Automated Merging in a Cooperative Adaptive Cruise Control (CACC) System

Wouter Klein Wolterink, Geert Heijenk, Georgios Karagiannis
University of Twente, Enschede, The Netherlands
{w.kleinwolterink, geert.heijenk, karagian}@utwente.nl

Abstract. Cooperative Adaptive Cruise Control (CACC) is a form of cruise control in which a vehicle maintains a constant headway to its preceding vehicle using radar and vehicle-to-vehicle (V2V) communication. Within the Connect & Drive\(^1\) project we have implemented and tested a prototype of such a system, with IEEE 802.11p as the enabling communication technology. In this paper we present an extension of our CACC system that allows vehicles to merge inside a platoon of vehicles at a junction, i.e., at a pre-defined location. Initially the merging vehicle and the platoon are outside each other’s communication range and are unaware of each other. Our merging algorithm is fully distributed and uses asynchronous multi-hop communication. Practical testing of our algorithm is planned for May 2011.

Keywords: automated merging, CACC, ITS, V2I, V2V

1 Introduction

Automated driving has long since been subject of research, especially when it comes to driving in platoon formation (see [1], [2], [3]). Current research generally focus on controlling the driving speed of a vehicle, thus keeping the headway to the preceding vehicle constant – steering is left to the human driver. One example of a platoon driving system is cooperative adaptive cruise control (CACC). Within the Connect & Drive\(^1\) project we have implemented and tested a prototype CACC system, see Fig. 1.

Research on merging maneuvers within platoons can be found in e.g., [2] and [4]. However, their goal was to optimize the merging procedure from the point of the merger’s benefits. Our approach focuses on the realization of a merging manoeuvre where the disturbances on the highway are minimized.

The goal of this paper is to present an extension to CACC that allows for automatic merging at a freeway junction. This extension consists of both hardware (an added road side unit, or RSU) and software (both on the RSU and the CACC vehicles). The RSU is responsible for tracking merging vehicles, estimate their arrival at the junction, and communicate this to the freeway vehicles. The CACC control algorithm has been adapted to allow for gap creation.

\(^1\) This work is supported by the Dutch NL Agency/HTAS project Connect & Drive, Project no. HTASD08002
The outline of this paper is as follows. The key points of our CACC system are highlighted in Section 2. In Section 3 an overview of the merging application is given, identifying the different parts and their roles. In Section 4 the extended CACC control algorithm is specified. We conclude this paper in Section 5.

2 Cooperative adaptive cruise control

CACC is a form of cruise control in which the speed of vehicles is automatically controlled in a cooperative matter using a front-end radar and V2V communication. Because of the short reaction time of CACC compared to human drivers, vehicles can drive relatively close together (time headway $< 1$s), forming platoons. The goals of CACC include increasing the capacity of the road network and decreasing vehicle emissions. For details about the control aspects of our CACC system see [1].

The specific CACC system considered here has been based on 802.11p. All vehicles periodically (10 Hz) transmit a one-hop broadcast packet, containing necessary vehicle information such as its location, speed, and acceleration. Based on radar input and received broadcast packets the CACC control algorithm constantly adapts its desired acceleration to keep the vehicle’s headway to its predecessor constant. The desired acceleration is the CACC’s input to the engine controller. The desired headway can be set by the driver, or can be overruled by the CACC system for safety reasons.

![Fig. 1. Four CACC operated vehicles during practical testing.](image)

3 The merging application

Figure 2 gives a sketch of the considered merging scenario. A mixed CACC/non-CACC platoon is driving along the freeway. A merging vehicle (merger for short) approaches the freeway and will join the flow of traffic at the merge area, where it is expected to arrive at about the same moment as the platoon. The merging vehicle and the platoon are initially unaware of each other. For the merger to be able to join the flow of traffic, a gap within the platoon is required that is of sufficient size for the merger to merge inside the platoon. We refer to this gap as the merging gap. This gap must be created automatically by the CACC system, i.e., without any intervention of the CACC drivers. The gap must also be properly aligned with the merger, i.e., it...
should be at about the same position as the merger when the merger reaches the merge area. When this is the case the driver of the merging vehicle will manually perform the merge manoeuvre.

![Fig. 2. The CACC merging scenario at a freeway junction.](image)

To be able to judge when the approaching non-CACC vehicle will reach the merge area we employ an RSU that is able to sense the merging vehicle and estimate (i) its arrival time at the merge area, and (ii) the size of the required merging gap in the platoon. Details on how to perform such an estimation are out the scope of this paper. This could be utilizing vehicle-to-infrastructure communications, if the merger is equipped with communication capabilities.

Having performed the estimation the RSU communicates its outcome by means of periodical 802.11p broadcasts, similar to how CACC vehicles broadcast. In this way CACC vehicles that are within reception range of the RSU are made aware of the merger’s approach. To support a larger communication range CACC vehicles include any estimation they have received directly from the RSU in their own broadcasts.

### 4 The extended CACC control algorithm

Figure 3 shows the state diagram of our extended CACC merging control algorithm, to decide whether or not a vehicle should create a merging gap. A gap is created by doubling the desired CACC headway. The goal of the algorithm is to have one vehicle create a gap, in a distributed fashion with asynchronous communication. Vehicles indicate that they are creating a gap by raising a flag in their periodical broadcast.

By default a vehicle operates in CACC mode with the default desired headway. When the vehicle receives a broadcast (either directly from the RSU or forwarded by a vehicle) that contains information about a new merging vehicle, the vehicle first checks if some other vehicle is already creating a gap for that specific merging vehicle. If not, then the vehicle estimates whether it will be inside the required merging gap. If so then it will double its desired CACC headway. It will keep this large headway until (i) someone has merged in front, (ii) the vehicle has passed the merge area, or (iii) a vehicle with a higher ID was detected creating a gap. In all cases the vehicle reverts back to CACC with default headway. Front-side merging is detected by the vehicle’s radar.
5 Conclusions

We have presented a fully distributed CACC merging application that allows for automated merging using asynchronous communication. The application uses an RSU to detect mergers and to calculate the required merging gap. The extended CACC control algorithm ensures that a single merging gap is created inside the platoon. The merger may be non-CACC operated. Currently the algorithm has been implemented and tested in Simulink (see [5]) – practical tests are planned for May 2011.

In earlier work (see [6]) we investigated the communication aspects of our CACC merging application. In a follow-up project to Connect & Drive we wish to apply our experiences, both w.r.t. communication- and control engineering aspects, to develop an improved merging application that can be deployed on a large scale.

References