Wireless vehicle-to-infrastructure data gathering for robot platooning

Reference:

Caruntu Constantin F., Copot Cosmin, Lazar Corneliu.- Wireless vehicle-to-infrastructure data gathering for robot platooning
25th Mediterranean Conference on Control and Automation, 3-6 July, 2017, Valletta, Malta - ISSN 2473-3504 - Piscataway, N.J., IEEE, 2017, p. 1-6
Full text (Publisher's DOI): http://dx.doi.org/doi:10.1109/MED.2017.7984262

Wireless vehicle-to-infrastructure data gathering for robot platooning*

Constantin F. Caruntu[1], Cosmin Copot[2], Corneliu Lazar[1]

Abstract— **This paper presents firstly the development of the infrastructure to emulate and verify the behavior of robots in a platooning scenario based on mOway mobile robots. Radio Frequency communications and how they are handled in both ends are the keys in this distributed wireless vehicle-to-infrastructure (V2I) data gathering implementation. Secondly, the paper investigates a state-space model predictive control strategy for platoon guidance using only longitudinal changes for the automatically controlled robots. The control strategy was implemented and tested in simulation and in real-time, while the data gathered through the distributed wireless V2I system was used to monitor the mobile robots on-line and afterwards to analyze the behavior of the mobile robot platoon. Studying robot platooning and the relationship with the communication issues allows one to understand the dynamics of the platoon and, therefore, to develop a suitable traffic control strategy.**

I. Introduction

Nowadays, road vehicles are still the most widely used transportation units worldwide for persons and goods, but the continuously growing number of vehicles has led to the increase of traffic flow on highways and city roads. The result is that there are more and more traffic jams leading to accidents, causing driving stress and passenger discomfort, losing the efficiency of vehicles and thus increasing the fuel consumption and pollution. One solution to these problems is given by the Automated Highway Systems, which are able to ensure safe and efficient coordination of vehicles [1].

The rapid growth of connected devices in the world lead to an exponential growth of opportunities for the automotive industry to take advantage of this newly available information. By equipping vehicles with technologies capable of broadcasting, receiving and processing pertinent data, the impact on safety, convenience, and mobility is groundbreaking [2]. The connected vehicle market consists of: Vehicle-to-Vehicle (V2V) communications between two vehicles and Vehicle-to-Infrastructure (V2I) communications between a vehicle and a fixed piece of the surrounding infrastructure. Both technologies rely on Dedicated Short Range Communications to transmit and receive information in a vehicle. As a natural result of the primary information broadcasted by the DSRC modules (vehicle location, speed, and direction), most of the applications of V2V technologies are focused on collision avoidance, specifically with other vehicles on the road [2]. V2I applications have a broader scope, as they tend to be less immediate or safety-critical. V2I opens up the possibility for more complex data analysis and for information to be stored over time. Contrary to V2V, V2I only requires the infrastructure and the vehicle to be connected, allowing for useful applications immediately without significant market penetration of connected vehicles [3].

V2I communications for safety is the wireless exchange of critical safety and operational data between vehicles and highway infrastructure, intended primarily to avoid motor vehicle crashes and enable a wide range of other safety, mobility, and environmental benefits. Preliminary studies show that an additional 12 percent of potential crash scenarios could be addressed by V2I safety applications [4]. The vision for the V2I research is to enable safety applications that are designed to avoid or mitigate vehicle crashes. In V2I communications, the infrastructure plays a coordination role by gathering global or local information on traffic and road conditions and then suggesting or imposing certain behaviors on a group of vehicles. The velocities and accelerations of vehicles and inter-vehicle distances would be suggested by the infrastructure on the basis of traffic conditions, with the goal of optimizing overall emissions, fuel consumption, and traffic velocities. Suggestions to vehicles could be broadcast to drivers via road displays or directly to vehicles via wireless connections. Looking further ahead, in some cases suggestions could be integrated into the vehicle controls and implemented semi-automatically.

Driving assistant systems with cameras (e.g., mono, stereo, surround view) and sensors (e.g., radar, ultrasonic) are expected as a solution for safe driving, congestion prevention and driving load reduction. A sensor-fusion technology could be used to identify the obstacles [5]. The ultrasonic sensor with high sensitivity and low price is used for short distance detection. For long distance obstacle detection and recognition a camera could be used. Moreover, the global positioning system (GPS) and inertial measurement unit (IMU) are used to measure the vehicles body angle information and current position information. All these measurements could be transmitted to the infrastructure through V2I communications and can be used on-line to inform the other vehicles about any emergency and off-line for complex data analysis.

