a platoon of five vehicles is simulated.

For $\Delta = 0.125$, we have

$$\bar{T}(z) = 0.1 \, \frac{1.416z^3 - 0.08382z^2 - 1.01z}{z^4 - 1.698z^3 + 1.332z^2 - 1.103z + 0.5012},$$

which also leads to a string-stable platoon since $||\bar{T}(z)||_\infty = 1$. This is depicted in Figure 10; however, it can also be observed that the performance of the platoon is deteriorated, compared to the case with $\Delta = 0.02$, due to the ripple and higher amplitudes exhibited by the tracking errors.

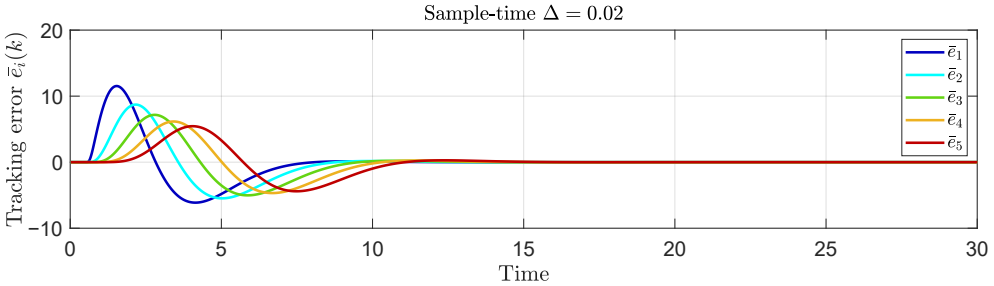

Figure 9. Simulation results for the tracking error for the vehicle platoon with a sampling time $\Delta = 0.02$ s.

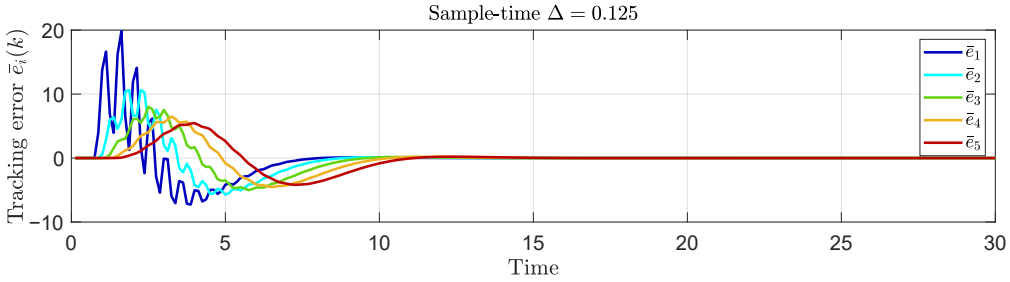

Figure 10. Simulation results for the tracking error for the vehicle platoon with a sampling time $\Delta = 0.125$ s.

Finally, for $\Delta = 0.170$, we have

$$\bar{T}(z) = 0.1\,\frac{2.453z^3 - 0.1751z^2 - 1.545z}{z^4 - 1.295z^3 + 0.8934z^2 - 1.089z + 0.5634}.$$

For this case, we have $||\bar{T}(z)||_\infty = 1.0388$, and, hence, the platoon becomes string-unstable. The platoon behavior is shown in Figure 11, where it is clear that the errors are increasing along the string of vehicles. We can observe in this example that string stability is lost due to the inappropriate sampling time choice. Note that the amplitudes of the errors also increase considerably for this case, reaching values near 50. The oscillatory behavior of each error is also undesired, and it is also a consequence of the sampling time choice. Nevertheless, each vehicle in the platoon reaches its references and the tracking errors converge to zero given that the loss of string stability does not imply the loss of internal stability.

