The Influence of Wireless Communication Transport Latencies and Dropped Packages on Vehicle Stability with an Offsite Steering Controller

Rory Bennett[1], Reyn Kapp [1], Theunis R. Botha [1*], Schalk Els [1]

[1] Department of Mechanical and Aeronautical Engineering, University of Pretoria, Lynnwood road, Pretoria, South Africa
[*]theunis.botha@up.ac.za

Abstract: In recent years, Advanced Driver Assistance Systems (ADAS) have been used to improve the safety of vehicles by either providing additional information to the driver or by taking over complete control. The majority of ADAS currently being utilised run entirely on the vehicle, only having access to information provided by the sensors that are on board the vehicle itself. Part of the next step in the evolution of ADAS is to incorporate information from other offsite sensors or obtain control inputs from infrastructure which can coordinate multiple vehicles simultaneously via a wireless interface. Wireless communication is inherently delayed and prone to dropped packets. This study looks at the effect of transport latencies and dropped packets on an off-site autoregressive steering controller supplying direct steering inputs to a vehicle. A fully non-linear vehicle simulation model is used to test the effect of delaying steering inputs and dropped packets in order to test the stability of the controller. The study shows that at dropped packet percentages of up to 40% adequate vehicle control is maintained, while transport latencies of up to 100ms allow for moderately accurate vehicle control.

1. Introduction

Recent years have seen rapid development and implementation of ADAS to mitigate the severity of, or sometimes entirely prevent, vehicle accidents. ADAS improves vehicle safety by providing the driver with additional information or, in certain instances, taking over control from the driver completely. The need for ADAS is apparent when considering that 90% of vehicle accidents are due to human error [1]. These types of accidents are largely avoidable if the correct driver action overrides the incorrect action. In most ADAS, the vehicle uses information from on board sensors to make decisions. The next advancement is to have the vehicle's on board sensor data and control supplemented by offsite data and control. This offsite supplementation is accomplished using a wireless communication channel between the offsite infrastructure and any vehicle falling under said infrastructure's supervision.

There is also a substantial push in the mining, agricultural and construction sectors to move vehicle operators from the actual vehicles to remote stations. This has been done to partly improve efficiency, since operators work in better conditions, and to improve safety, since operators are not immediately at risk [2]. These kinds of changes have mainly been implemented on very slow-moving vehicles but pose a possible solution for control of high-speed mining vehicles, while improving safety - whether remotely controlled by an offsite operator or a fully autonomous control system controlled by an offsite controller. Thus the next step in mining, agricultural and construction would be to completely automate these vehicles by making use of offsite control as this would bring an improvement in safety and efficiency.

Vehicle to Infrastructure (V2I), Vehicle to Vehicle (V2V), as well as other Vehicle to Everything (V2X), communication systems are required to facilitate such offsite control systems. Communication between Vehicle to Infrastructure (V2I) and other Vehicle to Vehicle (V2V) is required to perform offsite control. Several studies have already developed automated/supervisory highway systems [3] and intersection supervisory control systems to mitigate collisions and improve traffic flow [4], [5]. These systems assume that uninterrupted information flow from the vehicles is available at any given time, thus the possibility of interrupted V2I or V2V communication is not addressed. These studies also solely focus on specifying vehicle longitudinal velocity, without noting how this control would be realised. Eidehall et al. [6] developed a steering based collision avoidance system that concentrated on providing information pertaining to the necessity of performing an emergency lane manoeuvre, but the effect of transmission delays was neglected. This study aims to determine whether a vehicle can be treated as a Hardware-in-the-Loop (HIL) simulation where the full vehicle is treated as the hardware in the simulation.

Studies have made use of network distributed HIL systems communicating with one another over distances. Different communication media has been utilised in a network-distributed HIL setup depending on the required application, distance, and limitations. Schreiber et al [7] uses Ethernet to couple an electro-hydraulic brake test bench and a brake dynamometer residing in the same local area network (LAN). Conversely in [8] an engine-in-the-loop test rig located in Ann Arbor, MI, USA is coupled to a driver-in-the-loop ride motion simulator located in Warren, MI, USA via the Internet. Kloc et al. [9] and Franchi et al. [10] propose concepts for wireless HIL testing of automotive electronic control units where a vehicle does not have on-board control intelligence and communicates with a remote master computer while driving. For offsite vehicle control making use of using V2I or V2V technologies, use of a wireless communication media needs to be used is needed. However wireless communication is fraught with longer communication delays and dropped packets - when compared to wired media. It is well-known that the incorporation of a communication network into a closed-loop system can cause degraded system performance or instability due to inherent transmission delays [11]. In general, the effects of time delays in closed-loop systems can resemble the effects of lowering sample frequencies as the controller is presented with old data [12]. Even though researchers have proposed metrics to quantify these detrimental effects in the frequency domain [13] and time-domain [14], it still presents a challenge and remains a topic of active research.

Although on board vehicle control is a well-researched topic, the effect of transport delays and packet drops between the vehicle and its controller has not been given nearly as much attention.

As presented above there is therefore a need to move towards autonomous vehicle control and making use of V2I and V2V communication - and possibly offsite control - to further improve vehicle safety and efficiency. However, the effects the inherent delays and dropped packets in wirelesses communicating will have on the performance and stability of offsite vehicle steering is are still largely unknown and need to be studied further. This article studies the effects of transport latencies and dropped packets on offsite vehicle control in simulations as well as experimental tests with a focus on mining, agricultural, and construction vehicles. The main contribution of this article is the insight into how the delays and dropped packets may affect lateral vehicle control and whether the delays and dropped packets represent a significant problem to the use of offsite vehicle control systems. The analysis and results would however also be beneficial to the passenger vehicle environment.

The remainder of this article is organised in the following manner: Section 2 deals with the wireless interface that was used during generation of the latency model of Sections 3 and 4. Section 5 covers the theory behind the vehicle model used by the controller of Section 6. Section 7 contains details of the simulations conducted prior to carrying out the experimental work of Section 8. Section 9 closes off with conclusions drawn from the work carried out.

2. Wireless Interface

A wireless interface is required to allow the vehicle to be remotely controlled by an offsite system. While 5G is specifically designed with V2V and V2I communication in mind, and therefore would most likely be the obvious choice for such an interface, at the time of writing, 5G is still in development and a few years away until freely available on a large enough scale. The other available options for the wireless connection comes in the form of the 802.11 standard as set up by the Institute of Electrical and Electronics Engineers (IEEE), Worldwide Interoperability for Microwave Access (WiMAX) set up by the IEEE as well, and older versions of Long Term Evolution (LTE). Due to WiMAX requiring stationary nodes and LTE only recently introducing Device-To-Device communications for V2V purposes with 5G, they are either not suitable or not yet mature enough for Direct Short-Range Communication (DSRC). The 802 project has been around since 1980, with the 802.11p amendment of 2010 having a special focus on Vehicular Ad-hoc NETworks (VANET), wireless networks with a relatively high mobility of nodes. The main usage for the 802.11p standard is for that of Vehicle-To-Everything (V2X), this incorporates Vehicle-To-Pedestrian (V2P), V2I, and V2V. To accomplish these kinds of communications, the 802.11p standard is set to operate seven 20MHz channels (six service channels and one control channel) [15] in the 5.9GHz band, and is able to reach bit-rates of 27Mbps within a required operating range of 1km. The spectrum/band used varies between countries, with the values presented in this study falling under the specifications set forth by the National Highway Traffic Safety Administration (NHTSA) of the United States of America (USA)..