It is difficult for traditional data collection efforts to collect accurate and reliable sensor measurements across large, outdoor environments, e.g., highways and city roads. Static sensors do not scale well as the hardware requirements increase quickly with large environments. Even dense static sensor deployments cannot capture fine-grain measurements, so they often use interpolation techniques to estimate fine-

*The work of C. F. Caruntu was supported by a grant of the Romanian National Authority for Scientific Research and Innovation, CNCS UEFIS-CDI, project number PN-II-RU-TE-2014-4-0970.

[1] C. F. Caruntu and C. Lazar are with the Department of Automatic Control and Applied Informatics Gheorghe Asachi Technical University of Iasi, Iasi, Romania. caruntuc@ac.tuiasi.ro

[2] C. Copot is with the Department of Electromechanics, Op3Mech, University of Antwerp, Antwerp, Belgium.

Fig. 1. mOway robot

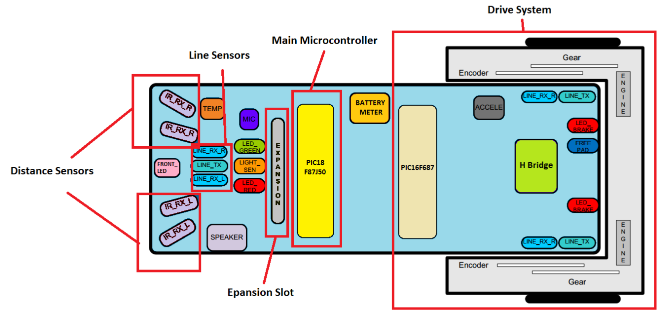

Fig. 2. Overview of the mOway robot

grain variations [6]. With the introduction of autonomous vehicles, we can opportunistically use them for collecting sensor measurements about their surroundings. The use of vehicles for data collection scales much better than the static sensors in terms of number of measurement locations and provide better accuracy and reliability. Moreover, using GPS devices, they can localize themselves continuously with high accuracy, while adding other sensing devices can improve the diversity of the collected measurements in order to prevent the dangerous situations.

Each measurement update can be represented as a time and value tuple $< t_i, s_i >$. While in operation, the vehicle can easily record sensor measurements as well as corresponding time and location as it moves across the environment. Given that different sensors update at different rates, one can align the measurement updates $< t_i, s_i >$ by time-stamp to identify the best location estimate $< t_i, x_i, y_i >$ for each sensor measurement $< t_i, x_i, y_i, s_i >$.

In this paper, the robot platooning problem is studied considering the relationship with the communication issues, which allows to understand the dynamics of the platoon and, therefore, to develop a suitable traffic control strategy. As such, firstly the development of the infrastructure to emulate and verify the behavior of robots in a platooning scenario based on mOway mobile robots is presented. Radio Frequency communications and how they are handled in both ends are the keys in this distributed wireless vehicle-to-infrastructure (V2I) data gathering implementation. Secondly, the paper investigates a state-space model predictive control strategy for platoon guidance using only longitudinal changes for automatically controlled robots. The control strategy was implemented and tested in simulation and in real-time, while the data gathered through the distributed wireless V2I system was used to monitor the mobile robots online and afterwards to analyze the behavior of the platoon.

II. System Description

The robot considered for this study is a differential wheeled mOway mobile robot (Fig. 1), which is an autonomous programmable small robot designed mainly to perform practical mini-robotics applications [7].

A. Hardware description

The mOway mobile robots are equipped with a wide range of sensors such as temperature, 3-shaft accelerometer, light intensity sensor, anti collision and infrared line sensors. The motors are controlled using the Inter-Integrated Circuit (I2C) protocol. The logical overview configuration of the robot is illustrated in Fig. 2. As it can be seen in the figure, the robot is governed by a master Microchip 8-bit PIC micro-controller (PIC18), which works with a frequency of 4Mhz [7] and has 4KB of RAM and 32 connections [8].