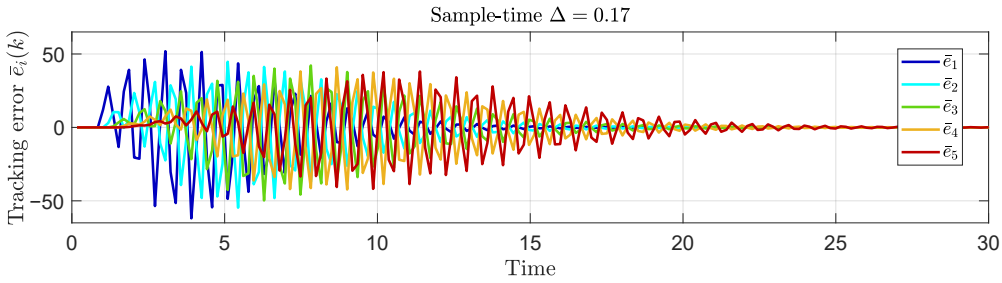

Figure 11. Simulation results for the tracking error for the vehicle platoon with a sampling time $\Delta = 0.17$ s.

4.3. Experimental Results

It is important to mention that the simulation results and the experimental results were not the same for the same design parameters. This could be due to differences in the actual dynamics of the cars compared to the proposed simplified model. Additionally, there may be non-linear phenomena that were not modeled and could influence the discrepancy between the experimental and simulated results. However, the design parameters were slightly adjusted to clearly observe the same qualitative behavior that demonstrates the effect of the sampling time. Thus, the controller maintains the same parameters as in the simulation, but the experimental value for the parameter h is $h = 0.3$ instead of 0.6. Furthermore, instead of using $\Delta = 0.125$ (s), an experimental value of $\Delta = 0.120$ (s) is used, and, instead of using $\Delta = 0.17$ (s), $\Delta = 0.18$ (s) is used experimentally. We also consider additional values of Δ in the experimental setup. In Figure 12, the tracking errors $\bar{e}_i$ are shown for different values of Δ. The effect of the sampling time in the platoon behavior can be observed as well, where the amplitude of the errors starts to increase along with Δ. The behavior of the transient is also affected by the sample time, where, for larger values of Δ, the response of the vehicles is much more oscillatory and faster than for smaller values of Δ. Next, we will focus on three Δ values similar to those used for the simulation results and analyze the corresponding graphs in more detail.

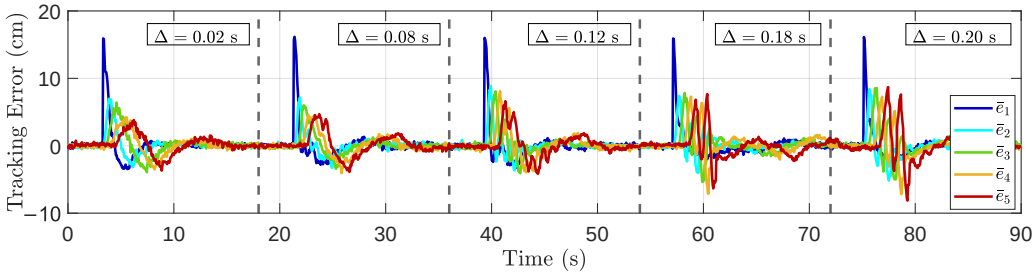

Figure 12. Tracking error for the experimental vehicle platoon with different sample times Δ in the range 0.02–0.20 s.

We begin by using a sampling time of $\Delta = 0.02$ (s). The results for the tracking errors are shown in Figure 13, where it can be inferred, by visual inspection, that the platoon exhibits string-stable behavior since the errors are not being amplified as it propagates along the vehicles. Therefore, we can conclude that the string stability property of the platoon, designed in a continuous-time framework, is preserved in the discrete-time implementation for this particular sampling time.

