Impact of packet loss on CACC string stability performance

Impact of Packet Loss on CACC String Stability Performance

C. Lei, M. van Eenennaam, W. Klein Wolterink, G. Karagiannis, G. Heijenk
Department of Computer Science
University of Twente
Enschede, The Netherlands

J. Ploeg
Integrated Vehicle Safety Department
TNO Technical Sciences
Helmond, The Netherlands

Abstract— **Recent development in wireless technology enables communication between vehicles. The concept of Co-operative Adaptive Cruise Control (CACC) - which uses wireless communication between vehicles - aims at string stable behaviour in a platoon of vehicles. "String stability" means any non-zero position, speed, and acceleration errors of an individual vehicle in a string do not amplify when they propagate upstream. In this paper, we will discuss the string stability of CACC and evaluate its performance with various packet loss ratios, beacon sending frequencies and time headway in simulations. The simulation framework is built up with a controller prototype, a traffic simulator, and a network simulator.**

Keywords-CACC; string stability; packet loss ratio; beacon sending frequency; time headway

I. Introduction

In Vehicular Ad hoc Networks (VANETs), on-board units (OBUs) give vehicles the ability of communication to make them "smart objects" more than mere transportation tools. VANET comprises (1) the communication between vehicles and (2) the communication between vehicles and road side units (RSUs) using the same ad hoc wireless technology, such as IEEE 802.11p [1]. An RSU can be a base station or other fixed infrastructure that is located at the road side. By using VANETs, many new services for vehicles are enabled and numerous opportunities for safety improvements are created.

Traffic congestion is a growing problem in industrialised nations worldwide. Using the concept of Adaptive Cruise Control (ACC), which has a positive impact on traffic safety and efficiency [2], can be a partial solution to this problem. By extending a Cruise Control system with a radar sensor, ACC allows a vehicle to maintain a pre-set speed, as well as to adapt its speed to the speed of its predecessor in order to keep a minimum distance from its predecessor [3].

However, ACC does not sufficiently improve the string stability. String stability represents any non-zero position, speed, and acceleration errors of an individual vehicle in a string of vehicles, do not amplify when they propagate upstream, e.g., see [10, 11]. As a result, at moderate traffic density, small disturbances may lead to traffic jams, negatively impacting a road's capacity. An enhancement on the ACC concept is the Co-operative ACC (CACC), where the OBU in a vehicle is using a communication medium to communicate with OBUs available in other vehicles or RSUs. The communicated information may include a vehicles' position, speed, acceleration, etc., which can be used to enhance the performance of the current ACC systems.

It is expected that CACC will increase vehicle traffic efficiency and traffic flow stability [2, 4]. CACC can be applied in traffic applications such as co-operative following [4, 5], or vehicle platooning [5, 6]. An implementation of CACC can be found in [7].

The main goal of this paper is to evaluate the impact of CACC on the string stability performance by using simulation experiments. The research questions that are answered by this paper in order to satisfy this goal are:

- How is the string stability evaluated?
- Which simulation environment can be used to evaluate the impact of CACC on the string stability performance?
- What is the impact of packet loss on the string stability performance of a CACC system?

In Section II we will briefly introduce the control theory of CACC and illustrate the concept of string stability. Then in Section III we will describe our simulation environment. The simulations, corresponding results and analysis will be shown and discussed in Section IV. Finally, in Section V, we will conclude this paper and give recommendations for future work.

II. Control Theory and String Stability

This section describes two main concepts used in this research work, which are (1) the control theory used by the applied adaptive cruise control mechanism and (2) the concept of string stability.