A servo-motor group for each of the two wheels in the backside is used to drive the robot. The drive system is illustrated in detail in Fig. 2 in the red rectangle from the right. The secondary Microchip 8-bit PIC micro-controller (PIC16) sends the I2C commands (i.e., PWM signals) to the drive system (i.e., H-bridge) that controls the motors, while the encoding sticker and infrared sensor are used to measure the speed of the wheel. An internal proportional controller with negative feedback from the encoders is used to control the speed of the robot, keeping it constant on different terrains.

The robot includes two line tracking sensors (Vishay CNY70) mounted on the front bottom of the robot. The emitting light source and the detector are arranged in the same direction, meaning that the reflective light can be detected on the spot. There are 3 types of surfaces that the sensor can detect:

- clear $\rightarrow$ the infrared light is almost completely reflected and the sensor registers a low voltage;
- colored $\rightarrow$ partially reflecting the beam, will register different levels depending on the color; in this way, colors can also be identified;
- dark $\rightarrow$ light is little to not reflected at all and the sensor registers a high voltage.

The obstacle detection sensor uses infrared light to detect objects in front of the mOway mobile robot. The sensor has two emitting sources (Kingbright KPA3010-F3C) and four receivers (PT100F0MP) placed on both sides in front of the robot. The output of the receivers are connected to the analog input of the micro-controller so it can detect the presence of any object and measure the distance to it. The light emitter generates a 70us pulse that allows the receivers to detect obstacles using a filter and an amplifying stage. Once the signal is processed electronically, the micro-controller can measure it using an Analog-to-Digital Converter (ADC) or as a digital input. The measured distance is between

3 cm to 11 cm and the surface color and environment brightness influences the sensor accuracy. To minimize the effects of disturbances on the distance reading a blank back was attached to the mOways as seen in Fig. 1.

The RF modules mounted on the mOways are bidirectional, but there is one particularity: the communication is half-duplex. According to Tanenbaum [9], a half-duplex (HDX) system provides communication in both directions, however they cannot be simultaneous (only one direction at a time is permitted). The main characteristics of the RF module are: 2.4GHz working frequency, low consumption and a transmission speed between 1 and 2 Mbps.

B. RF communications

One of the challenging tasks was to develop a strategy to use the RFUsb as a MultiPoint-to-Point (MP2P) device [10] because the default configuration for the mOway communication system (using the libraries and transmitting without any data control) lead to the following problems: the data received was badly structured, many packages were lost and there was no index parameter to inform if a package was a transmission or a retransmission.

The first problem to be addressed was the package configuration. Important data to be gathered from the system were: distance sensor reading (ssr), speed (spd), traveled distance (km), battery status ($batt$) and transmission id (id), which is monitored by the communication protocol, for each mOway robot. Taking into account that the mOway could only send 8 bytes at a time, the package structure was defined as $[0, 0, batt, km\_2, km\_1, ssr, spd, id]$. Other data that could be useful in vehicle platooning would include: acceleration, heading direction, exact position, future behavior for use in distributed control strategies [11] and other safety-critical information, e.g., accidents on the road, blocked roads, queues at an intersection.

The PC side development was based on a polling technique to receive the data. The time needed to perform a 1 round poll in every mOway highly scales with the number of present subjects. To avoid that, a passive (PC side) - active (mOway side) technique was developed in which the mOway continuously sends data to the RFUsb module. The sampling time was too fast and unstable for the RFUsb to process the data and wait for the next one to arrive. That lead, again, to package losses and various delays as retransmissions from every mOway were happening continuously. As such, instead of changing the mOways, the PC-side application was redesigned in which buffers were used and threads called. This way, the minimum time was achieved on actually processing and storing the package, avoiding being locked in the process and, therefore, not being able to receive more data. Although improvements were perceived, this was not enough as the sampling time from each mOway was inconsistent and causing overload in the channel. According to the identification performed, the minimum sampling time could go as high as 50ms. In order to set the sampling time, the algorithm in the mOway robots was slightly modified so it would accept timed interruptions and then the sending procedure could be controlled.