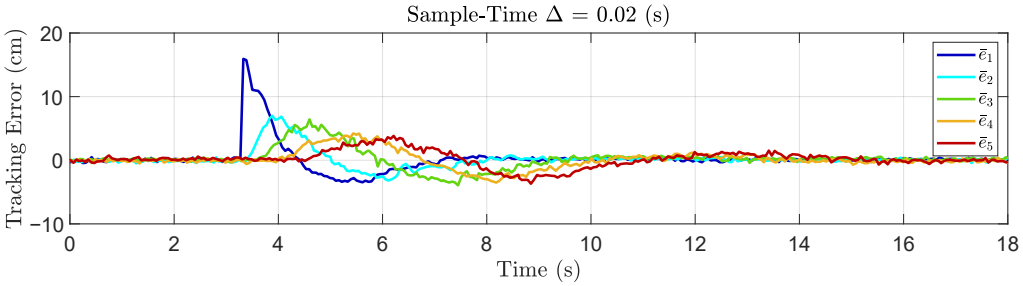

Figure 13. Tracking error for the experimental vehicle platoon with a sampling time $\Delta = 0.02$ (s).

Then, we increase the sampling time to $\Delta = 0.12$ (s), and the corresponding results are depicted in Figure 14. In this scenario, it is evident that the error amplitudes are higher compared to the previous case with $\Delta = 0.02$ (s). Additionally, there is a notorious presence of oscillations in the tracking errors. Nonetheless, it is not evident in this case whether the platoon is string-stable or not. At first glance, it appears that the error is not being amplified at each agent. However, due to the small size of the platoon, it is not possible to confidently determine this solely through visual inspection. It is plausible that, with the addition of more agents to the platoon, the tracking error might exhibit clear string-unstable behavior for this particular sample time.

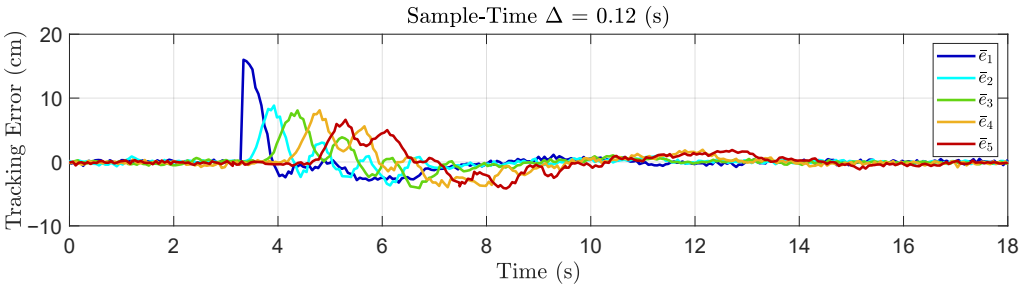

Figure 14. Tracking error for the experimental vehicle platoon with a sampling time $\Delta = 0.12$ (s).

Lastly, we conduct the experiment using a sampling time of $\Delta = 0.18$ (s). The results of the platoon tracking error under this sample time are illustrated in Figure 15. We observe here a greater amplification in the error signals and a higher prevalence of oscillations. In this scenario, the presence of string instability can be stated with more confidence as the error amplifies in the agents while propagating through the platoon. This can be observed by comparing the magnitudes of the errors when they are negative, wherein the magnitudes are higher than those of their preceding agents.

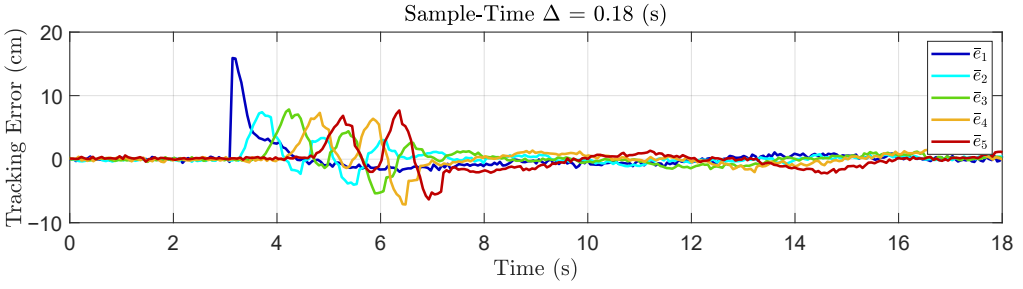

Figure 15. Tracking error for the experimental vehicle platoon with a sampling time $\Delta = 0.18$ (s).