A. Control Theory

Control theory [8] deals with the behaviour of dynamical systems. The control objective is to realize a desired distance to the preceding vehicle. This desired distance may be an increasing function of vehicle velocity in order to take safety aspects into account. The result is commonly referred to as a "constant time-headway spacing policy". In order to realize the

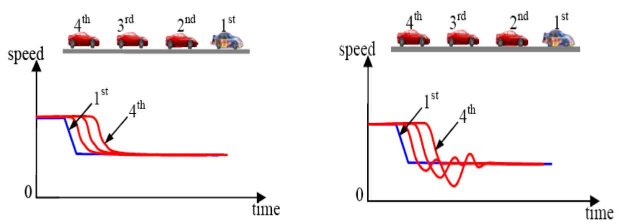
This work is supported by the Dutch NL Agency/HTAS (High Tech Automotive Systems) Project Connect&Drive, Project no. HTASD08002.

control objective, the control system acts on the desired acceleration of the vehicle by means of actively influencing the drive force, based on radar measurements and (in case of CACC) on data obtained through wireless communications. The controller's main task is to reject disturbances caused by velocity variations of the preceding vehicles. In our work, the disturbance is caused by the behaviour of other traffic, such as sudden deceleration of preceding vehicles. An ideal feedback control system should be able to cancel out all errors, effectively mitigating the effects of any forces that might or might not arise during operation and producing a response in the system that perfectly matches the designer's wishes. In reality, this might be difficult to achieve when taking measurement errors in the sensors, delays in the controller, and imperfections in the control input into consideration.

Though many solutions exist to implement a CACC controller, we will focus on a control structure that can be applied in an ad-hoc vehicle platoon scenario, see [7]. In this scenario, the concept of a platoon leader is not supported and all the vehicles in a platoon support the same type of one-vehicle-look-ahead CACC controller topology. The main reason of choosing this CACC controller structure is the fact that the one-vehicle-look-ahead topology is the simplest possible structure and therefore it has the highest probability of being deployed. Furthermore, this CACC controller structure has been developed within the Connect &Drive [9] project.

B. String Stability

The term “string stability” is often used interchangeably with “platoon stability” in this field, which means any non-zero position, speed, and acceleration errors of an individual vehicle in a string do not amplify when they propagate upstream, e.g., see [10, 11].



(a) (b)

Figure 1. Platoon stability: (a) stable (b) unstable, copied from [10])

According to [19], the vehicle speed should be taken as a basis for string stability, which is more relevant than distance error in view of traffic analysis.

A simple scenario which can be used to explain string stability is illustrated in Fig. 1 (a), (b).

In Fig. 1 (a) (b), a string of four vehicles moving from left to right is shown. The leading vehicle is denoted as 1st while the last vehicle is denoted as 4th. In each of these figures, below the shown string of vehicles, a speed vs. time coordinate graph for each of the 4 vehicles is shown. As time goes by, the leading vehicle decelerates linearly and we can see different response of the following vehicles in the platoon depending on whether the platoon is string stable or not.

In Fig. 1 (a), the situation is shown where the platoon is string stable: the deceleration of the leading vehicle is not amplified through the following vehicles and the deceleration of following vehicles’ is smooth without any fluctuation of the speed. In Fig. 1 (b), the platoon is considered not string stable (string unstable): the following vehicles decelerate even more than the leading vehicle. Though finally, the speeds of the following vehicles approach to the leading vehicle’s speed, their speeds fluctuate significantly. Actually, during the period of fluctuation, the distance between neighbouring vehicles also fluctuates, as a result, collisions between vehicles are more likely to happen, and in other words, safety is worse.

String stability can be improved if the information of the preceding vehicle is used in the feedback loop, and the information of the preceding vehicle can be collected by a low latency communication medium [7]. The most distinctive difference between ACC and CACC is that besides the preceding vehicle's speed and position used as inputs in ACC, the acceleration of the preceding vehicle transmitted through the wireless channel is also adopted as input in CACC, see Fig. 2 and [7]. Therefore, CACC is treated as a solution to achieve a desired following distance with string stability.

III. SIMULATION MODEL

The used simulation environments and models include: (1) the vehicle behaviour (the controller prototype), including the ACC and CACC models, which have been implemented using SIMULINK [20]; (2) the mobility behaviour of vehicles, which has been modelled using SUMO (Simulation of Urban Mobility) [12]; (3) the communication networking behaviour, which has been modelled using OMNeT++ [13] together with its MiXiM (a MiXed siMulator) Framework extension.