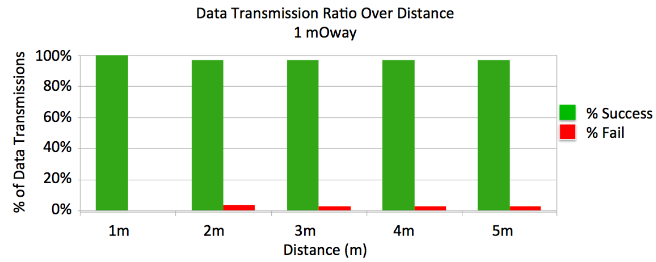

Fig. 3. Success data transmission for 1 mOway over distance

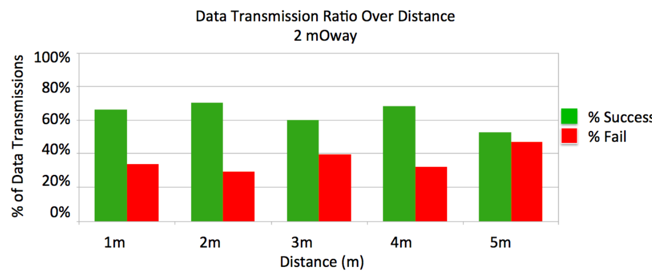

Fig. 4. Success data transmission for 2 mOways over distance.

After synchronizing the sending by setting an initial pulse when the system was ready to start, the timers in every mOway could throw exceptions at the same time. The packages were transmitted, but the ratio of success deliveries over the total sending was about 25% to 30%. As such, this ratio was improved by setting a higher time for the sending interruptions, while keeping the data sampling at 50ms. As only 8 bytes could be sent at a time, data had to be narrowed down and the final package had this configuration: $[km\_2, km\_1, ssr, spd, id, ssr, spd, id]$.

In the following figures, the success ratio over distance, number of mOways used in the test bench and total packages is illustrated. As it can be seen in Fig. 3, the transmission behavior on the tests for one mOway was almost perfect.

When two mOways were put together, the success rate dropped immediately from almost 100% to about 60% and as it can be noticed from Fig. 4 the distance did not influenced too much the communications.

Finally, a test with 4 mOways was set and the transmission rate is illustrated in Fig. 5. The success rate dropped from 60% to 50%. The abnormal loss of packages for the tests in 4m presented in Fig. 5 are correlated with a problem of the transceiver of one robot.

C. mOway robot modeling

The model of the robot was determined using a sequence of steps response identification experiment illustrated in Fig. 6. The speed of the mobile robot is internally controlled, so a first order transfer function can be used to represent the robots dynamics since the robot is a simple process based on DC motors and only a proportional (P) controller.

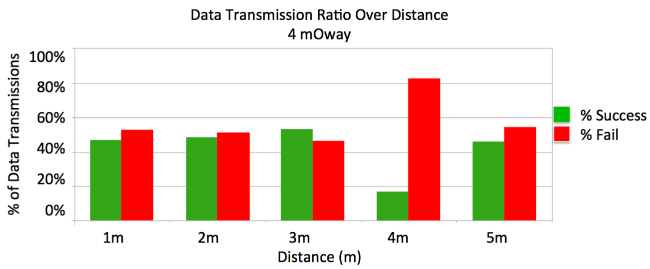

Fig. 5. Success data transmission for 4 mOway over distance.

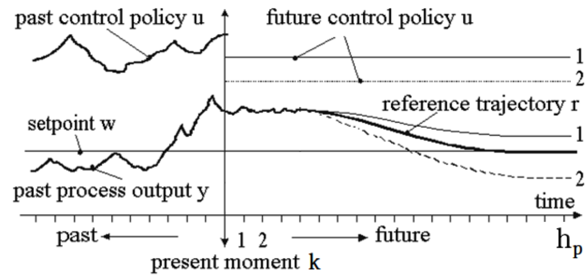

Fig. 7. The MPC principle

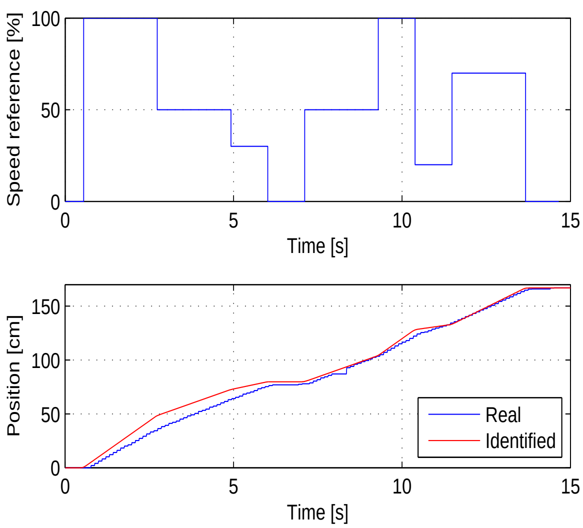

Fig. 6. Identification experiment.