In Table 1, we present the integral of the squared error for our experimental results. We can clearly observe that the first agent is the most affected by the disturbance in all cases, which is reasonable for small platoons. When $\Delta = 0.02$, we can see that the error tends to decrease, reaching values more than three times lower than the error of the first car. We can infer with some confidence that this is stable string behavior. However, for $\Delta = 0.12$, we notice that, while the error of the second car is considerably lower than that of the first car, the errors of the remaining cars do not exhibit a consistent trend of increase or decrease. This lack of consistent behavior can be attributed to the unclear string stability for this particular value of Δ. It is probably influenced by the fact that agents' models are not exactly the same and, hence, the small variations in the agent's models could have an impact $||\bar{T}(z)||_\infty$ in each case. Only by experimentally adding more vehicles could we determine instability. In the last case ($\Delta = 0.18$), as the error increases from the second car to the last, it clearly indicates unstable string behavior.

Table 1. Comparison of the ISE index for the tracking error for different sample times.

	$\Delta = 0.02$ (s)	$\Delta = 0.12$ (s)	$\Delta = 0.18$ (s)
$\bar{e}_1$	5.2655	4.9257	3.7994
$\bar{e}_2$	2.1130	1.9896	1.9605
$\bar{e}_3$	2.6374	2.2446	2.6644
$\bar{e}_4$	1.8563	2.9717	2.9070
$\bar{e}_5$	1.6580	2.6531	3.1332

Our simulation and experimental results suggest that there is a critical value for the choice of Δ, below which string stability is achieved. It is important to mention that, in the previous analysis, for all the reported choices of Δ, internal stability and zero steady-state error are achieved. Thus, the analysis focuses solely on string stability. Certainly, a poor choice of sampling time can also deteriorate the performance of the controller and even cause internal instability, as illustrated in the previous section, but this is well-known in the literature on sampled data feedback control and not a consequence of the string stabilization analysis.

5. Conclusions and Future Work

Our study examined the dynamics of vehicle platoons operating in a predecessor-following configuration using sampled-data control systems. The key objective was to assess the impact of the sampling time on the string stability of the platoon. Starting with the design of a string-stable platoon in continuous time, we then discretized the controller and simulated the resulting sampled data control system. Finally, we implemented the control strategy on a scaled-down experimental platform. Through simulation and real experimentation, we analyzed the effects of varying sampling times on the platoon's string stability performance. Our findings indicate that an improper selection of the sampling time could significantly undermine the platoon's string stability. Our results also suggest that there is a critical choice of the sample time below which string stability is guaranteed. Thus, careful consideration of the sampling time is essential in ensuring the desired string stability properties when implementing control systems for platooning applications. This study provides valuable insights into the relationship between sampling time and the stability of vehicle platoons.

Future research in this field entails conducting a more comprehensive theoretical analysis to elucidate the impact of sampling on string stability and obtaining necessary and sufficient conditions on the sampling time to preserve string stability. Studying the effect of non-uniform sampling is also part of future work, as well as platoons of vehicles moving beyond a one-dimensional setting.