A. SIMULINK Model

In the “Car” part of Fig. 2, the module “(C)ACC Controller” together with the module “Vehicle” provides the prototype of CACC and ACC controllers. At the beginning of each simulation step, the module “(C)ACC Controller” reads relative speed and distance to the preceding vehicle from the “Radar” module. The host vehicle’s acceleration and speed are read from the module “sensor” as inputs. In addition, CACC would read the acceleration of the preceding vehicle from the “Wireless Medium” by Wi-Fi interface, which is not necessary for ACC. The desired time headway, desired distance at standstill and cruise speed are pre-set before the simulation starts. The control objective is to realize a desired distance, taking into account a pre-defined maximum speed, referred to as the cruise speed. Note that the cruise speed is a maximum speed when the vehicle operates in (C)ACC mode. If there is no target vehicle, the system switches to a cruise control mode, in which case the cruise speed becomes the target speed. The time headway is the time it takes for vehicle “i” to reach the current position of its preceding vehicle “i−1” when continuing to drive with a constant velocity [7]. The primary control objective is to follow the preceding vehicle at a desired distance D(t):

$$D(t) = D_{standstill} + h*V(t) \quad (1)$$

Here, "$D_{standstill}$" denotes the desired distance to the preceding vehicle at standstill; "h" denotes the desired time headway and "V(t)" is the current vehicle velocity.

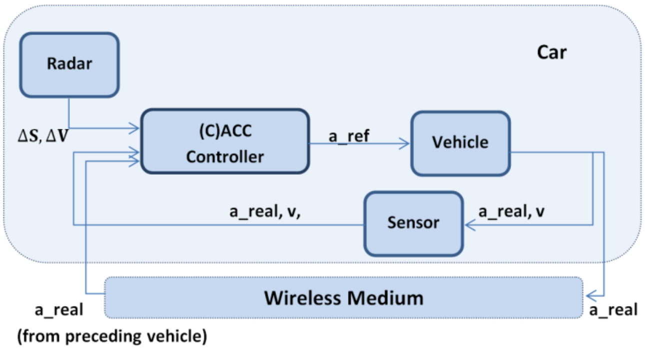


Figure 2. A vehicle's control system

Based on these inputs, the CACC/ACC controller can calculate a reference acceleration "a_ref". The "Vehicle" module mimics the response of a real vehicle which will revise this reference acceleration generated by the CACC/ACC controller to a resulted acceleration "a_real". Actually, "a_real" is the acceleration value to be used by the "Car". Then, still in the "Vehicle" module, this resulted acceleration will be translated into resulted velocity "v" and position "s" and coupled to the "Sensor" block so that "a_real" and "v" can be read in the next simulation time step. Meanwhile, the resulted acceleration will be transmitted on the "Wireless Medium". Note that when due to impairments of the wireless communication medium a packet loss occurs and an updated acceleration value is lost, then the CACC controller uses a previously received and stored acceleration value.

B. SUMO Model

Figure 3, shows a part of the generated road network and a platoon of 10 vehicles. We mark each vehicle with an ID, where the leading vehicle's ID is "veh0", that of the first following vehicle is "veh1" and the last vehicle's ID is "veh9". The leading vehicle moves from left to right (the downstream direction) and the other 9 vehicles equipped with ACC/CACC controllers follow the leading vehicle.

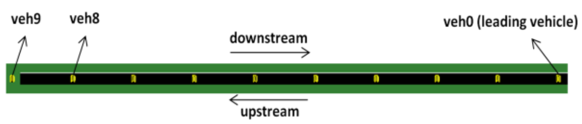


Figure 3. SUMO traffic model

C. MiXiM/OMNeT++ Model

MiXiM/OMNeT++ is used to simulate the wireless communication between vehicles in a platoon. Vehicles are simulated in the form of communication nodes, which have the same positions as the vehicles and are able to exchange information by sending beacons. In this way, every vehicle (communication node) can get its preceding vehicle's acceleration.