Although very mature, the 802.11p standard is not without its own set of shortfalls, namely the hidden terminal problem and the possibility of unbounded delays while using Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA) at the Media Access Control (MAC) layer. The hidden terminal problem occurs when two nodes, that are a part of the same network, are not within communicating range of one another. This means that they are unable to sense whether or not the other node is transmitting a message before attempting to transmit their own message, resulting in a collision at the receiving node. The possibility of unbounded delays is as a result of a device, making use of CSMA/CA, waiting for an idle channel before sending its message. If the channel is permanently busy the message will never get a chance to be sent. It was shown in simulations by [16] that during heavy congestion periods up to 50% of packets were discarded as the transmitting node received the next packet of information before it had a chance to send the current packet. It has been suggested by some that to combat both of these problems a Self-organised Time-Division Multiple Access (STDMA) protocol at the MAC layer be used, where a schedule for the sending of packets is generated by the nodes of the network [17]. This scheduling for the sending of packets also helps create a more reliable real-time network, as each node in the network is granted a period during which it can send a packet. In line with this, the Nv2 protocol by Mikrotik, making use of a proprietary Time-Division Multiple Access (TDMA) MAC layer protocol [18] was used in this study.

The development of the protocols for V2V and V2I interfaces is an ongoing research topic with researchers attempting to reduce network latencies and improve network throughput and overall robustness to ensure collision avoidance systems composed of multiple vehicles remain safe [19], [20], [21], [22]. The field of research will also change significantly when 5G becomes fully available. This study will not attempt to improve any aspect of the protocols.

3. Latency Model

A wireless latency model of the Nv2 protocol was developed and incorporated within a vehicle simulation environment to determine the effect of delay and dropped packets on vehicle stability and path following ability. Any network experiences some form of latency between a packet being sent and the packet being received at the other end. In a wireless network, the latency the packet experiences does not come from one source, rather it is a combination of many smaller latencies from different sections of the network. The four main sources of latencies in a WLAN are shown in Fig. 1, which corresponds to the network simulated in this study.

These four sources are: propagation delays, transmission delays, processing delays, and packet drops.

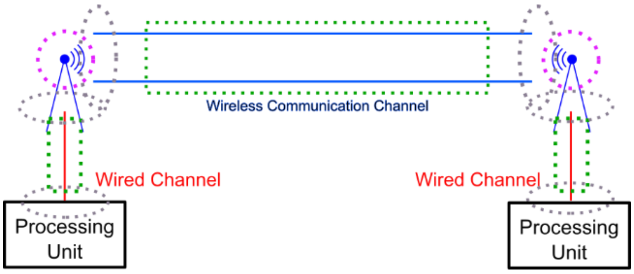

Fig. 1. Latencies in a wireless communication network

Propagation delay is the time taken for information to travel across a channel, in VANETs this is the air between the transmitter and the receiver Fortunately, its effect is largely negligible unless the distances between nodes reaches more than a few kilometres [23]. Transmission delay is the time taken to put a packet of information on a channel and is a function of the packet size and the bit-rate of the transmitter. Processing delays are the result of the handling of packets within the WLAN. In the network simulated, there are four locations for processing delays during a packet's life-cycle: At each of the processing units, as the messages are encrypted and decrypted for analysis, and at each of the Mikrotik routers, as they are analysed to determine their required destination. In a network that makes use of some form of ACKnowledgements (ACKs), packet drops can add substantially to the overall latency experienced by a packet due to the resending of the dropped packets. By making use of a User Datagram Protocol (UDP) between the processing units and the Mikrotik routers, and the Nv2 protocol not making use of ACKs, packet drops do not add to the latencies experienced in this study as if a packet is dropped it will not be resent and therefore not affect latency.

4. Latency Model Generation Procedure

To generate the required data for the latency model used during simulations two embedded computers with real time Linux kernels are used, along with two Mikrotik routers where one acts as an AP (required by the Nv2 protocol) and the other as a node. The embedded computers have a wired communication channel with the routers, with a bit-rate of 100Mpbs, while the wireless frequency of the routers is set to run near the 5.9GHz band, adhering as closely as possible to the standards being used by the NHTSA The tests are conducted using stationary placements for the routers with a clear line of site between the two routers while the distance between the routers is gradually increased from a distance of 20m up to 700m. This provides the optimal wireless condition without any additional noise or obstructions. A direct wired test is also conducted as a baseline test, the two processing units are connected directly with an Ethernet cable, with no intermediate Mikrotik routers. Each test lasts 30 minutes and is run three times at each distance to ensure the results obtained are truly representative of a network making use of the Nv2 protocol. One processing unit is set up to transmit 100 Byte packets at a rate of 100Hz, while the other processing unit is set to receive the packet and return it immediately to the original processing unit. Due to the difficulty in synchronising the two processing units' clocks with each other, the Round-Trip Time (RTT) is measured instead of measuring the one-way latency. In WLANs there is no guarantee that one-way latency = RTT/2, as asymmetrical transmission times can occur, so RTT was used as the baseline for the latency calculations. To calculate the RTT, the 100 Byte message includes a time stamp of when the message is sent. Upon the message returning to the original processing unit, this original time stamp is compared with the current time on the original processing unit and from this comparison the RTT is determined. This process adds up all the delays from packing the message, sending it, receiving it from the remote node, and unpacking the message.

The results of the wired baseline test and the 700m test are given in Fig. 2 with a statistical summary of all the tests given in Table 1. When considering the wireless results from Table 1 the general pattern appears to be that as the distance between the antennae is increased, the mean and the standard deviation around the mean also increases. However, when looking at the histogram of the 700m wireless test in Fig. 2, three normal distributions each with their own mean values can be seen. As the distance is increased from 20m up to 700m, the distribution goes from a single normal distribution to three clearly defined normal distributions. Another important feature of the statistical results is that of the minimum RTT of roughly 1.9ms present in the wireless tests. This is due to the TDMA schedule used by the Mikrotik routers, and shows that in this particular set up, processing delays immediately increase the minimum possible latency by roughly 1.4ms, from the 0.5263ms of the wired baseline tests. To generate the latency model for simulation purposes, the results of the 700m wireless test are used. These results are split into four distributions, shown in Table 2, where the fourth distribution (packets with an RTT of above 12.5ms) is clipped from the data set.

Table 1 Latency statistical results from experimental tests.