Author Contributions: Conceptualization, F.J.V.; methodology, F.J.V.; software, F.I.V. and A.A.P.; validation, F.I.V.; formal analysis, all authors; investigation, all authors; resources, F.J.V. and A.A.P.; data curation, F.I.V. and F.J.V.; writing—original draft preparation, all authors; writing—review and editing, all authors; visualization, all authors; supervision, F.J.V.; project administration, F.J.V.; funding acquisition, F.J.V. and A.A.P. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been funded by UTFSM, ANID FONDECYT 11221365 grant, and CORFO Project 14ENI226865.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank Felipe Ulloa of UAI for their valuable technical support.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Liang, K.Y.; Mårtensson, J.; Johansson, K.H. Heavy-duty vehicle platoon formation for fuel efficiency. *IEEE Trans. Intell. Transp. Syst.* **2015**, *17*, 1051–1061. [CrossRef]
2. Jia, D.; Lu, K.; Wang, J.; Zhang, X.; Shen, X. A survey on platoon-based vehicular cyber-physical systems. *IEEE Commun. Surv. Tutorials* **2015**, *18*, 263–284. [CrossRef]
3. Bonnet, C.; Fritz, H. *Fuel Consumption Reduction in a Platoon: Experimental Results with Two Electronically Coupled Trucks at Close Spacing*; Technical Report, SAE Technical Paper; SAE International: Warrendale, PA, USA, 2000.
4. Shladover, S.E.; Nowakowski, C.; Lu, X.Y.; Ferlis, R. Cooperative adaptive cruise control: Definitions and operating concepts. *Transp. Res. Rec.* **2015**, *2489*, 145–152. [CrossRef]
5. Wang, Z.; Wu, G.; Barth, M.J. A review on cooperative adaptive cruise control (CACC) systems: Architectures, controls, and applications. In Proceedings of the IEEE 2018 21st International Conference on Intelligent Transportation Systems (ITSC), Maui, HI, USA, 4–7 November 2018; pp. 2884–2891.
6. Feng, S.; Zhang, Y.; Li, S.E.; Cao, Z.; Liu, H.X.; Li, L. String stability for vehicular platoon control: Definitions and analysis methods. *Annu. Rev. Control* **2019**, *47*, 81–97. [CrossRef]
7. Stüdli, S.; Seron, M.M.; Middleton, R.H. From vehicular platoons to general networked systems: String stability and related concepts. *Annu. Rev. Control* **2017**, *44*, 157–172. [CrossRef]
8. Balador, A.; Bazzi, A.; Hernandez-Jayo, U.; de la Iglesia, I.; Ahmadvand, H. A survey on vehicular communication for cooperative truck platooning application. *Veh. Commun.* **2022**, *35*, 100460. [CrossRef]
9. Middleton, R.H.; Braslavsky, J.H. String instability in classes of linear time invariant formation control with limited communication range. *IEEE Trans. Autom. Control.* **2010**, *55*, 1519–1530. [CrossRef]