The MiXiM/OMNeT++ model applies a cooperative awareness mechanism using beaconing, see [15]. The beaconing procedure is using a simple timer, which means that a node transmits a beacon every τ seconds, with a small, randomly chosen variation or offset. By tuning the value of τ, the beacon sending rate/frequency can be varied. The MAC and Physical layers used in the MiXiM/OMNeT model are based on the IEEE 802.11p standard. The model used in this research was realized by modifying the currently available IEEE 802.11 MiXiM example, i.e., Mac80211, such that it could operate as an 802.11p model.

D. Complete Simulation Model

The complete experiment structure can be seen in Fig. 4 (a), while the corresponding simulation model is shown in Fig. 4 (b):

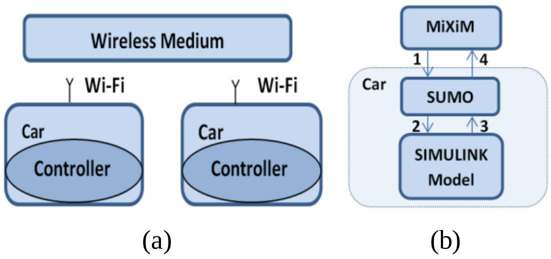


Figure 4. Experiment structure in (a) reality and (b) simulation

As shown in Fig. 4 (a), corresponding to Fig. 2, "Cars" equipped with "Controllers" (CACC) communicate with each other with their "Wi-Fi" interfaces. In the simulation structure shown in Fig. 4 (b), "Cars" are simulated by the SUMO model and "Controllers" are originally built in the form of a SIMULINK model. MiXiM simulates the wireless transmission.

In order to allow the SIMULINK model to be used by vehicles implemented in SUMO, it is first converted into a C++ shared library by using the Real-Time Workshop tool in SIMULINK so that it can be called in the source code of SUMO.

In this work, the method described in [16, 17] is used for bidirectional coupling between OMNeT++/MiXiM and SUMO, where they communicate with each other through a traffic control interface (TraCI) by transmitting TCP messages with OMNeT++/MiXiM acting as the TraCI client and SUMO acting as the TraCI server.

For the simulation of CACC, four steps are noted in Fig. 4 (b): (1) at the beginning of each simulation time step, MiXiM sends the information received from other communicating nodes (i.e. preceding vehicle's acceleration) to SUMO. This information is collected by each communication node from the latest received beacon sent by its preceding vehicle; (2) in the SUMO part, this received acceleration from MiXiM and the other parameters are used as inputs for the CACC controller for each vehicle as described in section III to calculate a real acceleration and velocity ("a_real" and "v" in Fig. 2); (3) the resulting velocity and position are used to simulate the movement of vehicles in SUMO; (4) after moving the vehicles, SUMO will send a trace back to MiXiM which comprises the vehicles' acceleration, velocity and position generated by the CACC controller and MiXiM moves its communication nodes according to the vehicles' position information from SUMO, followed by the transmission of a beacon by each communication node. Note that the received information is buffered before the start of the next simulation time-step.

For ACC, we just need the SUMO model and the shared library converted from the SIMULINK model. Details of our simulation environments can be found in [18].

IV. Experiments, Results and Analysis

In order to investigate the impact of ACC and CACC on the string stability performance, we have done a set of simulations. By observing the speed and acceleration of following vehicles it can be investigated whether the disturbance of the leading vehicle is amplified upstream through the platoon, as was described in section II B. Therefore, the vehicle speed as well as its undershoot or overshoot in situations of traffic disturbances, can be used as string stability performance measures. Vehicle speed undershoot (or overshoot) can be defined as the absolute difference between the lowest (or highest) vehicle speed of the last following vehicle and the (target) speed of the leading vehicle.

A. Experiment Setup

The topology that is used in all our experiments illustrated in Fig. 3. A platoon of ten vehicles is placed in a straight single lane road 5000 meters long. We use a pre-defined time headway of 0.7s and a pre-defined cruise speed of 50m/s. The distance at standstill is set to 7.7m and vehicle length is 4.46m. Furthermore, the upper limit of the vehicle's acceleration is specified to be $2m/s^2$ and the minimal acceleration is specified to be $-9m/s^2$, i.e., the deceleration does not go below $-9m/s^2$. These parameters apply to all experiments in our work except for those where we investigate the influence of different time headway values. In order to guarantee a high statistical accuracy of the obtained results, multiple runs have been performed and 90% confidence intervals have been calculated. For all performed experiments, the largest calculated confidence interval is ±3.1052 % of the shown calculated mean values.