Test	Mean [ms]	Standard Deviation [ms]	Maximum [ms]	Minimum [ms]
Wired	0.543	0.010	0.698	0.5263
5m	2.917	0.398	8.092	1.9975
20m	2.929	0.445	15.274	1.9748
150m	4.807	2.511	41.688	2.1291
500m	4.933	2.495	47.708	1.9608
700m	5.018	2.787	77.171	2.0012

Table 2 Latency model parameters

Range [ms]	Mean [ms]	Standard Deviation [ms]	Size [%]
0 < x < 4.5	2.911	0.366	55.40
4.5 <= x < 8.5	6.705	0.672	35.39
8.5 <= x < 12.5	10.41	0.788	7.76
12.5 <= x	NA	NA	1.43

Fig. 2. Wireless latency test conducted at 700m

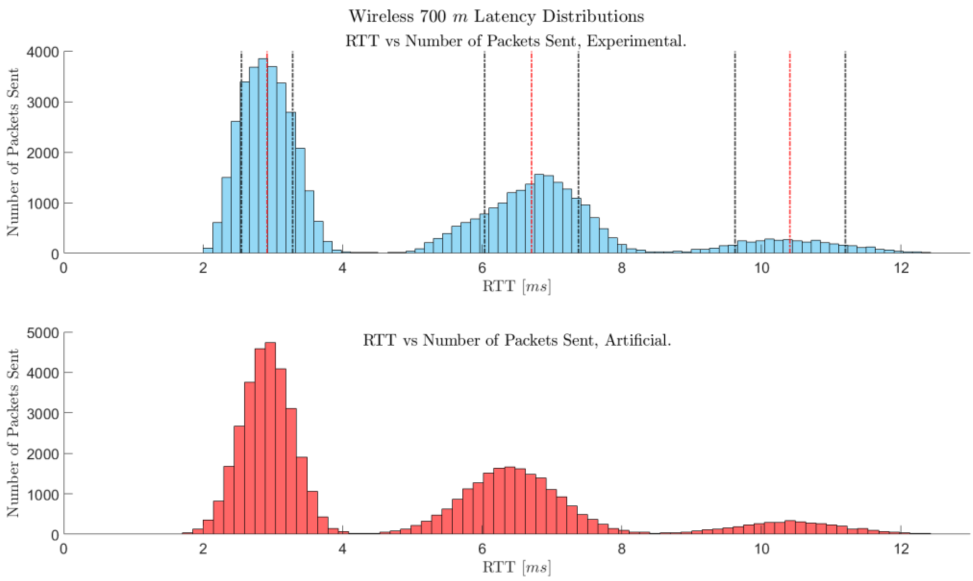

Fig. 3. Comparison of latency model and experimental results

The mean and standard deviation of the remaining distributions are calculated, along with the percentage of packets in each distribution. Using these values, a model is generated that selects a latency randomly from one of the three distributions. Fig. 3 shows a comparison of the clipped wireless 700 m latency results and an artificially generated data set. Although not an exact match, with the experimental distributions not being perfectly normal and the artificial results having a few values below 1.9ms, the latency model was able to accurately reproduce the real-world results.

5. Steering Controller Design: Vehicle Model

A robust controller that is able to handle unknown environmental changes while remaining stable is sought for offsite control. The reason for this is especially in the mining, agricultural and construction environment the conditions experienced by the vehicle can vary significantly and quickly. In mining the weight of the vehicle may increase by up to an order in magnitude between unladen and fully laden. The surface conditions may also change rapidly for all these environment. It is therefore believed that a robust controller would be able to safely control the vehicle through a larger range of uncertainties which are invariably present in a wireless network that is subjected to larger transport latencies and packet drop percentages. In this study, a controller making use of an Auto-Regressive vehicle model with eXongenous (ARX) inputs is formulated. The controller is used in a simulation where it is subjected to transport latencies as well as permanent packet drops. The controller is not chosen as an optimal controller for a system that is subjected to offsite control. However, it is believed that this robustness of the controller resulted in better performance metrics at higher transport latencies and packet drop percentages.

Fig. 4 illustrates the naming convention used when designing the controller, where ψ is the yaw angle of the vehicle measured from the X-axis, δ is the steer angle of the vehicle's front wheels measured from the X'-axis, X and Y are the axes of the global coordinate system, and X' and Y' are the local coordinate axes with zeros at the centre-of-gravity of the vehicle. It is assumed that both front wheels experience the same steer angle during a turn. The yaw angle of the vehicle is used to describe the vehicle's heading with respect to some reference heading.

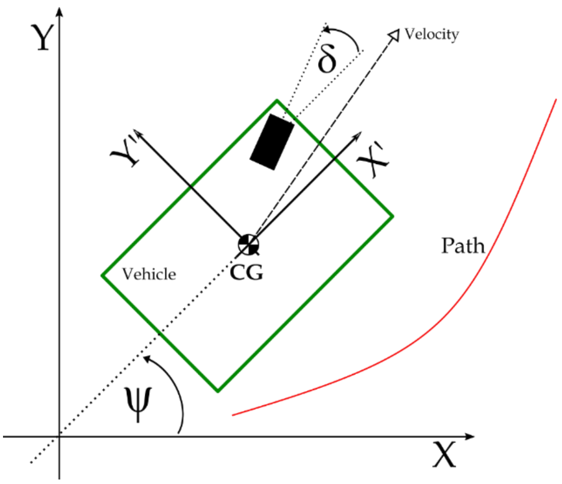

Fig. 4. Vehicle parameter convention showing steer angle and yaw angle of the vehicle

An Auto-Regressive (AR) model is a model in which the future values of a time-varying random processes are determined using a weighted sum of the past values of the process. This method is used to model the plant (vehicle) used by the controller instead of linearizing around set operating points. To expand on the AR-model, the effect of exogenous inputs (external excitation signals) is included, resulting in an ARX-model. It was noted in [24] that the relationship between the steer rate of a vehicle's front wheels and the vehicle's yaw acceleration, and by extension its steer angle and yaw rate, is non-linear and changes based on the level of lateral acceleration experienced by the vehicle. It is this relationship, between the steer angle of a vehicle and its yaw rate, which is used as the time-varying random process to be estimated by the ARX-model.

The general equation of the expanded ARX-model is given in (1), where $\widetilde{Y}_t$ is the estimated value of the model at the current time step, Y_{i-1} are the auto-regressive terms, s is the number of autoregressive terms, i.e. the number of previous time-steps' values to be used in the calculation of the current model estimate, u_{t-i-n} are the exogenous input terms with a time-step delay of n, g is the number of exogenous input terms, φ_i and η_i are weighting variables, and ϵ_t represents the unmodeled residuals that are invariably present in most real processes.

$$\widetilde{Y}_t = \sum_{i=1}^{s} \varphi_{i-1} Y_{i-1} + \sum_{i=0}^{g} \eta_i u_{t-i-n} + \epsilon_t \quad (1)$$

