Halmstad University Post-Print

Safety Considerations for Cooperating Vehicles using Wireless Communication

Kristoffer Lidström, Tony Larsson and Lars Strandén

N.B.: When citing this work, cite the original article.

©2007 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

Lidström K, Larsson T, Strandén L. Safety Considerations for Cooperating Vehicles using Wireless Communication. In: The 5th IEEE International Conference on Industrial Informatics, 2007, Piscataway, NJ : IEEE; 2007. p. 995-1000.

DOI: <http://dx.doi.org/10.1109/INDIN.2007.4384910>

Copyright: IEEE

Post-Print available at: Halmstad University DiVA

<http://urn.kb.se/resolve?urn=urn:nbn:se:hh:diva-1832>

Safety Considerations for Cooperating Vehicles using Wireless Communication

Kristoffer Lidström, Tony Larsson, and Lars Strandén

Abstract— Even though improvements in automotive safety have caused a significant decline in the number of traffic fatalities there is a strong need for further work. One important area is wireless communication from vehicle-to-vehicle and vehicle-to-infrastructure which enables a host of new cooperative traffic applications ranging from collision avoidance to intelligent cruise control. However, using cooperation between vehicles as an enabler for safety-related functionality raises new issues on system dependability. In this paper we characterize the domain of cooperating vehicles and cooperative situation awareness and suggest a system architecture that promotes independent development and verification of safety functions.

I. INTRODUCTION

OVER the last two decades the number of traffic fatalities in the European Union has almost halved [1]. Until recently passive safety systems such as seatbelts and air-bags have been the norm. With the advent of cost-efficient, compact sensors and computational nodes advanced proactive safety systems have been introduced. These include systems that warn the driver if a vehicle is in a blind spot or if the vehicle inadvertently leaves its lane. The next step in advanced proactive safety is to allow vehicles to share information about the traffic state directly with each other and with infrastructure using wireless communication.

As illustrated in Fig. 1 in-vehicle sensors, such as radar, are effective used at short-range in front of the vehicle while centralized information systems, such as FM radio bulletins, provide the driver with long-range situation awareness. The spatial requirements on medium-range, i.e. beyond line-of-sight but rapidly approaching, situation awareness rule out the sole use of in-vehicle sensors. Using radar to see around corners, behind other vehicles or over hills is simply infeasible. At the same time the latency, cost and scalability issues of centralized cell-based networks make them

inappropriate for highly dynamic medium-range situation awareness.

Inter-vehicle communication fills the void in proactive safety that exists between in-vehicle sensors and centralized wireless information sources by allowing localized low-latency communication directly between vehicles. Automotive safety applications can take advantage of vehicle-to-vehicle communication to extend the situation awareness of the driver. Several research efforts are under way in the field of inter-vehicle communication e.g. [2], [3].

II. DESCRIPTION OF THE DOMAIN

A. Objects, concepts and relations of the domain

The real world in which vehicles operate contains innumerable objects, properties of objects and relations between them. One of the main tasks of a proactive safety system is to perceive this complex world, interpret its perceptions and reason about them in order to warn the driver.

When we talk about objects we distinguish between physical objects such as cars, buses, trees and pedestrians and conceptual objects such as virtual fences (geofences) and platoons of vehicles. Both conceptual and physical objects can have properties such as size, location and speed. The definition of some conceptual objects such as platoons are in terms of physical objects and are inferable from the environment while other conceptual objects such as geofences are artificially introduced constructs.

We also distinguish between objects that can communicate and thus cooperate and those that cannot. Additionally, properties associated with objects can be either static or dynamic:

- Cooperative objects are vehicles, beacons and other objects adapted to the same standard.
- Non-cooperative objects are non standard compliant, old vehicles, pedestrians, trees etc.
- Static properties are identities, type descriptions etc.
- Dynamic properties are position, direction, speed, acceleration etc.

B. System topology

The topology of the system, as defined by the composition of different types of cooperative objects, affects properties such as scalability, robustness and cost.

Manuscript submitted January 31, 2007. This work was carried out within the Vehicle Alert System (VAS) project at Halmstad University supported in part by the Knowledge Foundation, SP Technical Research Institute of Sweden, Volvo Technology AB and Free2Move AB.

K. Lidström and T. Larsson are with the Centre for Research on Embedded Systems, Halmstad University, Halmstad, Box 823, 30118, Sweden (phone: (+46)035-167385; e-mail: {kristoffer.lidstrom, tony.larsson}@ide.hh.se).