The resulting transfer function between the reference speed and the robot position is given by a first order system plus an integrator [1], [12]

$$G(s) = \frac{K}{s(Ts+1)} = \frac{0.22}{s(0.04s+1)}. \tag{1}$$

The comparison between the real output of the robot (position) and the output of the identified model is illustrated in Fig. 6, the fitting between the two signals being of 94%. Please note that the input of the process is not a force, but a reference for the robot speed given as a percentage. At the same time, the speed is measured in cm/s and has a maximum of 22 cm/s yielding a gain for the transfer function equal to 0.22.

III. Predictive Controller Design

Model based predictive control (MPC) is a control methodology that uses a process model for calculating on-line the predictions of the future plant output and based on that it optimizes the future control actions. Also, constraints can be taken into account in this optimization. The MPC-strategy is simple to understand and makes good practical sense. First, a process model is used to predict the evolution of the process output as a function of future (intended) control actions. Secondly, a specified cost index is minimized over these control actions. This cost typically includes the errors between the desired and predicted process outputs. The MPC principle is depicted in Fig. 7. Referring to this figure, the following strategy is followed:

• at each 'current' moment k, the process output $y(k+i)$ is predicted over a time horizon $i = 1, ..., h_p$; the predicted values are indicated by $y(k+i|k)$ and the value of h_p is called the prediction horizon; the prediction is done by means of a model of the process and depends on the past inputs and outputs, but also on the future control scenario $\{u(k+i|k), i = 0, ..., h_p - 1\}$;

• a reference trajectory $\{r(k+i|k), i = 1, ..., h_p\}$, evolving towards the setpoint w is defined over the prediction horizon, describing how to guide the process output from its current value $y(k)$ to the setpoint w;

• the control vector $\{u(k+i|k), i = 0, ..., h_p - 1\}$ is calculated in order to minimize a specified cost function, e.g., the most simple type of cost function, where the control effort is not taken in account, is defined as:

$$\sum_{i=1}^{h_p} (r(k+i|k) - y(k+i|k))^2 \xrightarrow{Minimize} u(k+i|k). \tag{2}$$

• the first element $u(k|k)$ of the optimal control vector is applied to the real process; all other elements of the calculated control vector are discarded.

As such, at the next sampling time this optimization is repeated, taking into account the new measurement information. This introduces actually the feedback component in the whole strategy, resulting in a closed-loop configuration.

Consider the discrete-time linear system corresponding to the model of the mOway mobile robot (1)

$$\mathbf{z}(k+1) = \mathbf{A_d}\mathbf{z}(k) + \mathbf{B_d}u(k), \quad k \in \mathbb{Z}_+, \tag{3}$$

where $\mathbf{z}(k) \in \mathbb{R}^2$ is the state vector, $u(k) \in \mathbb{R}$ is the control input (reference for the mobile robot speed) at the discrete-time instant k and $\mathbf{A_d}$ and $\mathbf{B_d}$ are the discrete-time system matrices.

For the mOway mobile robots, the discrete-time model (3) was obtained by discretizing model (1) with a sampling period $T_s = 5$ ms and is defined by the following system matrices

$$\mathbf{A_d} = \begin{pmatrix} 1 & 0.0287 \\ 0 & 0.2901 \end{pmatrix}, \mathbf{B_d} = \begin{pmatrix} 0.0047 \\ 0.1570 \end{pmatrix} \tag{4}$$

with $\mathbf{z}(k) = [x(k) \quad v(k)]$, where $x(k)$ is the absolute position and $v(k)$ the speed of the mobile robot at time instant k.