A study performed on the required order of the ARX-model shows that a single order $s = 1$ can sufficiently model the dynamics with any higher order model having a negligible effect. The input dependency of the controller is set as one, i.e. the steer angle of the previous time-step. The resultant transfer function for the model is given in (2), where the unmodeled residuals have been dropped, the numerator represents the exogenous inputs, and the denominator the previous samples

$$H(z) = \frac{\eta_0 z^{-n}}{1 + \varphi_0 z^{-1}} \quad (2)$$

Re-writing (2) in the difference form results in (3), where $\tilde{y}(k)$ represents the estimated yaw rate of the vehicle at the current time-step, $y(k-1)$ represents the measured yaw rate of the vehicle at the previous time-step, $u(k-n) \rightarrow u(k-1)$ represents the steering angle of the vehicle at the previous time-step (n=1 as the input dependency's time-step delay), while φ_0 and η_0 are the weighting parameters on the previous time-step's yaw rate and previous time-step's steer angle respectively

$$\tilde{y}(k) = \varphi_0 y(k-1) + \eta_0 u(k-n) \quad (3)$$

To solve for the weighting variables of the ARX-model, the principle of Linear Least Squares Estimation is used, as presented in [25]. This method requires a set of sample points that are used to estimate the weighting variables of (3). A sample size of 10 time-steps is used at a sampling frequency of 20Hz – a frequency which had already been shown to capture the yaw natural frequency of the vehicle used during experiments [26]. The update rate of the ARX-model, i.e. the rate at which new weighting variables is calculated, is set at 5Hz. Converting (3) to vector form for the 10 samples returns Equation (4), where Y is a vector of measured output values, Φ is the regression vector, and θ is the least squares estimator containing the weighting variables that needed to be solved for.

$$Y = \Phi\theta \quad (4)$$

The solution of the least squares estimator is given by:

$$\theta = (\Phi'\Phi)^{-1}\Phi' Y \quad (5)$$

This ARX vehicle model has the advantage over other vehicle models used in vehicle controller design that no actual vehicle parameters need to be known beforehand and that the model updates quickly if conditions where to suddenly change. This is a requirement for a steering controller to be used on mining, agricultural and construction vehicles.

6. LQR controller

A Linear Quadratic Regulator (LQR) is used as the controller design methodology, where the controller gain K_{LQR} is calculated by optimally solving some chosen cost function [27]. The LQR controller has been used in prior studies for steering controllers [28], [29]. Snider [30] performed an in depth analysis on the standard LQR controller and found that while a well-tuned controller could provide good path following there was no single controller which provided good control over changing vehicle speeds. Ideally, LQR control also requires full-state feedback for optimal control [31], however this is often not possible. In such cases, a plant model can be used to estimate the actual plant's states. In [29] an observer is used to estimate states that cant be measured. In this study the plant is estimated using the ARX-model, resulting in an LQR controller that makes use of an adaptive plant model that changes as the vehicle travels. This solves the problem of full-state feedback, as well as, that the controller now adapts to the changing vehicle conditions and yields good path following at any vehicle speed. For this reason, the controller is aptly named the Linear Quadratic Self-Tuning Regulator (LQSTR). Variants of the LQSTR have been used in prior studies as steering controllers for four wheel steering [32] and autonomous steering controllers [33]. The controllers vary in the plant model used and the method to identify the plant model parameters. The LQSTR methodology was found to give overall better performance than the standard LQR due to it adapting to the changing vehicle dynamics and thus yields a very robust steering controller.

The general LQR block diagram with full-state feedback is shown in Fig. 5, where r represents the reference signal that is to be followed, i.e. the required yaw rate, u is the input to the vehicle system, a required steer angle. A, B, and C are the state-space matrices. K_{LQR} is the calculated LQR gain matrix, with k the time-step. $\bar{N}$ is the pre-compensation gain matrix which is added to reduce steady state errors [34]. y is the output of the system, in this study this is the actual yaw rate. The shaded square, in Fig 5, encloses the open-loop plant that is estimated by the ARX-model.

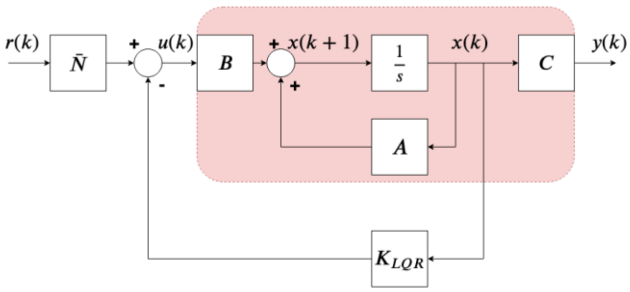

Fig. 5. LQR block diagram for a discrete-time system

To realise the control presented in Fig. 5 requires the solution of two values: the optimal gain K_{LQR}, and the pre-compensation gain $\bar{N}$. To solve for K_{LQR}, the state space method is used, with the state space representation of the ARX-model given in (6), where δ_{k-1} is the control input and in this study is steer input u_{k-1}.

$$X_k = AX_{k-1} + B\delta_{k-1} \quad (6)$$
$$Y_k = CX_k + D\delta_{k-1}$$

Where

$$Xk = [\tilde{y}_k], A = [\varphi_0], B = [\eta_0], C = [1], D = [0],$$
$$\delta_{k-1} = u_{k-1}$$

The chosen cost function is the quadratic performance function with infinite horizon of (7) used in conjunction with the infinite horizons approach. This allows for weighting on both the set-point (x) and control input (u) with the symmetric matrices Q and R [35]. Here $x(\tau)$ represents the error between the desired yaw rate and the actual yaw rate of the vehicle, while $u(\tau)$ represents the steer angle of the vehicle at the previous time step.

$$J_{(LQR)} = \int_t^T [x'(\tau)Q(\tau)x(\tau) + u'(\tau)Ru(\tau)]d\tau \quad (7)$$

The optimal feedback control law that minimises J_{LQR} is given by (8) with $\delta_{required}$ the input to the plant/vehicle ($u(k)$) of Fig. 5), and $\dot{\psi}_{required}$ the required yaw rate - the calculation of which is presented in (8).

$$\delta_{required} = \dot{\psi}_{required}\overline{N} - K_{LQR}X_{k-1} \quad (8)$$

The optimal gain matrix K_{LQR} is calculated using:

$$K_{LQR} = R_{LQR}^{-1}B^T P \quad (9)$$

Where P is solved for using the discrete dynamic algebraic Ricatti Equation. Since the state space formulation no longer contains any matrices, the simplified iterative solution of P is used, given in (10), where the lower-case constants correspond to their upper-case matrices presented thus far. This simplification, from matrices to constants resulting from the reduction in model order of the ARX-model, removes any matrix inversions – greatly reducing the complexity as well as the required iteration count. A limit of 1000 iterations is still imposed on the maximum allowable iterations.

$$p_{k+1} = q + a^2 p_k - \frac{(apb)^2}{r + b^2 p_k} \quad (10)$$

The calculation of $\overline{N}$ follows the steps presented in [36], [37], where for a discrete-time system the controller designer chooses a constant value. A constant $\overline{N}$ value based on the speed of the vehicle is used, shown in (11) where V is the speed of the vehicle in km/h.

$$\overline{N} = -\frac{0.3}{90}V + 1.3 \quad (11)$$

The required yaw rate is determined from an overall heading error Δψ which is to be minimized Δt in the future. From finite differences the required yaw rate can be determined as:

$$\dot{\psi}_{required} = \Delta\psi\Delta t \quad (12)$$

The overall heading error is a combination of two errors: the heading error and the lateral error. The heading error is the difference between the vehicle's actual current heading and the vehicle's required heading at some time-step Δt in the future. The heading of the path is obtained by projecting the vehicle position forward based on the time-step and current vehicle speed. The lateral error is obtained from projecting a point ahead of the vehicle. The projection point may have the same Δt as that used by the heading error or be different, and is used to determine the angle between the projected point and the nearest point on the required path, with the vehicle being treated as the origin. The overall heading error $\Delta\psi$ is then a summation of the heading and lateral errors.