L. Strandén, is with SP, Technical Research Institute of Sweden, Borås, Sweden. (e-mail: lars.stranden@sp.se).

Fig. 1. There are three main types of situation awareness; short-range (1), medium-range (2) and long-range (3). Communication types include vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I) and infrastructure-to-infrastructure (I2I).

There are three main types of system topology; vehicle-based, infrastructure-based and hybrid.

Purely vehicle-based networks do not require any fixed roadside infrastructure, all communication takes place between vehicles and all warnings and decisions are taken between peers within the vehicle network. The main advantage of a purely vehicle-based system is the same as for any distributed system in general such as scalability, cost and robustness. Problems arising from a vehicle-based approach are those of security and coordination, how is data verified and agreed upon in a peer-network, and of market penetration, how many vehicles must be equipped to achieve an acceptable quality of service?

Infrastructure-based systems have a high usability for early-adopters assuming that a sufficient amount of roadside units have been deployed. Security issues are also more easily handled since authorized roadside units can be assumed to be trusted. Road operators and other ITS control centers can have a more complete picture of the road network since all communication passes through deployed infrastructure. The main disadvantages are the high costs associated with deploying fixed infrastructure densely enough to achieve acceptable quality of service, especially in areas with low traffic volume.

The hybrid approach attempts to achieve a system with the advantages of both pure vehicle-based and pure infrastructure-based systems. In a hybrid system roadside units can be deployed at high-risk locations such as intersections in urban areas, while vehicle-based communication is used in areas without infrastructure. For a multi-purpose system the hybrid approach is preferred because there are application scenarios in which using only vehicle-to-vehicle communication is infeasible, such as

electronic toll collection, and other scenarios in which using only vehicle-to-infrastructure communication is infeasible, such as cooperative cruise control. The need for a hybrid topology is reflected in recent standardization efforts for vehicular communication [4].

C. Sensing the real world

Vehicles monitor their own state and can thus provide other cooperative objects with accurate information about themselves. Objects cooperative to a vehicle are categorized as *primary* if they are in direct communication range and as *secondary* if they are only in indirect communication range via a primary object. Observations about the real world are analogously either primary or secondary. We define *primary observations* to be observations that are received directly from primary objects and that only concern these primary objects monitored state (e.g., their position, direction, speed etc.). *Secondary observations* concern secondary objects and their respective state and are accessed indirectly relayed via primary objects. Thus a primary object cannot only inform other objects about its own state but can also relay what it knows about other objects in direct range. In practice the communication range limits what can be received as primary and secondary information.

As an extension and a complement to information received from cooperative objects vehicles may be equipped with sensors able to detect physical objects and properties such as, light, rain, road friction, velocity, pedestrians (cameras) or other vehicles (radar). Interpretations of primary in-vehicle sensor data are dependent on the type of sensor, e.g. feature-extraction algorithms can be used to detect pedestrians in images from

an on-board camera using a-priori knowledge about the domain, while other algorithms can identify objects based on radar signatures. However, there is always a certain degree of uncertainty associated with this type of data refinement. A low signal-to-noise ratio in the input or an incomplete model can cause a detection algorithm to fail in correctly identifying a physical object which leads to a false or incomplete view of the world.

To increase the belief in a primary observation, secondary observations can be used for corroboration purposes. We use the term belief to describe the level of certainty ascribed to observations. Belief in an observation can additionally be strengthened by further primary sensing, or by complementing the model used in the detection algorithm. However the use of secondary information as supporting evidence is one of the main advantages of vehicle-to-vehicle communication.

The types and capabilities of in-vehicle sensors vary from vehicle to vehicle, however certain capabilities should be present in each vehicle. These include the capability to measure velocity, acceleration, heading, position and time, for example through the use of systems combining GPS with other positioning techniques [5]. The basic set of measurements should form the basis of a commonly understood vocabulary for describing the real world. We refer to this collection of concepts as a *common ontology*. Higher-level composite concepts can be constructed from the basic set of concepts using defined relations, which allows for the use and exchange of specialized application-specific vocabularies. For example an application can request information about traffic jams on the intended route from other cooperative objects. If these objects do not understand the concept of a traffic jam the application can describe it in terms of core concepts such as the amount of vehicle objects within a certain geographic area all having low average speeds. The usability of the common ontology is defined by its compactness and expressiveness. From an implementation perspective the early standardization of a common ontology is an important step in ensuring system interoperability. Observations about the world, the *context*, can then be stored and exchanged with other cooperative objects in a mutually understood meaning.