The predicted state in matrix form has the following representation

$$\hat{\mathbf{Z}}(k) = \mathbf{M}\mathbf{z}(k) + \mathbf{C}\mathbf{U}(k), \tag{5}$$

where

$$\mathbf{U}(k) := \begin{pmatrix} u(k|k) \\ u(k+1|k) \\ \vdots \\ u(k+h_p-1|k) \end{pmatrix}, \hat{\mathbf{Z}}(k) := \begin{pmatrix} \hat{\mathbf{z}}(k+1|k) \\ \hat{\mathbf{z}}(k+2|k) \\ \vdots \\ \hat{\mathbf{z}}(k+h_p|k) \end{pmatrix}, \tag{6}$$

$u(k+i|k)$, $i = 0, \ldots, h_p - 1$, is the future control sequence with h_p the prediction horizon, $\hat{\mathbf{z}}(k+i|k)$ is the predicted value of the state vector

$$\begin{aligned} &\hat{\mathbf{z}}(k+i+1|k) = \\ &\quad = \mathbf{A_d}\hat{\mathbf{z}}(k{+}1|k) + \mathbf{B_d}u(k+i|k), i = 0, 1, 2, \ldots \end{aligned} \tag{7}$$

with the initial condition defined by

$$\hat{\mathbf{z}}(k|k) = \mathbf{z}(k), \tag{8}$$

$$\mathbf{M} = \begin{bmatrix} \mathbf{A_d} \\ \mathbf{A_d}^2 \\ \vdots \\ \mathbf{A_d}^{h_p} \end{bmatrix}, \mathbf{C} = \begin{bmatrix} \mathbf{B_d} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{A_d}\mathbf{B_d} & \mathbf{B_d} & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A_d}^{h_p-1}\mathbf{B_d} & \mathbf{A_d}^{h_p-2}\mathbf{B_d} & \cdots & \mathbf{B_d} \end{bmatrix}. \tag{9}$$

The predictive control law is computed as

$$\mathbf{U}^*(k) = \arg\min_{\mathbf{U}} J(k) \tag{10}$$

and is based on the minimization of a cost function

$$\begin{aligned} &J(k) = \hat{\mathbf{z}}(k+hp|k)^T\bar{\mathbf{Q}}\hat{\mathbf{z}}(k+hp|k) + \\ &\sum_{i=0}^{h_p-1} \left(\hat{\mathbf{z}}(k+i|k)^T\mathbf{Q}\hat{\mathbf{z}}(k+i|k) + u(k+i|k)^T\mathbf{R}u(k+i|k)\right), \end{aligned} \tag{11}$$

that can be defined in accordance with the future control sequence $\mathbf{U}$ and the predicted state $\hat{\mathbf{Z}}$ and has the following matrix form

$$J(k) = \mathbf{U}^T(k)\mathbf{H}\mathbf{U}(k) + 2\hat{\mathbf{Z}}^T(k)\mathbf{F}^T\hat{\mathbf{Z}}(k) + \hat{\mathbf{Z}}^T(k)\mathbf{G}\hat{\mathbf{Z}}(k), \tag{12}$$

where

$$\begin{aligned} \mathbf{H} &= \mathbf{C}^T\tilde{\mathbf{Q}}\mathbf{C} + \tilde{\mathbf{R}} \\ \mathbf{F} &= \mathbf{C}^T\tilde{\mathbf{Q}}\mathbf{M} \\ \mathbf{G} &= \mathbf{M}^T\tilde{\mathbf{Q}}\mathbf{M} + \mathbf{Q}, \end{aligned} \tag{13}$$

with

$$\tilde{\mathbf{Q}} = \begin{bmatrix} \mathbf{Q} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \ddots & & \vdots \\ \vdots & & \mathbf{Q} & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \bar{\mathbf{Q}} \end{bmatrix} \text{ and } \tilde{\mathbf{R}} = \begin{bmatrix} \mathbf{R} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \ddots & & \vdots \\ \vdots & & \mathbf{R} & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{R} \end{bmatrix} \tag{14}$$

where $\mathbf{Q}, \mathbf{R}$ and $\bar{\mathbf{Q}}$ are positive defined matrices (the matrix $\mathbf{Q}$ can be positive semi-defined) of appropriate dimensions.