The computational efficiency of the controller was tested on an ARM Cortex-A53, 1.2GHz processor and found the computing time for a single iteration of the ARX and LQR takes $138\mu s$. The LQSTR controller has not been tested with delays, as almost no prior study into controller delays haven been performed on any steering controller. However, the LQSTR is a robust controller design which should still provide robust steering with delays in the system.

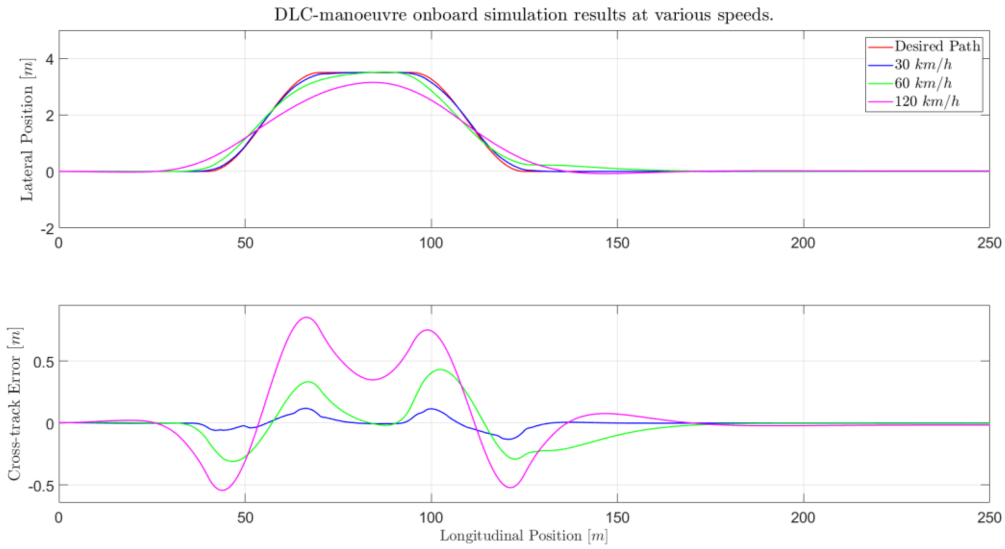

Fig. 6. ISO 3888-1 Double Lance Change of controller without delay

7. Simulation Study

The simulation model is a 16 degree-of-freedom non-linear model, developed by [38]. The MSC.Software ADAMS environment is used to simulate the multi-body dynamics of the Land Rover platform using experimentally obtained vehicle parameter values. The Pacejka '89 tyre model of [39] is used to model the tyre road interface. The self-aligning moment as well as the longitudinal characteristics of the tyre model are excluded. The Adams model is linked with Simulink, which allows for efficient solving of the non-linear suspension components. Botha [26] carried out a full vehicle model validation study, where it was concluded that there is a good correlation between the platform and the model.

The controller baseline stability, without any transport latencies, is determined using the ISO 3888-1 Double-Lane-Change (DLC) [40]. The ISO 3888-1 DLC represents a severe accident avoidance manoeuvre. The manoeuvre is designed to never allow the vehicle to obtain a steady state cornering behaviour. Fig. 6 shows simulation results for the controller conducting the DLC at various speeds, under the assumption that the controller is on-board the vehicle i.e. no transport latencies and no dropped packets. The model remains stable at speeds in excess of 120km/h, even though the vehicle is unsuccessful in navigating the DLC – due to the vehicle cutting corners. At such high speeds it is physically impossible for the vehicle to successfully navigate the manoeuvre. The controller is thus very robust and stable at high vehicle speeds, while maintaining a relatively high path following accuracy at lower speeds.The effect of transport latencies is determined by incorporating latencies into the simulation environment. Fig. 7 shows in a broader sense where the latencies are present during experimentation, and subsequently where they are simulated. During simulations, latencies are added on both transport latency paths of Fig. 6, while during experiments all the artificial latencies are lumped onto the path between the base station and the vehicle as the source of latency does not matter, only its overall magnitude.

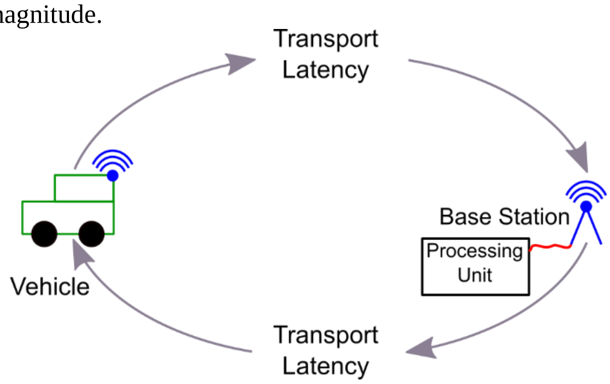

Fig 7. Latency setup used in experiment and simulation

Simulations are conducted for a vehicle performing the DLC-manoeuvre at 60km/h with various latencies and probabilities of packet drops. A vehicle speed of 60km/h is chosen due to it being the maximum speed on most surface mines as well as the speed limit for most urban driving.

Fig. 8 and Fig. 9 show the results of the wireless simulations as a summary of RMSE on the cross-track error for transport latencies and a constant packet drop percentage. The latency values of Fig. 8 represent the total latency that a packet is withheld for, half from vehicle to base station (controller) and half from base station back to vehicle. Added to each of these halves is a latency amount drawn from the latency model which, in general, is small compared to the added delays but adds some variability in the delays as well as a minimum latency. A certain percentage of the packets sent in either direction, from the vehicle to the controller or the controller back to the vehicle, are dropped independently based on a constant probability of dropping packets.