D. Context and context change

By context we mean observations about the world made by objects themselves as well as observations received from other cooperative objects. An important part of the context is the subset of observations made about other objects and their present and sometimes previous state or history of states. The context seen by an object at any point in time is a snapshot of all close, fixed and mobile, real objects and their states as composed of both primary and secondary observations.

The amount of context information needed for a particular purpose depends to a great extent on the application interested in it; objects need not know all about each other. Safety related positioning information is however important, such as the objects location, speed, acceleration, and direction relative to each other. Most important is information related to neighboring objects that are close and rapidly approaching.

The contexts that vehicles close to each other see are similar and thus there may be room for inheritance and difference handling techniques to reduce the information exchange flow. However, as previously mentioned, secondary observations are important for strengthening beliefs about observations. Thus the opposite – diversity based on exchange of redundant information can be used to verify interpretations about surrounding objects whereabouts and motions. Additionally, discrepancies in the perception of objects made by different primary and secondary observers can be used as an indicator of the communication quality. When two or more objects give supporting evidence of an observation this indicates that the communication service quality is satisfactory. Discrepancies between secondary and primary observations could also indicate sensor failures or identify temporary disturbances.

Context information is valid within different space and/or time intervals and can be updated and removed accordingly. Different types of information have different priorities for different objects and thus the receiver of information should in general define the priorities. However, to avoid overloading shared communication media with less important information applications should be able to specify requirements on context updates.

To minimize and make the context information handling efficient it is divided into two kinds called *primary context* and *secondary context* respectively. The primary context is defined by the neighboring objects, and their properties, that a vehicle has direct communication with while the secondary context is defined by secondary observations. In the secondary context the number of correlating observations of each object are tracked which gives an objective ground for the belief support.

The context of an object is initially regarded as empty and from then on sampled by its own sensors and updated by messages from other cooperative objects. Information has a “best before” time stamp and since changes of context can be very frequent and complex there is a need for overload handling, pruning and prioritization.

The one-to-many nature of broadcast communication is suitable for quick dissemination of observations among groups of cooperative objects. However this means that a cooperative object will typically receive more information than it relays to other objects. The responsibility of filtering and selection of relevant information should thus lie with

the receiver.

Cooperative objects are able to exchange context information with each other. One key question is on which level context information should be shared. The limited capacity of the wireless channel and the dynamics of moving vehicles limit the amount of information that can be transferred between two objects. By performing most of the data fusion locally and communicating only highly abstracted concepts the amount of data to transfer could be kept down. However, performing data fusion locally and in isolation lowers the accuracy and reliability of the information since only a subset of all available sensors in the network is used. The level of abstraction of exchanged information is dependent on the type of cooperative strategy used.

E. Cooperation

Cooperation between objects can take many forms; we distinguish three main classes of cooperative behavior.

Informative cooperation occurs when objects simply extend each others sensory range by transmitting their own observations about the world. This type of cooperation depends mainly on one-way communication, e.g. vehicles that periodically broadcast their own position, speed and direction or fixed beacons that broadcast local information such as road temperature.

Descriptive cooperation between objects augments the basic sensor information with intentions and goals, such as the intended path or preferred speed of the vehicle. By adding intentions to the transmitted information improved decisions can be taken by the receiver. In both informative and descriptively cooperating systems each participating object analyses the context individually. Higher level coordination such as agreeing on, or committing to, certain behavior is not possible.

Coordinative cooperation occurs when communication bandwidth and time limits allow objects to reason about goals and intentions, commit to actions and alter their behavior depending on the goals and intentions of others. This type of cooperation implies two-way communication, i.e. dialogues between cooperative objects.

Different types of applications require various forms of cooperation. Factors such as available communication bandwidth, connection setup time and real-time constraints limit the type of cooperation that can be supported.

III. SAFETY

A. Recommendation failures

Several concurrent applications running in the vehicle system produce as output recommendations to the driver. Recommendations are given through some form of human-machine interface (HMI), for example visual or aural cues

or through haptic feedback devices. The nature of the recommendations is application specific, for example the level of detail and urgency in time. The given recommendations shall not violate system safety, i.e. the absence of catastrophic consequences on the users of the system [6], and this is the key to acceptance of cooperative traffic safety systems.