In order to validate the controller model and traffic model, we simulate the performance of ACC without, and CACC with perfect communication, where for CACC each vehicle can always get its preceding vehicle's acceleration within SUMO without loss and delay. The results (not shown here) are very similar to the results obtained in [7]. In these baseline experiments, CACC outperforms ACC on string stability. Details of this experiment and its corresponding results and analysis can be found in [18].

1) Simulation Scenarios

The leading vehicle starts with an initial speed of 20m/s that is kept constant until t = 80s (i.e., up to the 8000th simulation time step, where one time step=10ms). During this period each following vehicle has a stable speed (no fluctuations) of 20m/s and distance between any two neighbouring vehicles is also stable. For the first scenario, at time step 8000, we let the leading vehicle decelerate with an acceleration of $-9\ m/s^2$, until the leading vehicle reaches the speed of 15 m/s. For the second scenario, at time step 8000, we let the leading vehicle accelerate with acceleration $2\ m/s^2$ until the speed of the leading vehicle reaches the value of 25 m/s. For experiments in this section, the packet loss ratio (PLR) and beacon sending frequency (BSF) are varied. The chosen values of packet loss ratio are 0%, 10%, 20%, 30%, 40%, 50% and that of beacon sending frequency are 25Hz, 20Hz, 15Hz, 10Hz, 5Hz. We also simulate the case with a default beacon sending frequency and packet loss ratio of 15Hz and 20%, and different time headway (TH) values: 2s, 1.5s, 1.0s, 0.9s, 0.8s, 0.7s, 0.6s, and 0.5s. Note that in these experiments the dropping of a packet is artificially accomplished according to a uniform distribution. Moreover, the dropping of a packet is independent from that of other packets.

In these experiments we are only observing the velocity response of the last following vehicle, because when the platoon is not string stable it is this vehicle that will experience the biggest disturbances.

2) Simulation Results and Analysis

For the first scenario, only the velocity of the last vehicle with beacon sending frequency of 10Hz is shown in Fig. 5 due to limited space and the velocity of the leading vehicle (veh0) is also shown. The simulation results of other vehicles with respect to both velocity and acceleration, and two-sided 90% confidence intervals for all the simulation results can be found in [18].

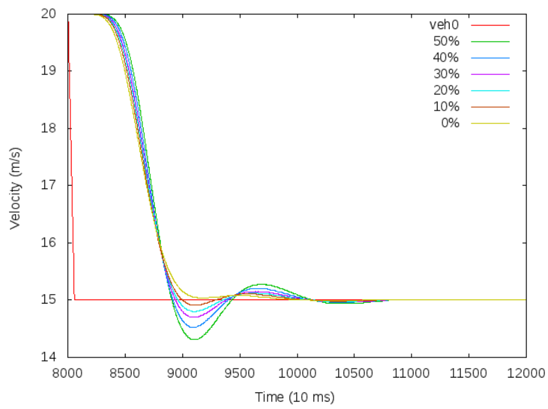


Figure 5. Velocity of veh0 and veh9, with TH=0.7s, BSF=10Hz

Note that the curves from bottom up at the 9000th time step of Fig. 5 indicate packet loss ratio in descending order. It can be seen from Fig. 5 that for a constant value of beacon sending frequency (10Hz) and time headway (0.7s), as the packet loss ratio increases, the velocity fluctuations of veh9 are increasing, which means that the disturbance of the leading vehicle is amplified more through the platoon upstream, in other words, the platoon is more string unstable.

Moreover, the undershoot of the velocity is also getting larger as the packet loss ratio increases according to Fig. 5.

The undershoot of velocity for the last vehicle is shown in Fig. 6. The undershoot is shown for different combinations of selected beacon sending frequencies and packet loss ratios.