Fig. 9 shows the vehicle path at different latencies from the baseline 0ms latency to a maximum of 100ms, and 0% probability of dropping packets. The figure shows that with a 0% probability of dropping packets and latencies of

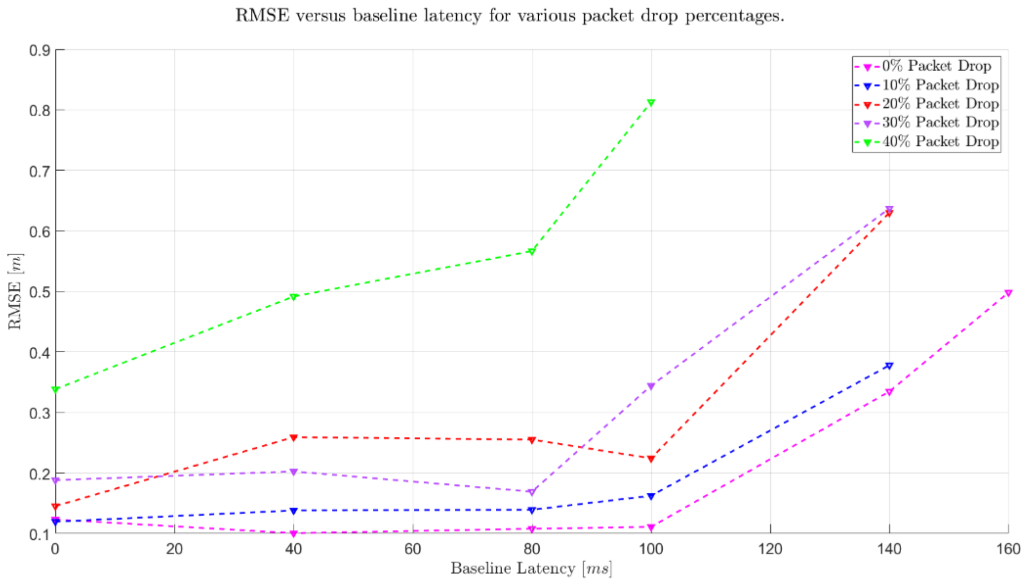

Fig. 8. Effect of latency and packet drop on the cross-track error of the vehicle performing a DLC at 60km/h

40- and 100ms the vehicle maintains a stable path following ability. At slightly higher latencies with 0% packet drop probability the vehicle becomes unstable. From Fig. 8 it can be seen that if the probability of dropped packets is reduced then higher latencies are possible before the vehicle becomes unstable. Although higher baseline latencies are simulated without total loss of control, the resultant RMSE values are deemed inadequate to be considered reliable control.

When comparing Fig. 8 and Fig. 9, it is noted that an increase in the baseline latency results in a more consistent increase in the oscillatory behaviour of the vehicle, while an increase in the packet drop percentage results in more unpredictable behaviour from the vehicle. This consistent increase in oscillatory behaviour, for the increasing baseline latency cases, continues until the oscillations are large enough to cause lateral accelerations beyond the handling limit of the vehicle. This occurs at baseline latencies of 200ms with a 0% packet drop percentage. Due to the random nature of the increasing packet drop percentage cases, a predictable output becomes unlikely. This is highlighted by the 40% packet drop percentage plot of Fig. 8, where there is a sudden increase in the cross-track error between 30% and 40% packet drop percentages. This is most likely due to a batch of packets being dropped in succession. If packets are dropped with a completely uniform distribution a more predictable pattern, such as the increasing baseline latency simulations, may be present. Unfortunately, packet drops in networks are generally not uniform, but rather clumped together.

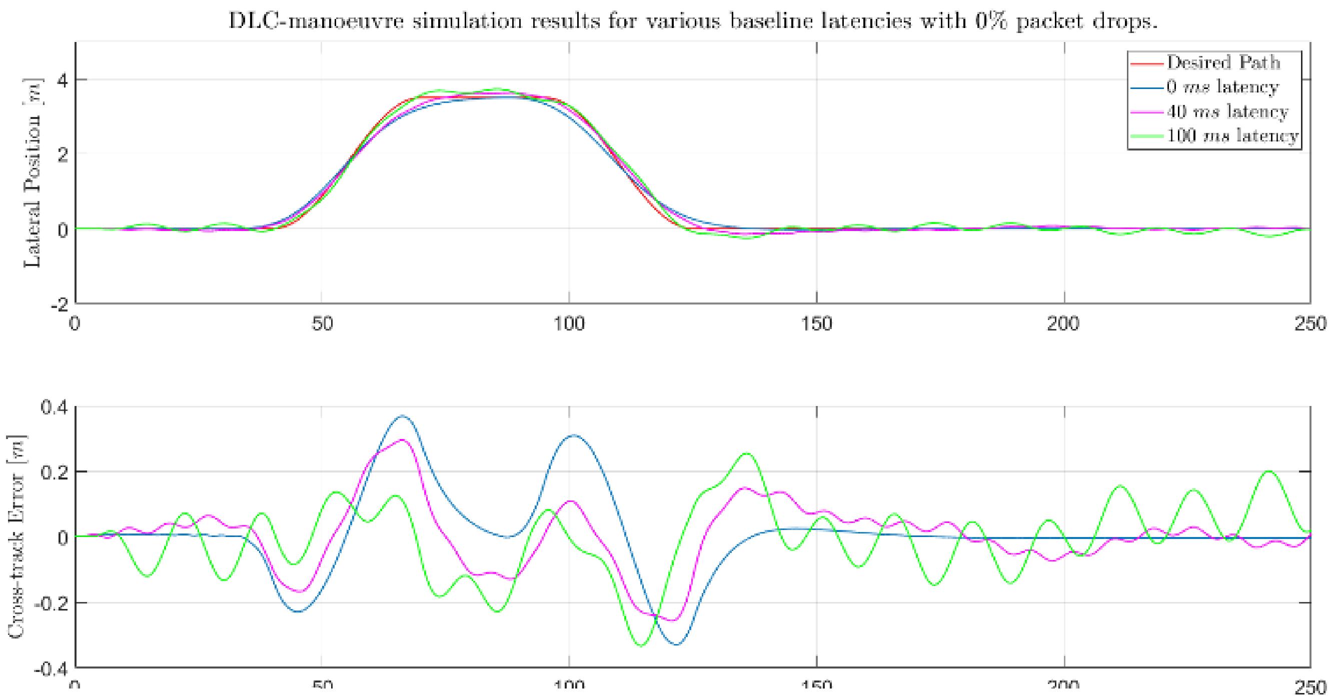

Fig. 9. Vehicle path and cross-track error through DLC at different latencies

8. Experimental Results

To verify the accuracy of the simulation the same DLC-manoeuvre is performed experimentally at the same speed as in simulations. Positioning information is provided by a NovAtel SPAN-CPT GNSS+INS system providing positioning data at 20Hz. Control is achieved using a dSPACE MicroAutobox II and a stepper motor connected to the vehicle steering link. Data is recorded on the vehicle using the dSPACE MicroAutobox II and history of data, as required by the steering controller, is sent to a remote base station which consists of real time embedded PC and Mikrotik Nv2 wireless router. The embedded system computes the required steering input and sends it back to the vehicle. The vehicle and base station setup is shown in Fig 10. The figure shows the vehicle antenna attached to the roof and the base station antenna raised to the same height as the vehicle antenna to provide good line of sight in most instances. Tests were conducted at the Gerotek test facilities long straight which consists of a narrow long straight which opens up into a section 25m in width where the actual DLC occurs. The base station situated in the area where the actual DLC takes place to improve communication fidelity. The vehicle however is controlled up to a distance of 500m away when it accelerates to build up speed.