10. Van Nunen, E.; Reinders, J.; Semsar-Kazerooni, E.; Van De Wouw, N. String stable model predictive cooperative adaptive cruise control for heterogeneous platoons. *IEEE Trans. Intell. Veh.* **2019**, *4*, 186–196. [CrossRef]
11. Ge, X.; Han, Q.L.; Ding, D.; Zhang, X.M.; Ning, B. A survey on recent advances in distributed sampled-data cooperative control of multi-agent systems. *Neurocomputing* **2018**, *275*, 1684–1701. [CrossRef]
12. Ge, X.; Han, Q.L.; Ding, L.; Wang, Y.L.; Zhang, X.M. Dynamic event-triggered distributed coordination control and its applications: A survey of trends and techniques. *IEEE Trans. Syst. Man Cybern. Syst.* **2020**, *50*, 3112–3125. [CrossRef]
13. Zhang, X.M.; Han, Q.L.; Ge, X.; Ning, B.; Zhang, B.L. Sampled-data control systems with non-uniform sampling: A survey of methods and trends. *Annu. Rev. Control* **2023**, *55*, 70–91. [CrossRef]
14. Åström, K.J.; Wittenmark, B. *Computer-Controlled Systems: Theory and Design*; Courier Corporation: Chelmsford, MA, USA, 2013.
15. Wang, P.; Deng, H.; Zhang, J.; Wang, L.; Zhang, M.; Li, Y. Model predictive control for connected vehicle platoon under switching communication topology. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 7817–7830. [CrossRef]
16. Feng, S.; Sun, H.; Zhang, Y.; Zheng, J.; Liu, H.X.; Li, L. Tube-based discrete controller design for vehicle platoons subject to disturbances and saturation constraints. *IEEE Trans. Control. Syst. Technol.* **2019**, *28*, 1066–1073. [CrossRef]
17. Ma, F.; Wang, J.; Zhu, S.; Gelbal, S.Y.; Yang, Y.; Aksun-Guvenc, B.; Guvenc, L. Distributed control of cooperative vehicular platoon with nonideal communication condition. *IEEE Trans. Veh. Technol.* **2020**, *69*, 8207–8220. [CrossRef]
18. Zhang, J.; Peng, C.; Xie, X. Platooning control of vehicular systems by using sampled positions. *IEEE Trans. Circuits Syst. II Express Briefs* **2023**, *early access*. [CrossRef]
19. Gordon, M.A.; Vargas, F.J.; Peters, A.A. Mean square stability conditions for platoons with lossy inter-vehicle communication channels. *Automatica* **2023**, *147*, 110710. [CrossRef]
20. Vargas, F.J.; Maass, A.I.; Peters, A.A. String stability for predecessor following platooning over lossy communication channels. In Proceedings of the 23rd International Symposium on Mathematical Theory of Networks and Systems (MNTS), Hong Kong, China, 16–20 July 2018; pp. 834–837.
21. Li, Z.; Hu, B.; Li, M.; Luo, G. String stability analysis for vehicle platooning under unreliable communication links with event-triggered strategy. *IEEE Trans. Veh. Technol.* **2019**, *68*, 2152–2164. [CrossRef]