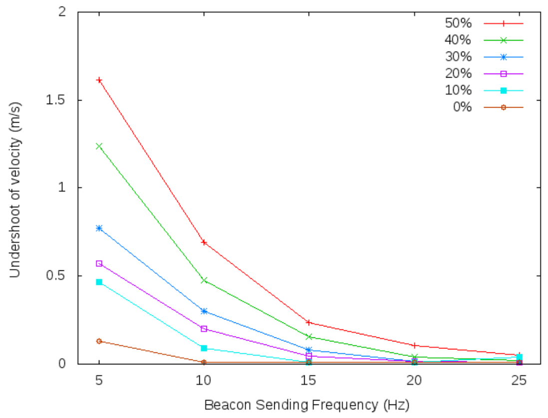


Figure 6. Undershoot for velocity of veh9, with TH=0.7s

According to Fig. 6, with a selected value of beacon sending frequency (not applicable to 25Hz) and time headway (0.7s), the undershoot of velocity for the last vehicle increases as the packet loss ratio increases, which means that the platoon becomes more string unstable. It can also be observed that for a selected value of packet loss ratio, the string stability becomes worse as the beacon sending frequency decreases. One vehicle is always using the acceleration value which is latest received from its preceding vehicle as the input of the CACC controller, therefore, a higher beacon sending frequency for preceding vehicle results in a higher possibility of receiving fresh information for a constant packet loss ratio. Besides, lower packet loss ratio can also result in a higher possibility of receiving fresh information for a constant beacon sending frequency. For a BSF of 25Hz packet loss has little effect because vehicles can still easily receive sufficiently fresh information.

With the selected values of 15 Hz for the beacon sending frequency and of 20% packet loss ratio, the velocity of the last vehicle corresponding to different time headway can be seen in Fig. 7.

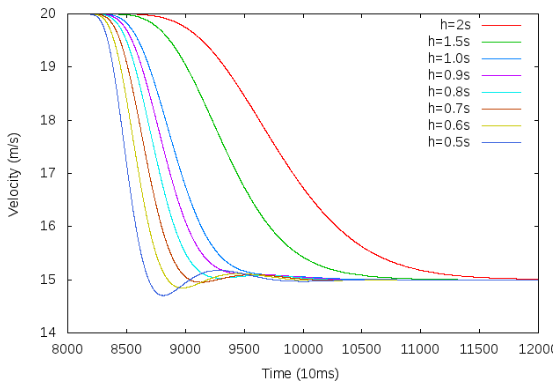


Figure 7. Velocity of veh9, with BSF=15Hz, PLR=20%

Note that the curves from left to right at a velocity of 18m/s of Fig. 7 show the headway in ascending order. It can be seen from Fig. 7 that with our selected beacon sending frequency and packet loss ratio, as time headway increases the platoon becomes more string stable, i.e., the velocity of the last vehicle decreases with less fluctuations, findings also reported in [7]. Furthermore, with larger time headways, the relative distance between vehicles is larger and when a disturbance occurs on a leading vehicle, the following vehicles do not react as fast as when small time headways are used. However this will decrease the road throughput and capacity.

Therefore, it is an important challenge to find the smallest time headway to guarantee string stability, while keeping the road capacity high.

For the second scenario, we again observe the velocity of the last vehicle. The results of the simulation are similar to the first scenario and can be found in Fig. 8, Fig.9, and Fig. 10, corresponding to Fig. 5, Fig. 6, and Fig. 7, respectively.