Fig. 10. Experimental test setup showing communication antennas and remote station

The wireless setup is very similar to the simulation set-up with a wireless link between a remote base station and the vehicle. Delays are added in one direction, which represent the total delay in control. A required vehicle path is recorded with a human driver controlling the vehicle. The path following results for different latencies are shown in Fig. 11 where it is evident that good path following is achieved for the experimental runs exposed to 50ms or less of round-trip delay. A more oscillatory steering response is also observed indicating that the effect of the induced delays may be more detrimental than shown in simulation. The vehicle also becomes unstable at lower latency values during experiments most likely due to the added sensor noise, the low positioning sampling rate of 20Hz, as well as any added steering controller delay. In experiments a 50ms delay results in satisfactory path following. This delay is relatively large when comparing it to the results of the study conducted to generate the latency model, with delays generally less than 10ms experienced up to distances of 700m.

The performance of the offsite control of a vehicle could be improved by having a controller which can compensate for the delay. The delay can be estimated in real time and measures can be taken to reduce it by using special observers [40]. While the results of this study show successful direct control of a vehicle using a remote base station, better results may be possible by splitting the path planning aspects and the control aspects between the base station and the vehicle. If the base station is solely responsible for path planning and sending optimised path information to the vehicles, while path following is performed exclusively on the vehicle. This could provide better results provided that the path planning information does not change rapidly. The focus of this study, however, was to determine the effect of delays on direct control of the vehicle, neglecting any path planning aspects.

The effect of packet loss was not experimentally tested due to the uncertainty of the controller, which posed a safety risk during testing. Whereas with latency the steering becomes more unstable as vehicle speed increases and the oscillations build slowly through the manoeuvre

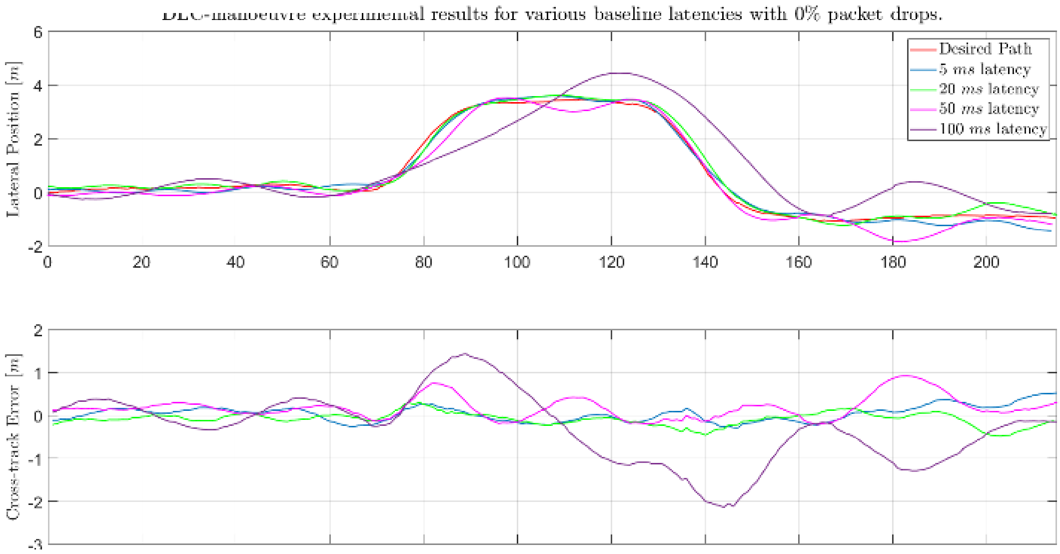

Fig. 11. Experimental results for a vehicle performing a wirelessly controlled DLC at 60 km/h

9. Conclusion

In this study, a hardware-in-the-loop wireless control system is developed that is able to achieve good path following through a severe double lane change manoeuvre while being controlled wirelessly via a remote base station. The goal of this study is to provide insight into the effect of communication delays and packet losses on a supervisory roadside system that would be able to mitigate collisions by taking control of a vehicle and performing an emergency steering manoeuvre on vehicles directly.

The simulation and experimental path-following ability of the system exposed to various magnitudes of communication delay and packet loss probabilities is evaluated in simulation using a validated multibody dynamics model of the Land Rover test vehicle. An emergency steering manoeuvre is simulated by the ISO 3888-1 double lane change both in simulation and experimentally.

The simulation model proved to be a good representation of the experimental system while performing the DLC-manoeuvre at approximately 60 km/h under the effect of roughly a 50ms round-trip communication delay. The simulation is not able to effectively predict the vehicle response at higher levels of delay as instabilities are proven to occur much earlier during physical experiments. This could be a result of a combination of factors such as noise in transducer measurements, delay in the steering system, and a low GPS update rate, all of which can negatively affect accuracy and bandwidth of the system. When adding larger communication delays the impact of these factors may be exaggerated.

A 50ms communication delay is a large delay by modern communication standards and even more so when considering that the measured typical delay was below 10ms when using the proprietary Nv2 protocol of Mikrotik. The simulation study also suggested that relatively accurate path following is maintained when in excess of 30% of the data packets are lost. It can thus be concluded that a vehicle controlled from a remote base station will be able to complete an accurate emergency steering manoeuvre with current technology while making use of the proposed LQSTR controller. Better performance may be possible with a controller that is aware of, while being able to quantify, the communication delays that are present in real time, thereby limiting their effect, something that should be investigated further. The study still provides an indication of how the performance of a steering controller will be affected by transport latencies as well dropped packages. Another possible future improvement could be to assign path planning to the remote base station, while control is performed entirely by the vehicle.

10. References

[1] 'National Highway Traffic Safety Administration, *NHTSA',* https://www.nhtsa.gov/press-releases/us-dot-and-iihs-announce-historic-commitment-20-automakers-make-automatic-emergency, accessed 27 March 2017.

[2] Sasaki, T., Miyata, T. and Kawashima, K., 'Development of remote control system of construction machinery using pneumatic robot arm', IEEE/RSJ

[3] Fenton, R.E.,'IVHS/AHS: Driving into the future', IEEE Control Systems Magazine, 1994, 14(6), pp.13-20.

4] Colombo, A. and Del Vecchio, D., 'Supervisory control of differentially flat systems based on abstraction', 50th IEEE Conference on Decision and Control and European Control Conference, 2011, pp. 6134-6139.

[5] Colombo, A. and Del Vecchio, D., 'Least restrictive supervisors for intersection collision avoidance: A scheduling approach', IEEE Transactions on Automatic Control, 2014, 60(6), pp.1515-1527.

[6] Eidehall, A., Pohl, J., Gustafsson, F. and Ekmark, J., 'Toward autonomous collision avoidance by steering', IEEE Transactions on Intelligent Transportation Systems, 2007, 8(1), pp.84-94.
International Conference on Intelligent Robots and Systems (IROS), 2004, Vol. 1, pp. 748-753

[7] Schreiber, V. Savitski, D, Augsburg, K. and Shyr, C., 'Advanced Braking System Control Prototyping Using Networked Hardware-In-The-Loop Technique', Proceedings of Eurobrake, Milan, Italy, 2016.

[8] Ersal, T., Stein, J. L., Brudnak, M., Filipi, Z., Salvi, A. and Fathy, H. K., 'Development Of An Internet-Distributed Hardware-In-The-Loop Simulation', Proceedings of the ASME Dynamic Systems and Control Conference, 2009, Hollywood, California, USA, 2009.