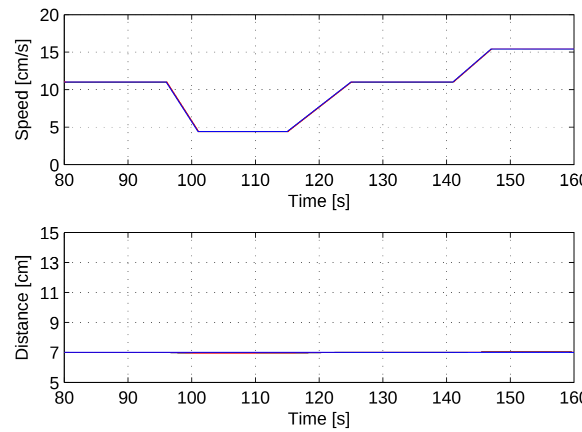

Fig. 8. Simulation: speeds of the mobile robots and distances

If there are no constraints, the solution of the optimization problem (10) can be obtained as

$$\mathbf{U}^*(k) = -\mathbf{H}^{-1}\mathbf{F}\mathbf{z}(k) \tag{15}$$

and according to the receding horizon principle only the first control command in $\mathbf{U}$ is actually applied to the process and is defined as

$$u(k) = u^*(k|k) = \mathbf{K}_{h_p}\mathbf{z}(k) \tag{16}$$

where $\mathbf{K}_{h_p} = -\begin{bmatrix}\mathbf{I}_{h_p} & \mathbf{0} & \cdots & \mathbf{0}\end{bmatrix}\mathbf{H}^{-1}\mathbf{F}$.

IV. Simulation and Experimental Results

The controller was designed to minimize the error $e(k) = d(k) - r(k)$ where $d(k)$ is the distance between the current robot and the robot in front of it and $r(k)$ is the reference distance, for $k \in \mathbb{Z}_+$. In our case, $r(k) = r$ was considered equal to 7 cm, for all $k \in \mathbb{Z}_+$. Note that a reference for the first state, i.e., the position of the mobile robot, can be computed as $x(k) = x(k-1) + e(k)$. The distance $d(k)$ was measured at each sampling period using the obstacle sensor and the second state (the speed of the robot) was not measured, but computed as the derivative of the first state (the position of the robot)

The simulation results obtained with the state-space MPC controller for which the design parameters were chosen to obtain a compromise between good performances and faster closed loop with a prediction horizon set to $hp = 5$ are illustrated in Fig. 8. The parameters of the MPC controller are as follows: $\bar{\mathbf{Q}} = 2\mathbf{I}_2$; $\mathbf{Q} = 1.5\mathbf{I}_2$; $\mathbf{R} = 1$; there were no constraints included in the optimization problem. The V2I communication was only used for data gathering for offline behavior analysis.

The experimental results obtained with the same SS-MPC controller are represented in Fig. 9, in which it can be seen that the distance error is close to 0. Note that the oscillatory responses are also due to the effects of disturbances on the distance sensor (reading). It can be observed that the speeds of the followers are higher than the speed of the leader because the followers have to travel more to maintain a predefined reference distance to the mobile robot in front

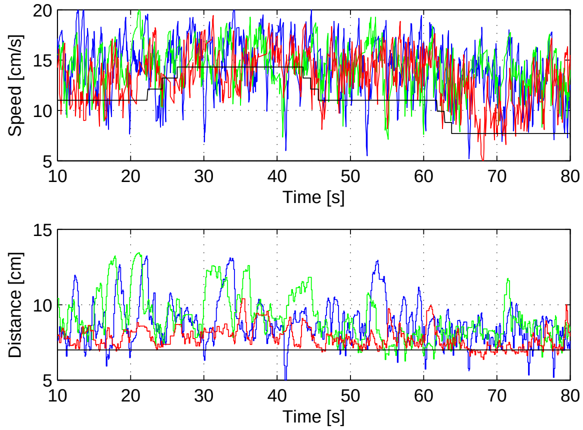

Fig. 9. Experiment: speeds of the mobile robots and distances

which implies that a higher average speed is reached (see subsection IV-A for further details).

A. Experimental remarks

In Fig. 9 it can be observed that the average speeds of the followers are continuously higher than the speed of the leader. This is due to several circumstances regarding the experimental setup as follows:

• in the performed experiments all the mobile robots follow a line marked on the floor;

• the mobile robots do not remain perfectly on the line while running, but if a deviation occurs from the imposed trajectory, this deviation is considered as a disturbance [13] that is rejected by modifying the angular speed of the left motor through a feed-forward *gain*; still, this means that they have to travel more to maintain a predefined reference distance to the mobile robot in front which implies that a higher average speed is reached;

• the movement around the line the robots should follow also influences the distance measured by each mobile robot (if the leader is straight on the line and the follower is orientated a little to the left/right than the distance measured by the latter is practically higher than the real distance which actually corresponds to a higher average speed);

• when the leader reaches a curve in the line, the follower is not able to detect the distance precisely (usually the measured distance is higher) which causes the speed to increase suddenly in order to reduce the artificial error.