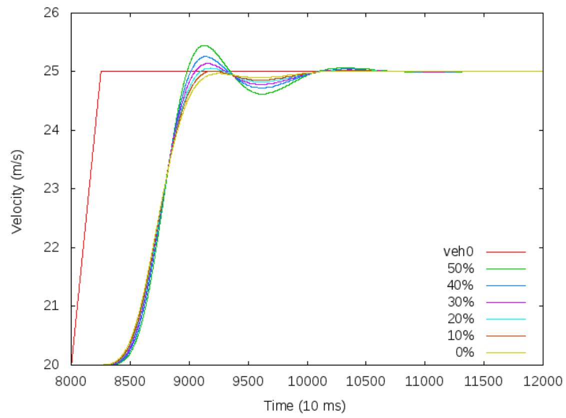


Figure 8. Velocity of veh0 and veh9, with TH=0.7s, BSF=10Hz

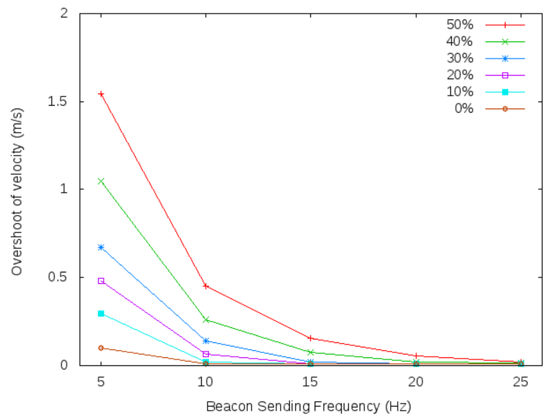


Figure 9. Overshoot for velocity of veh9, with TH=0.7s

Note that the curves from top down at the 9100th time step of Fig. 8 indicate packet loss ratio in descending order and the velocity of the leading vehicle (veh0) is also shown in Fig. 10. Different from Fig. 6, Fig. 9 depicts the “overshoot” of the velocity associated with the last vehicle. From Fig. 8 and Fig. 9 it can be seen that for a given value of beacon sending frequency and time headway, the CACC controller’s performance on string stability is decreasing with a higher packet loss ratio. Accordingly, for a given value of packet loss ratio and time headway, the string stability gets worse with a lower beacon sending frequency.

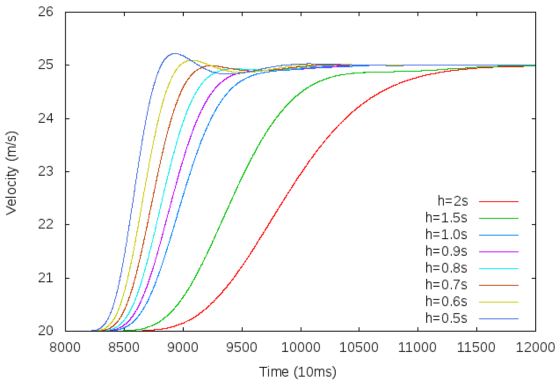


Figure 10. Velocity of veh9, with BSF=15Hz, PLR=20

Note that the curves from left to right at a velocity of 22m/s of Fig. 10 indicate time headway in ascending order. From Fig. 10, the same conclusions can be derived as the ones derived from Fig. 7. In particular, it can be observed that string stability is improving when the time headway is increased.

V. Conclusions and Future Work

In this paper, the string stability of a CACC controller in the presence of imperfect communication has been investigated. For that purpose, a simulation environment integrating time-driven controller to traffic simulations (in SIMULINK and SUMO respectively) has been combined with event-driven communication simulation (in MiXiM/OMNeT++). We observed that beacon sending frequency and packet loss ratio have significant influence on the performance of the evaluated CACC controller, i.e. lower beacon sending frequency and/or higher packet loss ratios which prevent vehicles from receiving fresh information from preceding vehicles will lower the CACC controller's performance on string stability. Therefore, given required time headway, strict requirements with respect to beacon sending frequency and packet loss ratio have to be set in order to guarantee string stability.

Regarding future work, we give the following recommendations: (1) study the impact of correlated (burst) losses; (2) study the impact of losses that are caused by real propagation problems or channel overload, instead of artificially generating these losses; (3) investigate string stability by using other performance measures.

References

[1] IEEE802.11p, "IEEE Standard for Information technology-Telecommunications and information exchange between systems-Local and metropolitan area networks-Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 6: Wireless Access in Vehicular Environments," ed, 2010.

[2] I. R. Wilmink, G. A. Klunder, and B. v. Arem, "Traffic flow effects of Integrated full-Range Speed Assistance (IRSA)," in 2007 IEEE Intelligent Vehicles Symposium, Vols 1-3, Istanbul, Turkey, 2007, pp. 1078-1084.