[9] Kloc, M., Weigel, R. and Koelpin, A., 'Making Real-Time Hardware-in-the-Loop Testing of Automotive Electronic Control Units Wireless', International Conference on Advanced Technologies for Communications (ATC), Hanoi, Vietnam, 2016.

[10] Franchi, N., Fischer, G. and Wiegel, R., 'Radio hardware in-the-loop emulation for testing vehicular communication systems', IEEE 5th International Symposium on Wireless Vehicular Communications (WiVeC), Dresden, Germany, 2013.

[11] Tandon, A., Brudnak, M. J., Stein, J. L. and Ersal, T., 'An Observer Based Framework to Improve Fidelity in Internet-Distributed Hardware-in-the-Loop Simulations', ASME Dynamic Systems and Control Conference, Palo Alto, California, USA, 2013.

[12] 'Delays in Control Systems', http://users.ece.utexas.edu/~buckman/H3.pdf. ,accessed 22 May 2017

[13] Lawrence, D. A., 'Stability and Transparency in Bilateral Teleoperation', IEEE Transactions in Robotics and Automation, 1993, vol. 9, no. 5, pp. 624-637

[14] Ersal, T., Brudnak, M., Stein, J. L. and Fathy, H. K., 'Statistical Transparency Analysis in Internet-Distributed Hardware-in-the-Loop Simulation', IEEE/ASME Transactions in Mechatronics, 2012, vol. 17, no. 2, pp. 228-238.

[15] Jiang, D. and Delgrossi, L., 'IEEE 802.11 p: Towards an international standard for wireless access in vehicular environments', Vehicular Technology Conference. VTC Spring, 2008, pages 2036-2040.

[16] Khairnar, V. D. and Kotecha, K., 2013, 'Performance of vehicle-to-vehicle communication using IEEE 802.11 p in vehicular ad-hoc network environment', arXiv preprintarXiv:1304.3357 2013

[17] Sjoberg, K., Uhlemann, E., and Strom, E. G., 'How severe is the hidden terminal problem in vanets when using csma and stdma', Vehicular Technology Conference (VTC Fall), 2011, pages 1-5. IEEE.

[18] 'N-Streme and Nv2',: https://mum.mikrotik.com//presentations/ IN12/soumil.pdf. accessed 18 November 2018.

[19] Dey, K.C., Rayamajhi, A., Chowdhury, M., Bhavsar, P. and Martin, J., 'Vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication in a heterogeneous wireless network–Performance evaluation', Transportation Research Part C: Emerging Technologies, 2016, 68, pp.168-184.

[20] Cunha, F., Villas, L., Boukerche, A., Maia, G., Viana, A., Mini, R.A. and Loureiro, A.A., 'Data communication in VANETs: Protocols, applications and challenges', Ad Hoc Networks, 2016, 44, pp.90-103.

[21] Ye, F., Adams, M. and Roy, S., 'V2V wireless communication protocol for rear-end collision avoidance on highways', In ICC Workshops-2008 IEEE International Conference on Communications Workshops, 2008, pp. 375-379.

[22] Kavitha, K.V.N., Bagubali, A. and Shalini, L., 'V2V wireless communication protocol for rear-end collision avoidance on highways with stringent propagation delay', International Conference on Advances in Recent Technologies in Communication and Computing, 2009, pp. 661-663

[23] Lopez-Aguilera, E., Casademont, J., and Cotrina, J., 'Propagation delay influence in IEEE 802.11 outdoor networks', Wireless networks, 2010, 16(4):1123-1142.

[24] Thoresson, M. J., Botha, T. R., and Els, P. S., 'The relationship between vehicle yaw acceleration response and steering velocity for steering control', International Journal of Vehicle Design, 2014, 64(2-4) pp.195-213.

[25] Van de Geer, S.A., 'High-dimensional generalized linear models and the lasso', The Annals of Statistics, 2008, 36(2):614-645.

[26] Botha, T.R., 'High Speed Autonomous Off-Road Vehicle Steering', M-dissertation, 2011 http://repository.up.ac.za/handle/2263/29665

[27] Triantafyllou, M. S. and Hover, F. S., 'Maneuvering and control of marine vehicles', Massachusetts of Institute of Technology. 2003, chapter 19, pp. 92 98

[28] Soudbakhsh, Damoon, and Azim Eskandarian. *A collision avoidance steering controller using linear quadratic regulator*. No. 2010-01-0459. SAE Technical Paper, 2010.

[29]Snider, Jarrod M. "Automatic steering methods for autonomous automobile path tracking." *Robotics Institute, Pittsburgh, PA, Tech. Rep. CMU-RITR-09-08* (2009).

[30] Hangos, Katalin M., József Bokor, and Gábor Szederkényi. *Analysis and control of nonlinear process systems*. Springer Science & Business Media, 2006.

[31] Shaout, A., M. A. Jarrah, H. Al-Araji, and K. Al-Tell. "A nonlinear optimal four wheels steering controller." In *Proceedings of the 43rd IEEE Midwest Symposium on Circuits and Systems (Cat. No. CH37144)*, vol. 3, pp. 1426-1429. IEEE, 2000.

[32] Alcala, Eugenio, Vicenç Puig, Joseba Quevedo, Teresa Escobet, and Ramon Comasolivas. "Autonomous vehicle control using a kinematic Lyapunov-based technique with LQR-LMI tuning." *Control Engineering Practice* 73 (2018): 1-12.

[33] ERKOL, Hüseyin Oktay. "Linear quadratic regulator design for position control of an inverted pendulum by grey wolf optimizer." *system* 9, no. 4 (2018).

[34] Friedland, Bernard. Control system design: an introduction to state-space methods. Courier Corporation, 2012.

[28] Usta, M.A., Akyazi, Ö. and Akpinar, A.S., 'Aircraft roll control system using LQR and fuzzy logic controller', In 2011 International Symposium on Innovations in Intelligent Systems and Applications. June, 2011, pp. 223-227.

[29] Nair, V.G., Dileep, M.V. and George, V.I. 'Aircraft yaw control system using LQR and fuzzy logic controller', International journal of computer applications, 2012, 45(9), pp.25-30.

[30] Uys, P. E., Els, P. S. and Thoresson, M. J., 'Suspension settings for optimal ride comfort of off-road vehicles travelling on roads with different roughness and speeds', Journal of Terramechanics, 2007, vol. 44, pp. 163-175.

[31] Bakker, E., Pacejka, H. B., and Lidner, L., 'A new tire model with an application in vehicle dynamics studies', SAE transactions, 1989, pp. 101-113.

[32] ISO, 'ISO3888-1: Passenger cars - Test track for a severe lane-change manoeuvre - Part 1: Double lane-change', The International Organisation for Standardisation, ISO, 1999

[33] Tandon, A., Brudnak, M. J., Stein, J. L. and Ersal, T., 'An Observer Based Framework to Improve Fidelity in Internet-Distributed Hardware-in-the-Loop Simulations', ASME Dynamic Systems and Control Conference, Palo Alto, California, USA, 2013.