B. Communication related remarks

Based on the data gathered by the mOway mobile robots, we can conclude that:

• the V2I communication system is running smoothly;

• the 4 mobile robots involved in the experimental results can keep a safety distance to drive in a platooning scenario;

• because of the communication hardware limitations, the gathered data was strictly related to the dynamics of the mobile robots; they could be upgraded with various sensors to provide in depth informations about the environment;

• the research could benefit from including more mobile robots in the platoon and improving the communication stability;

• the robots could communicate directly to each other using V2V communications for fastening and optimizing the decision making.

V. Conclusion

The focus of this paper was on implementing a wireless V2I communication system for data gathering in a robot platooning scenario. The developed infrastructure to emulate a platoon of mOway mobile robots was presented firstly together with the RF communication system used for data gathering. Then, a state-space model predictive control strategy was designed to maintain a safety distance between the robots in the platoon and the developed strategy was tested both in simulation and in real-time. The behavior of individual mobile robots and the whole platoon was analyzed based on the data gathered using the V2I communication system. It was shown that the platoon of mobile robots behaves as expected, but as pointed out in subsections IV-A and IV-B the whole infrastructure could be improved to obtain more information about the environment and to stabilize the communication system.

References

[1] C. F. Caruntu, C. Copot, C. Lazar, and R. De Keyser, "Longitudinal control of vehicle platoons for stop-and-go waves mitigation," in *18^{th} International Conference on System Theory, Control and Computing*, Sinaia, Romania, 2014, pp. 670–675.

[2] J. Harding, G. R. Powell, R. Yoon, J. Fikentscher, C. Doylec, D. Sade, M. Lukuc, J. Simons, and J. Wang, "Vehicle-to-vehicle communications: Readiness of V2V technology for application," National Highway Traffic Safety Administration, Washington, DC, Tech. Rep., 08 2014.

[3] W. Chen, *Vehicular Communications and Networks*. Woodhead Publishing, 2015.

[4] L. Zha, P. Songchitruksa, and K. N. Balke, "Next generation safety performance monitoring at signalized intersections using connected vehicle technology," Texas A&M Transportation Institute, Tech. Rep., 08 2014.

[5] O. Gietelink, J. Ploeg, B. De Schutter, and M. Verhaegen, "Development of advanced driver assistance systems with vehicle hardware-in-the-loop simulations," *Vehicle System Dynamics*, vol. 44, 2006.

[6] C. Phillips, M. Ton, D. Sicker, and D. Grunwald, "Practical radio environment mapping with geostatistics," in *IEEE International Symposium on Dynamic Access Networks*, Bellevue, WA, USA, 2012, pp. 422–433.

[7] *mOways User Manual*, 2012.

[8] *PIC18F87J50 Family Data Sheet*. Microchip Technology, 2009.

[9] A. Tanenbaum, *Computer Networks*. Prentice Hall PTR, 2003.

[10] C. Copot, C. F. Caruntu, J. P. S. De Azevedo, and R. De Keyser, "Networked communications over a lab-scale test bench for autonomous highway systems," in *19^{th} International Conference on System Theory, Control and Computing*, Cheile Gradistei, Romania, 2015, pp. 454–459.

[11] A. Maxim, C. F. Caruntu, and C. Lazar, "Distributed model predictive control algorithm for vehicle platooning," in *20th International Conference on System Theory, Control and Computing*, Sinaia, Romania, 2016, pp. 657–662.

[12] C. Copot, C. F. Caruntu, and R. De Keyser, "Advanced control technique to mitigate the stop-and-go waves on a highway traffic with different vehicles," in *21st International Conference on Methods and Models in Automation and Robotics*, Miedzyzdroje, Poland, 2016, pp. 948–953.

[13] D. V. Neamtu, E. Fabregas, B. Wyns, R. De Keyser, S. Dormido, and C. M. Ionescu, "A remote laboratory for mobile robot applications," in *18th IFAC World Congress*, Milano, Italy, 2011, pp. 7280–7285.