In general three main failure modes related to recommendations can be identified.

Erroneous recommendations are recommendations that instruct the driver to take some action that leads to a more unsafe traffic situation than if the driver had not been given any recommendations at all, this also includes correct recommendations given at the wrong time. An unsafe traffic situation is a situation where the risk of injury to any user of the system is above some acceptable, possibly dynamic, threshold. From this definition it follows that all recommendations that cause drivers of other vehicles to experience higher risk are also erroneous. A primary method to detect erroneous recommendations is to use secondary information as supporting evidence. The urgency of the recommendation limits the type of corroboration that can be used. Low-urgency notifications can be checked with n other cooperative objects by asking for second opinions which creates a type of n -modular redundancy. For high-urgency recommendations this dialogue with other objects may not be possible due to lack of time. In this case supporting information stored in the context database can be used. A-priori verification and behavioral standardization should also be used to ensure that applications behave correctly and predictably given a certain context.

A lack of recommendations can lead to a more unsafe traffic situation when compared to a situation in which correct recommendations were given. A lack of recommendations due to a component failure should be indicated so that the driver is aware that parts of the system have failed. In the case of a functioning system the methods for detecting erroneous recommendations using secondary supporting information are also applicable, e.g. inconsistencies between recommendations given to neighboring drivers as compared to those given in the own vehicle can be used as a detection mechanism.

Conflicting recommendations are multiple instructions given to the driver that are inconsistent, i.e. combinations of instructions in a short time span that contradict each other. This situation can arise when several applications provide recommendations independently of each other. Each of the conflicting recommendations can in turn be erroneous or correct. However, even in a situation where all conflicting recommendations are correct the existence of different recommendations may confuse the user and lead to an unsafe traffic situation [7]. Conflicting recommendations

Fig. 2. Architectural overview of in-vehicle system

may be due to several causes. Applications may be designed to provide the same functionality, such as avoiding collisions, but have conflicting strategies for providing that functionality. In other cases the functions themselves can be in conflict with each other, e.g. an application that minimizes fuel consumption running concurrently with another application that minimizes travel time.

A-priori identification of conflicting recommendations is infeasible due to the number of possible combinations. If dynamic application loading and unloading is allowed, a-priori verification becomes, in principle, impossible as the composite control strategy of the system as a whole becomes dynamic.

B. Resource availability

Due to resource limitations, the system may not be able to fulfill the aggregated requirements of multiple applications running concurrently which can lead to recommendation failures. Resource management in on-board units includes assigning processor time to the various system components. Applications with real-time requirements must be guaranteed execution time so that deadlines can be met. Storage space must also be reserved for applications as well as guarantees on the integrity of data stored in that space. Intentional and unintentional reading and writing in a non-reserved memory space must be prevented.

To reduce cost and installation overhead in-vehicle sensors can be shared between applications. Applications with high requirements on context perception must be guaranteed a certain sampling resolution. Management of sensors and sensor infrastructure, e.g. communication buses, must be done so that requirements on sampling rate are met.

When the system boundary is expanded to include more than one cooperative object, the wireless medium becomes an important shared resource. The total capacity available to each node is a function of the quality of the wireless channel and the number of cooperative objects that must share it. In situations where more communication resources are required than are available a coordinated effort to limit the network traffic must be made. These situations include overload situations where the number of cooperative objects is high as well as situations in which the wireless channel is deteriorated, e.g. due to environmental factors. The coordinated behavior can be seen as the ability to degrade the functionality of the system in response to the quality of the channel. The degradation of system functionality must be done so that traffic-safety related functions are prioritized i.e. a graceful degradation must be made. The specific degradation strategy is context dependent and must be determined before the deterioration of the channel reaches a point where communication is no longer possible.

IV. ARCHITECTURE

A. Overview

To address the various facets of cooperating vehicle systems mentioned in the previous sections of this paper we propose an architecture given by the block diagram depicted in Fig. 2.

The *Fusion Module* combines secondary and primary observations into beliefs about physical and conceptual objects that together form the perceived context. The output of the *Fusion Module* is described using the common ontology and stored in either the primary or secondary parts of the *Context Database*. As complementary input in-

vehicle sensors may be used.

The *Communication Module* handles communication related aspects such as medium access control (MAC) as well as choice of routing strategy. The *Communication Module* can also provide information about the state of the channel.