[3] J. Vreeswijk, B. v. Arem, K. Malong, and C. v. Driel, "Deployment scenarios for speed assistance systems," in the 15th World Congress on Intelligent Transport Systems, New York, USA, 2008.

[4] B. van Arem, C. M. J. Tampere, and K. M. Malone, "Modelling traffic flows with intelligent cars and intelligent roads," IEEE IV2003: Intelligent Vehicles Symposium, Proceedings, pp. 456-461, 2003.

[5] Automated Highway Systems: Plennum Press New York, 1997.

[6] D. Reichardt, M. Miglietta, L. Moretti, P. Morsink, and W. Schulz, "CarTALK 2000: safe and comfortable driving based upon inter-vehicle-communication," in the IEEE Intelligent Vehicle Symposium (IV'02), 2002.

[7] G. J. L. Naus, R. P. A. Vugts, J. Ploeg, M. J. G. van de Molengraft, and M. Steinbuch, "String-Stable CACC Design and Experimental Validation: A Frequency-Domain Approach," IEEE Transactions on Vehicular Technology, vol. 59, pp. 4268-4279, Nov 2010.

[8] S. Kumarawadu, Control systems : theory and implementation: Alpha Science International, 2010.

[9] Project C Project Connect&Drive, Dutch NL Agency/HTAS (High Tech Automotive Systems), Project no. HTASD08002, to be found via (URLs visited in January 2011): http://www.htas.nl/?pid=111, http://www.ctit.utwente.nl/research/projects/national/senter/connect-drive.doc/

[10] R. Pueboobpaphan and B. v. Arem, "Understanding the Relation between Driver and Vehicle Characteristics and Platoon and Traffic Flow Stability for Design and Assessment of Cooperative Adaptive Cruise Control," in Transportation Research Board (TRB) 89th Annual Meeting, Washington DC, 2010.

[11] A. Bose and P. Ioanno, "Analysis of traffic flow with mixed manual and intelligent cruise control (ICC) vehicles: Theory and experiments," California PATH Res. Rep. UCB-ITS-PRR-2001-13.

[12] D. Krajzewicz, G. Hertkorn, C. Rössel, and P. Wagner., "SUMO (Simulation of Urban MObility); An open-source traffic simulation.," presented at the 4th Middle East Symposium on Simulation and Modelling (MESM2002), Sharjah, United Arab Emirates, 2002.

[13] A. Varga, "The OMNeT++ discrete event simulation system," Modelling and Simulation 2001, pp. 319-325, 2001.

[14] UT-DACS (Design and Analysis of Communication Systems) official web site. to be found via (URLs visited in May 2011): http://www.utwente.nl/ewi/dacs/

[15] M. v. Eenennaam, G. Karagiannis, and G. Heijenk, "Towards Scalable Beaconing in VANETs," presented at the Fourth ERCIM workshop on eMobility, Luleå, Sweden:Luleå University of Technology, 2010.

[16] C. Sommer, Z. Yao, R. German, and F. Dressler, "On the Need for Bidirectional Coupling of Road Traffic Microsimulation and Network Simulation," presented at the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing (ACM Mobihoc 2008): 1st ACM SIGMOBILE International Workshop on Mobility Models for Networking Research (MobilityModels'08), Hong Kong, China, 2008.

[17] C. Sommer, R. German, and F. Dressler, "Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis," IEEE Transactions on Mobile Computing, vol. 10, pp. 3-15, Jan 2011.

[18] C. Lei., "Cooperative Adaptive Cruise Control model study based on traffic & network simulation", to be found via (URL visited in May 2011):http://www.utwente.nl/ewi/dacs/assignments/completed/bachelor/reports/2011_lei_chenxi.pdf

[19] B. van Arem, C. J. G. van Driel, and R. Visser, "The impact of cooperative adaptive cruise control on traffic-flow characteristics," IEEE Trans. Intell. Transp. Syst., vol. 7, no. 4, pp. 429–436, Dec. 2006.

[20] Simulink introduction in mathworks official website (visited in May 2011) http://www.mathworks.com/products/simulink/description1.html