22. Zhao, C.; Cai, L.; Cheng, P. Stability analysis of vehicle platooning with limited communication range and random packet losses. *IEEE Internet Things J.* **2020**, *8*, 262–277. [CrossRef]
23. Gordon, M.A.; Vargas, F.J.; Peters, A.A. Comparison of simple strategies for vehicular platooning with lossy communication. *IEEE Access* **2021**, *9*, 103996–104010. [CrossRef]
24. Villenas, F.I.; Vargas, F.J.; Peters, A.A. A Kalman-Based Compensation Strategy for Platoons Subject to Data Loss: Numerical and Empirical Study. *Mathematics* **2023**, *11*, 1228. [CrossRef]
25. Acciani, F.; Frasca, P.; Heijenk, G.; Stoorvogel, A.A. Stochastic string stability of vehicle platoons via cooperative adaptive cruise control with lossy communication. *IEEE Trans. Intell. Transp. Syst.* **2021**, *23*, 10912–10922. [CrossRef]
26. Li, Z.; Hu, B.; Yang, Z. Co-design of distributed event-triggered controller for string stability of vehicle platooning under periodic jamming attacks. *IEEE Trans. Veh. Technol.* **2021**, *70*, 13115–13128. [CrossRef]
27. Zhao, N.; Zhao, X.; Chen, M.; Zong, G.; Zhang, H. Resilient distributed event-triggered platooning control of connected vehicles under denial-of-service attacks. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 6191–6202. [CrossRef]
28. Xiao, S.; Ge, X.; Han, Q.L.; Zhang, Y. Dynamic event-triggered platooning control of automated vehicles under random communication topologies and various spacing policies. *IEEE Trans. Cybern.* **2021**, *52*, 11477–11490. [CrossRef] [PubMed]
29. Shen, Z.; Liu, Y.; Li, Z.; Nabin, M.H. Cooperative spacing sampled control of vehicle platoon considering undirected topology and analog fading networks. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 18478–18491. [CrossRef]
30. Kianfar, R.; Falcone, P.; Fredriksson, J. A control matching model predictive control approach to string stable vehicle platooning. *Control. Eng. Pract.* **2015**, *45*, 163–173. [CrossRef]
31. Dolk, V.S.; Ploeg, J.; Heemels, W.M.H. Event-triggered control for string-stable vehicle platooning. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 3486–3500. [CrossRef]
32. Thormann, S.; Schirrer, A.; Jakubek, S. Safe and efficient cooperative platooning. *IEEE Trans. Intell. Transp. Syst.* **2020**, *23*, 1368–1380. [CrossRef]
33. Murillo, A.; Vargas, F.; Peters, A. Effects of speed saturation in a predecessor-following vehicle platoon. In Proceedings of the 2019 IEEE CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON), Valparaiso, Chile, 13–27 November 2019; pp. 1–7.
34. Naus, G.J.L.; Vugts, R.P.A.; Ploeg, J.; van de Molengraft, M.J.G.; Steinbuch, M. String-Stable CACC Design and Experimental Validation: A Frequency-Domain Approach. *IEEE Trans. Veh. Technol.* **2010**, *59*, 4268–4279. [CrossRef]
35. Segata, M.; Bloessl, B.; Joerer, S.; Sommer, C.; Gerla, M.; Cigno, R.L.; Dressler, F. Toward communication strategies for platooning: Simulative and experimental evaluation. *IEEE Trans. Veh. Technol.* **2015**, *64*, 5411–5423. [CrossRef]
36. Johnson, M.; Hayes, M.J. String Stability Experimentation using a Low-Cost Mobile Robotic Testbed. In Proceedings of the 2019 30th Irish Signals and Systems Conference (ISSC), Maynooth, Ireland, 17–18 June 2019; pp. 1–6. [CrossRef]
37. Qin, W.B.; Orosz, G. Experimental Validation of String Stability for Connected Vehicles Subject to Information Delay. *IEEE Trans. Control. Syst. Technol.* **2020**, *28*, 1203–1217. [CrossRef]
38. Escobar, C.; Vargas, F.J.; Peters, A.A.; Carvajal, G. A Cooperative Control Algorithm for Line and Predecessor Following Platoons Subject to Unreliable Distance Measurements. *Mathematics* **2023**, *11*, 801. [CrossRef]
39. Peters, A.A.; Vargas, F.J.; Garrido, C.; Andrade, C.; Villenas, F. PL-TOON: A Low-Cost Experimental Platform for Teaching and Research on Decentralized Cooperative Control. *Sensors* **2021**, *21*, 2072. [CrossRef]
40. Klinge, S.; Middleton, R.H. Time headway requirements for string stability of homogeneous linear unidirectionally connected systems. In Proceedings of the 48h IEEE Conference on Decision and Control (CDC) Held Jointly with 2009 28th Chinese Control Conference, Shanghai, China, 15–18 December 2009; pp. 1992–1997. [CrossRef]
41. Goodwin, G.C.; Graebe, S.F.; Salgado, M.E.; Elsley, G. *Control System Design*; Prentice Hall: Upper Saddle River, NJ, USA, 2001; Volume 240.
42. Peters, A.A.; Middleton, R.H.; Mason, O. Leader tracking in homogeneous vehicle platoons with broadcast delays. *Automatica* **2014**, *50*, 64–74. [CrossRef]
43. Seron, M.M.; Braslavsky, J.H.; Goodwin, G.C. *Fundamental Limitations in Filtering and Control*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2012.
44. Badillo, D.; Huidobro, C.; Villenas, F.; Peters, A.; Vargas, F. Sensor Calibration and Filtering for an Agent of the PL-TOON Platooning Platform. In Proceedings of the 2021 IEEE CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON), Virtual, 6–9 December 2021; pp. 1–6. [CrossRef]