The *Resource Manager* guarantees that application requirements on access to in-vehicle resources, such as processing units or storage space, are met. However for shared resources outside the complete control of the *Resource Manager*, such as the wireless communication channel, only best-effort commitments can be made. To measure the quality of the wireless channel the *Resource Manager* can use hard estimates from the *Communication Module* as well as comparisons between the primary and secondary *Context Databases*. Discrepancies between primary and secondary contexts can indicate low link quality between other cooperative objects in the network, e.g. poorly connected objects generate less secondary observations. In response to degraded communication quality the *Resource Manager* can gracefully degrade the functionality of applications whose communication requirements cannot be met. Additionally the *Resource Manager* should be able to negotiate with other cooperative objects about a prioritized set of information in order to utilize the channel in the best possible way.

The *Conflict Resolver* resolves, selects and prioritizes among conflicting recommendations at runtime. The *Conflict Resolver* is responsible for evaluating the output of the various control applications and the handling of potential conflicts. Possible implementation of such a function could use a simple prioritization scheme to resolve conflicts or more sophisticated learning systems [8] that evaluate the conflicting control signals based on analysis of the consequences of choosing one or the other or of a compromise between the two.

Traffic safety applications can be implemented in many different ways and are expected to be developed independently by several manufacturers. However, applications must be able to interface to the common ontology and they must specify a *Safety Profile*. When an application is loaded into the system it must successfully register its *Safety Profile* with the *Resource Manager* before being allowed to run. The *Safety Profile* must specify all resource requirements of the application as well as provide an interface for safely degrading the functionality of the application.

B. Safety properties

Forcing application developers to explicitly state the resource requirements of their application in a *Safety Profile* prevents development faults by introducing safety thinking as a part of the development process. *Safety*

Profiles should be expressed in a format that is easily understandable to non-experts.

The *Conflict Resolver* observes all recommendations produced by applications and has the ability to override these recommendations if they are found to violate the overall safety requirements of the system which means that the safety policy of a node can be specified in a single component. The *Resource Manager* handles in-vehicle resources but most importantly also the cooperative utilization of the available communication capacity. The use of a *Conflict Resolver* and *Resource Manager* thus promotes independent development of application and safety functionality.

V. CONCLUSIONS

We have given an introduction to the domain of cooperating vehicle safety systems and defined important concepts relating to objects in the domain, information exchange, context perception and cooperation.

In relation to the safety of such systems the use of information diversity is seen as a major tool in detecting and resolving recommendation failures. Additionally, the quality of the wireless channel greatly influences the performance and safety of the system. Thus, there is a need for application-independent cooperative functions to monitor and prioritize the use of the medium in order to maximize traffic safety. From a systems-engineering viewpoint separation of safety and application functionality into explicit components promotes independent development, maintenance and verification.

REFERENCES

- [1] European Commission, Directorate General Energy and Transport, "Road Safety Evolution in the EU", <http://ec.europa.eu>, January 2007.
- [2] European Commission, 6th Framework Program Subproject, "Cooperative Vehicle-Infrastructure Systems, CVIS", <http://www.cvisproject.org>, January 2007.
- [3] European Commission, Information Society Technologies, 6th Framework Program Subproject, "SafeSpot: Cooperative Systems for Road Safety", <http://www.safespot-eu.org>, January 2007.
- [4] ISO TC204, Working Group 16, "Continuous Air interface for Long and Medium distance, CALM", <http://www.calm.hu>, April 2007.
- [5] H. Tan, and J. Huang, "DGPS-Based Vehicle-to-Vehicle Cooperative Collision Warning: Engineering Feasibility Viewpoints," in *IEEE Trans. Intelligent Transportation Systems*, vol. 7, December 2006.
- [6] A. Avizienis, J. Laprie, B. Randell, and C. Landwehr, "Basic concepts and taxonomy of dependable and secure computing", in *IEEE Trans. Dependable and Secure Computing*, vol. 1, January 2004.
- [7] L. Song, and J. K. Kuchar, "Describing, Predicting and Mitigating Dissonance Between Alerting Systems", in *Proc. 4th International Workshop on Human Error, Safety and System Development*, June 2001.
- [8] L. Strandén, "Agents in Safety Related Systems Including Ubiquitous Networks", in *Proc. 26th SGAI International Conference on Innovative Techniques and Applications of Artificial Intelligence*, December 2006.